Lab Exercise #7

Objective

The overall objective for this lab is to further the understand the VHDL language and how to use it to describe synthesizable time dependant logic circuits. The specific goal in this lab is to design a parallel-to-serial transmitter circuit to send data serially and a serial-to-parallel receiver circuit to capture data that was sent serially.

References

Example VHDL circuit descriptions in the Chu textbook.
Example VHDL shown in the Xilinx XST synthesizer user manual - chapter 3 in particular

Problem Statement

1) Design an 8-bit serial data transmitter and implement on a WWU FPGA3 board. Assume a three signal interface will be used to transfer data to a receiving circuit: a data line, data clock, and data start signal (see figure 1). The transmitter will have an 8-bit data input (sw7 to sw0) , a Go signal (sw11), and system clock. When the Go signal is asserted, a byte will be read from sw7-0 and placed into transmit memory, and then transmission will begin. Bit 0 will be transmitted first, Bit 1 next, etc. Transmitter outputs will be as follows:

   data clock - extout0
   start signal - extout1
   data line - extout2

   The data clock rate will be 1.0 MHz (although for debugging it may be helpful to use a slower clock).

2) Design an 8-bit serial data receiver for the three wire interface and implement. Assign incoming data to these ports:

   data clock - extin0
   start signal - extin1
   data line - extin2

   Received data is to be displayed on 8 LEDs, LED 7-0. Recall that data bit reception will be "clocked" by the transmitted data clock, not by a clock local to the reciever..

3) Use small state machines to control transmission of data and receiving of data.

Use a scope or logic analyzer to display the bit clock, data start signal, and serial bit stream simultaneously.

A general approach to transmitting serial data is to use a shift register that is parallel loaded with the byte to be transmitted. Then shift with the LSB becoming the bit to transmit. On the receiving end, use a shift register with bits coming in one at a time. Use a state machine to control the shift register.

Design Flow

The general flow for today's lab is:

- FIRST draw a block diagram showing circuit functional blocks..
- Figure out timing and sequence of needed signals.
- Describe the design with VHDL statements
- Synthesize, place, route, and create a bit map file using Xilinx ISE software
- Download to FPGA board
- Verify operation.
- Iterate as required

Data Transmission Format

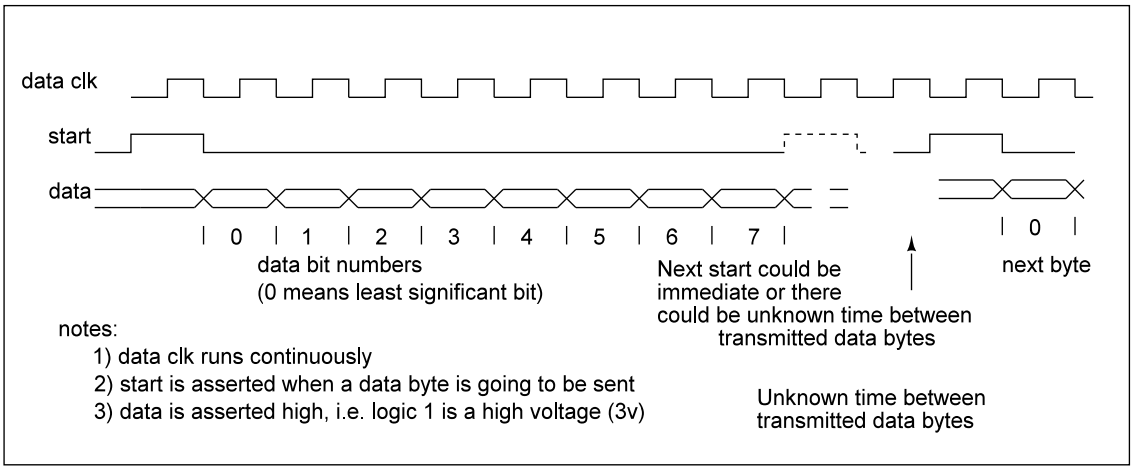

Figure 1 - Transmission timing

To Turn In

By next week, a short report containing:

a) An abstract
b) A block diagram
c) State diagram(s)
d) Any further design notes
e) VHDL listing