AN99202

SCK Controller Data Out for MirrorBit® Multi I/O SPI Family

AN99202 discusses a new method of implementing a HOLD function in MirrorBit® Multi I/O SPI Family devices.

1 Introduction

Data transfers across a bus occur when the behavior of the master and slave devices resident on the bus follow a common set of known rules defined for such transfers. In addition to data transfers, rules or protocols for other operations such as reading and writing of status information or configuration data, etc. may be defined. The set of all protocols for a specific bus interface form a standard. And when standards are followed universally but not controlled by a governing body, they are known as de facto standards. In the case of the SPI bus, a de facto standard exists and is being followed by the group of devices developed by SPI flash suppliers.

The popularity of the SPI bus continues to increase, fueled by the release of Multi I/O SPI flash memory devices over the past couple of years. These devices support a superset of commands that add x2, or dual mode, and x4, or quad mode, data bus width transfers. The Multi I/O SPI devices, performing in dual-mode, can support up to the maximum read transfer bandwidth of 20 MB/s. And when performing in quad-mode, they can support up to the maximum read transfer bandwidth of 40 MB/s. These new protocols are being added to the SPI Bus de facto standard as they become supported by the SPI flash suppliers.

The new protocols for dual mode reads use the SI and the SO signals as two-bit parallel data bus bits. And the new protocols for quad mode use the SI, SO, HOLD#, and WP# as four-bit parallel data bus bits. Because the new quad-mode protocols map the HOLD# signal to IO4, a new method of supporting a HOLD function is now needed. A new method of implementing a HOLD function is defined within this Application Note.

2 HOLD Functionality

The HOLD functionality essentially informs the flash memory that the processor wants to prevent data overruns by disabling additional data transfers, i.e. holding off the flash.

In single mode and dual mode command execution, the HOLD# signal is left intact and is pulled low (active) to perform a data transfer Hold to the SPI flash.

However, in quad mode command execution, the HOLD# signal is mapped for use as IO4. It cannot be used to hold off data transfers. A new method is required to be defined.

3 A New HOLD Function

The SPI flash memory device is always a slave device on the SPI bus. The master is the external circuitry that drives the SPI interface, and can be either a processor or a set of other logic. The master controls the transfer of data on the bus. If the HOLD# signal is being used for data transfer in a quad mode command, another method is to disable the flow of data transfer by preventing the SCK signal from toggling. When the SCK is held low or high, the SPI flash will stop clocking new data on the IO[3:0] bus. When SCK resumes toggling, the SPI flash will commence putting valid data onto the data bus, but only if the setup and hold parameter specifications are met on the SPI bus. Otherwise incorrect data could be transferred. It is also important to maintain clean SCK and a CS# signals; otherwise this could lead to resetting the read sequence.

While this method of disabling SCK is required to prevent data overruns in quad mode reads, it is optional when executing a serial or dual command. In other words you could still use the HOLD# signal.

The timing diagrams in Figure 1 and Figure 2 show two valid examples for holding data output with the Cypress SPI SCK signal (Low State or High State).

3.1 Valid Examples for Holding Data Output (Low State and High State)

Figure 1. Holding Data Output with SPI SCK Low

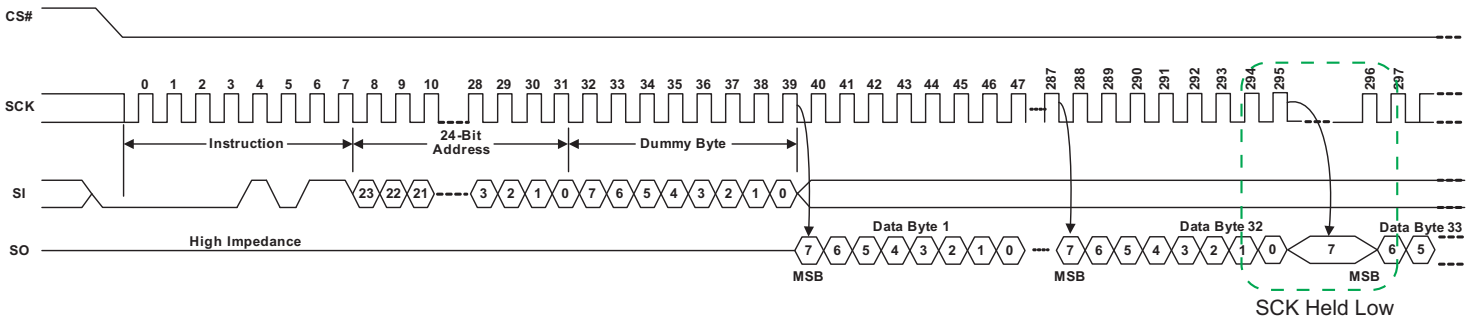
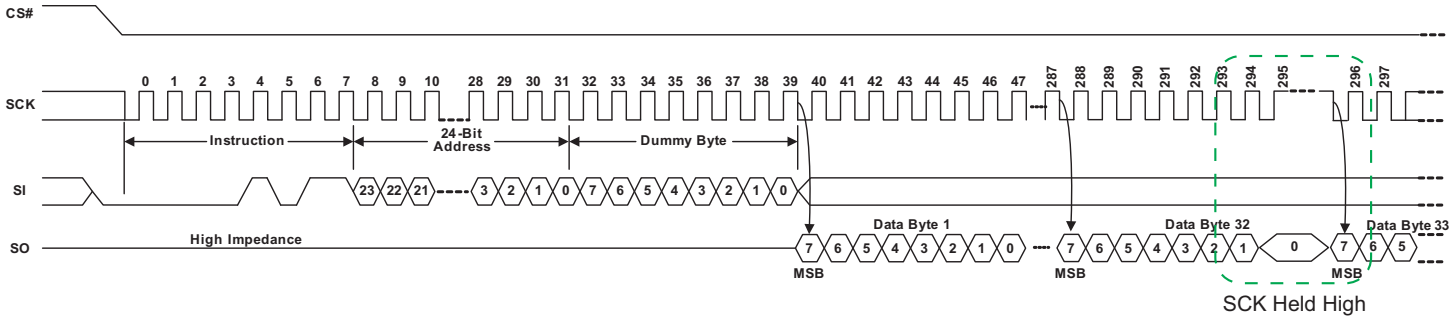


Figure 2. Holding Data Output with SPI SCK High



4 Summary

When the HOLD# signal is used in quad-mode Multi I/O SPI Support, the usual method for commanding the flash device to pause is no longer available. An alternative way to inhibit the transfer of data is to disable the SCK toggling. When doing so, it is important to maintain adherence to the AC Timing Parameter specification, especially for Tsetup and THold. Since SCK is an input to the SPI flash device, the disabling of the SCK signal will usually be handled by the processor, but may be performed through external methods.

Document History Page

Document Title: AN99202 - SCK Controller Data Out for MirrorBit® Multi I/O SPI Family Document Number: 001-99202				
Rev.	**ECN No.**	**Orig. of Change**	**Submission Date**	**Description of Change**
**	–	–	08/19/2010	Initial version
*A	4928104	MSWI	10/13/2015	Updated in Cypress template
*B	5867704	AESATMP8	08/30/2017	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