EECS150
Midterm1
10/5/06
Professor Pister

Note: problem 1 and 2 are a little trickier than last semester! Read carefully!

1) Given F = A'B' + CD'
    a. Write F' in product of sums notation

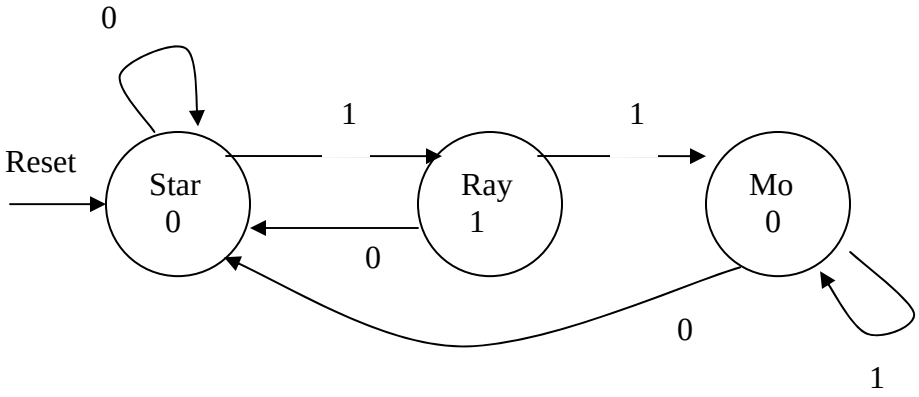
    b. Implement F using as few 2 input NOR gates as possible. Assume that only the true literals (A,B,C,D) are available, not their complements(A', B', C', D').

2) Given G = (A'+ B')(C+D')
    a. Write G' in sum of products notation.

    b. Implement G using as few 2 input NAND gates as possible. Assume that only the true literals are available, not their complements.

3) Answer the following questions for the FSM below:
   a. Briefly describe function of this sequence detector. When is the output 1?

   b. Write a verilog module which would implement this FSM for input variable "In" and output variable "Out." Use the same standard format as was presented in the Lab 3 lecture and used in Lab 3. (Define your states; use one always block for next state and output; use one always block for state transitions.



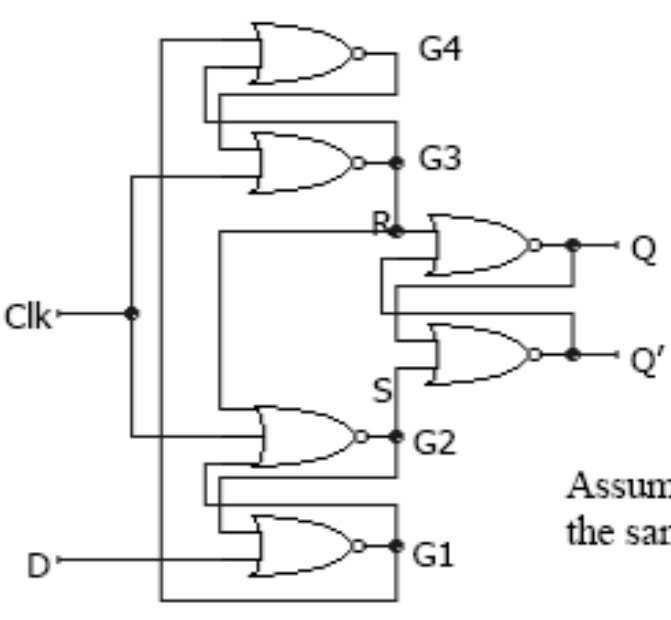
4) A finite state machine has one input and one output. The output becomes 1 and remains 1 thereafter when at least two 0s and at least two 1s have occurred as inputs, in any order after reset. Draw a state diagram of this FSM as a Moore machine. Try to minimize the number of states.

5) A Moore machine has one input and output. The output should be 1 if the total number of 0s at the input is odd, and the total number of 1s at the input is an even number greater than 0. Draw a state diagram. Try to minimize the number of states.

6) Design a 3 FlipFlop counter which transitions through states $Q_2Q_1Q_0$ = 000, 100, 110, 111, 011, 011 and then repeats.

   a. Draw the state diagram and state transition table
   b. Draw the Karnaugh maps, clearly indicating the implicants that you use in your covers of the next-state functions.
   c. Implement the counter using D flip flops and whatever gates you like.
   d. Is your counter self starting? If yes, show the transitions of the unused states. If no, change it to make it self starting, and show the transitions of the unused states.

7)



Assume all gates have exactly the same delay.