SEE 3243/4243

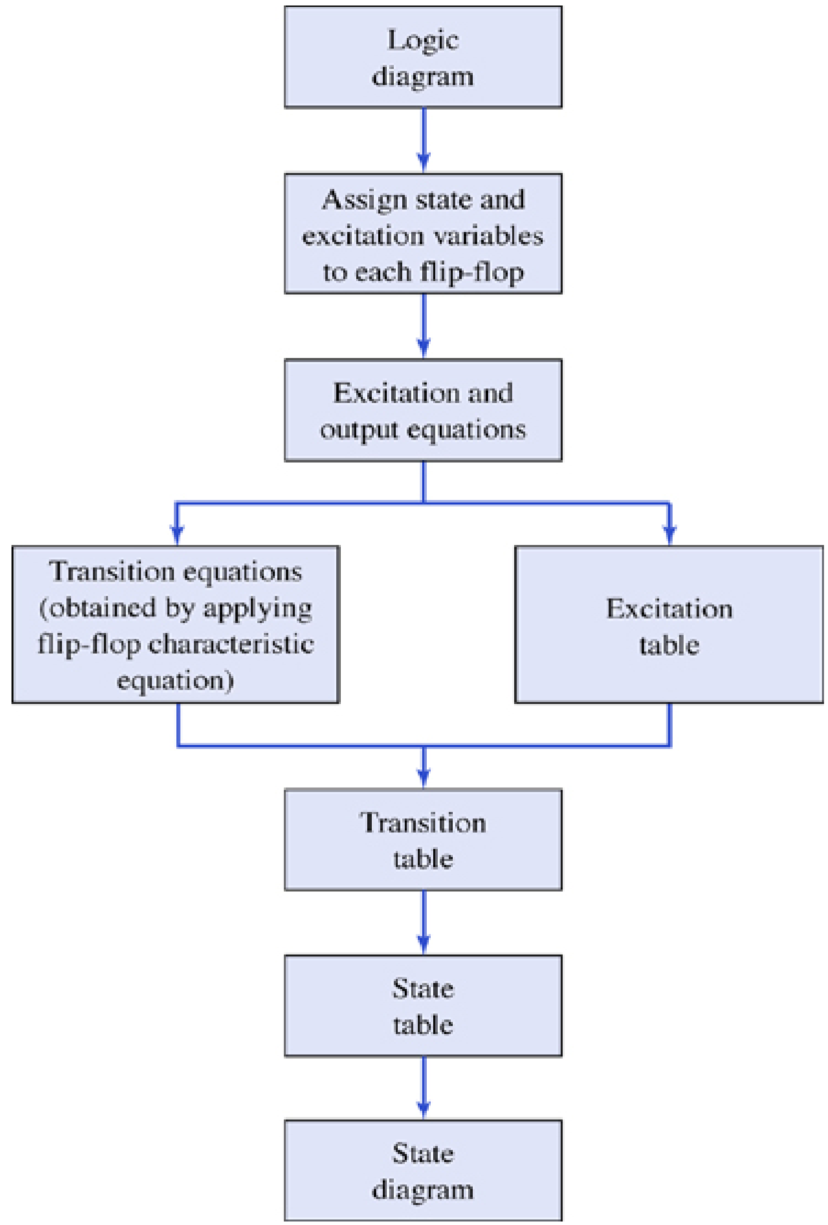
FSM Analysis

Week 11

- Getting state diagram from a circuit
  - Moore examples
  - Mealy examples

Analysis procedure

- Two techniques for reverse engineering:
  - **Ad Hoc**: Try input combinations to derive transition table
    - Try avoiding ad hoc method
  - **Formal**: Derive transition by analyzing the circuit



State-machine analysis steps

- Assumption: Starting point is a logic diagram.
  1. Determine next-state function F and output function G.

  2a. Construct state table

  For each state/input combination, determine the excitation value.

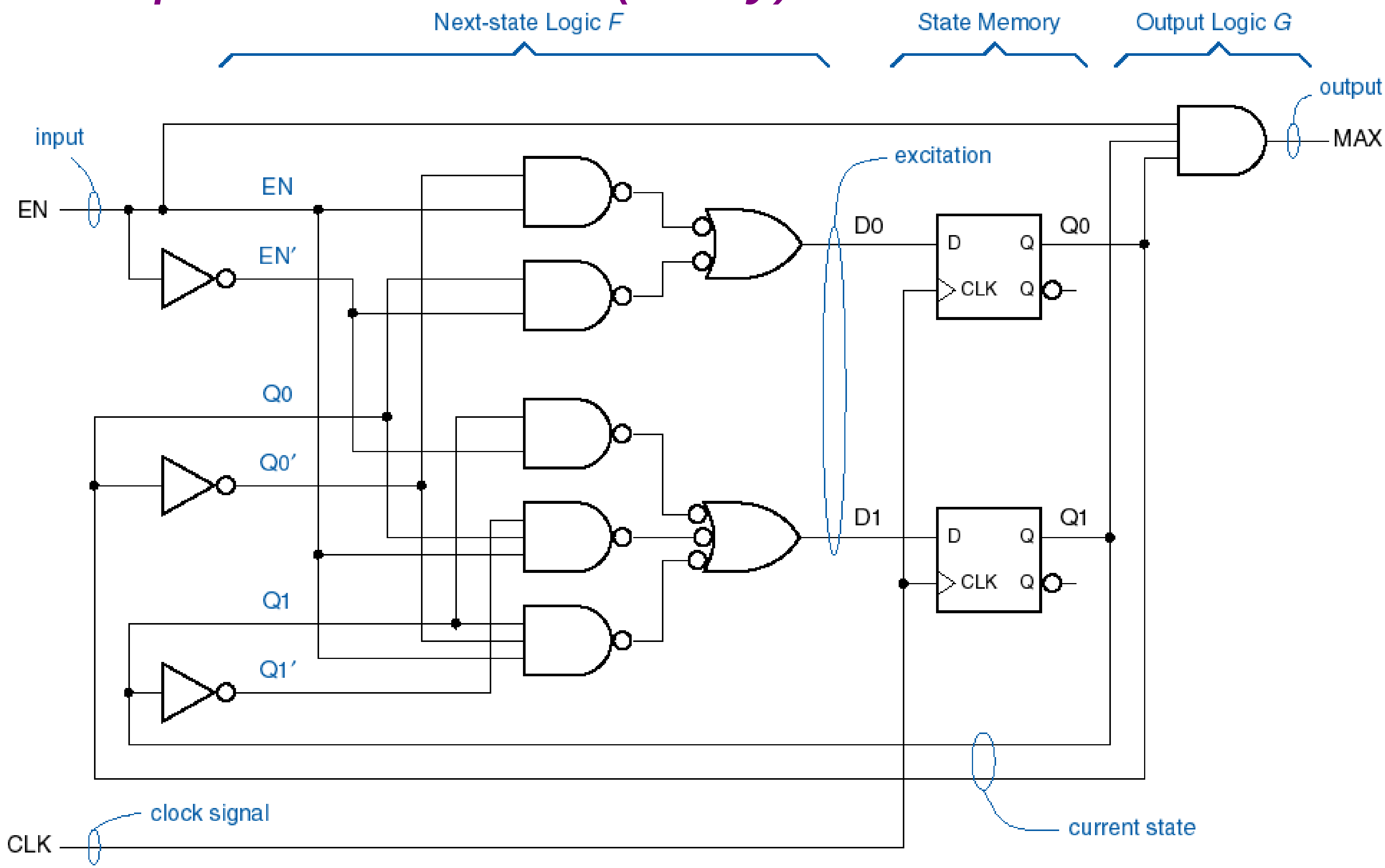
  Using the characteristic equation, determine the corresponding next-state values (trivial with DFFs).

  2b. Construct output table

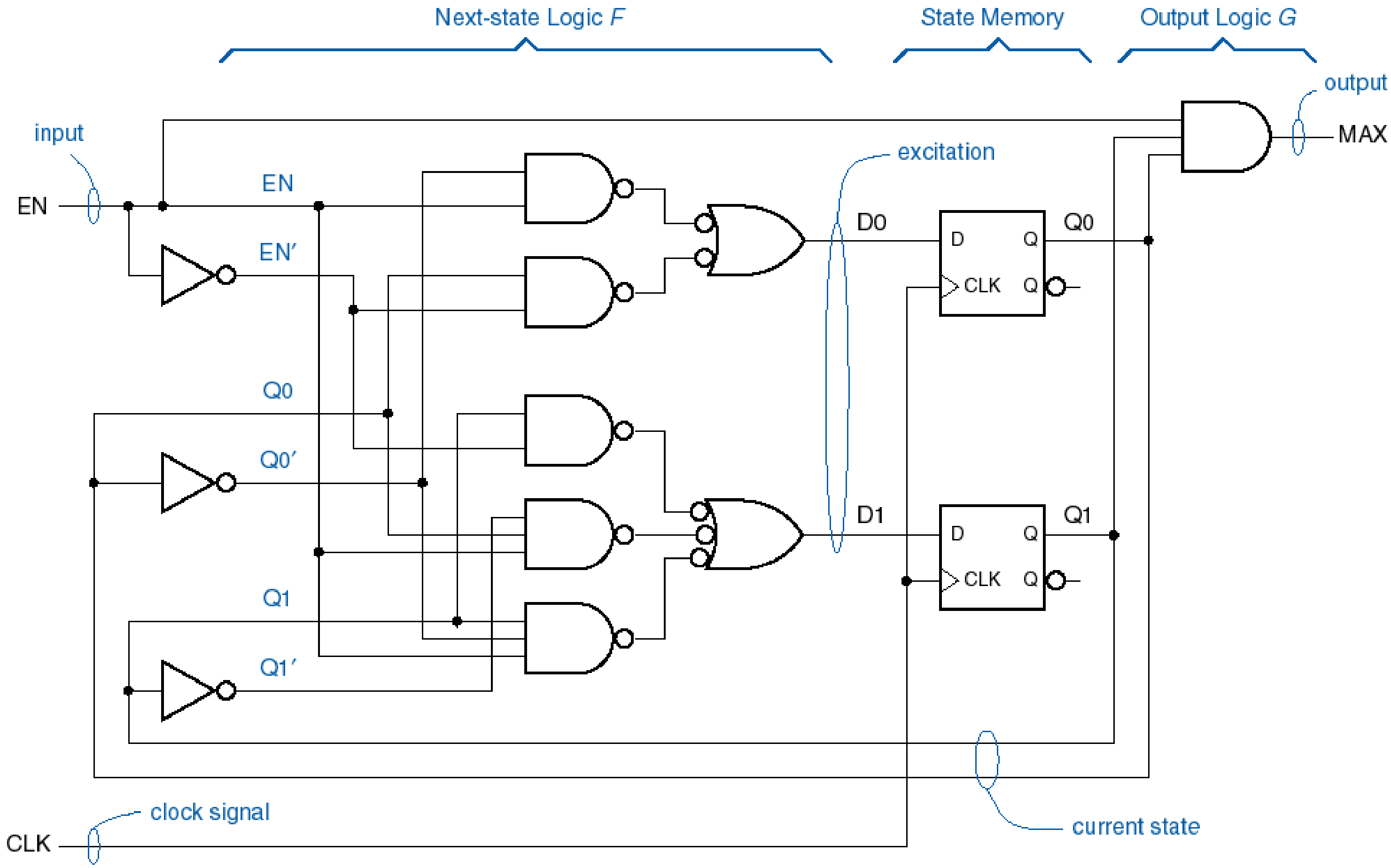
  For each state/input combination, determine the output value. (Can be combined with state table.)

  3. (Optional) Draw state diagram

Example state machine (Mealy)



Excitation equations



$$D0 = Q0 \cdot EN' + Q0' \cdot EN$$

$$D1 = Q1 \cdot EN' + Q1' \cdot Q0 \cdot EN + Q1 \cdot Q0' \cdot EN$$

Transition equations

- Excitation equations

$$D_0 = Q_0 \cdot EN' + Q_0' \cdot EN$$
$$D_1 = Q_1 \cdot EN' + Q_1' \cdot Q_0 \cdot EN + Q_1 \cdot Q_0' \cdot EN$$

- Characteristic equations (Trivial for DFF!)

$$Q_0^+ = D_0$$
$$Q_1^+ = D_1$$

- Substitute excitation equations into characteristic equations

$$Q_0^+ = Q_0 \cdot EN' + Q_0' \cdot EN$$
$$Q_1^+ = Q_1 \cdot EN' + Q_1' \cdot Q_0 \cdot EN + Q_1 \cdot Q_0' \cdot EN$$

Transition and state tables

$$Q0* = Q0 \cdot EN' + Q0' \cdot EN$$
$$Q1* = Q1 \cdot EN' + Q1' \cdot Q0 \cdot EN + Q1 \cdot Q0' \cdot EN$$

(transition equations)

$$MAX = Q1 \cdot Q0 \cdot EN$$ (output equation)

	EN	
Q1 Q0	***0***	***1***
00	00	01
01	01	10
10	10	11
11	11	00
	Q1* Q0*	

transition table

	EN	
S	***0***	***1***
A	A	B
B	B	C
C	C	D
D	D	A
	S*	

state table

	EN	
S	***0***	***1***
A	A, 0	B, 0
B	B, 0	C, 0
C	C, 0	D, 0
D	D, 0	A, 1
	S*, MAX	

state/output table

Transition and state tables (using K-map)

$$Q0* = Q0 \cdot EN' + Q0' \cdot EN$$
$$Q1* = Q1 \cdot EN' + Q1' \cdot Q0 \cdot EN + Q1 \cdot Q0' \cdot EN$$

(transition equations)

$$MAX = Q1 \cdot Q0 \cdot EN$$ (output equation)

Q1*

EN \ Q1Q0	00	01	11	10
0	0	0	1	1
1	0	1	0	1

Q0*

EN \ Q1Q0	00	01	11	10
0	0	1	1	0
1	1	0	0	1

MAX

EN \ Q1Q0	00	01	11	10
0	0	0	0	0
1	0	0	1	0

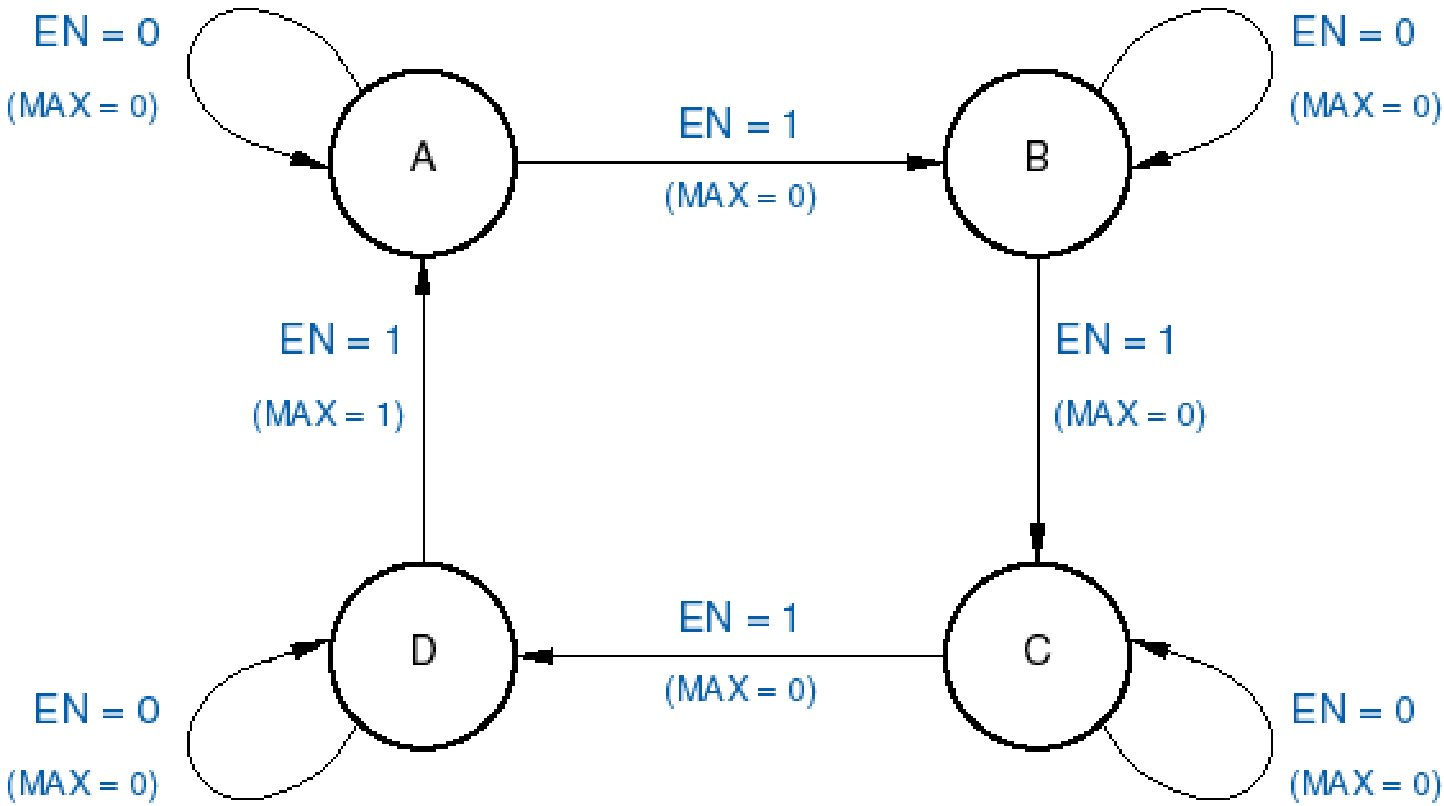
	EN	
Q1 Q0	*0*	*1*
00	00	01
01	01	10
10	10	11
11	11	00
	Q1* Q0*	

transition table

	EN	
S	*0*	*1*
A	A, 0	B, 0
B	B, 0	C, 0
C	C, 0	D, 0
D	D, 0	A, 1
	S*, MAX	

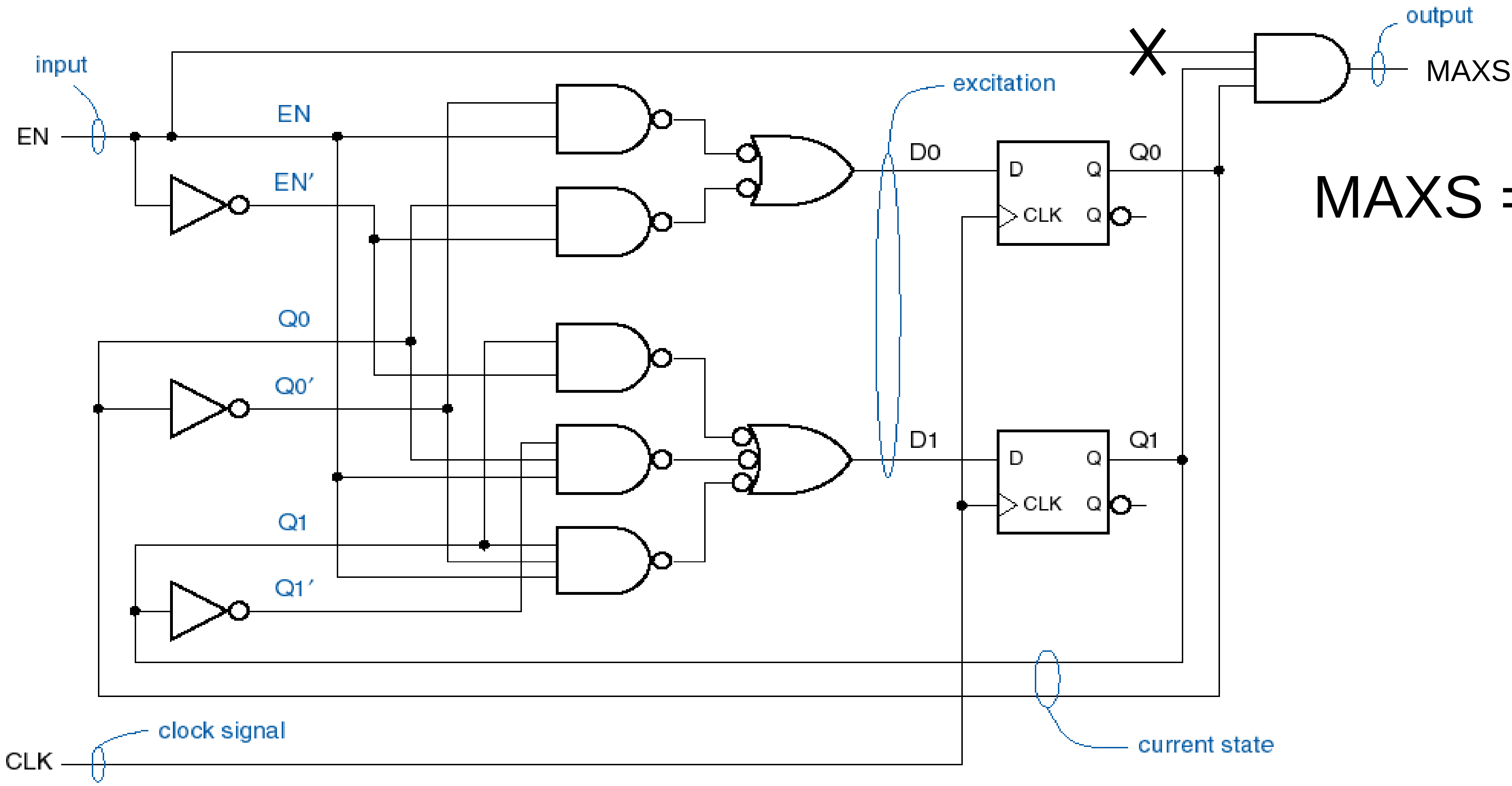
state/output table

State diagram



- Circles for states
- Arrows for transitions (note output info)

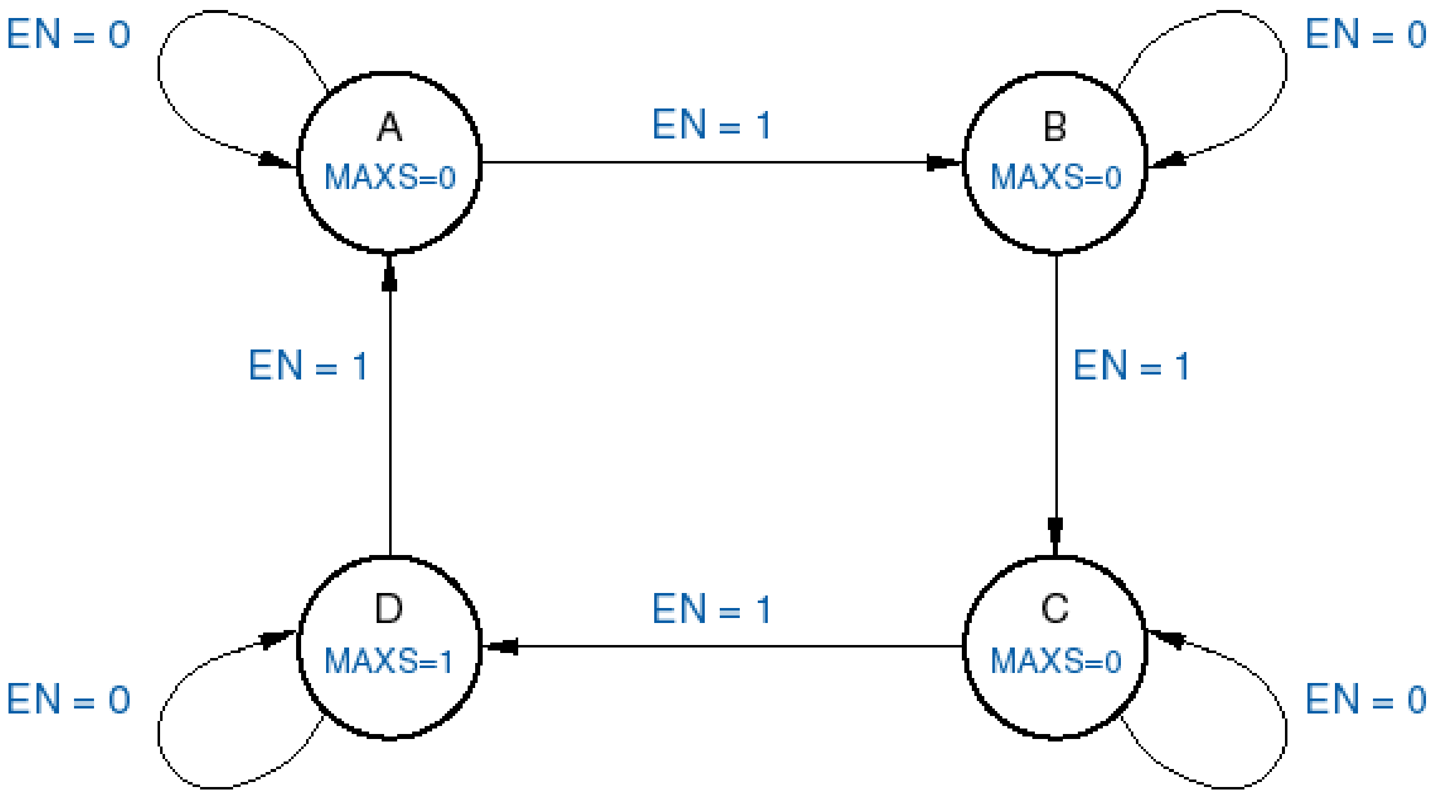
Modified state machine (Moore machine)



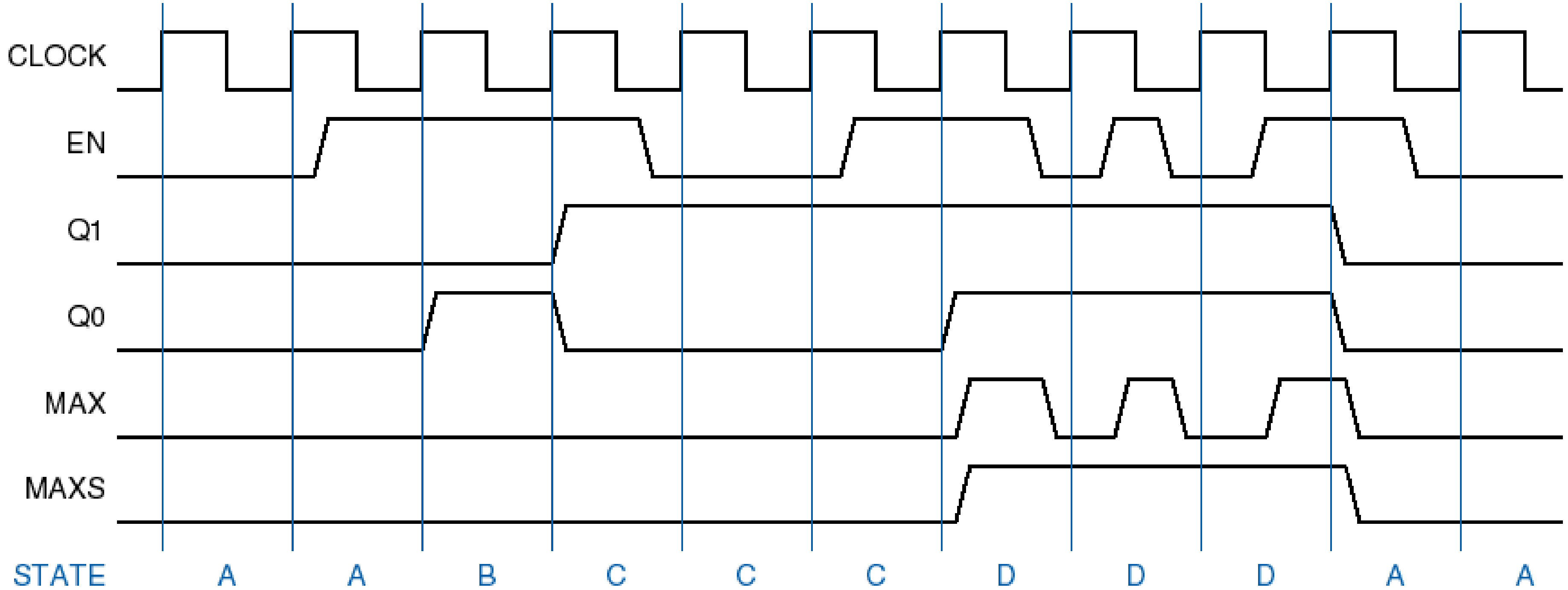
MAXS = Q1·Q0

Updated state/output table, state diagram

	EN		
S	0	1	MAXS
A	A	B	0
B	B	C	0
C	C	D	0
D	D	A	1
	S*		



Timing diagram for state machine

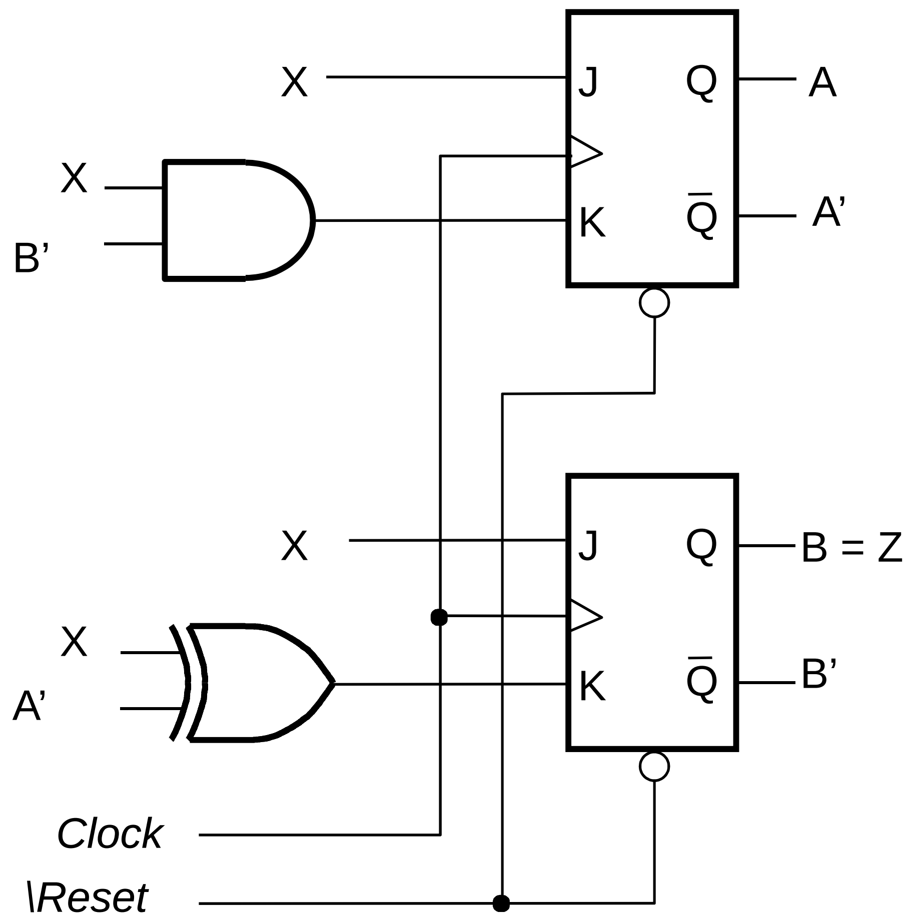


- ***Not*** a complete description of machine behavior. State diagram gives better description than timing diagram.

Another example: analysis of Moore machine

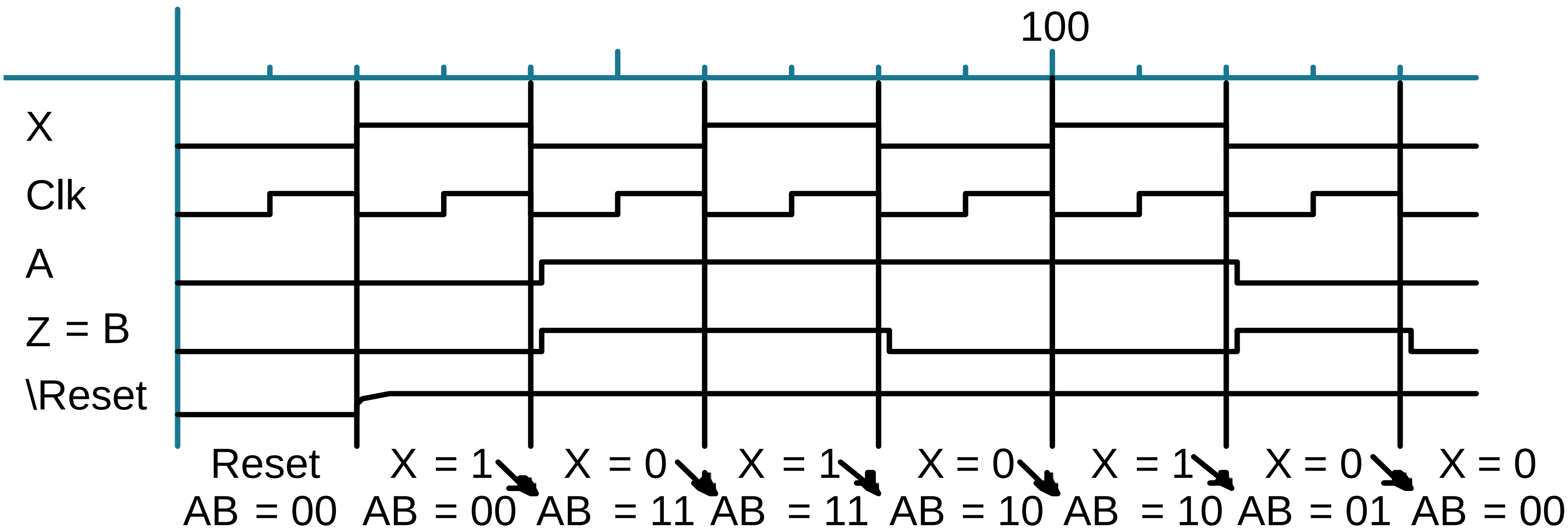
- Reverse engineer the following

Input X
Output Z
State A, B
(Z = B)



Ad Hoc Reverse Engineering

Behavior in response to input sequence 1 0 1 0 1 0:



Partially Derived State Transition Table

A	B	X	A+	B+	Z
0	0	0	?	?	0
		1	1	1	0
0	1	0	0	0	1
		1	?	?	1
1	0	0	1	0	0
		1	0	1	0
1	1	0	1	1	1
		1	1	0	1

Ad Hoc Reverse Engineering

Partially Derived State Transition Table

A	B	X	A+	B+	Z
0	0	0	?	?	0
		1	1	1	0
0	1	0	0	0	1
		1	?	?	1
1	0	0	1	0	0
		1	0	1	0
1	1	0	1	1	1
		1	1	0	1

We can complete the transition table by evaluating the circuit for the unknown conditions.
For eg. when A = B = X = 0,

$J_a = K_a = 0$, hence A+ = 0

$J_b = 0$, $K_b = 1$, hence B+ = 0

Formal Reverse Engineering

- Derive transition table from next state and output combinational functions presented to the flipflops.

$J_a = X$ $K_a = XB'$ $Z = B$

$J_b = X$ $K_b = X \oplus A'$

- Characteristic equation for J-K FF

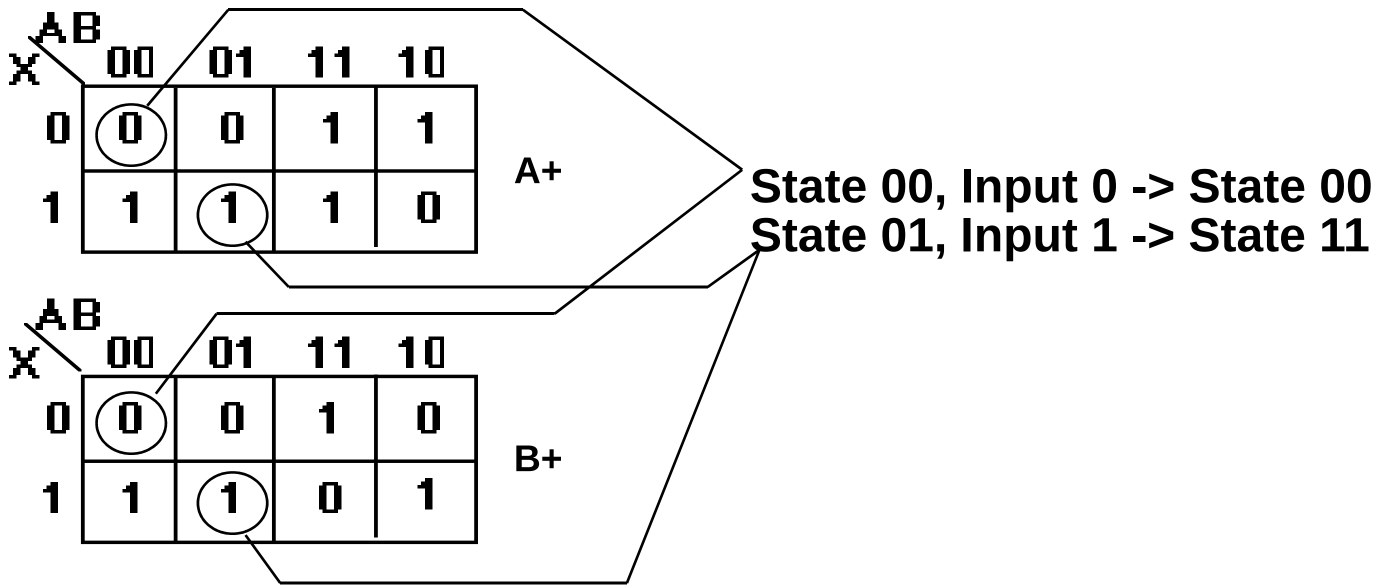
$$Q^+ = JQ' + K'Q$$

- FF excitation equations for J-K FF:

$$A^+ = J_aA' + K_a'A$$
$$= XA' + (XB')'A = XA' + (X' + B)A = A'X + AX' + AB$$
$$B^+ = J_bB' + K_b'B$$
$$= XB' + (X \oplus A')'B$$
$$= XB' + (X \text{ xnor } A')B = XB' + (AX' + A'X)B = B'X + ABX' + A'BX$$

Formal Reverse Engineering

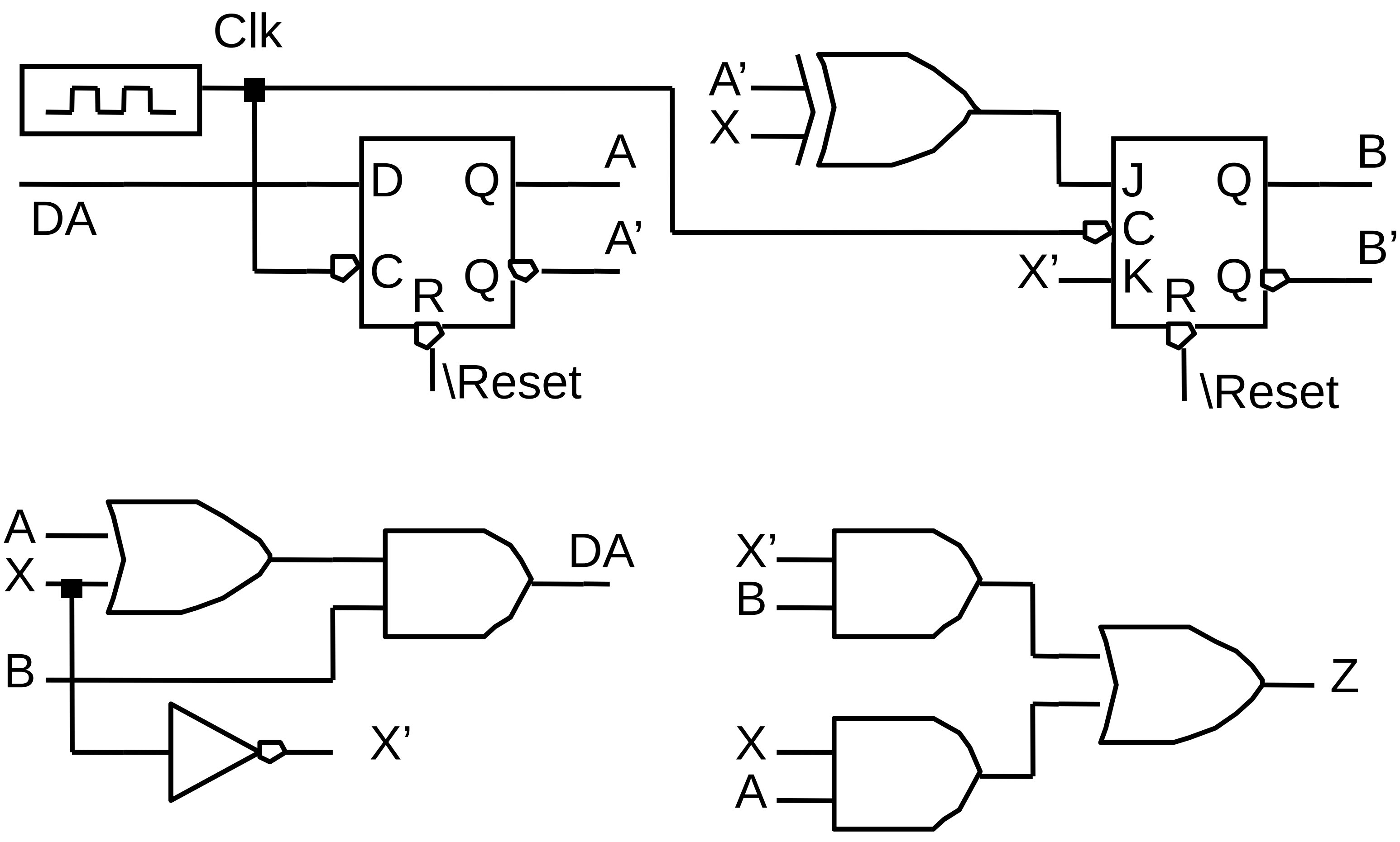
- Next state K-maps:



- Exercise: draw the state diagram.

Another example: analysis of Mealy machine

- Reverse engineer the following

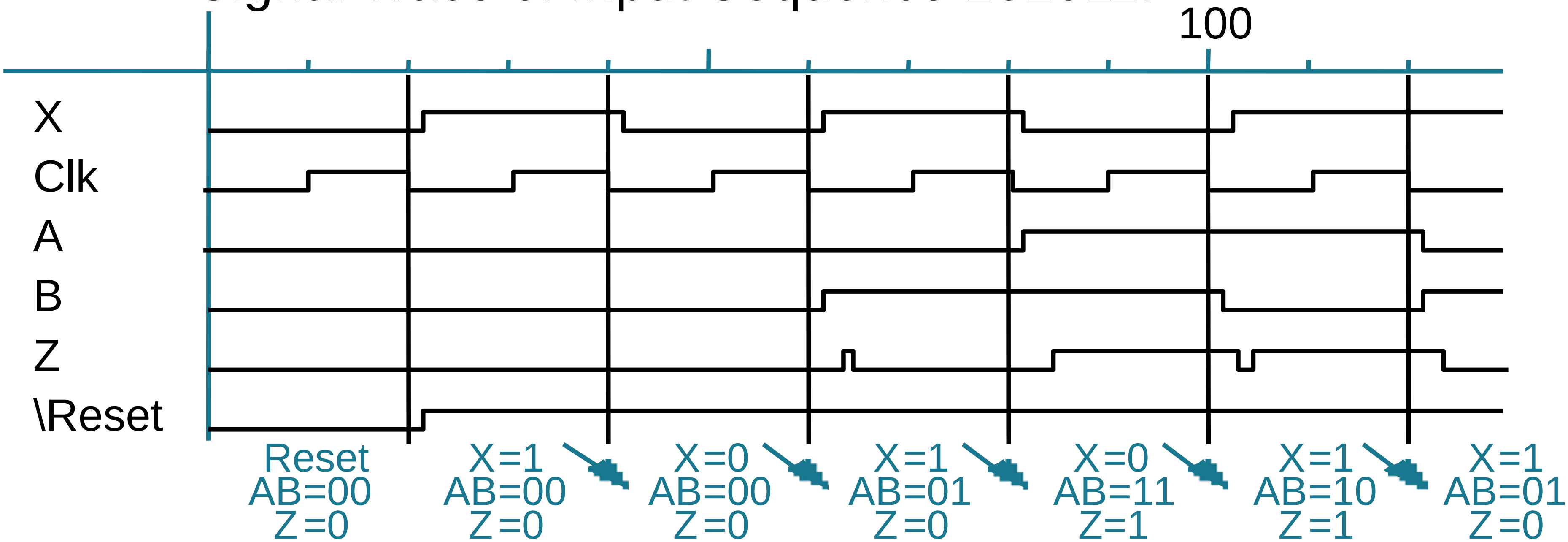


Input X, Output Z, State A, B

State register consists of D FF and J-K FF

Ad Hoc Reverse Engineering

Signal Trace of Input Sequence 101011:



Partially completed state transition table based on the signal trace

A	B	X	A+	B+	Z
0	0	0	0	1	0
		1	0	0	0
0	1	0	?	?	?
		1	1	1	0
1	0	0	?	?	?
		1	0	1	1
1	1	0	1	0	1
		1	?	?	?

Formal Reverse Engineering

- Derive transition table from next state and output combinational functions presented to the flipflops.

$DA = (A + X)B$
$J = AX + A'X'$ $\quad K = X'$ $\quad Z = BX' + AX$

- Characteristic equation for FF

$Q^+ = D$ (D FF)

$Q^+ = JQ' + K'Q$ (J-K FF)

- FF excitation equations:

$A^+ = DA$

$= (A + X)B = AB + BX$

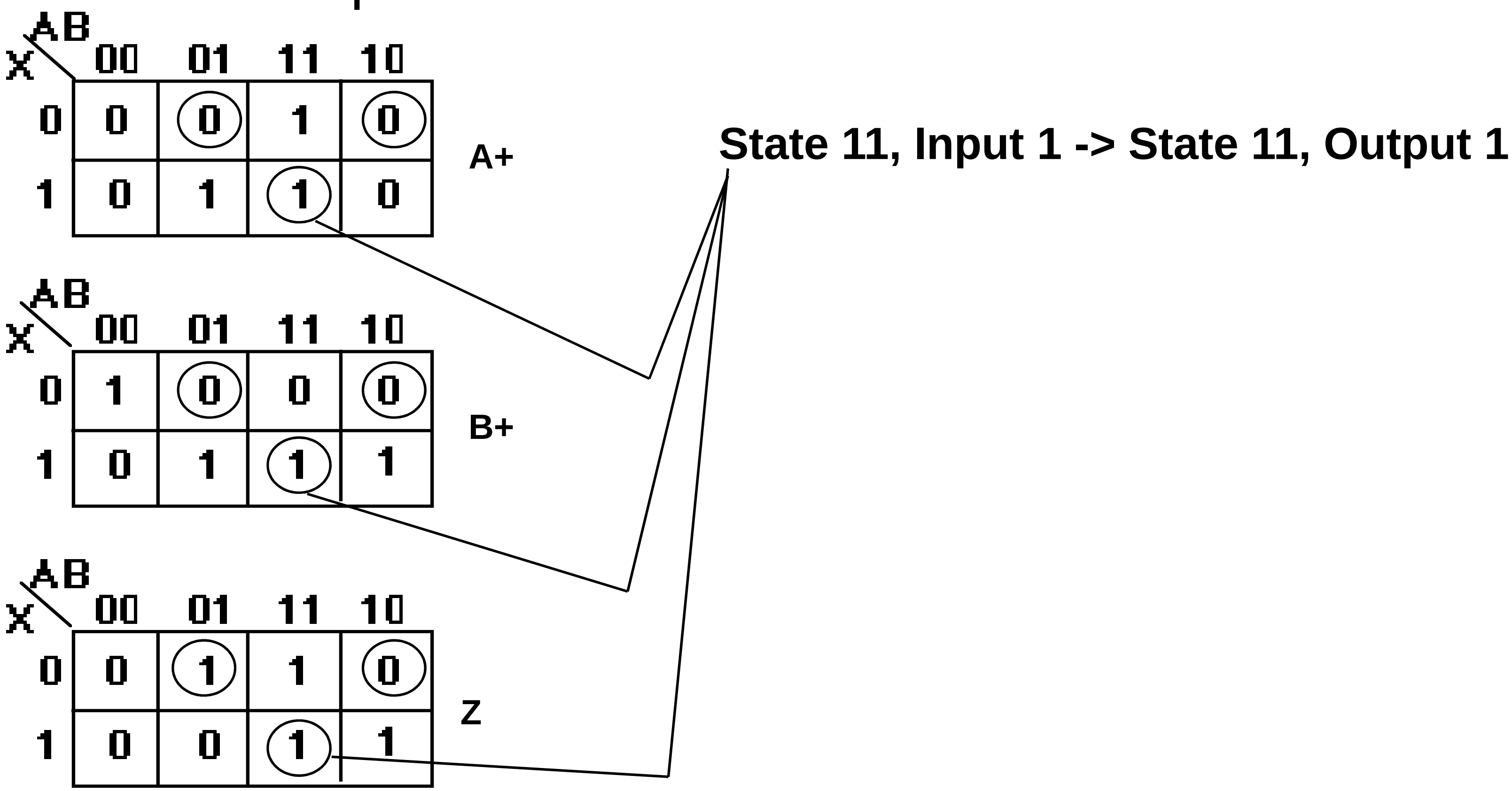
$B^+ = JB' + K'B$

$= (AX + A'X')B' + (X')'B$

$= AB'X + A'B'X' + BX$

Formal Reverse Engineering

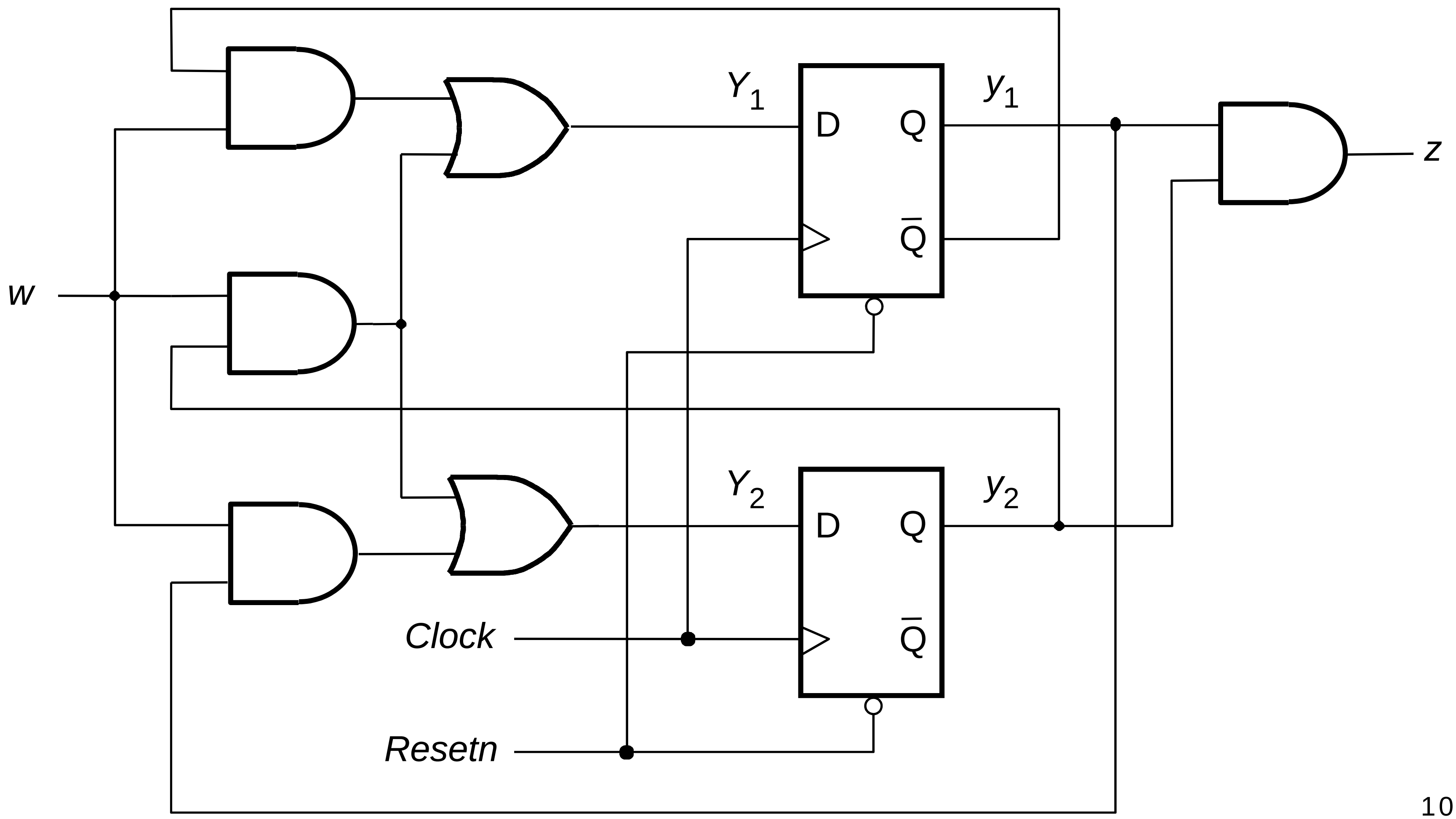
- Next state K-maps:



- Exercise: draw the state diagram.

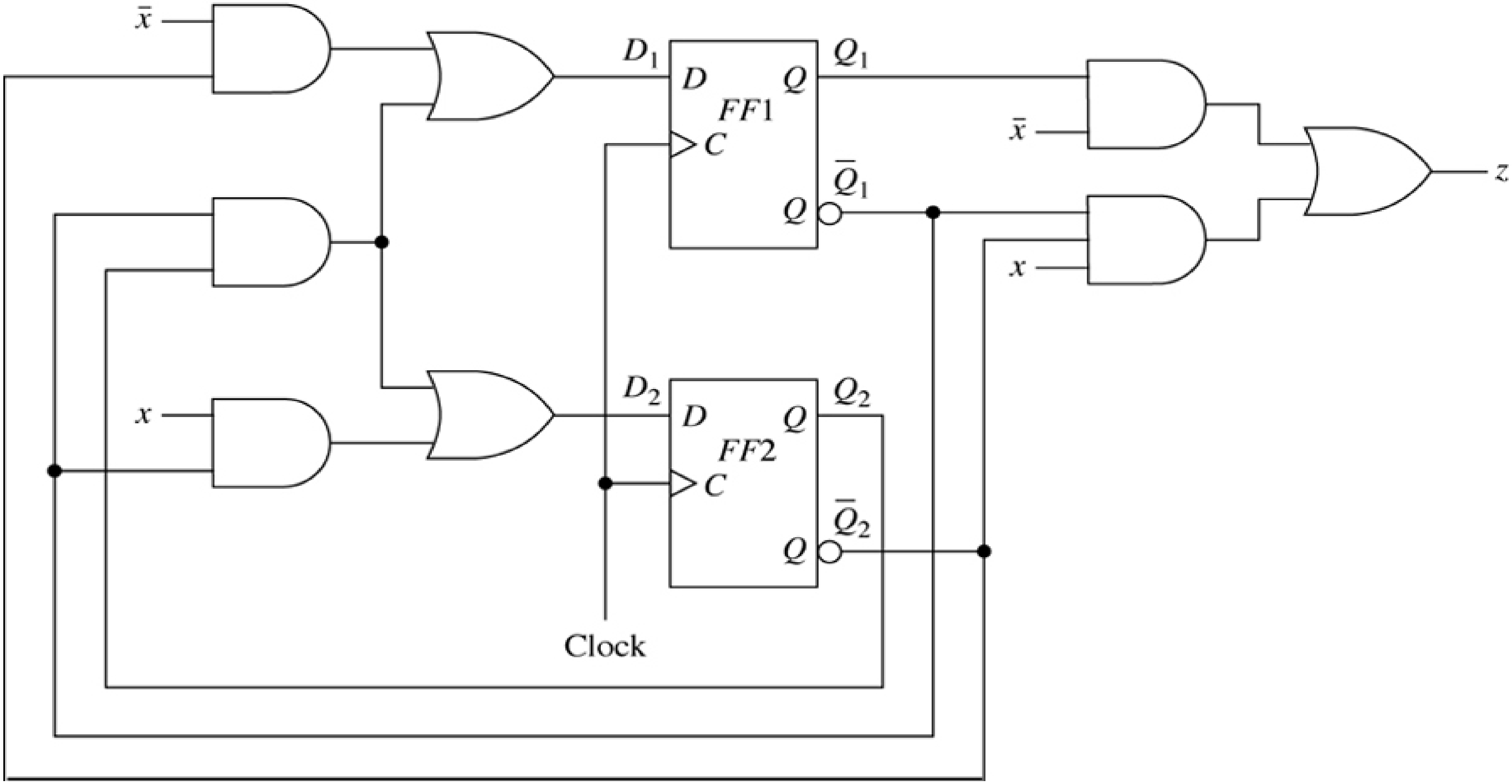
Exercise 1

- Draw the state diagram for FSM below. The machine has one input w and one output z.



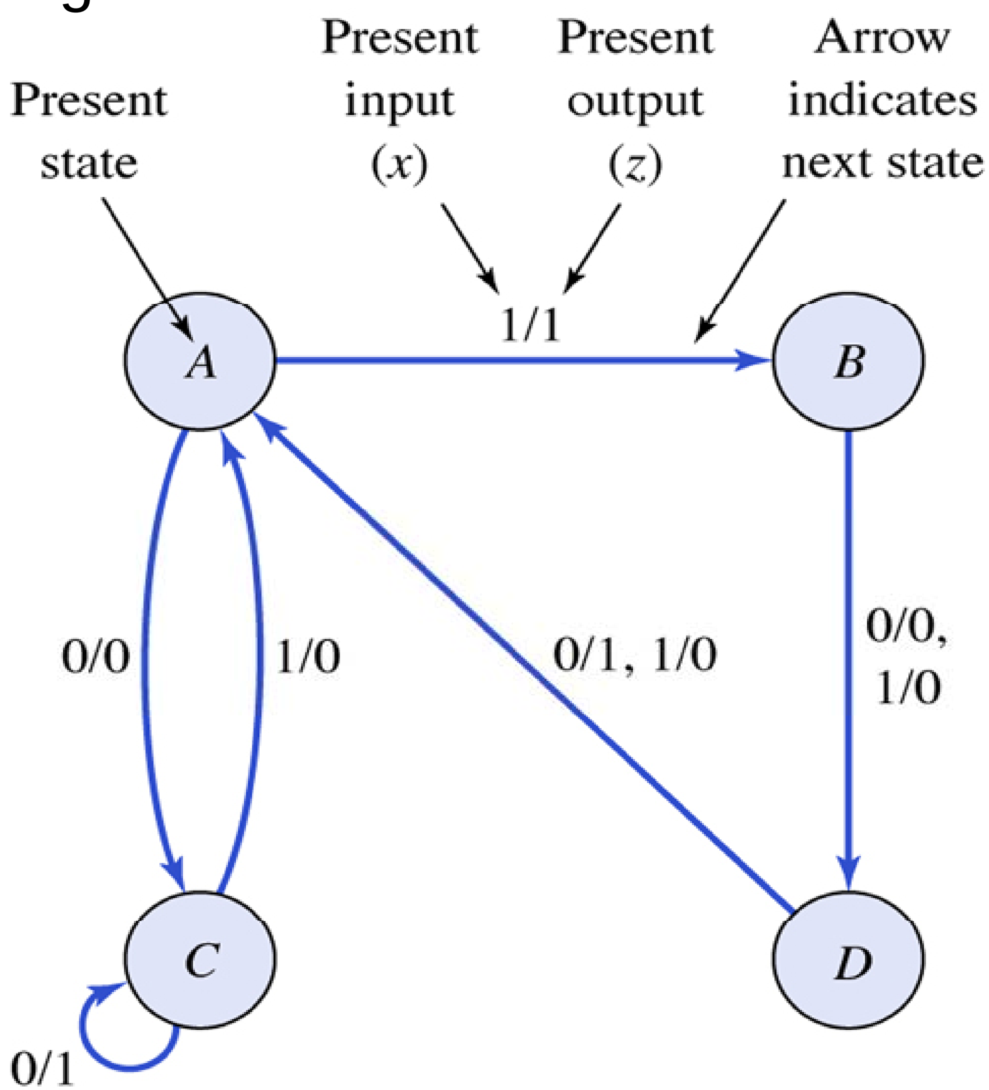
Exercise 2

- Derive the state diagram for FSM below. The machine has one input x and one output z.

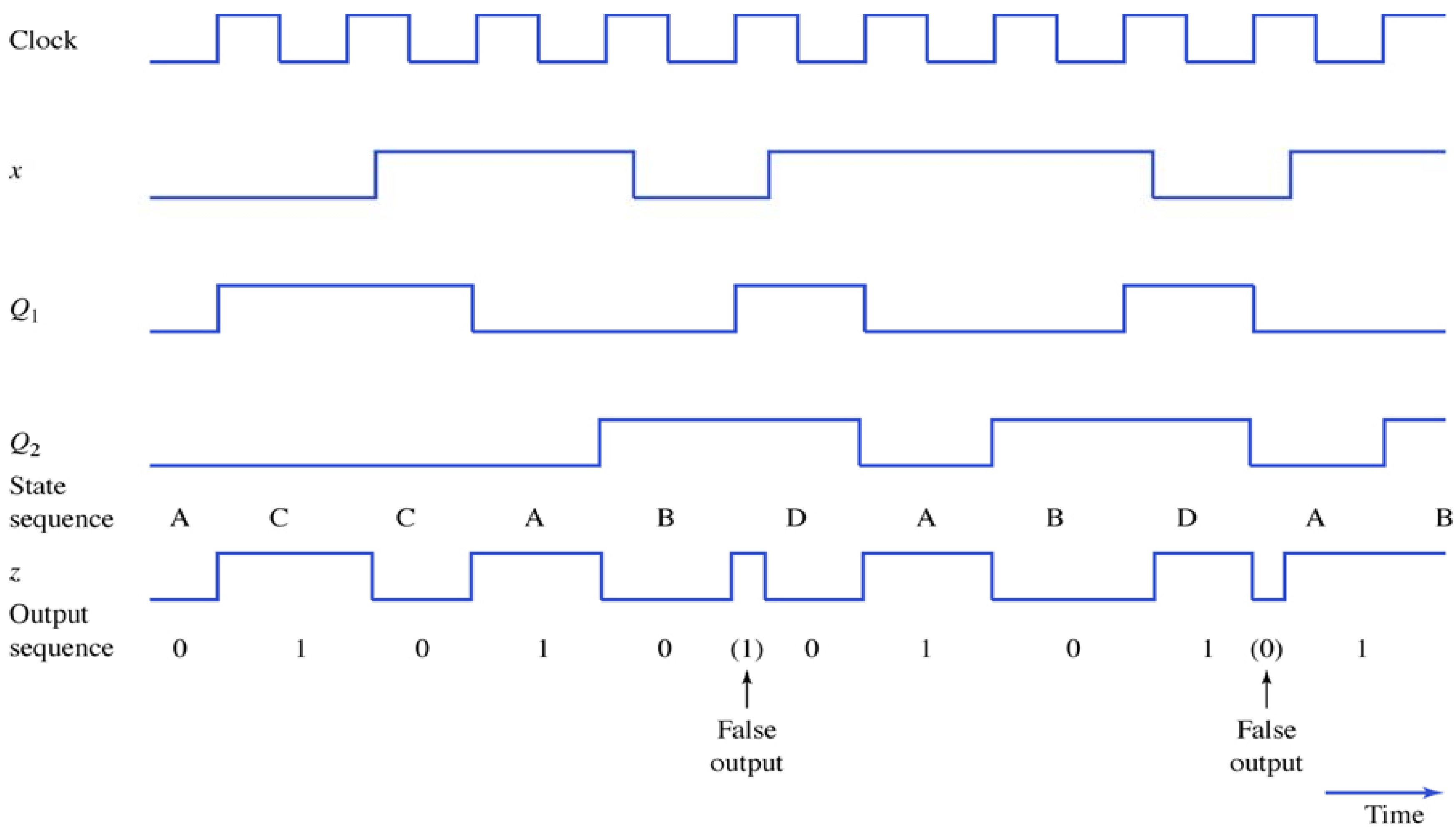


Solution to Exercise 2

- The state diagram.

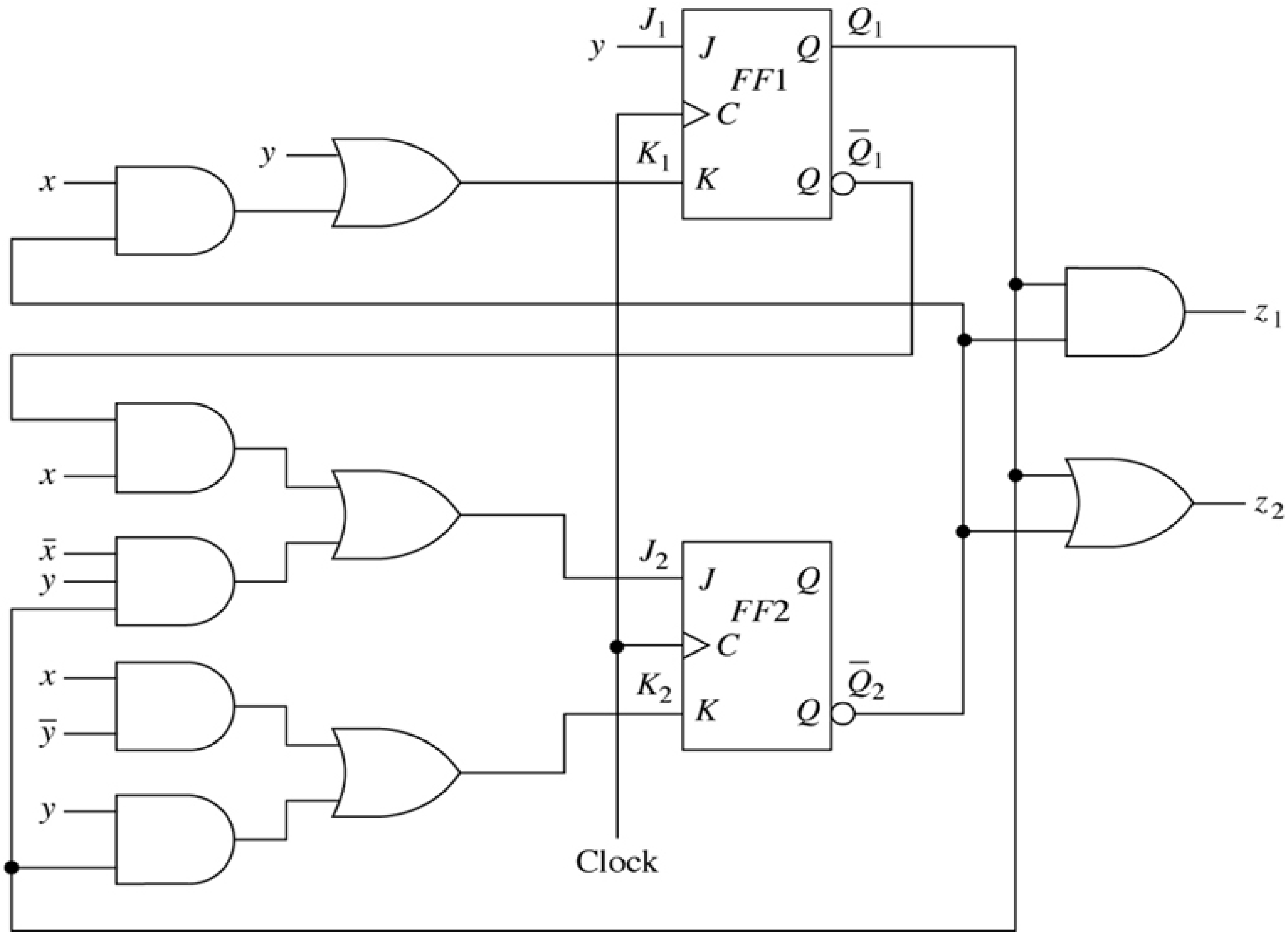


Timing diagram for Exercise 2 FSM



Exercise 3

- Draw the state diagram for FSM below. The machine has two inputs x and y and two outputs z_1 and z_2.



Solution to Exercise 3