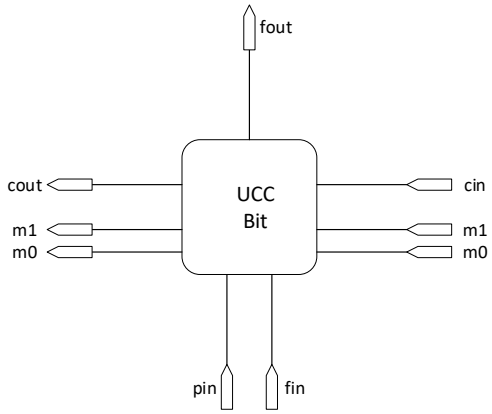
UNIVERSITY OF TEHRAN
Electrical and Computer Engineering Department
Digital Logic Design, ECE 367, Fall 1397
Computer Assignment 4
Basic flip-flops and memory elements, Iterative structures
Week 10

Name:
Date:

In this assignment, you will design an iterative multi-mode counter. A cell will be designed, and then cascaded to build an n-bit counter.

Shown below is a Universal Counter Combinational Bit. This is a combinational cell that can be cascaded to form an n-bit combinational circuit that can be used for building a universal counter. The mode inputs select functions 00: do nothing, 01: count up (increment), 10: count down (decrement), and 11: parallel load. Function input and output (fin, fout) are for feedback lines through the counter register part. Carry input and output (cin, cout) are for carry from one cell to the next when incrementing or decrementing is to be done. The parallel input (pin) appears as is on the function output of the circuit when mode is 3.



1. Do the gate level design of UCC Bit as described above. Use 7 ns for two and three input gate delays. Describe this in Verilog and test the circuit.
2. Using eight UCC Bits built a combinational circuit that can perform four functions according to the mode value as described above. Use Verilog generate statements. Develop a testbench and test this circuit. Find the worst-case delay of the circuit.

3. Using four NAND gates design an SR latch with active high S R inputs and the clock.

4. Use Verilog for describing the circuit of Part 3. Use approximate gate delays that are based on gate level delays of 7 NS. In a testbench, test your circuit, and, among other inputs, apply simultaneous active inputs (S and R) and see the loss of memory.
5. Use an inverter to convert the SR latch of Part 3 to a clocked D-latch. The inverter delay is the same as a gate delay.
6. Simulate your circuit of Part 5 to verify its operation.

7. Use eight latch structures as discussed above and the combinational circuit of Part 2 to design a circuit that we refer to as HIC (Hope It Counts). Generate a structural Verilog description of the HIC circuit and test its operation. By changing the functional mode of the circuit, show which of the functionalities of this circuit fail.

8. Use two D-latches of Part 5 to build a master-slave D-type flip-flop.
9. Create a synchronous reset for the flip-flop of Part 8 such that when this input becomes 1, the output becomes 0 with the clock.
10. Simulate your circuits of Part 6 to verify its operation and its resetting.
11. Use the flip-flop of Part 9 to build a four-function counter using the combinational circuit of Part 2.
12. Develop a testbench to test your counter of Part 11. Make sure your clock is not too fast for the complete propagation of the combinational part of your circuit plus the register part. Find this clock frequency. In your testbench demonstrate count-up, count-down, stall, reset, and other functionalities of your circuit.

13. Using an **always** statement show the behavioral description of the counter of Part 11.
14. Using as **assign** statement, make the above circuit one with bidirectional input-output, and add

15. Test the counter of Part 14 for all its operations including one that controls the three-state IO of the circuit. Use constructs such as **repeat** and **$random()** for test data generation.