Professor Fearing EECS150/Problem Set 7 Fall 1997

Due at 5 pm, Thu. Oct. 16 in box outside 218 Cory Hall

1. Termination and Reflections (20 pts)

A bus of length 1 m with propagation velocity of $2 \times 10^8 ms^{-1}$ has a driver at the transmitting end with reflection coefficient of $-\frac{1}{2}$ and a receiver at the opposite end with a reflection coefficient of +1. Initally, the bus is a at 0 V. At time $t = 0ns$ a pulse wave of amplitude 3.5 V starts propagating from transmitter to receiver.

a. Sketch the voltage along the bus at 3, 8, 13, and 18 ns.

b. Roughly sketch the voltage versus time at the transmitter end of the wire for the first 30 ns.

2. ROM Based FSM Design (15 pts)

Design a Moore FSM that takes a synchronized serial data stream IN.H as input, and FOUND.H as output. The output FOUND.H should be asserted for one clock cycle whenever the sequence 100100 has been input on IN.H. Note that patterns may be overlapping, e.g. IN = 100100100000... should generate FOUND = 00000100100... The machine should start assuming that a "0" has already been input. Draw a state diagram and state table, and design the FSM using type D flip flops in a ROM based FSM. (Hint: this problem takes less than 5 flip flops). Make sure your FSM is self-starting. Both NSD and OD should be in the same ROM.

3. Counter Design (15 pts)

Design a counter using two x74_163 4 bit synchronous counters and minimum external logic which counts in the following sequence: $200_{TEN}, 201, 202, ..., 239, 240, 200, 201, ...$ Make sure the counter is self starting. Show with a timing diagram the clock, LOAD.L, and Q0-Q7 signals for counts 239, 240, 200, 201 to verify that the counter should work as designed.

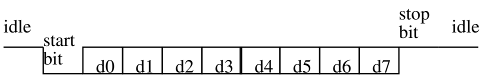
4. Asynchronous Serial Receiver Design (50 pts)

The universal asynchronous receiver transmitter or UART is at the heart of many serial communications connections, such as the serial port on an IBM PC. A transmission character is composed of an 8 bit information byte with a start bit before the first data bit, and a stop bit after the last data bit as shown in the figure below. The line may or may not be idle (at the 1 level) in between characters, with a start bit directly following a stop bit.

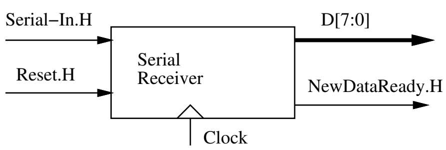
Design an 8 bit serial-to-parallel converter which has inputs of SerialIn.H, a clock at 8 times the nominal serial data rate, and a synchronous RESET.H. For a 9600 baud data rate, the clock is at 76.8 KHz. The outputs are the 8 data bits D0...D7, and NewDataReady.H. NewDataReady.H should be asserted for one bit time (8 clock cycles) after all 8 bits have been read in. A top level block diagram for the serial data receiver is shown below.

The method for sampling the serial data is as follows: wait for SerialIn.H to go low (signalling a start bit), then wait four clock periods and check that the data line is still low. (Hint: preload a counter....) By shifting every 8 clock periods thereafter, you will sample each serial bit roughly in the middle of a bit. All sequential devices must be driven from the same clock, but you can use the CE input on SR4RLE to enable shifting. (Hint: if the shift register is preset to all ones before shifting, then the start bit may be used to indicate that all ten bits (start + 8 data + stop) have been shifted in.)

Show a detailed block diagram of the data path (to the level of Xilinx library components), a state diagram for the FSM controller, and a functional timing diagram showing clock, SerialIn.h, other control signals, and NewDataReady.H.



Serial Transmission Format



Top Level Block Diagram of S/P Converter