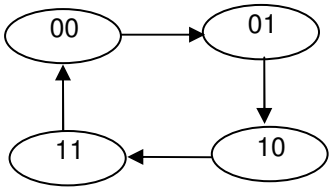
Digital Logic Design - Chapter 5

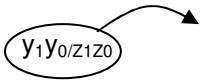
1S. Design a 2-bit binary up counter
   a) using positive-edge-triggered D flip-flops.
   b) using positive-edge-triggered T flip-flops.
   c) using positive-edge-triggered JK flip-flops.

Solution:

a)

Step 1) Draw a state diagram.
After every rising edge clock count increases by one.





Step 2) There are 4 states; therefore we need 2 FFs (2^2=4) "using full encoding"

Step 3) Assign a unique code for each state. In this case it is the same as the count

Step 4) Derive excitation input and external output equations.
The output and present state are the same (Moore machine); below is the PS/NS table to derive excitation equation:

Present State Y_1 Y_0	Next State Y^+_1 Y^+_0	**Part a - D FF** $D = y^+$ D_1 D_0	**Part b - T FF** $T=y\oplus y^+$ T_1 T_0	**Part c - JK FF** $J= Y^+$, $K= Y^{+'}$ J_1 K_1 J_0 K_0
0 0	0 1	0 1	0 1	0 1 1 0
0 1	1 0	1 0	1 1	1 0 0 1
1 0	1 1	1 1	0 1	1 0 1 0
1 1	0 0	0 0	1 1	0 1 0 1

D-FF Excitation Equations:

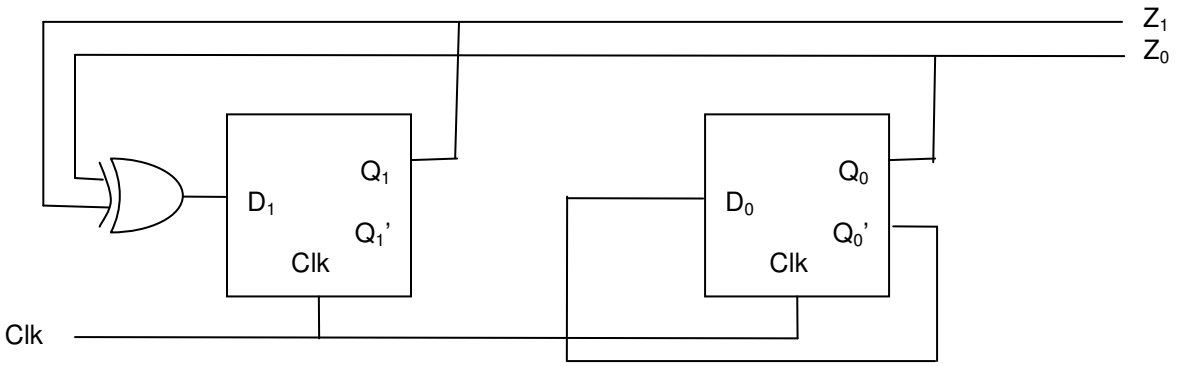
y_1 \ y_0	0	1
0	0	1
1	1	0

$D_1 = y_1 \oplus y_0$

y_1 \ y_0	0	1
0	1	0
1	1	0

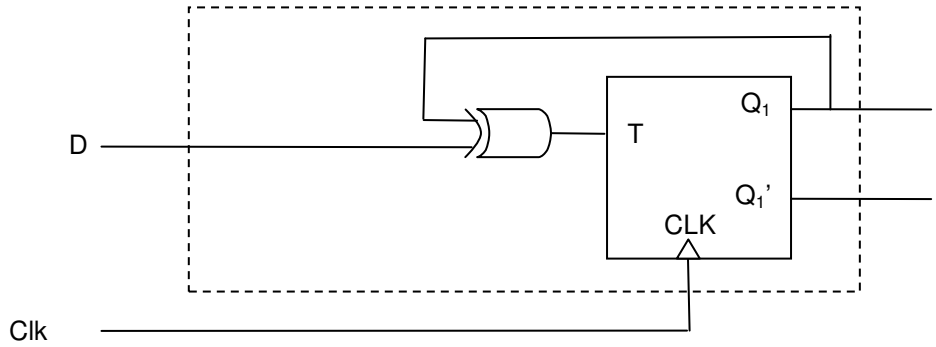
$D_0 = y_0'$

Step 5) Schematic:



b – Alternative solution 1)

Since we already have designed using D flip-flop then use T-FF flip-flop to build a D flip-flop as shown below and use them to implement the circuits with T flip flop.



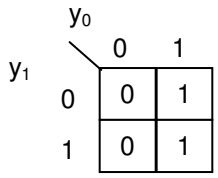
b – Alternative solution 2)

Steps 1,2 and 3 are the same regardless of flip-flop type. So start with step 4 for T flip-flop:

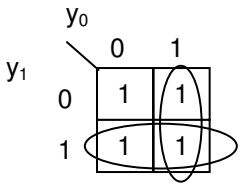
Step 4) Derive excitation input and external output equations.
The output and present state are the same (Moore machine); below is the PS/NS table to derive excitation equation:

Present State Y_1 Y_0	Next State Y^+_1 Y^+_0	**Part b - T FF** $T=y\oplus y^+$ T_1 T_0
0 0	0 1	0 1
0 1	1 0	1 1
1 0	1 1	0 1
1 1	0 0	1 1

T-FF Excitation Equations:

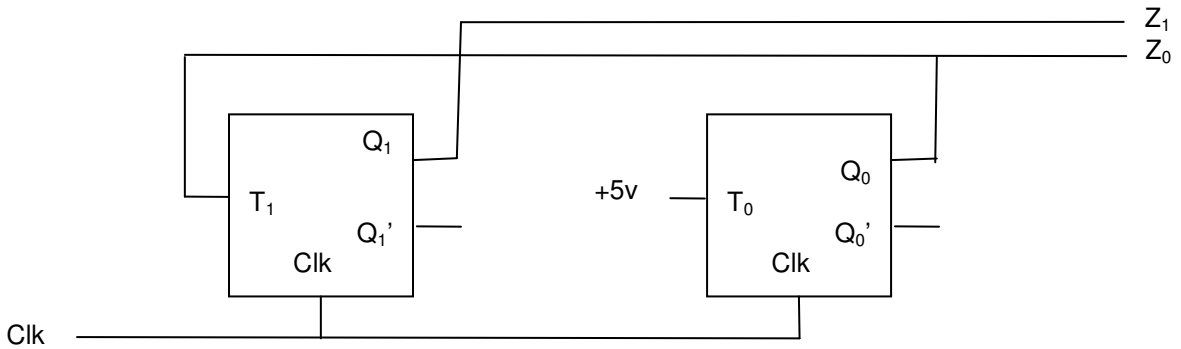


$T_1 = y_0$



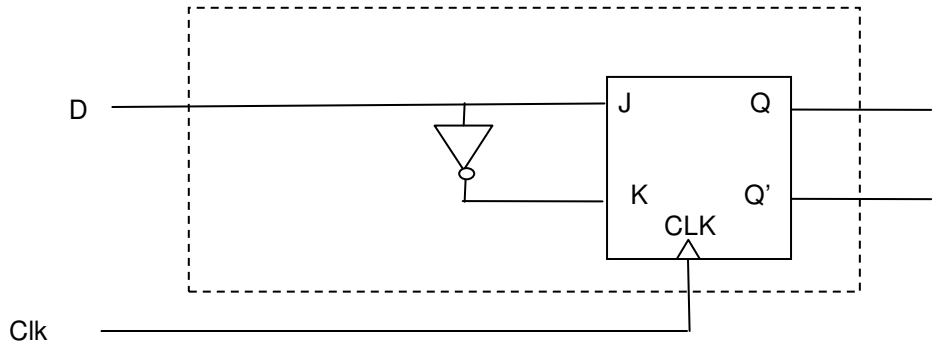
$T_0 = 1$

Step 5) Schematic:



c – Alternative solution 1)

Since we already have designed using D flip-flop, use JK-FF flip-flop to build a D flip-flop as shown below and use them to implement the circuits with JK flip flop.



c – Alternative solution 2)

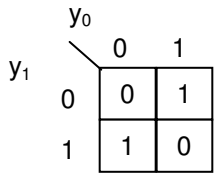
Steps 1,2 and 3 are the same regardless of flip-flop type. So start with step 4 for JK flip-flop:

Step 4) Derive excitation input and external output equations.

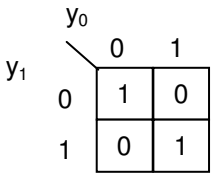
The output and present state are the same (Moore machine); below is the PS/NS table to derive excitation equation:

Present State Y_1 Y_0	Next State Y^+_1 Y^+_0	**Part c - JK FF** $J= Y^+$, $K= Y^{+'}$ J_1 K_1 J_0 K_0
0 0	0 1	0 1 1 0
0 1	1 0	1 0 0 1
1 0	1 1	1 0 1 0
1 1	0 0	0 1 0 1

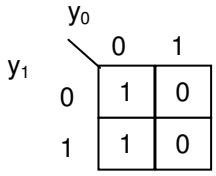
D-FF Excitation Equations:



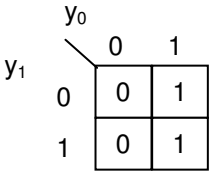
$J_1 = y_1 \oplus y_0$



$K_1 = (y_1 \oplus y_0)'$

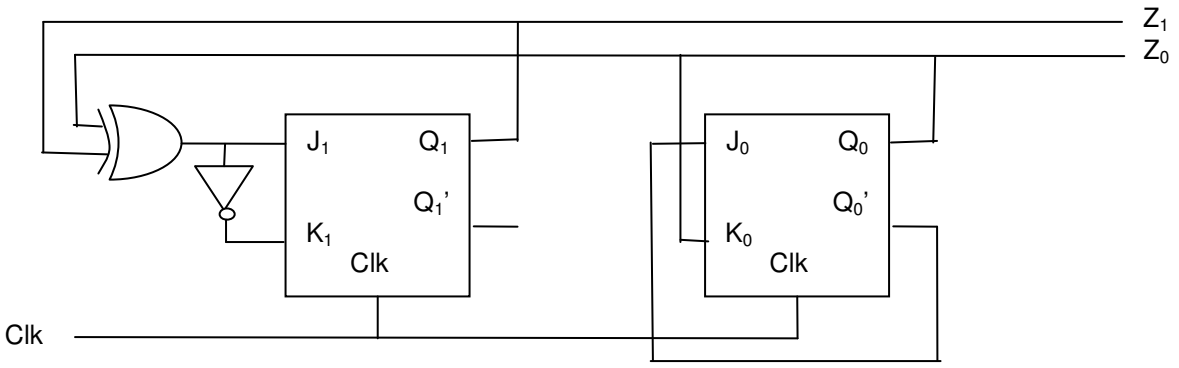


$J_0 = y_0'$



$K_0 = y_0$

Step 5) Schematic:



1U. Design a 2-bit binary down counter
   a) using positive-edge-triggered D flip-flops.
   b) using positive-edge-triggered T flip-flops.
   c) using positive-edge-triggered JK flip-flops.

Solution:

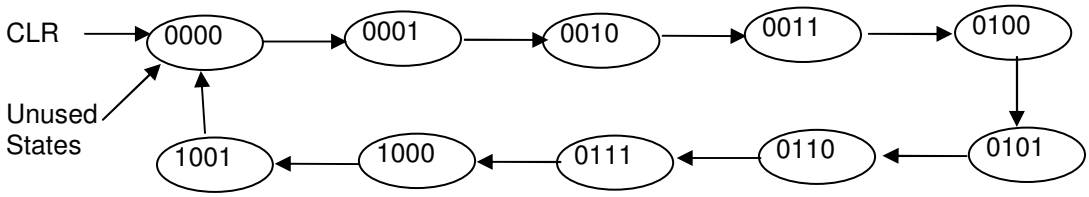
2S. Design a (Binary Coded Decimal) counter using:

   a) positive-edge-triggered D flip-flops.
   b) positive-edge-triggered T flip-flops.
   c) positive-edge-triggered JK flip-flops.

Solution:

(Steps 1 through 3 of design are independent of the type of FF used.)

Step 1) Draw a state diagram.



Step 2) There are 10 states; therefore we need 4 FFs (2^4=16> 10) (using full encoding).

Step 3) Assign a unique code for each state. (Use the BCD value as the state value.)

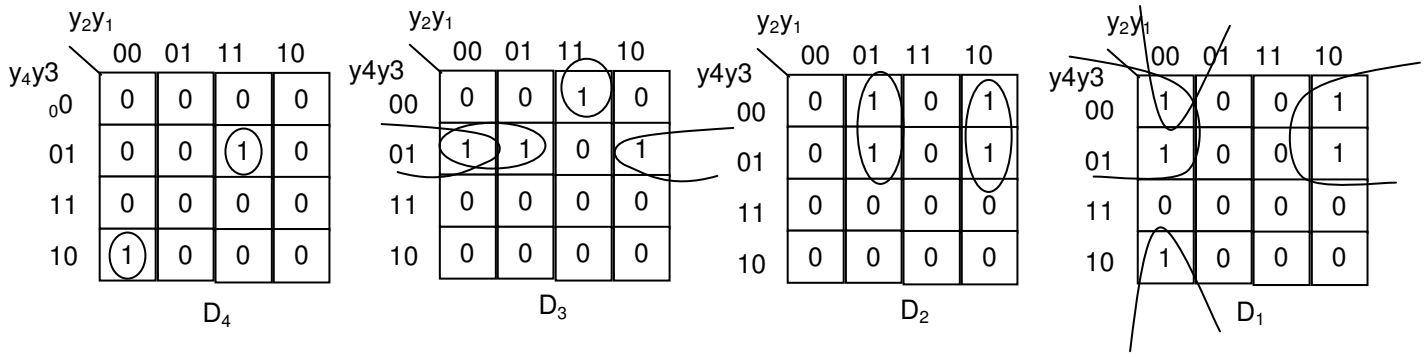
Step 4) Derive excitation input and external output equations.
Note: The output and present state are the same (Moore machine); therefore:

$Z_1 = Y_1$
$Z_2 = Y_2$
$Z_3 = Y_3$
$Z_4 = Y_4$

Present State $Y_4\ Y_3\ Y_2\ Y_1$	Next State $Y^+_4\ Y^+_3\ Y^+_2\ Y^+_1$	**Part a - D FF** $D = y^+$ $D_4\ D_3\ D_2\ D_1$	**Part b - T FF** $T=y\oplus y^+$ $T_4\ T_3\ T_2\ T_1$	**Part c - JK FF** $J= Y^+\ ,\ K= Y^{+'}$ $J_4\ K_4\ J_3\ K_3\ J_2\ K_2\ J_1\ K_1$
0 0 0 0	0 0 0 1	0 0 0 1	0 0 0 1	0 1 0 1 0 1 1 0
0 0 0 1	0 0 1 0	0 0 1 0	0 0 1 1	0 1 0 1 1 0 0 1
0 0 1 0	0 0 1 1	0 0 1 1	0 0 0 1	0 1 0 1 1 0 1 0

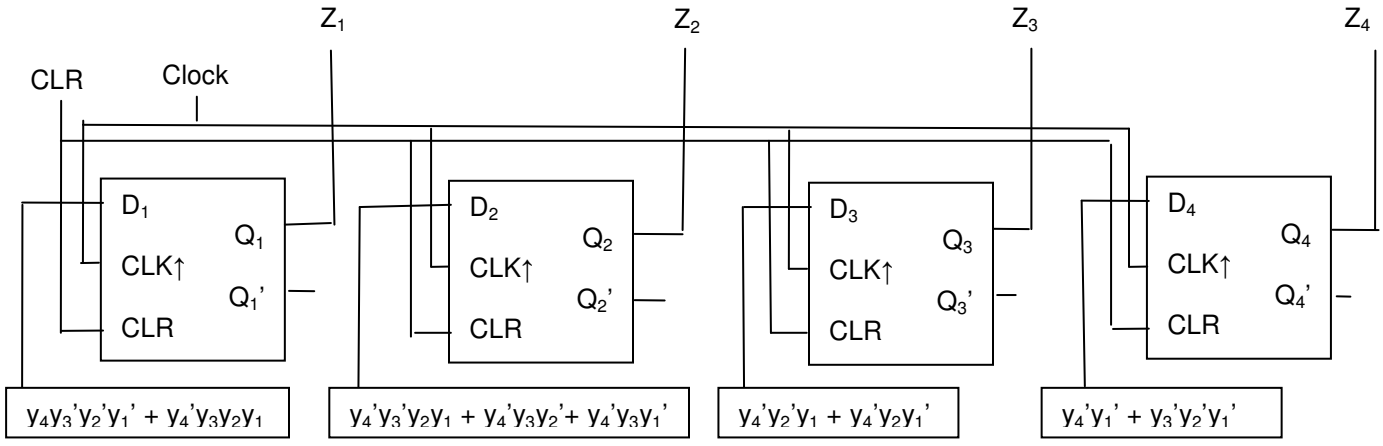
0	0	1	1	0	1	0	0	0	1	0	0	0	1	1	1	0	1	1	0	0	1	0	1
0	1	0	0	0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	0	1	1	0
0	1	0	1	0	1	1	0	0	1	1	0	0	0	1	1	0	1	1	0	1	0	0	1
0	1	1	0	0	1	1	1	0	1	1	1	0	0	0	1	0	1	1	0	1	0	1	0
0	1	1	1	1	0	0	0	1	0	0	0	1	1	1	1	1	0	0	1	0	1	0	1
1	0	0	0	1	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	1	1	0
1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0	1	0	1	0	1
1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	1	0	1
1	0	1	1	0	0	0	0	0	0	0	0	1	0	1	1	0	1	0	1	0	1	0	1
1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	1	0	1	0	1
1	1	0	1	0	0	0	0	0	0	0	0	1	1	0	1	0	1	0	1	0	1	0	1
1	1	1	0	0	0	0	0	0	0	0	0	1	1	1	0	0	1	0	1	0	1	0	1
1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	0	1	0	1	0	1	0	1

D-FF Excitation Equations:

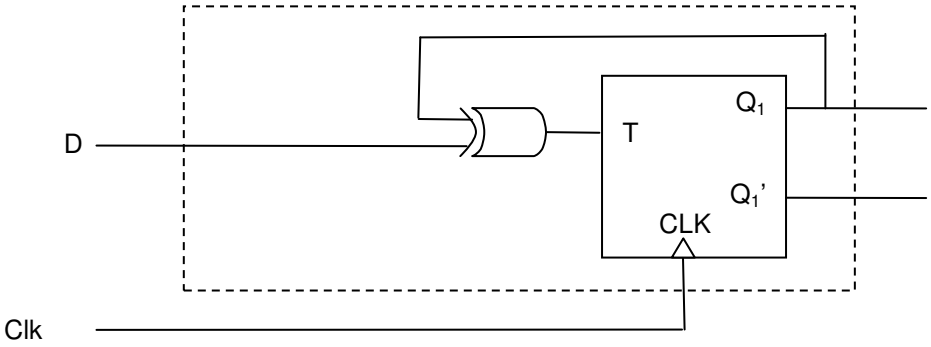


$D_4 = y_4y_3'y_2'y_1' + y_4'y_3y_2y_1$
$D_3 = y_4'y_3'y_2y_1 + y_4'y_3y_2' + y_4'y_3y_1'$
$D_2 = y_4'y_2'y_1 + y_4'y_2y_1'$
$D_1 = y_4'y_1' + y_3'y_2'y_1'$

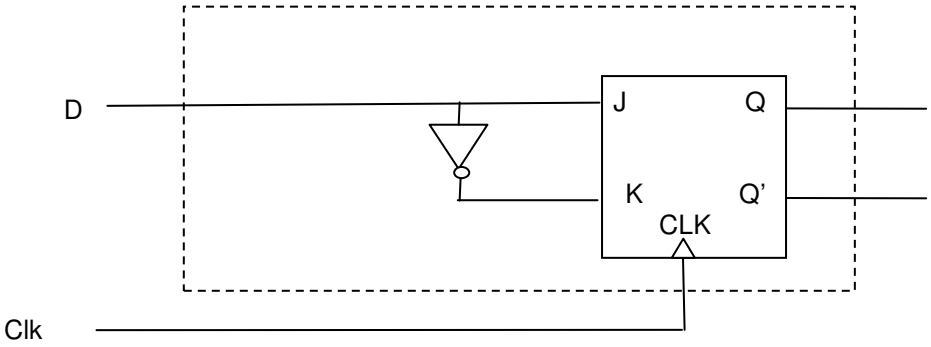
Step 5) Schematic:



b) To design using T flip-flop, build a D flip-flop from T- flip-flop as shown below and use it to re-implement the circuit from section a.



c) To design using JK flip-flop, build a D flip-flop using JK-FF as shown below and use it to re-implement the circuit from section a.



2U. Design a binary counter that counts from 2 to 7 and then restarts from 2:

a) positive-edge-triggered D flip-flops.
b) positive-edge-triggered T flip-flops.
c) positive-edge-triggered JK flip-flops.

Solution:

3S. Design a synchronous FSM that performs present state and next state in accordance to the following table..

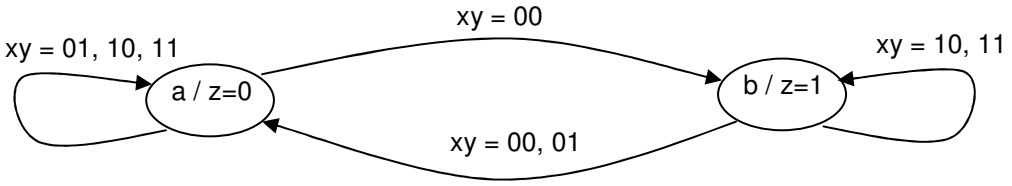
Present State	Next State for respective input (XY) 00 01 10 11	Output Z
a	b a a a	0
b	a a b b	1

a) Derive the minimum excitation and output equations for a full-encoded design using D flip-flops, and show the implementation.
b) Derive the minimum excitation and output equations for a one-hot-encoded design using D flip-flops, and show the implementation.

Solution:

Part a):
Given→ X, Y are inputs; two states a & b; output Z.

Step 1 - State diagram:



Step 2. Two states → 1 FF for full encoding.

Step 3 Assign a unique code for each state:
(State variable is Q.)

a → Q=0
b → Q=1

Step 4. Create a PS/NS table:
(A D FF is required and there is only one Moore-machine output.)

PS/NS Table:

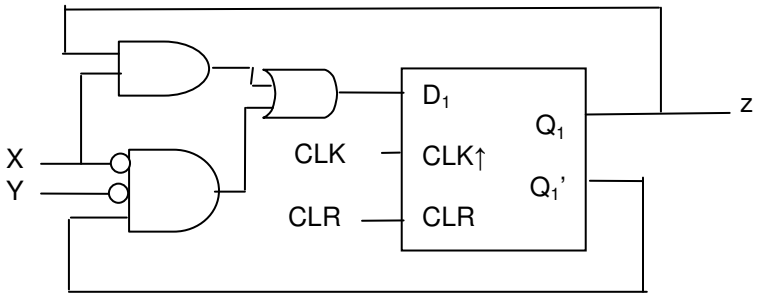
Input X	Input Y	PS Q	NS Q^+	$D=Q^+$ D	Output Z
0	0	0	1	1	0
0	0	1	0	0	1
0	1	0	0	0	0
0	1	1	0	0	1
1	0	0	0	0	0
1	0	1	1	1	1
1	1	0	0	0	0
1	1	1	1	1	1

XY \ Q	0	1
00	1	0
01	0	0
11	0	1
10	0	1

$D_1 = X.Q + X'Y'Q'$

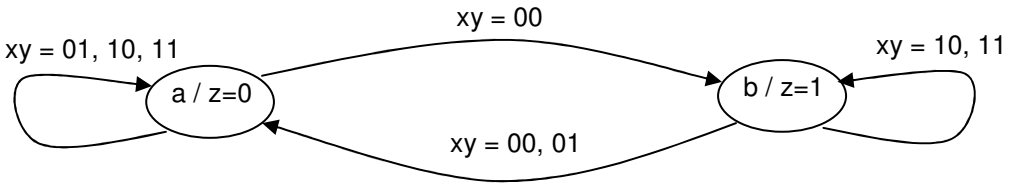
XY \ Q	0	1
00	0	1
01	0	1
11	0	1
10	0	1

Z=Q



Part b):
Give→ X, Y are inputs; two states a & b; output Z.

Step 1 - State Diagram



Step 2. Two states → 2 FF for one-hot encoding

Step 3 Assign a unique code for each state:
(State Variable is Q_1 & $Q_{2.}$)

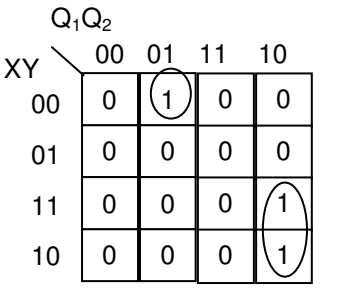
a → $Q_1Q_2 = 01$
b → $Q_1Q_2 = 10$

Step 4. Create a PS/NS table:
(D FFs are required and we only have one Moore-machine output.)

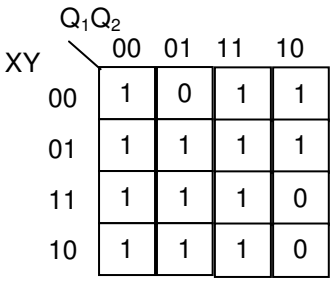
PS/NS Table:

Input		PS		NS		$D=Q^+$		Output
X	Y	Q_1	Q_2	Q_1^+	Q_2^+	D_1	D_2	Z
0	0	0	1	1	0	1	0	0
0	0	1	0	0	1	0	1	1
0	1	0	1	0	1	0	1	0
0	1	1	0	0	1	0	1	1
1	0	0	1	0	1	0	1	0
1	0	1	0	1	0	1	0	1
1	1	0	1	0	1	0	1	0
1	1	1	0	1	0	1	0	1

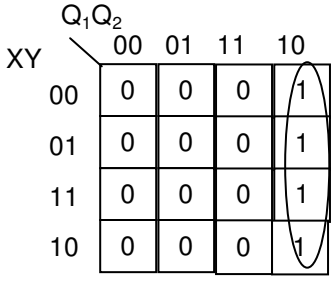
Note: Invalid states will be directed to $Q_1Q_2 = 01$ state with output Z=0



$D_1 = X.Q_1.Q_2'. + X'.Y'.Q_1'.Q_2$

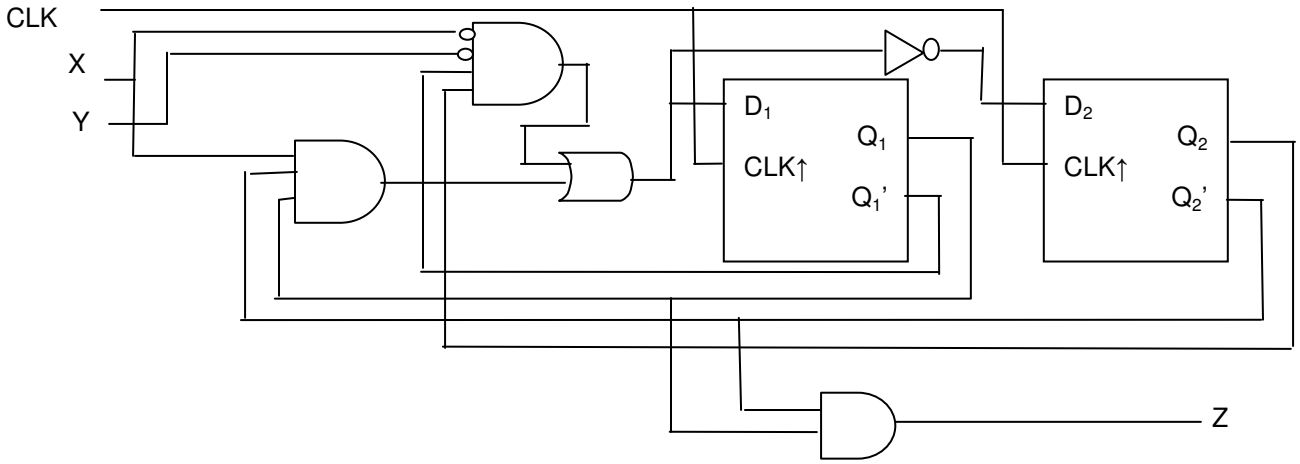


$D_2 = D_1'$



$Z = Q_1.Q_2'$

Step 5. Schematics



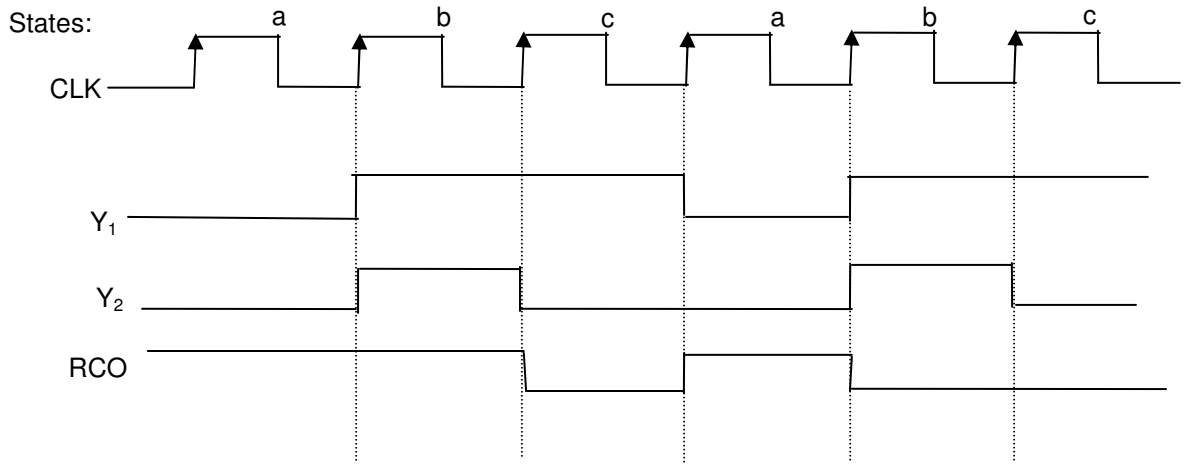
3U. Design a synchronous FSM that performs present state and next state in accordance to the following table..

Present State (PS)	Next State for respective input (XY) 00 01 10 11	Output Z
a	a b a b	0
b	a b b a	1

a) Derive the minimum excitation and output equations for a full-encoded design using T flip-flops, and show the implementation.
b) Derive the minimum excitation and output equations for a one-hot-encoded design using T flip-flops, and show the implementation.

Solution:

4S. Design a synchronous FSM according to the following diagram using rising-edge D flip-flops. Direct any illegal states to state "a".



Solution:

Assign a unique code for each state assignment based on two state variables.

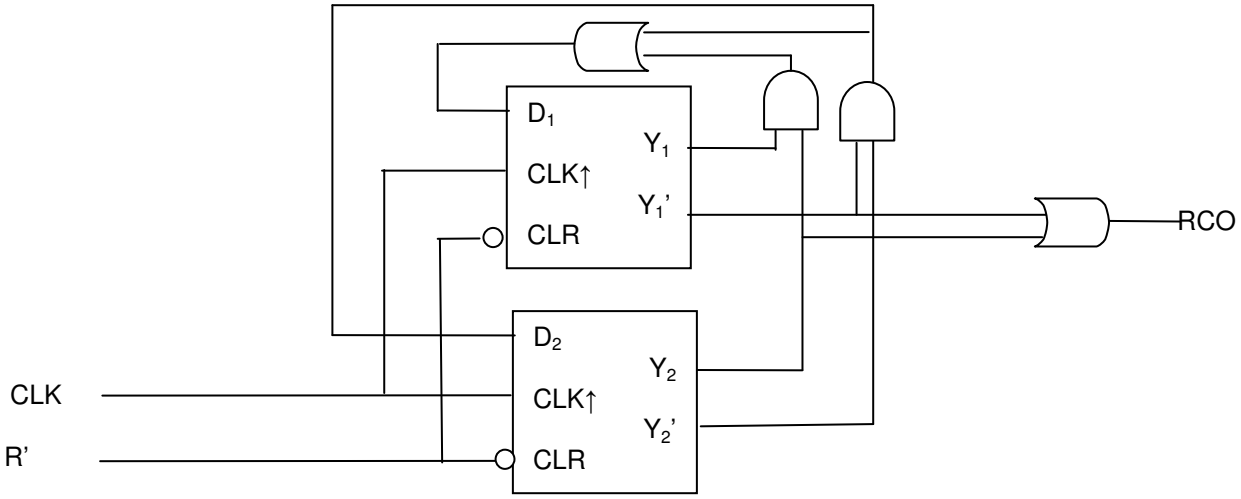
State	Y1	Y2
a	0	0
b	1	1
c	1	0

Note : Illegal state output is 01 and is directed to state "a".

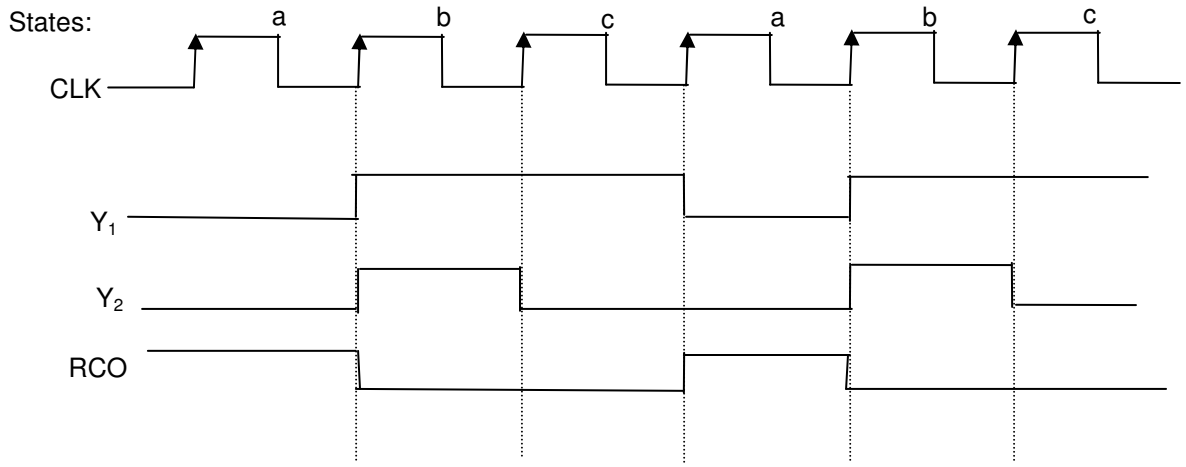
Excitation and output equations:

PS Y_1 Y_2	NS Y_1' Y_2'	$D=Y^+$ D_1 D_2	Output RCO
0 0	1 1	1 1	1
0 1	0 0	0 0	1
1 0	0 0	0 0	0
1 1	1 0	1 0	1

$D_1 = Y_1'.Y_2' + Y_1.Y_2$; $D_2 = Y_1'.Y_2'$; $RCO = Y_1' + Y_2$

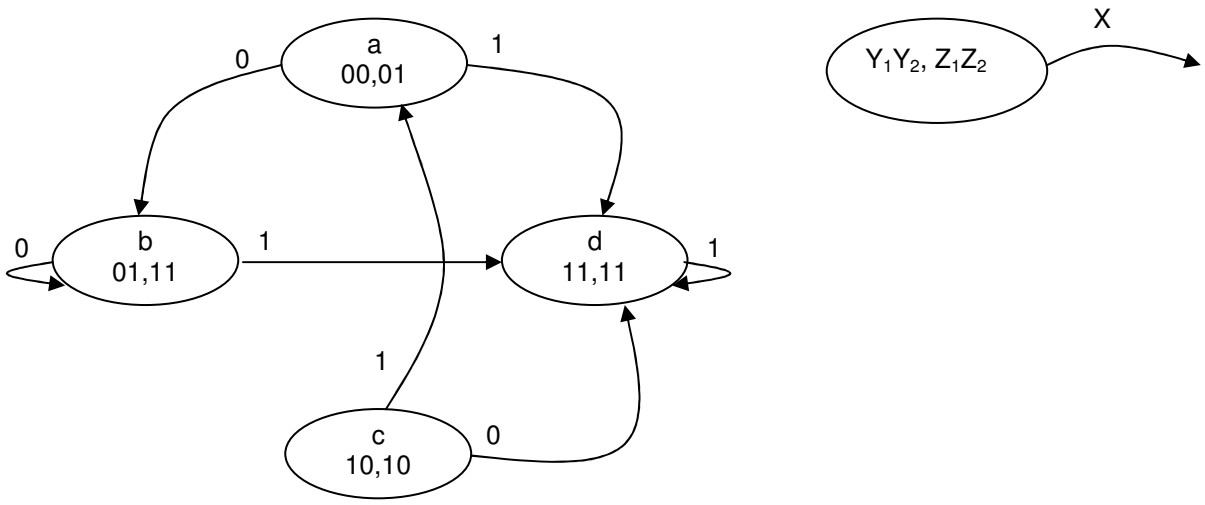


4U. Design a synchronous FSM according to the following diagram using rising-edge D flip-flops. Direct any illegal states to state "a".



Solution:

5S. Design a synchronous circuit using positive-edge-triggered T flip-flops for the following state diagram.



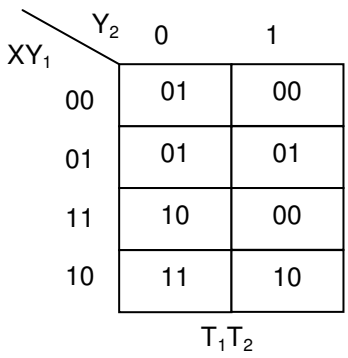
a) Derive the PS/NS table for the machine.
b) Derive the minimum excitation and external output equations.
c) Draw the circuit schematics.

Solution:

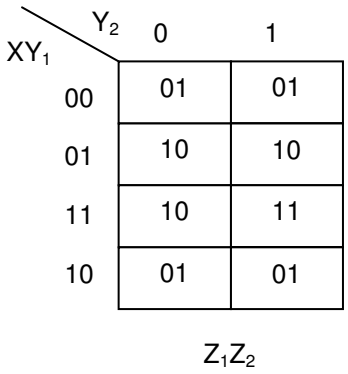
a) PS/NS table with state codes

PS			NS		Excitation Input		Output	
X	Y_1	Y_2	Y_1^+	Y_2^+	T_1	T_2	Z_1	Z_2
0	0	0	0	1	0	1	0	1
0	0	1	0	1	0	0	1	1
0	1	0	1	1	0	1	1	0
0	1	1	1	0	0	1	1	1
1	0	0	1	1	1	1	0	1
1	0	1	1	1	1	0	1	1
1	1	0	0	0	1	0	1	0
1	1	1	1	1	0	0	1	1

b) K-maps

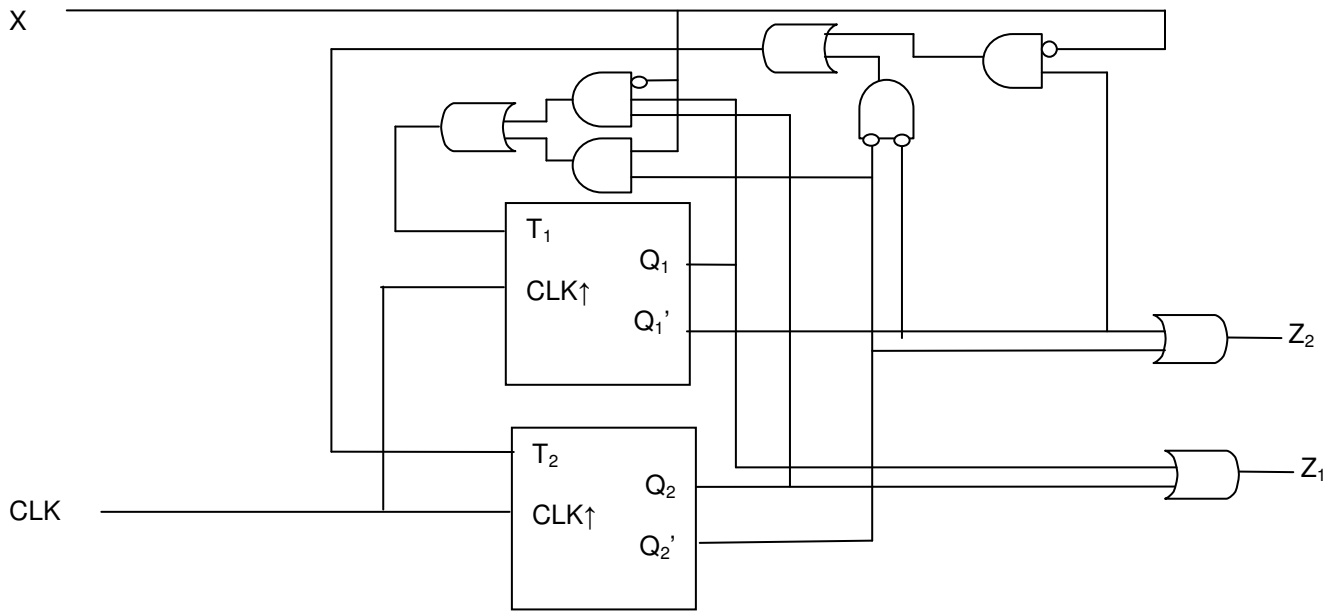


$T_1 = X.Y_2' + X.Y_1'$
$T_2 = Y_1Y_2' + X'.Y_1$

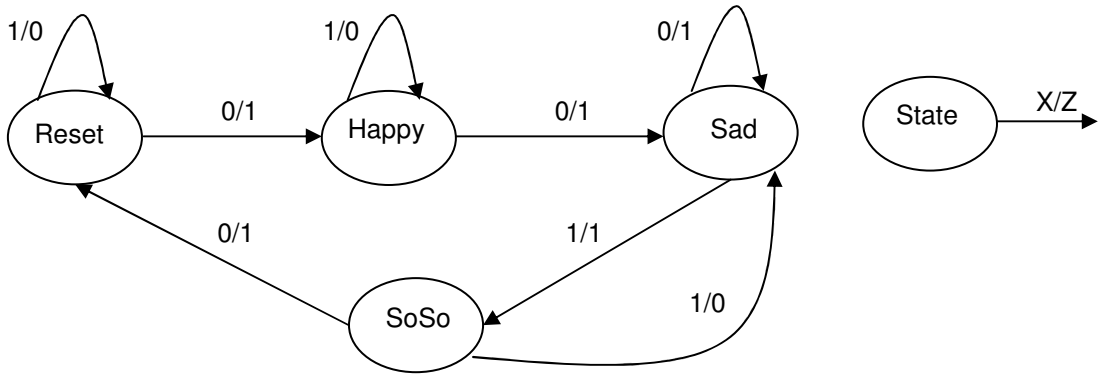


$Z_1 = .Y_1$
$Z_2 = Y_1' + X\,Y_2$

c) Schematic



5U. Design a circuit (FSM) using JK-ff that performs the function described by the following state:



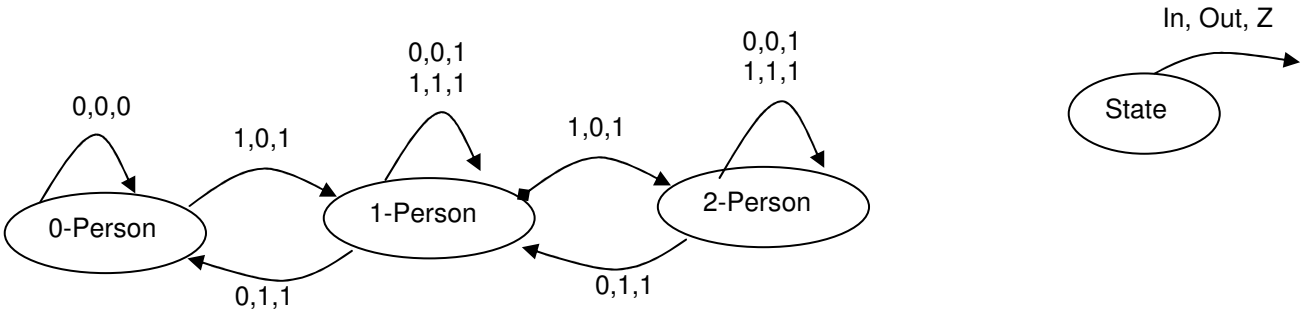
Solution:

6S. Use D flip-flops to design an automatic room light control that turns the light on only when someone is in the room. Light should be off when no one is in the room. Assume the room capacity is 2. (Door has two sensors, one detecting a person entering "in=1" and one detecting a person leaving "out=1"). Start with no one in the room.

Solution:

Step 1 – System Diagram

Step 2 – State Diagram



Step 3 – State Assignment

State Name	Q_1	Q_2
0-Person	0	0
1-Person	0	1
2-Person	1	0
Not used	1	1

Step 4 – Excitation and Output Equations

In	out	Q_1	Q_2	Q_1^+	Q_2^+	Z
0	0	0	0	0	0	0
0	0	0	1	0	1	1
0	0	1	0	1	0	1
0	0	1	1	x	x	x
0	1	0	0	x	x	0
0	1	0	1	0	0	1
0	1	1	0	0	1	1
0	1	1	1	x	x	x
1	0	0	0	0	1	0
1	0	0	1	1	0	1
1	0	1	0	x	x	1
1	0	1	1	x	x	x
1	1	0	0	0	0	0
1	1	0	1	0	1	1
1	1	1	0	1	0	1
1	1	1	1	x	x	x

In Out \ $Q_1 Q_2$	00	01	11	10
00	0	0	x	1
01	x	0	x	0
11	0	0	x	1
10	0	1	x	x

$D_1=Q_1^{+\ +}$= in'.out'.Q_1' + in.out.Q_1 + in.out'.Q_2

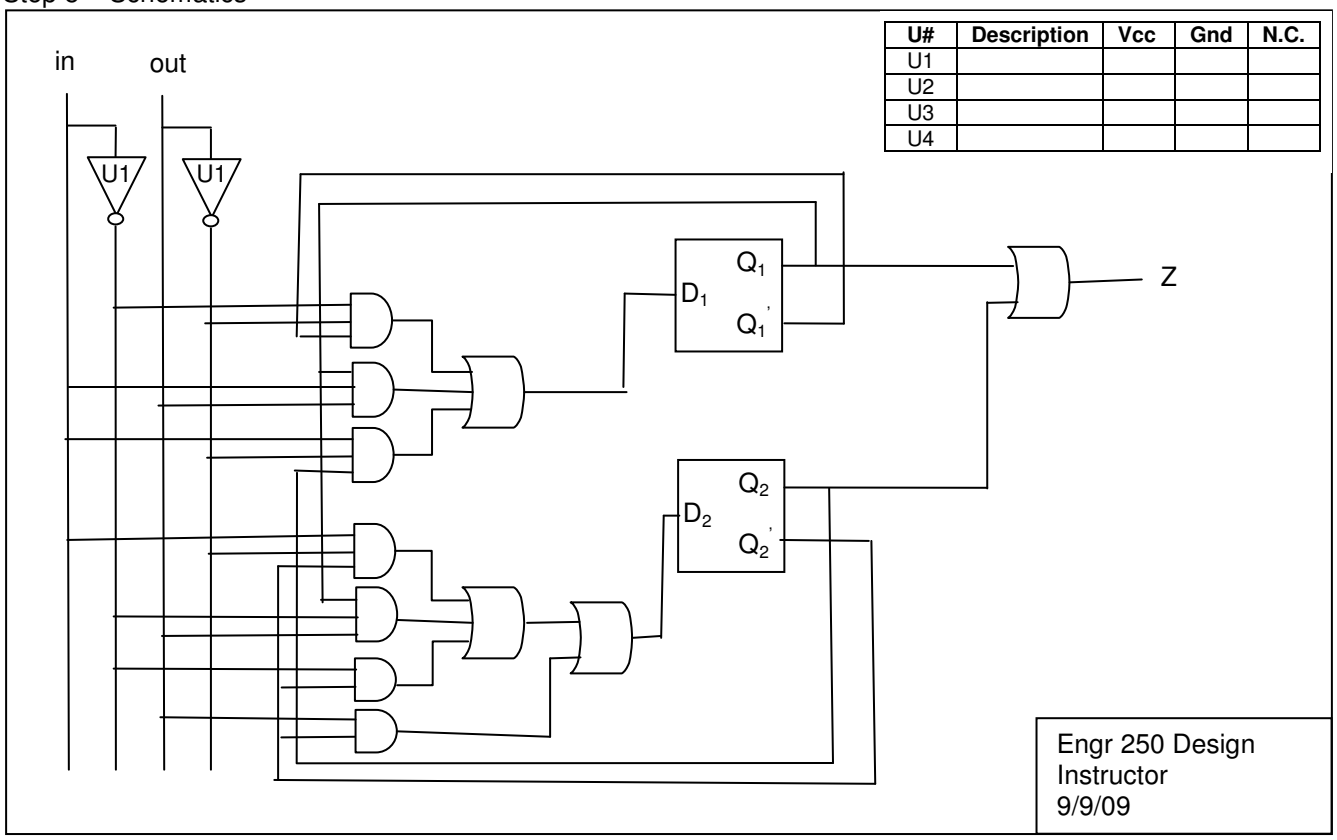
In Out \ $Q_1 Q_2$	00	01	11	10
00	0	1	x	0
01	0	1	x	1
11	0	1	x	0
10	1	0	x	x

$D_2=Q_2^{+\ +}$= in.out'.Q_2' + in'.out.Q_1 + in'.Q_2 + out.Q_2

In Out \ $Q_1 Q_2$	00	01	11	10
00	0	1	x	1
01	0	1	x	1
11	0	1	x	1
10	0	1	x	1

$Z= Q_1 + Q_2$

Step 5 – Schematics



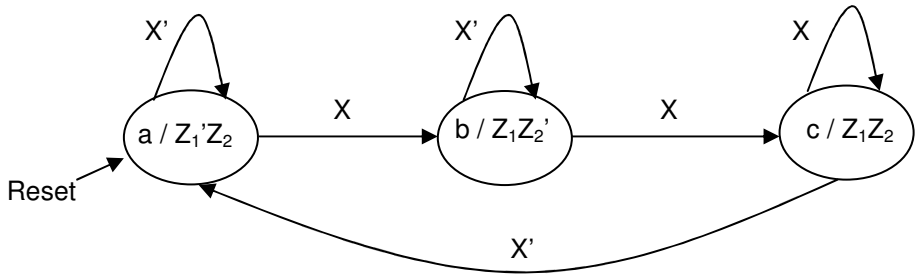
U#	Description	Vcc	Gnd	N.C.
U1				
U2				
U3				
U4				

6U. Design a locking system with 8 keys (0 through 6 and * keys). The system is locked normally and it unlocks only when the valid code sequence "*236" has been entered. Entering any other sequence locks the system. *Hint: start with a system diagram.*

Solution:

7S. Design a ILCW sequence detector using the classical design technique, full encoding and rising edge D flip flops. ILCW operation is described by the following state diagram when X is the input and Z_1 Z_2 is the 2-bit

output. *Show your work for each of the classical design steps.*



Solution:

Step 1. Organizing design ➔ State Diagram (Done)

Step 2. $2^{\#FF} \geq$ # of State ➔ # of State = 3

Step 3. Assign unique code/state

y1	y2	State
0	0	a
0	1	b
1	0	c
1	1	--

Step 4. Select D-ff, draw PS/NS table, Write Excitation equation and output equation

Input	PS		NS		Output		
x	y1	y2	$y1^+$	$y2^+$	Z1	Z2	
0	0	0	0	0	0	1	
0	0	1	0	1	1	0	
0	1	0	0	0	1	1	
0	1	1	-	-	-	-	You could also go to 00 state for the unassigned states
1	0	0	0	1	0	1	
1	0	1	1	0	1	0	
1	1	0	1	0	1	1	
1	1	1	-	-	-	-	You could also go to 00 state for the unassigned states

x \ y_1y_2	00	01	11	10
0	0	0	-	0
1	0	1	-	1

$D_1 = y_1^+ = x.y_2 + x.y_1$

x \ y_1y_2	00	01	11	10
0	0	1	-	0
1	1	0	-	0

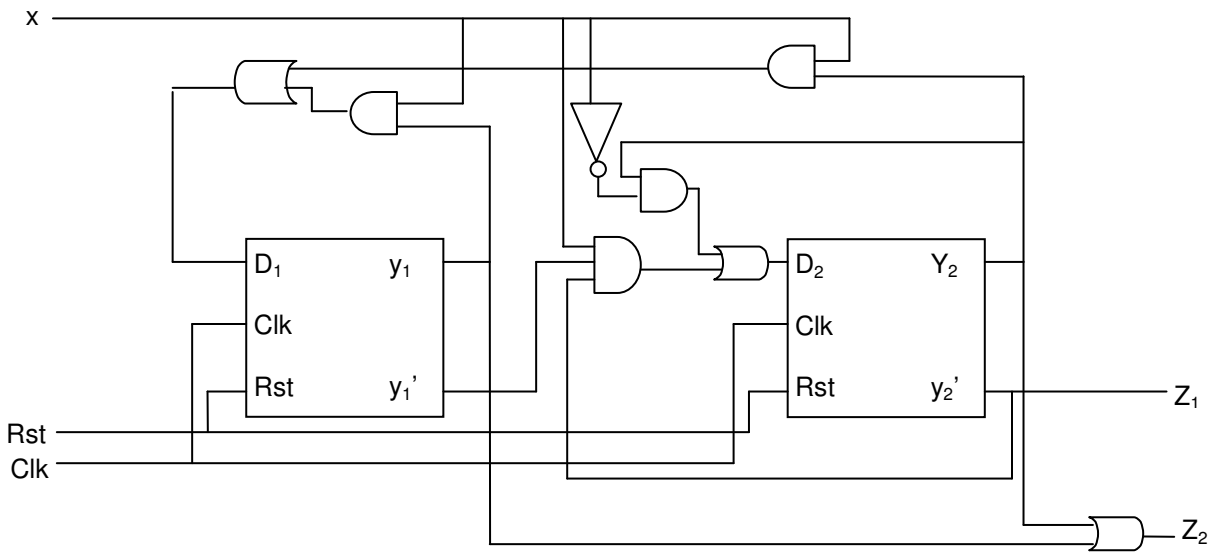
$D_2 = y_2^+ = x.y_1'.y_2' + x'.y_2$

x \ y_1y_2	00	01	11	10
0	0	1	-	1
1	0	1	-	1

$Z_1 = y_1 \oplus y_2$

x \ y_1y_2	00	01	11	10
0	1	0	-	1
1	1	0	-	1

$Z_2 = y_2'$



7U. Design a 3-bit gray code counter (000 → 001 → 011 → 010 →110 → 111 → 101 →100 →000 …) using D flip-flop.

Solution: