

Introduction

The serial peripheral interface (SPI) is a widely used, 4-wire, serial communication interface. Applications such as digital audio, digital signal processing, and telecommunication channels require high-speed data streams. The low-power, high-speed Altera® MAX® II CPLD is perfectly suited for an SPI master, external to the host. This application note details the implementation of the SPI master in a MAX II CPLD. A microprocessor is used to control the master, which you can use to select a slave device to read and write data to and from it.

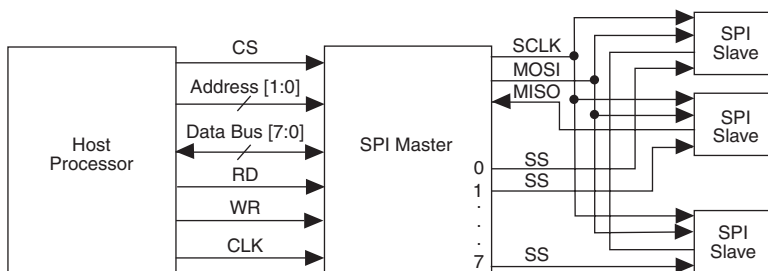
Serial Peripheral Interface

SPI is an industry standard protocol that is widely used in embedded systems for interfacing microprocessors and various devices such as sensors, memory chips, shift registers, port expanders, display drivers, data converters, printers, data storage devices, sensors, and multimedia cards. This interface standard has several advantages:

- Low pin count and simple wiring
- Full-duplex communication for higher throughput (allowing faster communication compared to other protocols such as I²C)
- No addressing; therefore, reduced overhead

The CPLD uses four signals to interact with an SPI slave. MISO is used to read from the slave; MOSI is used to write to the slave. The host processor controls the CPLD with the signals, as shown in [Figure 1](#).

Figure 1. Implementing an SPI Master Using a MAX II CPLD



SPI Master on a MAX II CPLD

Figure 1 shows the implementation of the SPI master. The various CPLD interfaces are explained in the following sections.

SPI Interface

The CPLD communicates with the slave devices using one data-out port (MOSI) and one data-in port (MISO) along with a clock (SCLK) and slave select signal (SS). Slave select is used to select the device with which the SPI master needs to communicate. A maximum of eight devices can be connected to the SPI master. The pin details for the interface are summarized in Table 1.

Table 1. SPI Interface Pin Descriptions <i>Note (1)</i>		
Signal	Purpose	Description
MOSI	master-out slave-in	Outputs data from the master to the inputs of the slaves.
MISO	master-in slave-out	Inputs data from the master to the outputs of the slaves.
SCLK	master-in slave-out	Transfers data from the slave to the input of the master.
SCLK	SPI clock	Clock driven by the master to the slaves. Used to synchronize the data bits.
SS	slave select	Select signal (active low) is driven by the master and sent to the individual slaves, and is used to select the target slave.

Note to Table 1:

(1) SPI clock = host clock / (CLK DIV + 2).

Host Interface

The host processor interface consists of an 8-bit bi-directional data bus (data bus [7:0]), a 2-bit address bus (address [1:0]), one chip select, one read flag, one write flag, and a clock signal. These signals form the host interface of the CPLD (the SPI master). Table 2 provides a description of the various signals.

Table 2. Host Interface Pin Description (Part 1 of 2)	
Signal	Description
Chip select (CS)	When high, the host-to-SPI-master interface is enabled.
Address bus (address [1:0])	To select the SPI master's register with the corresponding address.

Signal	Description
Data bus (data bus [7:0])	Bi-directional data bus connecting the host to the SPI master.
Read (RD)	When high, the host reads the SPI register whose address is being transmitted on the address bus.
Write (WR)	When high, the host writes into the SPI register whose address is being transmitted on the address bus.
Clock (CLK)	Host clock signal.

The host selects the SPI master by holding the CS signal high. When CS is high, the address bus bits determine to which SPI register the host is pointing. The host reads a particular register by keeping RD high and writes into a register by keeping WR high. After completing the transmission of an SPI word, the interrupt flag in the status register goes high. The host frequently monitors the status registers. [Table 3](#) shows the SPI registers.

Address	Register	Width
00	Control register	8 bit
01	Status register	8 bit
10	Transfer register	8 bit
11	Receiver register	8 bit

[Figure 2](#) shows the control register map and the status register map.

Figure 2. Control and Status Register Maps

Control Register Map

7	6	5	4	3	2	1	0
SS	SS	SS	CPOL	CPHA	CLK DIV	CLK DIV	CLK DIV

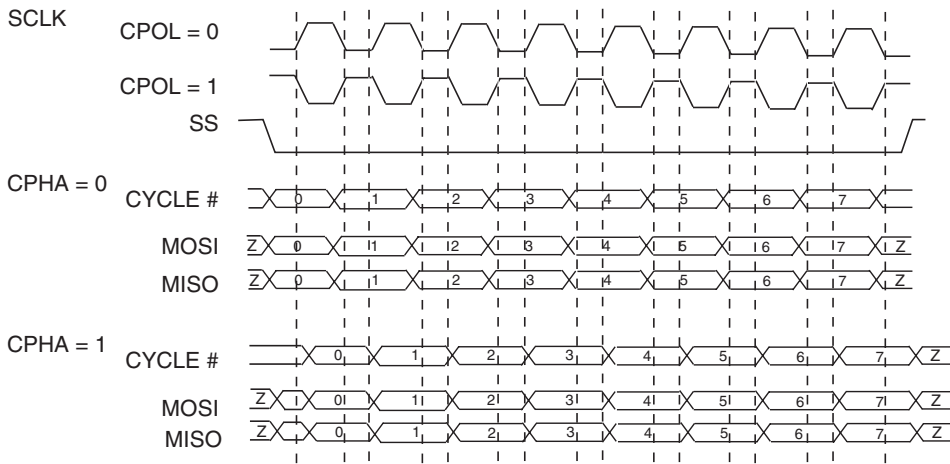
Status Register Map

7	6	5	4	3	2	1	0
X	X	X	X	X	X	X	INT

When an interrupt is detected, the host reads the receive buffer, writes a new command word, if required, and updates the transmit buffer. When the transmit buffer is updated, the interrupt flag in the status register is reset and a new SPI word is sent. The receive buffer is updated by the data received in this SPI word and an interrupt is again generated at the end of the word. The SPI slave select, clock polarity, clock phase, and SPI clock frequency can be controlled by the control register setting, as shown in Figure 2.

Typical SPI timing characteristics are shown in Figure 3.

Figure 3. Typical SPI Timing Characteristics



Implementation

This design may be implemented using an EPM240 or any other MAX II CPLD. The design source code is compiled and can be programmed into a MAX II CPLD. Host interfacing ports and SPI slave interfacing ports are shown in Figure 1.

Source Code

This design has been implemented in Verilog. The source code, testbench, and complete Quartus® II project are available at:

www.altera.com/literature/an/an485_design_example.zip

Conclusion

As shown in this design, MAX II CPLDs are an excellent choice to implement industry standard interface controllers such as the SPI master. Their low power and cost, their ability to tolerate any possible power-on sequence, and their multi-volt features make them ideally suited for implementing interface controllers such as the SPI master.

Additional Resources

The following are additional resources for this application note:

- MAX II CPLD home page:
<http://www.altera.com/products/devices/cpld/max2/mx2-index.jsp>
- MAX II Device Literature page:
<http://www.altera.com/literature/lit-max2.jsp>
- MAX II Power-Down Designs:
<http://www.altera.com/support/examples/max/exm-power-down.html>
- MAX II Application Notes:
AN 428: MAX II CPLD Design Guidelines
AN 422: Power Management in Portable Systems Using MAX II CPLDs

Document Revision History

Table 4 shows the revision history for this application note.

Date and Document Version	Changes Made	Summary of Changes
December 2007, v1.0	Initial release.	—



101 Innovation Drive
San Jose, CA 95134
www.altera.com
Technical Support:
www.altera.com/support
Literature Services:
literature@altera.com

Copyright © 2007 Altera Corporation. All rights reserved. Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. All other product or service names are the property of their respective holders. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

