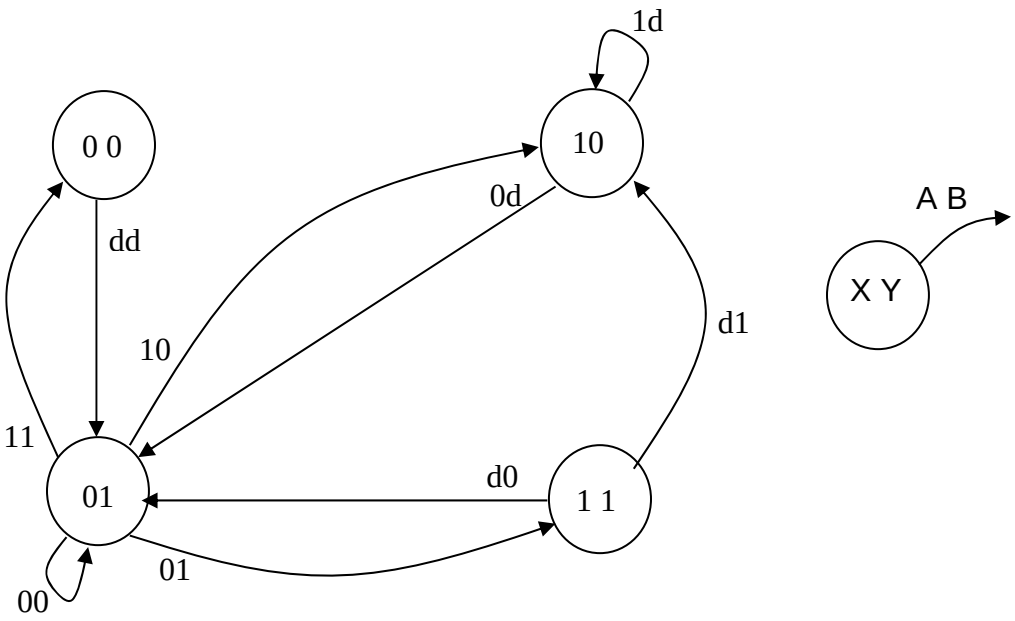
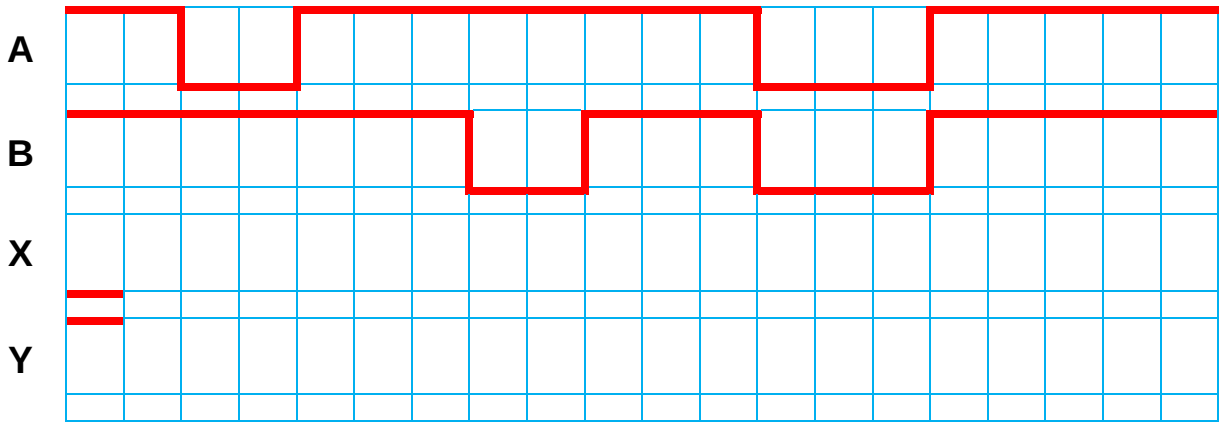
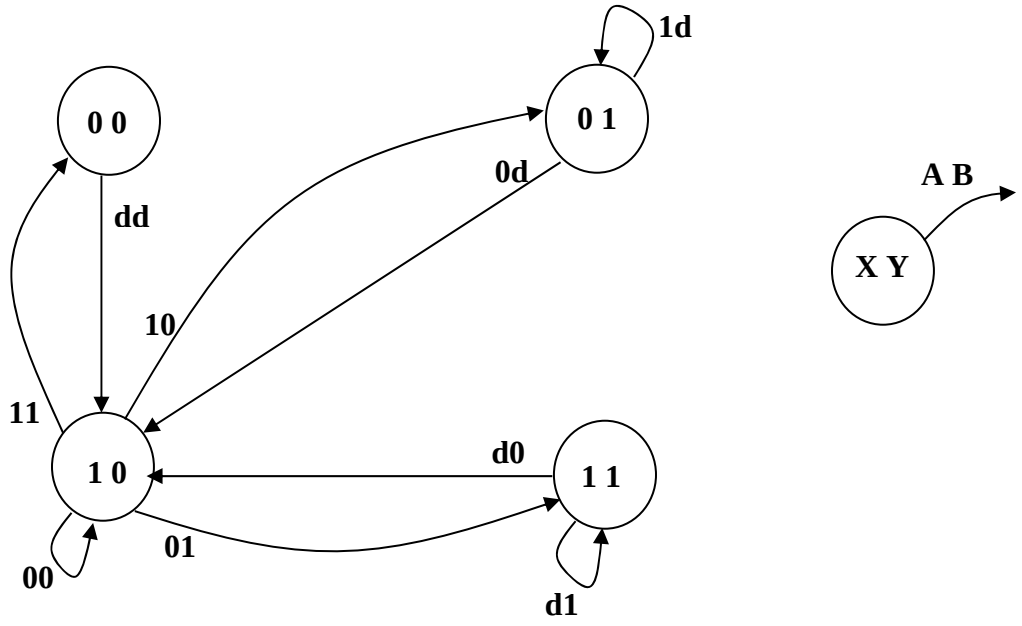
Practice Homework Problems for Module 3

1. Given the following state transition diagram, complete the timing chart below.



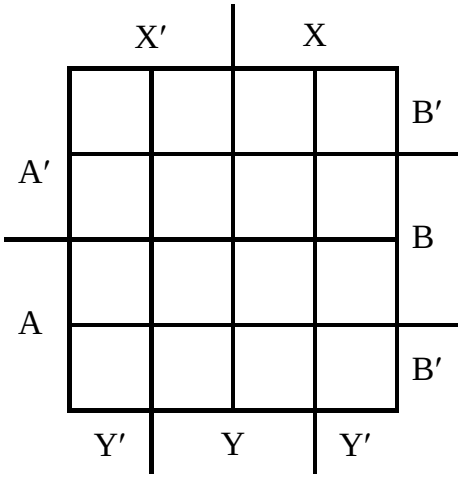


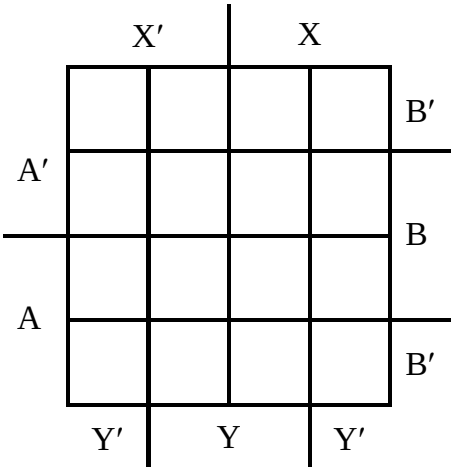
2. Given the following state transition diagram, determine the *next state equations* it represents in *minimum sum-of-products form.*



X	Y	A	B	X*	Y*
0	0	0	0		
0	0	0	1		
0	0	1	0		
0	0	1	1		
0	1	0	0		
0	1	0	1		
0	1	1	0		
0	1	1	1		
1	0	0	0		
1	0	0	1		
1	0	1	0		
1	0	1	1		
1	1	0	0		
1	1	0	1		
1	1	1	0		
1	1	1	1		

X* and Y* are "shorthand" for the next state of X and Y

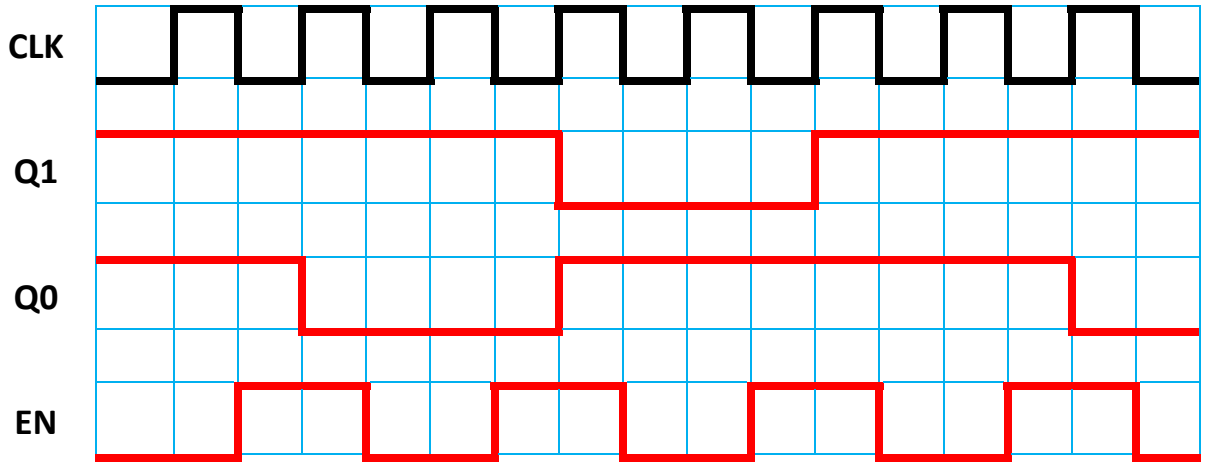


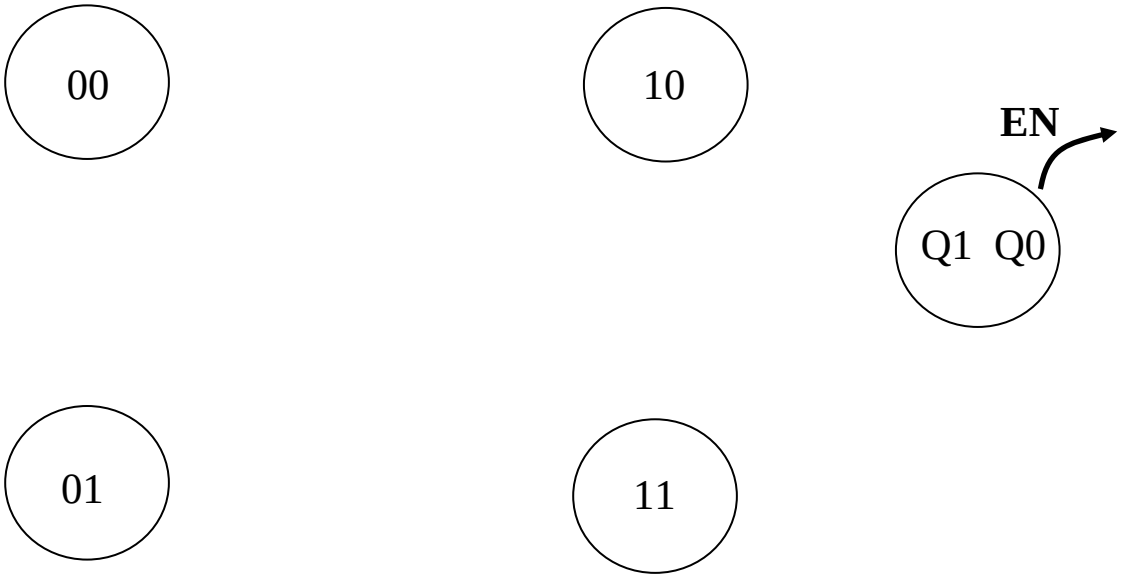


X* = _____

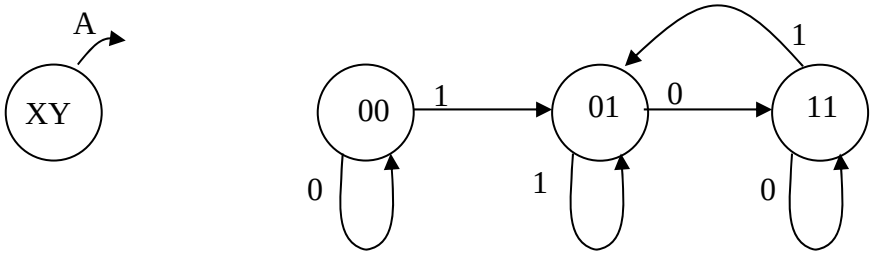
Y* = _____

3. Given the timing diagram, below, for a state machine that has one input (EN) and two state variables (Q1 and Q0), derive a state transition diagram:





4. Given the following state transition diagram, determine:



(a) The next state equation for X if the state machine is designed for **minimum cost**

(b) The next state equation for X if the state machine is designed for **minimum risk**

(c) The next state equation for Y if the state machine is designed for **minimum cost**

(d) The next state equation for Y if the state machine is designed for **minimum risk**

---

5. A "new" type of flip-flop, the RG ("Raul Good"), is described by the following PS-NS table. Derive its next state equation and excitation table.

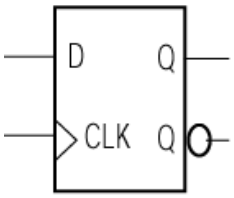
R	G	Q	Q*
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

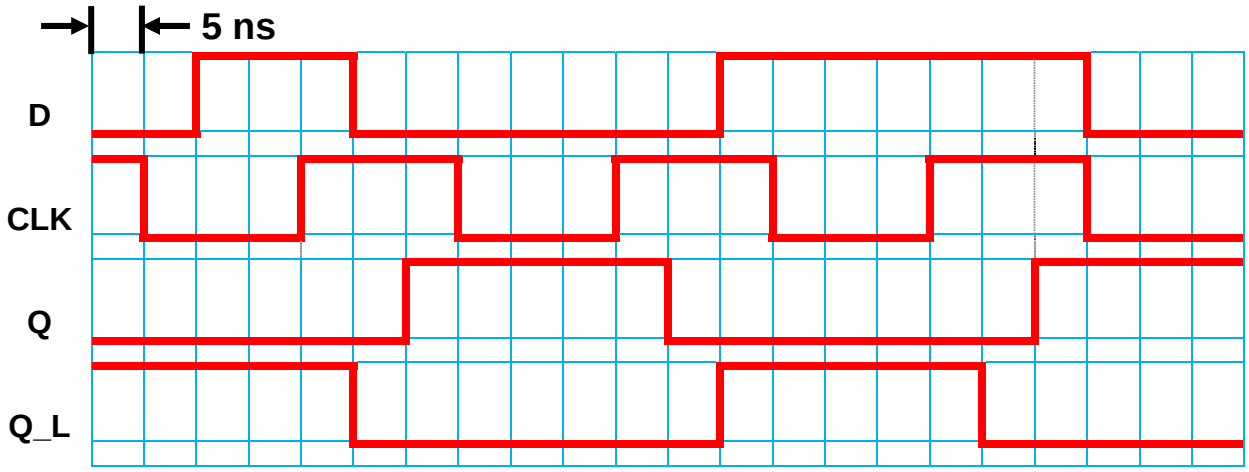
Excitation table:

Q	Q*	R	G
0	0		
0	1		
1	0		
1	1		

Q* = _____

6. Given the following timing chart for an edge-triggered D flip-flop, determine the following based on the excitation signals (D and CLK) depicted:





(a) The **nominal setup time** provided for the D flip-flop

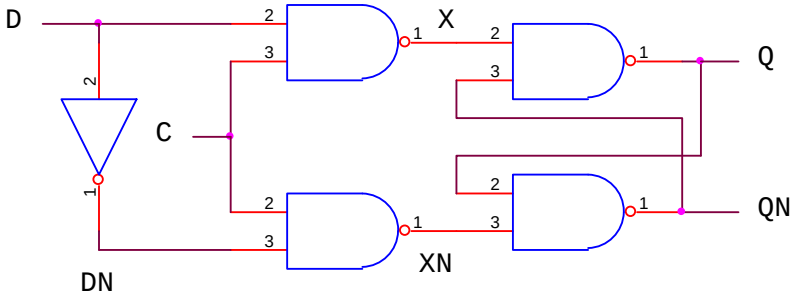
(b) The **nominal hold time** provided for the D flip-flop

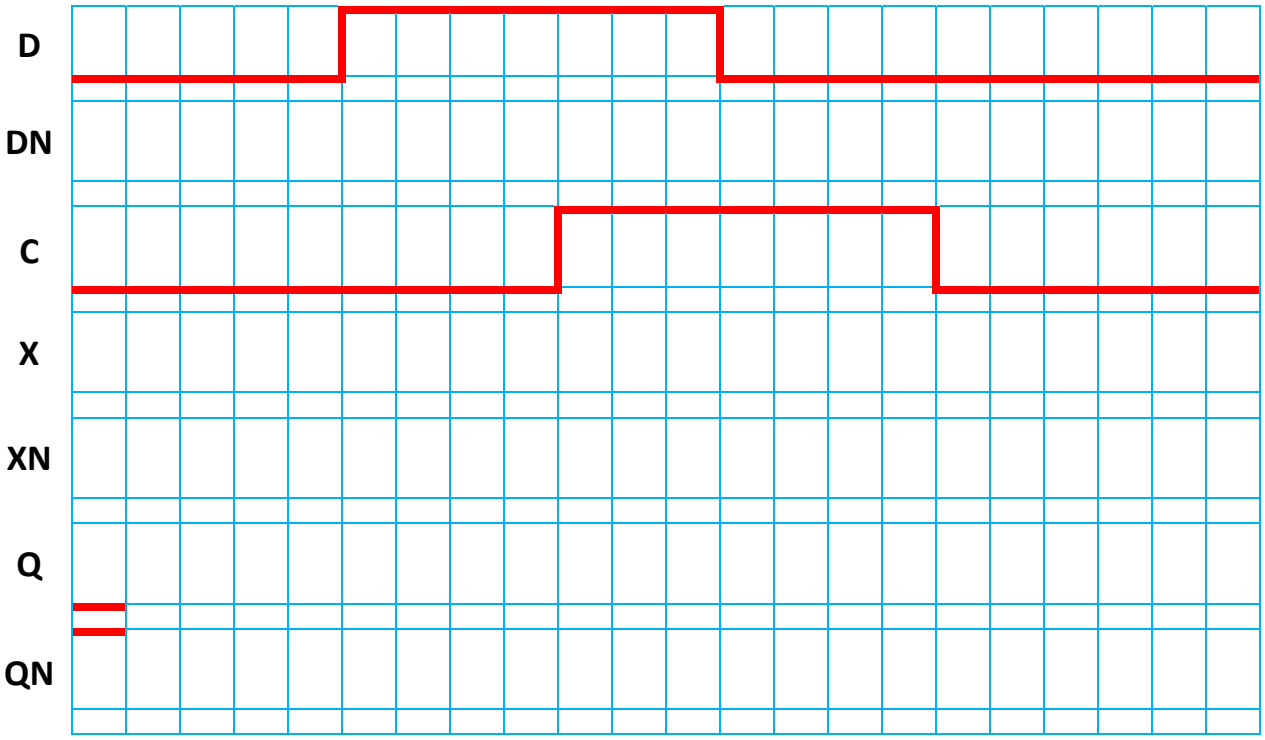
(c) The **nominal clock pulse width** provided for the D flip-flop

(d) The $\mathbf{t_{PHL(C \rightarrow Q)}}$ of the D flip-flop

(e) The $\mathbf{t_{PLH(C \rightarrow Q)}}$ of the D flip-flop

7. Complete the timing chart, below, for a D latch, and answer the questions that follow. Assume each gate has 5 ns of delay (t_{PLH} and t_{PHL}), and that each division on the chart is 5 ns. (See the Clicker Quiz for a detailed analysis of the latch portion of the circuit.)





(a) Determine the **minimum time** input C should be asserted (while the D input remains stable) to ensure reliable operation of the latch.

(b) Determine the **nominal setup time** provided for the D latch.

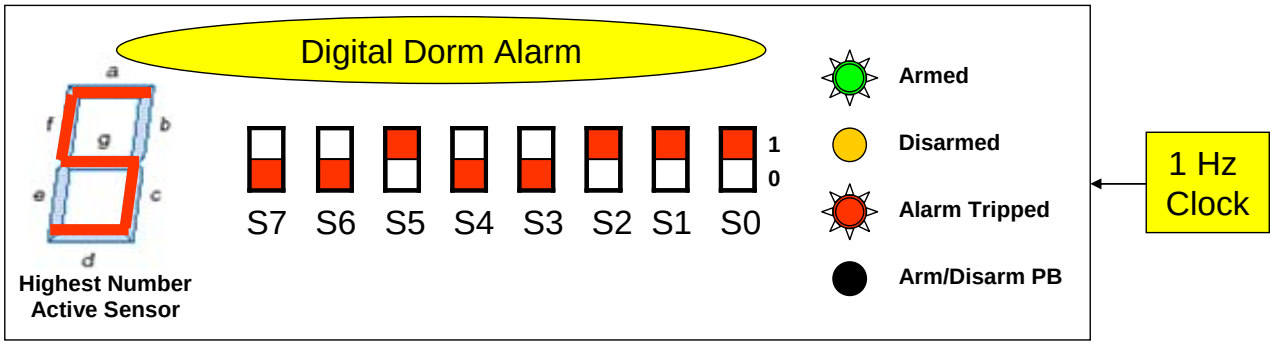
(c) Determine the **nominal hold time** provided for the D latch.

(d) Determine the $\mathbf{t_{PLH(C \rightarrow Q)}}$ of the D latch.

(e) Determine the $\mathbf{t_{PHL(D \rightarrow Q)}}$ of the D latch.

8. Implement a "dorm-room alarm" that accommodates eight sensor inputs, labeled S0 through S7, plus an ARM/DISARM pushbutton than can be used to "toggle" the state of the alarm system (a GREEN LED should be illuminated if the system is armed, and a YELLOW LED should be illuminated if the system is disarmed). If any sensors are asserted while the alarm is armed, the number of the *highest* sensor input asserted should be displayed on a 7-segment LED and a RED LED (that indicates the alarm has been tripped) should start *blinking* (at a 1 Hz rate, based on a clock signal provided by the function generator). The RED LED should stop blinking when the alarm is disarmed, and the 7-segment display should be blank (the 7-segment display should also be blank if the alarm is armed and none of the sensor inputs are asserted). Draw a Moore model for the "arm/disarm" state machine, and a separate Moore model for the "alarm tripped" state machine. Create an Verilog source file for your design, with all inputs and outputs clearly defined.

   Create the following:
   a. Moore model of "arm/disarm" state machine
   b. Moore model of "alarm tripped" state machine
   c. Verilog source file listing

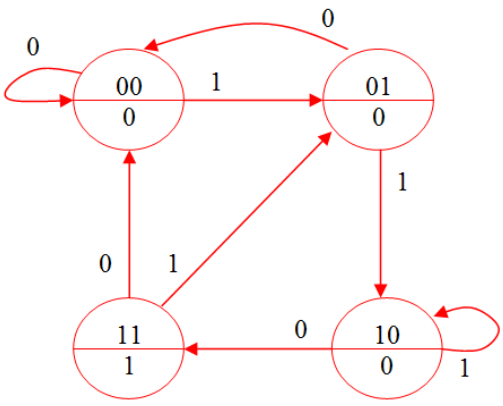


The DDA (Digital Dorm Alarm) in action. The GREEN LED indicates the alarm is in the "armed" state, the blinking RED LED indicates the alarm has been tripped, and the 7-segment display indicates the highest number sensor that is active. The Arm/Disarm pushbutton "toggles" the alarm between the armed and disarmed states. The 7-segment display is *blank* if the alarm is disarmed or, if armed, none of the sensor inputs are asserted.

9. Given the following state transition diagram, determine:

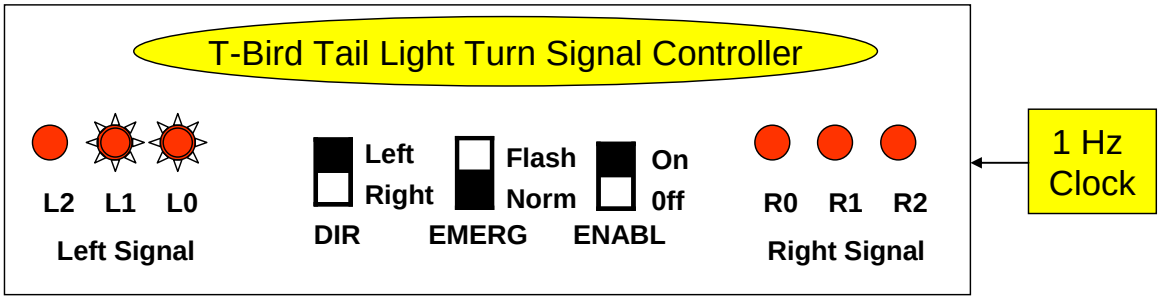
   (a) Assuming the state machine depicted is initialized to state **00**, determine the output sequence generated by the input sequence **111000111000**

   (b) Determine the **embedded binary sequence** recognized by this state machine



10. Inspired by the (ancient) Beach Boys hit single, *Fun Fun Fun*, you wish to implement a T-Bird Tail Light Turn Signal Controller (TBTLTSC)…hoping, at long last, you've found *something* that your friends can actually relate to (maybe not the Beach Boys, though…). Here, each "tail light" will consist of three LEDs, which will be illuminated in a "building dot" mode to indicate the turn direction (either "left" or "right", selected by a DIP switch). An "emergency flash" mode (in which all the tail lights alternate between the on and off states) will be controlled by a second DIP switch. The overall taillight enable will be controlled by a third DIP switch; if disabled (EN=0), all LEDs should be off.

    Create the following:
    a. Mealy model of state machine
    b. Verilog source file listing



The TBTLTSC taking a *left turn,* for which the output sequence should be: (a) L0, (b) L0 and L1, (c) L0, L1, and L2, (d) all off. This sequence should continuously repeat as long as the enable signal is asserted (ENABL=1). If the "emergency flash mode" is selected, the six lights should alternate between the "all on" and "all off" states (DIR is ignored). When disabled (ENABL=0), all LEDs should be off.

11. Design a 3-bit, self-correcting RING counter with glitch-free decoded outputs. Draw a state transition diagram to prove your design is self-correcting. NOTE: The initial state should be "001", and the counter should SHIFT LEFT.

    Create the following:
    a. Moore model of state machine, clearly showing the "self-correcting" mechanism
    b. Verilog source file listing