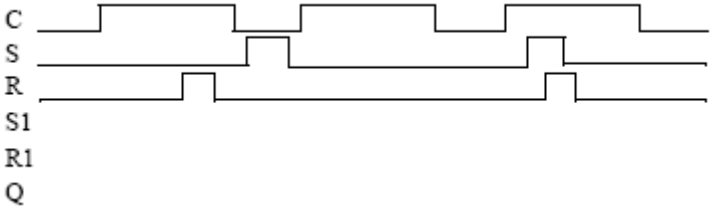
ECE 274 – Digital Logic
Homework Assignment 2
Due September 26 *(beginning of class)*

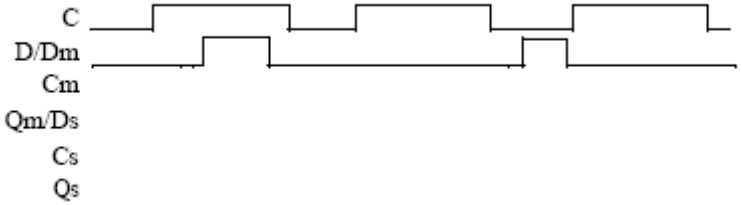
1.) *(5 points)* Trace the behavior of an SR latch for the following situation: Q, S and R are 0 and have been for a long time, then S changes to 1 and stays there for a long time, then S changes back to 0. Using a timing diagram, show the values that appear on every wire for every change on a wire. Assume logic gates have a tiny but non-zero delay.

2.) *(5 points)* Repeat the previous problem, but assume that S was changed to 1 just long enough for the signal to propagate through one logic gate, after which S was changed back to 0 – in other words, S did not satisfy the hold time of the latch.

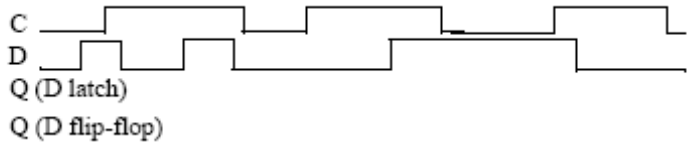
3.) *(5 points)* Trace the behavior of a level-sensitive SR latch for the input pattern below. Complete the timing diagram, assuming logic gates have a tiny but non-zero delay.



4.) *(5 points)* Trace the behavior of an edge-triggered D flip-flop using the master-servant design for the input pattern shown below. Complete the timing diagram, assuming logic gates have a tiny but non-zero delay.



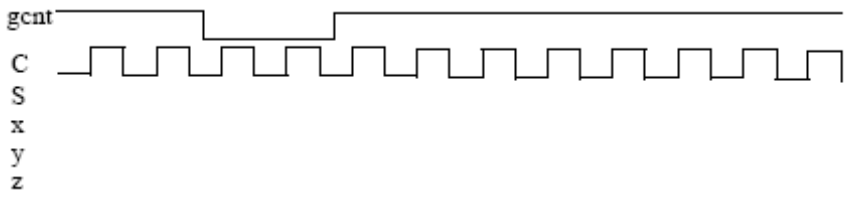
5.) *(5 points)* Compare the behavior of D latch and D flip-flop devices by completing the following timing diagram. Provide a brief explanation of the behavior of each device.



6.) *(10 points)* Draw a state diagram for an FSM that has an input X and an output Y. Whenever X changes from 0 to 1, Y should become 1 for two clock cycles and then return to 0 -- even if X is still 1.

7.) *(15 points)* Draw a state diagram for an FSM with an input *gcnt* and three outputs, *x, y* and *z*. The *xyz* outputs generate a sequence called a Gray code in which exactly one of the three outputs changes from 0 to 1 or from 1 to 0. The Gray code sequence the FSM should output is 000, 010, 011, 001, 101, 111, 110, 100, repeat. The output should change only on a rising clock edge when the input *gcnt* = '1'.

8.) *(15 points)* Trace through the execution of the FSM you created in the previous exercise by completing the following timing diagram, where C is the clock input and S is the n-bit state register.



9.) *(15 points)* Using the five-step process for designing a controller, convert the FSM you created in Exercise 7 to a controller, implementing the controller using a state register and logic gates.

10.) *(20 points)* Create an FSM that has an input *X* and an output *Y*. Whenever *X* changes from 0 to 1, *Y* should become 1 for five clock cycles and then return to 0 -- even if *X* is still 1. Using the five-step process for designing a controller, convert the FSM to a controller, stopping once you have created the state table.