UNIVERSITY OF TEHRAN
Electrical and Computer Engineering Department
ECE 367
Digital Logic Circuits – Fall 1396
Computer Assignment 4
Counters, Registers and Simple controllers
Week 14

Problem description: The circuit to design in this assignment is one that takes a 16-bit binary number as its parallel input that is between 0 and 9999, and as output, it generates the BCD equivalent of its input. In addition to the BCD cascading blocks used in the previous experiment, the circuit has a 16-bit shift registrar and a counter that counts up to 16. We also have a controller that handles *start-done* handshaking and counts the number of shifts.

Controller: The circuit has a *start* input to start its operation and a *done* signal that signals the end of Binary to BCD conversion. When *start* goes from 0 to 1 and then back to 0, the 16-bit parallel input is loaded in a shift register. A clock cycle after that, shifting of binary data from the left of the input register and shifting it into the Binary to BCD circuit begins. This continues for 16 clock pulses, and when completed, the conversion is complete. At this time, the *done* signal becomes 1 and stays 1 while *start* is 0. The *done* output is 1 while the circuit is in its initial state waiting for *start*.

Implementation:

A) **The controller:** Implement the controller using a state machine and a counter. The state machine has *start* and *countComplete* inputs. The outputs of the controller are *done*, *loadInput* and *shiftCount* signals. In the first state (**Idle**), the *done* signal is issued and the circuit waits for a 1 on *start*. The circuit moves to the **Starting** state when *start* becomes 1, and moves to **Loading** state when *start* returns back to 0. In the **Loading** state, the 16-bit parallel input that is the Binary data in loaded into a shift register in the datapath. The circuit moves out of the **Loading** state and into the **Processing** state on the next clock. In this state the *shiftCout* signal is issued that causes left-shift of the binary data and counting the number of shifts. The circuit remains in the **Processing** state for as long as *countComplete* from the counter remains at 0. When *countComplete* is 1, the circuit exits the **Processing** state and returns to the **Idle** state**.** Implement the state machine at the gate level and develop a Verilog description with an **always** statement for the counter.

B) **The datapath:** the datapath has a shift-register that left-shifts parallelly loaded data into a cascade of four cascadable Binary to BCD structures. Use Verilog descriptions for these parts.

C) Write Verilog description of your circuit according to the design done above. Write a testbench to test your circuit.