Serial Peripheral Interface Master in Altera MAX Series

2014.09.22

AN-485 Subscribe Send Feedback

The serial peripheral interface (SPI) is a 4-wire, serial communication 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 I^2C)
- No addressing; therefore, reduced overhead

Applications such as digital audio, digital signal processing, and telecommunication channels require high-speed data streams. The low-power, high-speed Altera® MAX® II, MAX V and MAX 10 devices are suitable for an SPI master, external to the host. This application note details the implementation of the SPI master in MAX II, MAX V and MAX 10 devices. 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.

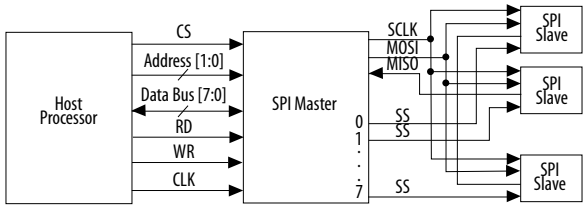
Related Information

- **Power Management in Portable Systems using Altera Devices**
  Provides more information about power management in portable systems using Altera devices
- **MAX II Device Design Guidelines**
  Provides more information about MAX II device design guidelines
- **Design Example for MAX II**
  Provides the MAX II design files for this application note (AN-485)
- **Design Example for MAX 10**
  Provides the MAX 10 design files for this application note (AN-485)

Using SPI Master in Altera Devices

Figure 1: Implementing an SPI Master using Supported Altera Devices

This figure shows the host interfacing ports and SPI slave interfacing ports. The host processor controls the CPLD with the signals.



SPI Interface

The device communicates with the slave devices using:

- one data-out port (MOSI)
- one data-in port (MISO)
- clock (SCLK)
- slave select signal (SS)

Table 1: SPI Interface Pin Descriptions

This table lists the pin details for the interface.

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	SPI clock	Clock driven by the master to the slaves. Synchronizes the data bits.
SS	slave select	Select signal (active low) is driven by the master and sent to the individual slaves. The SS signal selects the device (target slave) with which the SPI master needs to communicate. You can connect up to eight devices to the SPI master.

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

Host Interface

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 2: Host Interface Pin Description

This table lists the host processor interface signals. These signals form the host interface of the device (the SPI master).

Signal	Description
Chip select (CS)	When high, the host-to-SPI-master interface is enabled.
Address bus (address [1:0])	A 2-bit address bus that selects the SPI master's register with the corresponding address.
Data bus (data bus [7:0])	An 8-bit bidirectional 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.

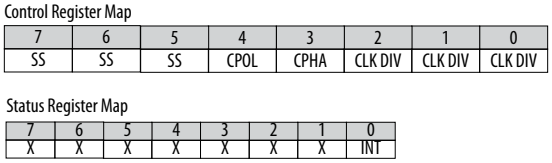
Table 3: SPI Address

This table lists the SPI registers.

Address	Register	Width (Bit)
00	Control register	8
01	Status register	8
10	Transfer register	8
11	Receiver register	8

Figure 2: Control and Status Register Maps

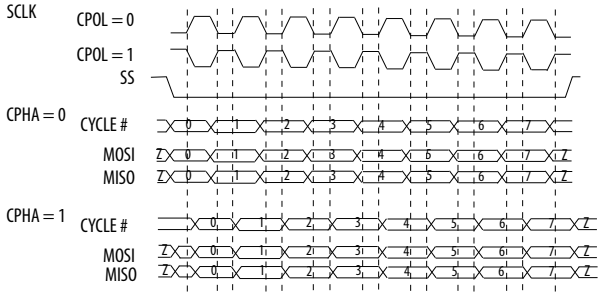
This figure shows the control register map and the status register map.



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.

Figure 3: Typical SPI Timing Characteristics

This figure shows the typical SPI timing characteristics.



Implementation

These designs may be implemented using MAX II, MAX V, and MAX 10 devices. The provided design source codes target the MAX II (EPM240) and MAX 10 (10M08) respectively. These design source codes are compiled and can be programmed directly to the MAX devices.

Source Code

These designs are implemented in Verilog.

Acknowledgments

Design example adapted for Altera MAX 10 FPGAs by:

Orchid Technologies Engineering and Consulting, Inc.

Maynard, Massachusetts 01754

TEL: 978-461-2000

WEB: **www.orchid-tech.com**

EMAIL: **info@orchid-tech.com**

Document Revision History

Table 4: Document Revision History

Date	Version	Changes
September 2014	2014.09.22	Added MAX 10 information.
December 2007	v1.0	Initial release.