Sequential Circuit Analysis

Synchronous vs. Asynch.

- **Synchronous sequential circuit:**
  - Its behavior can be determined knowing its input signals at discrete instants of time.
  - Achieves synchronization among components by using a timing signal called the *clock:*
    - Its outputs change synchronized with the clock
- **Asynchronous sequential circuit:**
  - Its behavior depends on the order of input signals changes over a continuous time
  - There is no need for synchronization:
    - No clock signal
    - Its outputs can change at **any** time

Clock Signal

Rising Clock Edge

Clock generator: Periodic train of clock pulses

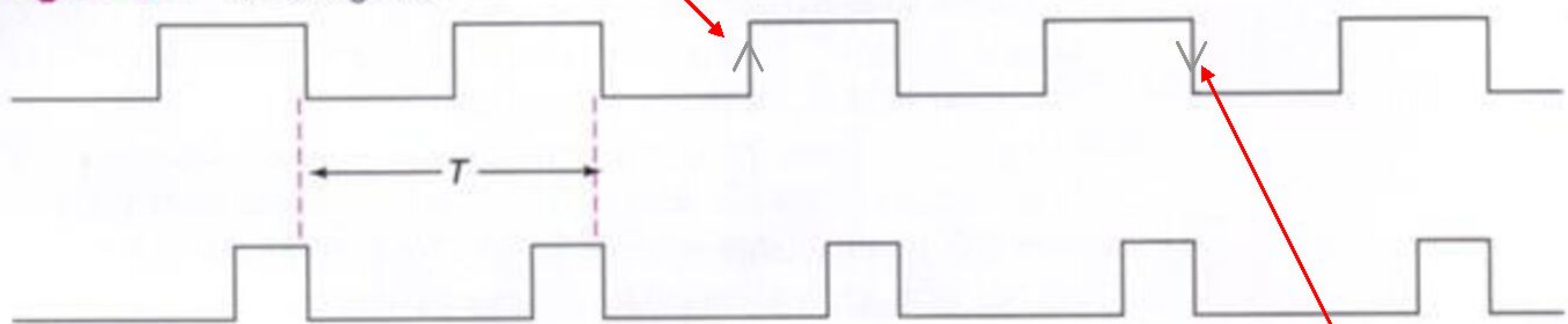


Figure 5.1 Clock signals.

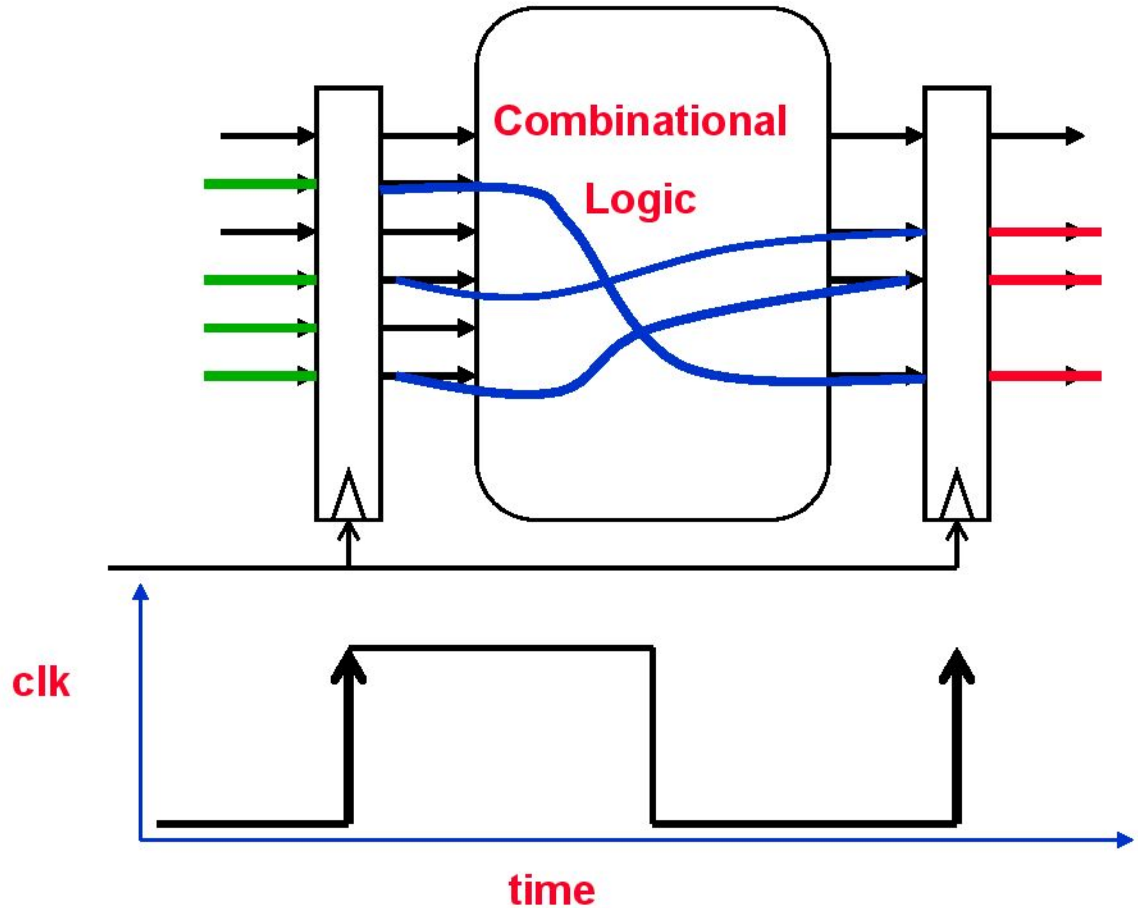
Different duty cycles

Falling Clock Edge

Clock Signal

- Clock is distributed throughout the entire design
- Each component synchronizes itself with it

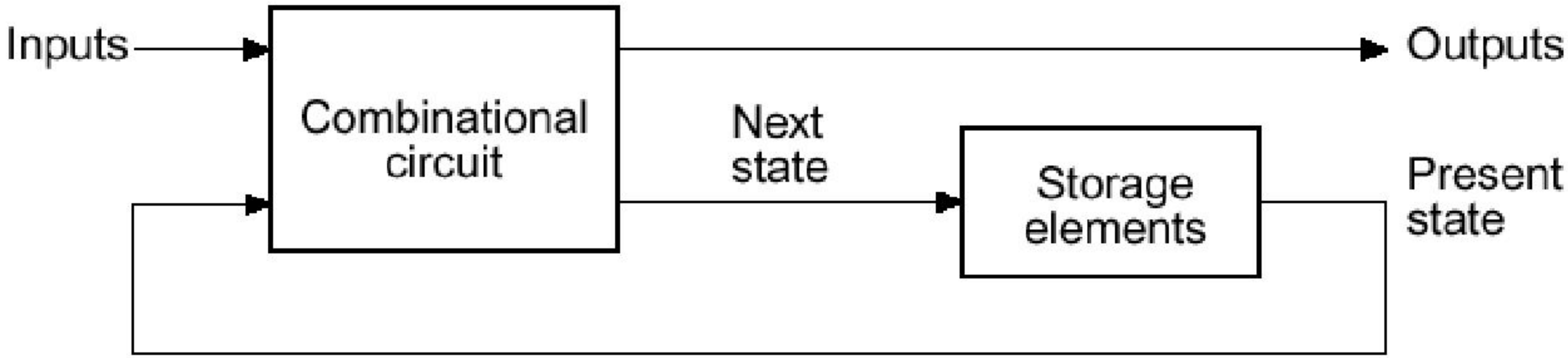
Synchronous Circuits



Sequential Circuit Analysis

- **Analysis:**
  - Obtaining a suitable description that demonstrates the time sequence of inputs, outputs, and states

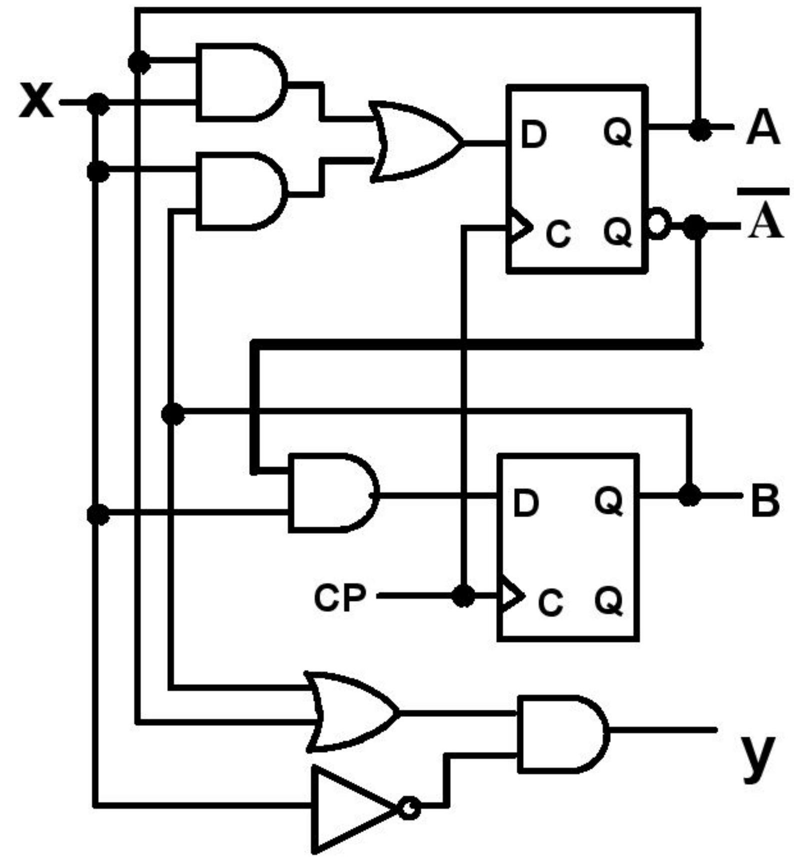
Sequential Circuits



- At each clock edge, the present state of the system is stored in storage elements (FFs)

Example 1

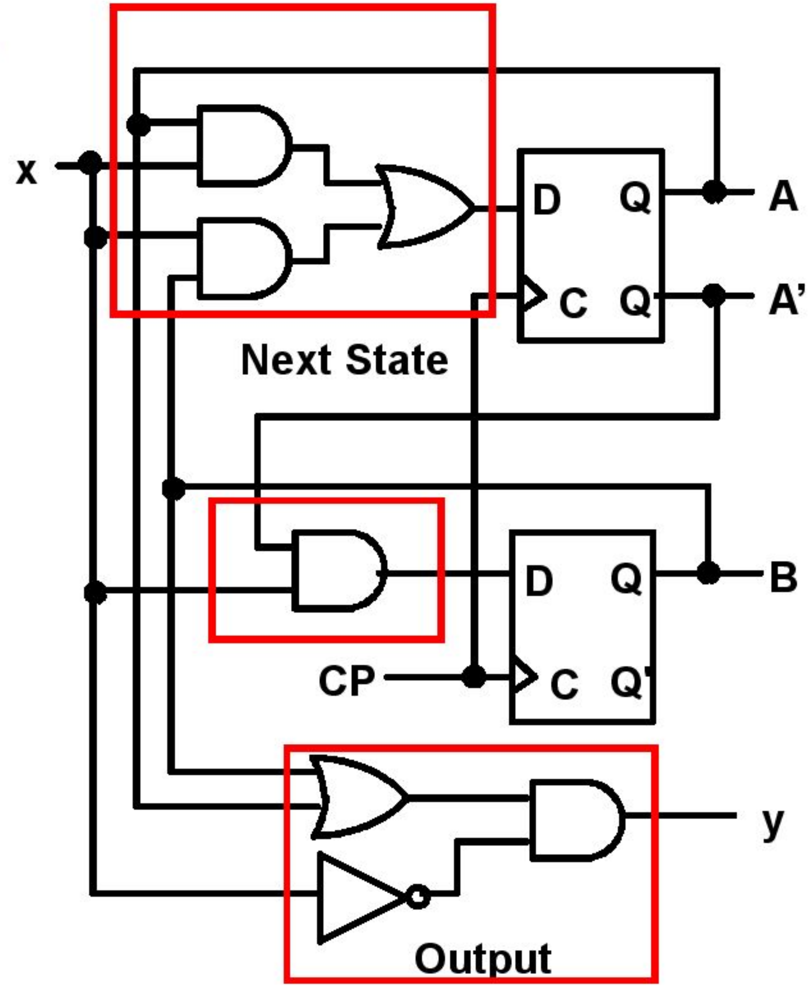
- **Input:** x(t)
- **Output:** y(t)
- **State:** (A(t), B(t))



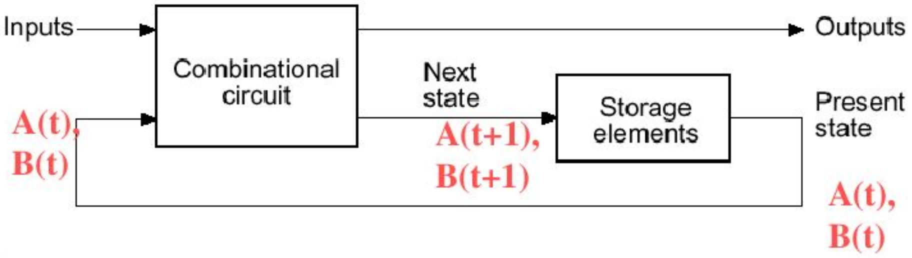
- What is the Output Function?
- What is the Next State Function?

Example 1 (Cont'd)

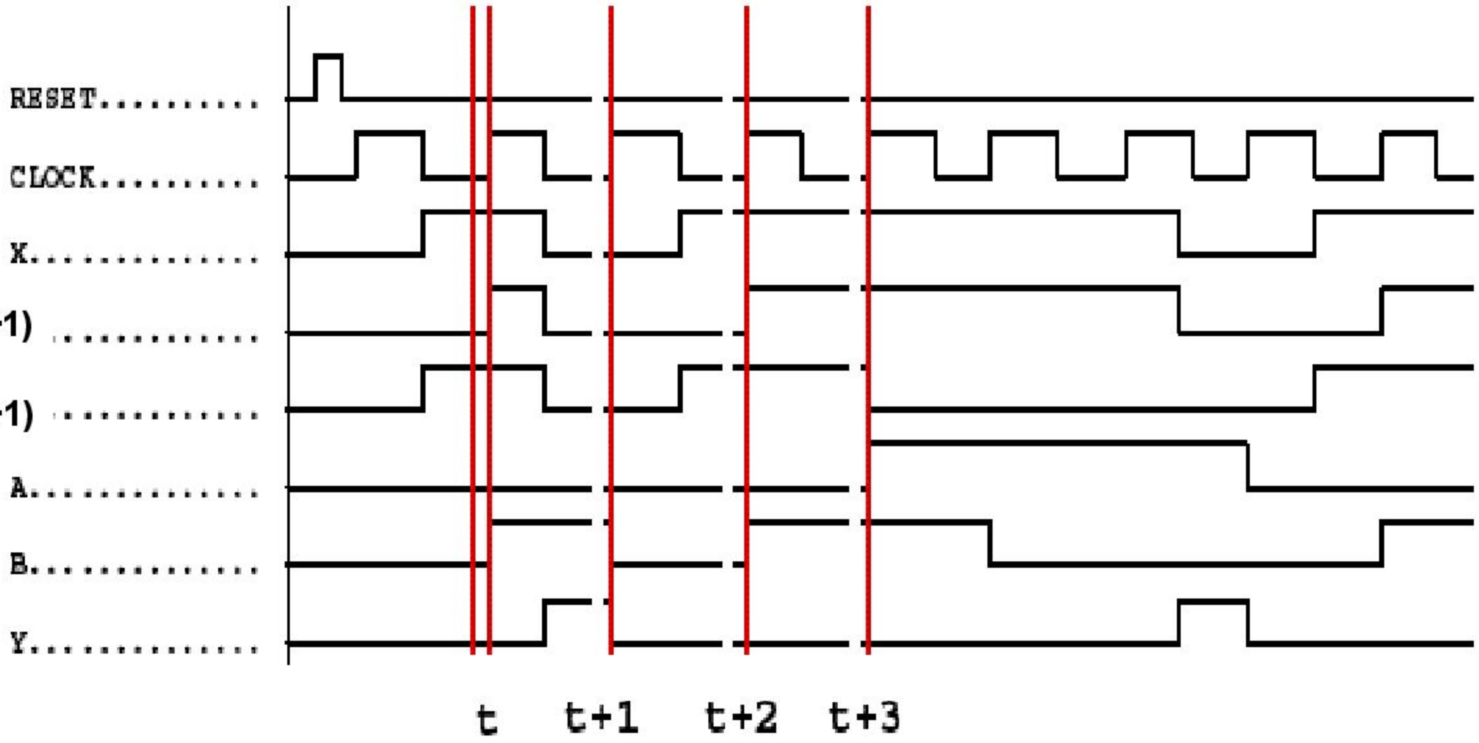
- **Boolean equations for the functions:**
  - $A(t+1) = A(t)x(t) + B(t)x(t)$
  - $B(t+1) = A'(t)x(t)$
  - $y(t) = x'(t) [B(t) + A(t)]$



Example 1 (Cont'd)



- $A(t+1) = A(t)x(t) + B(t)x(t)$
- $B(t+1) = A'(t)x(t)$
- $y(t) = x'(t)\ [B(t) + A(t)]$



State Table Definition

- ***State table***
  - A multiple variable table with the following four sections:
  - *Present State*
    - The values of the state variables for each allowed state
  - *Input*
    - The input combinations allowed
  - *Next-state*
    - The value of the state at time (t+1) based on the present state and the input
  - *Output*
    - The value of the output as a function of the present state and (sometimes) the input

State Table Characteristics

- **From the viewpoint of a truth table:**
  - ➢ The inputs: Input, Present State
  - ➢ The outputs: Output, Next State

State Table

- **Example:**
  - The state table can be filled in using the next state and output equations:

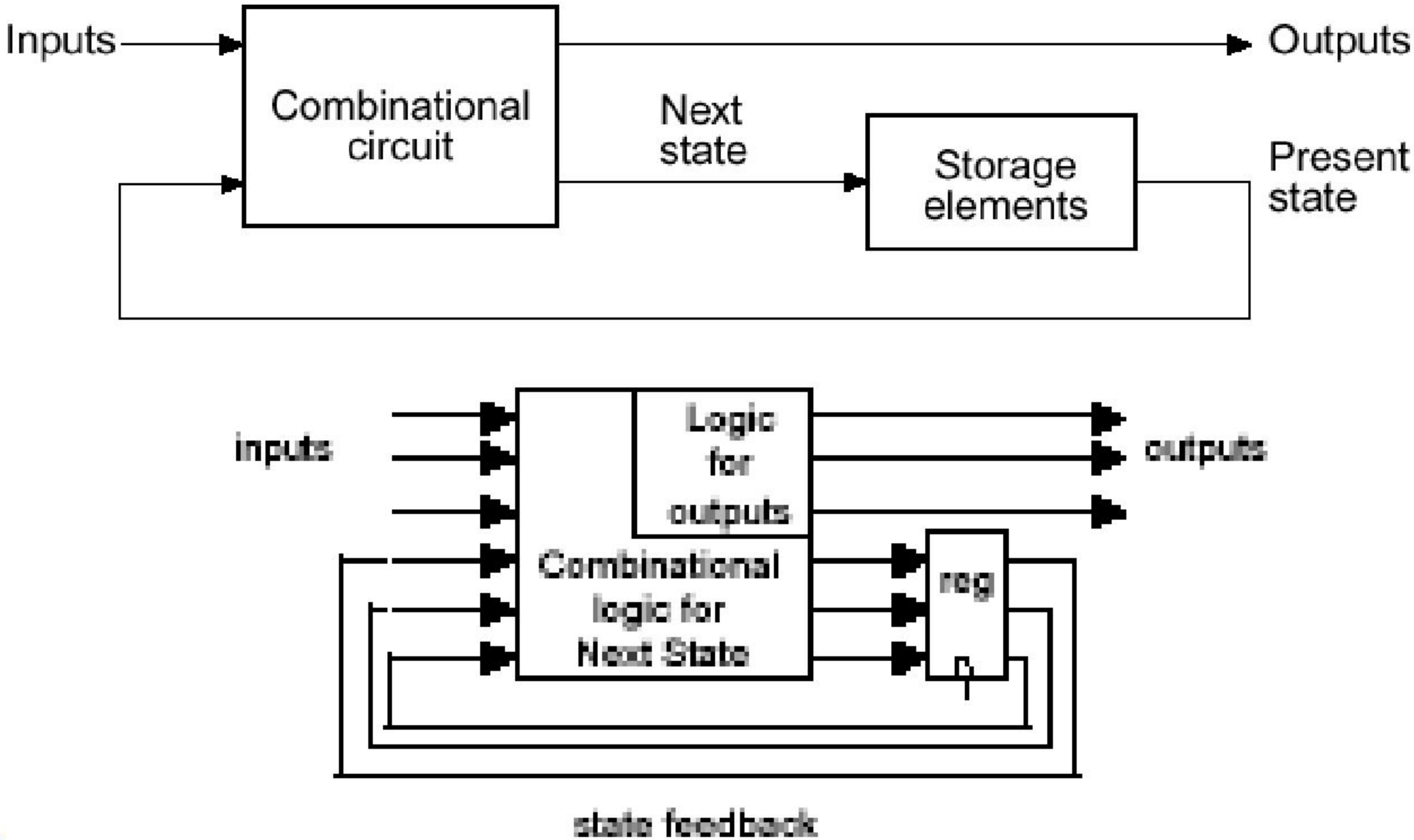
$$A(t+1) = A(t)x(t) + B(t)x(t)$$

$$B(t+1) = \bar{A}(t)x(t)$$

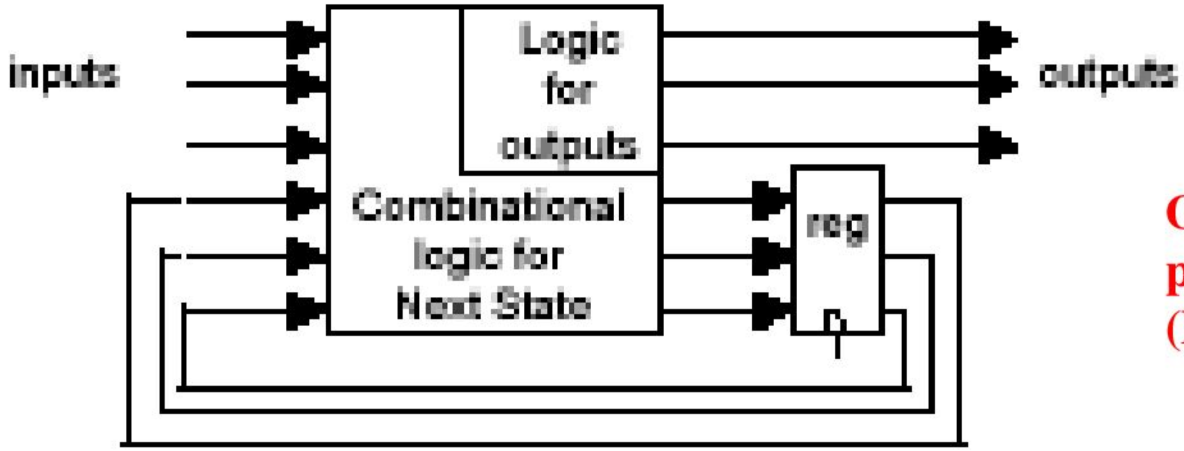
$$y(t) = \bar{x}(t)[B(t) + A(t)]$$

Present State		Input	Next State		Output
A(t)	B(t)	x(t)	A(t+1)	B(t+1)	y(t)
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

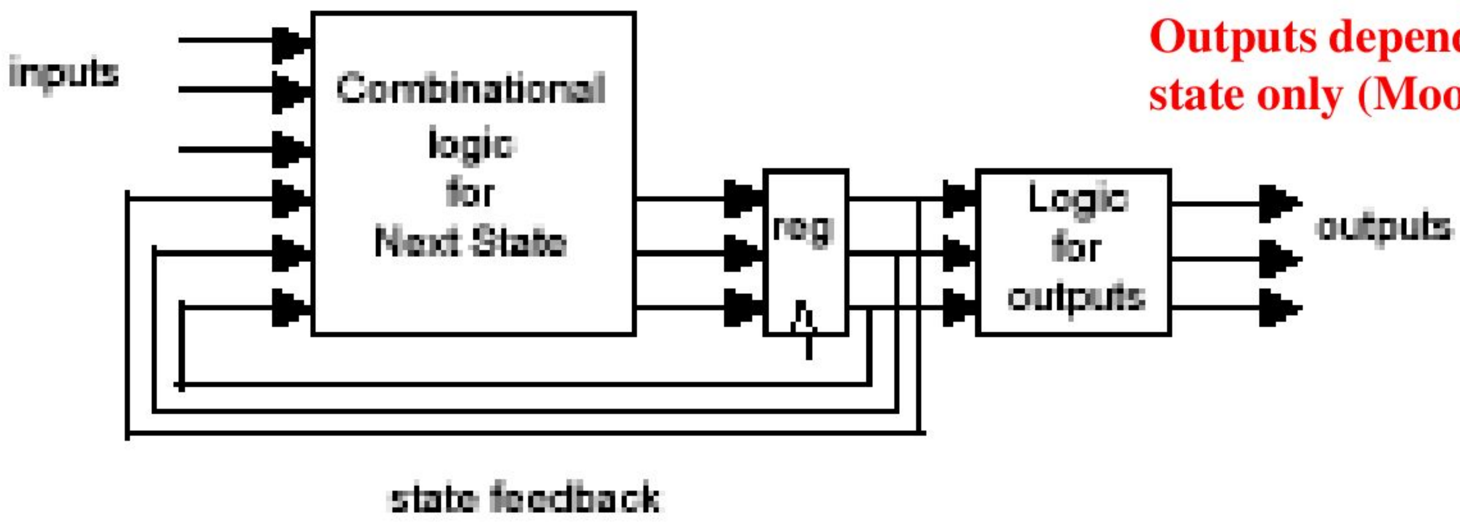
General Forms of Sequential Circuits (1)



General Forms of Sequential Circuits (2)



Outputs depend on present state and inputs (Mealy machine)



Outputs depend on present state only (Moore machine)

Mealy vs. Moore Machines

- **Mealy model:**
  - Both outputs and next state depend on primary inputs AND present state
    - Out = f(inputs, state)
    - Top diagram in the previous slide
- **Moore model:**
  - Only next state depends directly on primary inputs AND present state. Outputs depend only on present state
    - Out = f(state)
    - Bottom diagram in the previous slide

Mealy/Moore Comparison

- Mealy machines react faster to inputs
  - React in same cycle – don't need to wait for clock
  - In Moore machines, more logic may be necessary to decode state into outputs – more gate delays after
- Moore machines are easier and safer to use
  - Outputs change at clock edge (always one cycle later)
  - In Mealy machines, input change can cause output change as soon as logic is done – a big problem when two machines are interconnected – asynchronous feedback

State Diagram

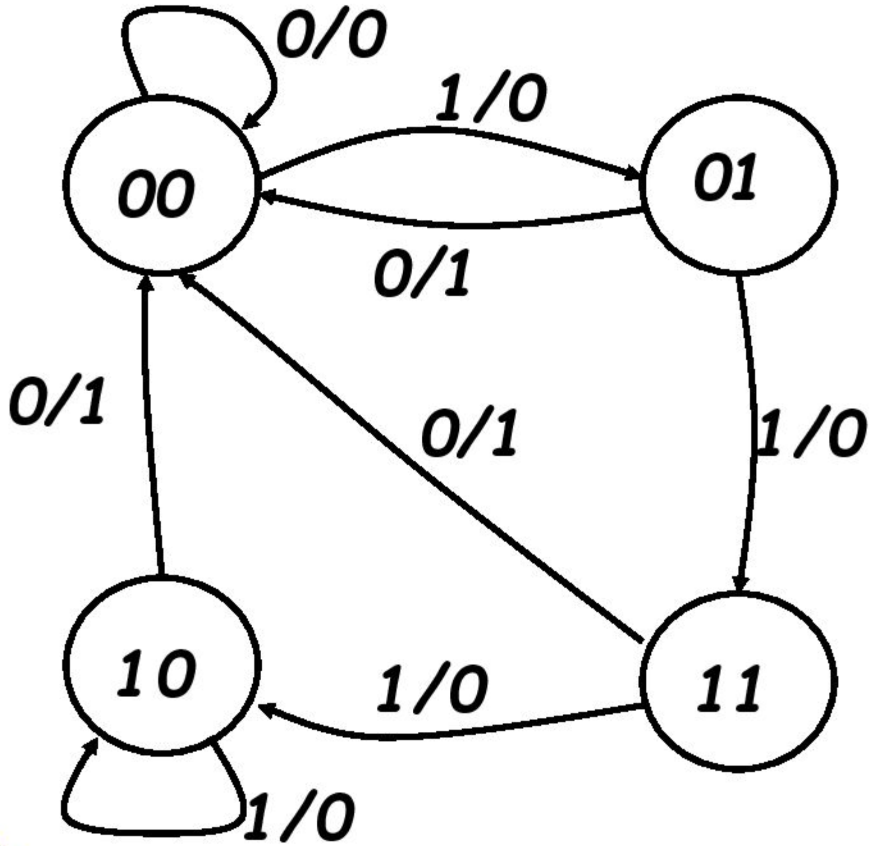
- The sequential circuit function can be represented in graphical form as a <u>state diagram</u> with the following components:
  - A <u>circle</u> with the state name in it for each state
  - A <u>directed arc</u> from the <u>Present State</u> to the <u>Next State</u> for each <u>state transition</u>
  - A label on each <u>directed arc</u> with the <u>Input</u> values which causes the <u>state transition</u>, and
  - A label:
    - In each <u>circle</u> with the <u>output</u> value produced (Moore), or
    - On each <u>directed arc</u> with the <u>output</u> value produced (Mealy).

State Diagram

- **Label form:**
  - In circle (for Moore machines):
    - state/output
    - **Moore** type output depends only on state
  - On directed arc (for Mealy machines):
    - input/output
    - **Mealy** type output depends on state and input

Example: Mealy Model

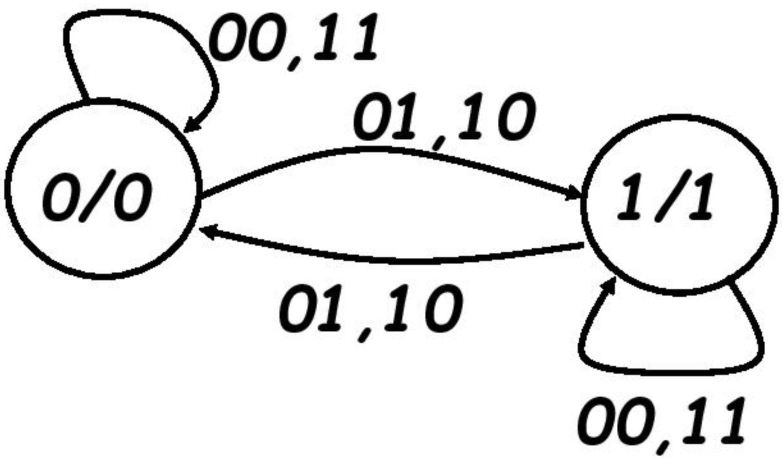
State Diagram

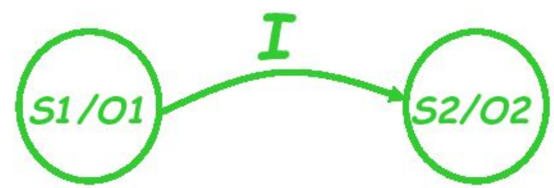


Reads as:
When at state *s1* and apply input *I*, we get output *O* and proceed to state *s2*.

Example: Moore Model

State Diagram

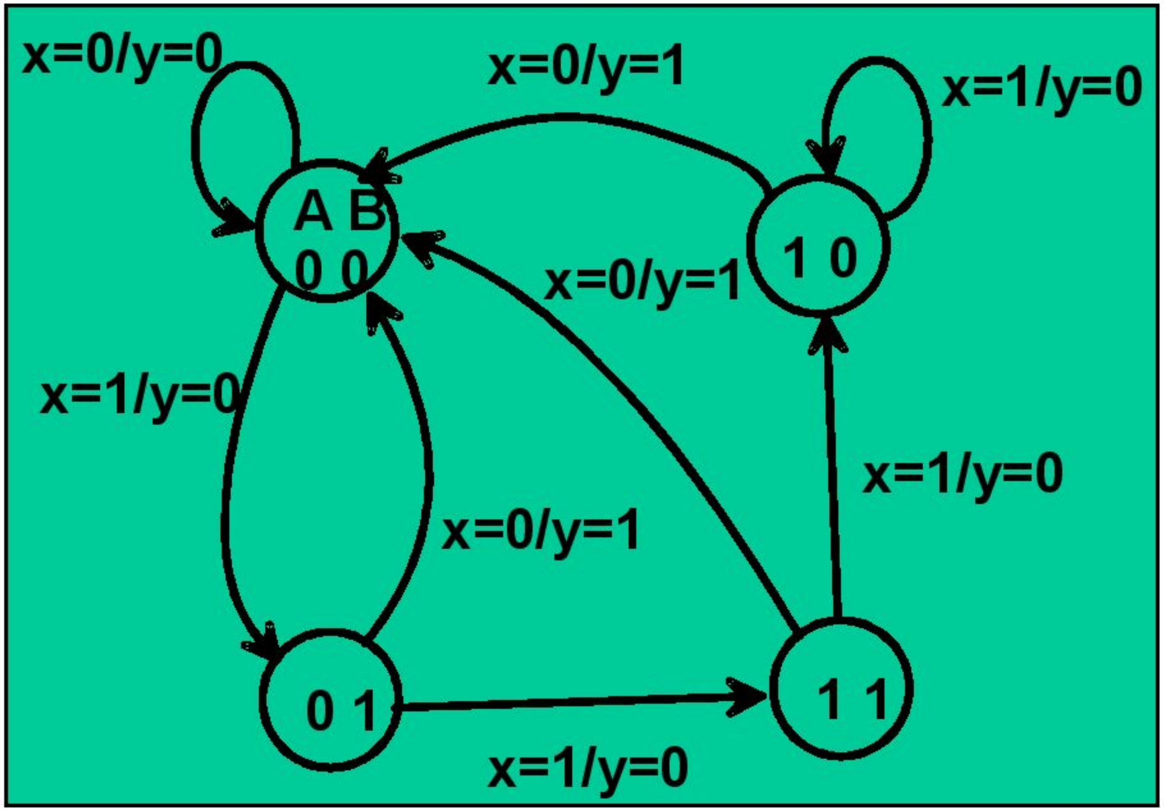




Reads as:
When at state ***s1*** with output ***O1*** and apply input ***I***,
we proceed to state ***s2*** with output ***O2***.

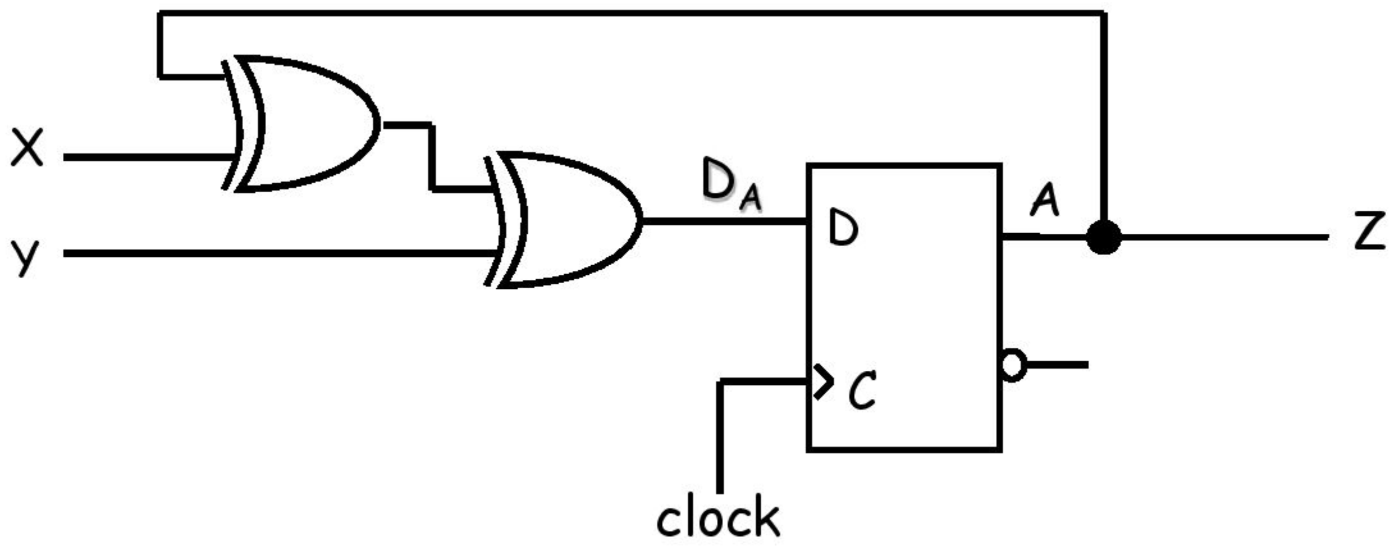
State Diagram Example

- Which type?

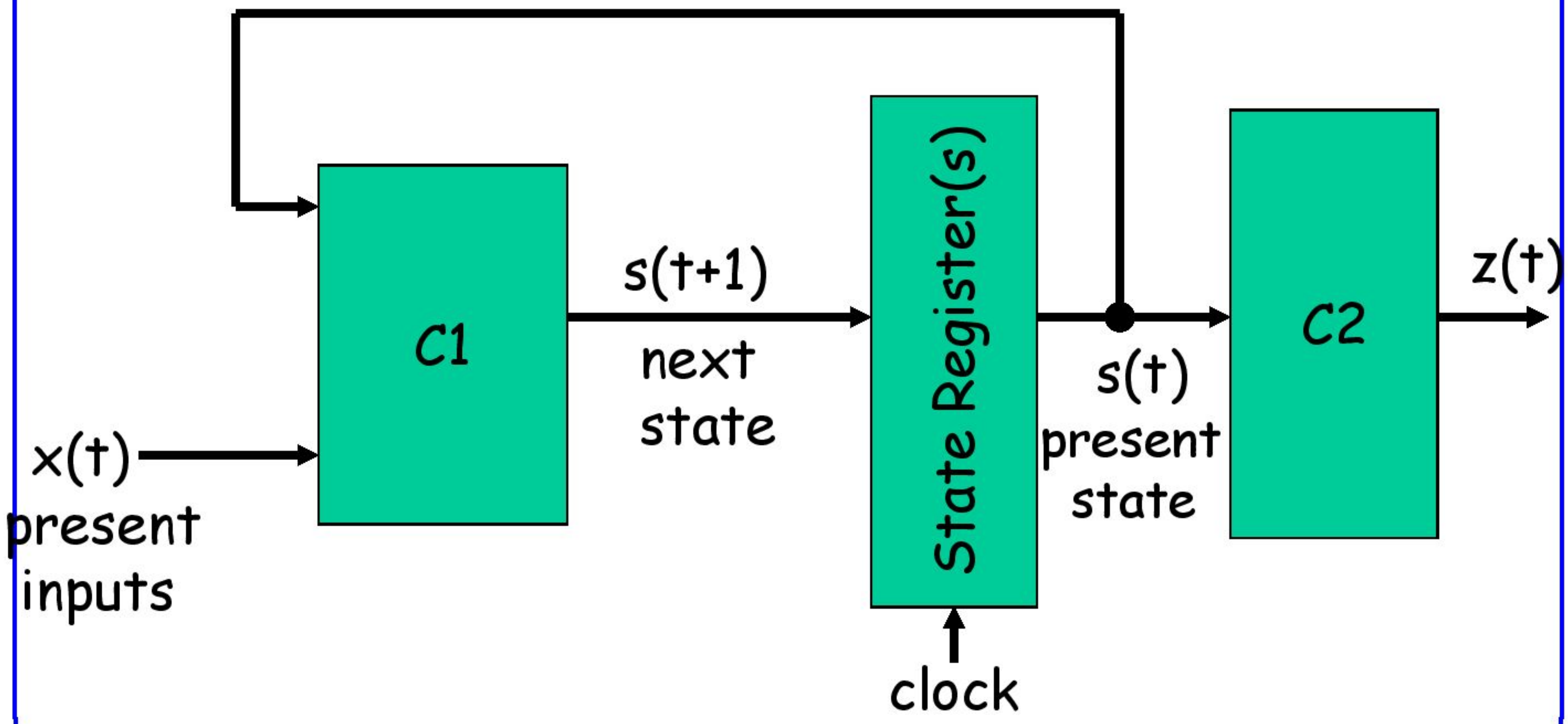


Example of a M…(?) Machine

- Obtain the logic expression and state table for this circuit:

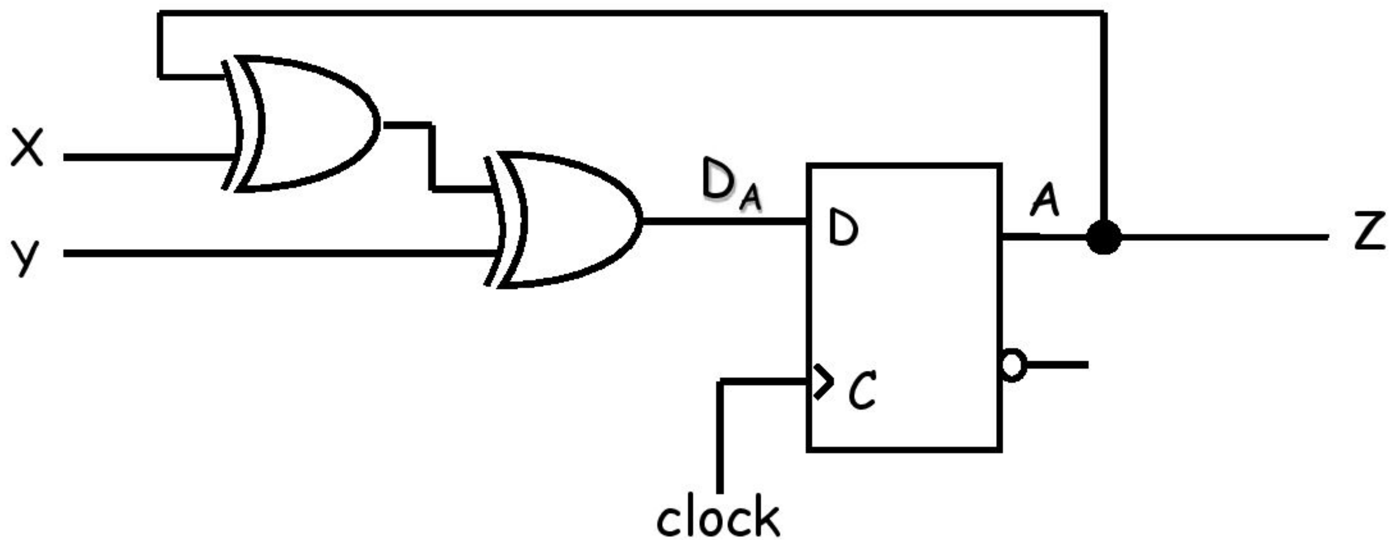


Moore Machine Reminder



Example of a Moore Machine

- **Obtain the logic expression and state table for this circuit:**



$D_A = A \oplus X \oplus Y$

$Z = A$

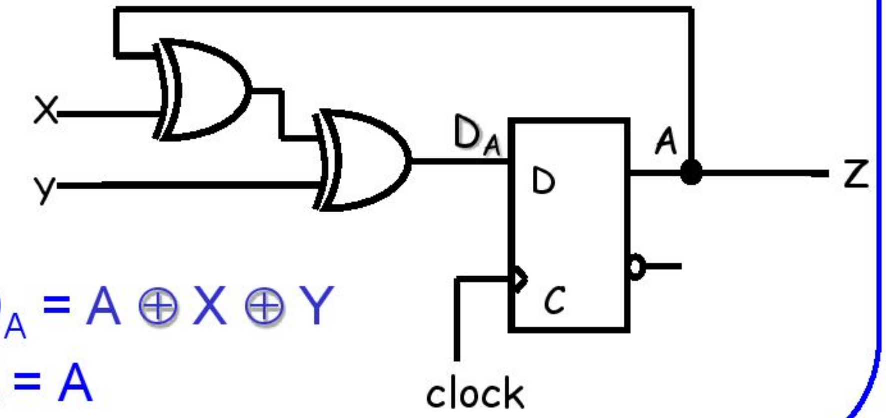
Example of a Moore Machine (cont.)

State Table

Present State	Inputs		Next State	Output
A(t)	X	Y	A(t+1)	Z
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Alternative State Table

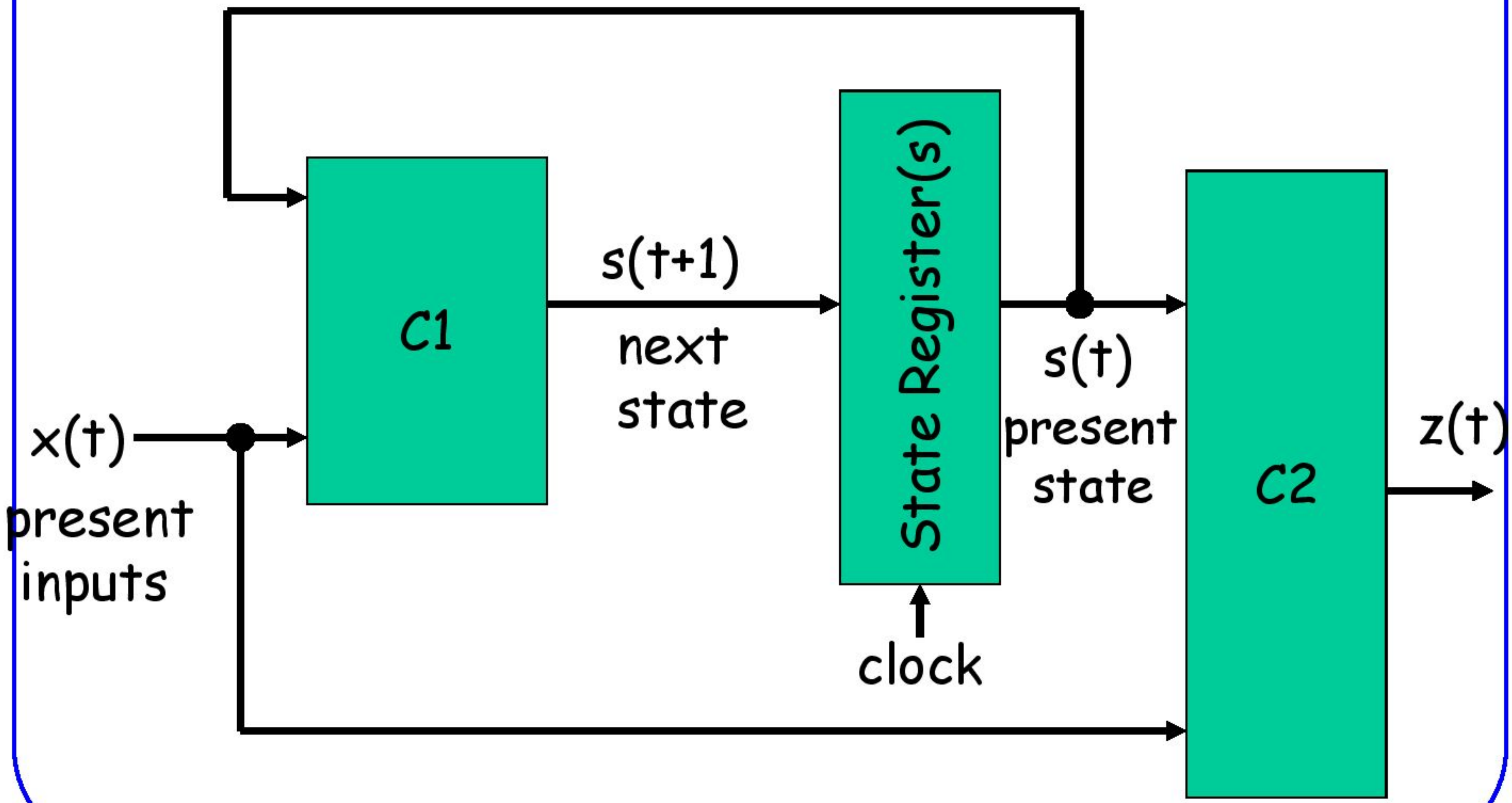
Present State	Next State				Output
	XY=00	XY=01	XY=10	XY=11	
A(t)	A(t+1)	A(t+1)	A(t+1)	A(t+1)	Z
0	0	1	1	0	0
1	1	0	0	1	1



$D_A = A \oplus X \oplus Y$

$Z = A$

Mealy Machine Reminder



Example: Mealy Model

Draw the state diagram of the machine whose state table is like this:

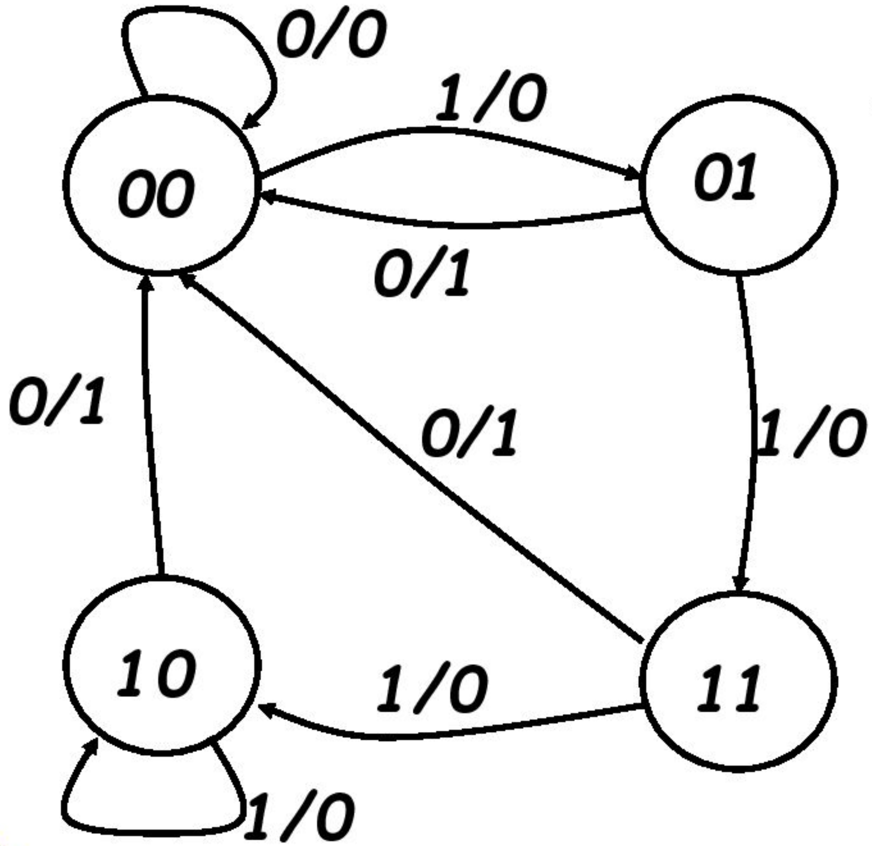
State Table

Present State		Input	Next State		Output
A(t)	B(t)	X	A(t+1)	B(t+1)	Y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

Possible states = {00, 01, 10, 11}
→ 4 nodes in state diagram

Example: Mealy Model (cont.)

State Diagram



Present State		Input	Next State		Output
A(t)	B(t)	X	A(t+1)	B(t+1)	Y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

Example: Moore Model

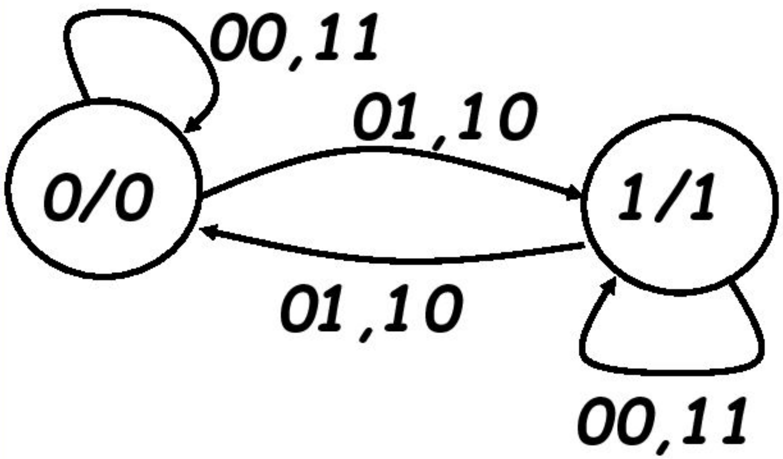
State Table

Present State	Inputs		Next State	Output
A(t)	X	Y	A(t+1)	Z
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Possible states = {0, 1}
→ 2 nodes in state diagram

Example: Moore Model (cont.)

State Diagram



Present State	Inputs		Next State	Output
A(t)	X	Y	A(t+1)	Z
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

State Tables for JK Flip-Flops

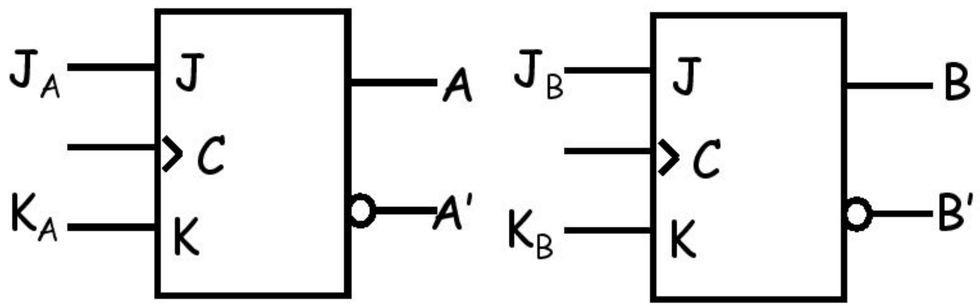
- **Two-step procedure:**
  1. Obtain binary values of each FF input equation in terms of present state and input variables.
  2. Use corresponding FF characteristic table to determine the next state.

Example

$J_A = B$, $K_A = BX'$

$J_B = X'$, $K_B = AX' + A'X = A \oplus X$

- → 2 JK-FFs:



JK-FF Characteristic Table

J	K	Q(t+1)
0	0	Q(t)
0	1	0
1	0	1
1	1	Q(t)'

Example (cont.)

Present State		Input	Next State		FF inputs			
A(t)	B(t)	X	A(t+1)	B(t+1)	J_A	K_A	J_B	K_B
0	0	0			0	0	1	0
0	0	1			0	0	0	1
0	1	0			1	1	1	0
0	1	1			1	0	0	1
1	0	0			0	0	1	1
1	0	1			0	0	0	0
1	1	0			1	1	1	1
1	1	1			1	0	0	0

$J_A = B$, $K_A = BX'$

$J_B = X'$, $K_B = AX' + A'X = A \oplus X$

Step 1:
Use FF input equations

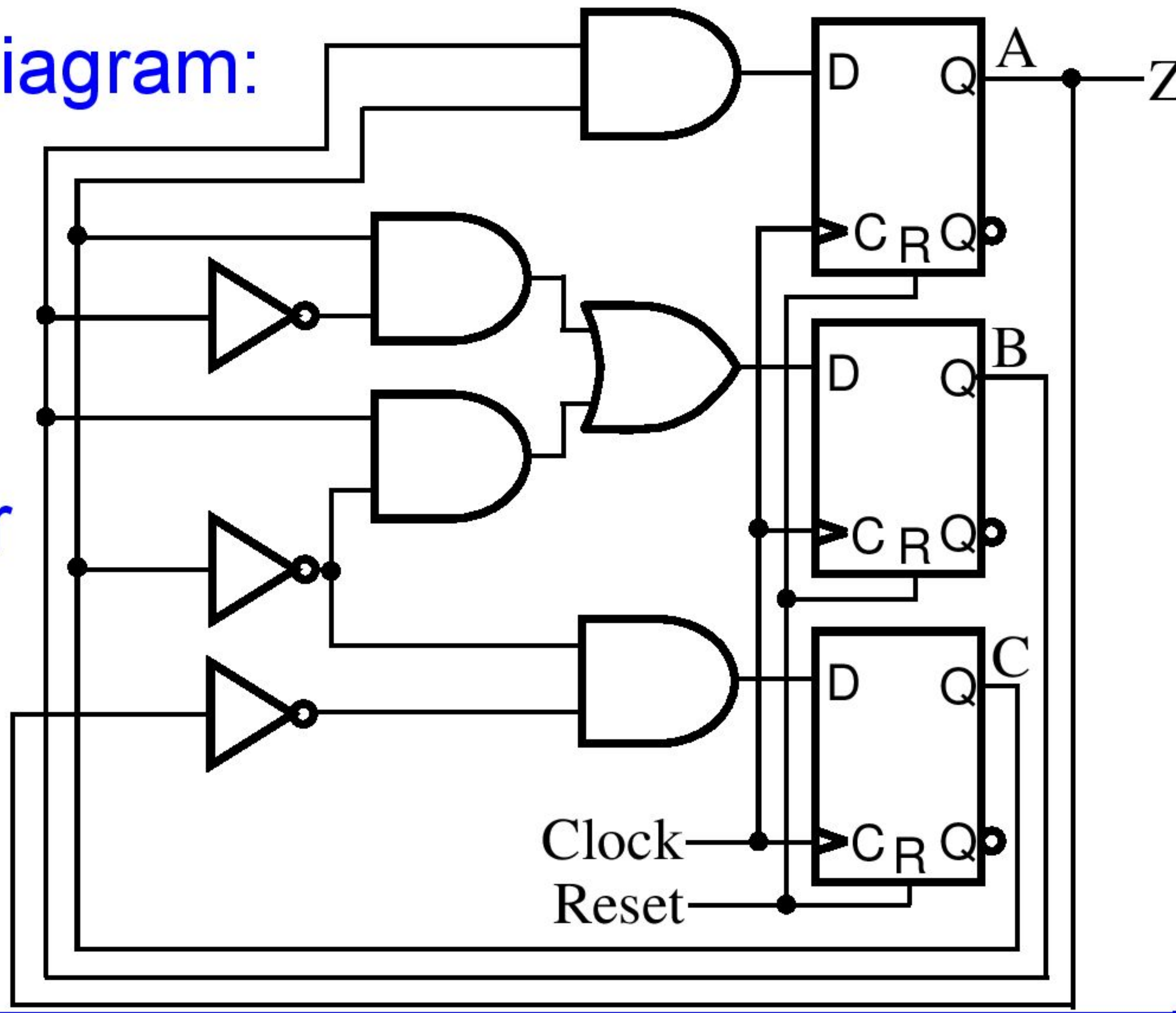
Example (cont.)

Present State		Input	Next State		FF inputs			
A(t)	B(t)	X	A(t+1)	B(t+1)	J_A	K_A	J_B	K_B
0	0	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0	1
0	1	0	1	1	1	1	1	0
0	1	1	1	0	1	0	0	1
1	0	0	1	1	0	0	1	1
1	0	1	1	0	0	0	0	0
1	1	0	0	0	1	1	1	1
1	1	1	1	1	1	0	0	0

Step 2:
Use FF inputs and JK characteristic table

Example 2: Sequential Circuit Analysis

- Logic Diagram:



- Mealy or Moore?

Example 2: Flip-Flop Input Equations

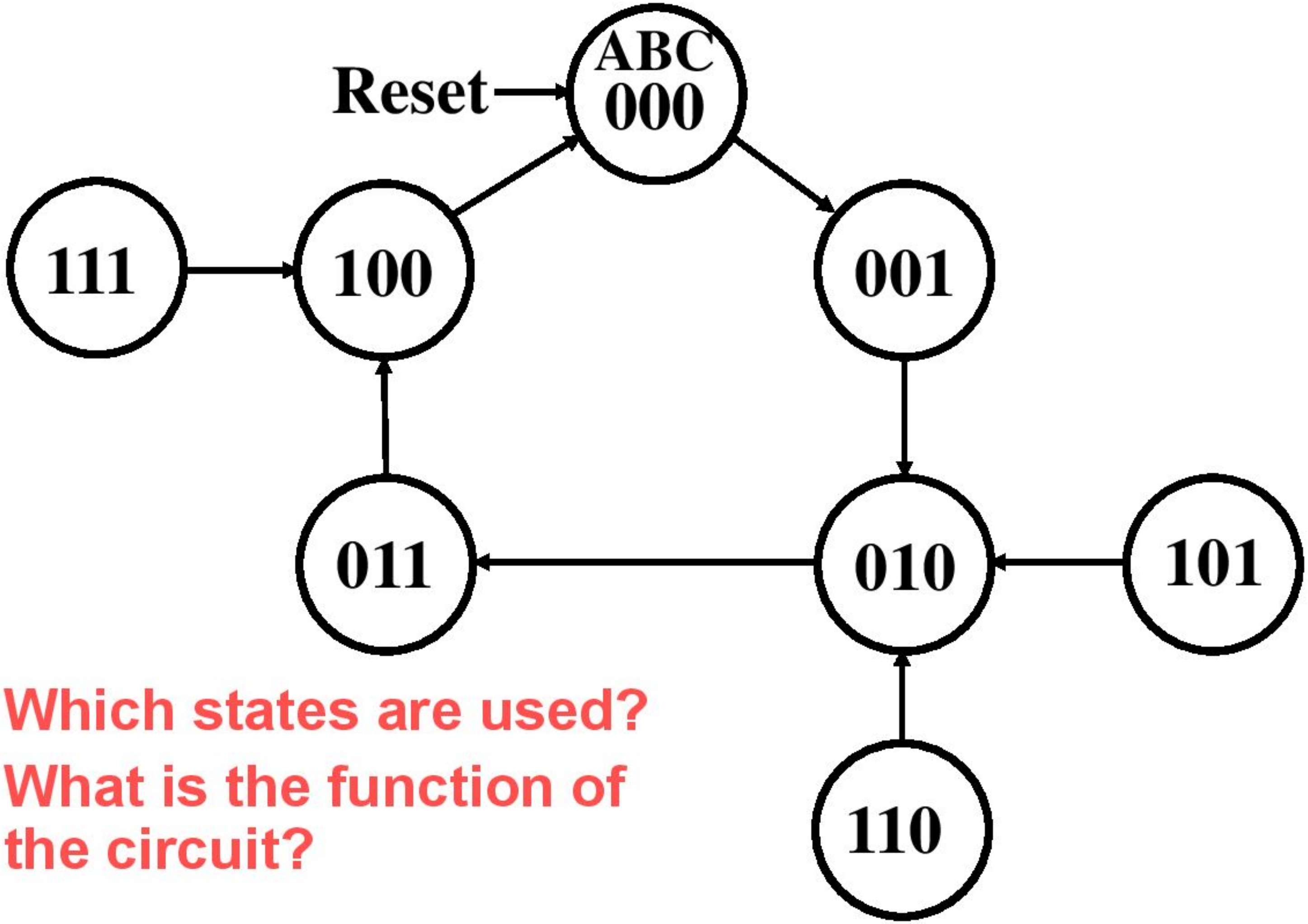
- **Variables**
  - Inputs: None
  - Outputs: Z
  - State Variables: A, B, C
- **Initialization:**
  - Reset to (0,0,0)
- **Equations**
  - $A(t+1) = ?$ BC
  - $B(t+1) = ?$ $B'C+BC'$
  - $C(t+1) = ?$ $A'C'$
  - $Z = ?$ A

Example 2: State Table

$S^{+}=S(t+1)$

A B C	$A^{+}B^{+}C^{+}$	Z
0 0 0		
0 0 1		
0 1 0		
0 1 1		
1 0 0		
1 0 1		
1 1 0		
1 1 1		

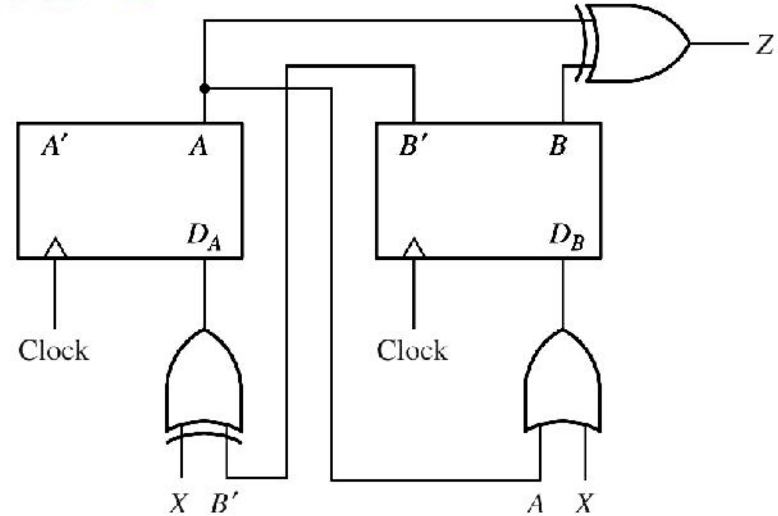
Example 2: State Diagram



- **Which states are used?**
- **What is the function of the circuit?**

Example 3

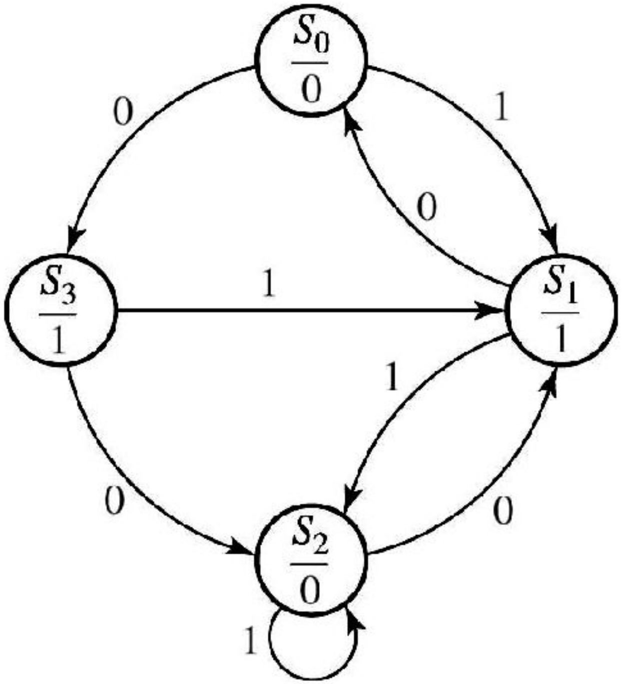
- **Mealy or Moore?**

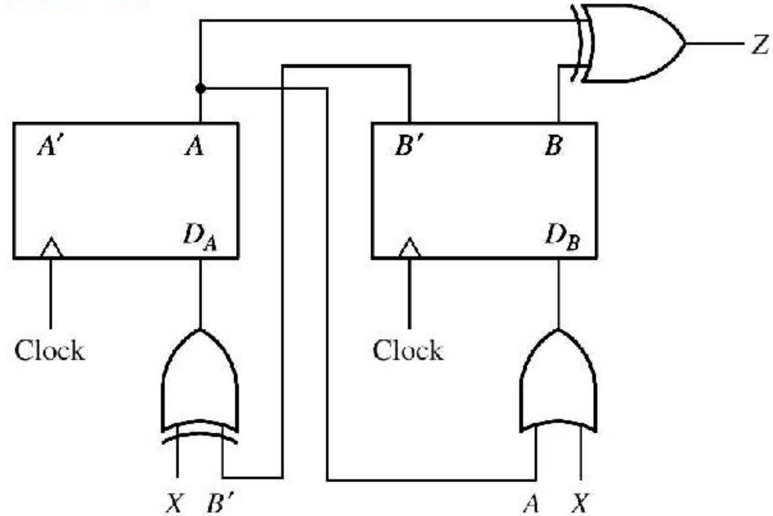


AB	$X=0$ A^+B^+	$X=1$ A^+B^+	Z
00	10	01	0
01	00	11	1
11	01	11	0
10	11	01	1

Present State	Next state $X=0$	Next state $X=1$	Present Output(z)
S_0	S_3	S_1	0
S_1	S_0	S_2	1
S_2	S_1	S_2	0
S_3	S_2	S_1	1

Example 3

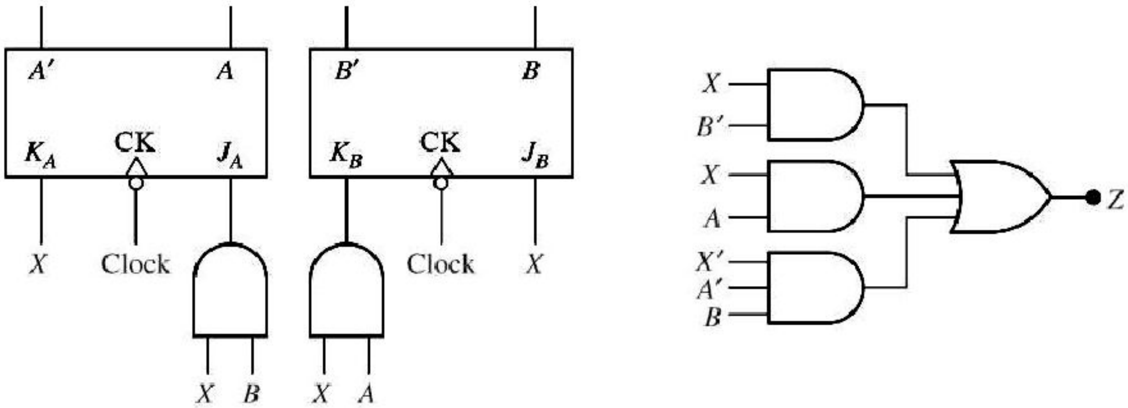




Present State	Next state $X=0$	Next state $X=1$	Present Output(z)
S_0	S_3	S_1	0
S_1	S_0	S_2	1
S_2	S_1	S_2	0
S_3	S_2	S_1	1

Example 4

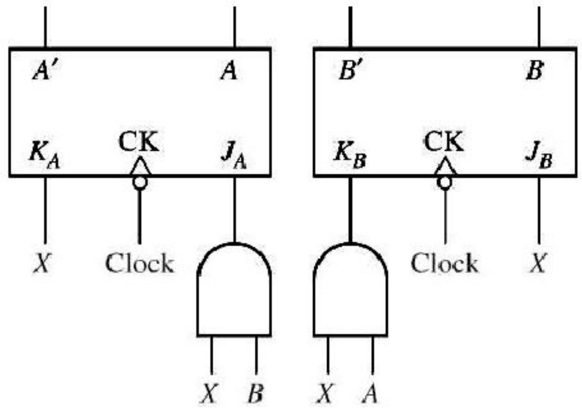
- Mealy or Moore?

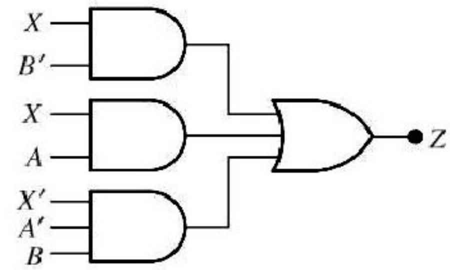


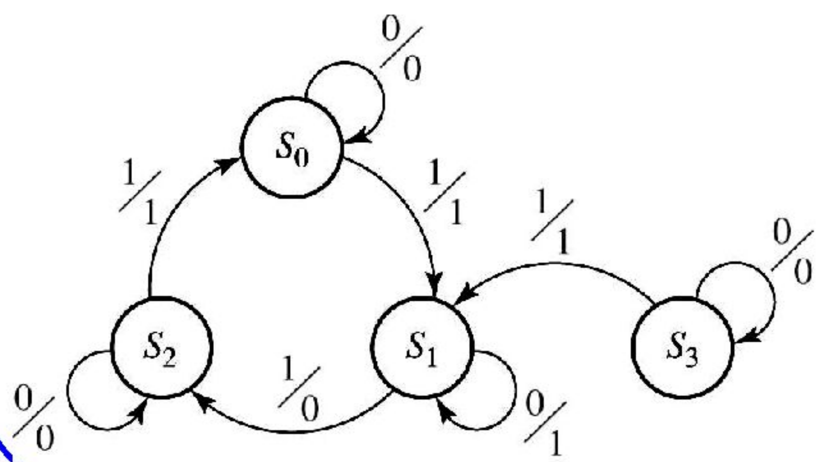
AB	A^+B^+ $X=0$	$X=1$	Z $X=0$	$X=1$
00	00	01	0	1
01	01	11	1	0
11	11	00	0	1
10	10	01	0	1

Present State	Next state $X=0$	$X=1$	Present Output(z) $X=0$	$X=1$
S_0	S_0	S_1	0	1
S_1	S_1	S_2	1	0
S_2	S_2	S_0	0	1
S_3	S_3	S_1	0	1

Example 4







Present State	Next state $X=0$	Next state $X=1$	Present Output(z) $X=0$	Present Output(z) $X=1$
S_0	S_0	S_1	0	1
S_1	S_1	S_2	1	0
S_2	S_2	S_0	0	1
S_3	S_3	S_1	0	1

Circuit Analysis by Signal Tracing

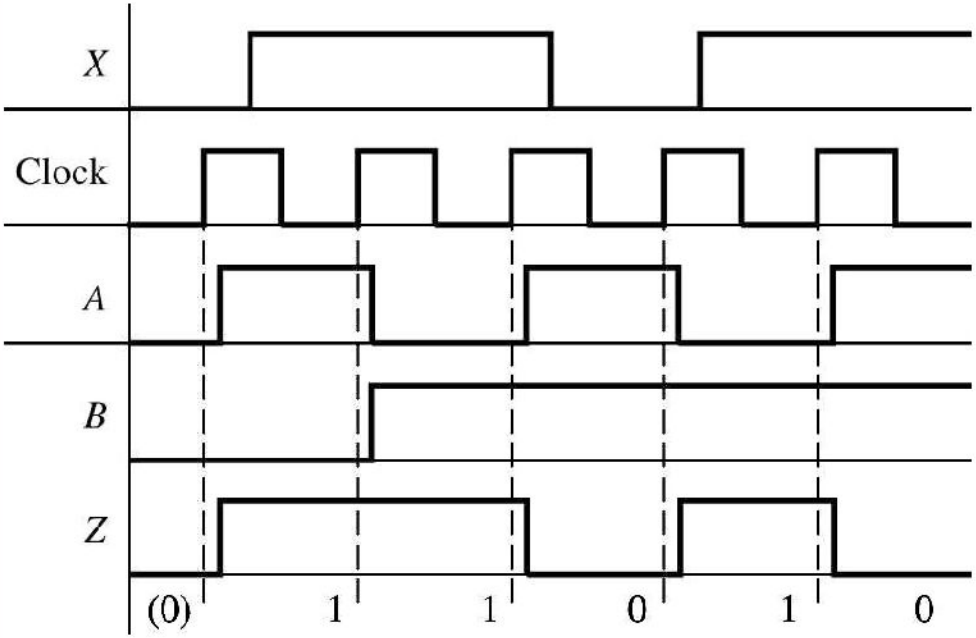
Waveforms

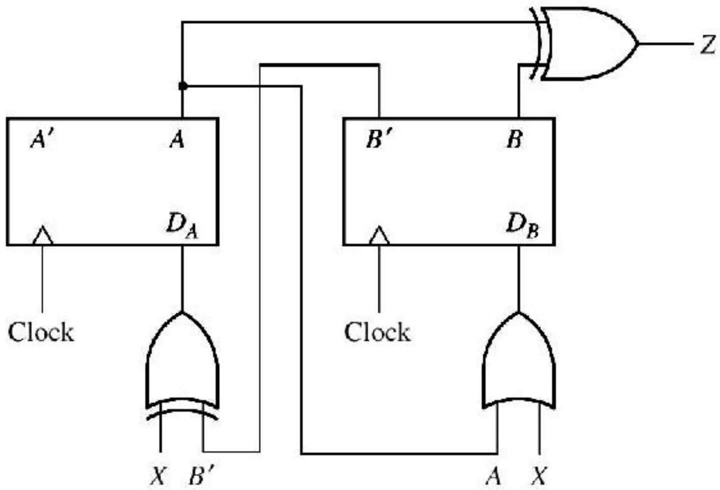
Analysis by Signal Tracing

- **<u>Output tracing</u>**
  1. Assume an initial state of FF
  2. For the first input in the given sequence, determine the output and FF inputs
  3. Determine the new sets of FF states after the next clock edge
  4. Determine the output that corresponds to the new states
  5. Repeat 2,3,4 for each in the given sequence

Example

Timing Chart for a Moore Machine





X = 0 1 1 0

A = 0 1 0 1 0 1

B = 0 0 1 1 1 1

Z = (0) 1 1 0 1 0

Timing Chart

- ## Construction and interpretation of a timing chart:

  - A state change can only occur after the rising (or falling) edge of the clock.
  - The input will normally be stable immediately before and after the active clock edge.
  - For a Mealy circuit, the output can change when the input changes as well as when the state changes.
    - A false output may occur between the state changes and the time the input changes to its new value.

  - False outputs are difficult to determine from the state diagram, so sometimes have to use signal tracing (timing chart).

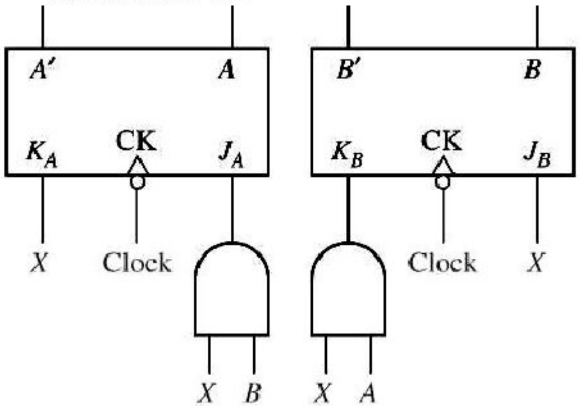
Timing Charts

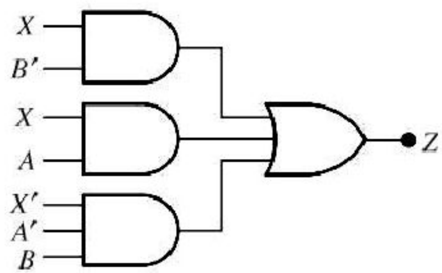
- When using a Mealy state table for constructing timing charts, the procedure is as follows:
  a) For the given input value, read the present output and plot it.
     a) If the input changes many times before clock edge, plot the effects on output.
  b) After the active clock edge, change the state according to the next state function.
  c) Repeat steps (a) and (b).

- For Mealy circuits, the best time to read the output is just before the active edge of the clock, because the output should always be correct at that time.

Example

Timing Chart for a Mealy Machine





$J_A=0$

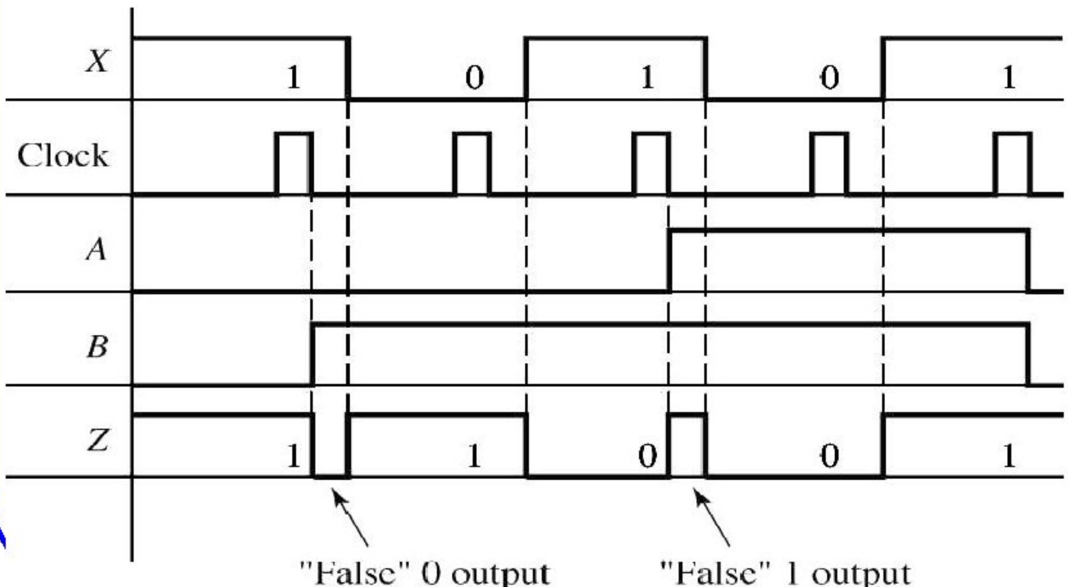
$K_A=1$

$J_B=1$

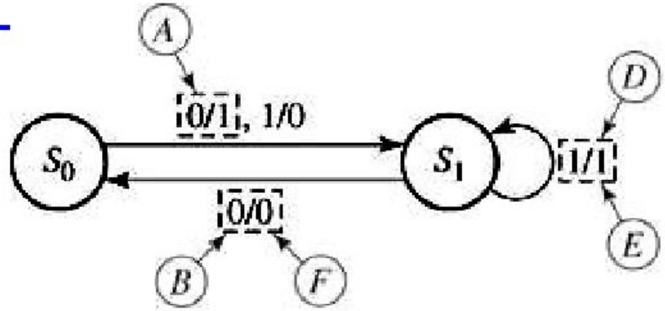
$K_B=0$

➔ **Reset A, Set B**

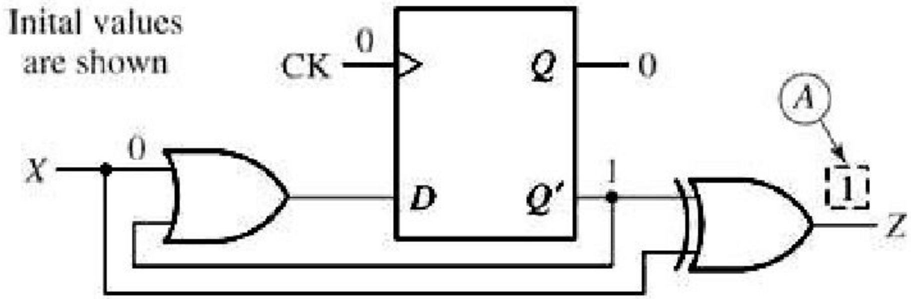
(False outputs are indicated in parentheses)

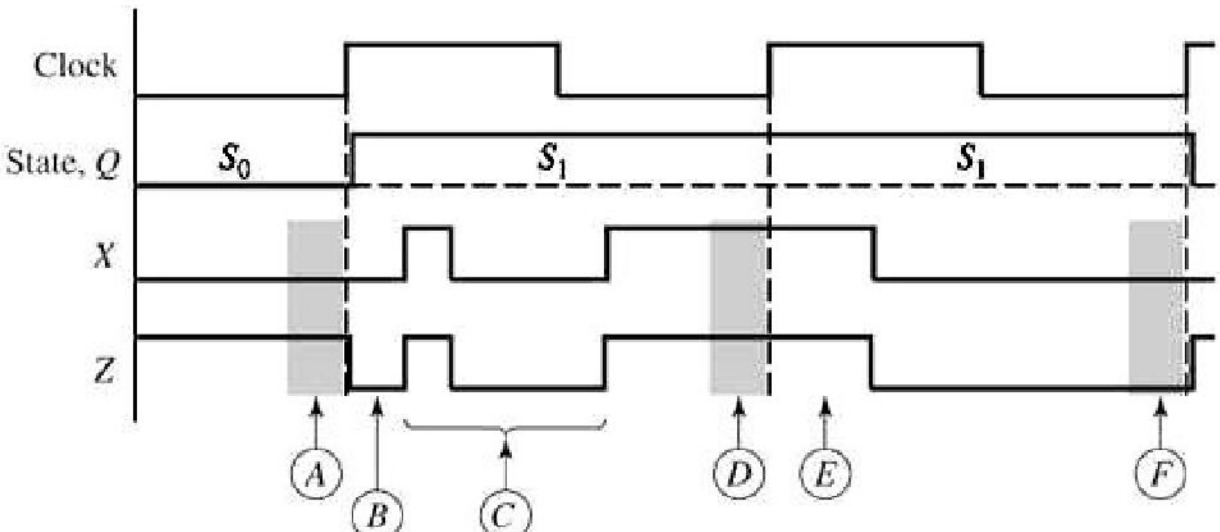


X =	1	0	1	0	1	
A =	0	0	0	1	1	0
B =	0	1	1	1	1	0
Z =	1	1	( )	0	1	



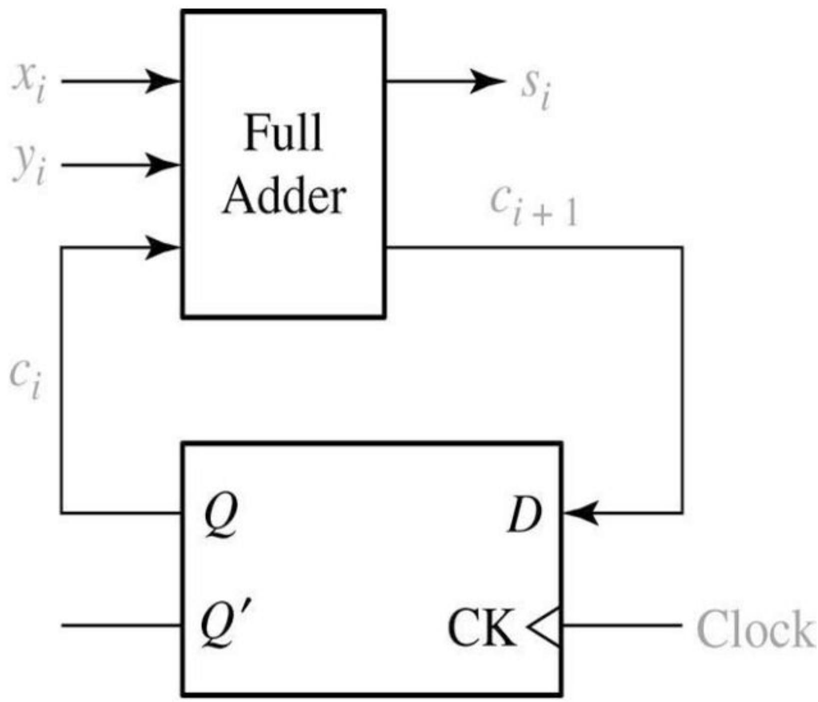
Q		$X = 0$	1	0	1
0	S_0	S_1	S_1	$\boxed{1}$	0
1	S_1	S_0	S_1	$\boxed{0}$	$\boxed{1}$





Read X and Z in shaded area
(before rising edge of clock).

Example: Serial Adder



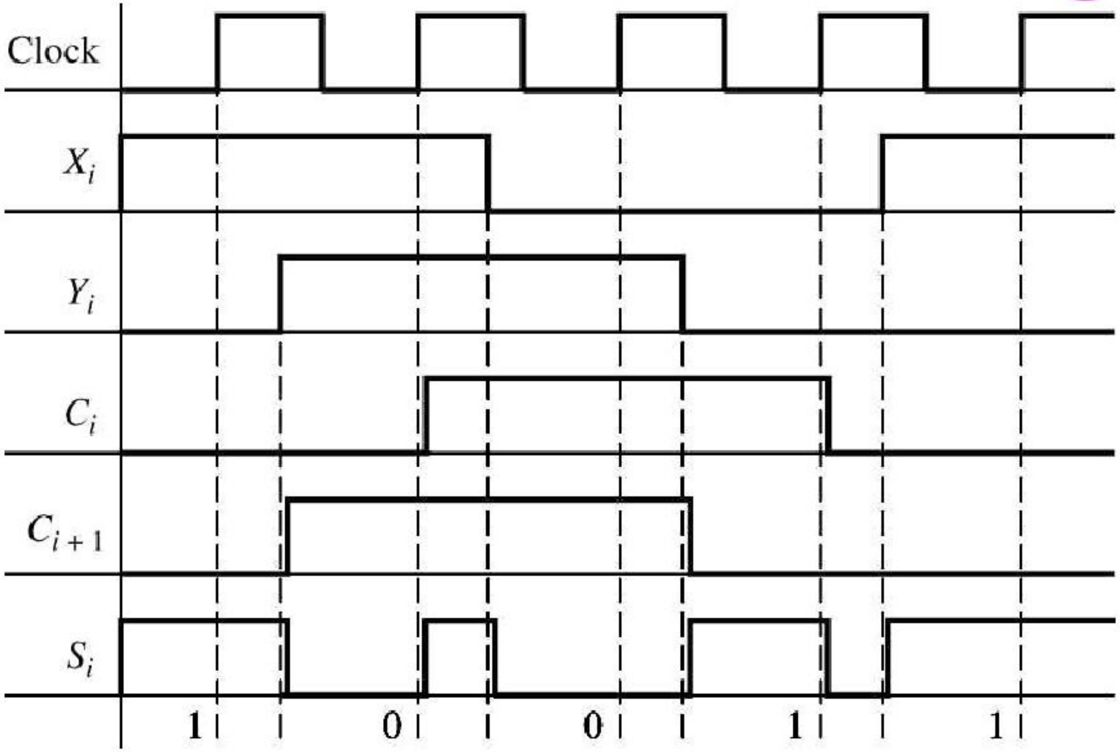
(a) With *D* flip-flop

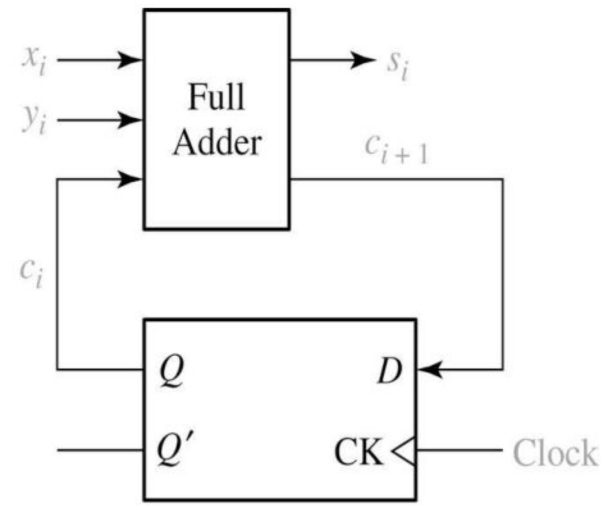
x_i	y_i	c_i	c_{i+1}	s_i
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

(b) Truth table

What kind of machine (Mealy or Moore)?

Serial Adder Timing Diagram



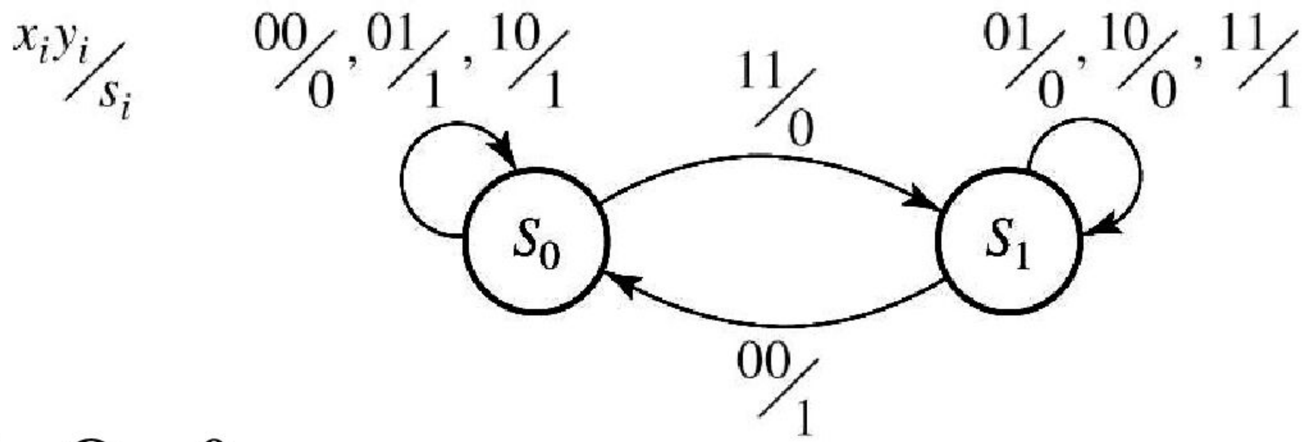


(a) With D flip-flop

- S_i can be read **before** the rising edge of clock
  - Before 2nd clock edge: $X_i=Y_i=1$, $C_i=0 \rightarrow S_i=0$, $C_{i+1}=1$
  - After clock: $C_{i+1} \rightarrow C_i$, X_i still has its old value $\rightarrow S_i$ momentarily changes to a wrong value

Serial Adder

- **State Diagram**



S_0: Q = 0

S_1: Q = 1

State Tables

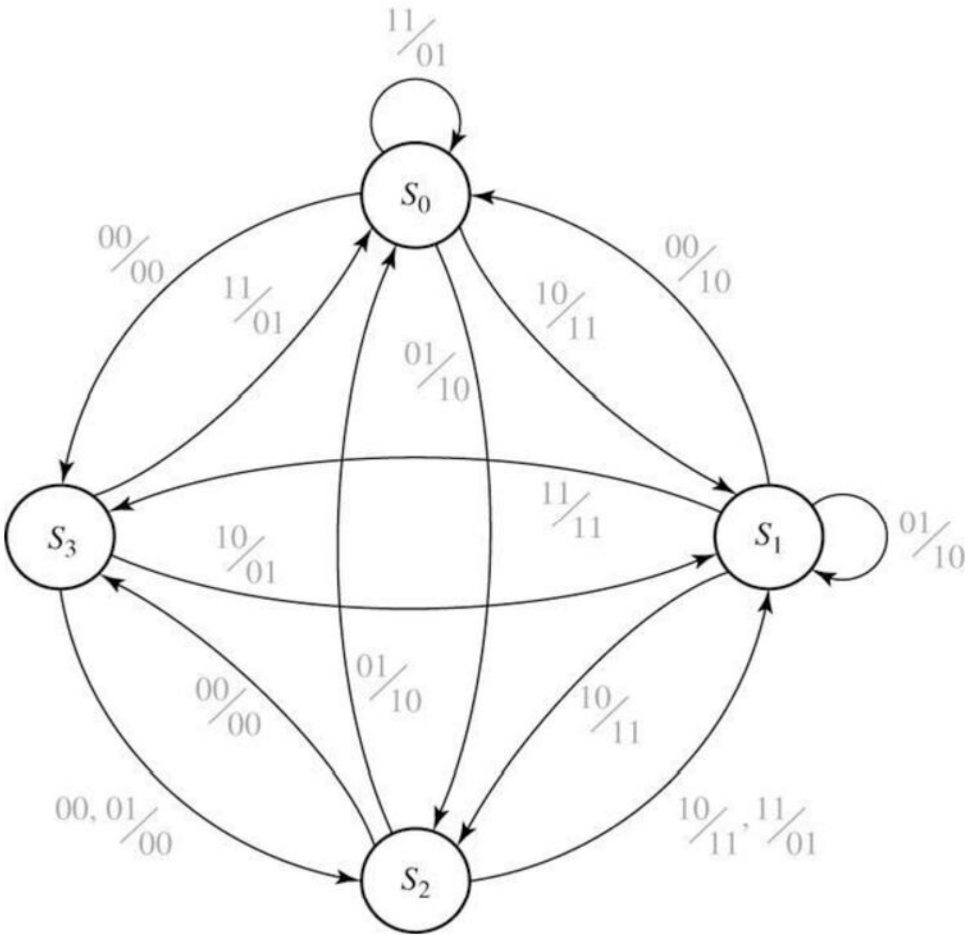
Example of a state table with multiple inputs and outputs:

(2 state variables => 4 states, 2 inputs, 2 outputs)

Present State	Next state				Outputs			
	$X_1X_2 = 00$	01	10	11	$X_1X_2 = 00$	01	10	11
S_0	S_3	S_2	S_1	S_0	00	10	11	01
S_1	S_0	S_1	S_2	S_3	10	10	11	11
S_2	S_3	S_0	S_1	S_1	00	10	11	01
S_3	S_2	S_2	S_1	S_0	00	00	01	01

State Diagram

The state diagram of the previous example



Mealy Machine State Diagrams

- A CLARIFICATION:
  - The state diagram notation for output values in Mealy machines is a little misleading:
    - You should remember that the listed output value is produced continuously when the machine is in the indicated state and has the indicated input, not just during the transition to the next state.