LOGIC DESIGN

Chapter 5 Synchronous Sequential Logic

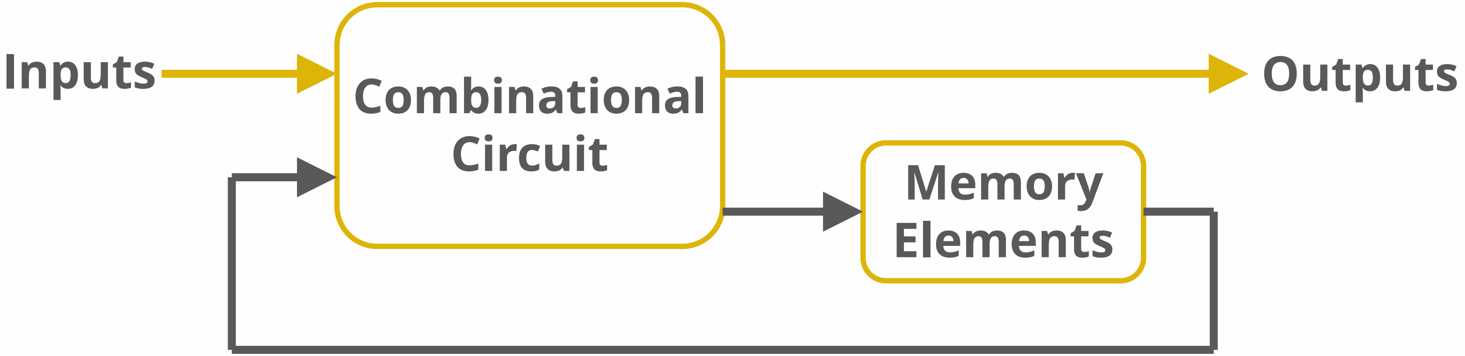
OUTLINE OF CHAPTER 5

- Sequential Circuits
- Latches
- Flip-flop
- Analysis of Clocked Sequential Circuits
- Design Procedure

5.1 SEQUENTIAL CIRCUITS

SEQUENTIAL CIRCUITS

- Every digital system is likely to have combinational circuits.
- Most systems encountered in practice also include **storage elements**, which require that the system be described in terms of **sequential logic**.



SEQUENTIAL CIRCUITS

- The storage elements are devices capable of storing binary information.
- The binary information stored in these elements at any given time defines the **state** of the sequential circuit at that time.
- The sequential circuit receives binary information from external inputs.
- These inputs, together with the present state of the storage elements, determine the binary value of the outputs.

SEQUENTIAL CIRCUITS

- They also determine the condition for changing the state in the storage elements.
- A sequential circuit is specified by a time sequence of inputs, output, and internal states.

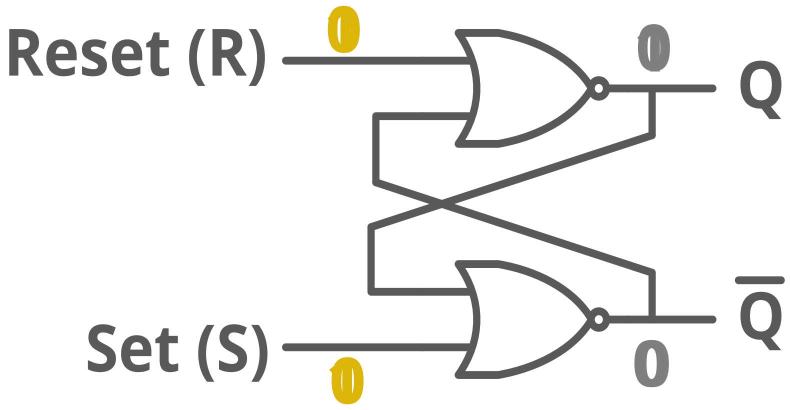
5.2 LATCHES

LATCHES

- **Latches** are the basic circuits from which all flip – flops are constructed.
- latches are useful for storing binary information

LATCHES

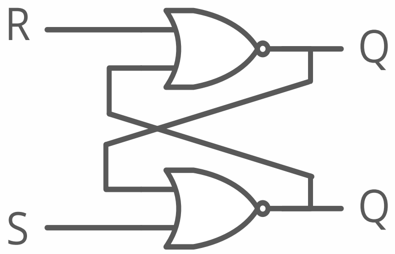
- **SR Latch**



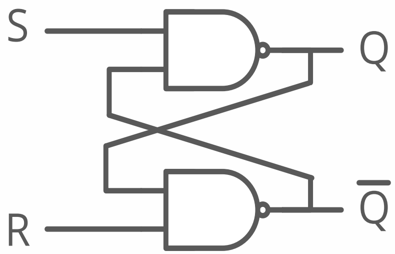
S	R	Q	Q’	
1	0	1	0	Set State
0	0	1	0	Hold State
0	1	0	1	Reset State
0	0	0	1	Hold State
1	1	0	0	Invalid State

LATCHES

- **SR Latch**



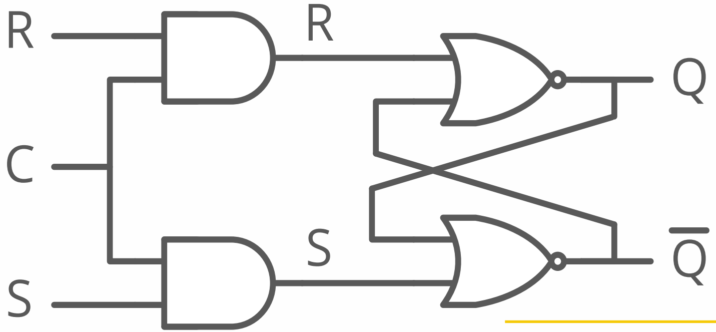
S	R	Q	
0	0	Q_0	No change
0	1	0	Reset
1	0	1	Set
1	1	Q=Q'=0	Invalid

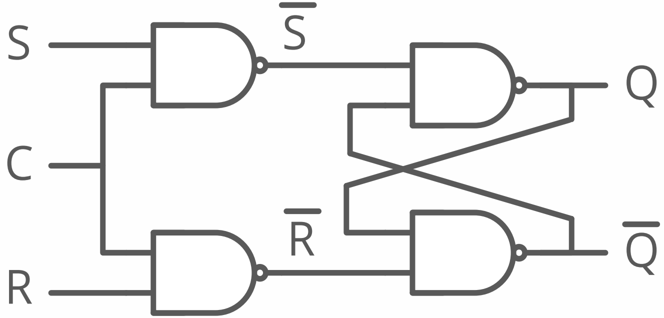


S	R	Q	
0	0	Q=Q'=1	Invalid
0	1	1	Set
1	0	0	Reset
1	1	Q_0	No change

LATCHES

- **SR Latch with Control Input**





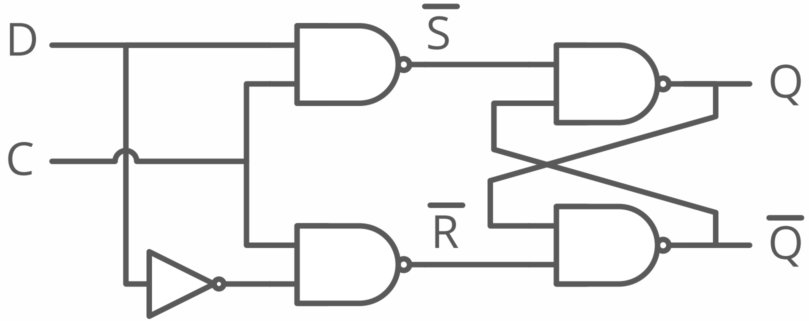
C	S	R	Q	
0	X	X	HOLD	No change
1	0	0	HOLD	No change
1	0	1	Q = 0	Reset
1	1	0	Q = 1	Set
1	1	1	Q = Q’	Invalid

LATCHES

- **D Latch (D = Data)**
  - One way to eliminate the undesirable condition of the indeterminate state in the SR latch is to ensure that inputs S and R are never equal to 1 at the same time.
  - D latch has two inputs
    - D (data) - directly goes to the S input and its complement is applied to the R input.
    - C (control)

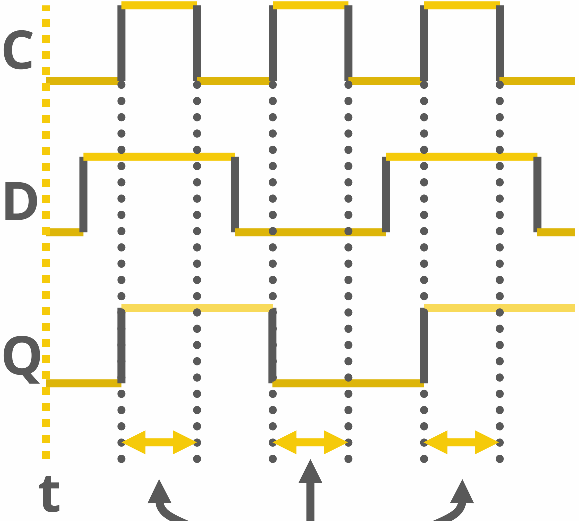
LATCHES

- **D Latch (D = Data)**



C	D	Q	
0	X	HOLD	No change
1	0	Q = 0	Reset
1	1	Q = 1	Set

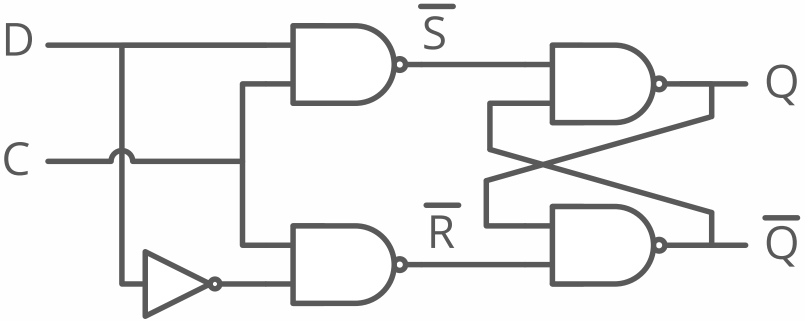
Timing Diagram



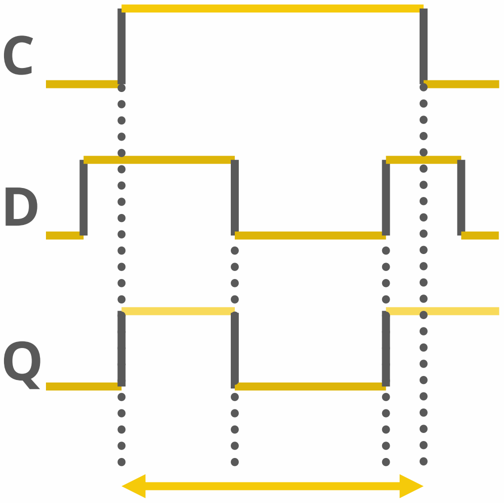
Output may change

LATCHES

- **D Latch (D = Data)**



Timing Diagram

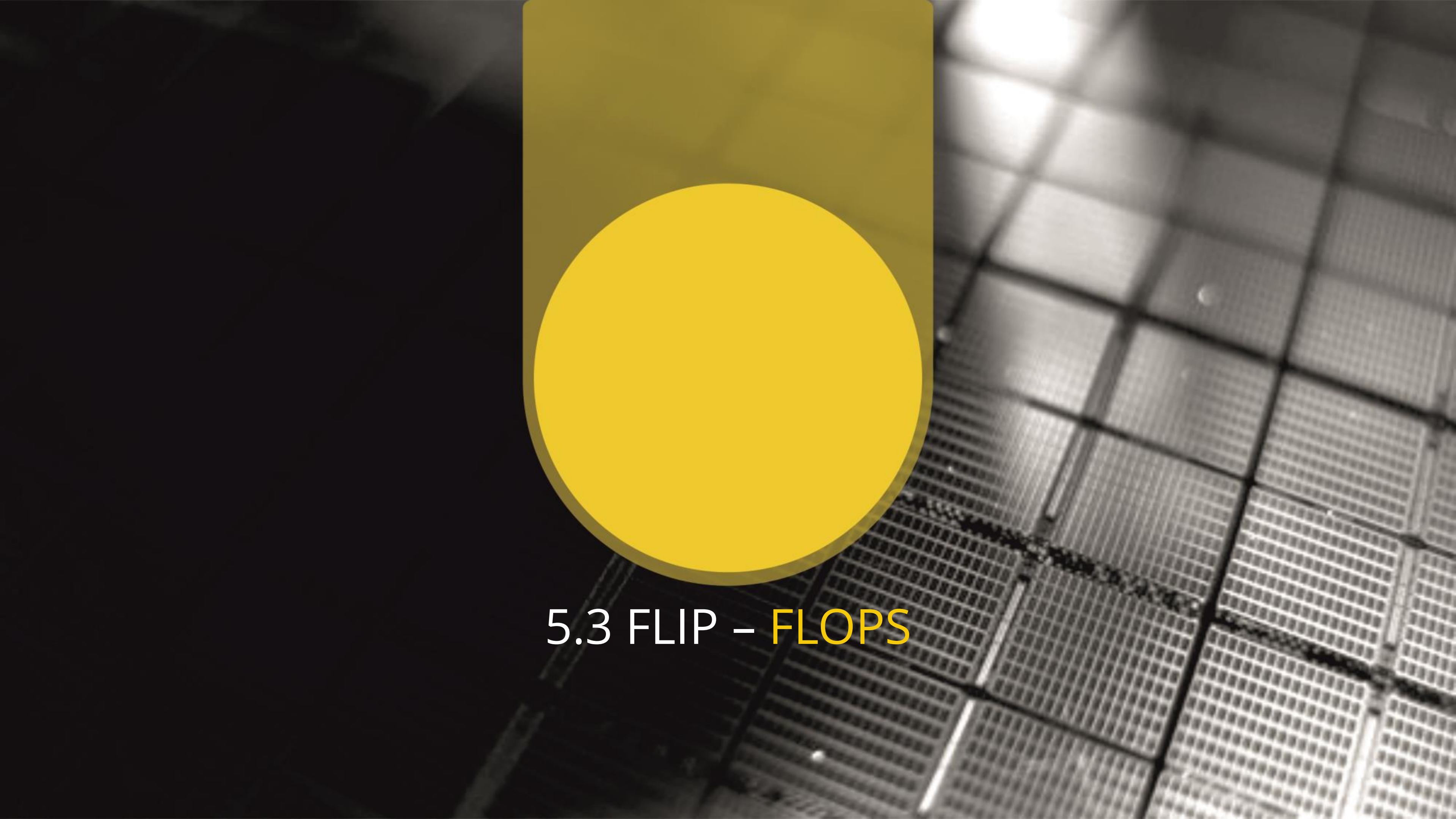


Output may change

C	D	Q	
0	X	HOLD	No change
1	0	Q = 0	Reset
1	1	Q = 1	Set

LATCHES

- **D Latch (D = Data)**
  - The D latch has an ability to hold data in its internal storage.
  - It is suited for use as a temporary storage for binary information.
  - This circuit is often called **transparent** latch.
    - The output follow changes in the data input as long as the control input is **enabled**.



5.3 FLIP – FLOPS

FLIP – FLOPS

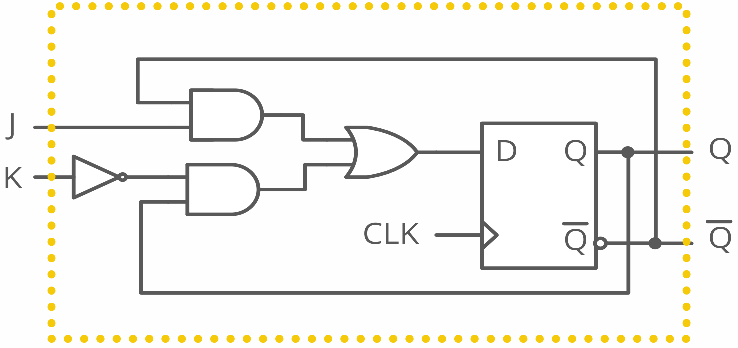
- The most economical and efficient flip – flop constructed is the edge – triggered D flip – flop.
  - It requires smallest number of gates.
- Other types of flip – flops can be constructed by using the D flip – flop and external logic.
  - JK flip – flops
  - T flip - flops

FLIP – FLOPS

- There are three operations that can be performed with a flip – flop:
  - Set it to 1
  - Reset it to 0
  - Complement its output

FLIP – FLOPS

- JK Flip – Flop
  - Performs all three operations.

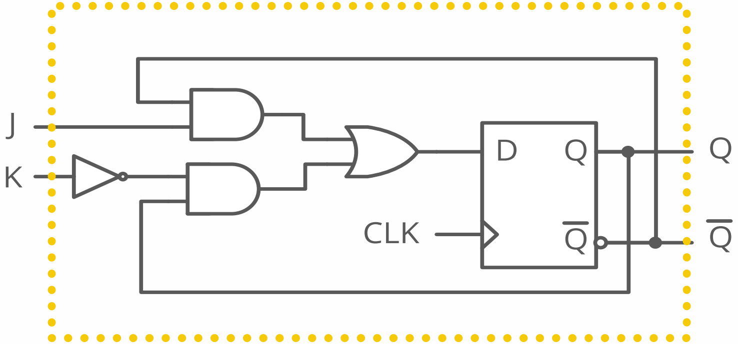


D = JQ’ + K’Q

- When J = 1, sets the flip – flop to 1.
- When K = 1, resets the flip – flop to 0.

FLIP – FLOPS

- JK Flip – Flop



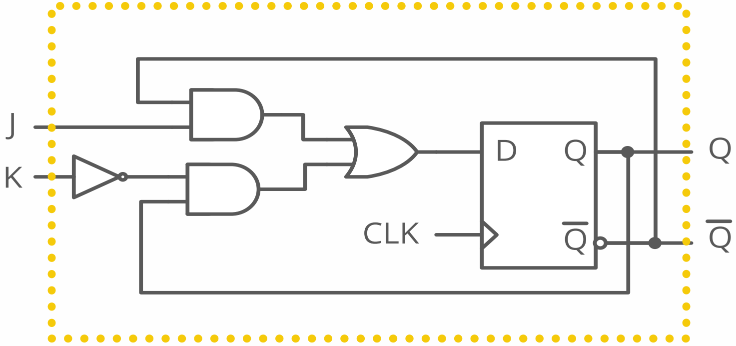
D = JQ’ + K’Q

Operation 1

- When J = 1 and K = 0,
  - D = 1.Q’ + 1.Q (Post2b)
  - D = Q’ + Q (Post5a)
  - D = 1
  - Next clock edge sets the output to 1.

FLIP – FLOPS

- JK Flip – Flop



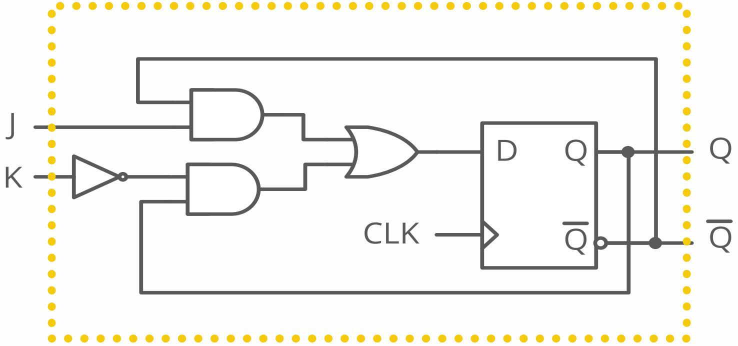
D = JQ’ + K’Q

Operation 2

- When J = 0 and K = 1,
  - D = 0.Q’ + 0.Q (Theo2b)
  - D = 0 + 0
  - D = 0
  - Next clock edge sets the output to 0.

FLIP – FLOPS

- JK Flip – Flop



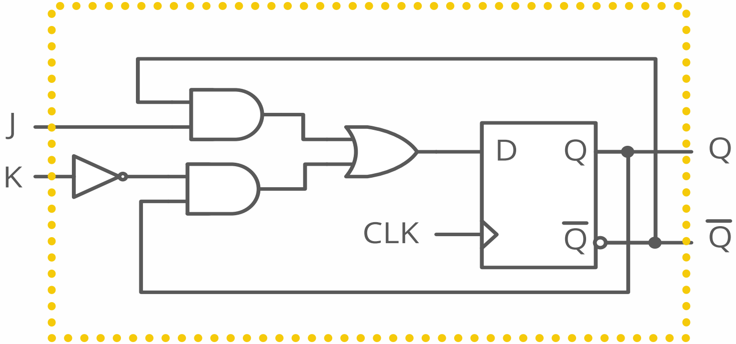
D = JQ' + K'Q

Operation 3

- When J = 1 and K = 1,
  - D = 1.Q' + 0.Q (Post2b)
  - D = Q' + 0 .Q (Theo2b)
  - D = Q' + 0 (Post2a)
  - D = Q'
  - Next clock edge complements the output.

FLIP – FLOPS

- JK Flip – Flop

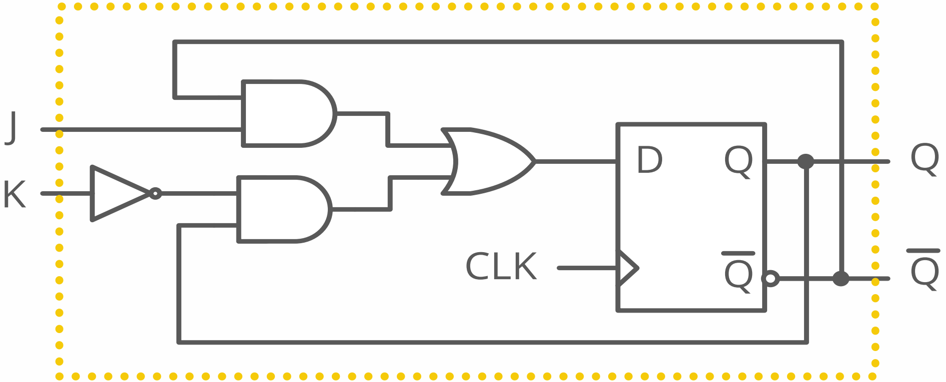


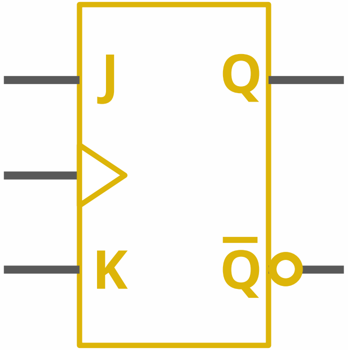
D = JQ’ + K’Q

- When J = 0 and K = 0,
  - D = 0.Q’ + 1.Q (Theo2b)
  - D = 0 + 1 .Q (Post2b)
  - D = 0 + Q (Post2a)
  - D = Q
  - Next clock edge the output is unchanged.

FLIP – FLOPS

- JK Flip – Flop

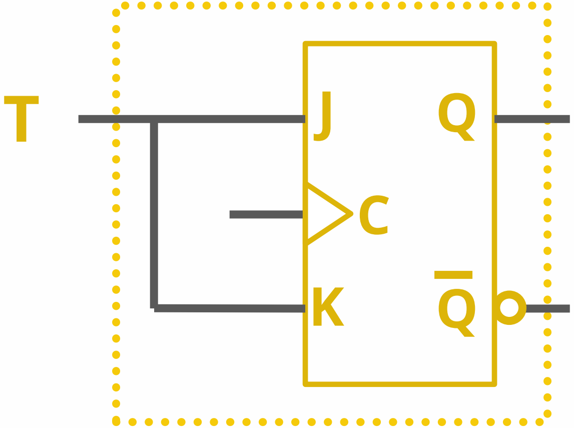




D = JQ’ + K’Q

FLIP – FLOPS

- T (toggle) Flip – Flop
  - Complementing flip – flop.
  - Can be obtained from a JK flip – flop.
  - When inputs J and K are tied together.
  - Useful for designing binary counters.

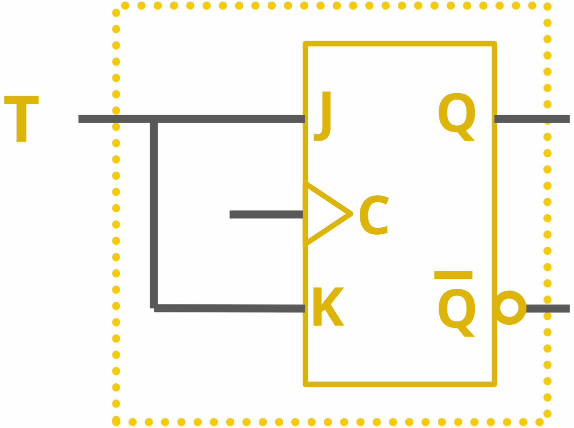


$$D = JQ' + K'Q$$

$$D = TQ' + T'Q = T \oplus Q$$

FLIP – FLOPS

- T (toggle) Flip – Flop
    - When T = 0 (J = K = 0)
    - A clock edge does not change the output.
    - When T = 1 (J = K = 1)
    - A clock edge complements the output.

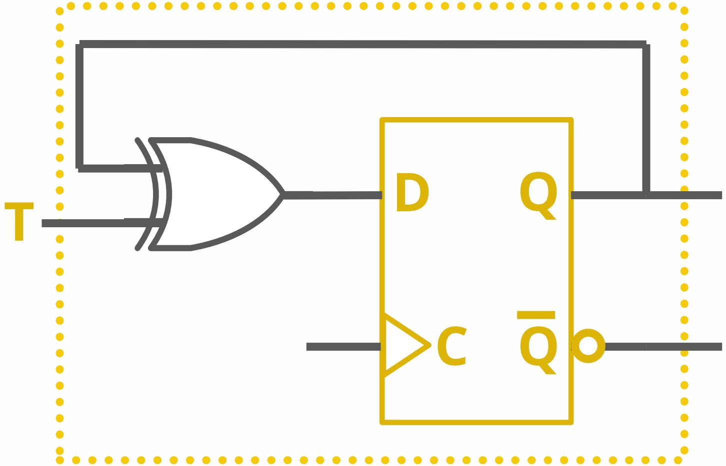


$$D = JQ' + K'Q$$

$$D = TQ' + T'Q = T \oplus Q$$

FLIP – FLOPS

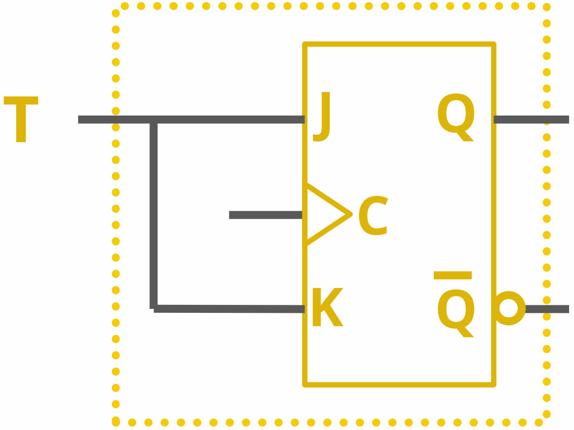
- T (toggle) Flip – Flop
  - Can be constructed with a D flip – flop and an XOR gate.
  - When T = 0 then D = Q
    - No change in the output.
  - When T = 1 then D = Q’
    - Output complements



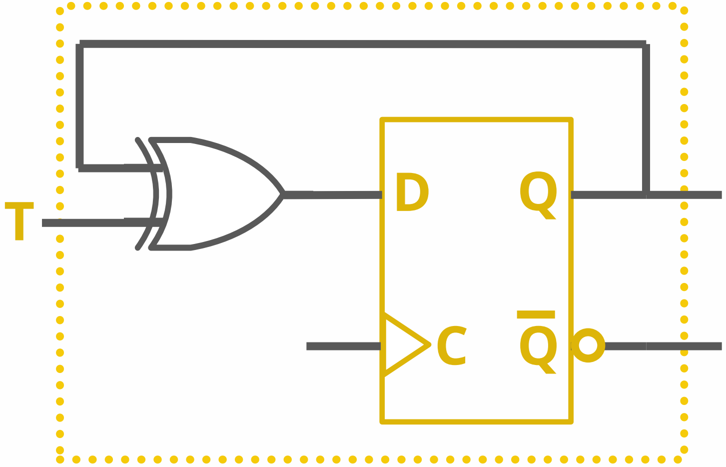
$D = TQ' + T'Q = T \oplus Q$

FLIP – FLOPS

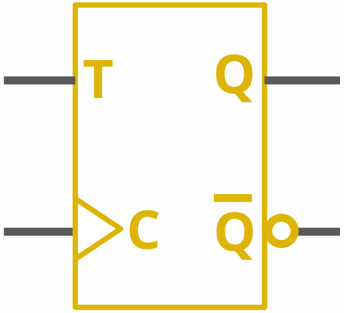
- T (toggle) Flip – Flop



(a) From JK Flip – Flop



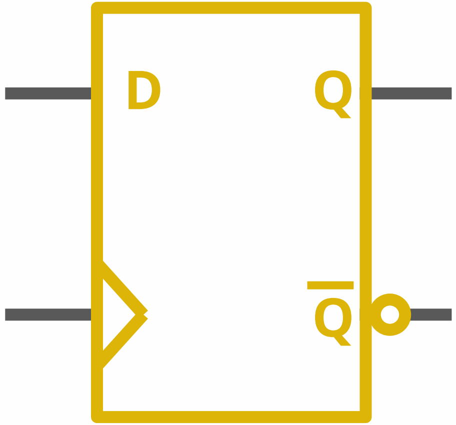
(b) From D Flip – Flop



(c) Graphic Symbol

FLIP – FLOPS

- Flip – Flop Characteristics Table

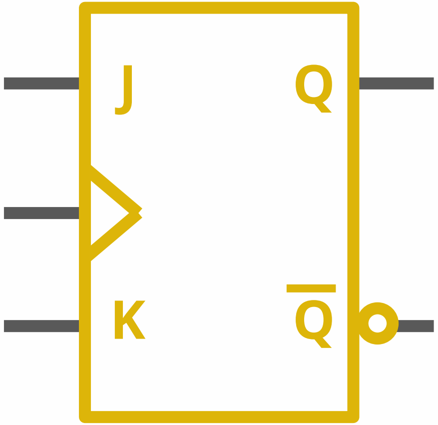


D	Q (t+1)	
0	0	Reset
1	1	Set

$$Q(t+1) = D$$

FLIP – FLOPS

- Flip – Flop Characteristics Table

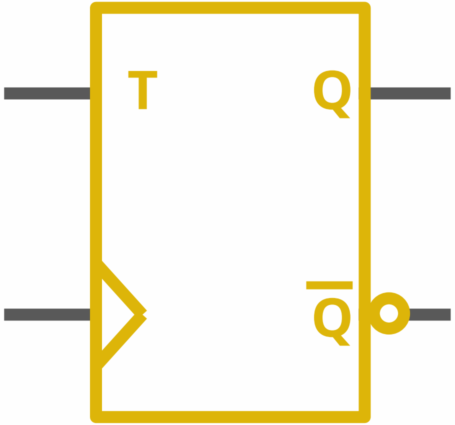


J	K	Q (t+1)	
0	0	Q(t)	No change
0	1	0	Reset
1	0	1	Set
1	1	Q’(t)	Toggle

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

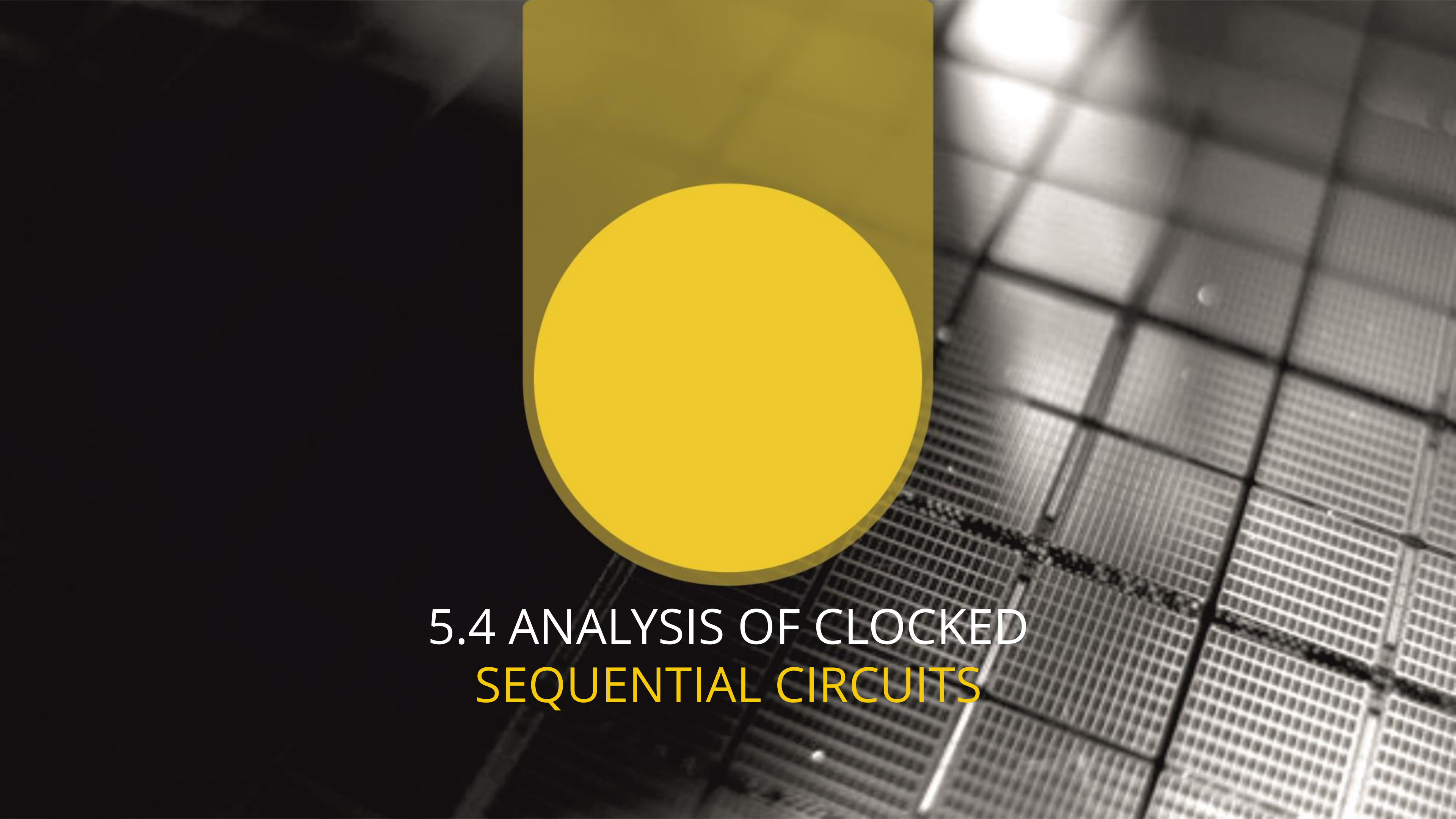
FLIP – FLOPS

- Flip – Flop Characteristics Table



T	Q (t+1)	
0	Q(t)	No change
1	Q’(t)	Toggle

$$Q(t+1) = T \oplus Q$$



5.4 ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

- The behaviour of a clocked sequential circuit is determined from:
    - The inputs
    - The outputs
    - The state of its flip – flops
- The outputs and the next state are both a function of
    - The inputs
    - The present state

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

- The analysis of sequential circuit consists of:
  - Obtaining a table or a diagram for the time sequence of
    - Inputs
    - Outputs
    - Internal states
  - It is also possible to write Boolean expression that describe the behaviour of the sequential circuit.

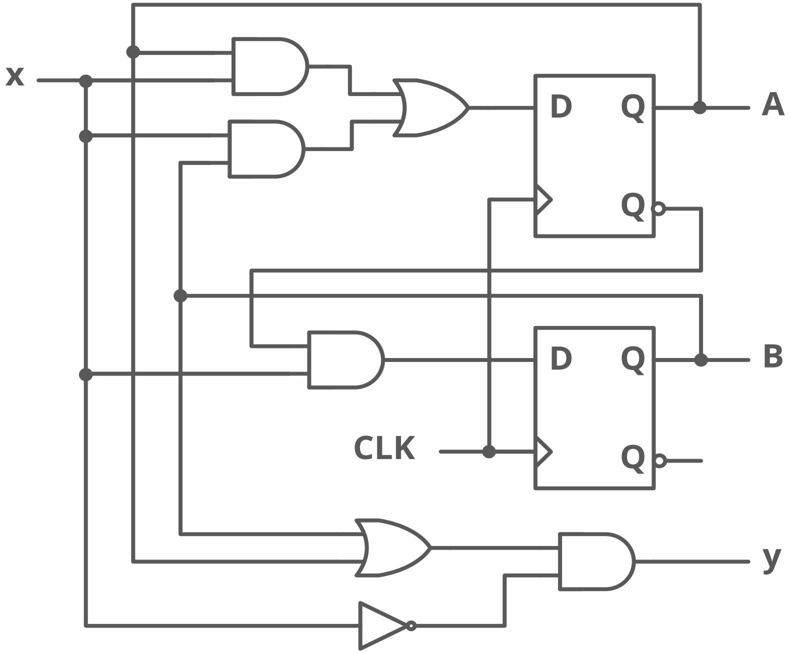
ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

State Equations

- The behaviour of a clocked sequential circuit can be described algebraically by means of **state equations** (**transition equations**).
- A state equation specifies the next state as a function of
  - The present state
  - Inputs

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

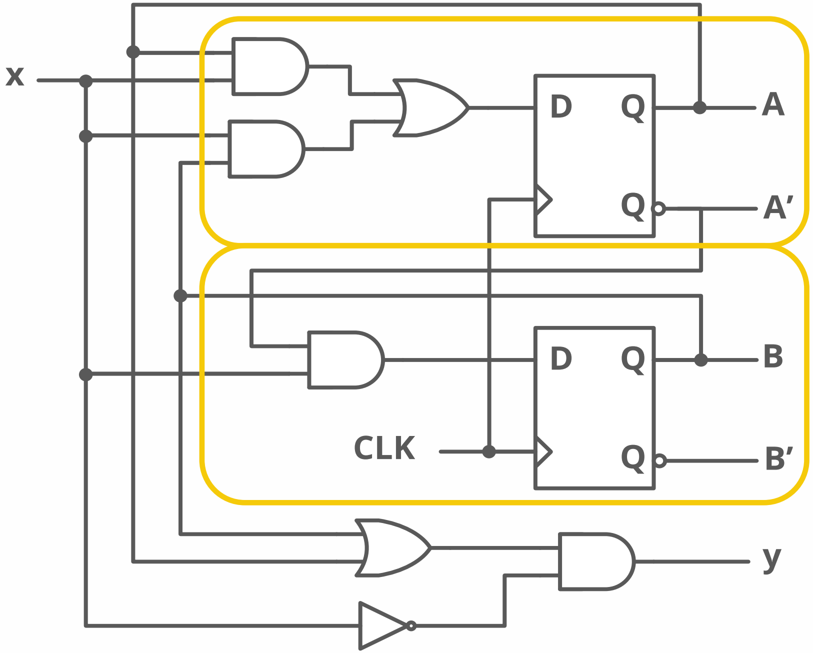
Consider:



- Circuit consists of:
  - Two D flip – flops **A** and **B**.
  - An input **x**.
  - An output **y**.
  - It is possible to write a set of equations for the circuit.

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

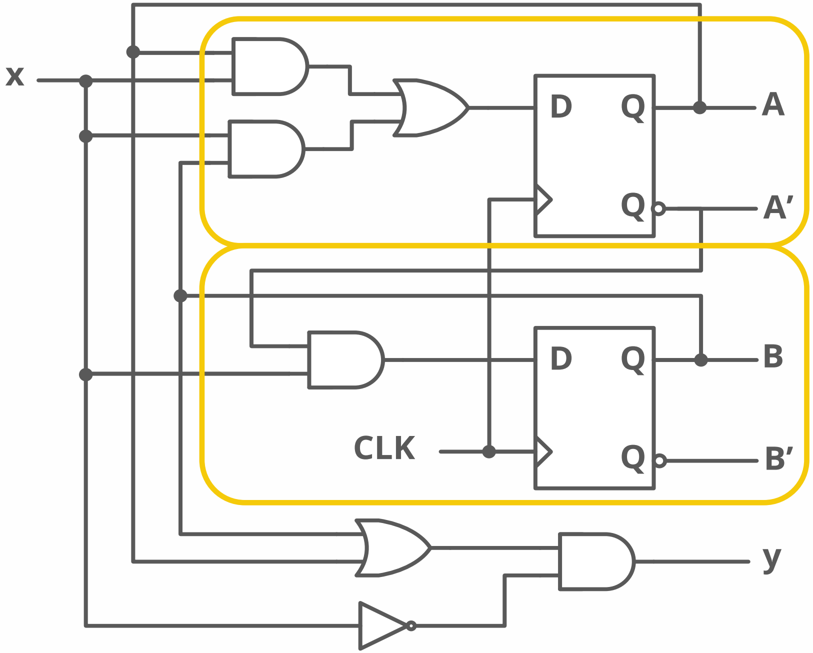
Consider:



- **A(t+1) = A(t) . x(t) + B(t) . x(t)**
- **B(t+1) = A'(t) . x(t)**
  - (t+1) → next state of the flip flop
  - Right side of the equation is a Boolean expression
    - Specifies the present state
    - Input conditions that make the next state = 1.

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

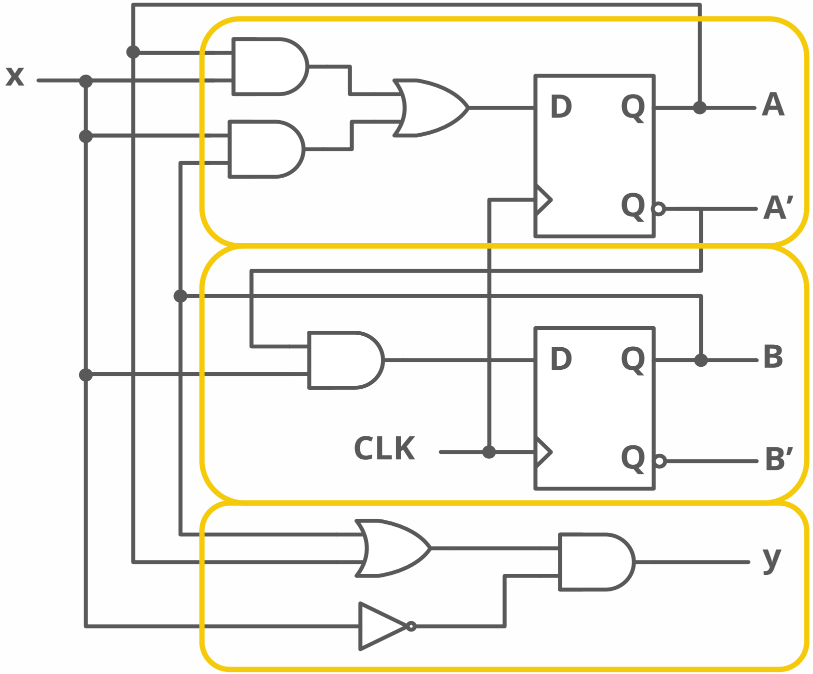
Consider:



- A(t+1) = A(t) . x(t) + B(t) . x(t)
- B(t+1) = A'(t) . x(t)
  - Since all the variables in the Boolean expression are a function of the present state
  - We can omit the designation (t)
- **A(t+1) = A . x + B . x**
- **B(t+1) = A' . x**

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

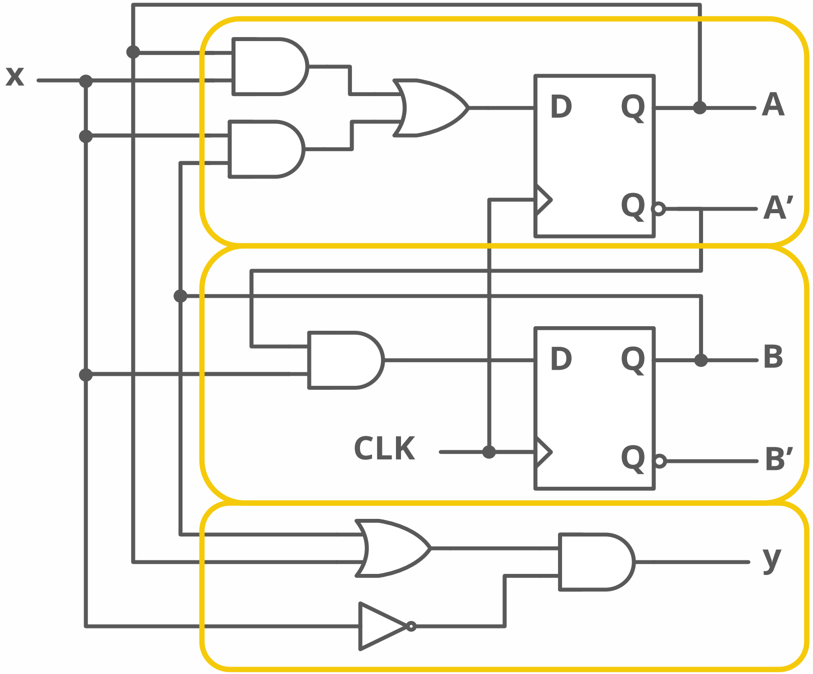
Consider:



- Similarly,
- **y(t) = [A(t) + B(t)] x’(t)**
- **y = (A + B) x′**

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

Consider:



- **A(t+1) = A . x + B . x**
- **B(t+1) = A’ . x**
- **y = (A + B) x’**

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

State Table

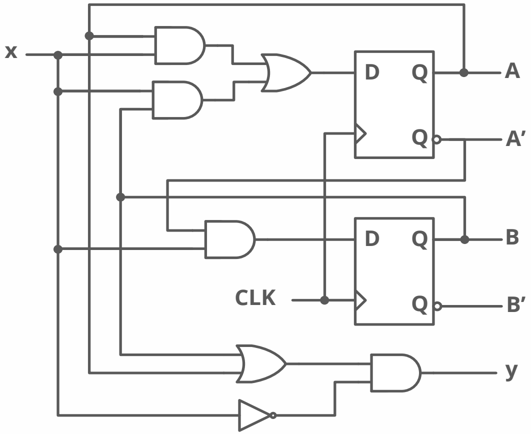
- The time sequence of inputs, outputs and flip – flop can be enumerated in **state table** (**transition table**).
- In general, a sequential circuit with m flip – flops and n inputs needs 2^{m+n} rows in the state table.

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

State Table

Present State (t)		Input (t)	Next State (t+1)		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	0	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

- $A(t+1) = A . x + B . x$
- $B(t+1) = A' . x$
- $y = (A + B) x'$

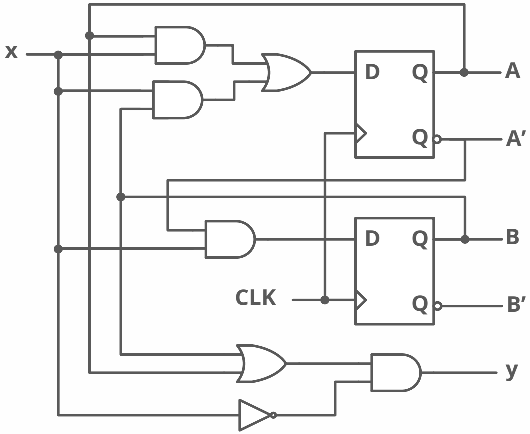


ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

State Table 2

Present State (t)	Next State (t+1)		Output	
	x=0	x=1	x=0	x=1
AB	AB	AB	y	y
00	00	01	0	0
01	00	11	1	0
10	00	10	1	0
11	00	10	1	0

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



ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

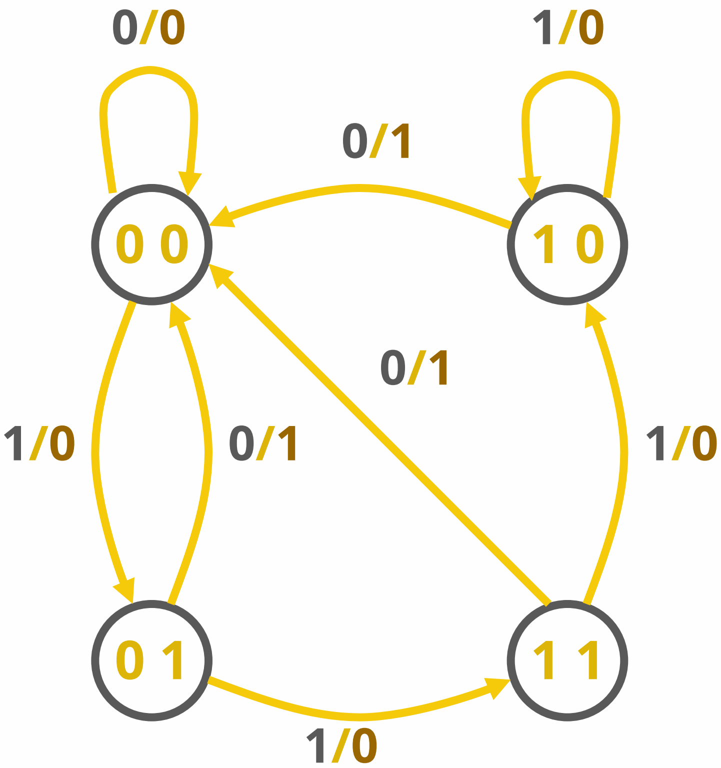
State Diagram

- The information available in a state table can be represented graphically in the form of a state diagram.
- State is represented by a circle
- Transition between states are indicated by directed lines connecting the circles.

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

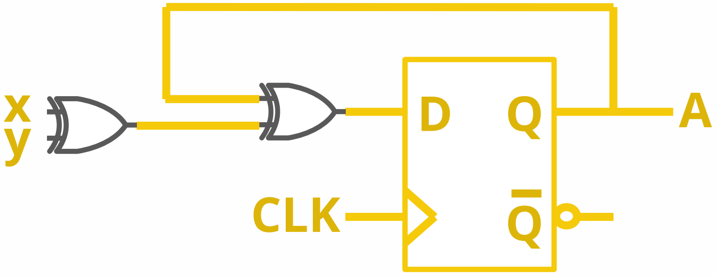
State Diagram

Present State (t)	Next State (t+1)		Output	
	x=0	x=1	x=0	x=1
AB	AB	AB	y	y
00	00	01	0	0
01	00	11	1	0
10	00	10	1	0
11	00	10	1	0

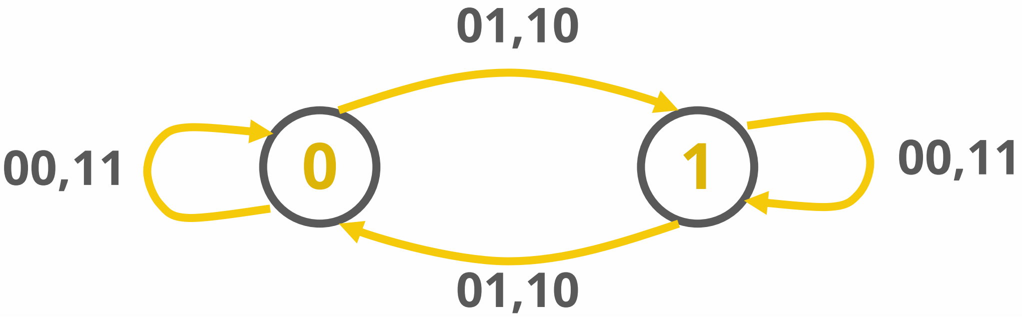


ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

Analysis with D Flip – Flops



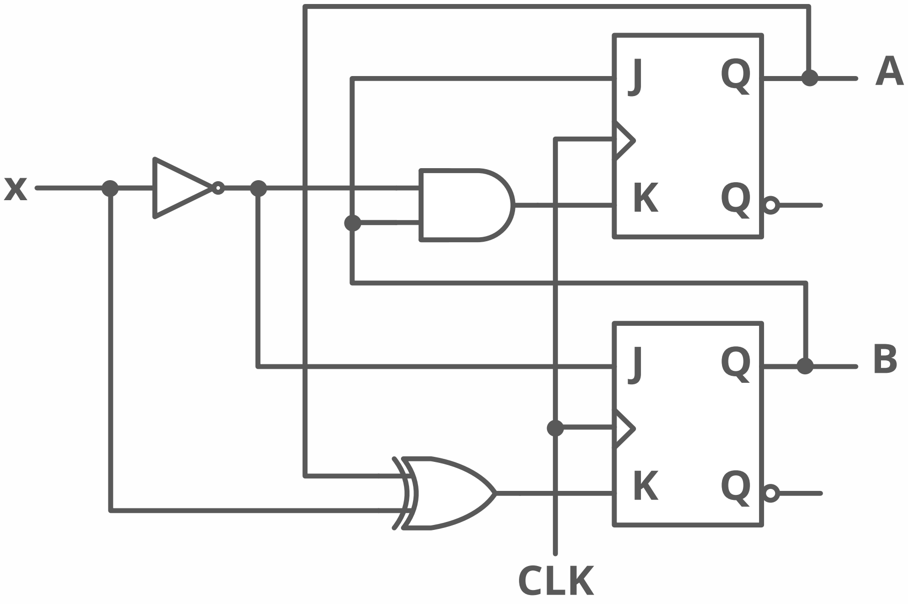
- $A(t+1) = D_A = A \oplus x \oplus y$



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

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

Analysis with JK Flip – Flops



- $J_A = B$ $K_A = B \,.\, x'$
- $J_B = x'$ $K_B = A \oplus x$

- $A(t+1) = J_A\, Q'_A + K'_A\, Q_A$
  $= A'B + AB' + Ax$

- $B(t+1) = J_B\, Q'_B + K'_B\, Q_B$
  $= B'x' + ABx + A'Bx'$

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

Analysis with JK Flip – Flops

- $J_A = B$ $K_A = B\,x'$
- $J_B = x'$ $K_B = A \oplus x$
- $A(t+1) = J_A\,Q'_A + K'_A\,Q_A$

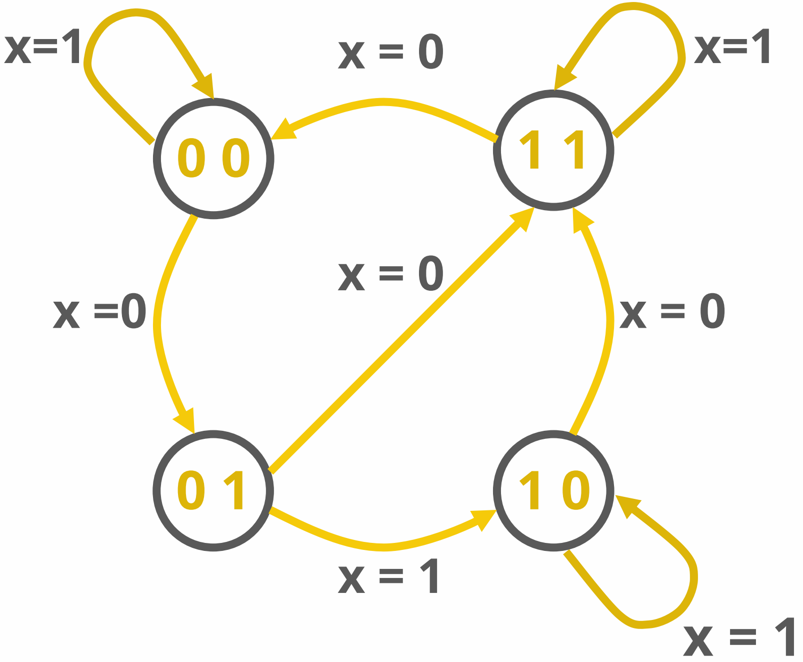
  $= A'B + AB' + Ax$
- $B(t+1) = J_B\,Q'_B + K'_B\,Q_B$

  $= B'x' + ABx + A'Bx'$

Present State		I/P	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

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

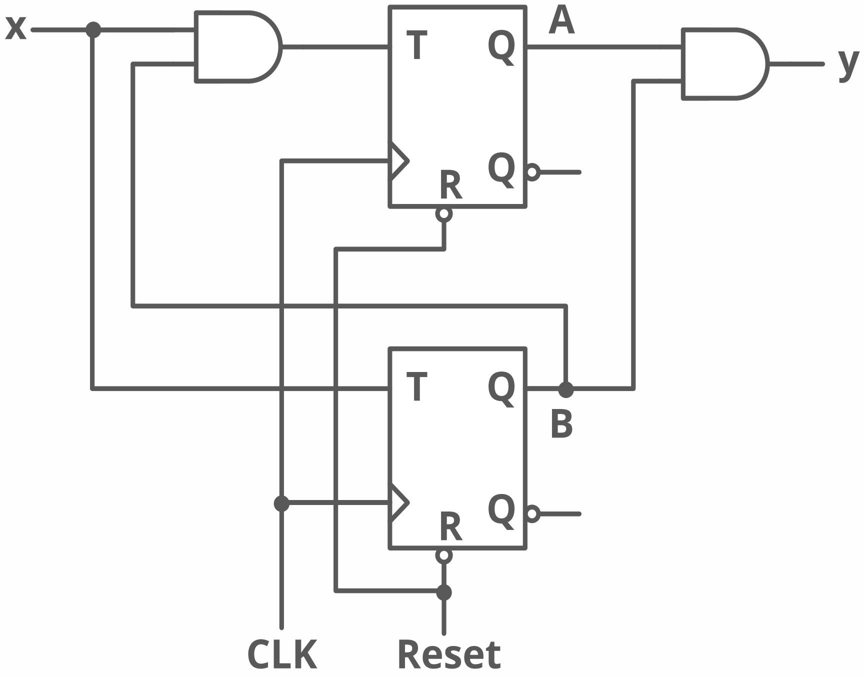
Analysis with JK Flip – Flops



Present State		I/P	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

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

Analysis with T Flip – Flops



- $T_A = B.x$ $T_B = x$
- $y = A\ .\ B$
- $Q(t+1) = T \oplus Q = T'Q + TQ'$
- $A(t+1) = T_A \oplus A = T_A{}' A + T_A A'$

  $= (Bx)' A + BxA'$

  $= (B' + x')A + A'Bx$

  $= AB' + Ax' + A'Bx$
- $B(t+1) = T_B \oplus B = T_B{}' B + T_B B'$

  $= x'B + xB'$

  $= x \oplus B$

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

Analysis with T Flip – Flops

- $T_A = B.x$ $T_B = x$
- $y = A . B$
- $Q(t+1) = T \oplus Q = T'Q + TQ'$
- $A(t+1) = T_A \oplus A = T_A' A + T_A A'$

  $= (Bx)' A + BxA'$

  $= (B' + x')A + A'Bx$

  $= AB' + Ax' + A'Bx$
- $B(t+1) = T_B \oplus B = T_B' B + T_B B'$

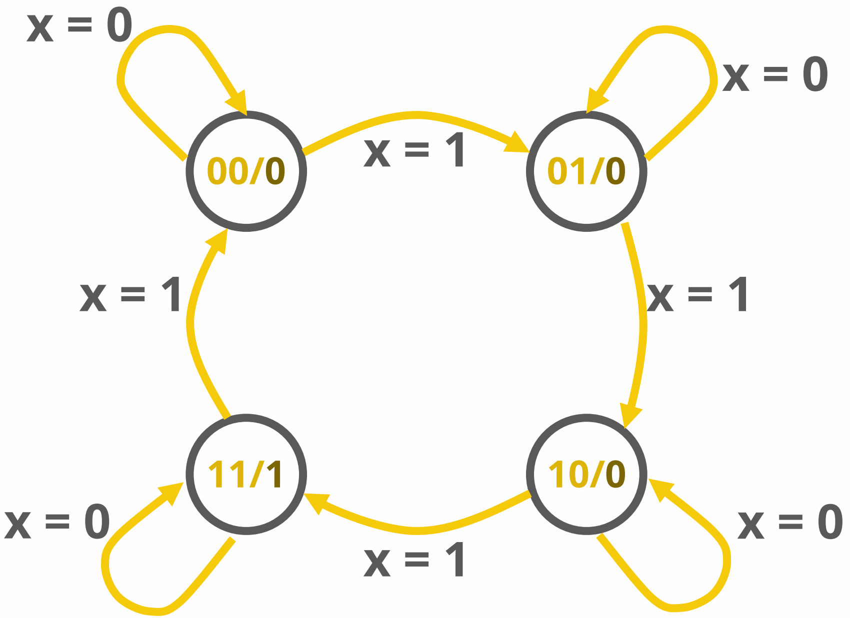
  $= x'B + xB'$

  $= x \oplus B$

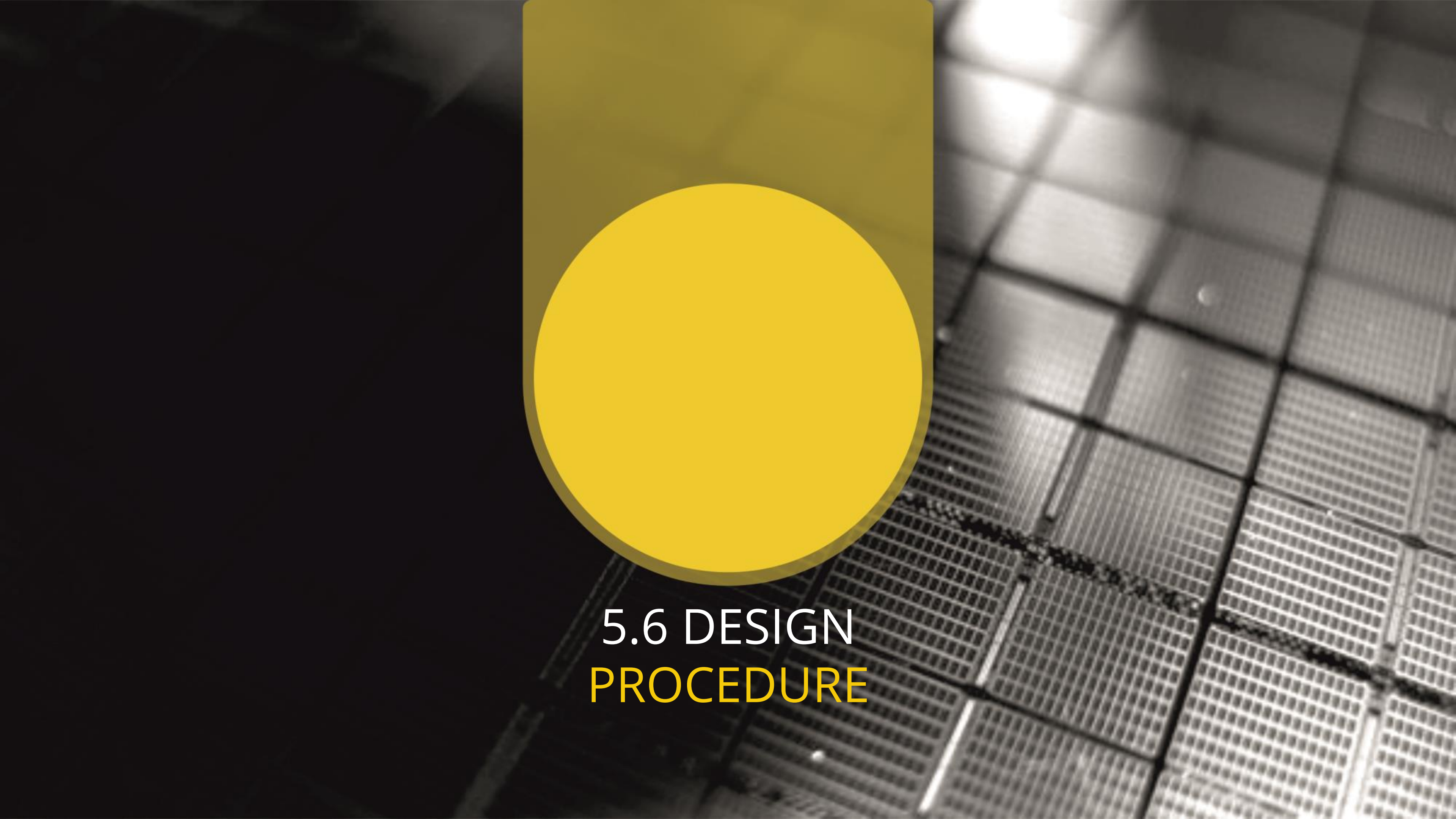
Present State		I/P	Next State		FF Inputs		Output
A	B	x	A	B	T_A	T_B	y
0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	0
0	1	0	0	1	0	0	0
0	1	1	1	0	1	1	0
1	0	0	1	0	0	0	0
1	0	1	1	1	0	1	0
1	1	0	1	1	0	0	1
1	1	1	0	0	1	1	1

ANALYSIS OF CLOCKED SEQUENTIAL CIRCUITS

Analysis with T Flip – Flops



Present State		I/P	Next State		FF Inputs		Output
A	B	x	A	B	T_A	T_B	y
0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	0
0	1	0	0	1	0	0	0
0	1	1	1	0	1	1	0
1	0	0	1	0	0	0	0
1	0	1	1	1	0	1	0
1	1	0	1	1	0	0	1
1	1	1	0	0	1	1	1



5.6 DESIGN PROCEDURE

DESIGN PROCEDURE

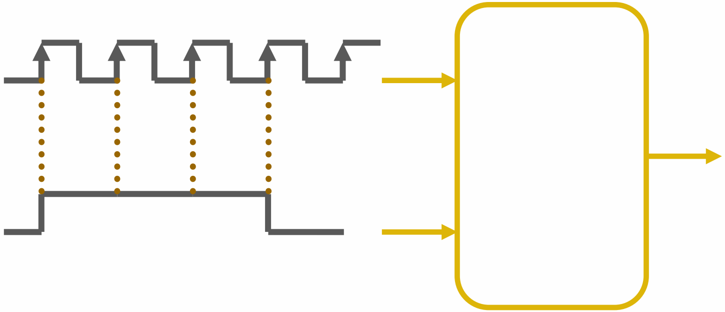
- The design of a clocked sequential circuit starts from
  - a set of specifications and
  - culminates in a logic diagram or
  - a list of Boolean functions from which the logic diagram can be obtained.

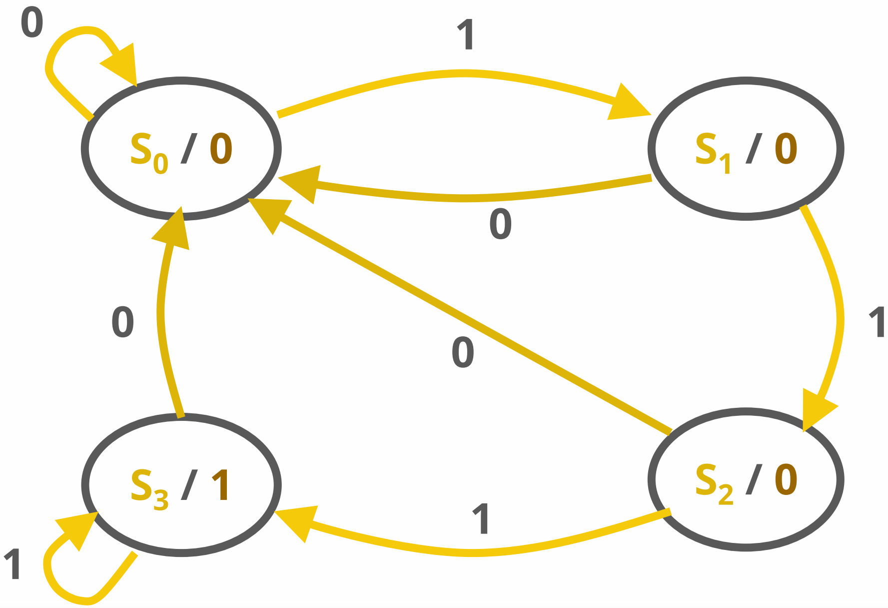
DESIGN PROCEDURE

1. Derive a state diagram for the circuit from the word description.
2. Reduce the number of states if necessary.
3. Assign binary values to the states.
4. Obtain the binary-coded state table.
5. Choose the type of flip-flops.
6. Derive the simplified flip-flop input equations and output equations.
7. Draw the logic diagram.

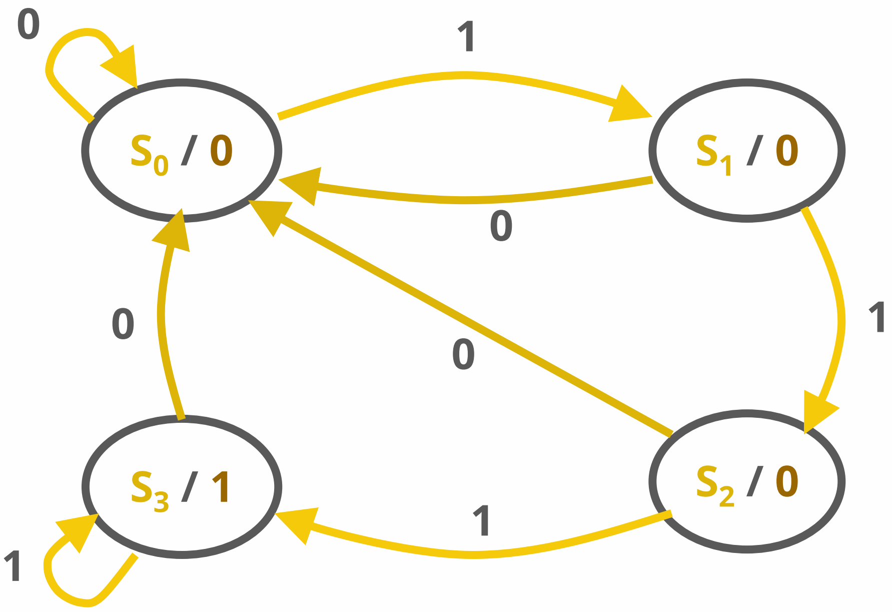
DESIGN PROCEDURE

- Example: We wish to design a circuit that detects three or more consecutive 1’s in a string of bits coming through an input line.
- State diagram:





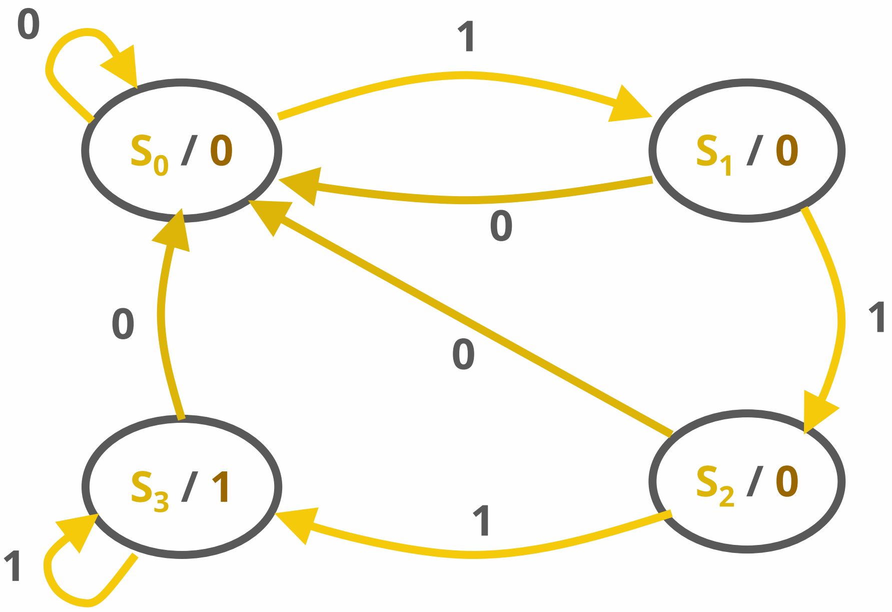
DESIGN PROCEDURE



- This is a Moore model sequential circuit since the output is 1 when the circuit is in State3 and 0 otherwise.

State	A	B
S_0	0	0
S_1	0	1
S_2	1	0
S_3	1	1

DESIGN PROCEDURE



Present State		I/P	Next State		O/P
A	B	x	A	B	y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	0
0	1	1	1	0	0
1	0	0	0	0	0
1	0	1	1	1	0
1	1	0	0	0	1
1	1	1	1	1	1

DESIGN PROCEDURE

- To implement the circuit,
  - Two D flip-flops are chosen to represent the four states and label their outputs A and B.
  - There is one input x.
  - There is one output y.
  - The characteristic equation of the D flip – flop is
    - Q(t+1) = DQ.

DESIGN PROCEDURE

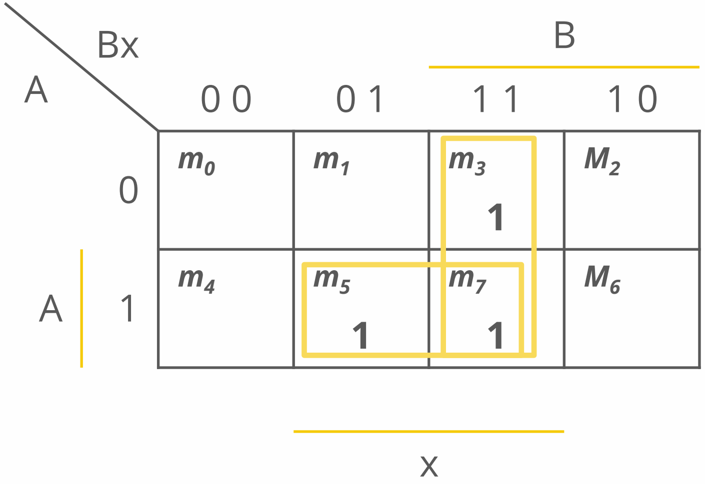
- To implement the circuit,
  - The flip – flop input equations can be obtained directly from the next – state columns of A and B and expressed in sum of minterms.
  - $A(t+1) = D_A(A,B,x) = \sum \mathbf{(3, 5, 7)}$
  - $B(t+1) = D_B(A,B,x) = \sum \mathbf{(1, 5, 7)}$
  - $y(A,B,x) = \sum \mathbf{(6, 7)}$

Present State		**I/P**	**Next State**		**O/P**
A	**B**	**x**	**A**	**B**	**y**
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	0
0	1	1	1	0	0
1	0	0	0	0	0
1	0	1	1	1	0
1	1	0	0	0	1
1	1	1	1	1	1

DESIGN PROCEDURE

- Synthesis using D Flip - flops
  - $A(t+1) = D_A(A,B,x) = \sum \mathbf{(3, 5, 7)}$
  - $B(t+1) = D_B(A,B,x) = \sum \mathbf{(1, 5, 7)}$
  - $y(A,B,x) = \sum \mathbf{(6, 7)}$

- D_A's K - Map



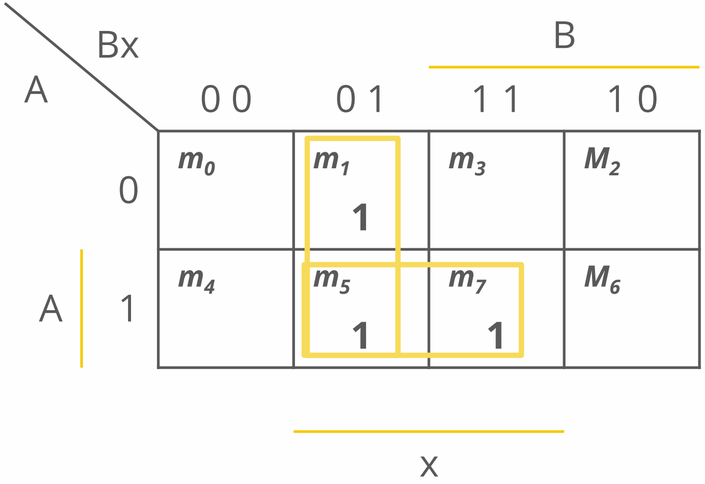
	0 0	0 1	1 1	1 0
0	m_0	m_1	m_3 1	M_2
1	m_4	m_5 1	m_7 1	M_6

$\mathbf{D_A = Ax + Bx}$

DESIGN PROCEDURE

- Synthesis using D Flip – flops
  - $A(t+1) = D_A(A,B,x) = \sum$ **(3, 5, 7)**
  - $B(t+1) = D_B(A,B,x) = \sum$ **(1, 5, 7)**
  - $y(A,B,x) = \sum$ **(6, 7)**

- D_B's K - Map



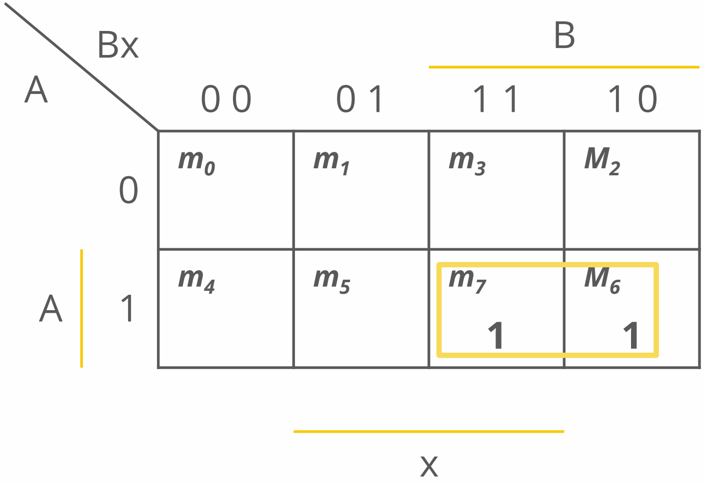
	0 0	0 1	1 1	1 0
0	m_0	m_1 1	m_3	M_2
1	m_4	m_5 1	m_7 1	M_6

D_A = Ax + B'x

DESIGN PROCEDURE

- Synthesis using D Flip – flops
  - $A(t+1) = D_A(A,B,x) = \sum \mathbf{(3, 5, 7)}$
  - $B(t+1) = D_B(A,B,x) = \sum \mathbf{(1, 5, 7)}$
  - $y(A,B,x) = \sum \mathbf{(6, 7)}$

- y’s K - Map

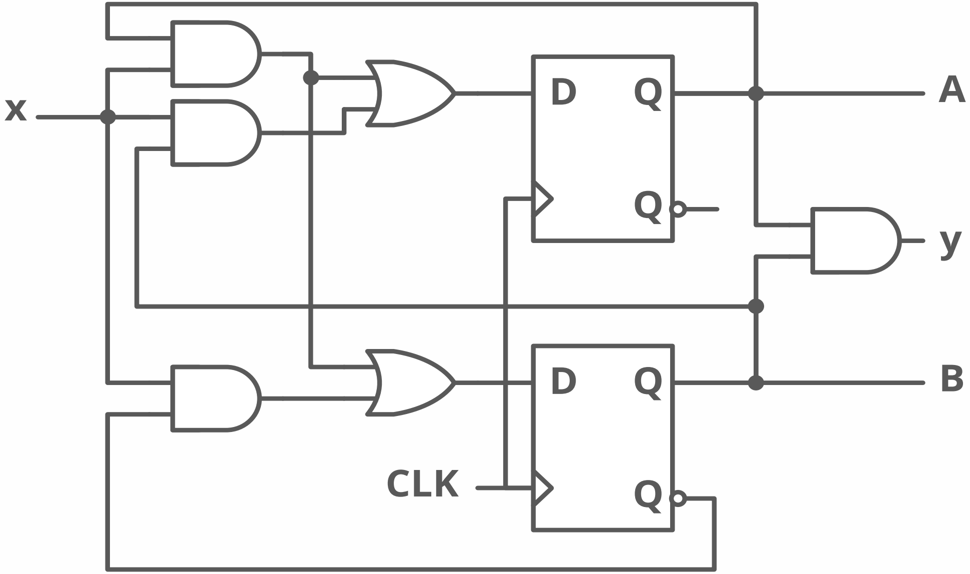


A \ Bx	0 0	0 1	1 1	1 0
0	m_0	m_1	m_3	M_2
1	m_4	m_5	m_7 **1**	M_6 **1**

y = AB

DESIGN PROCEDURE

- Synthesis using D Flip – flops
  - **$D_A = Ax + Bx$**
  - **$D_B = Ax + B'x$**
  - **$y = AB$**
- Logic Diagram of Sequence Detector



DESIGN PROCEDURE

- When – D type flip-flops are employed, the input equations are obtained directly from the next state.
- This is not the case for the JK and T types of flip-flops. In order to determine the input equations for these flip flops, it is necessary to derive a functional relationship between the state table and the input equations.

DESIGN PROCEDURE

- During the design process we usually know the transition from present state to the next state and wish to find the flip – flop input conditions that will cause the required transition.
- For this reason, we need a table that lists the required inputs for a given change of state. Such table is called an **excitation table.**

DESIGN PROCEDURE

- D Flip – Flop Excitation table

D Flip – Flop Characteristic Table

D	Q (t+1)
0	0
1	1

Q(t+1) = D

Present State	Next State	F.F. Input
Q(t)	Q(t+1)	D
0	0	0
0	1	1
1	0	0
1	1	1

DESIGN PROCEDURE

- JK Flip – Flop Excitation table

JK Flip – Flop Characteristic Table

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

Q(t+1) = JQ’ + K’Q

Present State	Next State	F.F. Input	
Q(t)	Q(t+1)	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

0 0 (No change)
0 1 (Reset)

1 0 (Set)
1 1 (Toggle)

0 1 (Reset)
1 1 (Toggle)

0 0 (No change)
1 0 (Set)

DESIGN PROCEDURE

- T Flip – Flop Excitation table

T Flip – Flop Characteristic Table

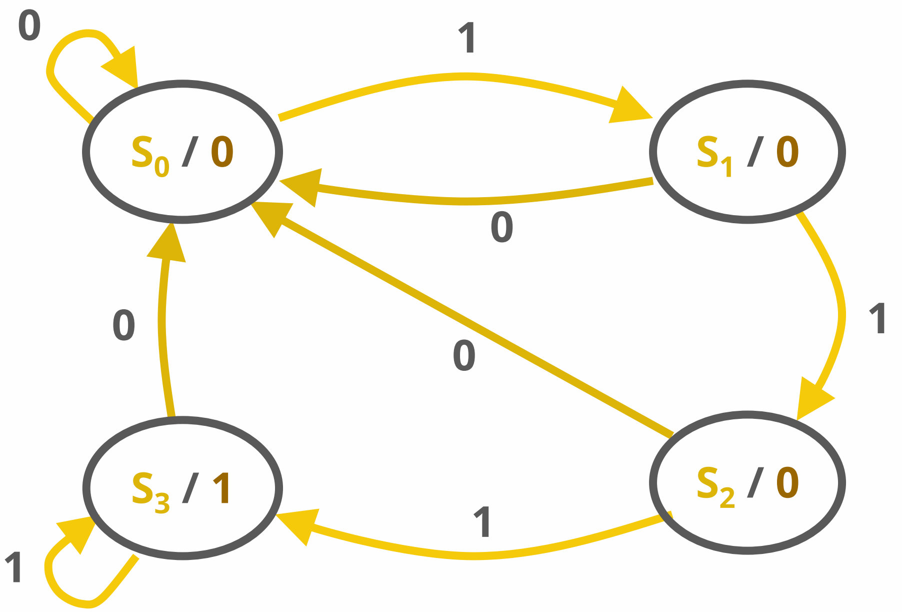
T	Q (t+1)
0	Q(t)
1	Q'(t)

$$Q(t+1) = T \oplus Q$$

Present State	Next State	F.F. Input
Q(t)	Q(t+1)	T
0	0	0
0	1	1
1	0	1
1	1	0

DESIGN PROCEDURE

- Synthesis Using JK Flip – Flops: Detect 3 or more consecutive 1’s



DESIGN PROCEDURE

- Synthesis Using JK Flip – Flops: Detect 3 or more consecutive 1's

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	0	0	X	0	X
0	0	1	0	1	0	X	1	X
0	1	0	0	0	0	X	X	1
0	1	1	1	0	1	X	X	1
1	0	0	0	0	X	1	0	X
1	0	1	1	1	X	0	1	X
1	1	0	0	0	X	1	X	1
1	1	1	1	1	X	0	X	0

$J_A(A, B, x) = \sum(3, 4, 5, 6, 7)$

$K_A(A, B, x) = \sum(0, 1, 2, 3, 4, 6)$

$J_B(A, B, x) = \sum(1, 2, 3, 5, 6, 7)$

$K_B(A, B, x) = \sum(0, 1, 2, 3, 4, 5, 6)$

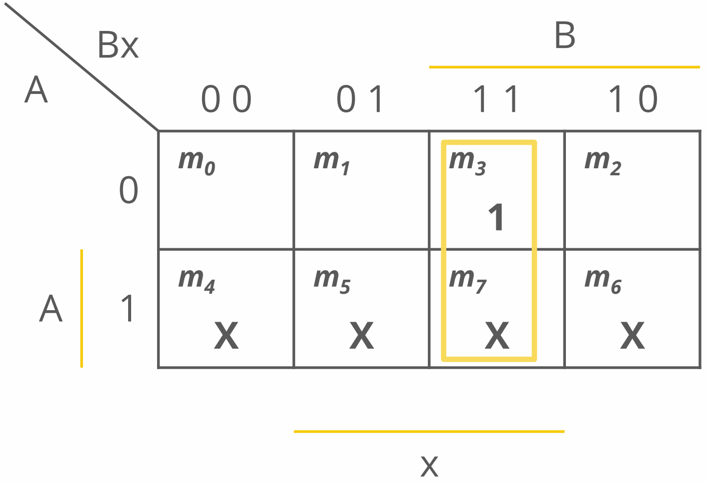
DESIGN PROCEDURE

- Synthesis Using JK Flip – Flops:

  Detect 3 or more consecutive 1's

  - $J_A(A, B, x) = \sum(3, 4, 5, 6, 7)$
  - $K_A(A, B, x) = \sum(0, 1, 2, 3, 4, 6)$
  - $J_B(A, B, x) = \sum(1, 2, 3, 5, 6, 7)$
  - $K_B(A, B, x) = \sum(0, 1, 2, 3, 4, 5, 6)$

- J_A's K-Map



A \ Bx	00	01	11	10
0	m_0	m_1	m_3 1	m_2
1	m_4 X	m_5 X	m_7 X	m_6 X

$J_A = Bx$

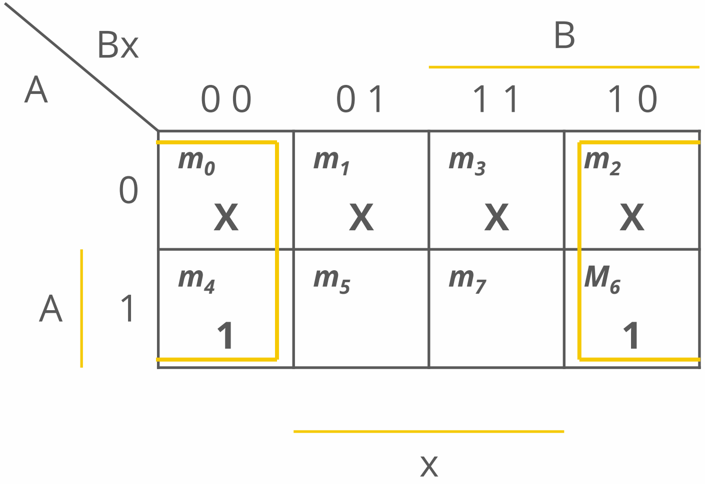
DESIGN PROCEDURE

- Synthesis Using JK Flip – Flops:

  Detect 3 or more consecutive 1's

  - $J_A(A, B, x) = \sum(3, 4, 5, 6, 7)$
  - $K_A(A, B, x) = \sum(0, 1, 2, 3, 4, 6)$
  - $J_B(A, B, x) = \sum(1, 2, 3, 5, 6, 7)$
  - $K_B(A, B, x) = \sum(0, 1, 2, 3, 4, 5, 6)$

- K_A's K-Map



A \ Bx	00	01	11	10
0	m_0 X	m_1 X	m_3 X	m_2 X
1	m_4 1	m_5	m_7	M_6 1

$K_A = x'$

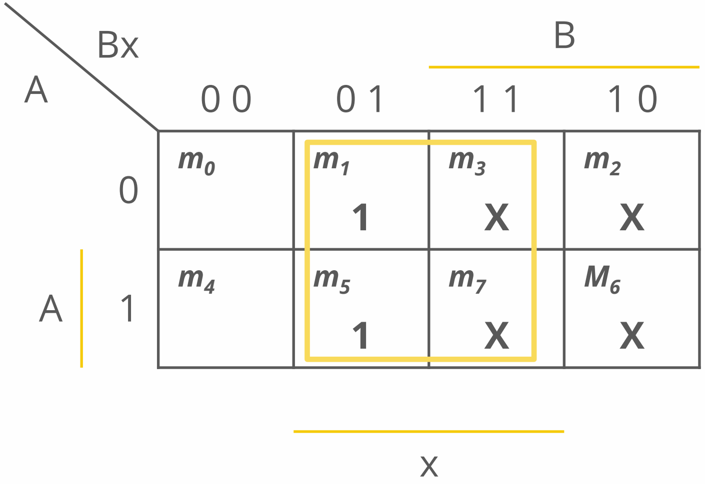
DESIGN PROCEDURE

- Synthesis Using JK Flip – Flops:

  Detect 3 or more consecutive 1's

  - $J_A(A, B, x) = \sum(3, 4, 5, 6, 7)$
  - $K_A(A, B, x) = \sum(0, 1, 2, 3, 4, 6)$
  - $J_B(A, B, x) = \sum(1, 2, 3, 5, 6, 7)$
  - $K_B(A, B, x) = \sum(0, 1, 2, 3, 4, 5, 6)$

- J_B's K-Map



A \ Bx	0 0	0 1	1 1	1 0
0	m_0	m_1 1	m_3 X	m_2 X
1	m_4	m_5 1	m_7 X	M_6 X

$J_B = x$

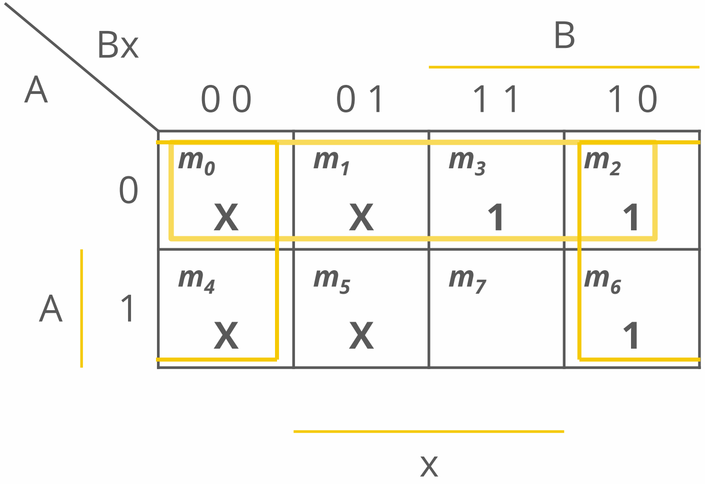
DESIGN PROCEDURE

- Synthesis Using JK Flip – Flops:

  Detect 3 or more consecutive 1's

  - $J_A(A, B, x) = \sum(3, 4, 5, 6, 7)$
  - $K_A(A, B, x) = \sum(0, 1, 2, 3, 4, 6)$
  - $J_B(A, B, x) = \sum(1, 2, 3, 5, 6, 7)$
  - $K_B(A, B, x) = \sum(0, 1, 2, 3, 4, 5, 6)$

- K_B's K-Map

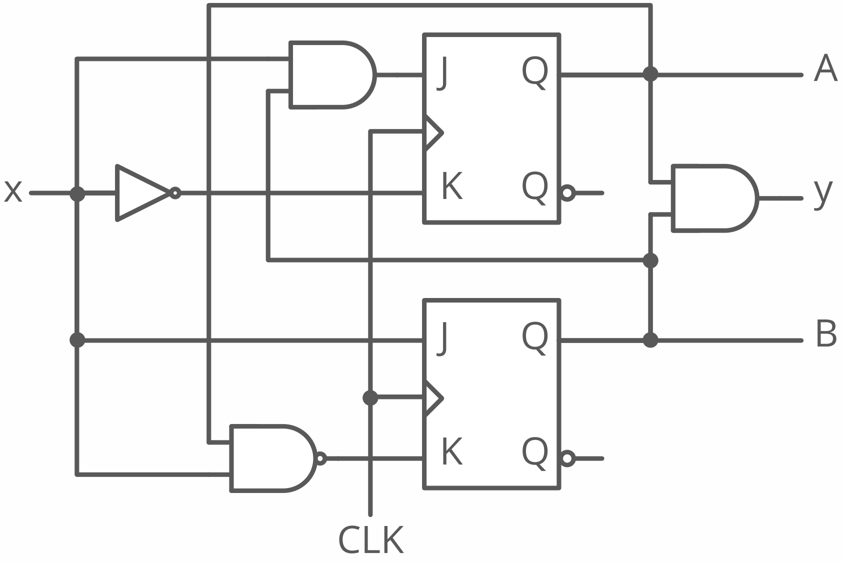


A \ Bx	0 0	0 1	1 1	1 0
0	m_0 X	m_1 X	m_3 1	m_2 1
1	m_4 X	m_5 X	m_7	m_6 1

$K_B = A' + x'$

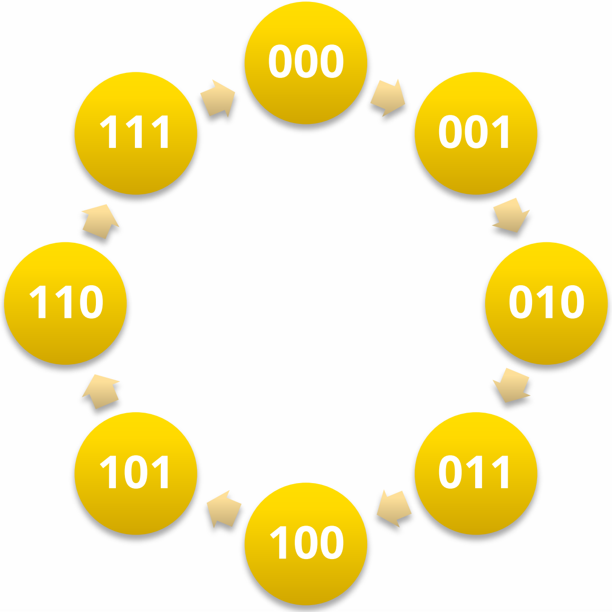
DESIGN PROCEDURE

- Synthesis Using JK Flip – Flops: Detect 3 or more consecutive 1's
  - $J_A = Bx$
  - $K_A = x'$
  - $J_B = x$
  - $K_B = A' + x'$
- Logic Diagram of Sequence Detector



DESIGN PROCEDURE

- Synthesis Using T Flip – Flops: 3-bit Counter. An n-bit binary counter consists of n flip – flops that can count in binary from 0 to $2^n – 1$.



DESIGN PROCEDURE

- Synthesis Using T Flip – Flops: 3-bit Counter.

Present State			Next State			Flip-Flop Inputs		
A_2	A_1	A_0	A_2	A_1	A_0	T_{A2}	T_{A1}	T_{A0}
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1

$T_{A2}(A_2, A_1, A_0) = \sum (3, 7)$

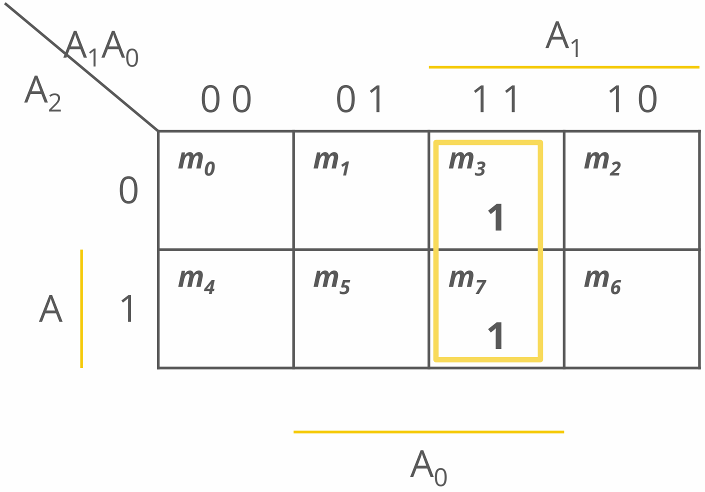
$T_{A1}(A_2, A_1, A_0) = \sum (1, 3, 5, 7)$

$T_{A0}(A_2, A_1, A_0) = \sum (0, 1, 2, 3, 4, 5, 6, 7)$

DESIGN PROCEDURE

- Synthesis Using T Flip – Flops: 3-bit Counter.
  - $T_{A2}(A_2, A_1, A_0) = \sum (3, 7)$
  - $T_{A1}(A_2, A_1, A_0) = \sum (1, 3, 5, 7)$
  - $T_{A0}(A_2, A_1, A_0) = \sum (0, 1, 2, 3, 4, 5, 6, 7)$

- T_{A2}'s K-Map

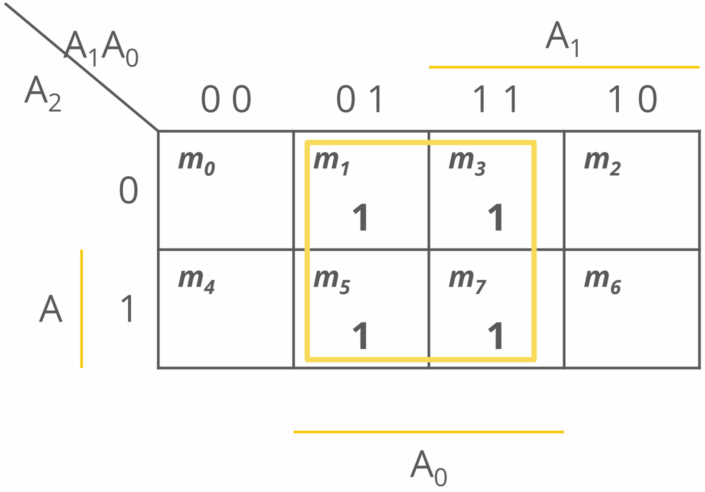


$T_{A2} = A_1A_0$

DESIGN PROCEDURE

- Synthesis Using T Flip – Flops: 3-bit Counter.
  - $T_{A2}(A_2, A_1, A_0) = \sum(3, 7)$
  - $T_{A1}(A_2, A_1, A_0) = \sum(1, 3, 5, 7)$
  - $T_{A0}(A_2, A_1, A_0) = \sum(0, 1, 2, 3, 4, 5, 6, 7)$

- T_{A1}'s K-Map



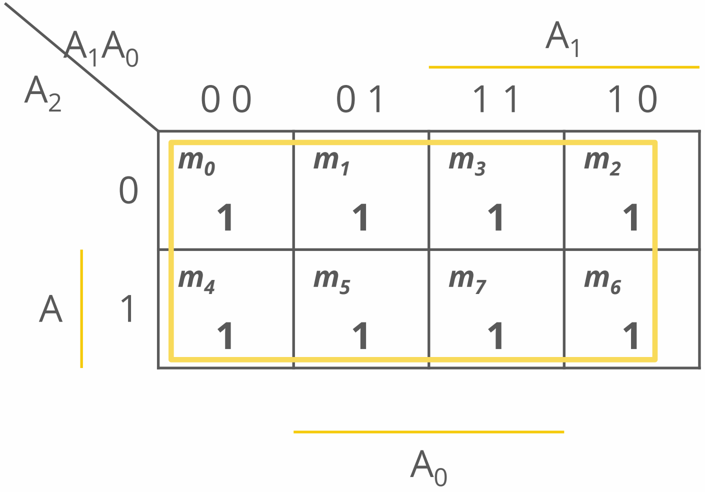
$A_2 \backslash A_1A_0$	00	01	11	10
0	m_0	m_1 1	m_3 1	m_2
1	m_4	m_5 1	m_7 1	m_6

$T_{A1} = A_0$

DESIGN PROCEDURE

- Synthesis Using T Flip – Flops: 3-bit Counter.
  - $T_{A2} (A_2, A_1, A_0) = \sum (3, 7)$
  - $T_{A1} (A_2, A_1, A_0) = \sum (1, 3, 5, 7)$
  - $T_{A0} (A_2, A_1, A_0) = \sum (0, 1, 2, 3, 4, 5, 6, 7)$

- T_{A0}'s K-Map



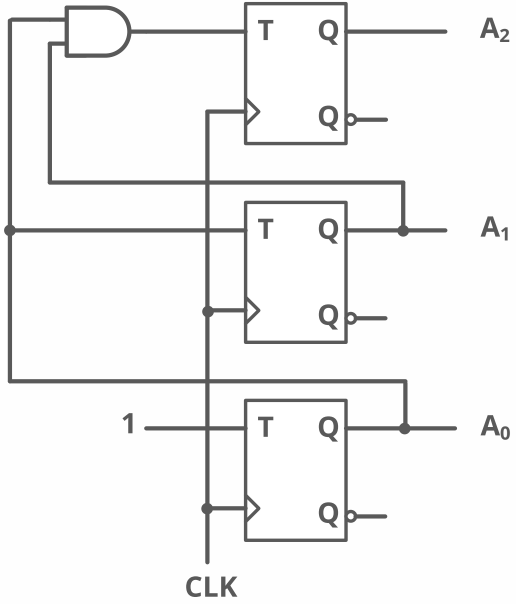
A_2 \ A_1A_0	0 0	0 1	1 1	1 0
0	m_0 **1**	m_1 **1**	m_3 **1**	m_2 **1**
1	m_4 **1**	m_5 **1**	m_7 **1**	m_6 **1**

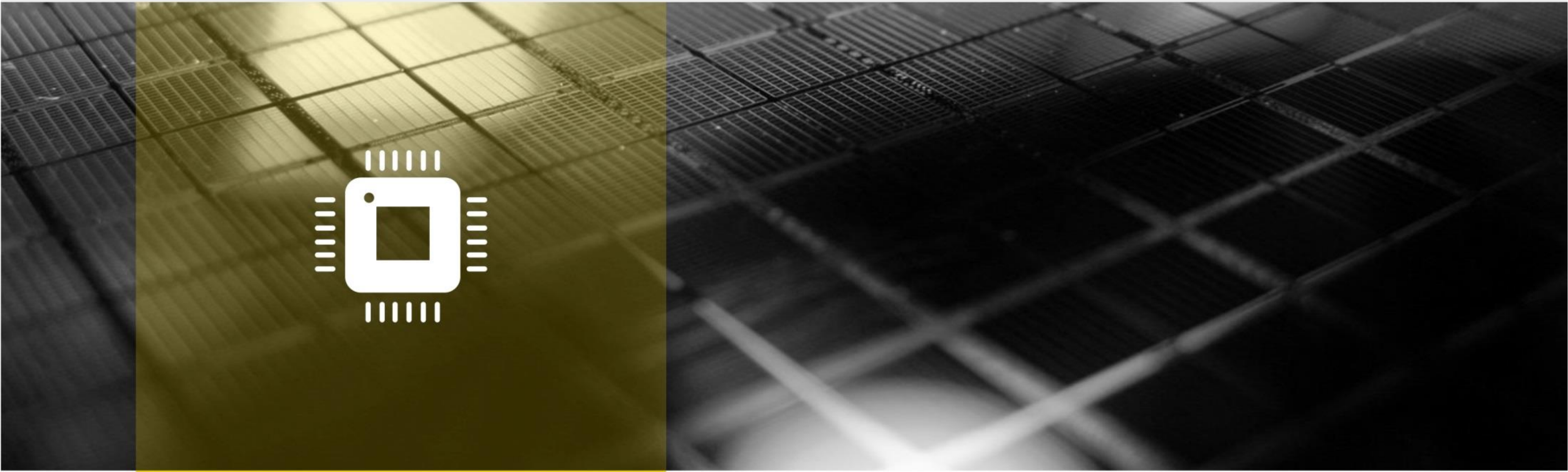
$T_{A0} = 1$

DESIGN PROCEDURE

- Synthesis Using T Flip – Flops: 3-bit Counter.
  - $T_{A2} = A_1A_0$
  - $T_{A1} = A_0$
  - $T_{A0} = 1$

- Logic Diagram of 3-bit Binary Counter





THANK YOU!