

P1 (20 points): We want to design a circuit with input W and an output Z , where Z will equal 1 if, for the last three clock cycles, W has been 1.

A: Draw a state diagram for a Moore Finite State Machine (FSM) that implements this circuit in four states, specified as follows:



B: Complete a state table for the above state diagram with the state assignments as shown below for state variables D_1 and D_0 .

	W=0	W=1	Z
S-0 : 00			
S-I : 01			
S-II : 10			
S-III : 11			

C: Use K-maps to show that the output and next-state variables can be expressed as:

$$\begin{aligned}
 D_1^{new} &= (w)(D_1 + D_0) \\
 D_0^{new} &= (w)(D_1 + \bar{D}_0) \\
 z &= D_1 D_0
 \end{aligned}$$

D: Let's consider if the states were encoded as:

$$S-0 = 00, S-I = 01, S-II = 11, S-III = 10$$

Use K-maps to show that the output and next-state variables with this new encoding can be expressed as:

$$\begin{aligned}
 D_1^{new} &= (w)(D_1 + D_0) \\
 D_0^{new} &= w\bar{D}_1 \\
 z &= D_1\bar{D}_0
 \end{aligned}$$

E: Draw the circuit for this FSM using only D Flip-Flops, AND gates, and one OR gate (Do not use any NOT gates).

P2 (8 points): Derive a minimal state table for a Moore FSM that acts as a three-bit parity generator. For every three bits that are observed on the input b during three consecutive clock cycles, the FSM generates the parity bit $p=1$ if and only if the number of 1s in the three-bit sequence is even. Note that the output p may only assert after a set of three bits (six, nine, twelve, etc.) have been input.

P3 (15 points): Let us build a Moore FSM that has a 1-bit input P and a 1-bit output Q. P will be either 1 or 0 on any particular clock cycle. Q=0 if P has been 1 for an even number of clock cycles; Q=1 if P has been 1 for an odd number of clock cycles.

A: Draw the state diagram for this Moore FSM.

B: Draw the state table for this FSM.

C: Draw a state assigned table for this FSM. The state should be the same as the output: Q.

D: Draw the truth table for this FSM's next-state variable.

E: Derive the expression for the next state variable and the output Q.

F: Draw the circuit for this FSM. If done properly, the circuit you create will implement a component that you have seen before. What component have you implemented?

P4 (15 points): We want to make a Mealy FSM with an input w and an output z which detects if particular sequences has been observed.

A: Draw a state diagram for a Mealy FSM that outputs z=1 if the input w follows the sequence 1→1→0→0→1. If the sequence has not been completed, then output z=0.

B: Draw a state diagram for a Mealy FSM that outputs z=1 if the input w follows either the sequence 0→0→1 or the sequence 1→1→0. If neither sequence has been completed, then output z=0.

P5 (8 points): Draw a state diagram for a Moore FSM that functions as a special modulo-4 counter with two-bit input N and two-bit output P. On each clock cycle, the value of N is added to the counter value. Both N and the counter output are expressed in 2's complement; if N=11 (-1) and P=01 (+1), then the next output values for P should be 00.

P6 (14 points): Design a circuit that with one bit input N and 3-bit output P that operates as follows:

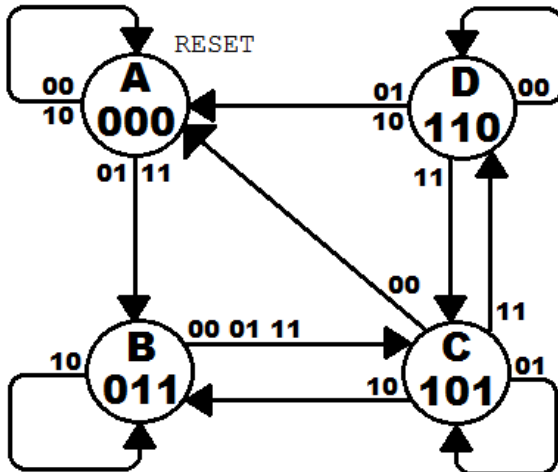
When N=0, the new value of P will be $P_{new} = 2 * P_{old}$. When N=1, the new value of P will be $P_{new} = 2 * P_{old} + 1$. Since P is only three bits, each operation must be modulo 8 (i.e., if P=5, then $2*P=10$ is too large to fit into the 3-bit value for P, so we subtract 8 from 10 to get 2, which should be the new value for P).

A: Draw the state table for the Moore FSM that implements this circuit. Your circuit should have a state for each possible output.

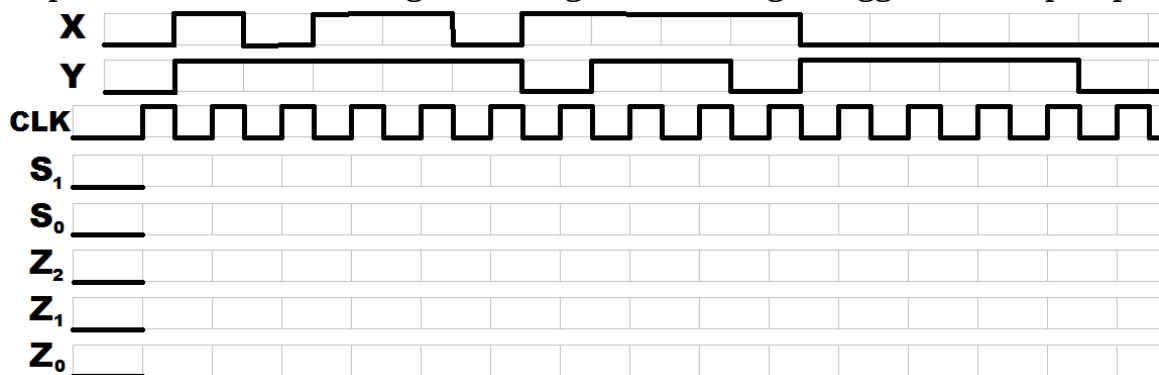
B: Show the next state Boolean expressions for each state variable.

C: The circuit that you end up with for this circuit is equivalent in functionality to a circuit that we have already discussed. What circuit is this?

P7 (20 points): Look at the state diagram below. The input variables are X and Y. The state variables are S₁ and S₀. The state encodings are as follows: A=00, B=01, C=10, and D=11. The output variables are Z₂, Z₁, and Z₀.



I: Fill in the timing diagram below given the state diagram for a circuit that implements this state diagram using Positive-Edge-Triggered D FlipFlops.



II: Fill in the state table with state assignments

	X=0, Y=0	X=0, Y=1	X=1, Y=0	X=1, Y=1	Z ₂ Z ₁ Z ₀
A					
B					
C					
D					

III: Draw the truth table and show that the next-state expressions can be expressed as follows:

$$S_0^{new} = \bar{S}_1 \bar{S}_0 (Y) + \bar{S}_1 S_0 (X\bar{Y}) + S_1 \bar{S}_0 (X) + S_1 S_0 (\bar{X}\bar{Y})$$

$$S_1^{new} = \bar{S}_1 \bar{S}_0 (0) + \bar{S}_1 S_0 (\bar{X} + Y) + S_1 \bar{S}_0 (X + Y) + S_1 S_0 (XY + \bar{X}\bar{Y})$$

IV: Derive expressions for the output variables Z₂, Z₁, and Z₀ in terms of S₁ and S₀.