Lecture – 11 - Chapter 5

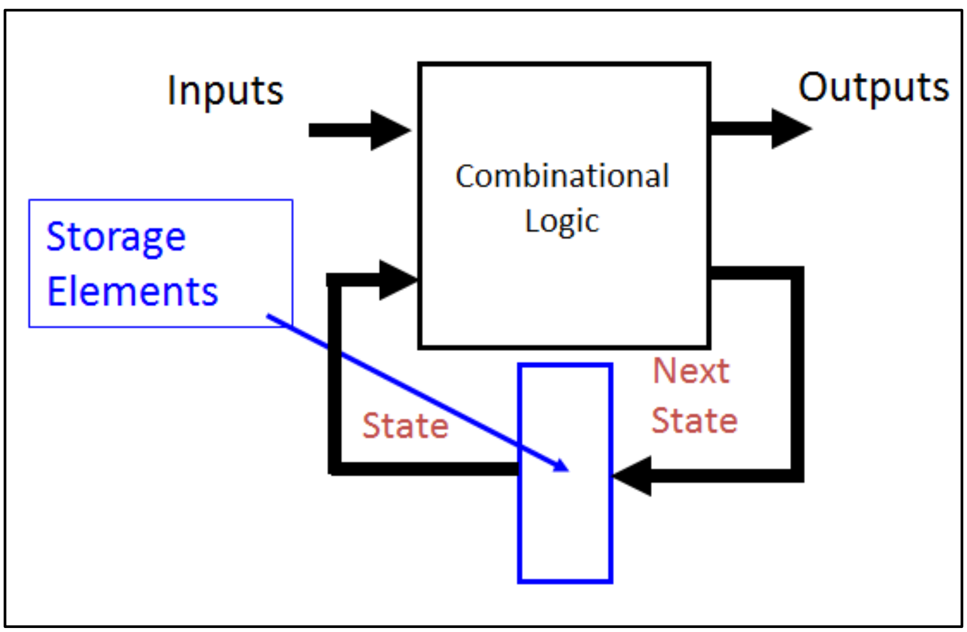
Outline

- sequential circuits
- storage elements: latch and flip-flop

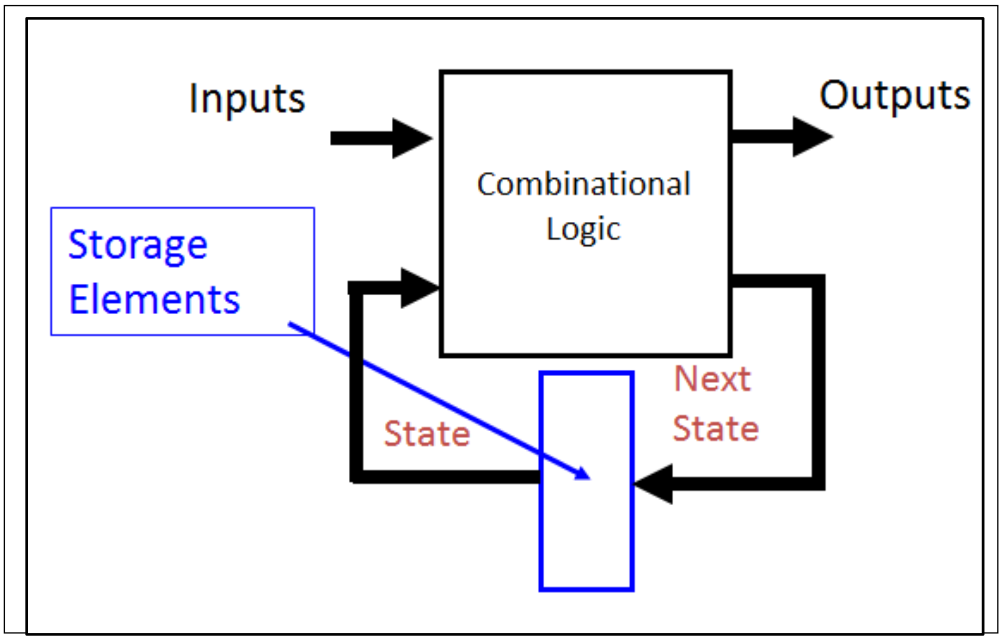
- Sequential circuit analysis
    - ***Moore and Mealy Models***
    - ***State tables***
    - ***State diagrams***

Introduction to Sequential Circuits

- A Sequential circuit contains:
  - Storage elements:
    Latches or Flip-Flops
  - Combinational Logic:
    - **Implements a multiple-output switching function**
    - Signals from the outside are **inputs.**
    - Signals to the outside are **outputs.**
    - **Other inputs,** State or Present State, are signals from storage elements.
    - **The remaining outputs, Next State are inputs to storage elements.**



- Combinatorial Logic



- *Next state function*
  Next State = f(Inputs, State)

- *Output is a function of inputs and State.*

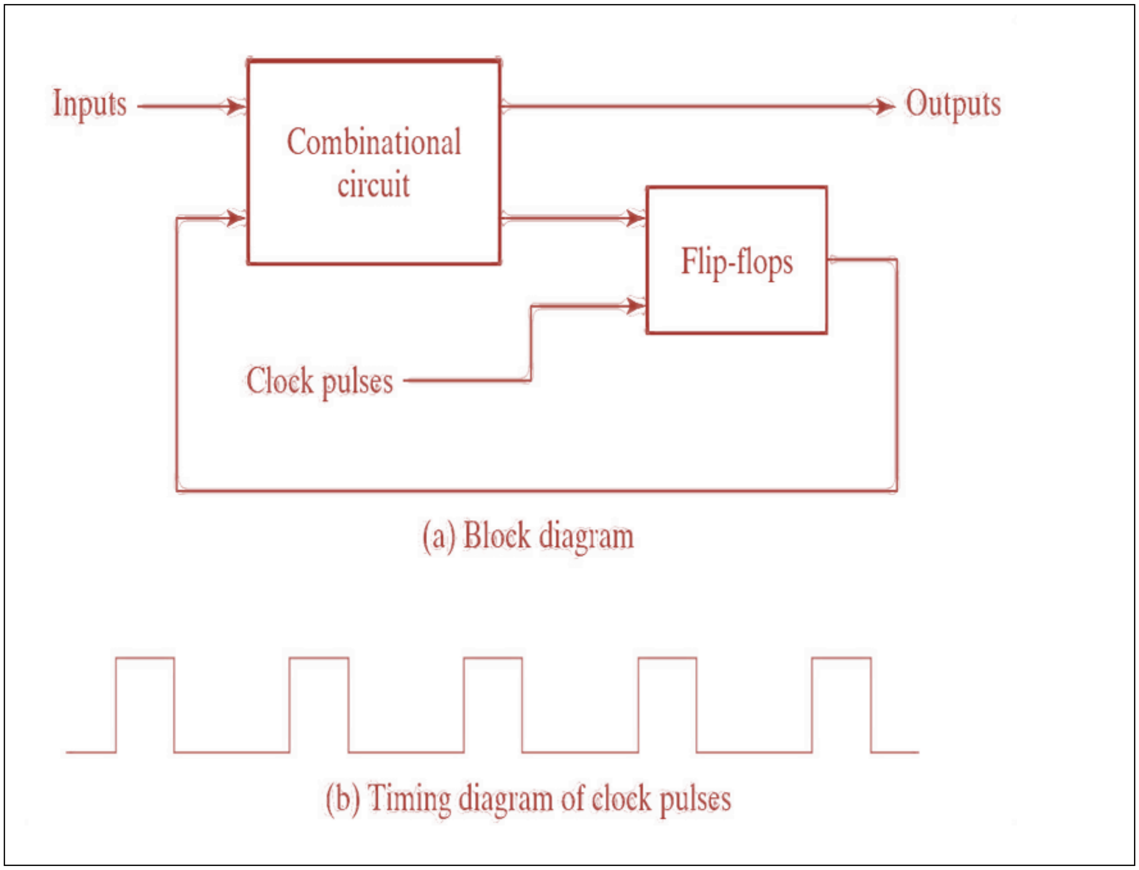
Storage element

- **Storage elements:** devices capable of storing binary information

  flip-flop: used in clocked sequential circuit, storing one bit of information: state

  The flip-flop is updated when a pulse of the clock signal occurs.

  Latch: basic component of flip-flop.



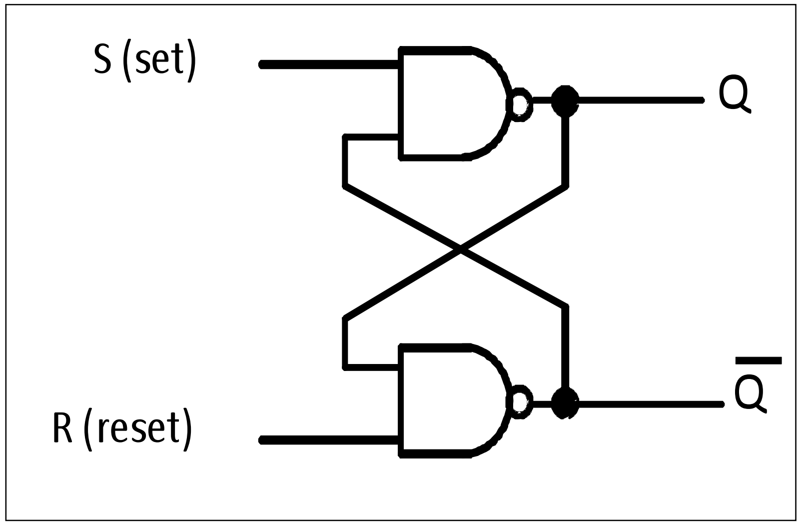
(a) Block diagram

(b) Timing diagram of clock pulses

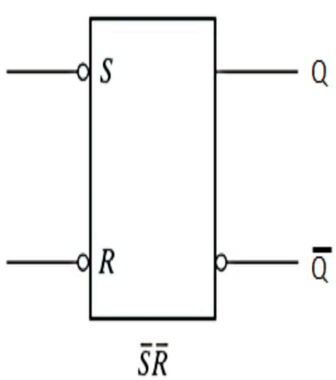
Basic (NAND) S – R (Set-Reset latch)Latch

- "Cross-Coupling" two NAND gates gives the S' -R' (Set – Reset) Latch:

  Which has the time sequence behavior:



S = 0, R = 0 is forbidden as input pattern

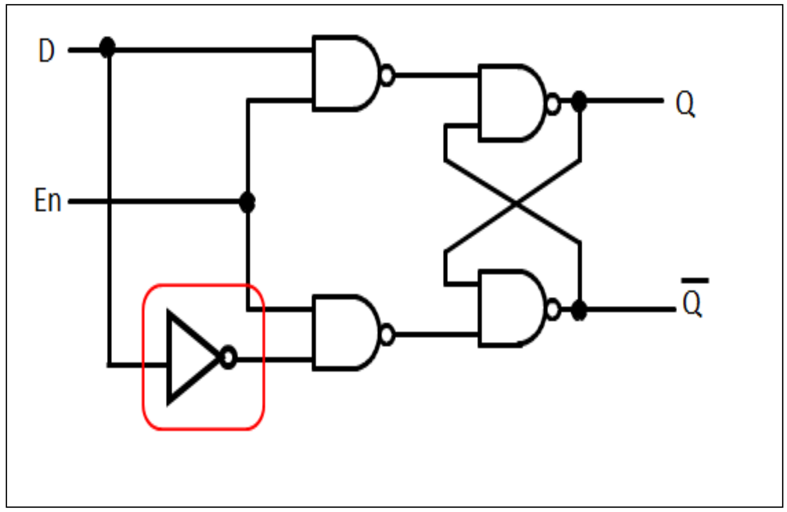


Sequence

R	S	Q	Q'	Comments
1	1	?	?	For our example, initial state unknown and we don't care
1	0	1	0	R and S are "active low": force Q to a 1 (and Q' to a 1)
1	1	1	0	No change: Keep Q as it is, a 1
0	1	0	1	Force Q to a 0 (and Q' to a 1)
1	1	0	1	No change: Keep Q as it is, a 0
0	0	1	1	Forbidden because both outputs go to 1

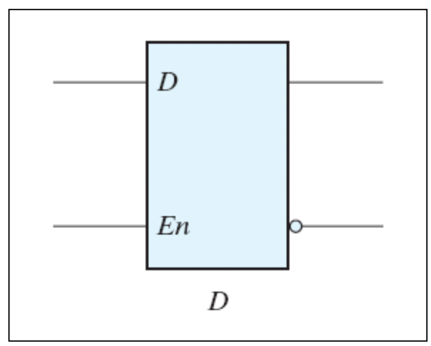
D Latch

- Adding an inverter to the S-R Latch, gives the D Latch



(shown with an enable input)

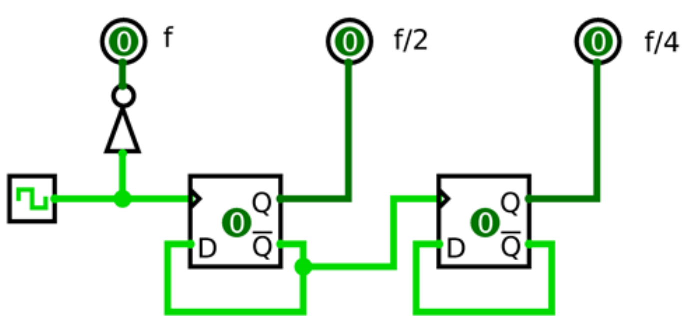
The graphic symbol for a D Latch is:



The enable input often acts as the clock.

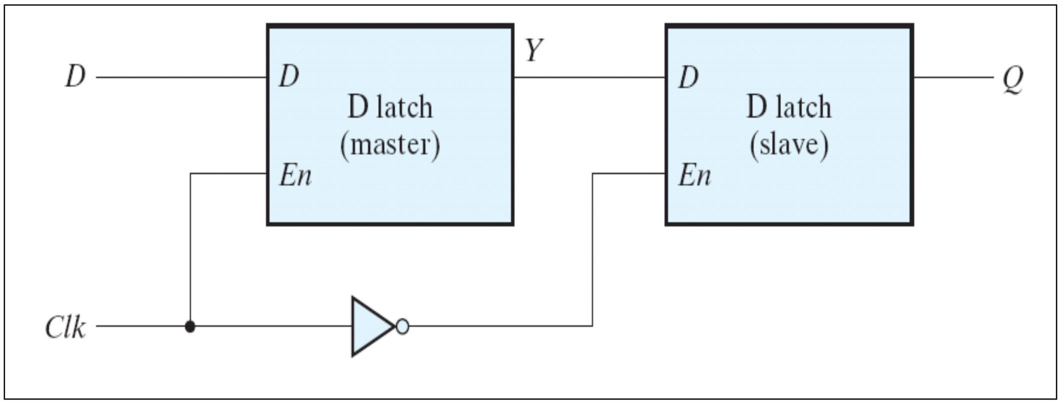
- Note that there are no "indeterminate" states!

En	*D*	Next state of *Q*
0	X	No change
1	0	*Q* = 0; reset state
1	1	*Q* = 1; set state

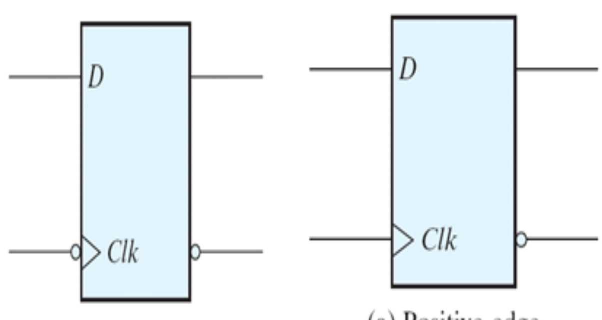


Edge-Triggered D Flip-Flop

- The edge-triggered D flip-flop is the master-slave D flip-flop



- The change of the D flip-flop output is associated with the negative (positive) edge at the end of the pulse



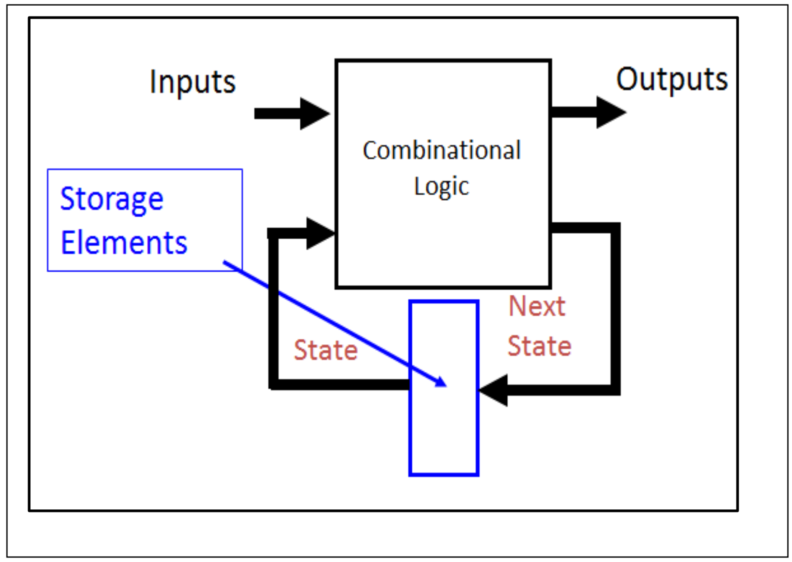
(a) Negative-edge (a) Positive-edge

D Flip-Flop

D	Q(t + 1)	
0	0	Reset
1	1	Set

Sequential Circuit Analysis

- General Model
  - Current State at time (t) is stored in flip-flops.



  - Next State at time (t+1) is a Boolean function of State and Inputs.
  - Outputs at time (t) are a Boolean function of State (t) and (sometimes) Inputs (t).

Moore and Mealy Models

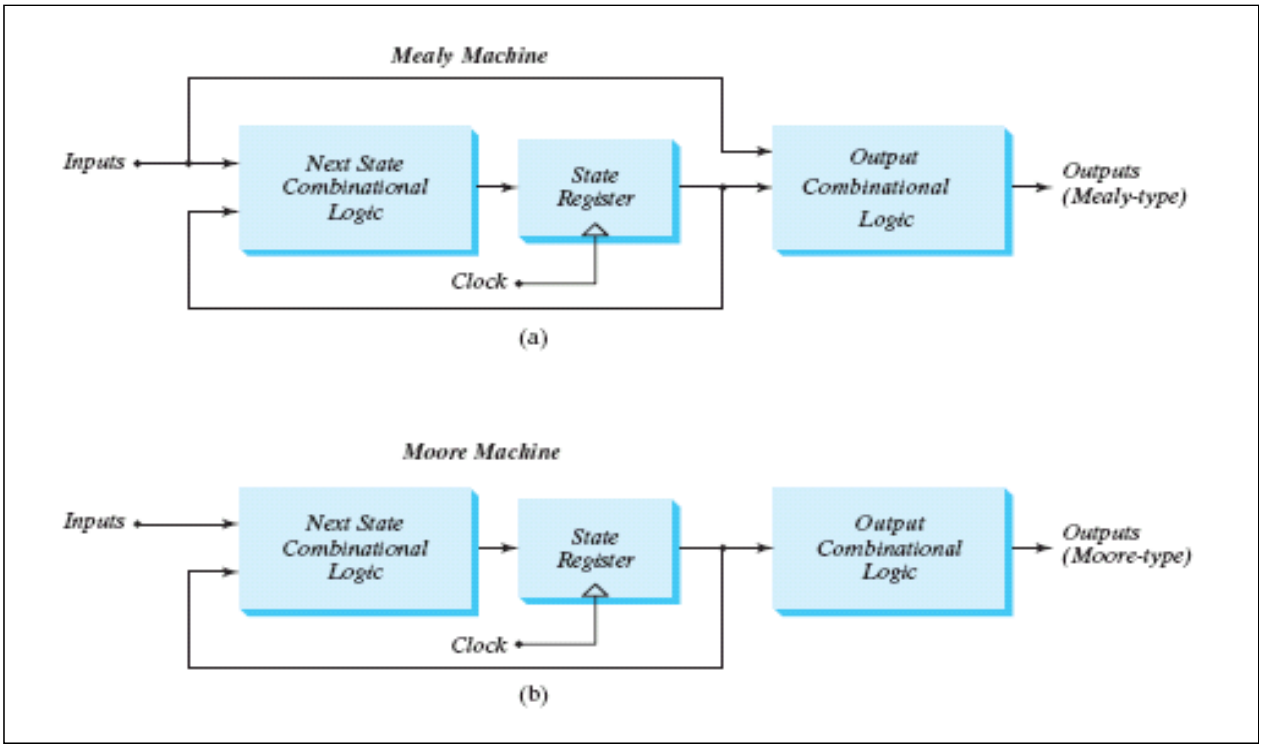
- Sequential Circuits or Sequential Machines are also called *Finite State Machines* (FSMs). Two formal models exist:

<u>Mealy Model</u>

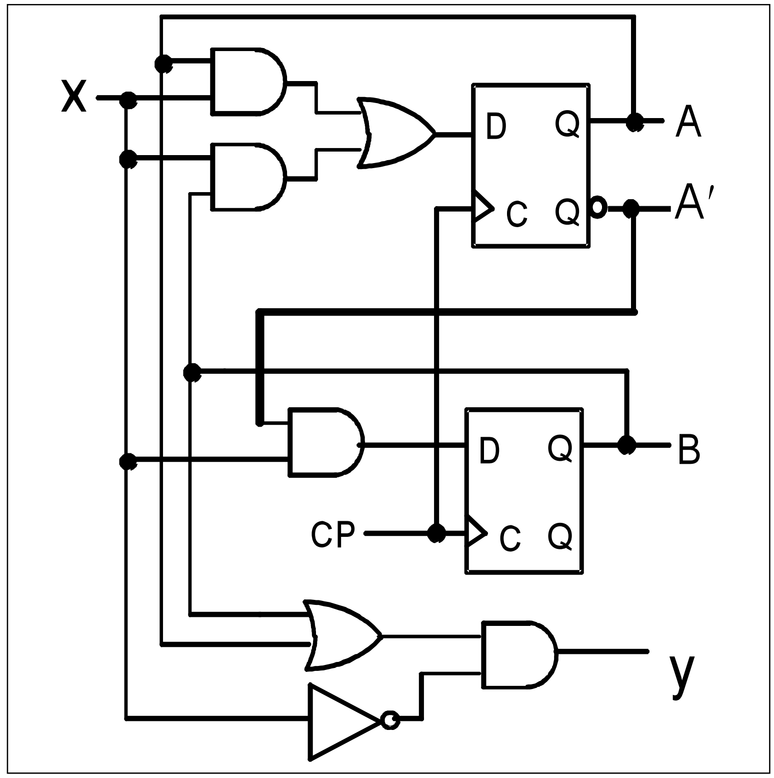
- Named after G. Mealy
- Outputs are a function of <u>inputs</u> AND <u>states</u>

<u>Moore Model</u>

- Named after E.F. Moore
- Outputs are a function ONLY of <u>states</u>



Example 1

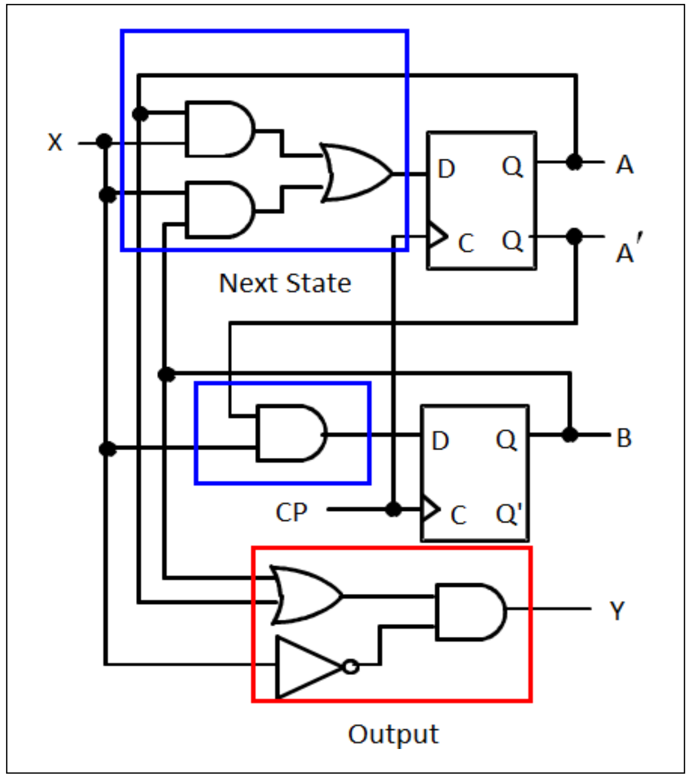


- Input: x
- Output: y
- State: (A, B)
- What is the Output Function?
- What is the Next State Function?
- Mealy or Moore?

Example 1 (continued)

Step 1:

- Input flip-flop equations:



$D_A = Ax + Bx$
$D_B = A'x$

Step 2:

- State equations:

$A(t+1) = D_A = Ax + Bx$
$B(t+1) = D_B = A'\,x$

Output equations:
$y = x'\,(B + A)$

Step 3:

- State Table and State Diagram

State Table Characteristics

- *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.
- Considered as a truth table:
    - the inputs are Input, Present State
    - and the outputs are Output, Next State

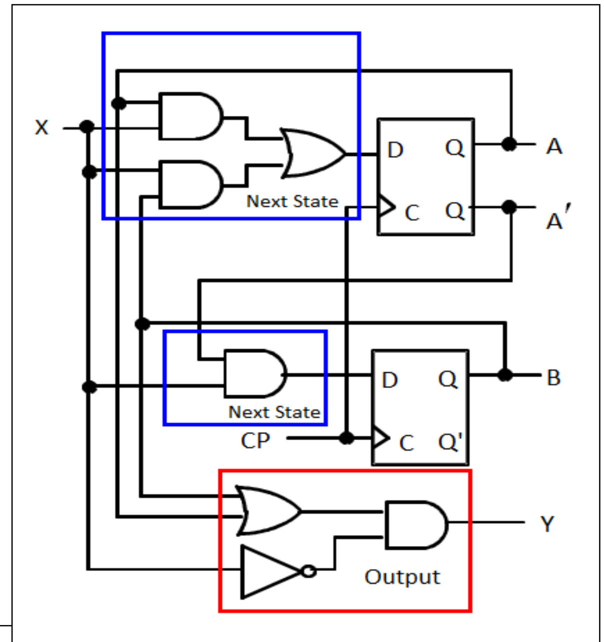
Example1: State Table.

- The state table can be filled in using state equations and output equations:

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

  $B(t+1) = D_B = A(t)'x(t)$

  $y(t) = x(t)'(B(t) + A(t))$



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

State Diagrams

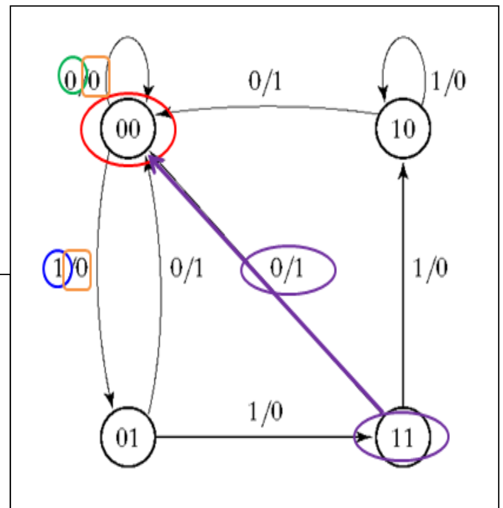
- The sequential circuit function can be represented in graphical form as a state diagram with the following components:
    - A circle with the state name in it for each state
    - A directed arc from the Present State to the Next State for each state transition
    - A label on each directed arc with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

Example 1: State Diagram

- On <u>directed arc</u> with the <u>output</u> included:

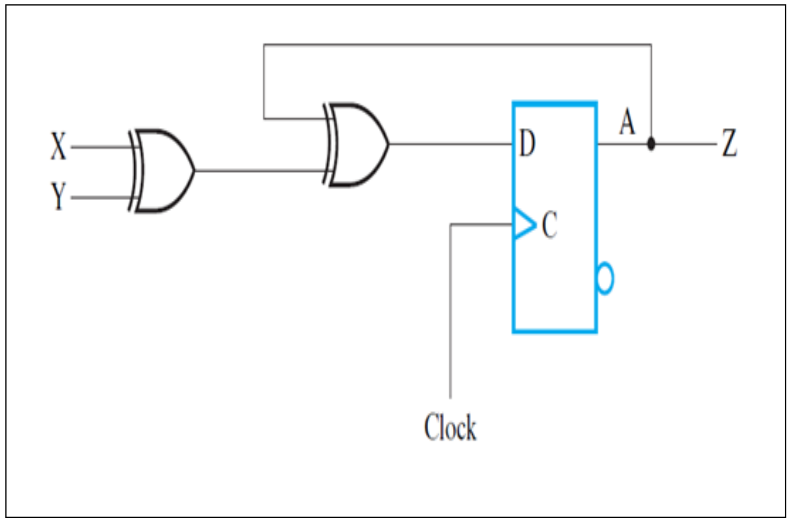
  - ***input/output***

  - ***Mealy type output depends on state and input***



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

Example 2

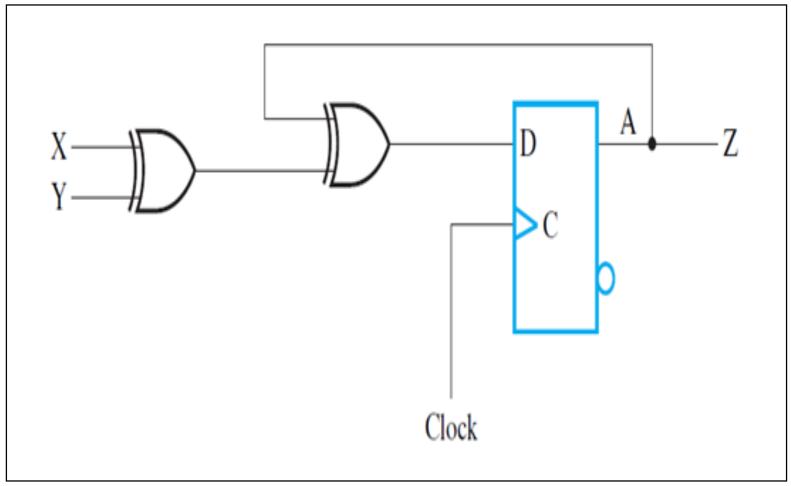


- Input: x, y
- Output: Z
- State: A
- What is the Output Function?

- What is the Next State Function?

- Mealy or Moore?

Example 2 (continued)



Step 1:

- Input flip-flop equations:
  $D_A = x \oplus y \oplus A$

Step 2:

- State equations:
  $A(t+1) = D_A = x \oplus y \oplus A$
  Output equations:
  $Z(t) = A(t)$

Step 3:

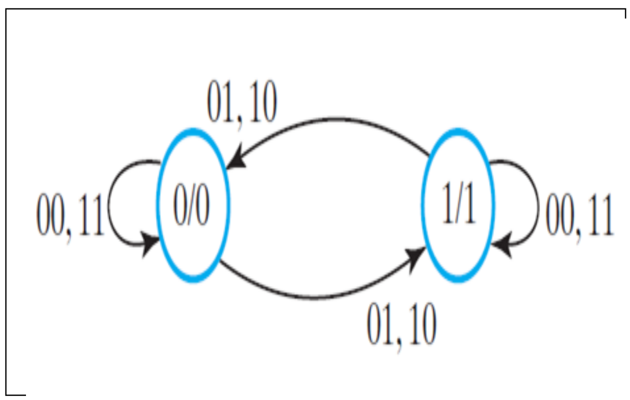
- State Table and State Diagram

Example2: State Table

- The state table can be filled in using state equations and output equation:

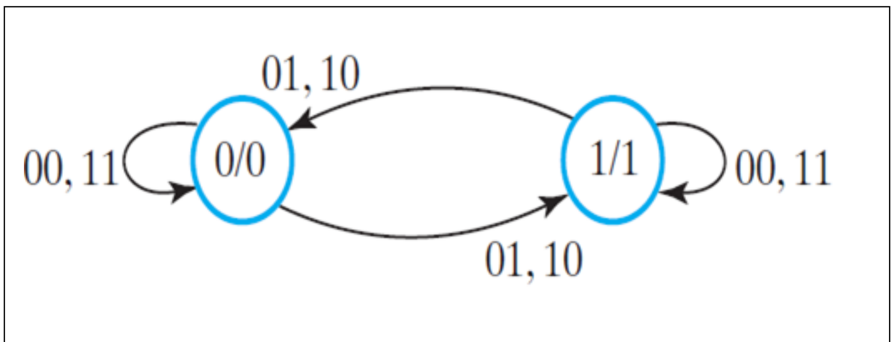
  $A(t+1) = D_A = x(t) \oplus y(t) \oplus A(t)$
  $Z(t) = A(t)$

Present state	Inputs		Next state	Output
A	X	Y	A	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



Example 2: State Diagram

- On circle with output included:
  - ***state/output***
  - ***Moore type output depends only on state***



Present state	Inputs		Next state	Output
A	X	Y	A	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