ECE 491

Laboratory 4 - RS-232 Serial Data Transmission

September 21, 2007

Goals

- To learn about the details of asynchronous data communication.
- To develop a simple-register transfer level design in Verilog that implements RS-232 asynchronous data transmission.
- To build a testbench that verifies the operation of your design.

Requirements

You will develop an FPGA-based Asynchronous Serial Transmitter that meets the following requirements:

- The transmitter will include an 8-bit data input, a 1-bit start input, and 1-bit outputs txd and rdy, as shown below. When start is asserted, the data input should be sampled and transmitted one bit at a time beginning with a "start bit" and finishing with a single "stop bit" (see the "Background" section).
- 2. The transmitter design must be parameterized so that it can be configured to operate at different Baud rates (9600 baud default).
- 3. The design must be verified using a self-checking testbench. This tesbench must test the transmission of several different byte values, including 8 'b01010101, 8 'b00110011, and 8 'b00001111. In addition, it must check that transmission occurs correctly both when successive bytes transmit immediately and when they are separated by several clock cycles. Your testbench will use the \$display system task to print out any errors which are found. In addition, it will print a message at the end indicating that it completed and then pause simulation using the \$stop system task.
- 4. All Verilog code must follow the Coding Guidelines discussed in class.



Deliverables

- 1. Demonstration to Lab Instructor of successfully operating design transmitting characters to a PC running HyperTerminal.
- 2. A short technical memorandum which describes (a) what was done, (b) what you learned, and (c) what difficulties you encountered. Include a block diagram of your design showing its major components.

- 3. Verilog listings of all files, including transmitter design and testbench. Timing diagram printouts showing correct simulation of your transmitter design.
- 4. Transcript printout showing correct operation of your clock divider testbench.
- 5. Entries in your Lab Notebook documenting design ideas, the steps taken to create and debug your design, any pitfalls you encountered, and recorded data (if any). Each Lab Notebook entry should be dated and signed by all students in the lab group.

Background

Asynchronous serial data transmission protocols such as RS-232 involve sending data serially at a known transmission rate called the baud rate. The RS-232 protocol is used to transmit data between computers and terminals. Data is transmitted in the following form:



When no characters are being transmitted, the DATA signal remains high. When a character is transmitted, it is sent one bit at a time at the baud rate but is preceded by a start bit that is always asserted low. In addition, the data bits are followed by a stop bit that is always asserted high. After the complete transmission of a character and its start and stop bits, additional characters may be sent immediately if desired following the same protocol. If an additional character is not sent immediately after the stop bit, the data signal remains high until a new character is sent an indefinite time later. The protocol is asynchronous because the transmitter sending the data and the receiver that receives the data operate using different clocks (they may be separated by a large distance, so distributing the clock is impractical).

An asynchronous serial transmitter is simple to implement using a 10-bit shift register and a little extra hardware. When a character is ready to be transmitted, it is loaded in parallel into the shift register along with start and stop bits. Data is then shifted out at the baud rate until the shift register has a value of zero (since the stop bit is always 1, the shift register will not be zero until the stop bit is shifted out of the shift register).

The Spartan 3 Starter Kit board includes an RS-232 interface that can be connected to a cable to a PC running HyperTerminal for testing. The RS-232 ports are defined in the top-level "s3board.v" file as input port "**rxd_in**" and output port "**txd_out**". Use the "txd_out" port to test your Transmitter Design.