ECE 124 digital circuits and systems
Assignment #9

Q1: Consider the state table shown below for a circuit with one input x and one output z.

Current State	Next State $x=0$	Next State $x=1$	Output z
A	C	D	0
B	B	A	0
C	D	A	0
D	C	B	1

Use one hot encoding and derive a suitable circuit using D flip-flops that implements this state table.

Solution:
One hot encoding implies A is 0001, B is 0010, C is 0100 and D is 1000. This leads to the state table given by

Current state	**Next state** $x=0$	**Next state** $x=1$	**Outputs** $x=0$	**Outputs** $x=1$
$q_3q_2q_1q_0$	$q_3q_2q_1q_0$	$q_3q_2q_1q_0$	z	z
0001	0100	1000	0	0
0010	0010	0001	0	0
0100	1000	0001	0	0
1000	0100	0010	1	1

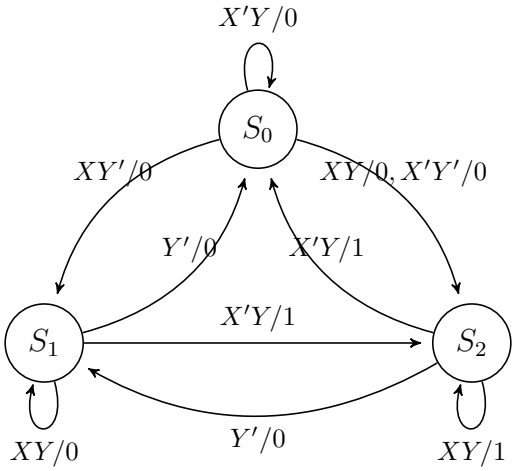
The DFF equations can be written down directly.

$$\begin{aligned} d_3 &= \overline{x}q_2 + xq_0 \\ d_2 &= \overline{x}q_3 + \overline{x}q_0 \\ d_1 &= \overline{x}q_1 + xq_3 \\ d_0 &= xq_2 + xq_1 \end{aligned}$$

The circuit output is given as

$$z = q_3$$

Q2: Implement the following state diagram using D flip-flops and one-hot encoding. Write down the necessary flip-flop input equations and output equations by inspecting the state diagram directly. Note that all possible input conditions are shown for the transitions leading away from every state.



Solution:
Simply look at incoming edges for each state to get the flip flop input equations:

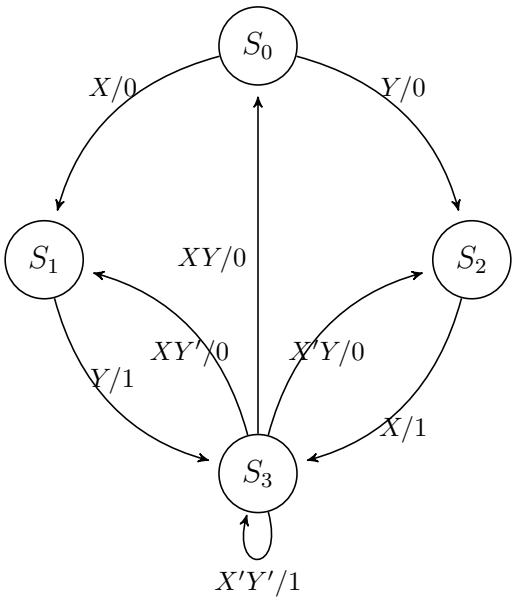
$$\begin{aligned} d_2 &= XYS_2 + X'YS_1 + X'Y'S_0 \\ d_1 &= XYS_1 + Y'S_2 + XY'S_0 \\ d_0 &= X'YS_0 + Y'S_1 + X'YS_2 \end{aligned}$$

The circuit output is a Mealy output given by

$$z = XYS_2 + X'YS_2 + X'YS_1$$

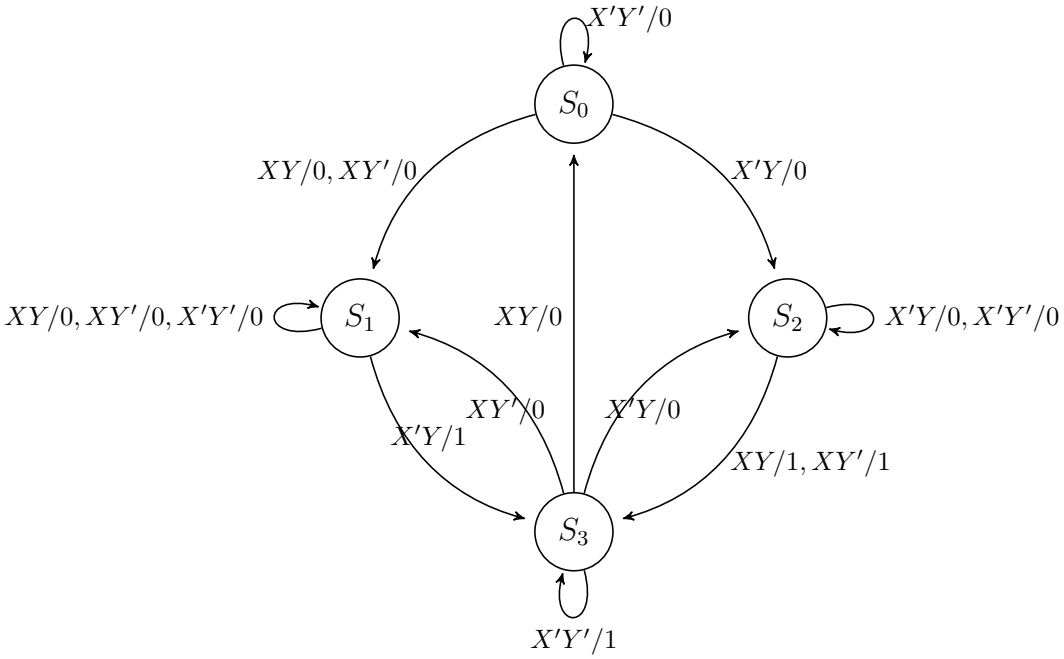
Note that I've written the flip flop inputs in terms of the symbolic state. You'd get the same thing if you did a state assignment; e.g., S_0 is 001, S_1 is 010, and S_2 is 100 and then called the flip flop outputs $q_2q_1q_0$ — the S_i would simply get replaced by q_i.

Q3: Consider the following state diagram for a circuit with two inputs X and Y and one output Z. It was drawn in a weird way; If $X = 1$ and $Y = 1$, then X takes precedence. Further, self-loops are not shown.



Clarify the state diagram by adding self loops where needed. Clarify the input labeling on the edges so that everything is very clear. Finally, using your clarified state diagram, implement the circuit using D flip-flops and one-hot encoding.

Solution:
Basically we are being asked to redraw the state diagram such that all edges are shown on the diagram without any confusion. Further, we want to remove the necessity of stating "X has precedence over Y". The solution is to examine each state and ensure that all 4 potential exit conditions (4 since there are two inputs X and Y) are shown. Here is the completed and clarified state diagram:



Given the completed stated diagram, we can write down DFF input equations assuming one hot encoding by looking at the incoming edges to each state:

$$\begin{aligned} d_3 &= XYS_2 + XY'S_2 + X'Y'S_3 + X'YS_1 \\ d_2 &= X'YS_3 + X'YS_2 + X'Y'S_2 + X'YS_0 \\ d_1 &= XYS_1 + XY'S_1 + X'Y'S_1 + XYS_0 + XY'S_0 \\ d_0 &= XYS_3 + X'Y'S_0 \end{aligned}$$

Note that the DFF input equations could likely be minimized a bit. The output is

$$Z = XYS_2 + XY'S_2 + X'Y'S_3 + X'YS_1$$

Q4: Consider the following state table for a circuit with one input x and one output z. The output is a function of the state only and does not depend on the input. Perform state minimization to reduce the required number of states. Derive a circuit for the reduced state table which requires the minimum number of flip-flops. Use D flip-flops.

Current State	Next State $x=0$	Next State $x=1$	Output z
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0

Solution:
We state with an implication chart — this one shows definitely equivalent, definitely not equivalent and conditions.

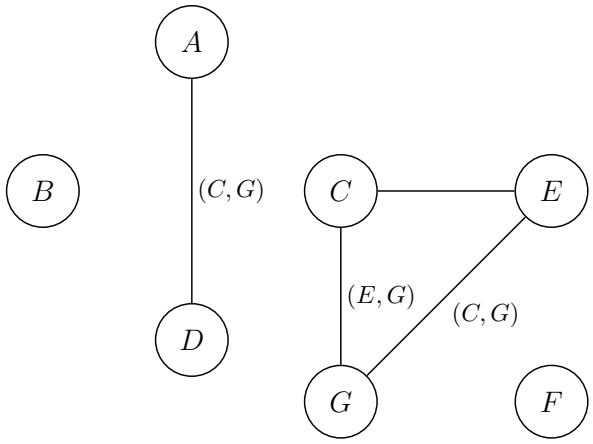
B	(B,D),(C,F)					
C	X	X				
D	(C,G)	(F,G)	X			
E	X	X	✓	X		
F	X	X	(E,F),(D,E)	X	(C,D)	
G	X	X	(E,G)	X	(C,G)	(E,F),(D,G)
	A	*B*	*C*	*D*	*E*	*F*

Completed implication chart after examining conditions:

B	(B,D) X,(C,F)					
C	X	X				
D	(C,G)	(F,G) X	X			
E	X	X	✓	X		
F	X	X	(E,F) X,(D,E)	X	(C,D) X	
G	X	X	(E,G)	X	(C,G)	(E,F) X,(D,G)
	A	*B*	*C*	*D*	*E*	*F*

Note that this is interesting because we have some conditions that we cannot mark as either true or false because they depend on each other.

The merger diagram for our implication chart:



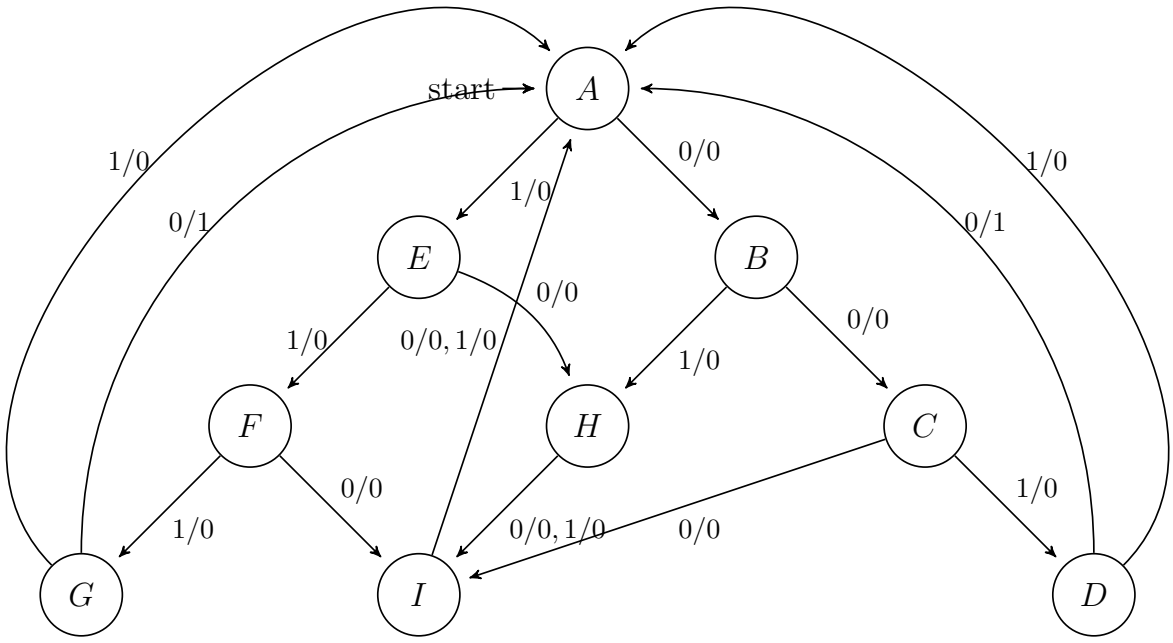
We can do the groupings (A,D) and (C,E,G). With check the condtions — all conditions are satisfied with this grouping so it's fine.

The reduced state diagram is (use state A to represent both A and D and use state C to represent states C, E, and G):

Current State	Next State $x=0$	Next State $x=1$	Output z
A	B	C	1
B	A	F	1
C	F	C	0
F	C	A	0

Q5: A Mealy sequential circuit has one input x and one output z. During four consecutive clock cycles, a sequence of four values of x are applied. The output is $z = 1$ when either the input sequence 0010 or 1110 is detected, otherwise $z = 0$. The circuit resets after four input bits are applied, ready for the next four bit sequence. Derive a *minimized* state table for the sequential circuit.

Solution:
The first step is to get a valid state diagram for the verbal problem description. Here's one:



The resulting state table:

Current state	**Next state**		**Output**	
	$x = 0$	$x = 1$	$x = 0$	$x = 1$
A	B	E	0	0
B	C	H	0	0
C	I	D	0	0
D	A	A	1	0
E	H	F	0	0
F	I	G	0	0
G	A	A	1	0
H	I	I	0	0
I	A	A	0	0

The implication chart:

B	(B,C),(E,H)							
C	(B,I),(D,E)	(C,I),(D,H)						
D	X	X	X					
E	(B,H),(E,F)	(C,H),(H,F)	(H,I),(D,F)	X				
F	(B,I),(E,G)	(C,I),(G,H)	(D,G)	X	(H,I),(F,G)			
G	X	X	X	✓	X	X		
H	(B,I),(E,I)	(C,I),(H,I)	(D,I)	X	(H,I),(F,I)	(G,I)	X	
I	(A,B),(A,E)	(A,C),(A,H)	(A,I),(A,D)	X	(A,H),(A,F)	(A,I),(A,G)	X	(A,I)
	A	*B*	*C*	*D*	*E*	*F*	*G*	*H*

Completed implication chart after examining conditions:

B	(B,C) X,(E,H)							
C	(B,I),(D,E) X	(C,I),(D,H) X						
D	X	X	X					
E	(B,H) X,(E,F)	(C,H) X,(H,F)	(H,I),(D,F) X	X				
F	(B,I),(E,G) X	(C,I) X,(G,H)	(D,G) ✓	X	(H,I),(F,G) X			
G	X	X	X	✓	X	X		
H	(B,I) X,(E,I)	(C,I) X,(H,I)	(D,I) X	X	(H,I),(F,I) X	(G,I) X	X	
I	(A,B) X,(A,E)	(A,C) X,(A,H)	(A,I),(A,D) X	X	(A,H),(A,F) X	(A,I),(A,G) X	X	(A,I) X
	A	*B*	*C*	*D*	*E*	*F*	*G*	*H*

Looking right at the implication chart, we can see that the two possible groupings are states D and G and states C and F. The grouping of C and F has a condition that D and G are grouped which is true.

So, we can reduce this by two states. The reduced state table is:

Current state	**Next state**		**Output**	
	$x = 0$	$x = 1$	$x = 0$	$x = 1$
A	B	E	0	0
B	C	H	0	0
C	I	D	0	0
D	A	A	1	0
E	H	C	0	0
H	I	I	0	0
I	A	A	0	0

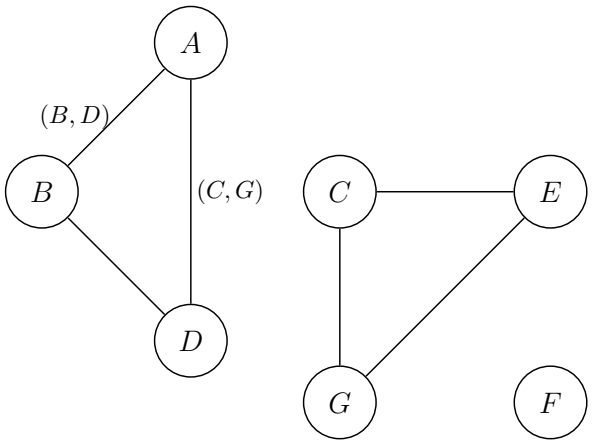
Q6: Shown below is a state table. Note that in some situations, the next state is left as *unspecified*; an unspecified state is like a don't care condition and can be matched appropriately. Derive an equivalent state table with a minimum number of states.

Current State	Next State $x=0$	Next State $x=1$	Output z $x=0$	Output z $x=1$
A	B	C	0	0
B	D	-	0	0
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	1

Solution:
The implication chart:

	A	B	C	D	E	F
B	(B,D) ✓					
C	X	X				
D	(C,G) ✓	✓	X			
E	X	X	✓	X		
F	X	X	(E,F),(D,E) X	X	(C,D) X	
G	X	X	✓	X	✓	(E,F) X

The merger diagram:



The groupings are states (A, B, D) and (C, E, G) and F. Conditions are satisfied.

The reduced state table is:

Current State	Next State $x=0$	Next State $x=1$	Output z $x=0$	Output z $x=1$
A	A	C	0	0
C	F	C	0	1
F	C	A	0	1

Q7: Shown below is a state table. Note that in some situations, the next state is left as *unspecified*; an unspecified state is like a don't care condition and can be matched appropriately. Derive an equivalent state table with a minimum number of states.

Current State	Next State $x=0$	Next State $x=1$	Output z $x=0$	Output z $x=1$
A	B	C	0	0
B	D	-	0	1
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	0

Solution:
The implication chart:

	A	*B*	*C*	*D*	*E*	*F*
B	X					
C	X	(D,F) X				
D	(C,G) X	X	X			
E	X	(D,F) X	✓	X		
F	X	(D,E) X	(E,F),(D,E) X	X	(C,D) X	
G	(B,F) X	X	X	(B,F) X	X	X

The only states that can be grouped at C and E. The reduced state table is:

Current State	Next State $x=0$	Next State $x=1$	Output z $x=0$	Output z $x=1$
A	B	C	0	0
B	D	-	0	1
C	F	C	0	1
D	B	G	0	0
F	C	D	0	1
G	F	-	0	0

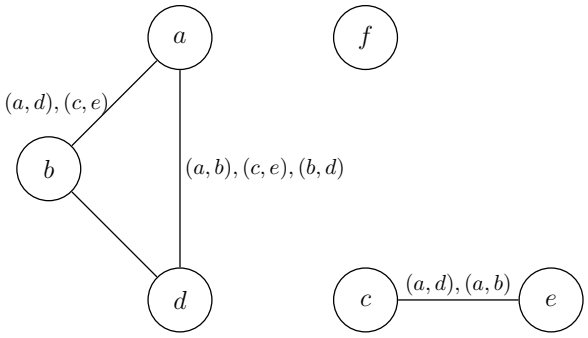
Q8: Reduce the following state table to a minimum number of states.

Current State	Next State $xy=00$	$xy=01$	$xy=11$	$xy=10$	Output z
a	a	c	e	d	0
b	d	e	e	a	0
c	e	a	f	b	1
d	b	c	c	b	0
e	c	d	f	a	1
f	f	b	a	d	1

Solution:
The implication chart:

b	(a,d),(c,e)				
c	X	X			
d	(a,b),(c,e),(b,d)	(c,e),(a,b)	X		
e	X	X	(a,d),(a,b)	X	
f	X	X	(e,f),(a,b),(a,f) X,(b,d)	X	(c,f),(b,d),(a,f) X,(a,d)
	a	b	c	d	e

The conditions remaining cannot be marked one way or another. Draw the merger diagram:



If we make the groupings (a, b, d) and (c, e) and leave f alone, then all the conditions are correct. The reduced state table:

Current State	Next State $xy=00$	$xy=01$	$xy=11$	$xy=10$	Output z
a	a	c	c	a	0
c	c	a	f	a	1
f	f	a	a	a	1

Q9: Reduce the following state table to a minimum number of states.

Current State	Next State $xy = 00$	$xy = 01$	$xy = 11$	$xy = 10$	Output z
a	b	i	c	g	0
b	b	c	f	g	0
c	h	d	d	f	1
d	h	c	e	g	1
e	b	c	i	g	0
f	f	i	i	k	0
g	j	k	g	h	0
h	e	f	c	g	0
i	i	i	i	d	0
j	b	f	c	g	0
k	a	c	e	d	1

Solution:
The implication chart (big one) showing differences in outputs and conditions.

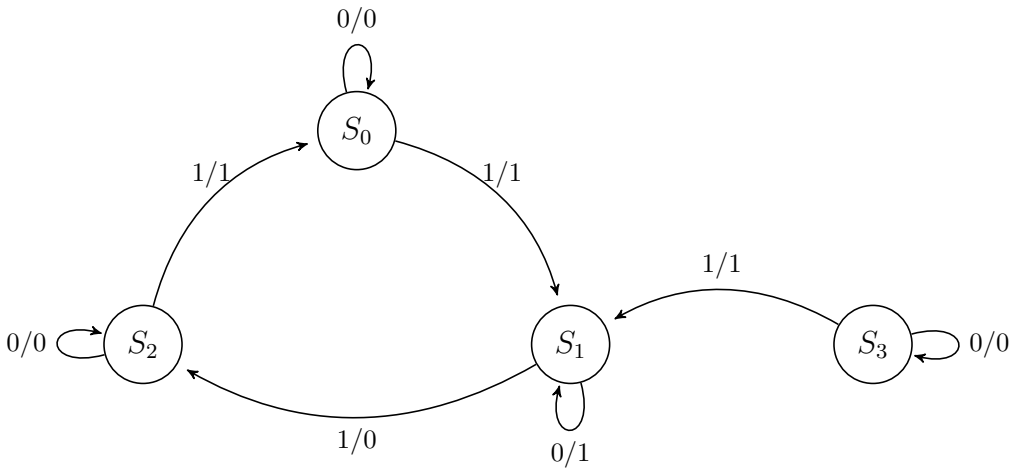
	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*
b	(c,i),(c,f)									
c	X	X								
d	X	X	(d,e),(f,g)							
e	(c,i)	(f,i)	X	X						
f	(b,f),(c,i),(g,k)	(c,i),(f,i),(g,k)	X	X	(b,f),(c,i),(g,k)					
g	(b,j),(k,i),(c,g),(g,h)	(b,j),(c,k),(f,g),(h,g)	X	X	(b,j),(c,k)(g,i),(g,h)	(f,j),(i,k),(g,i),(h,k)				
h	(b,e),(f,i)	(b,e),(c,f)	X	X	(b,e),(c,f),(c,i)	(f,e),(f,i),(c,i),(g,k)	(e,j),(f,k),(c,g)			
i	(b,i),(c,i),(d,g)	(c,i),(f,i),(d,g)	X	X	(b,i),(c,i),(d,g)	(k,d)	(i,j),(i,k),(d,h)	(i,e),(f,i),(c,i),(d,g)		
j	(f,i)	(c,f)	X	X	(c,f),(c,i)	(b,f),(c,f),(c,i),(g,k)	(b,j),(f,k),(c,g),(g,h)	(b,e)	(b,i),(f,i),(c,i),(d,g)	
k	X	X	(a,h),(c,d),(d,e),(d,f)	(a,h),(d,g)	X	X	X	X	X	X

The implication chart again with conditions that are not true marked (I've not marked all the things that are not true — just enough to find that two states are not equivalent).

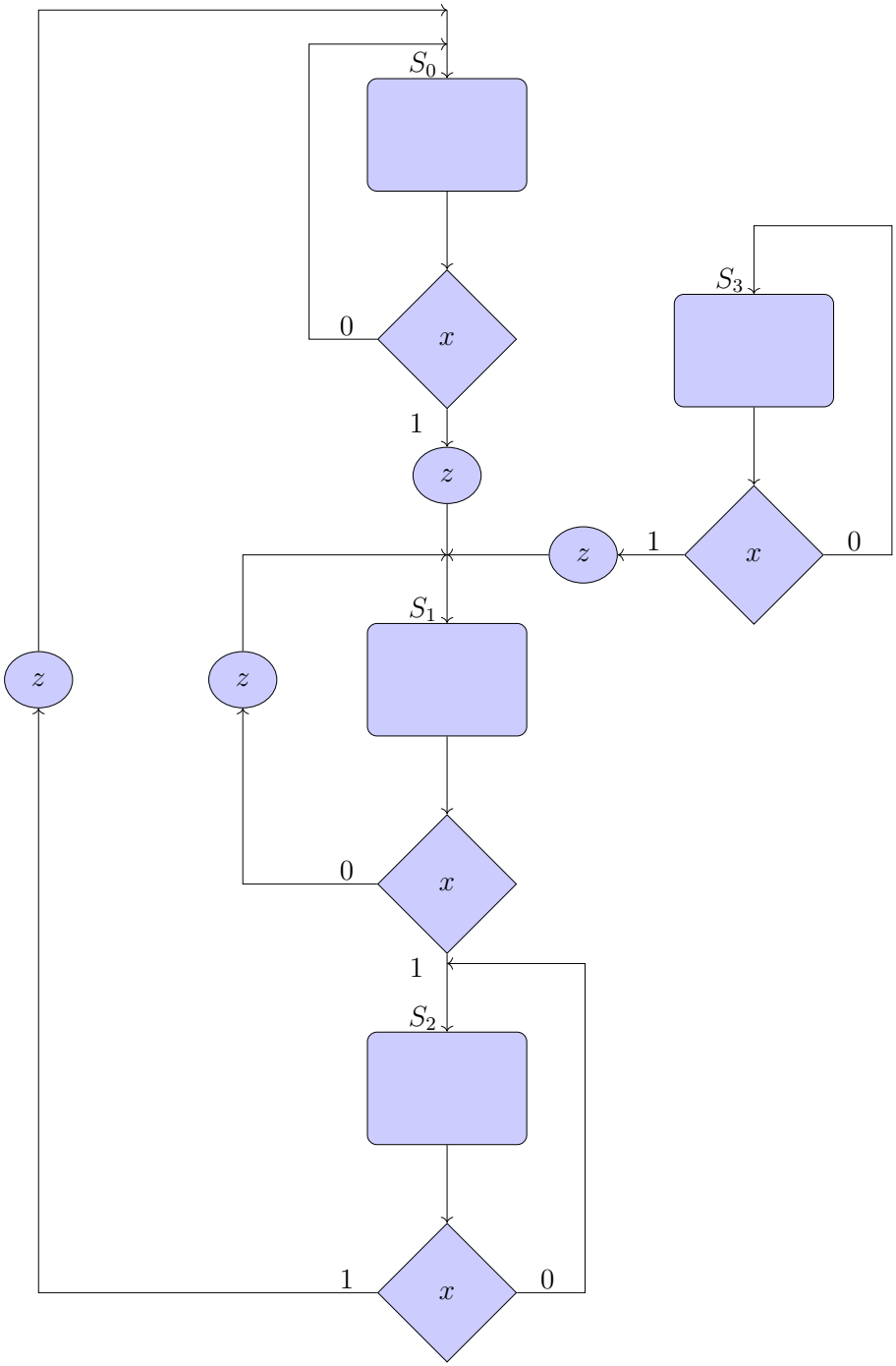
b	(c,i) X,(c,f)									
c	X	X								
d	X	X	(d,e) X,(f,g)							
e	(c,i) X	(f,i) X	X	X						
f	(b,f),(c,i) X, (g,k)	(c,i) X,(f,i), (g,k)	X	X	(b,f),(c,i) X, (g,k)					
g	(b,j) X,(k,i), (c,g),(g,h)	(b,j) X,(c,k), (f,g),(h,g)	X	X	(b,j) X,(c,k) (g,i),(g,h)	(f,j),(i,k) X, (g,i),(h,k)				
h	(b,e) X,(f,i)	(b,e),(c,f) X	X	X	(b,e),(c,f) X, (c,i) X	(f,e),(f,i), (c,i) X,(g,k)	(e,j) X,(f,k), (c,g)			
i	(b,i),(c,i) X, (d,g) X	(c,i) X,(f,i), (d,g) X	X	X	(b,i),(c,i) X, (d,g) X	(k,d) X	(i,j),(i,k) X, (d,h) X	(i,e),(f,i), (c,i) X,(d,g) X		
j	(f,i) X	(c,f) X	X	X	(c,f),(c,i) X	(b,f) X,(c,f), (c,i) X,(g,k)	(b,j) X,(f,k) (c,g),(g,h)	(b,e) X	(b,i),(f,i), (c,i) X,(d,g)	
k	X	X	(a,h),(c,d), (d,e) X,(d,f)	(a,h), (d,g) X	X	X	X	X	X	X
	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*

At this point, there are no possible state equivalencies and there can be no state mimimization.

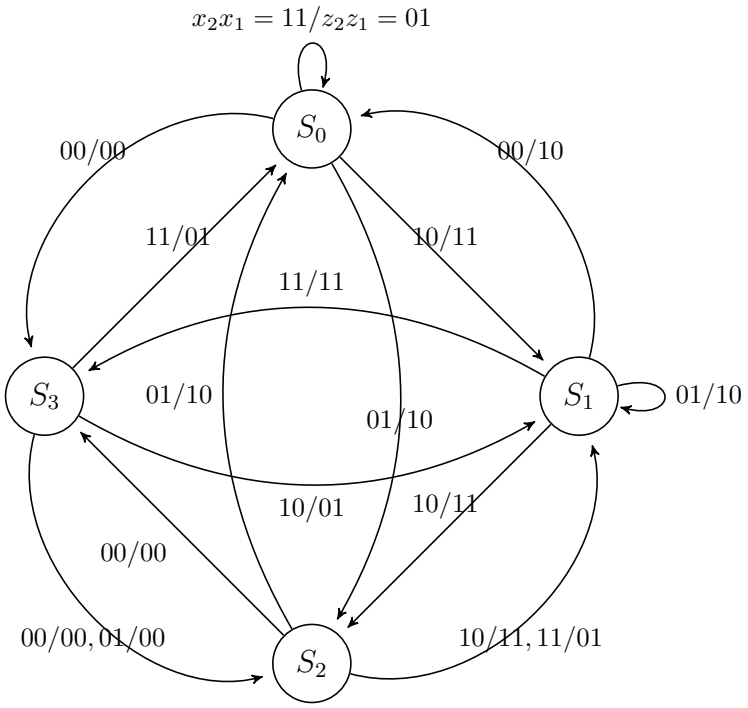
Q10: Shown below is a state diagram for a circuit with one input x and one output z. Convert the following state diagram into an ASM chart.



Solution:

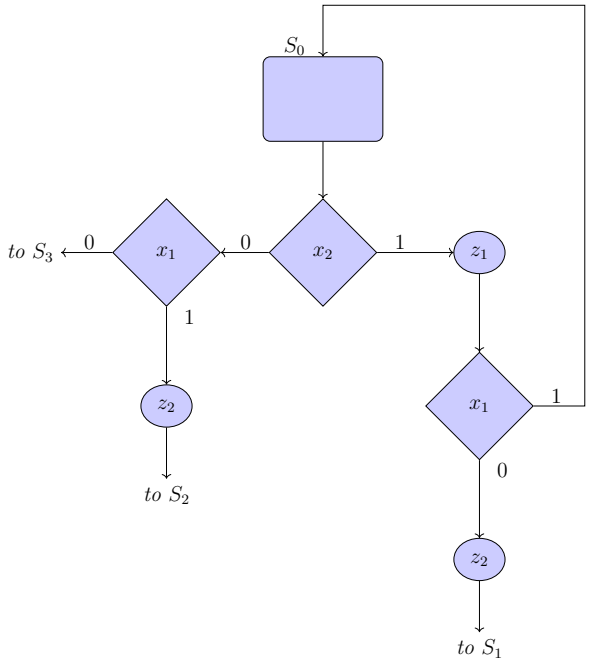


Q11: Shown below is a state diagram for a circuit with two inputs x_1 and x_2 and two outputs z_1 and z_2. Convert the state diagram into an ASM chart. Test only one variable in each decision box. Try to minimize the number of decision boxes.

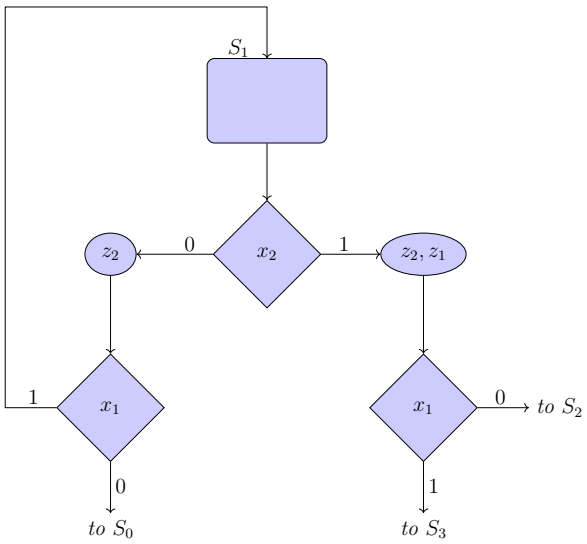


Solution:
I will draw the ASM chart in pieces to avoid having to draw a single large diagram.

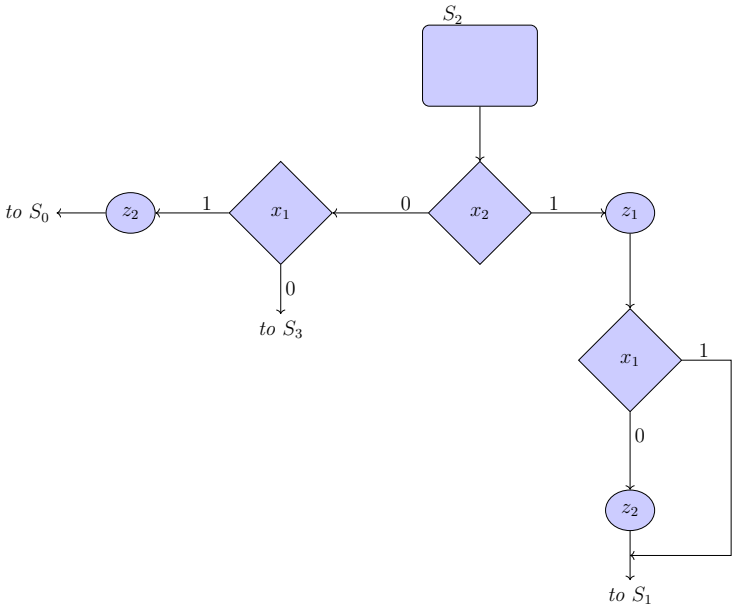
Showing S_0...



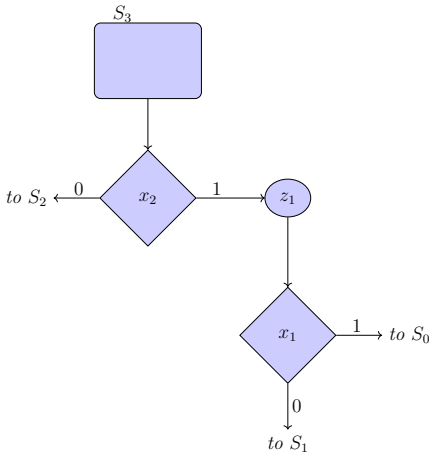
Showing S_1...



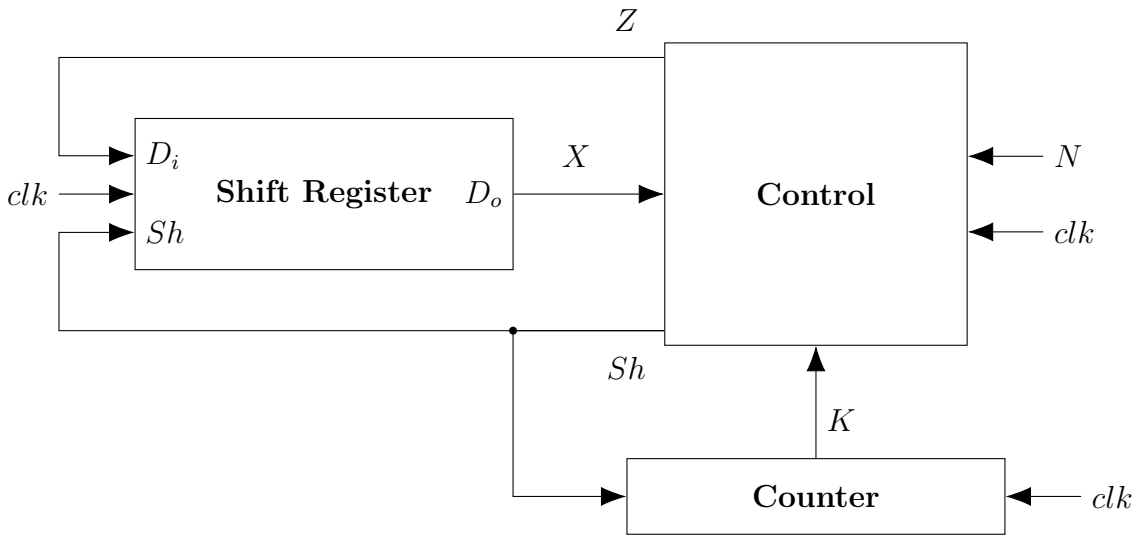
Showing S_2...



Showing S_3...



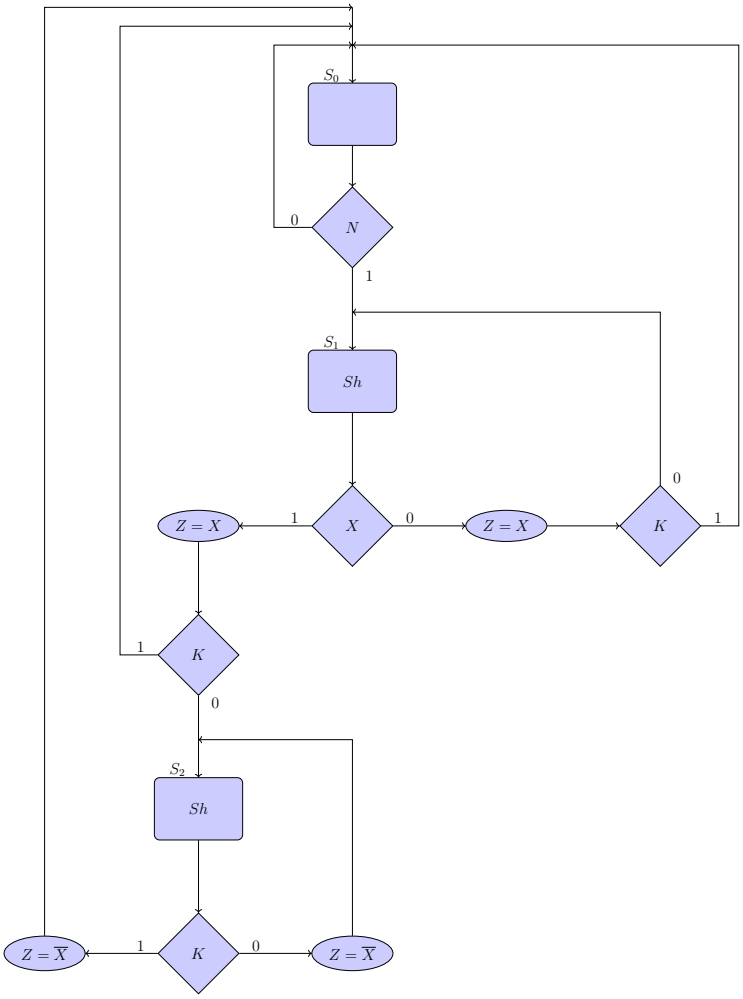
Q12: Shown below is a circuit which forms the 2's complement of a 16-bit binary number. The circuit consists of three main parts — a 16-bit shift register which initially holds the number to be complemented, a control circuit, and a counter circuit which counts the number of shifts. The control circuit processes the number in the shift register one bit at a time (via input X) and outputs the 2's complemented result (via output Z) which is stored back in the shift register one bit at a time. The control circuit also generates the necessary shift signals. The counter counts the number of shifts and outputs $K = 1$ once the counter reaches 15, otherwise $K = 0$. Finally, the control circuit has input N which is a start signal; when $N = 1$ the circuit performs its operation.



To perform the 2's complement using this circuit, the control circuit implements the rule "starting with the least significant bit, complement all the bits to the left of the first 1".

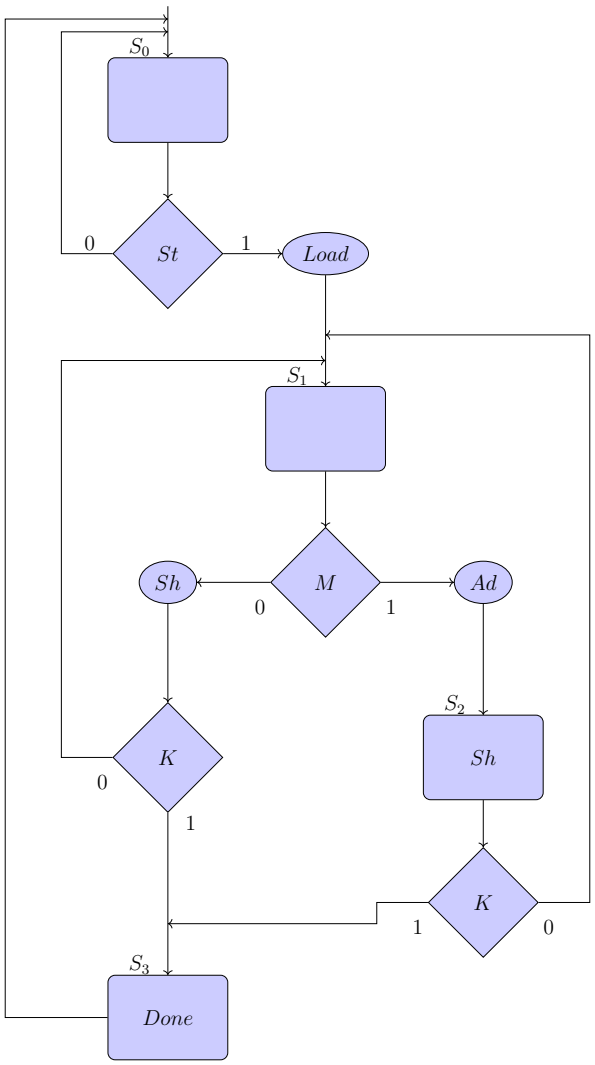
Draw an ASM chart for the control unit (only three states are required).

Solution:



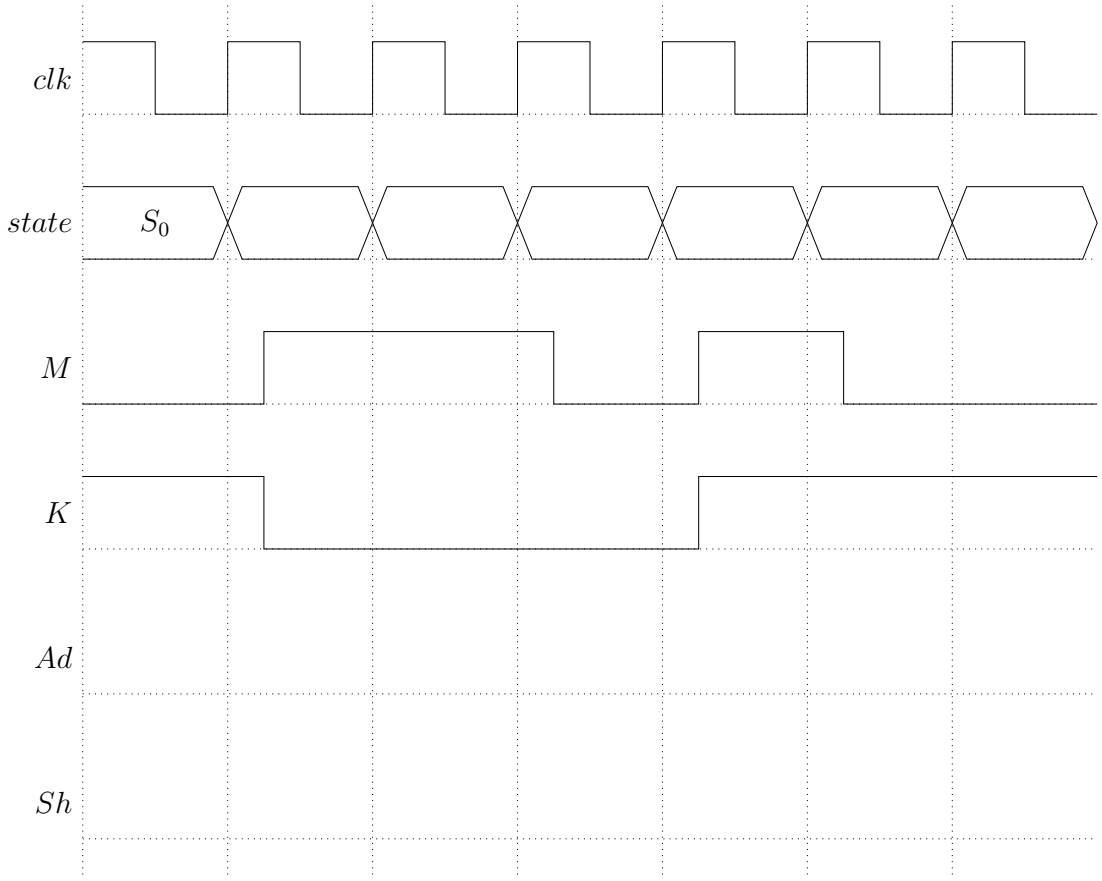
The ASM remains in state S_0 until told to start. It then enters state S_1 where it shifts and copies X to Z until the first 1 occurs. It then goes to S_2 where it shifts and flips X to Z until done.

Q13: Shown below is an ASM chart (4 states) for a circuit with three inputs (St, M and K) and four outputs (*Load*, *Ad*, *Sh* and *Done*).

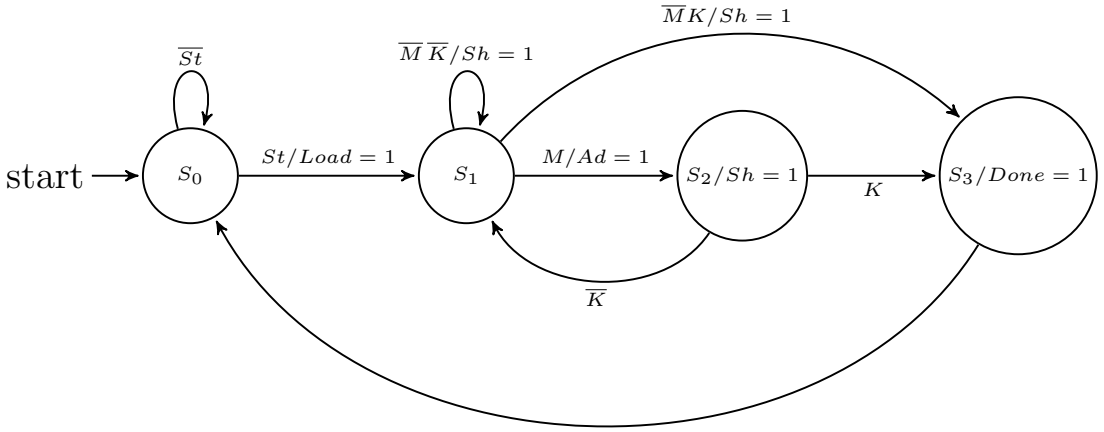


(a) Draw an equivalent state diagram for this ASM chart.

(b) Complete the following timing diagram showing states and the output signals Ad and Sh. You may assume $St = 1$.


clk
state
S_0
M
K
Ad
Sh

Solution:
The equivalent state diagram:



Note that this state diagram is "complete" in that if in any state, with what's labeled in the diagram, we always know what the next state will be (there are no missing self loops).

The timing diagram: