UNIVERSITY OF TEHRAN
Electrical and Computer Engineering Department
Digital Logic Design, ECE 367 / Digital Systems I, ECE 894
Spring 1404
Computer Assignment 5

Synthesis of Counters, Shifters and State Machines - Week 13

Name:	**Date:**
Username:	

A serial communication device has a start sequence 01111110 (a zero, six 1's and a 0) and an abort sequence that is the complement of the start sequence, i.e., 10000001. The start-sequence detector is what you completed in Computer Assignment 4. The input of the serial communication device is *rcIn* and its output is *txOut*. White looking for the start sequence, the circuit is transmitting all 1's on its output, *txOut* output. After receiving the start sequence, the serial transmitter circuit puts the *rcIn* input on *txOut* for 97 clock cycles and issues *txValid* during this time. After the transmission is complete, the circuit goes back to its initial state looking for the next start sequence, where *txValid* becomes 0 and *txOut* stays at 1.

During the transmission of data, if an abort sequence is detected, transmission of data on *txOut* is stopped and an *txAbort* pulse is issued. The transmission circuit then returns to the initial state looking for the next start sequence.

a. Show a block diagram for the various FSMs and counters in this circuit and show signals that go between them.
b. For the start sequence part, use the complete design from Computer Assignment 4.
c. Use the machine of Part b to start the data transmission part. Show the necessary counter(s) and their incorporation in the FSM part.
d. Show the part for the abort sequence detection. This part has a separate state diagram and can use the counter of the start part.

Take the following steps to implement this circuit.

a. In Quartus, use existing library components, i.e., *lpm*, and other necessary gates and structures to design the transmitter part of your design. Use an asynchronous reset and the positive edge of the clock. This is your pre-synthesis design of the transmitter part of your circuit.
    i. Build a symbol for the transmitter circuit. Synthesize this design and see its timing, floor-plan and cells used. The synthesis process generates the post-synthesis description of the transmitter circuit. The .vo and .sdo files that are produced contain the netlist and timing of the synthesized circuit.
    ii. Instantiate the post-synthesis transmitter circuit in a SystemVerilog testbench and observe its timing.
b. In a new Quartus project, import the transmitter symbol and that of the start-sequencer of Computer Assignment 4 to build the complete serial transmitter circuit.

i. Synthesize this design and see its timing, floor-plan and cells used. The synthesis process generates the post-synthesis description of the serial transmitter circuit. The .vo and .sdo files that are produced contain the netlist and timing of the synthesized circuit.
ii. Instantiate the post-synthesis serial transmitter circuit in a SystemVerilog testbench and observe its timing and its functionality.

Deliverables:

Generate a report that includes all the items below:

A. Prior coming to the lab, for all the above problems hand-drawn schematic diagrams and partial timing diagrams are required. Your SystemVerilog descriptions must correspond to the circuit diagrams. Your simulation run and the project built for this purpose must be demonstrated to the TA.
B. Document your Quartus projects of Parts a, b, and c. Make sure you understand the synthesis outputs and their corresponding timings.
C. For all three parts, you should look at the FPGA layouts, device view and RTL view. Be able to explain the details of various views. In the layout be able to identify FPGA cells that use a memory element versus those that are purely combinational.
D. For all problems, be prepared to answer questions asked about the timings, generated hardware, pre- and post- synthesis, and FPGA mappings.

Make a PDF file of your report and name it with the format shown below:

FirstinitialLastnameStudentnumber-CAnn-ECEmmm

Where *nn* is a two-digit number for the Computer Assignment, *mmm* is the three-digit course number under which you are registered, and hopefully you know the rest. For the *Firstinitial* use only one character. For *Lastname* and for the multi-part last names use the part you are most identified with. Use the last five digits of your student id (exclude 8101) for the *Studentnumber* field of the report file name.