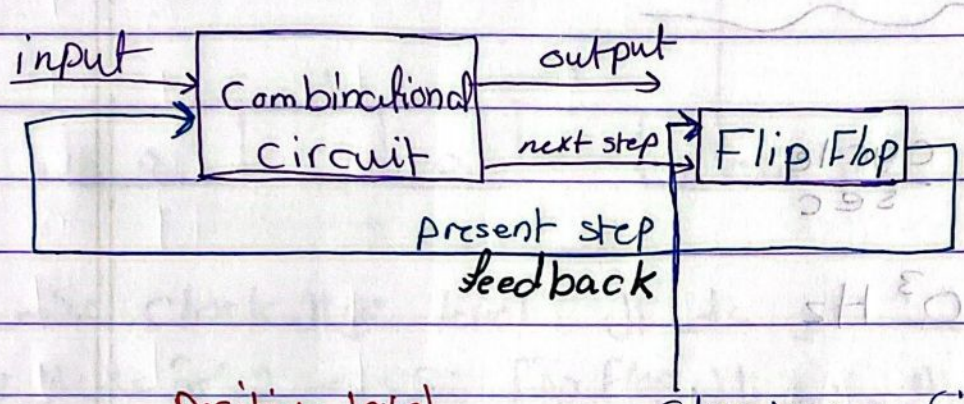
ENCS 2340 Summary Chapter 5

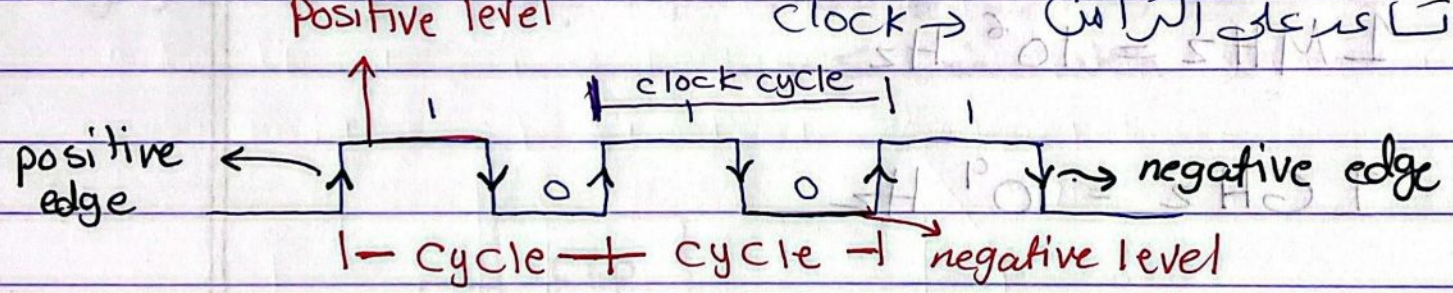
By : Malak Obaid

Sequential circuits :- previous outputs/ output تعتمد على

Sequential circuit

- Synchronous Sequential Circuit ↓ (متزامن)
- Asynchronous Sequential circuit (غير متزامن)





Time here on seconds ≡ clock cycle

$$\text{Frequence} = \frac{1}{\text{Time}} \equiv \frac{1}{s} \equiv Hz$$

كل Flip Flop بتخزن 1 bit وهي عبارة عن Synchronous memory

يعني ممكن يكونو عنا عدد كبير منها ولكنه بيكونوا مشبوكين على نفس الـ edge

Ex Given the clock cycle = 0.5 ns = 0.5×10^{-9} sec

Find the clock frequency = $\frac{1}{0.5 \times 10^{-9}} = 2 \times 10^{9}$ Hz = 2 GHz

$1\ ms = 10^{-3}\ sec$

$1\ \mu s = 10^{-6}\ sec$

$1\ ns = 10^{-9}\ sec$

$1\ ps = 10^{-12}\ sec$

$1\ Hz = 1\ \frac{cycle}{sec}$

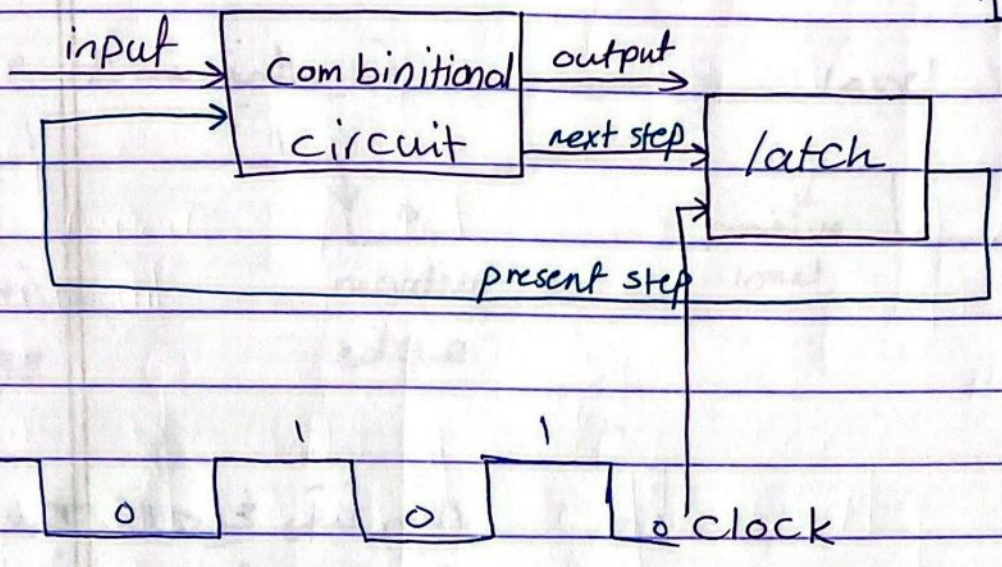
$1\ KHz = 10^{3}\ Hz$

$1\ MHz = 10^{6}\ Hz$

$1\ GHz = 10^{9}\ Hz$

Asynchonous Sequential circuit

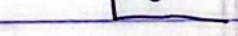
1 latch = 1 bit



ال latch بتشتغل على ال level في ال clock عكس ال Synchonous اللي فيها ال Flip Flop بتكون مشروطة على ال edge لان ال latch فترتها كبير

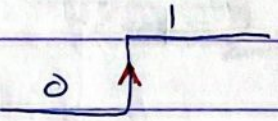
⇒ 2 bits

Note: لو الدارة انتقلت من واحد لصفر هيك



bubble نحط ع Flip Flop جنب ال clock دالها بس تغير قيمتها ويعتبرو سوا

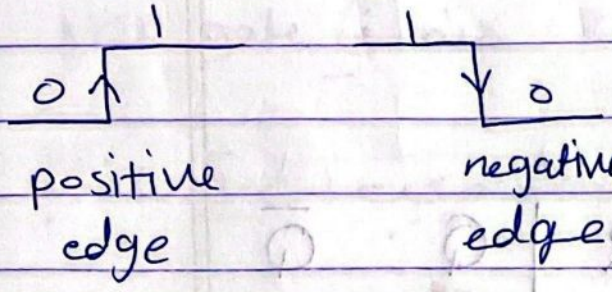
* اذا كانو مشبوكين هيك فرح تتغير قيمهم سوا بنفس الوقت يعني الدارة بتنتقل من صفر لواحد



يعني الثنين بصير 0,0 او 1,0 أو 0,1 أو 1,1 (مستحيل وحدة فقط تتغير)

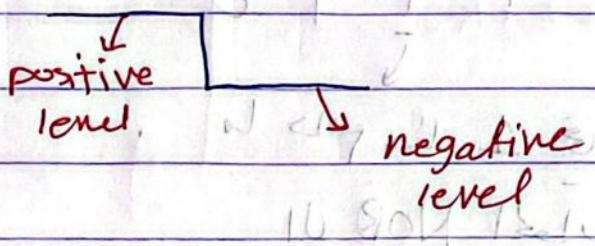
Flip Flop = 1 bit
(synchonorous)

بتغير الحالة عند الـ edge



latch = 1 bit
(asynchonous)

بتتغير عند الـ level



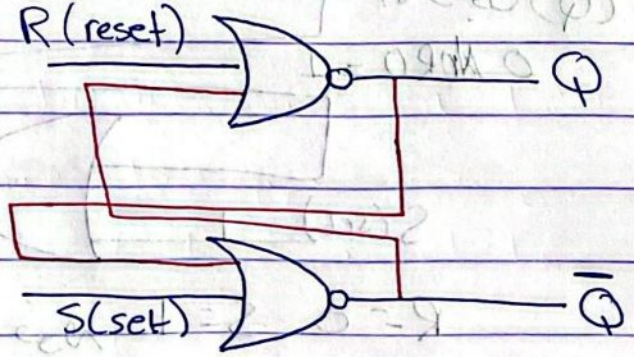
اللتش بتبقى في حالة تخزين للقيم
حتى تاخد cycle جديدة
(اي اشارة بتتغير من
+ edge اي
- edge اي

* Storage element (latch)

① SR Latch with nor gate = 1 bit

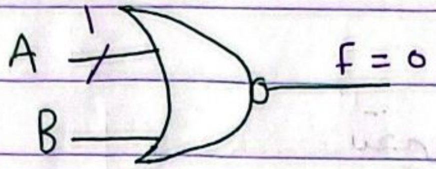
active (high)

- block diagram -



NOR

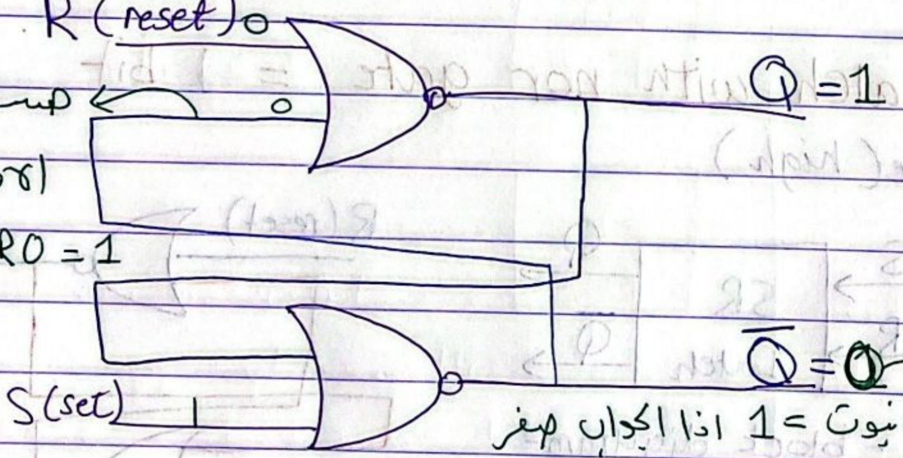
A	B	F
0	0	1
0	1	0
1	0	0
1	1	0



ما دام عنا 1 في
ال NOR الجواب
رح يكون صفر

S	R	Q	$\bar{Q}$	
1	0	1	0	
memory = 0	0	1	0	no change
0	1	0	1	
0	0	0	1	
invalid operation ← 1	1	0	0	forbiddin

function table بنسميه اي جدول يوصف ال latch



لانه مادام في NOR gate عنا انبوت = 1 اذا الجواب صفر

① في اول حالة حسب الجدول فوقه R = 0 S = 1
رح ينحفظ الاوتبوت وتضل للمرحلة اللي بعد اللي هي الحالة اللي بعد
R = 0 / S = 0 ⇐ هنا لا يمكننا الحكم لمندما واحد
من الانبوت = 0 فالانبوت الثاني رح يكون اوتبوت العملية التي
[illegible] قبلها هيك يجيب لدينا اوتبوت [illegible]

* حسنا عند input S=0 R=0 هنا تكون عنا زي حالة (memory) no change

اي ان الاوتبوت يحل نفسه .

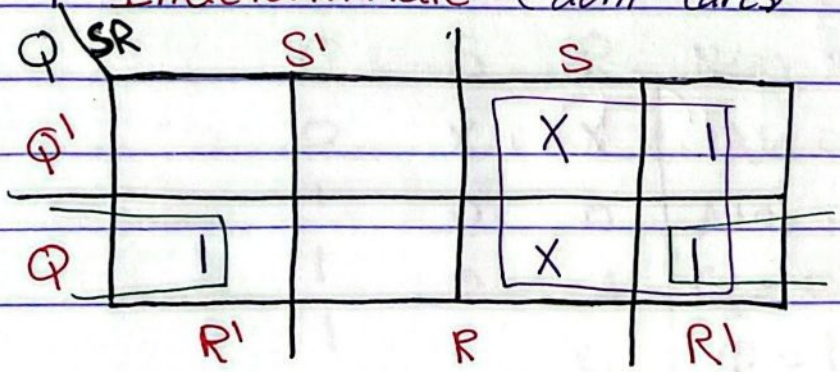
* لم نبدأ اول حالة بان S=0, R=0 لانه في NOR gate

لا عكنا الحكم من الصفر عند الاوتبوت ويجب ان يكون لدينا

متى مخزنة مسبقاً حتى نقدر ان نكمل (Unknown State)

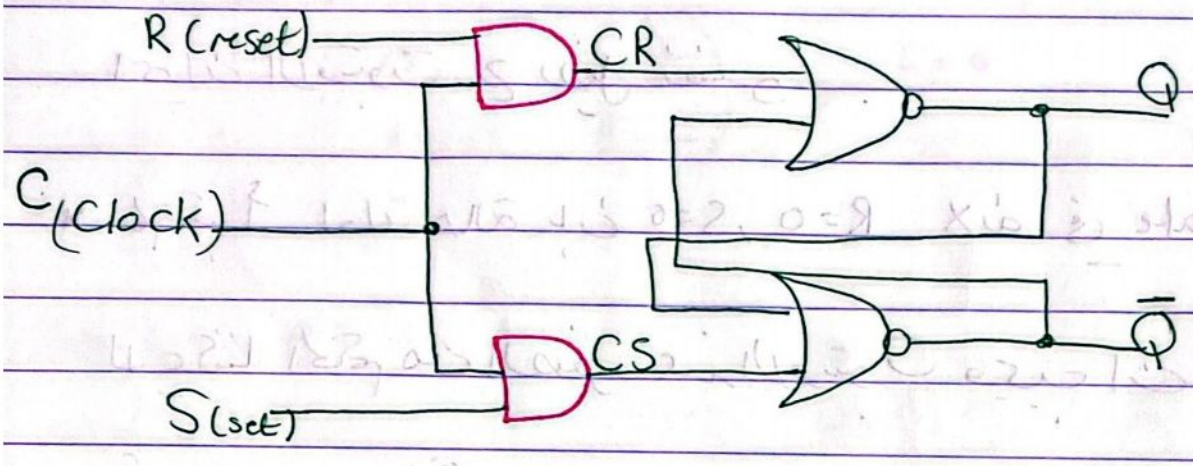
Characteristic Equation of the SR Latch

Q(t)	S	R	Q(t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	Indeterminate (dont Cares)
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	Indeterminate (dont cares)



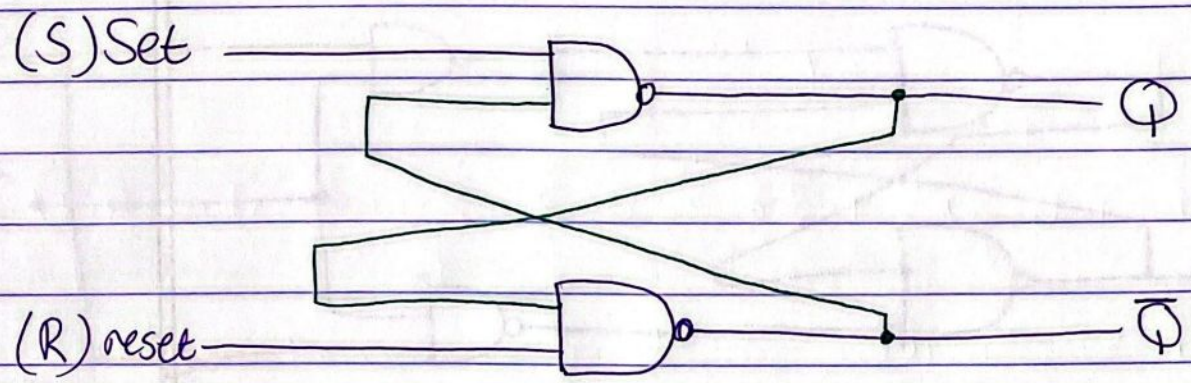
$Q(t+1) = S + R'Q$

Gated SR Latch with Clock Enable



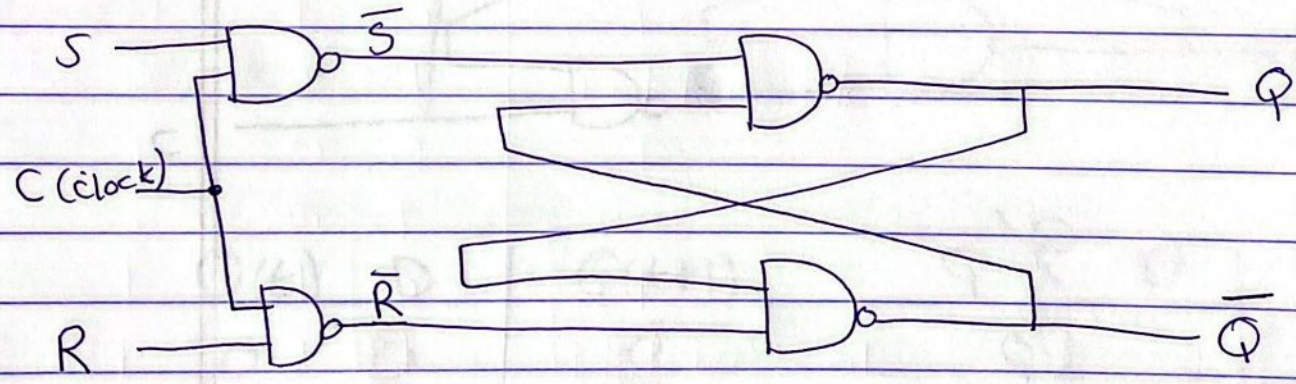
C	S	R	Next State of Q
0	X	X	No Change
1	0	0	No Change
1	0	1	Q=0 Reset State
1	1	0	Q=1 Set State
1	1	1	Undefined

$\overline{S}\overline{R}$ Latch with NAND gates. (active low)



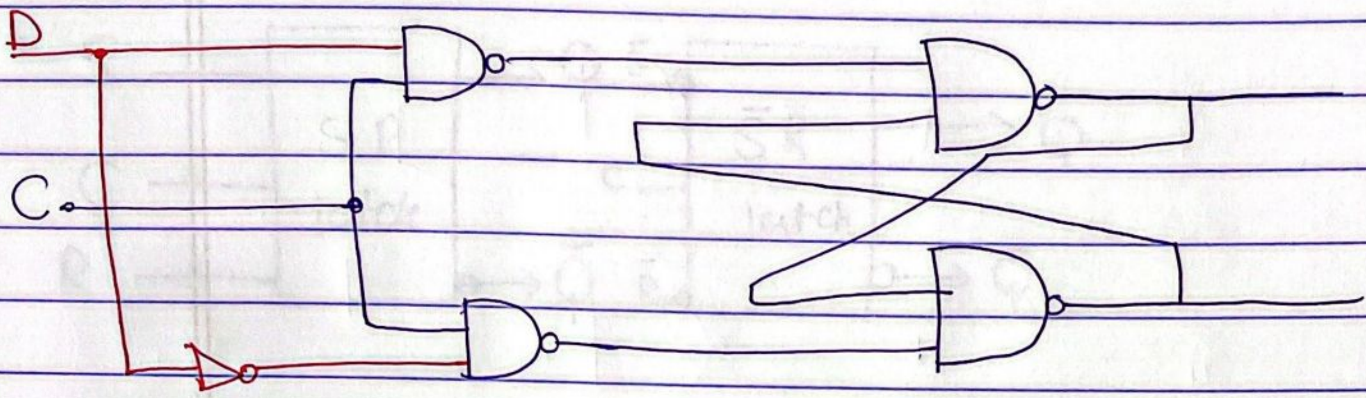
S	R	Q	$\overline{Q}$	
0	1	1	0	Set State
1	1	1	0	
1	0	0	1	
1	1	0	1	Reset State
0	0	1	1	Undefined

Gated SR Latch with Clock Enable



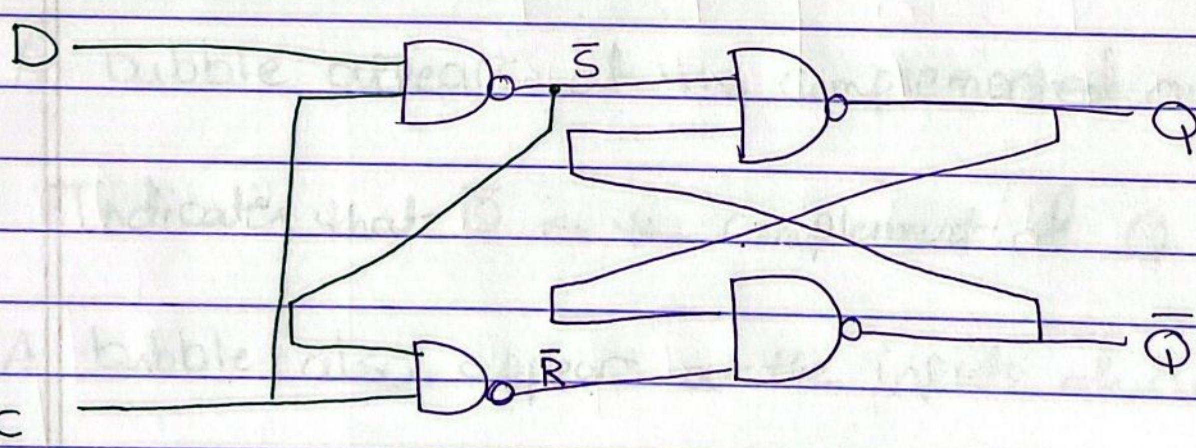
C	S	R	Next state of Q
0	X	X	No change
1	0	0	No change
1	0	1	Q = 0 Reset
1	1	0	Q = 1 Set
1	1	1	Undefined

D-Latch with Clock Enable (حتى نمنع S , R يكونوا سوا 1)



C	D	Next State of Q
0	X	No change
1	0	Q = 0 Reset
1	1	Q = 1 Set State

نفس الحركة



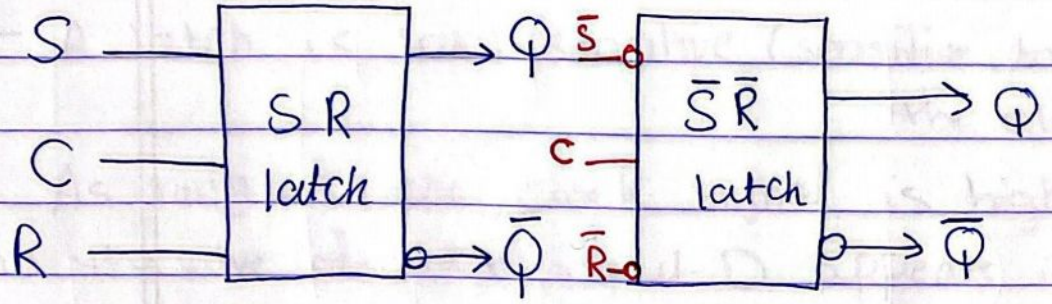
Q(t)	D	Q(t+1)
0	0	0
0	1	1
1	0	0
1	1	1

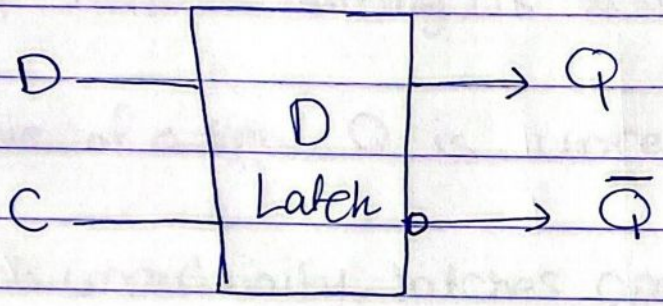
⇒

Q \ D	D'	D
Q'		1
Q		1

$Q(t+1) = D$

Graphic Symbols for latches





* A bubble appears at the complemented output

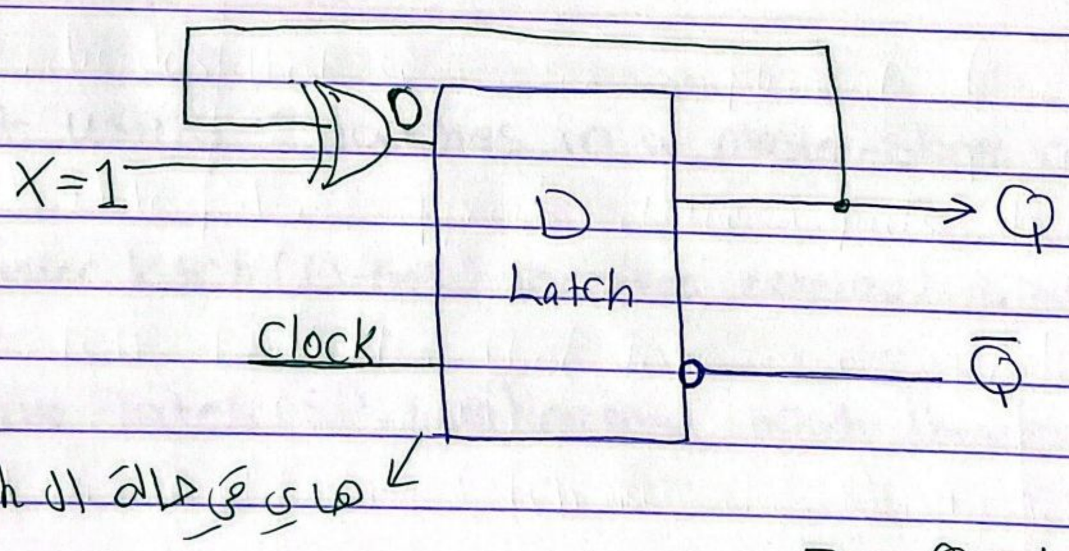
$\bar{Q}$ Indicates that $\bar{Q}$ is the complement of Q

* A bubble also appears at the inputs of an $\bar{S}\bar{R}$ Latch Indicates that logic-0 is used (not 1) to set or Reset the latch (as in the NAND latch Implementation)

Problem with latches

- A latch is level sensitive (sensitive to the level of the clock)
- As long as the clock signal is high, Any change in the value of ~~output~~ input D appears in the output Q
- Output Q keeps changing its value during a clock cycle
- Final value of output Q is uncertain

due to this uncertainity, latches are Not used as memory elements in synchronous circuits



$$D = Q \oplus 1$$
$$= \overline{Q}$$

Flip-Flops

- is a better memory element for synchronous circuits

- Solves the problem of latches in synchronous sequential circuits.

- A latch is sensitive to the level of the clock but Flip-Flop is sensitive to the edge of the clock

- A Flip-Flop is also called edge-triggered memory element

- It changes it output value at the edge of the clock

① D Flip-Flop

- Built using 2 latches in a master-slave configuration

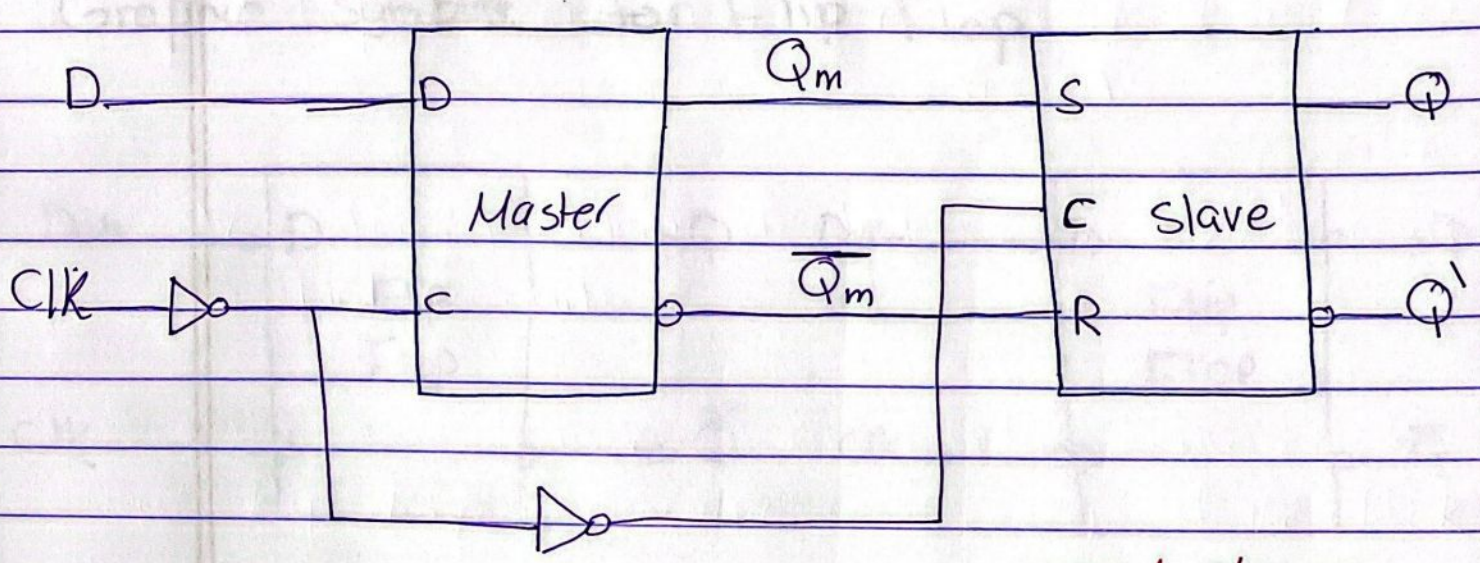
- A master latch (D-type) receives external inputs

- A slave latch (SR-type) receives inputs from the master latch

- only one latch is enabled at any given time

when CLK = 0 the master is enabled & the D input latched (slave disabled)

when CLK = 1 the slave is enabled to generate the output (master is disabled)

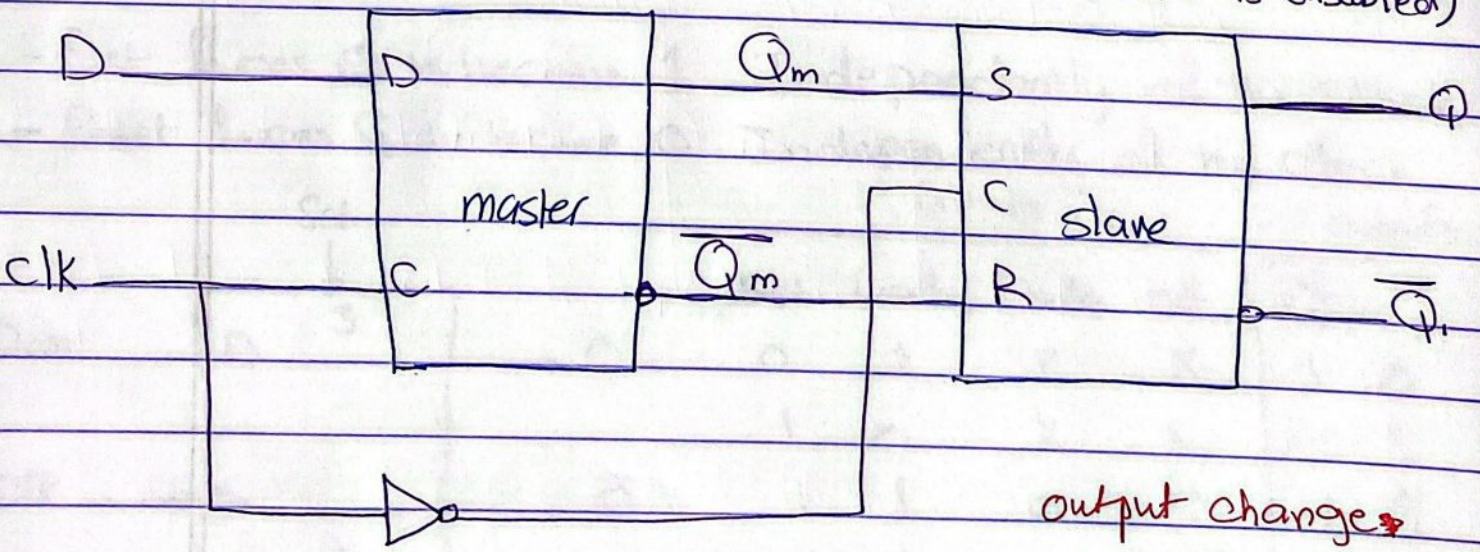


output change when clk changes from 0 to 1

Negative Edge-Triggered D Flip Flop

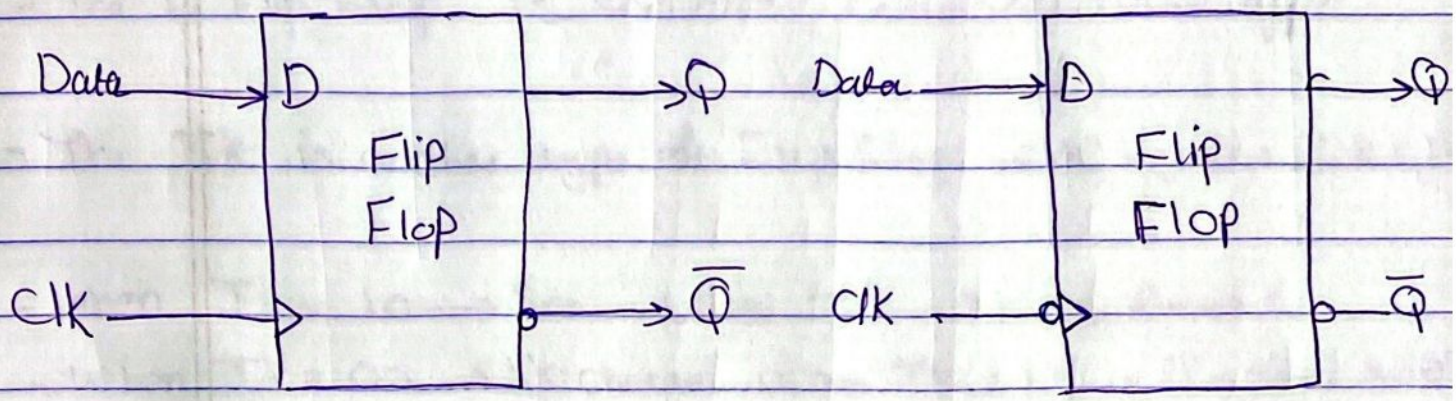
when clk = 1 the master is enabled and the D input is latched (slave disabled)

when clk = 0 the slave is enabled to generate the outputs (master is disabled)



output change when clk changes from 1 to 0

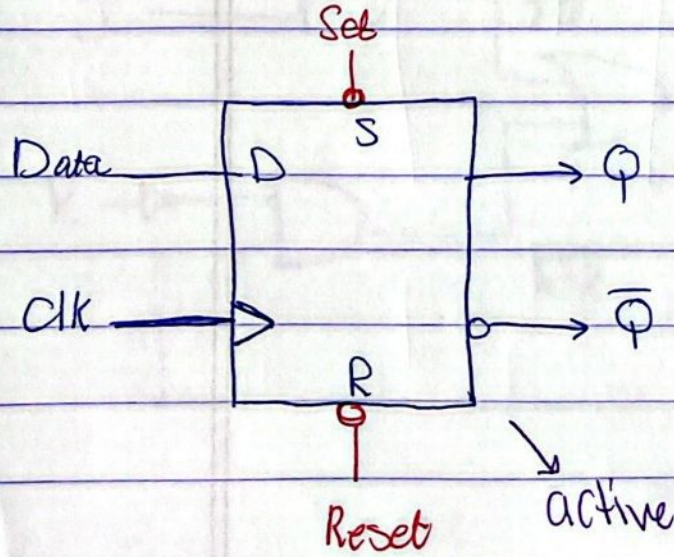
Graphic symbols for Flip Flop



positive edge negative edge

D-FF with Asynchronous Set & Reset

- when Flip Flops are powered their initial state is Unknown
- Some Flip-Flops have an Asynchronous Set and/or Reset input
- Set forces Q to become 1 Independently of the clock
- Reset forces Q to become 0 Independently of the clock

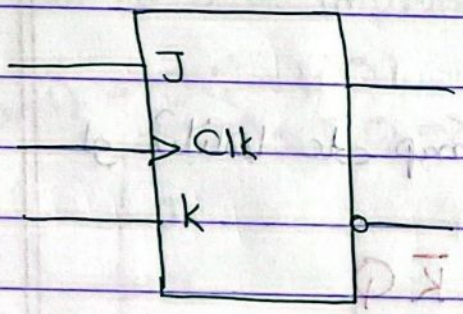


inputs				outputs	
Set	Reset	Data	clk	Q	$\overline{Q}$
0	1	X	X	1	0
1	0	X	X	0	1
1	1	0	↑	0	1
1	1	1	↑	1	0

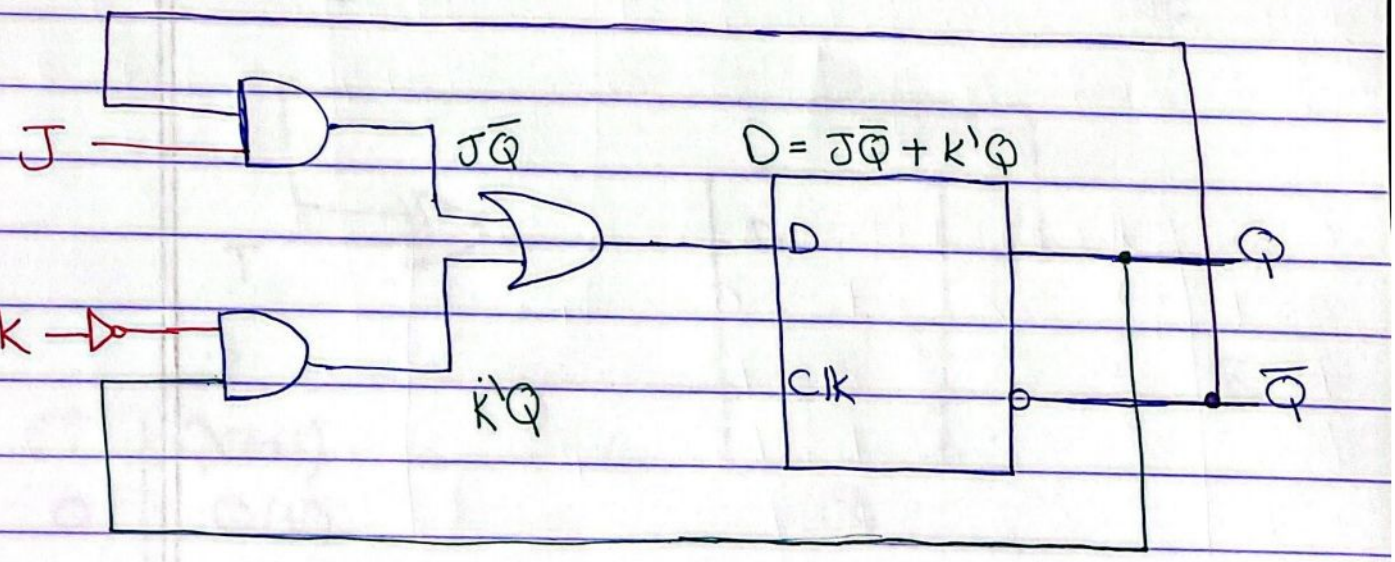
rising edge

JK Flip Flop

- The D Flip Flop is the most commonly used type
- The JK is another type of Flip Flop with inputs: J, K, Clk
- When JK = 10 → Set, when JK = 01 → Reset
- When JK = 00 → No charge, when JK = 11 → Inverted output
- JK can be implemented using D FF and gates.



J	K	$Q_{(t+1)}$
0	0	$Q_{(t)}$
0	1	0
1	0	1
1	1	$\bar{Q}_{(t)}$



Characteristic Equation of the JK Flip Flop

Q(t)	J	K	Q(t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

لو مثلناها على k-map رح ينتج لدينا ان

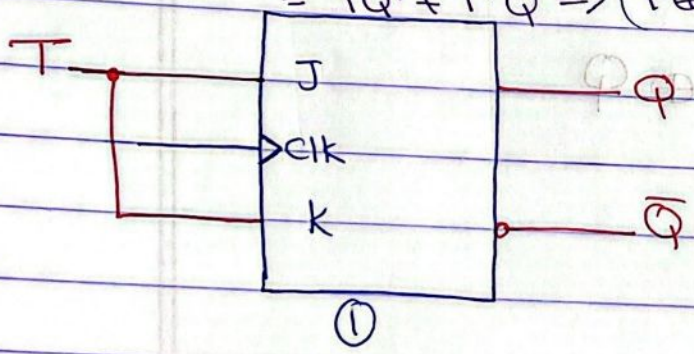
$$Q(t+1) = J\overline{Q} + \overline{K}Q$$

T Flip Flop (Toggle):

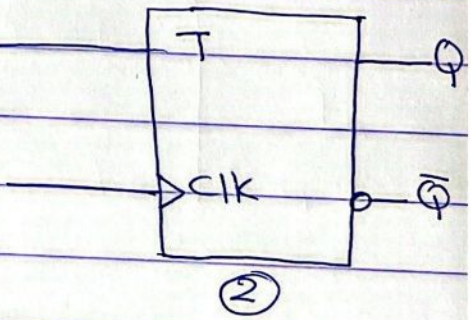
- has T and clk inputs
- When T = 0 → No change
- When T = 1 → Invert outputs
- T Flip Flop can be implemented using JK FF
- it can be also implemented using D-Flip Flop & XOR gate

$Q(t+1) = JQ' + K'Q$

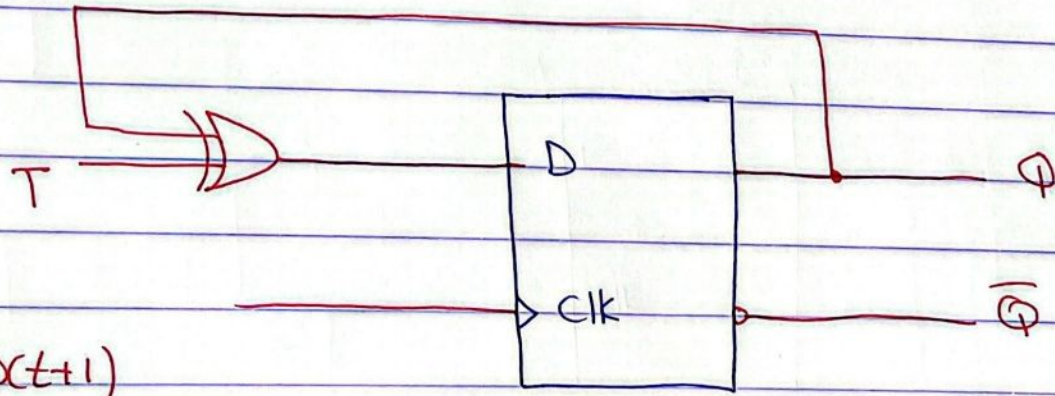
$= TQ' + T'Q \Rightarrow (T \oplus Q)$ → هاي فرع الدارة التي تحت



①



②



③

T	Q(t+1)
0	Q(t)
1	$\overline{Q}(t)$

Characterisitic Equation of T-Flip Flop

Q(t)	T	Q(t+1)	
0	0	0	→ T نفس القيم
0	1	1	
1	0	1	→ T عكس القيم
1	1	0	

بعد عمل k-map له نستنتج أن

$$Q(t+1) = TQ' + T'Q$$

$$= T \oplus Q$$

Analysis of Clocked Sequential Circuits

Analysis is describing what a given circuit will do

The output of a clocked sequential circuit is determined by

1. Inputs
2. State of the Flip-Flops

Analysis Procedure:

1. Obtain the equations at the inputs of the Flip-Flops
2. Obtain the next state and the output equations
3. Fill the state table for all possible input and state values
4. Draw the state diagram

Example 1 :-

- Is this a clocked sequential circuit?

  YES!

- What type of Memory?

  D Flip-Flops

- How many state variables?

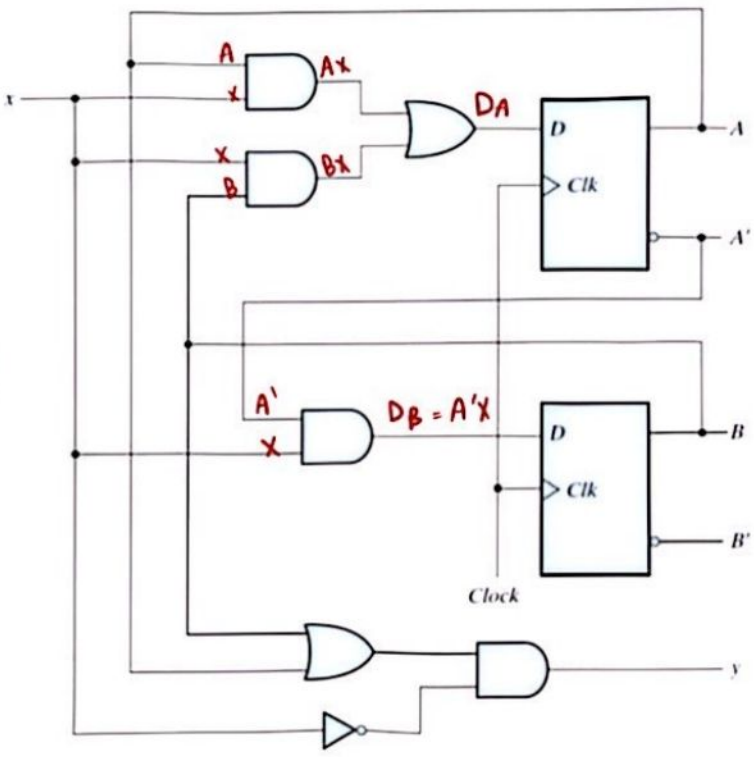
  Two state variables: A and B

- What are the Inputs?

  One Input: x

- What are the Outputs?

  One Output: y



❖ What are the equations on the D inputs of the flip-flops?

$D_A = A\,x + B\,x$

$D_B = A'\,x$

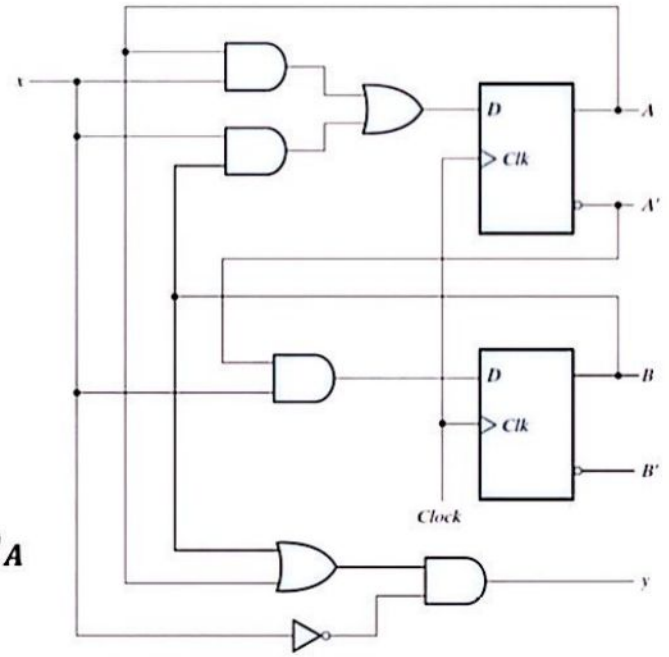
❖ A and B are the **current state**

$A(t) = A, \quad B(t) = B$

❖ D_A and D_B are the **next state**

$A(t+1) = D_A, \quad B(t+1) = D_B$

❖ The values of A and B will be D_A and D_B at the next clock edge



❖ The next state equations define the **next state**

At the **inputs** of the Flip-Flops

❖ Next state equations?

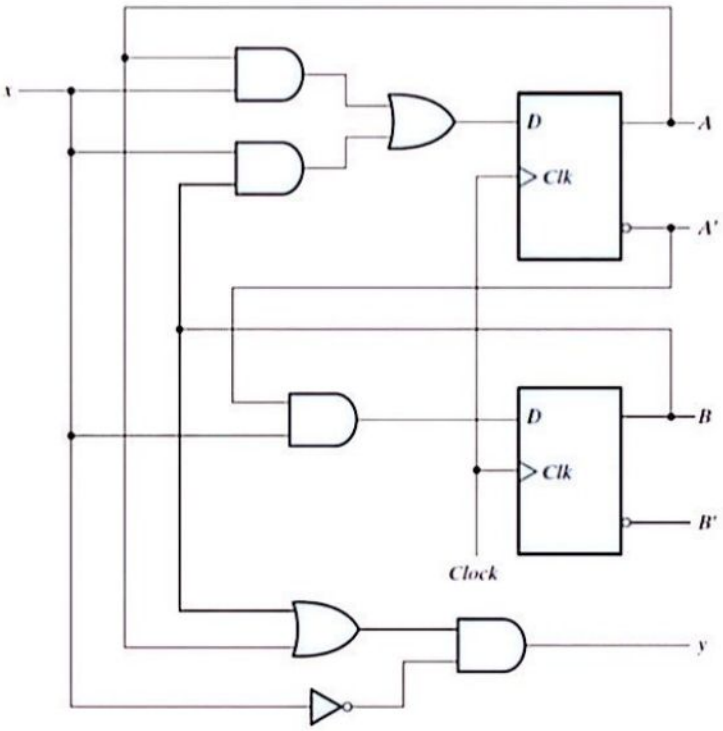
$A(t+1) = D_A = A\,x + B\,x$

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

❖ There is only one output y

❖ What is the output equation?

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



- State table shows the Next State and Output in a tabular form
- Next State Equations: $A(t+1) = A\,x + B\,x$ and $B(t+1) = A'\,x$
- Output Equation: $y = (A + B)\,x'$

Present State		Input	Next State		Output
A	B	x	A	B	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

$A.x + B.x$
$= 1.1 + 1.1$
$1 + 1 = 1$
1 يكون

$(A+B)x'$
ونعوض

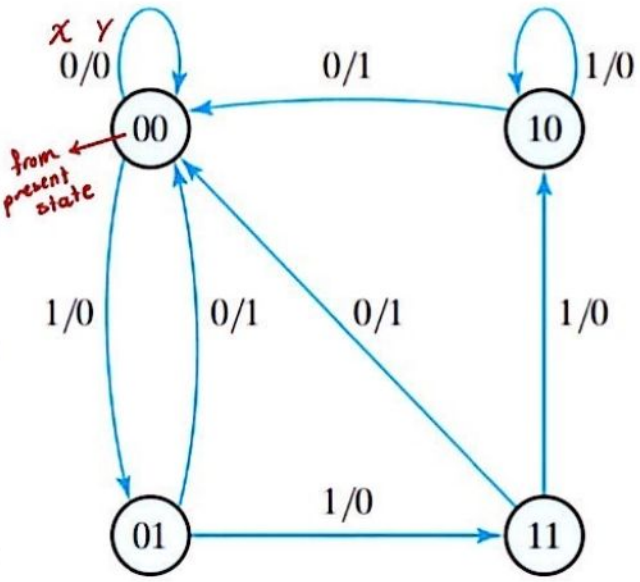
Another form of the state table

Present State		Next State				Output	
		x = 0		x = 1		x = 0	x = 1
A	B	A	B	A	B	y	y
0	0	0	0	0	1	0	0
0	1	0	0	1	1	1	0
1	0	0	0	1	0	1	0
1	1	0	0	1	0	1	0

- State diagram is a graphical representation of a state table
- The circles are the states
- Two state variable ➔ Four states (ALL values of A and B)
- Arcs are the state transitions

  Labeled with: Input x / Output y

Present State		Next State				Output	
		x = 0		x = 1		x = 0	x = 1
A	B	A	B	A	B	y	y
0	0	0	0	0	1	0	0
0	1	0	0	1	1	1	0
1	0	0	0	1	0	1	0
1	1	0	0	1	0	1	0



Combinational versus Sequential Analysis

Analysis of Combinational Circuits

- Obtain the Boolean Equations
- Fill the Truth Table

Output is a function of input only

Analysis of Sequential Circuits

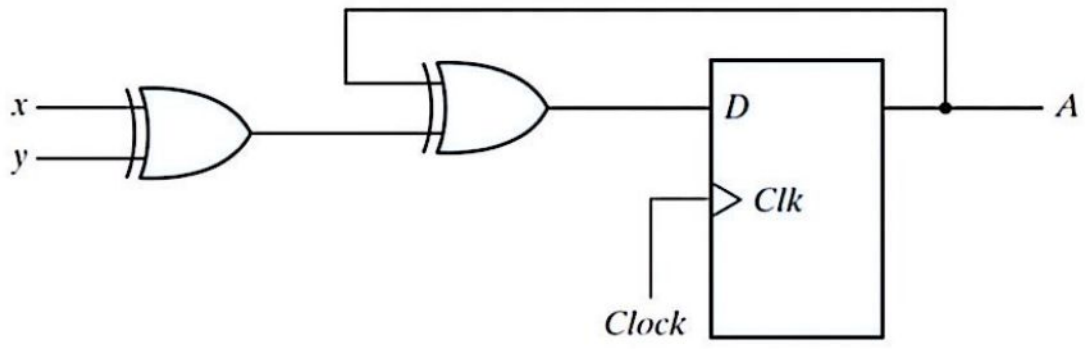
- Obtain the Next State Equations
- Obtain the Output Equations
- Fill the State Table
- Draw the State Diagram

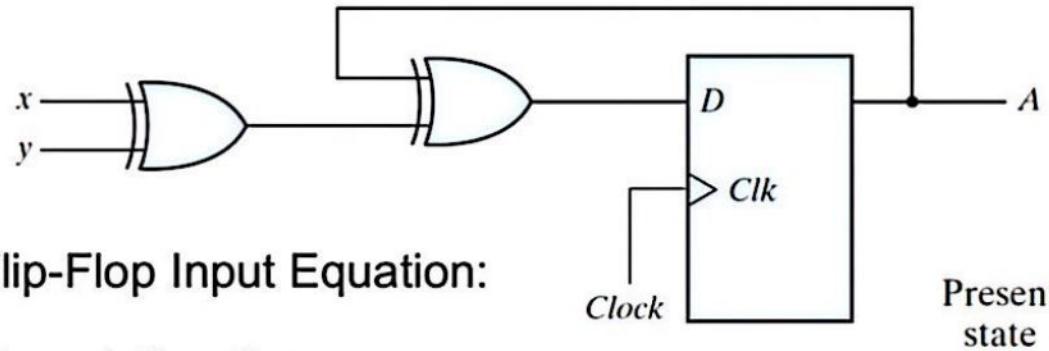
Next state is a function of input and current state

Output is a function of input and current state

Example with Output = Current State

- Analyze the sequential circuit shown below
- Two inputs: x and y
- One state variable A
- No separate output ➔ Output = current state A
- Obtain the next state equation, state table, and state diagram



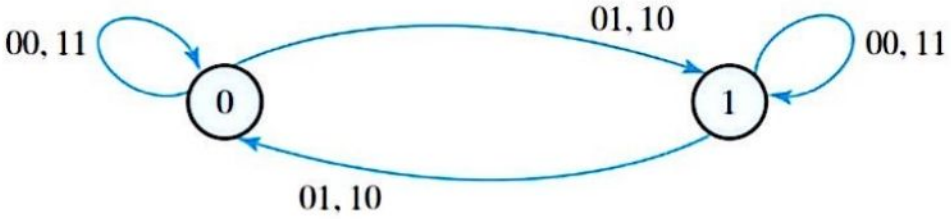


❖ Flip-Flop Input Equation:

$D_A = A \oplus x \oplus y$

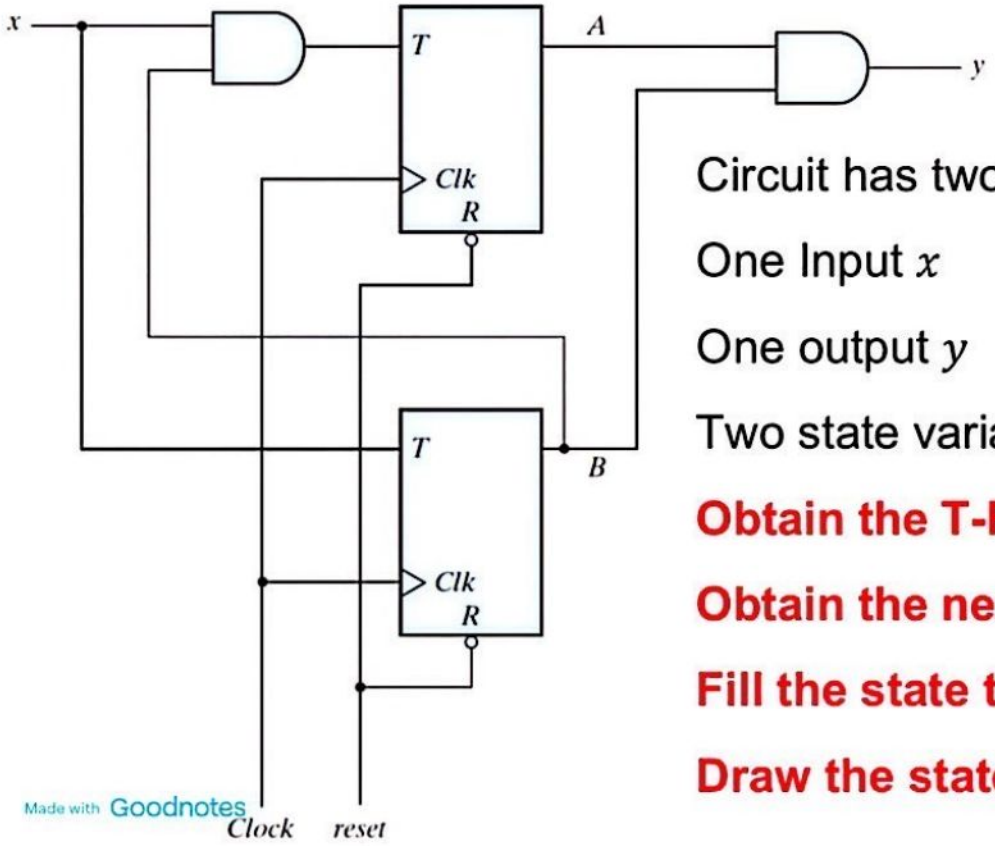
❖ Next State Equation: $A(t+1) = A \oplus x \oplus y$

Present state	Inputs		Next state
A	x	y	A
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1



Example 3:

Sequential Circuit with T Flip-Flops



Circuit has two T Flip-Flops

One Input x

One output y

Two state variables: A and B

Obtain the T-FF input equations

Obtain the next state equations

Fill the state table

Draw the state diagram

❖ For D Flip-Flop: $Q(t+1) = D$

❖ For T Flip-Flop: $Q(t+1) = T \oplus Q(t)$

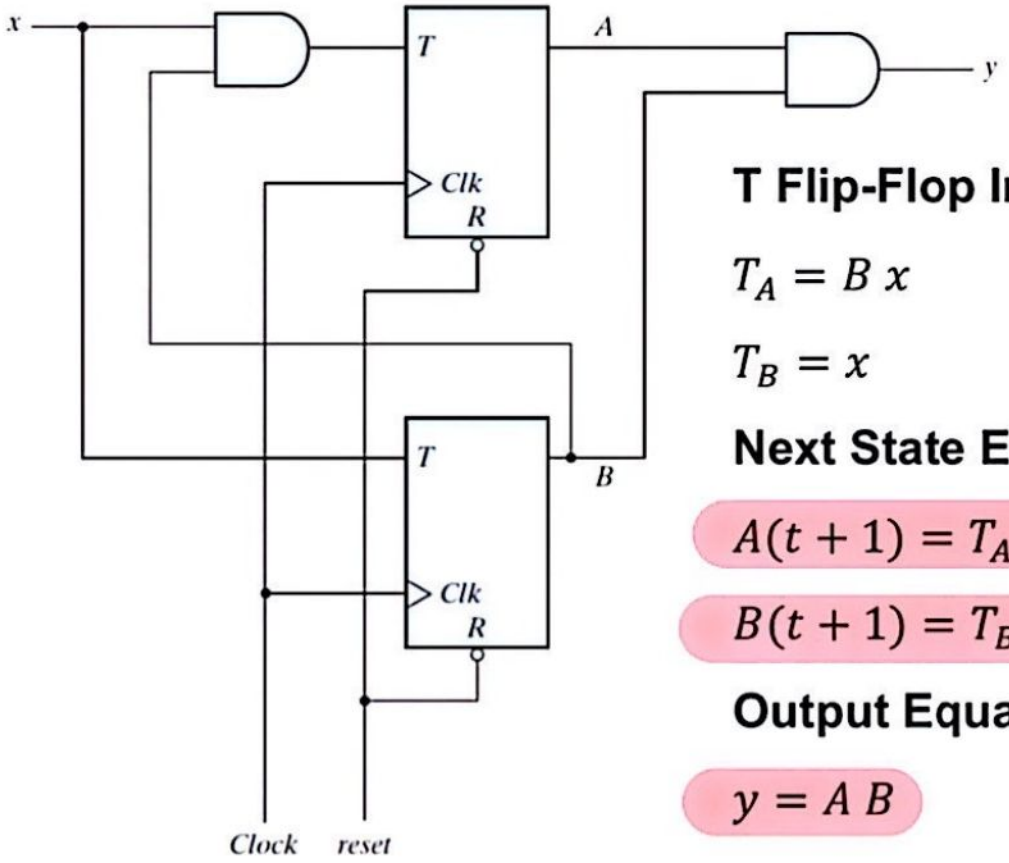
These equations define the Next State

❖ For JK Flip-Flop: $Q(t+1) = J\,Q'(t) + K'\,Q(t)$

D Flip-Flop		
D	Q(t+1)	
0	0	Reset
1	1	Set

T Flip-Flop		
T	Q(t+1)	
0	Q(t)	No change
1	Q'(t)	Complement

JK Flip-Flop		
J K	Q(t+1)	
0 0	Q(t)	No change
0 1	0	Reset
1 0	1	Set
1 1	Q'(t)	Complement



T Flip-Flop Input Equations:

$T_A = B\,x$

$T_B = x$

Next State Equations:

$A(t+1) = T_A \oplus A = (B\,x) \oplus A$

$B(t+1) = T_B \oplus B = x \oplus B$

Output Equation:

$y = A\,B$

T Flip-Flop Input Equations:

$T_A = B\,x$

$T_B = x$

Next State Equations:

$A(t+1) = (B\,x) \oplus A$

$B(t+1) = x \oplus B$

Output Equation:

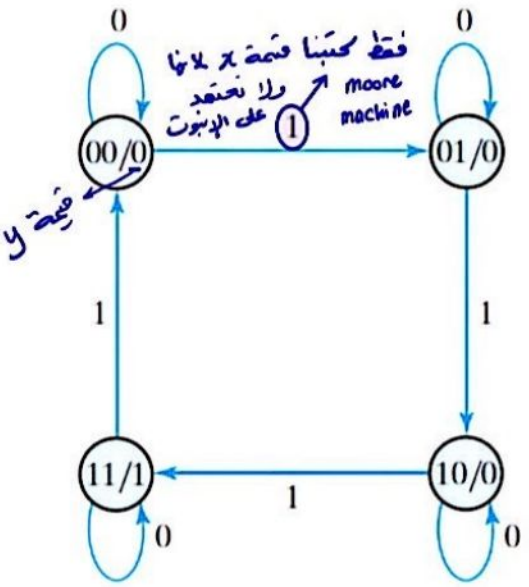
$y = A\,B$

State Table

Present State		Input	Next State		Output
A	B	x	A	B	y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	1	0	0
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	1

Notice that the output is a function of the present state only.
It does **NOT** depend on the input x

Present State		Input	Next State		Output
A	B	x	A	B	y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	1	0	0
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	1



- Four States: $AB = 00,\ 01,\ 10,\ 11$ (drawn as circles)
- Output Equation: $y = A\,B$ (does not depend on input x)
- Output y is shown inside the state circle (AB/y)

Sequential Circuit with a JK Flip-Flops

One Input x and two state variables: A and B (outputs of Flip-Flops)

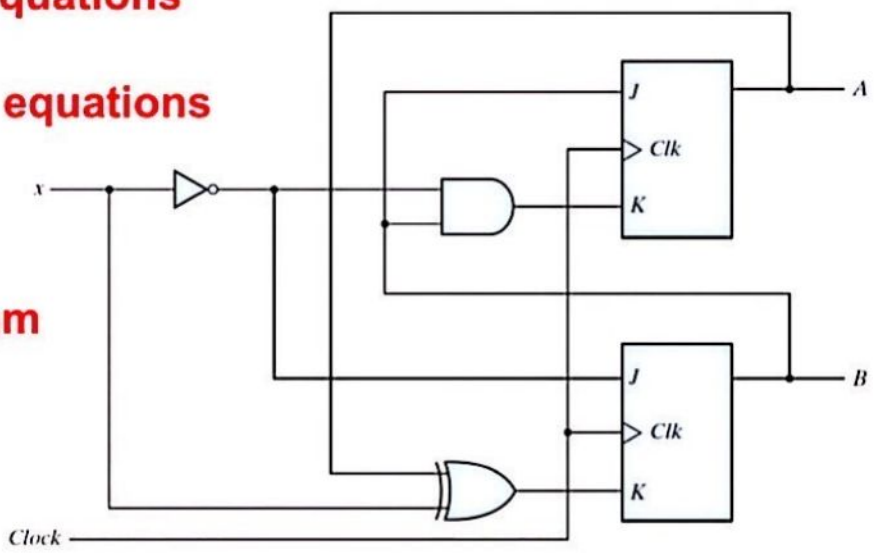
No separate output ➔ Output = Current state $A\ B$

Obtain the JK input equations

Obtain the next state equations

Fill the state table

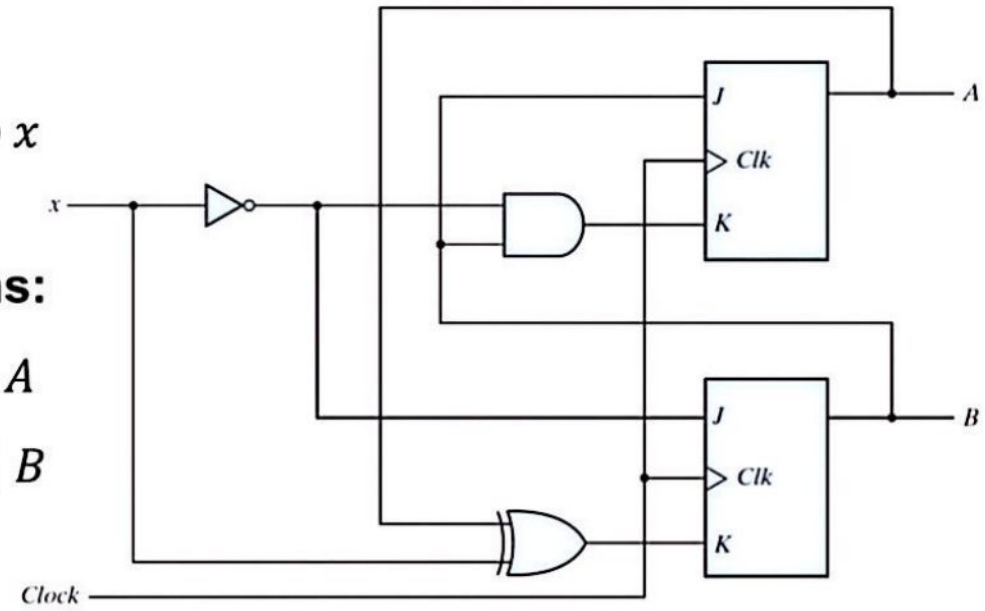
Draw the state diagram



JK Flip-Flop Input Equations:

$J_A = B$ and $K_A = B\,x'$

$J_B = x'$ and $K_B = A \oplus x$



Next State Equations:

$A(t+1) = J_A\,A' + K_A'\,A$

$B(t+1) = J_B\,B' + K_B'\,B$

Substituting:

$A(t+1) = B\,A' + (Bx')'A = A'B + AB' + Ax$

$B(t+1) = x'B' + (A \oplus x)'B = B'x' + A\,B\,x + A'B\,x'$

From JK Input Equations to State Table

JK Input Equations: $J_A = B$, $K_A = B\,x'$, $J_B = x'$ and $K_B = A \oplus x$

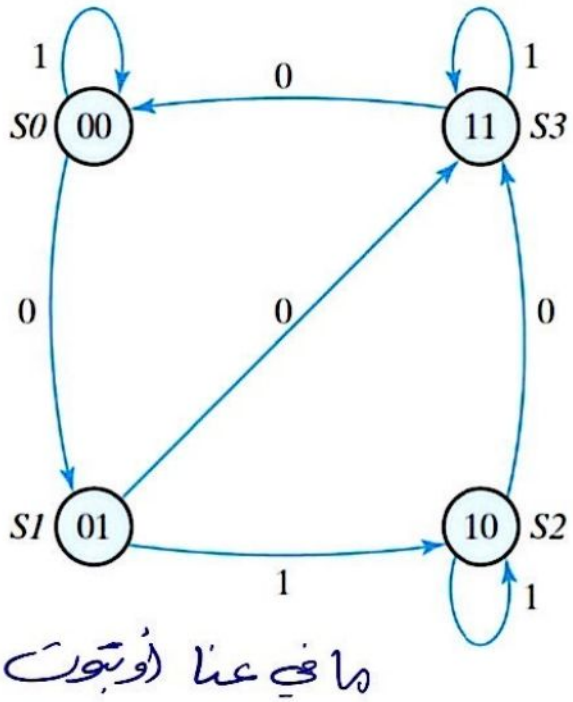
Present State		Input	Next State		Flip-Flop Inputs			
A	B	x	A	B	J_A	K_A	J_B	K_B
0	0	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0	1
0	1	0	1	1	1	1	1	0
0	1	1	1	0	1	0	0	1
1	0	0	1	1	0	0	1	1
1	0	1	1	0	0	0	0	0
1	1	0	0	0	1	1	1	1
1	1	1	1	1	1	0	0	0

From State Table to State Diagram

Four states: $A\,B = 00, 01, 10, and\ 11$ (drawn as circles)

Arcs show the input value x on the state transition

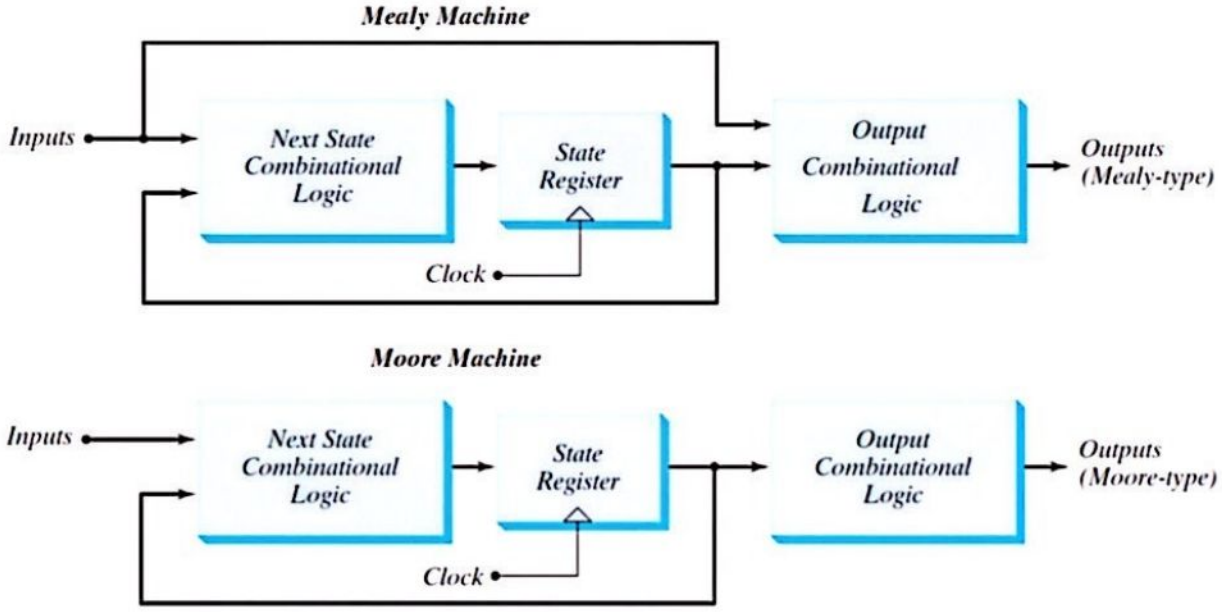
Present State		Input	Next State	
A	B	x	A	B
0	0	0	0	1
0	0	1	0	0
0	1	0	1	1
0	1	1	1	0
1	0	0	1	1
1	0	1	1	0
1	1	0	0	0
1	1	1	1	1



Mealy versus Moore Sequential Circuits

There are two ways to design a clocked sequential circuit:

1. **Mealy Machine:** Outputs depend on present state and inputs
2. **Moore Machine:** Outputs depend on present state only

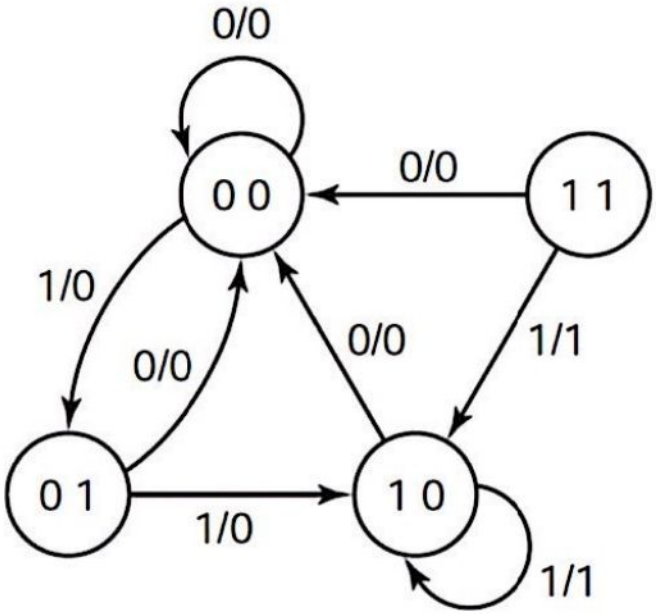


Mealy Machine

- The outputs are a function of the present state and Inputs
- The outputs are **NOT** synchronized with the clock
- The outputs may change if inputs change during the clock cycle
- The outputs may have momentary false values (called glitches)
- The correct outputs are present just before the edge of the clock

Mealy State Diagram

- An example of a Mealy state diagram is shown on the right
- Each arc is labeled with: **Input / Output**
- The output is shown on the arcs of the state diagram
- The output depends on the current state and input
- Notice that State 11 cannot be reached from the other states

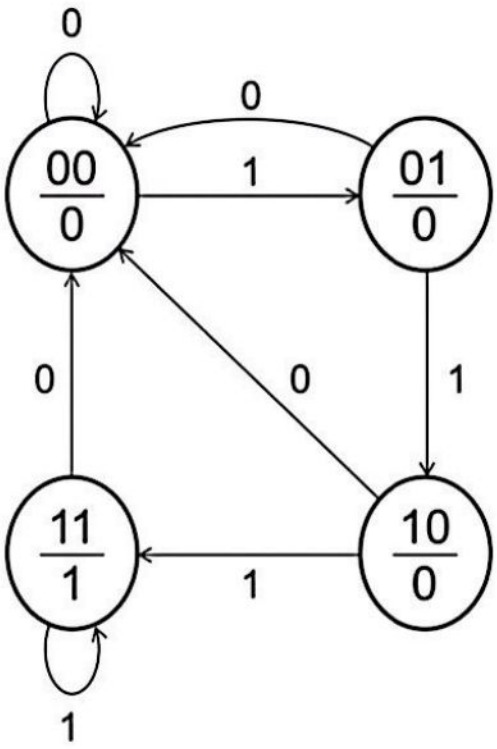


Moore Machine

- The outputs are a function of the Flip-Flop outputs only
- The outputs depend on the current state only
- The outputs are synchronized with the clock
- Glitches cannot appear in the outputs (even if inputs change)
- A given design might mix between Mealy and Moore

Moore State Diagram

- An example of a Moore state diagram is shown on the right
- Arcs are labeled with input only
- The output is shown inside the state: (State / Output)
- The output depends on the current state only



Summary

- To analyze a clocked sequential circuit:

1. Obtain the equations at the **Inputs** of the flip-flops
2. Obtain the **Next State** equations
   - For a D Flip-Flop, the Next State = D input equation
   - For T and JK, use the characteristic equation of the Flip-Flop
3. Obtain the **Output** equations
4. Fill the **State Table**
   - Put all the combinations of current state and input
   - Fill the next state and output columns
5. Draw the **State Diagram**

- Two types of clocked sequential circuits: **Mealy** versus **Moore**