UART-to-SPI Interface - Design Example

Table of Contents

Overview

This document provides design details and usage for the universal asynchronous receiver/transmitter (UART) to serial peripheral interface (SPI). The UART-to-SPI interface can be used to communicate to SPI slave devices from a PC with a UART port. SPI is a full duplex, serial bus commonly used in the embedded world because of its simple hardware interface requirements and protocol flexibility. SPI devices are normally smaller in size (low I/O count) when compared to parallel interface devices. This design example is implemented on an Microsemi ProASIC®3 device, but can easily be implemented in any of low-power field programmable gate arrays (FPGAs) by Microsemi to optimize system power, size, or performance requirements.

Design files for this design example can be downloaded from Microsemi website:

www.microsemi.com/soc/download/rsc/?f=UART_to_SPI_DF.

Design Description

The top-level block diagram of the design is shown in Figure 1.

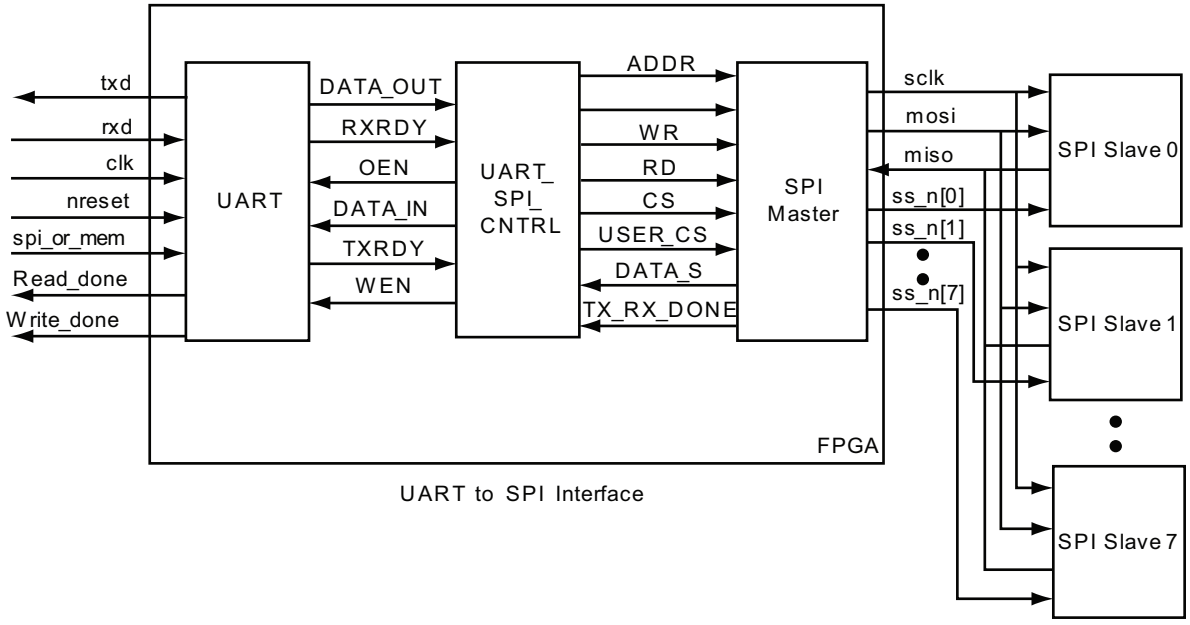


***Figure 1* • Top-Level Block Diagram**

This design example consists of three blocks: the UART interface, UART-to-SPI control block, and SPI master interface.

The UART interface is achieved using CoreUART by Microsemi. This block handles the data at the UART end. Refer to the *CoreUART Handbook* (www.microsemi.com/soc/ipdocs/CoreUART_HB.pdf) for more information.

The internal UART-to-SPI control blocks stitches the CoreUART and SPI master.

The SPI master block generates the control signals to interface to external slave devices. This interface communicates with the slave devices using the serial data out port (MOSI), serial data in port (MISO), output clock (SCLK), and slave select ports (SS_N [7:0]).

There are three internal registers in the design: control register, transmit register, and receive register. The control register sets the different control bits, the transmit register sends the TX data to the SPI bus, and the receive register receives the Rx data from the SPI bus.

After every reset, data received from the external UART go to the control register. The control bit positions are given in Table 1 and described in Table 2 on page 3.

***Table 1* • Control Bit Position**

7	6	5	4	3	2	1	0
SS			CPOL	CPHA	CLKDIV		

Table 2 • **Control Bit Description**

Control Register Bit	Function	Description
CLKDIV	Clock divider bits	These bits determine the frequency of SCK by selecting a quotient of the system clock. 000 – SCK is 1/4 of the system clock 001 – SCK is 1/8 of the system clock 010 – SCK is 1/16 of the system clock 011 – SCK is 1/32 of the system clock 100 – SCK is 1/64 of the system clock 101 – SCK is 1/128 of the system clock 110 – SCK is 1/256 of the system clock 111 – SCK is 1/512 of the system clock
CPHA	Clock phase bit	This bit determines the clock phase of SCK in relationship to the serial data. 0 – Data is valid on first SCK edge (rising or falling) after slave select has asserted. 1 – Data is valid on second SCK edge (rising or falling) after slave select has asserted.
CPOL	Clock polarity bit	This bit determines the polarity of SCK. 0 – SCK is Low when idle. 1 – SCK is High when idle.
SS	Slave select	These bits determine the selection of slave devices when USER_CS = '0' or CS = '0'. 000 – SS_N[0] is asserted 001 – SS_N[1] is asserted 010 – SS_N[2] is asserted 011 – SS_N[3] is asserted 100 – SS_N[4] is asserted 101 – SS_N[5] is asserted 110 – SS_N[6] is asserted 111 – SS_N[7] is asserted

When the UART-to-SPI communicates to any of the slave devices, it enables only the corresponding slave select signal. Only one slave device should be transmitting data during a particular data transfer. Slave devices that are not selected do not interfere with SPI bus activities during that period. Other slave devices ignore the clock signal and keep the MISO output pin in a high impedance state, unless the slave select pin is enabled.

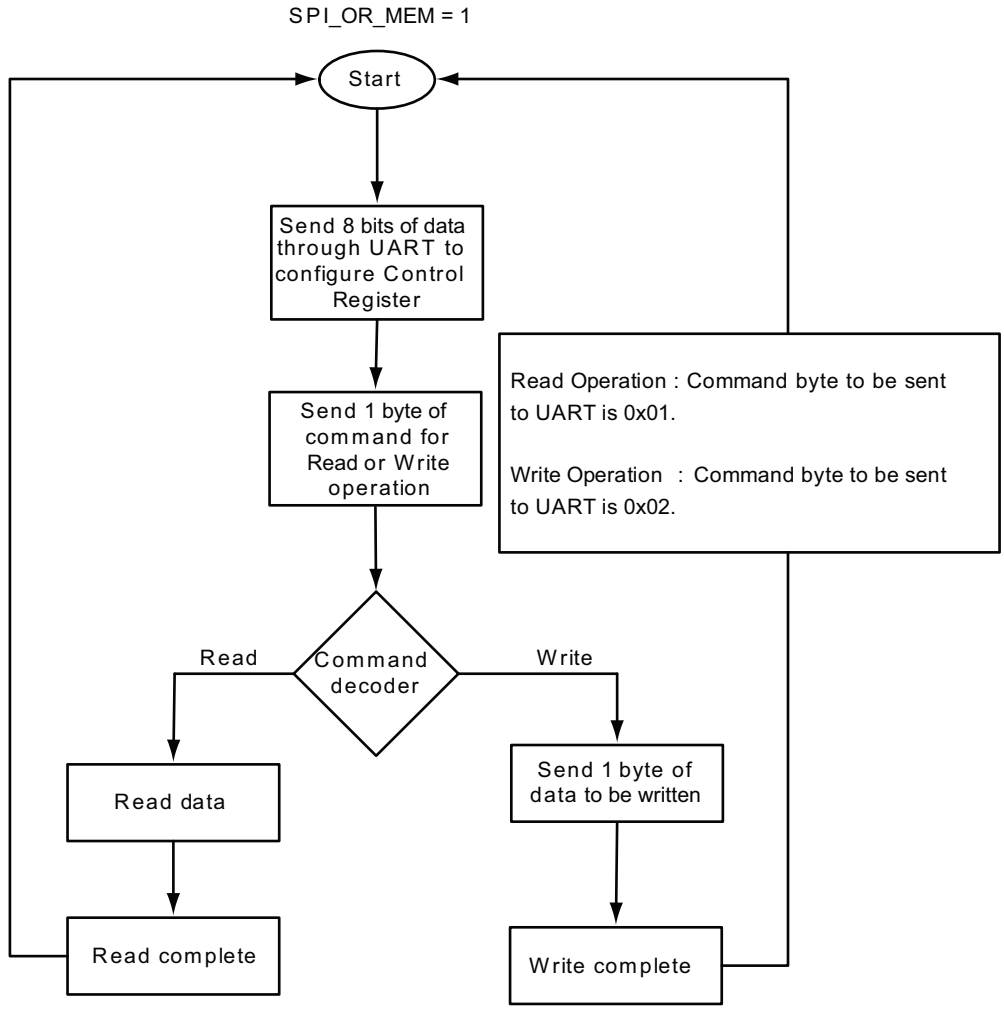
The SPI_OR_MEM hardcoded value sets the operation mode: when SPI_OR_MEM is set to 1, the slave select signal SS_Nx will be asserted Low for a 1-byte (8 bit) transaction only; when SPI_OR_MEM is set to 0, the SPI slave device will be treated as a SPI memory, and the SS_Nx signal can be asserted Low for multiple bytes of data. This mode is required when performing the page/sector mode of operations with memories. The slave select will be Low for the command byte, address bytes, and data bytes.

When SPI_OR_MEM is set to 1 (Table 3), the command byte 0x01 is used for read operation and the command byte 0x02 is used for write operation.

Table 3 • **Read/Write Selection when SPI_OR_MEM is 1**

Operation	Description
Read	0x01 command byte is sent over UART Tx, enabling data read from the UART Rx line.
Write	0x02 command byte is sent over UART Tx, followed by the data to be written.

Figure 2 shows the flowchart for SPI_OR_MEM set to 1. Figure 3 on page 5 shows the flowchart for SPI_OR_MEM set to 0. The command definitions are shown in Table 4 on page 5.



Soon after read process completes, read back data will be available at TX pin of the UART.

Figure 2 • **SPI Read and Write Cycle Mode 1**

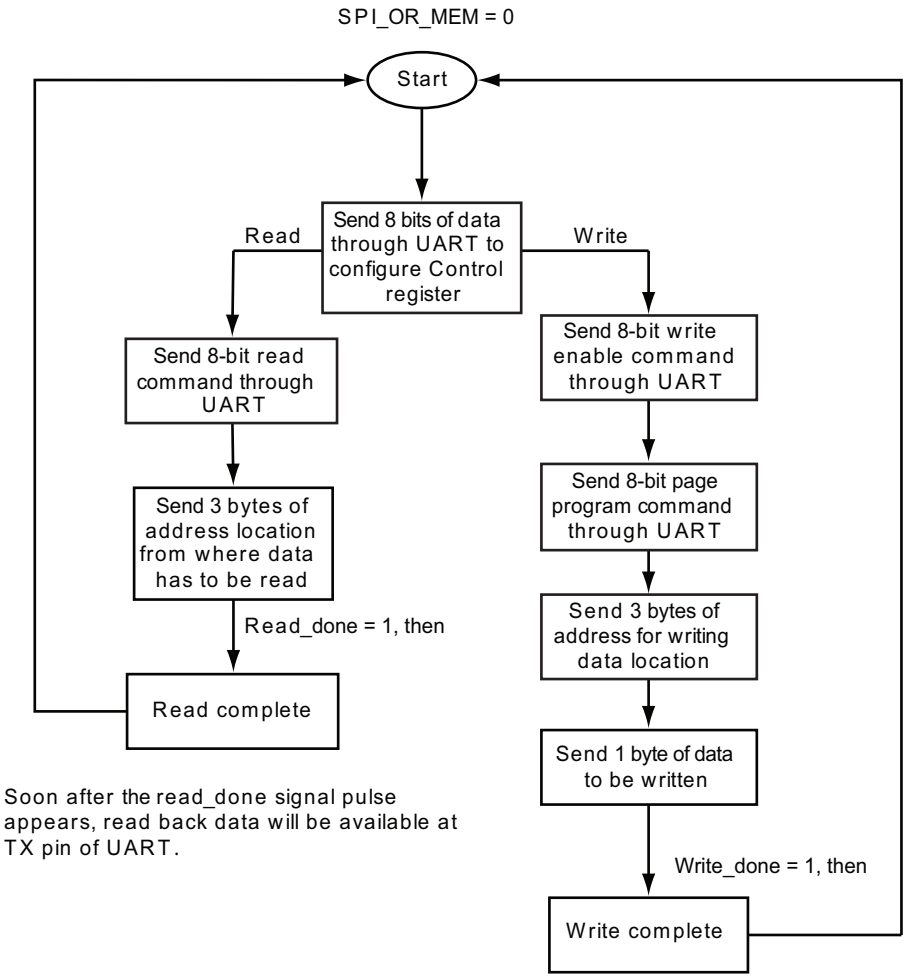


***Figure 3* • SPI Read and Write Cycle Mode 0**

***Table 4* • Command Definitions**

Operation	Command	Description	One-Byte Command Code
Write Control	WREN	Write enable	06H (0000 0110)
	WRDI	Write disable	04H (0000 0100)
Program	PP	Page program	02H (0000 0010)
Read	READ	Read data bytes	03H (0000 0011)
	FAST_READ	Read data bytes at higher speed	0BH (0000 1011)
Status Register	RDSR	Read from status register	05H (0000 0101)
	WRSR	Write to status register	01H (0000 0001)

Interface Description

The interface details of the IP are given in Table 5.

Table 5 • **Port Descriptions**

Port	Direction	Description
NRESET	Input	Active Low reset signal aborts current operation and resets all the control signals.
CLK	Input	Input clock of 20 MHz
RXD	Input	UART serial input
TXD	Output	UART serial output
SPI_OR_MEM	Input	This bit selects single-byte or multiple-byte slave select. If SPI_OR_MEM = 1, the slave select will be enabled for single byte. If SPI_OR_MEM = 0, the slave select will be enabled for multiple bytes.
SCLK	Output	Output clock to the SPI slave. Frequency depends upon the control bit settings.
MOSI	Output	Master out slave in
MISO	Input	Master in slave out
SS_N [7:0]	Output	Active Low slave select output signal
Write_done	Output	Output signal. Pulse High for one CLK cycle after completion of write cycle.
Read_done	Output	Output signal. Pulse High for one CLK cycle after completion of read cycle. Once the read_done signal is High, read back data will be available at TX pin of UART.

Utilization Details

This design was verified in M1A3P1000-484FBGA ProASIC3 device by Microsemi, but can easily be instantiated in other ProASIC3 and IGLOO® devices that contain the minimum required resources. The utilization details for the M1A3P1000-484FBGA are given in Table 6.

Table 6 • **Utilization Details**

Resource	Used	Total	Percentage
Core	575	24,576	2.34 %
I/Os	13	300	3.67 %
Differential I/Os	0	74	0.00 %
Global (chip + quadrant)	2	18	11.11 %
PLL	1	1	100.00 %
RAM/FIFO	0	32	0.00 %
Low Static ICC	0	1	0.00 %
FlashROM	0	1	0.00 %
User JTAG	0	1	0.00 %

Testing Scheme

Verification of the core is done by simulation in Model*Sim*®. In the simulation environment, a phase-locked loop (PLL) is used to create the 20 MHz input clock and passed to the UART-to-SPI interface. The simulation model of an S25FL016A from SPANSION® is used as the SPI device. Timing simulation for the UART-to-SPI interface is performed, taking into consideration the timing parameters of the Winbond SpiFlash® device, W25X16 - VSSIG.

The page program cycle waveform is shown in Figure 4.

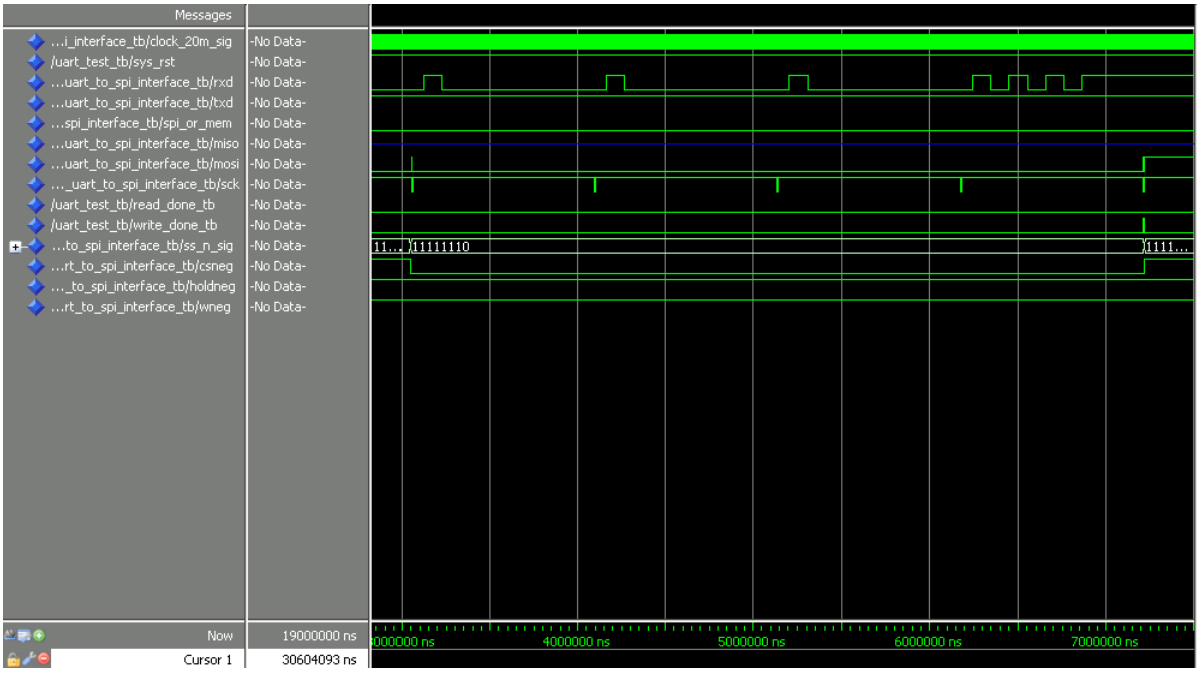


***Figure 4* • Page Program Cycle**

The page program command waveform is shown in Figure 5.

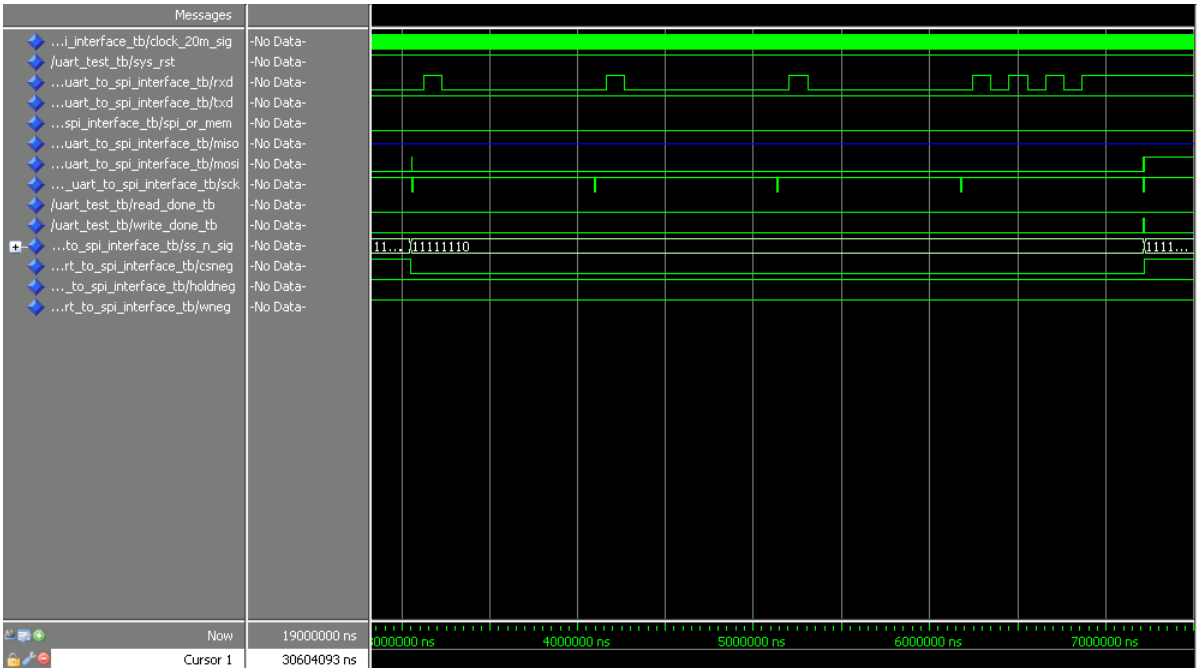


***Figure 5* • Page Program Command**

The read cycle waveform is shown in Figure 6.

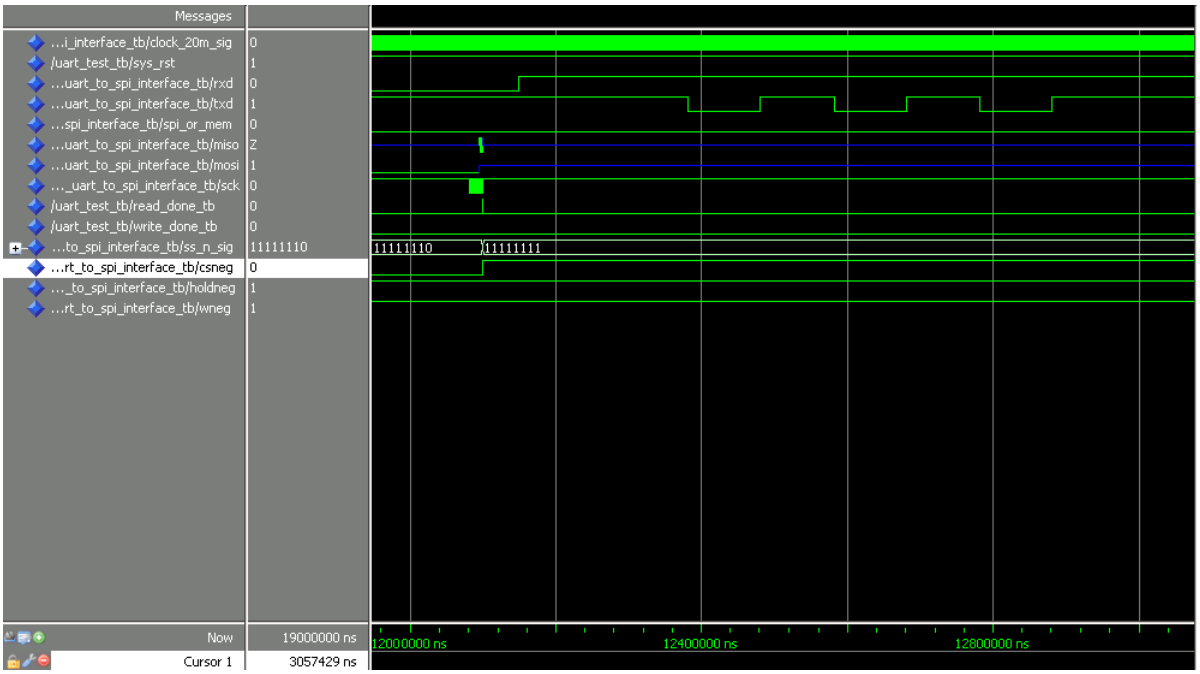


***Figure 6* • Read Cycle**

Application Area

The UART-to-SPI IP can fit in any application where an SPI device has to be used. Typical applications include interfacing of EEPROM, flash memories, and sensors.

Conclusion

The portable electronics market requires devices that are small and highly efficient. The UART-to-SPI core described in this document reduces the real estate required on the board, saving both cost and board space, while providing a nearly universal high-performance serial interface.

Microsemi Corporate Headquarters
One Enterprise, Aliso Viejo CA 92656 USA
Within the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996

Microsemi Corporation (NASDAQ: MSCC) offers a comprehensive portfolio of semiconductor solutions for: aerospace, defense and security; enterprise and communications; and industrial and alternative energy markets. Products include high-performance, high-reliability analog and RF devices, mixed signal and RF integrated circuits, customizable SoCs, FPGAs, and complete subsystems. Microsemi is headquartered in Aliso Viejo, Calif. Learn more at **www.microsemi.com**.

51900195-0/10.12