Cpr E 281 HW11 SOLUTION

ELECTRICAL AND COMPUTER ENGINEERING
IOWA STATE UNIVERSITY

Synchronous Sequential Circuits
Assigned Date: Thirteenth Week
Due Date: Monday, Nov. 28, 2016

P1. (10 points)

Design the modulo-5 counter which counts in the sequence 0, 1, 2, 3, 4, 0, 1, 2, 3, 4… when input *w*=1, and stops counting when *w*=0. Use D flip-flops in your circuit.

Solution:

Grading criteria

(3 points) state-assigned table

(3 points) simplified logic expressions for next state and output

(4 points) circuit diagram

State table

Present State	Next State w=0	Next State w=1	Output
A	A	B	0
B	B	C	1
C	C	D	2
D	D	E	3
E	E	A	4

State-assigned table

	Present State $y_2\ y_1\ y_0$	Next State w=0 $Y_2\ Y_1\ Y_0$	Next State w=1 $Y_2\ Y_1\ Y_0$	Output $z_2\ z_1\ z_0$
A	0 0 0	0 0 0	0 0 1	0 0 0
B	0 0 1	0 0 1	0 1 0	0 0 1
C	0 1 0	0 1 0	0 1 1	0 1 0
D	0 1 1	0 1 1	1 0 0	0 1 1
E	1 0 0	1 0 0	0 0 0	1 0 0

$$D_2 = Y_2 = \bar{w}\, y_2 + w\, y_1\, y_0$$

$$D_1 = Y_1 = \bar{w}\, y_1 + y_1 \bar{y_0} + w \bar{y_1}\, y_0 = w y_0 \oplus y_1$$

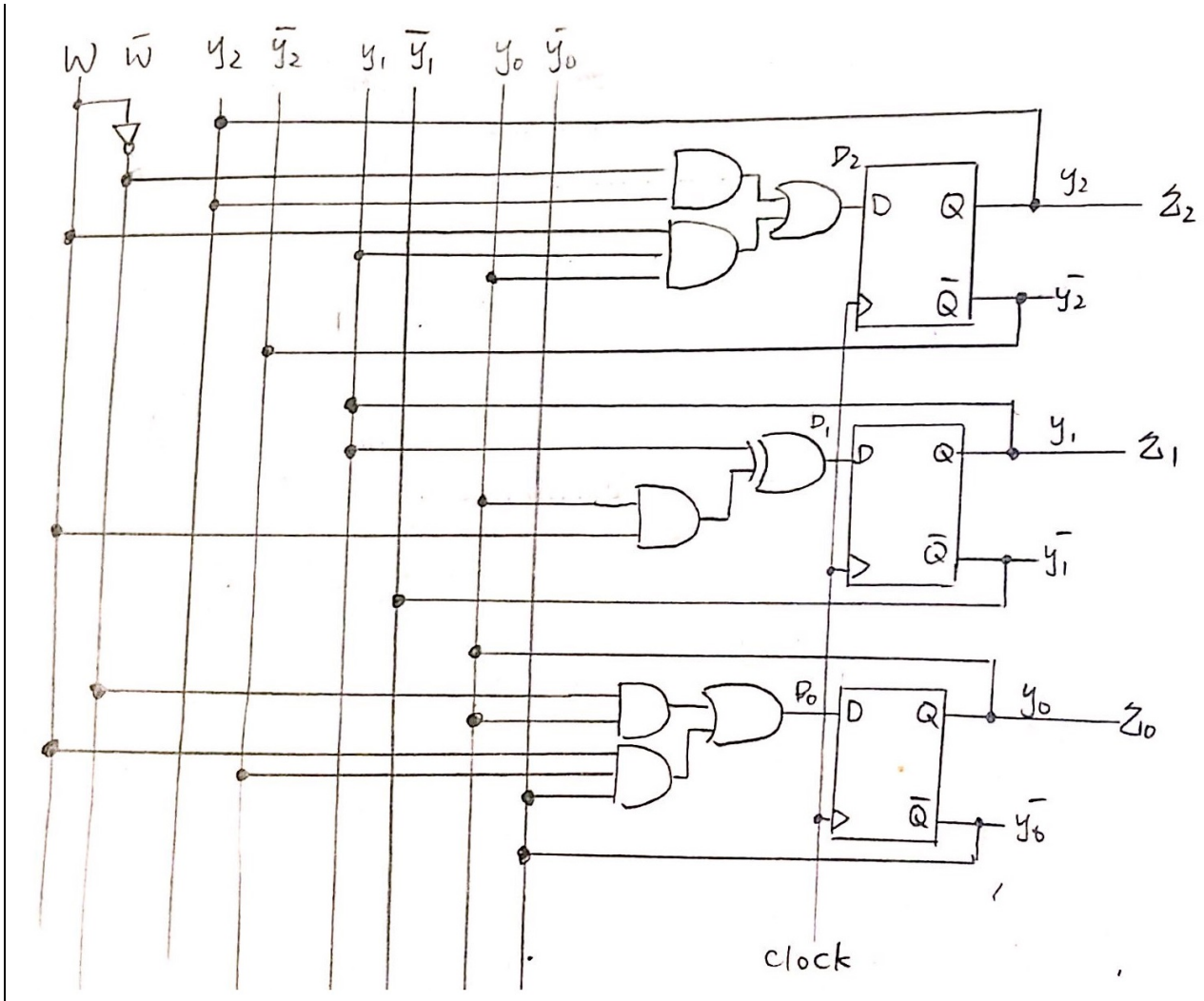
$$D_0 = Y_0 = \bar{w}\, y_0 + w \bar{y_2}\, \bar{y_0}$$

$$z_2 = y_2,\ z_1 = y_1,\ z_0 = y_0$$

Cpr E 281 HW11 SOLUTION

ELECTRICAL AND COMPUTER ENGINEERING
IOWA STATE UNIVERSITY

Synchronous Sequential Circuits
Assigned Date: Thirteenth Week
Due Date: Monday, Nov. 28, 2016



P2. (10 points)

Repeat P1 using T flip-flops.

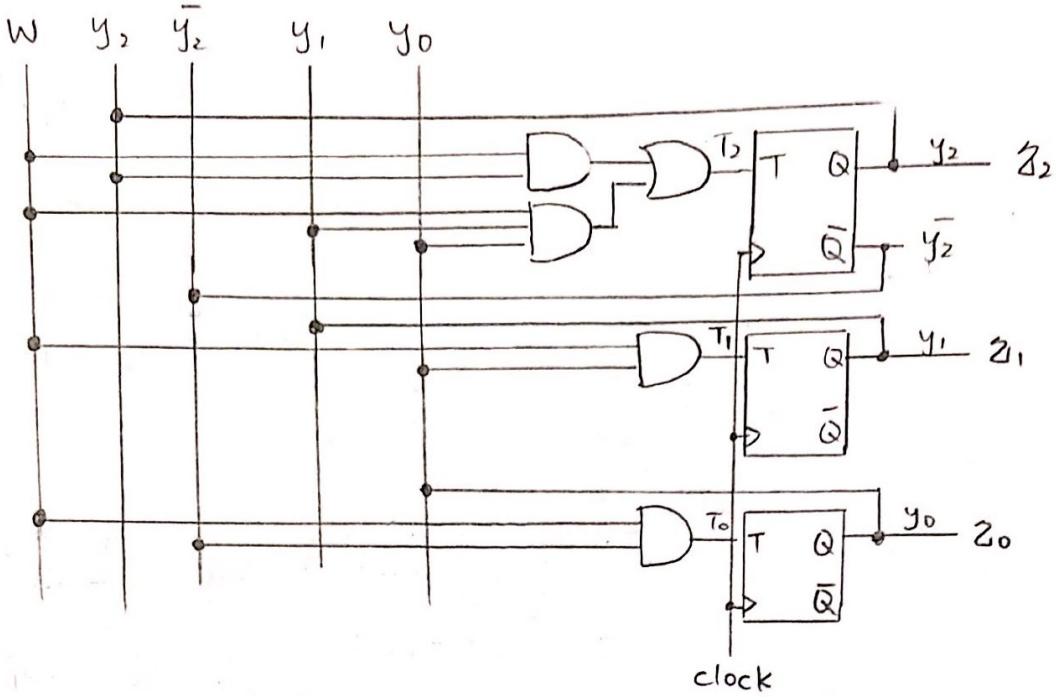
Solution:

Present State	Next State		T Flip-Flops	
	W=0	W=1	W=0	W=1
$y_2\ y_1\ y_0$	$Y_2\ Y_1\ Y_0$	$Y_2\ Y_1\ Y_0$	$T_2\ T_1\ T_0$	$T_2\ T_1\ T_0$
0 0 0	0 0 0	0 0 1	0 0 0	0 0 1
0 0 1	0 0 1	0 1 0	0 0 0	0 1 1
0 1 0	0 1 0	0 1 1	0 0 0	0 0 1
0 1 1	0 1 1	1 0 0	0 0 0	1 1 1
1 0 0	1 0 0	0 0 0	0 0 0	1 0 0.

$T_2 = W y_2 + W y_1 y_0$.

$T_1 = W y_0$

$T_0 = W \bar{y}_2$



P3. (10 points)

Repeat P1 using JK flip-flops.

Solution:

JK Flip-Flops

Present State $y_2\ y_1\ y_0$	Next State $w=0$ $Y_2\ Y_1\ Y_0$	Next State $w=1$ $Y_2\ Y_1\ Y_0$	$w=0$ J_2K_2	J_1K_1	J_0K_0	$w=1$ J_2K_2	J_1K_1	J_0K_0
0 0 0	0 0 0	0 0 1	0 X	0 X	0 X	0 X	0 X	1 X
0 0 1	0 0 1	0 1 0	0 X	0 X	X 0	0 X	1 X	X 1
0 1 0	0 1 0	0 1 1	0 X	X 0	0 X	0 X	X 0	1 X
0 1 1	0 1 1	1 0 0	0 X	X 0	X 0	1 X	X 1	X 1
1 0 0	1 0 0	0 0 0	X 0	0 X	0 X	X 1	0 X	0 X

X: don't care

$J_2 = wy_1y_0$

$K_2 = w$.

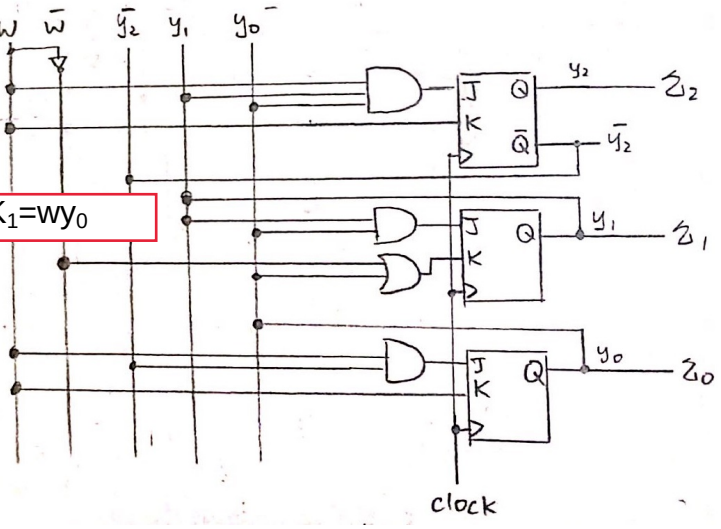
~~$J_1 = y_1y_0$~~

~~$K_1 = \bar{w} + y_0$~~

$J_1=K_1=wy_0$

$J_0 = w\bar{y}_2$

$K_0 = w$.



P4. (10 points)

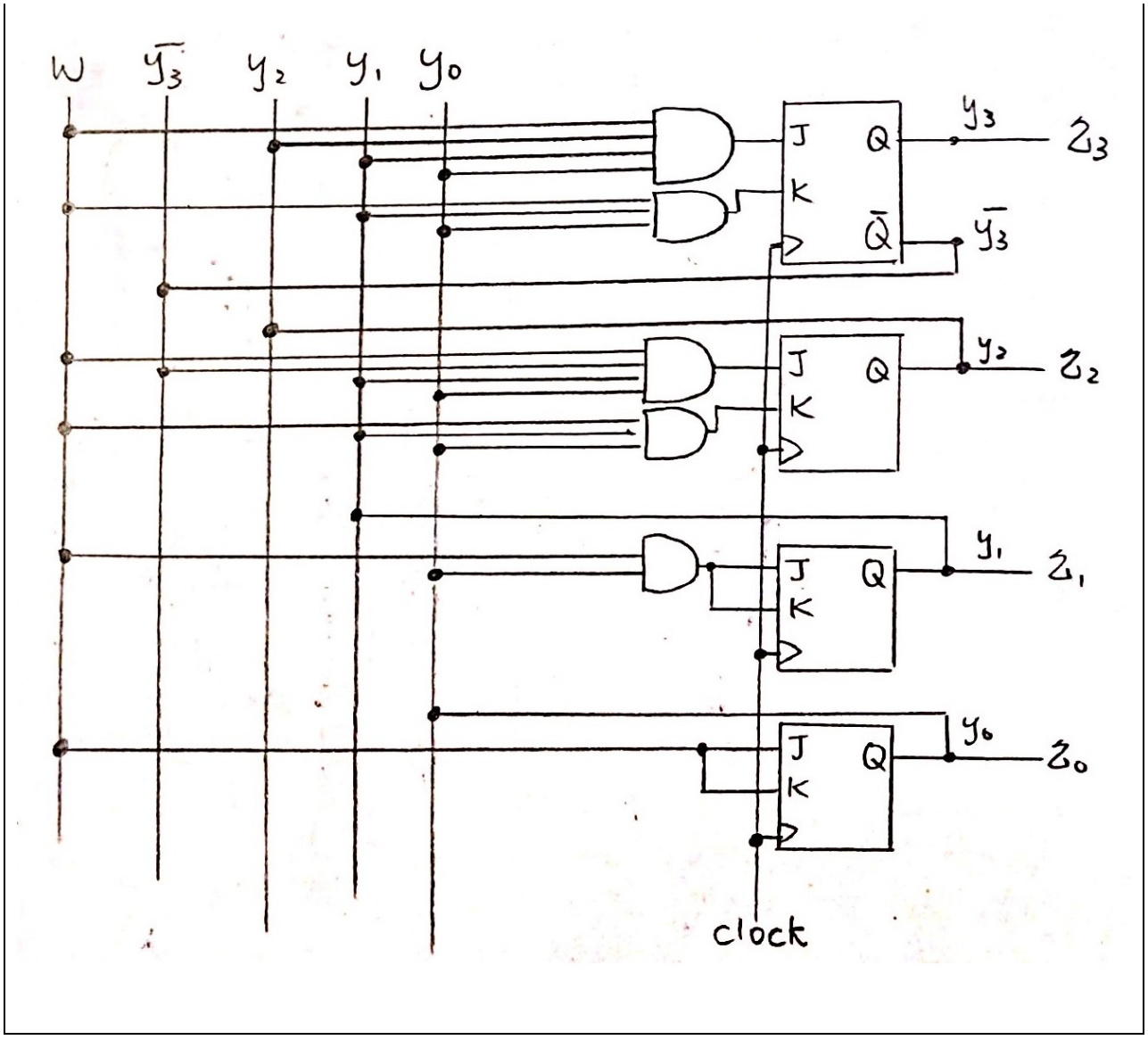
Design the modulo-12 up-counter which counts in the sequence 0, 1, 2, 3, …, 9, 10, 11, 0…. with input *w* as an enable. Use JK flip-flops in your circuit.

Solution:

	Present State	Next State w=0	Next State w=1	JK Flip-Flops w=0				JK Flip-Flops w=1			
	$y_3\ y_2\ y_1\ y_0$	$Y_3\ Y_2\ Y_1\ Y_0$	$Y_3\ Y_2\ Y_1\ Y_0$	J_3K_3	J_2K_2	J_1K_1	J_0K_0	J_3K_3	J_2K_2	J_1K_1	J_0K_0
0	0 0 0 0	0 0 0 0	0 0 0 1	0 x	0 x	0 x	0 x	0 x	0 x	0 x	1 x
1	0 0 0 1	0 0 0 1	0 0 1 0	0 x	0 x	0 x	x 0	0 x	0 x	1 x	x 1
2	0 0 1 0	0 0 1 0	0 0 1 1	0 x	0 x	x 0	0 x	0 x	0 x	x 0	1 x
3	0 0 1 1	0 0 1 1	0 1 0 0	0 x	0 x	x 0	x 0	0 x	1 x	x 1	x 1
4	0 1 0 0	0 1 0 0	0 1 0 1	0 x	x 0	0 x	0 x	0 x	x 0	0 x	1 x
5	0 1 0 1	0 1 0 1	0 1 1 0	0 x	x 0	0 x	x 0	0 x	x 0	1 x	x 1
6	0 1 1 0	0 1 1 0	0 1 1 1	0 x	x 0	x 0	0 x	0 x	x 0	x 0	1 x
7	0 1 1 1	0 1 1 1	1 0 0 0	0 x	x 0	x 0	x 0	1 x	x 1	x 1	x 1
8	1 0 0 0	1 0 0 0	1 0 0 1	x 0	0 x	0 x	0 x	x 0	0 x	0 x	1 x
9	1 0 0 1	1 0 0 1	1 0 1 0	x 0	0 x	0 x	x 0	x 0	0 x	1 x	x 1
10	1 0 1 0	1 0 1 0	1 0 1 1	x 0	0 x	x 0	0 x	x 0	0 x	x 0	1 x
11	1 0 1 1	1 0 1 1	0 0 0 0	x 0	0 x	x 0	x 0	x 1	0 x	x 1	x 1
12	1 1 0 0										
13	1 1 0 1										
14	1 1 1 0										
15	1 1 1 1										

Doing 5-variable K maps to get J_3K_3, J_2K_2, J_1K_1, and J_0K_0.

$$\Rightarrow \begin{cases} J_3 = w y_2 y_1 y_0 \\ K_3 = w y_1 y_0 \end{cases} \quad \begin{cases} J_2 = w \bar{y}_3 y_1 y_0 \\ K_2 = w y_1 y_0 \end{cases} \quad J_1 = K_1 = w y_0 \quad J_0 = K_0 = w$$


w
$\bar{y}_3$
y_2
y_1
y_0
J
K
Q
$\bar{Q}$
y_3
Z_3
$\bar{y}_3$
y_2
Z_2
y_1
Z_1
y_0
Z_0
clock

P5. (40 points)

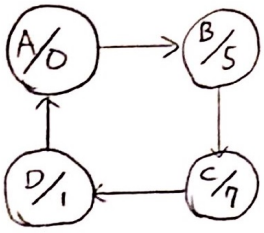
A counter has a special counting sequence: 0, 5, 7, 1, 0, 5, 7, 1, and so on. Design this counter with minimal number of states.

a) (5 points) Draw a state diagram for the counter.
b) (5 points) Construct a state-assigned table including the next state and output.
c) (10 points) Draw the circuit diagram for the counter using D flip-flops.
d) (10 points) Repeat (c) using T flip-flops.
e) (10 points) Repeat (c) using JK flip-flops.

Solution:

This counter does not require an enable. Students may have different state assignments from this solution which could also be a correct approach.

a)



(optional)

Present State	Next State	Output
A	B	0
B	C	5
C	D	7
D	A	1

b)

Present State $y_1\ y_0$	Next State $Y_1\ Y_0$	Output $z_2\ z_1\ z_0$
0 0	0 1	0 0 0
0 1	1 0	1 0 1
1 0	1 1	1 1 1
1 1	0 0	0 0 1

$Y_1 = y_1 \oplus y_0$

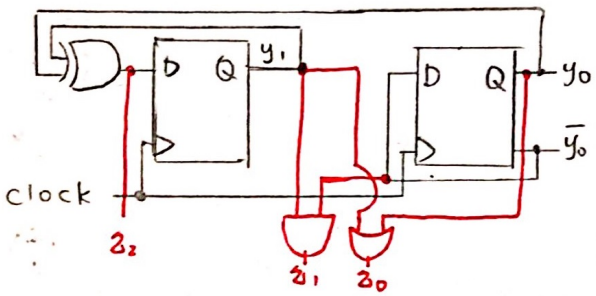
$Y_0 = \bar{y}_0$

$z_2 = y_1 \oplus y_0 = Y_1$

$z_1 = y_1 \bar{y}_0$

$z_0 = y_1 + y_0$

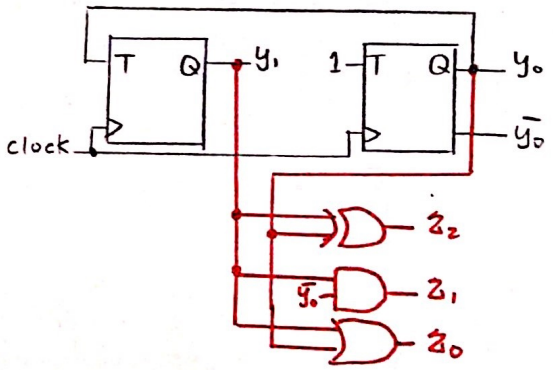
c)



d)

Present State y_1 y_0	Next State Y_1 Y_0	T Flip-Flops 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

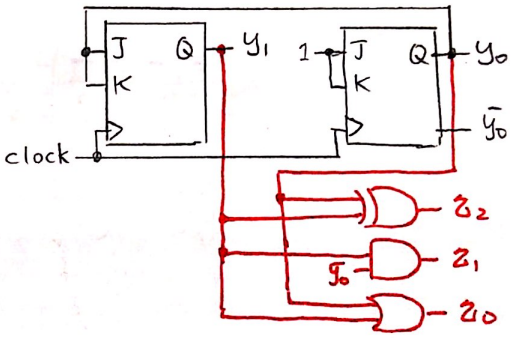
$T_1 = y_0$
$T_0 = 1$



e)

Present State y_1 y_0	Next State Y_1 Y_0	JK Flip-Flops $J_1 K_1$	$J_0 K_0$
0 0	0 1	0 X	1 X
0 1	1 0	1 X	X 1
1 0	1 1	X 0	1 X
1 1	0 0	X 1	X 1

$J_1 = y_0$, $K_1 = y_0$
$J_0 = 1$, $K_0 = 1$



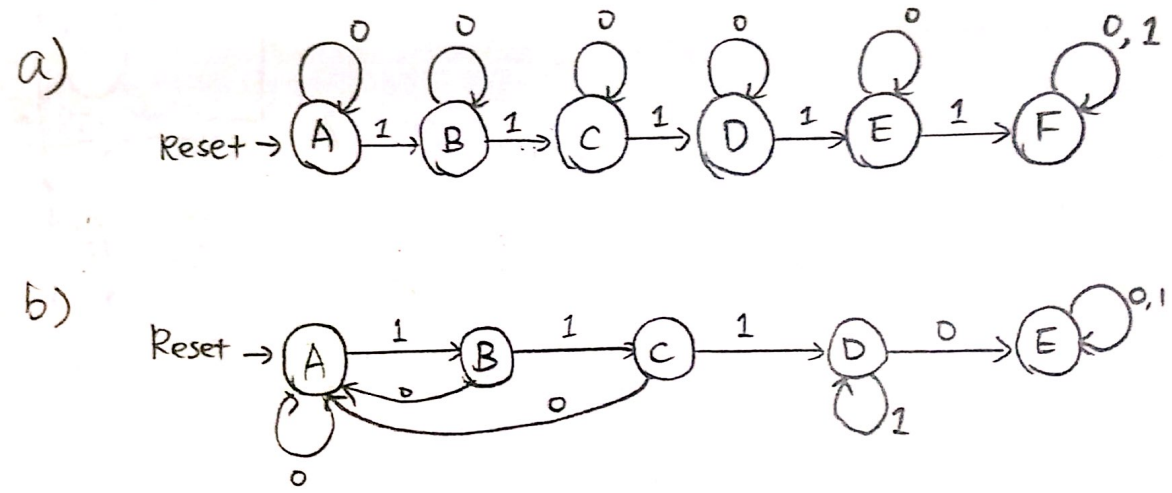
P6. (10 points)

Draw a state transition diagram for:

a) (5 points) A state machine that reads in a sequence of binary digits, one at a time, and stops when it has read in a total of five 1s (need not to be consecutive). To "stop" the machine, merely have it loop repeatedly in a final state.

b) (5 points) A state machine that stops when it has read in at least three consecutive 1s followed by a 0.

Solution:



P7. (10 points) Arbiter Circuits

The arbiter FSM defined in Section 6.8 (Figure 6.72) may cause device 3 to never get serviced if devices 1 and 2 continuously keep raising requests, so that in the Idle state it always happens that either device 1 or device 2 has an outstanding request. Modify the proposed FSM to ensure that device 3 will get serviced, such that if it raises a request, the device 1 and 2 will be serviced only once before the device 3 is granted its request.

Solution: