Sequential Logic Part 1

Dr. Bassem A. Abdullah

Computer and Systems Department

Lectures Prepared by Dr.Mona Safar, Edited and Lectured by Dr.Bassem A. Abdullah

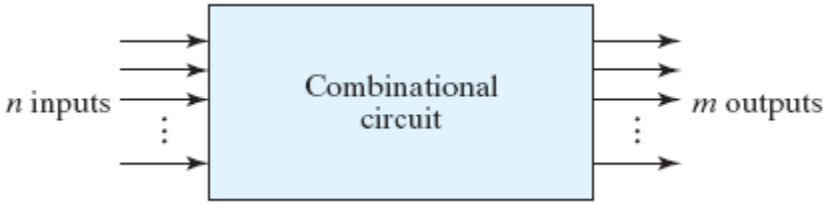
Outline

1. Sequential Logic
2. Latches
3. Flip-Flops
4. Analysis of clocked sequential circuits
5. State Reduction and assignment
6. Design Procedure

Logic Circuits: Combinational versus Sequential Circuits

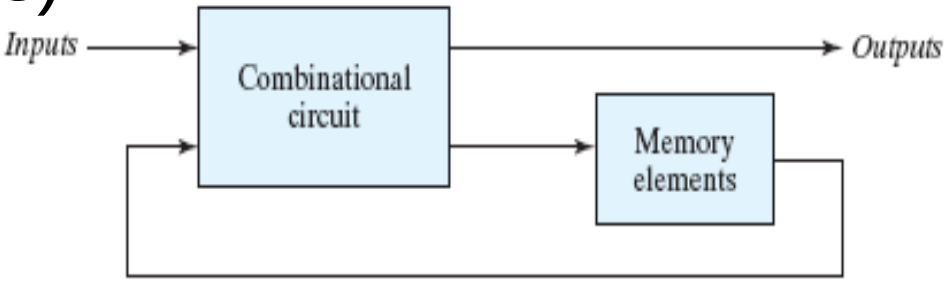
- Combinational Circuits:

  Output only depends on the present combination of inputs

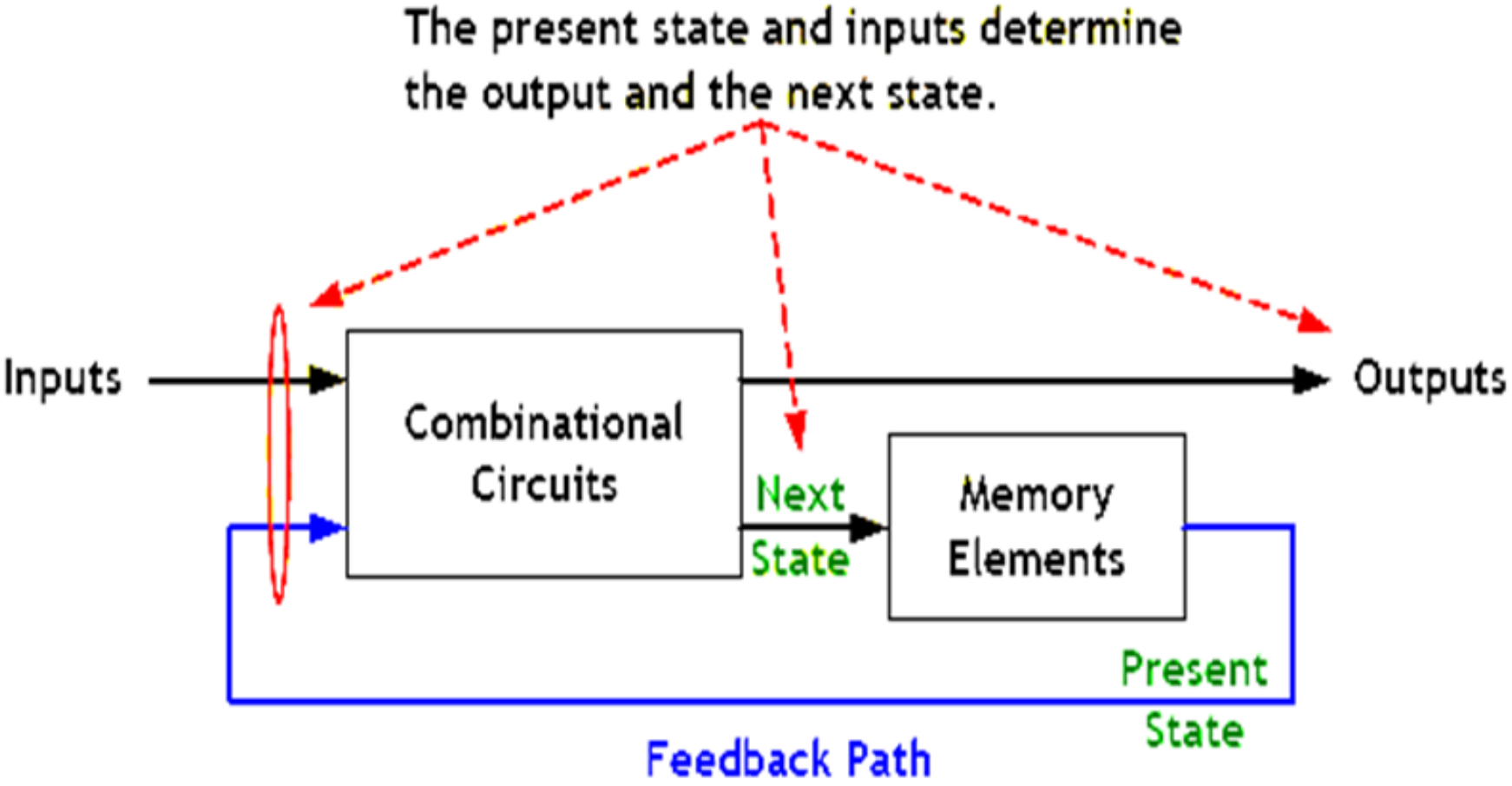


- Sequential Circuits:

  Output depends on the input and the state of the storage (past inputs)



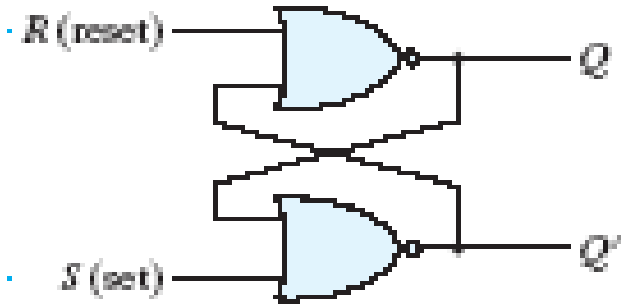
Block Diagram of Sequential Circuits



Latch / Flip-Flop

- Memory elements: Latches/Flip-Flop
- Latch / Flip-Flop can maintain a binary state indefinitely until directed by an input signal to change the state.

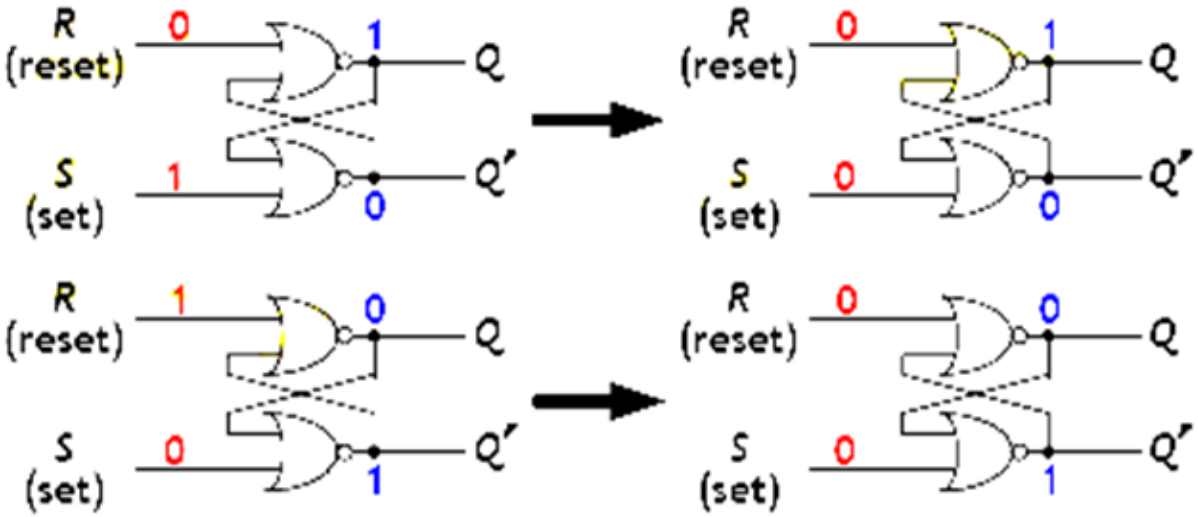
SR Latch with NOR Gates



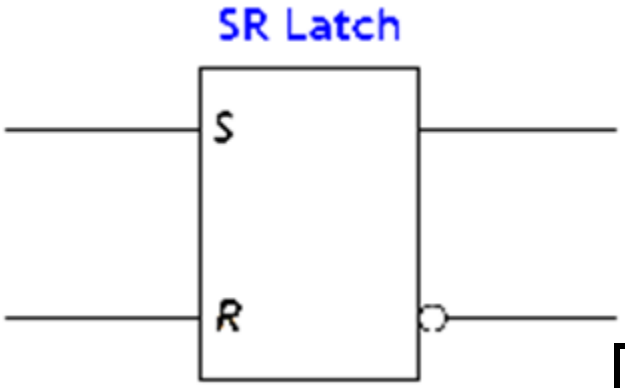
(a) Logic diagram

S	R	Q	Q'	
1	0	1	0	
0	0	1	0	(after $S = 1, R = 0$)
0	1	0	1	
0	0	0	1	(after $S = 0, R = 1$)
1	1	0	0	(forbidden)

(b) Function table



SR Latch Characteristic table



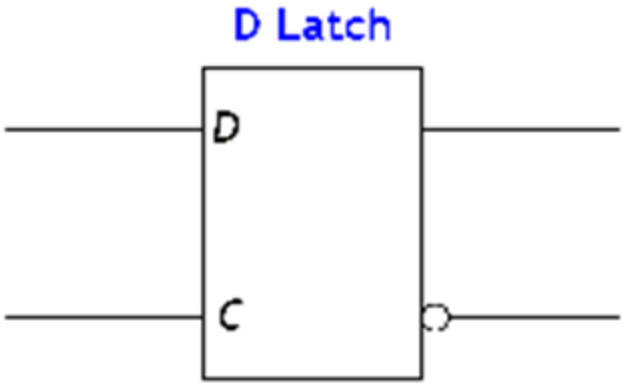
S	R	Q	Q'	
0	0	No change		
0	1	0	1	Reset
1	0	1	0	Set
1	1	0	0	Indeterminate

SR Latch With Control Input Characteristic table

C	S	R	Q	Q’	
0	x	x	No change		
1	0	0	No change		
1	0	1	0	1	Reset
1	1	0	1	0	Set
1	1	1	0	0	Indeterminate

D Latch

- Eliminate the undesirable of the indeterminate state in the SR latch



C	D	Q	Q’	
0	X	No change		
1	0	0	1	Reset
0	1	1	0	Set

Latches Versus Flip Flops

- **Trigger:** The response to the change of input control signal to switch state of a latch or flip-flop
- Latch: Level Trigger
- Flip-Flop: Edge Trigger

Positive level trigger

(a) Response to positive level

Positive level trigger

(b) Positive-edge response

Negative level trigger

(c) Negative-edge response

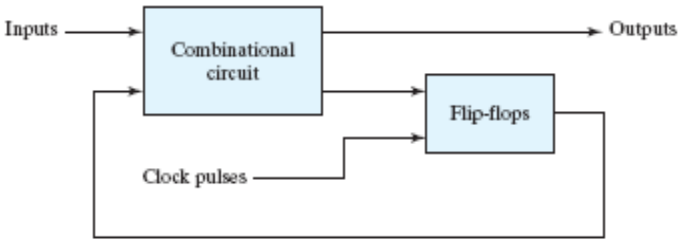
Classifications of Sequential Circuits

- Asynchronous Sequential Circuit:

The behavior depends on the input signals at any time instant and the order in which input changes. **(Chapter 9)**

- Synchronous Sequential Circuit:

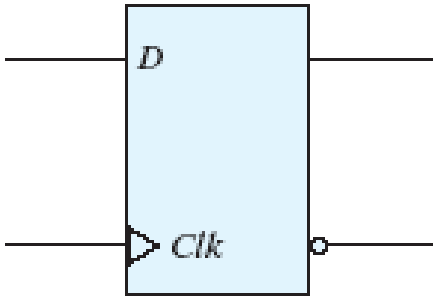
The behavior is defined at discrete instants of time.



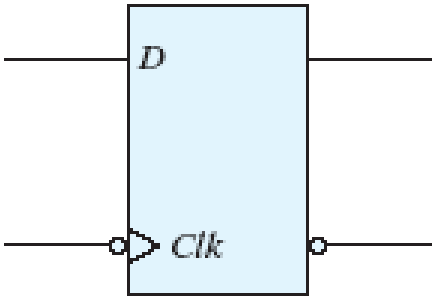
(a) Block diagram

(b) Timing diagram of clock pulses

D Flip-Flop



(a) Positive-edge

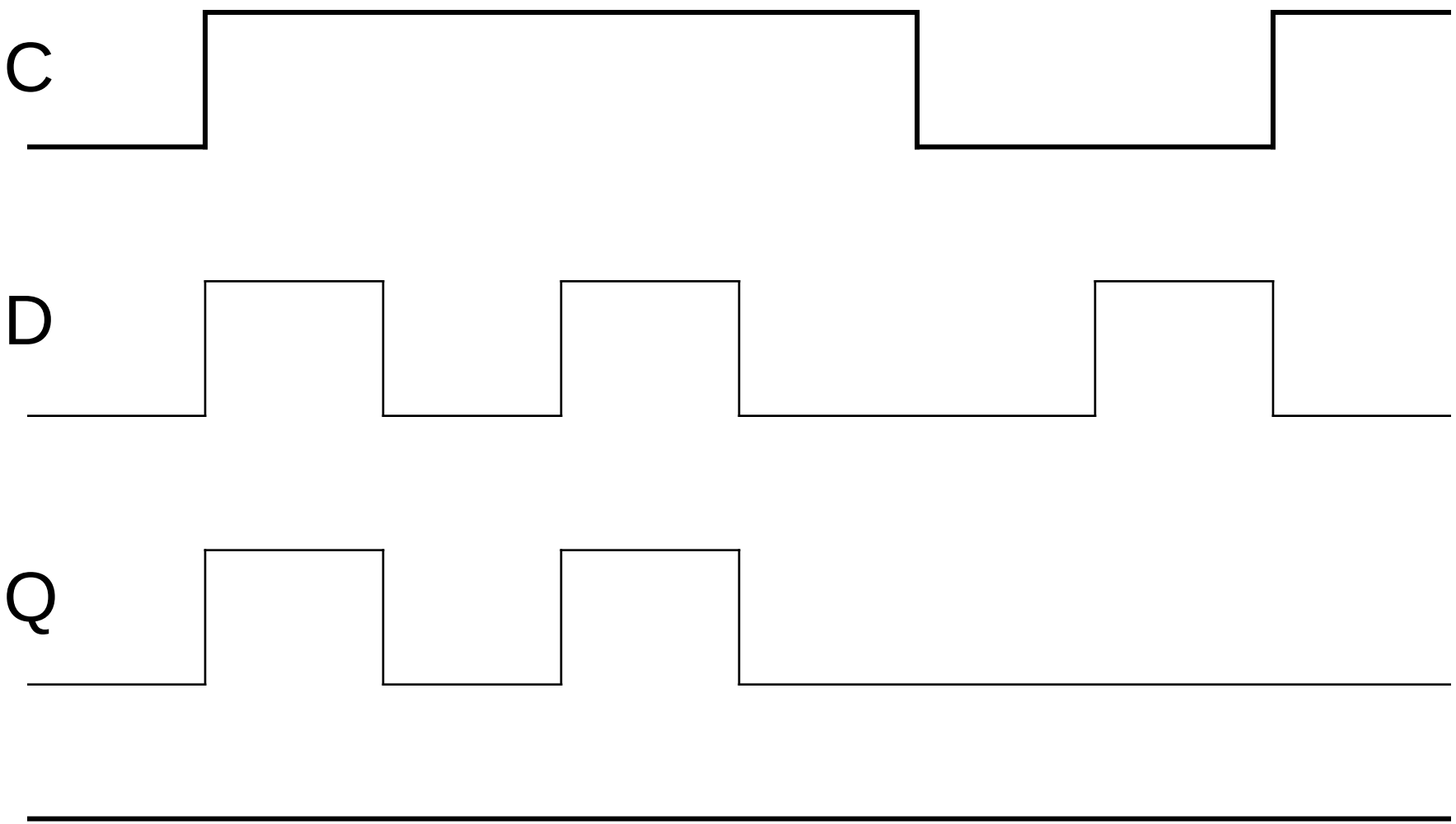


(a) Negative-edge

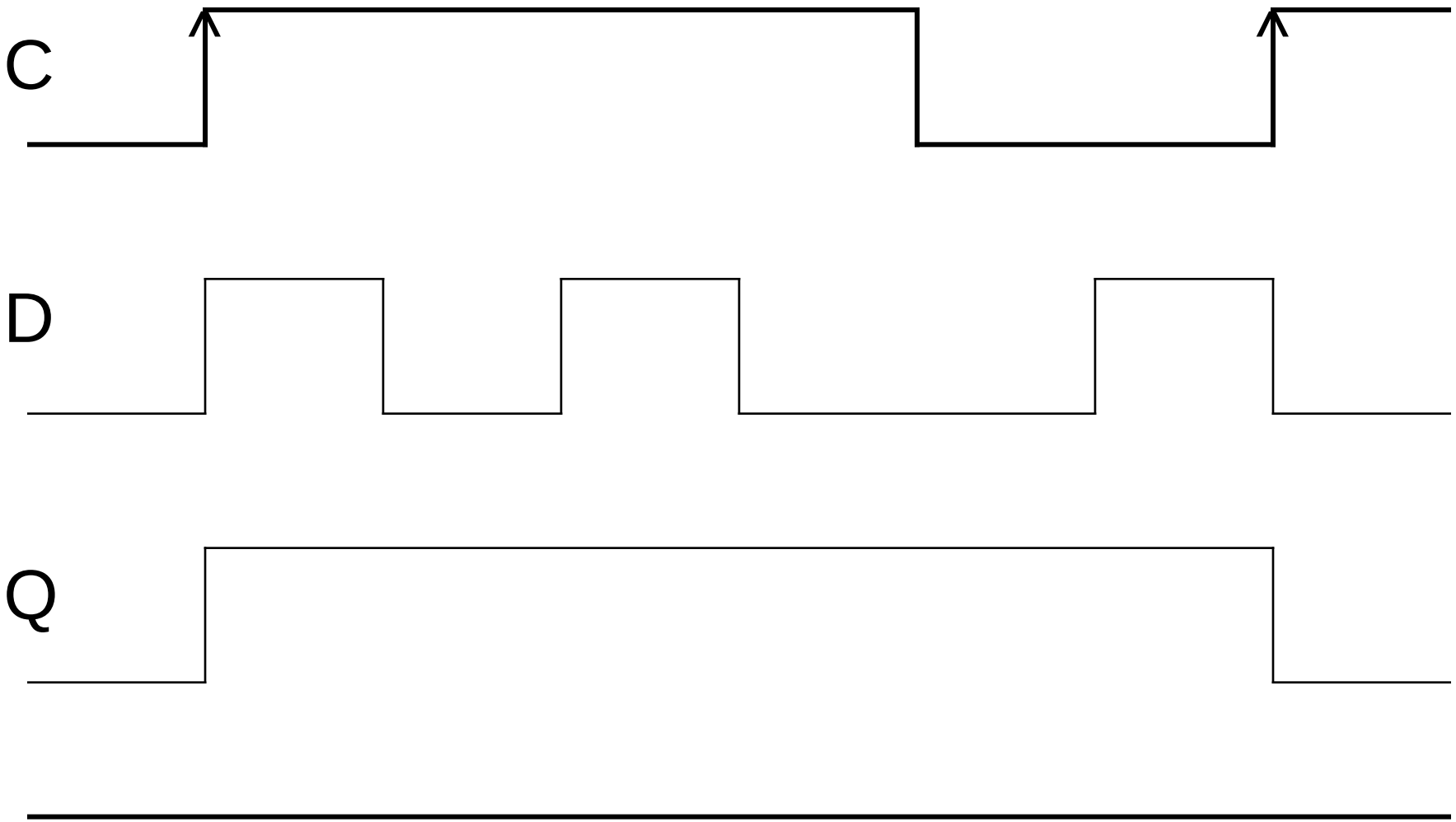
D Flip-Flop

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

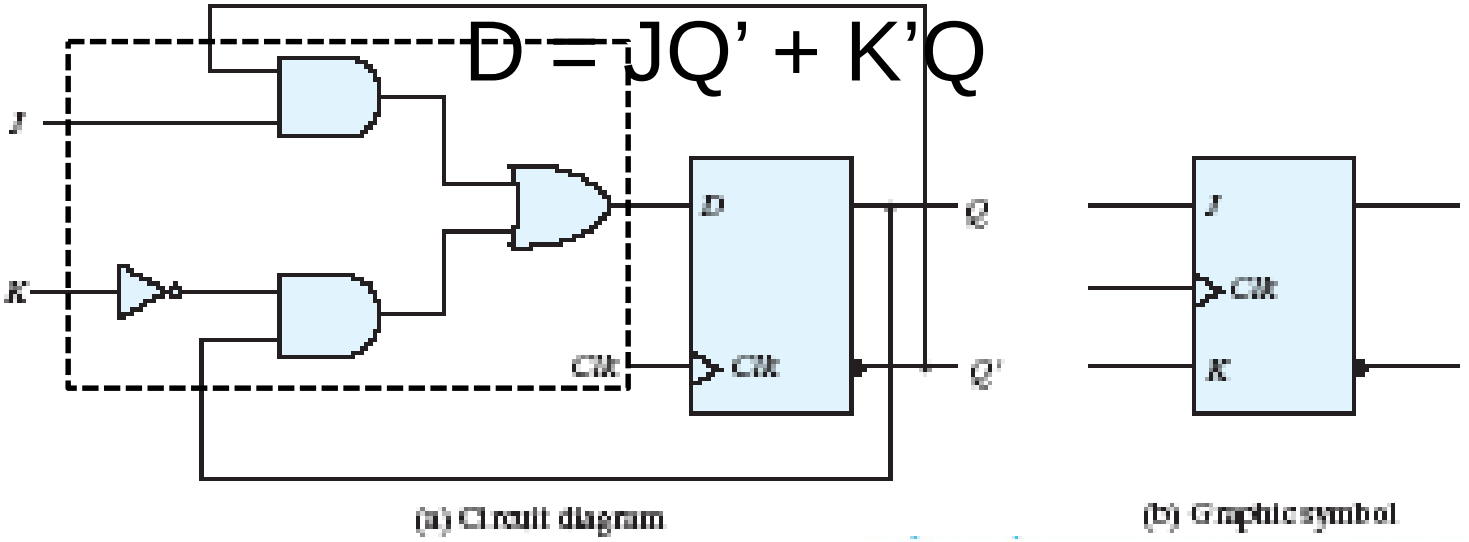
D Latch



D Flip-Flop (Positive Edge Triggered)



JK Flip-Flop

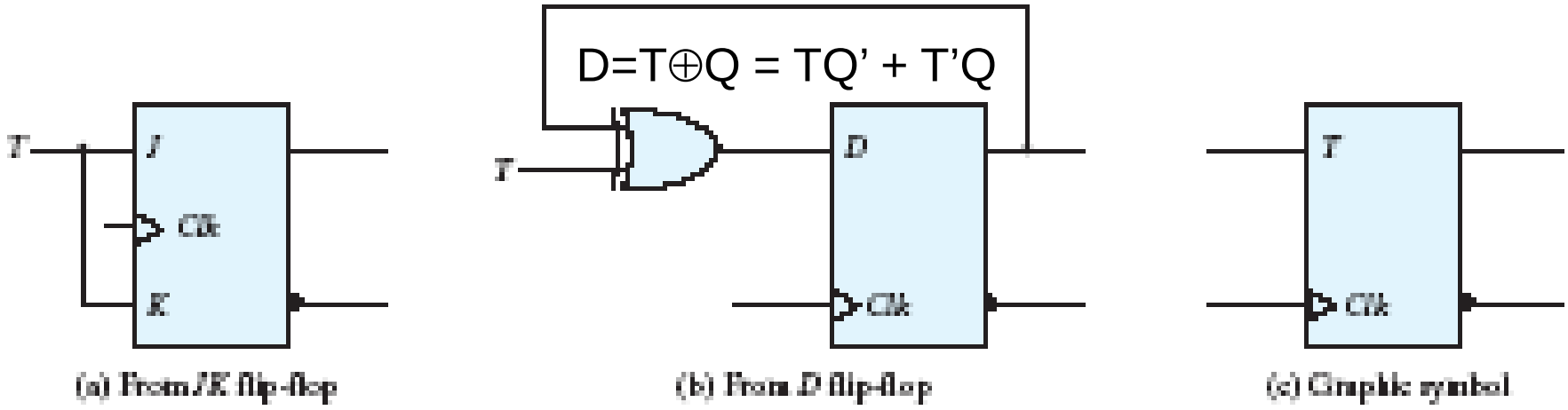


(a) Circuit diagram (b) Graphic symbol

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 (Toggle) Flip-Flop



(a) From *JK* flip-flop (b) From *D* flip-flop (c) Graphic symbol

T Flip-Flop

T	Q(t + 1)	
0	$Q(t)$	No change
1	$Q'(t)$	Complement

Characteristic Equations

- The logical properties of a flip-flop as described in the characteristic table can be expressed also algebraically with a characteristic equation

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

JK FF: $Q(t+1) = JQ' + K'Q$

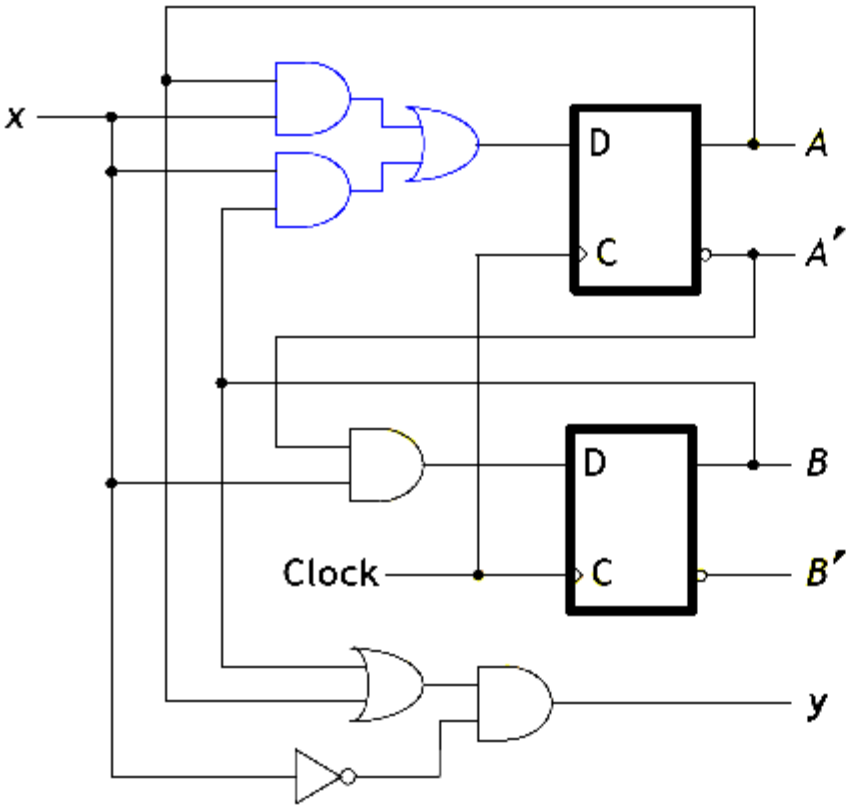
T FF: $Q(t+1) = T \oplus Q = TQ' + T'Q$

Analysis of Clocked Sequential Circuits

- State Table (Transition Table)
  - Consists of:
    1. Present State
    2. Inputs
    3. Next State
    4. Outputs

Example of State Table (Transition Table)

Example → **State Table**



Depend on Circuit

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

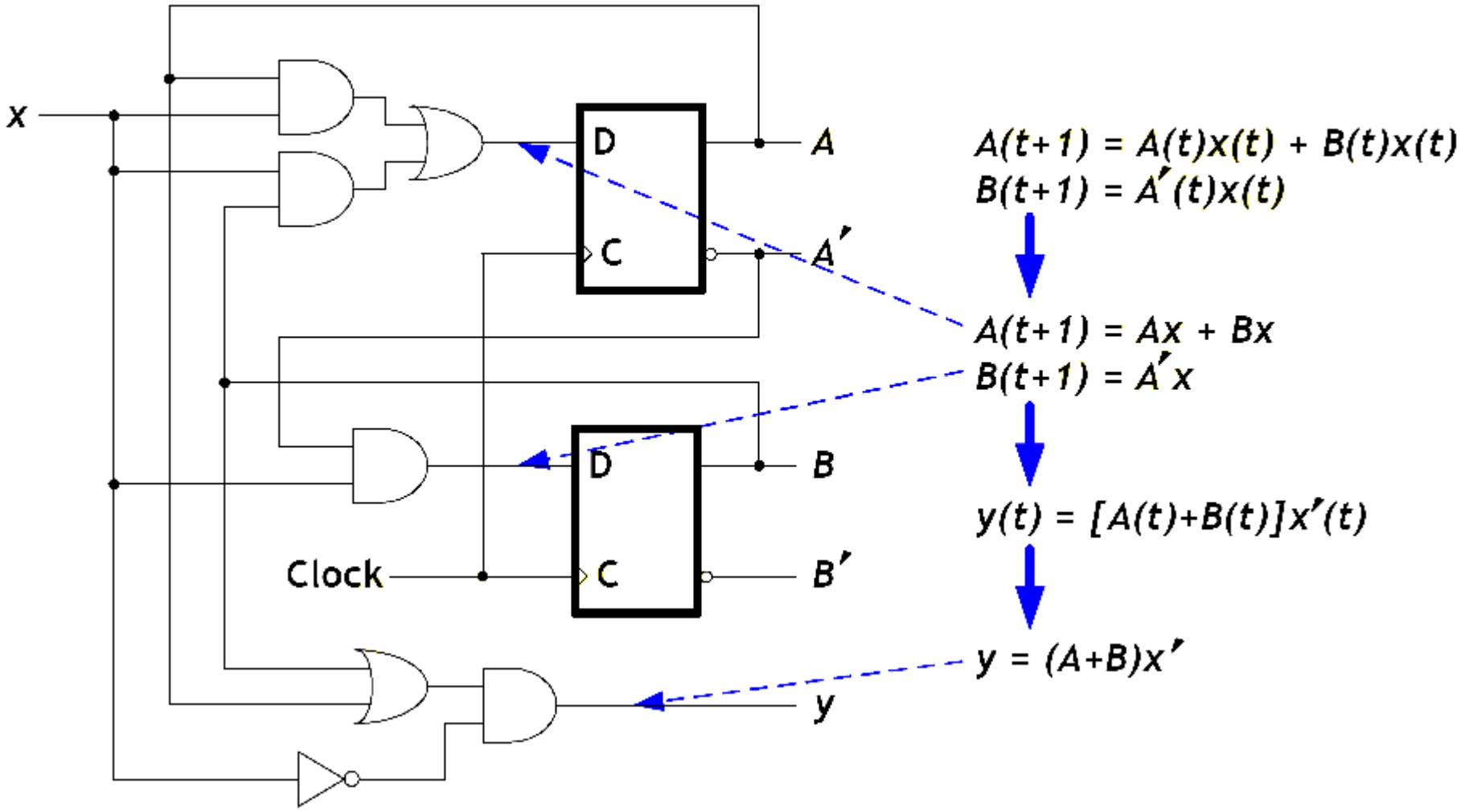
2^3 Items

Analysis of Clocked Sequential Circuits

- **State Equation (Transition Equation):**

A state equation (also called transition equation) specified the next state as a function of the present state and inputs.

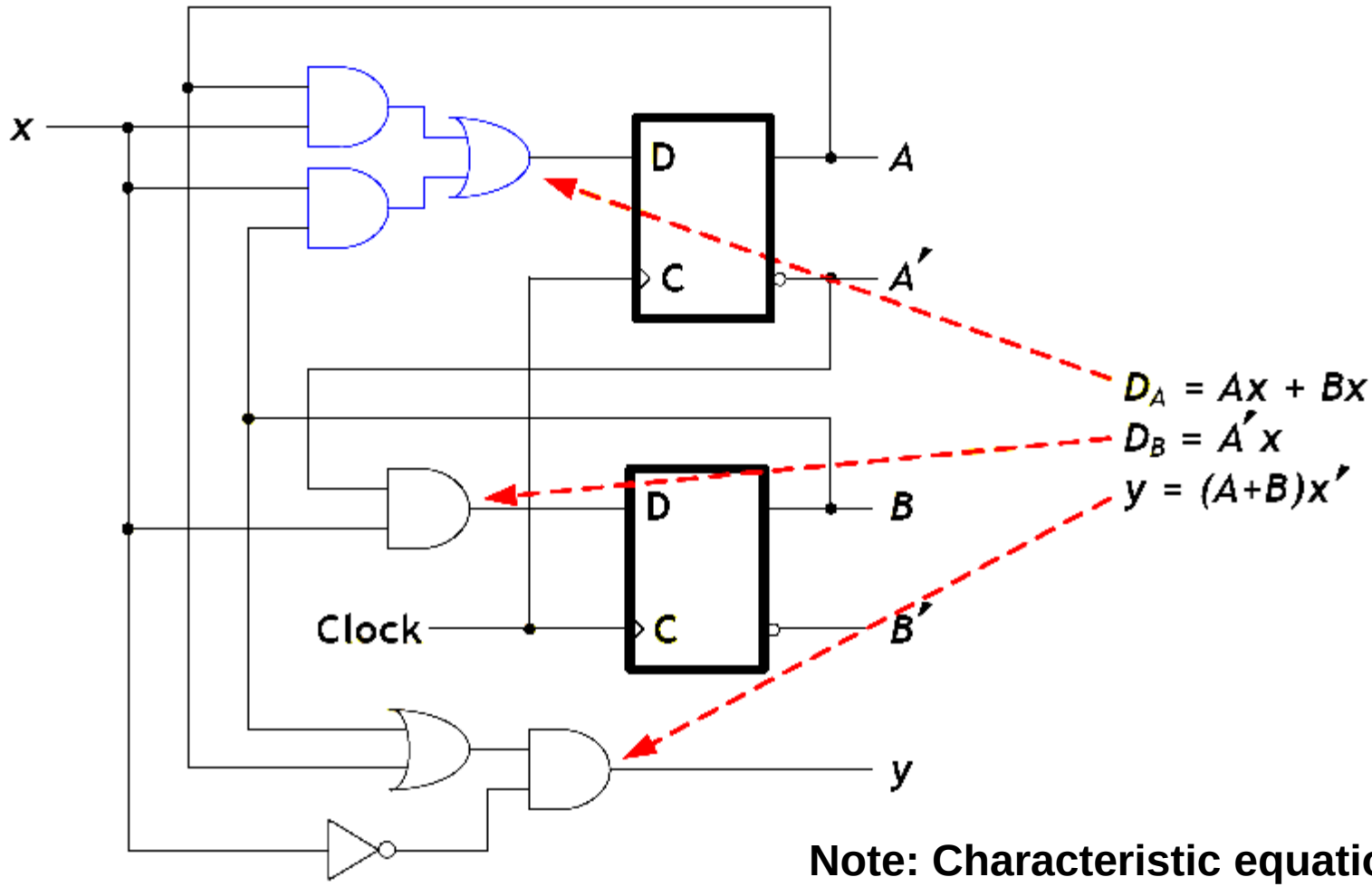
Example of Sequential Circuit



Flip-Flop Input Equations

1. The logic diagram of a sequential circuit consists of flip-flops and gates.
2. The part of combination circuit that generates external outputs is described algebraically by a set of Boolean functions called output equations.
3. The part of the circuit that generates the inputs to flip-flops is described algebraically by a set of Boolean functions called flip-flop input equations (sometimes called excitation equations).
4. The designer adopts the convention of using the flip-flop input symbol to denote the input variable and subscript to designate the name of the flip-flop outputs.

Example of Flip-Flop Input Equations

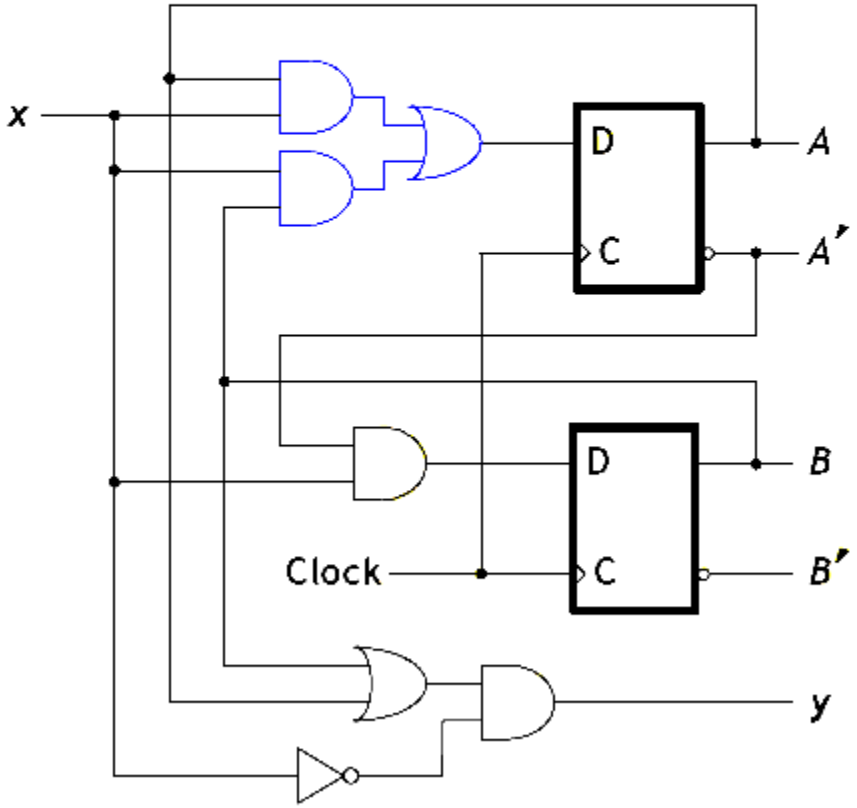


Note: Characteristic equation

$D_A = A(t+1)$

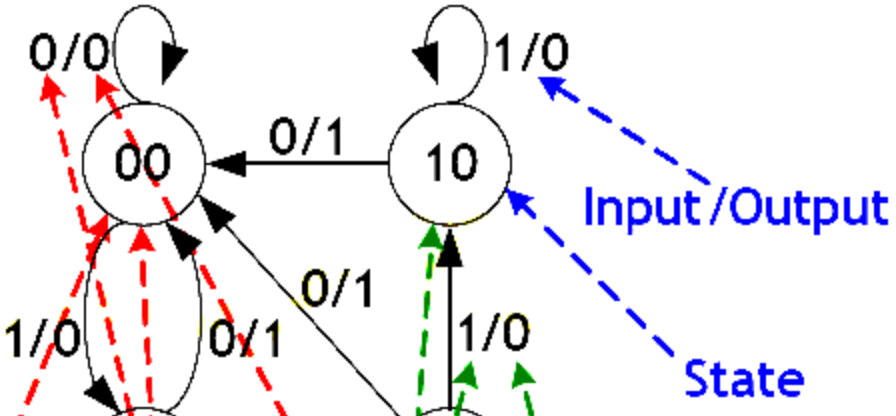
Example of State Table (Transition Table)

Example → Second Form of the State Table



Present State	Next State		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

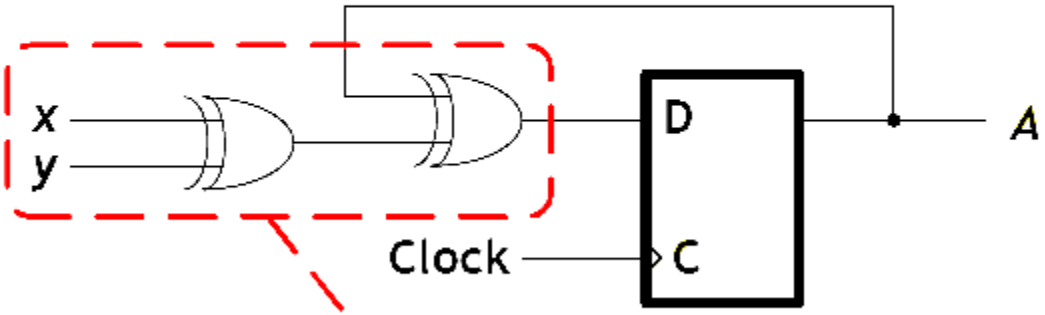
- State Diagram



Present State	Next State		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 with D Flip-Flops

Circuit Diagram

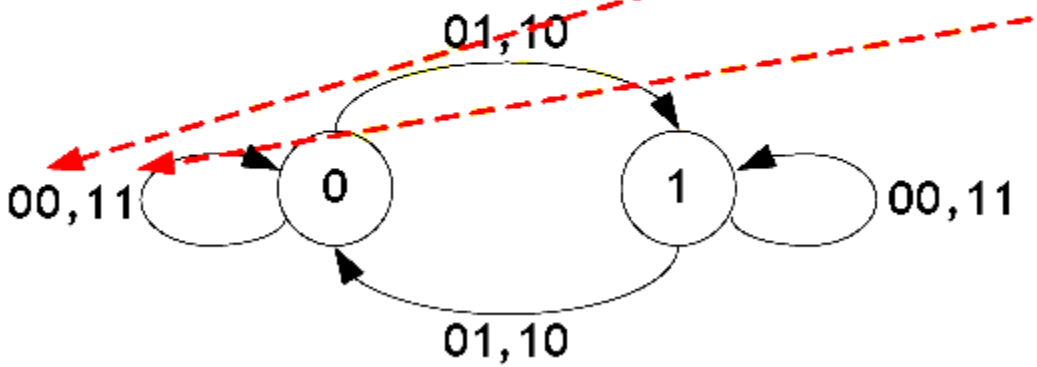


$A(t+1) = A \oplus x \oplus y$

State Table

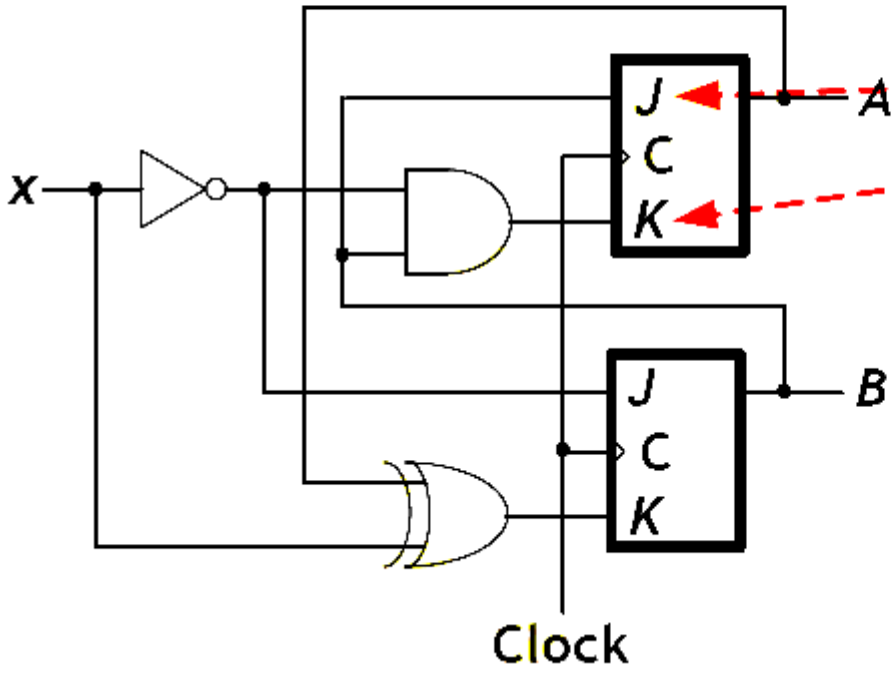
Present State	Inputs	Next State
A	x y	A
0	00	0
0	01	1
0	10	1
0	11	0
1	00	1
1	01	0
1	10	0
1	11	1

State Diagram



Analysis with JK Flip-Flops

Sequential Circuit with *JK* Flip-Flop



Flip=Flop Input Equations

$$J_A = B$$
$$K_A = Bx'$$

$$J_B = x'$$
$$K_B = A'x + Ax' = A \oplus x$$

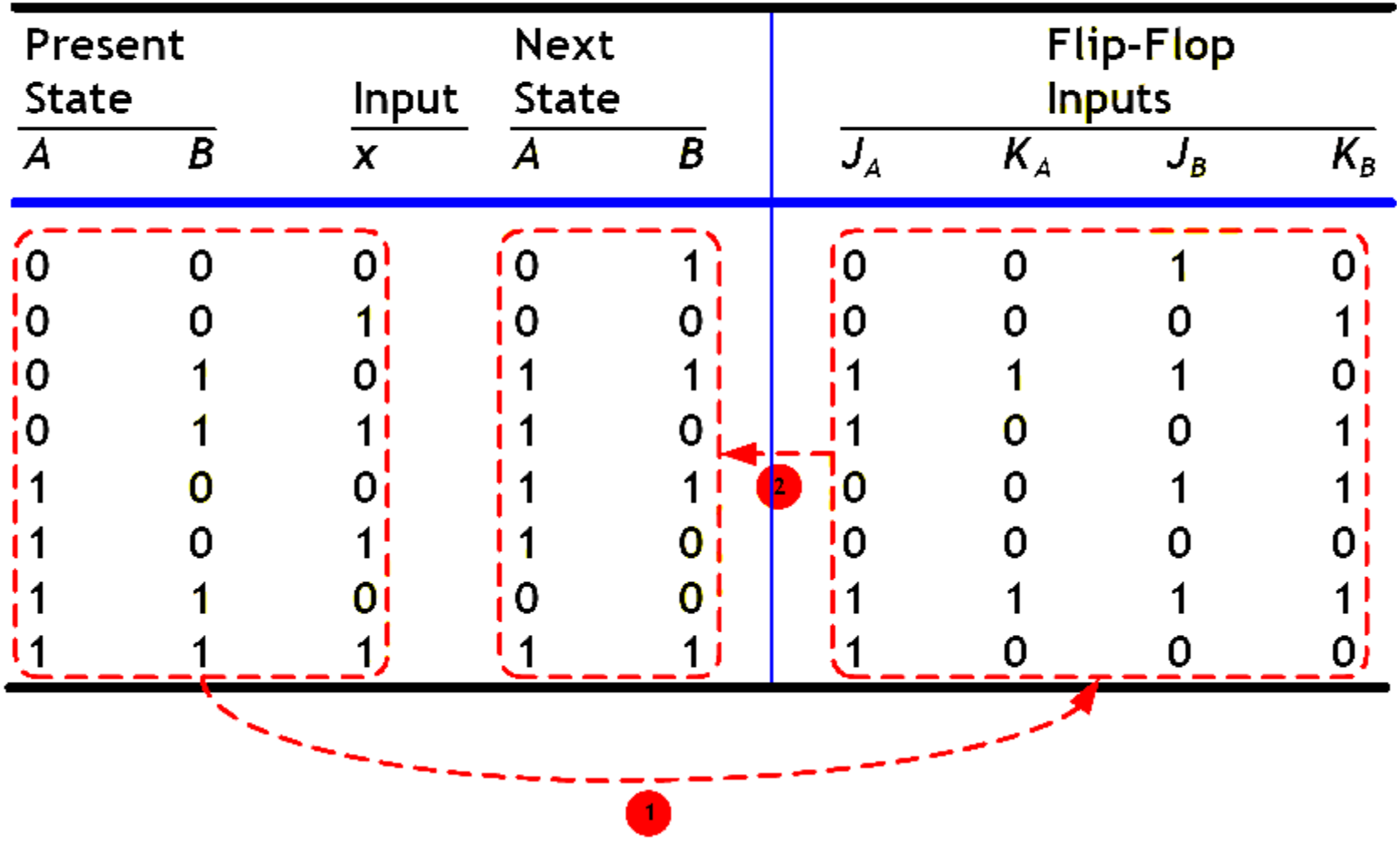
- The next-state can be derived by following procedure

1. Determine the flip-flop equations in terms of the present state and input variables
2. List the binary values of each input equation
3. Use the corresponding flip-flop characteristic table to determine the next state values in the state table

Analysis with JK Flip-Flops (Continued)

Sequential Circuit with *JK* Flip-Flop

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



- The next-state can be obtained by evaluating the state equation from the characteristic equation

1. Determine the flip-flop equations in terms of the present state and input variables
2. Substitute the input equations into the flip-flop characteristic equation to obtain the state equation
3. Use the corresponding state equations to determine the next state values in the state table

Analysis with JK Flip-Flops (Continued)

JK Flip-Flop Characteristic Equations

$$\begin{cases} A(t+1) = JA' + K'A \\ B(t+1) = JB' + K'B \end{cases}$$

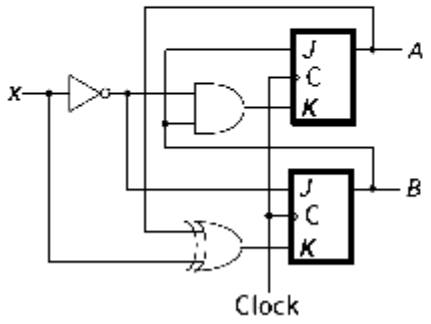
Flip-Flop Input Equations

$$\begin{cases} J_A = B \\ K_A = Bx' \end{cases}$$

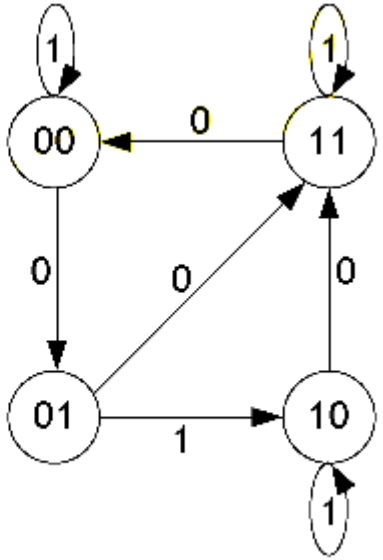
$$\begin{cases} A(t+1) = BA' + (Bx)'A = A'B + AB' + Ax \\ B(t+1) = x'B' + (A \oplus x')B = B'x' + ABx + A'Bx' \end{cases}$$

$$\begin{cases} J_B = x' \\ K_B = A'x + Ax' = A \oplus x \end{cases}$$

Sequential Circuit with *JK* Flip-Flop

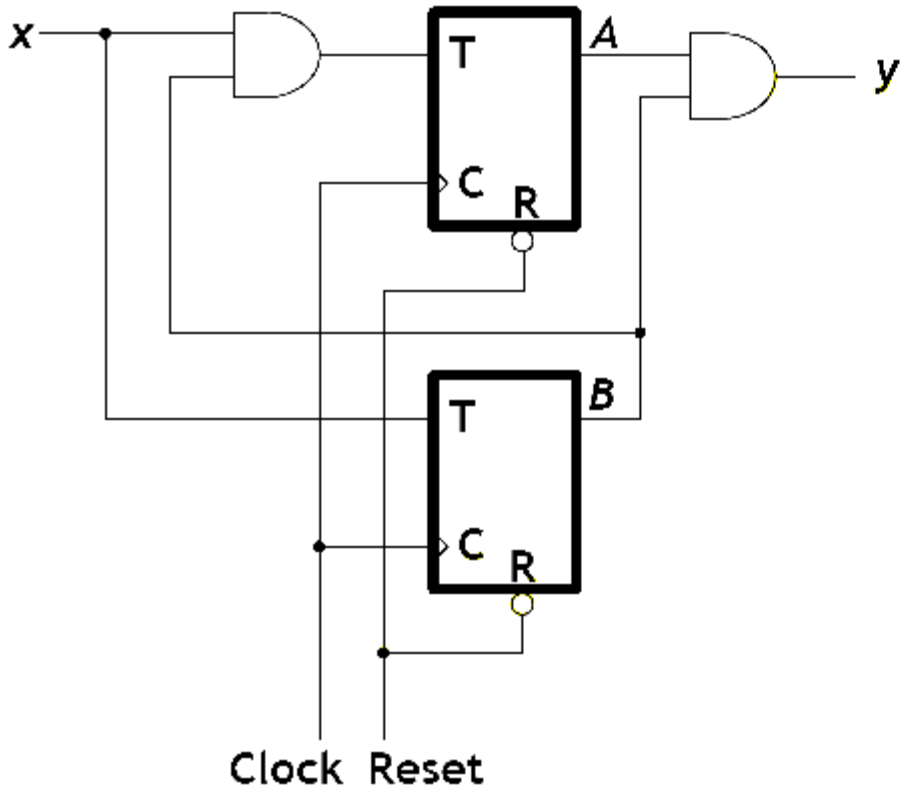


State Diagram



Analysis with T Flip-Flops

Sequential Circuit with T FF



Characteristic Equation

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

$$T_A = Bx$$
$$T_B = x$$
$$y = AB$$

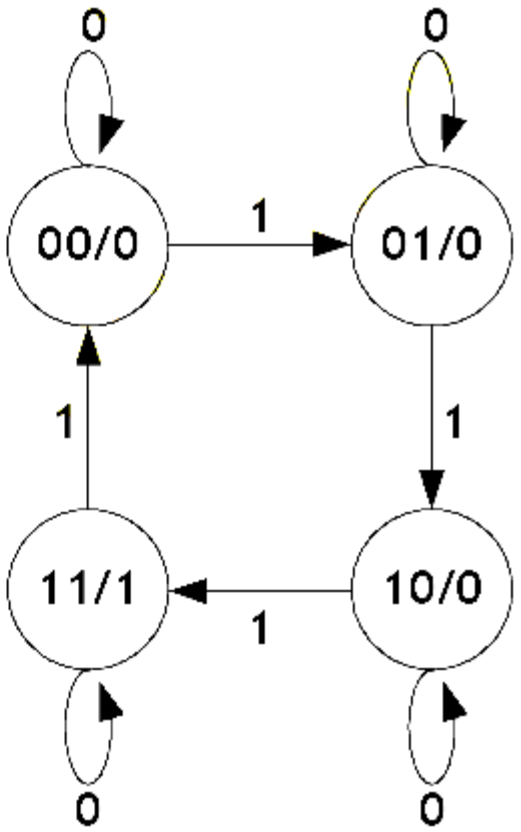
$$A(t+1) = (Bx)'A + (Bx)A' = AB' + Ax' + A'Bx$$
$$B(t+1) = x \oplus B$$

Analysis with T Flip-Flops (Continued)

State Table

Present State AB	Input x	Next State AB	Output y
00	0	00	0
00	1	01	0
01	0	01	0
01	1	10	0
10	0	10	0
10	1	11	0
11	0	11	1
11	1	00	1

State Diagram



Mealy and Moore Models

Model Classification of Sequential Circuits

Mealy Model (Mealy FSM, Mealy Machine):

The output is a function of both the present state and input.

Moore Model (Moore FSM, Moore Machine):

The output is a function of present state only.

The output are synchronous with the clock.

Mealy FSM

- State Diagram

Present State	Next State		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

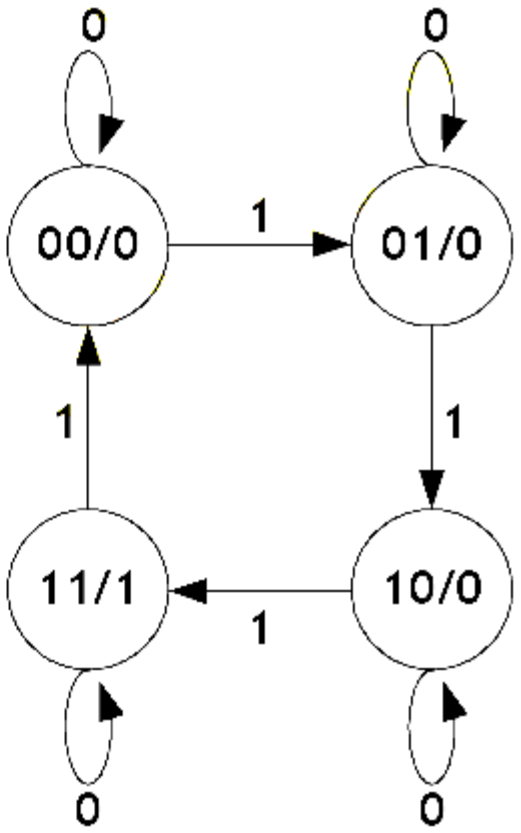
Analysis with T Flip-Flops (Continued)

Moore FSM

State Table

Present State AB	Input x	Next State AB	Output y
00	0	00	0
00	1	01	0
01	0	01	0
01	1	10	0
10	0	10	0
10	1	11	0
11	0	11	1
11	1	00	1

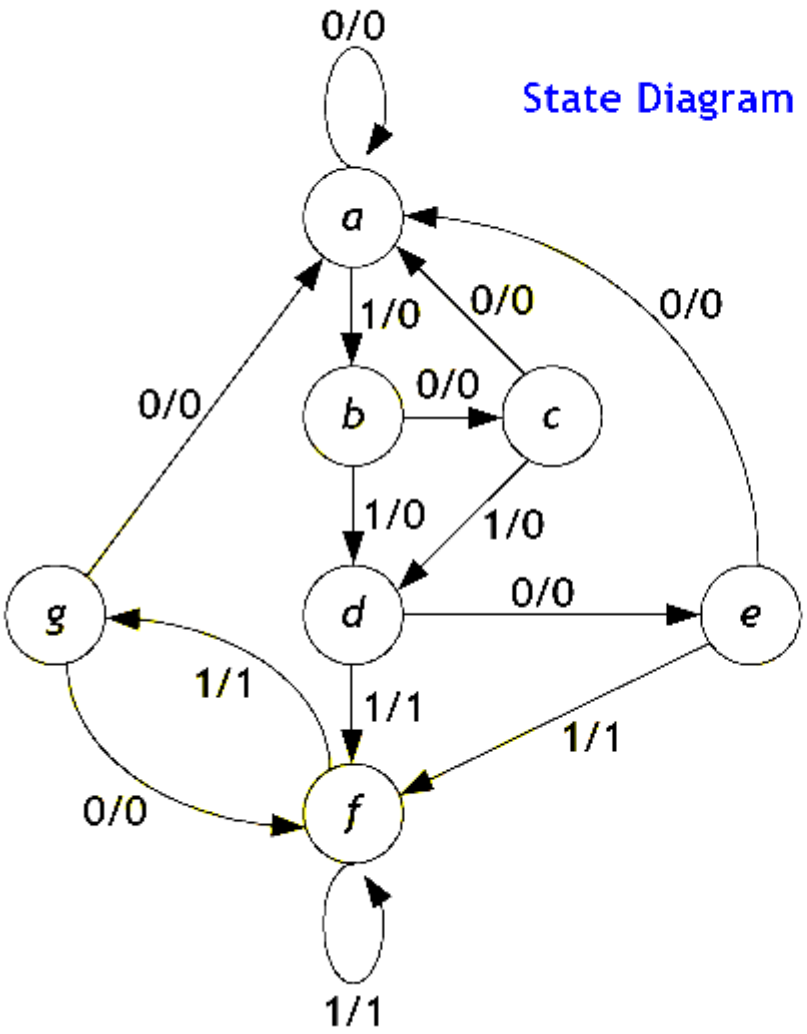
State Diagram



State Reduction of Sequential Circuits

1. State Reduction: Lower the requirement of flip-flop
2. *m* flip-flop produce 2^m states
3. The characteristic of a sequential circuit: input-output sequences (not the internal circuits)

Example of State Reduction



State Diagram

Initial state a, I/O sequence:

State	a	a	b	c	d	e	f	f	g	f	g	a
Input	0	1	0	1	0	1	1	0	1	0	0	
Output	0	0	0	0	0	1	1	0	1	0	0	

Example of State Reduction (Continued)

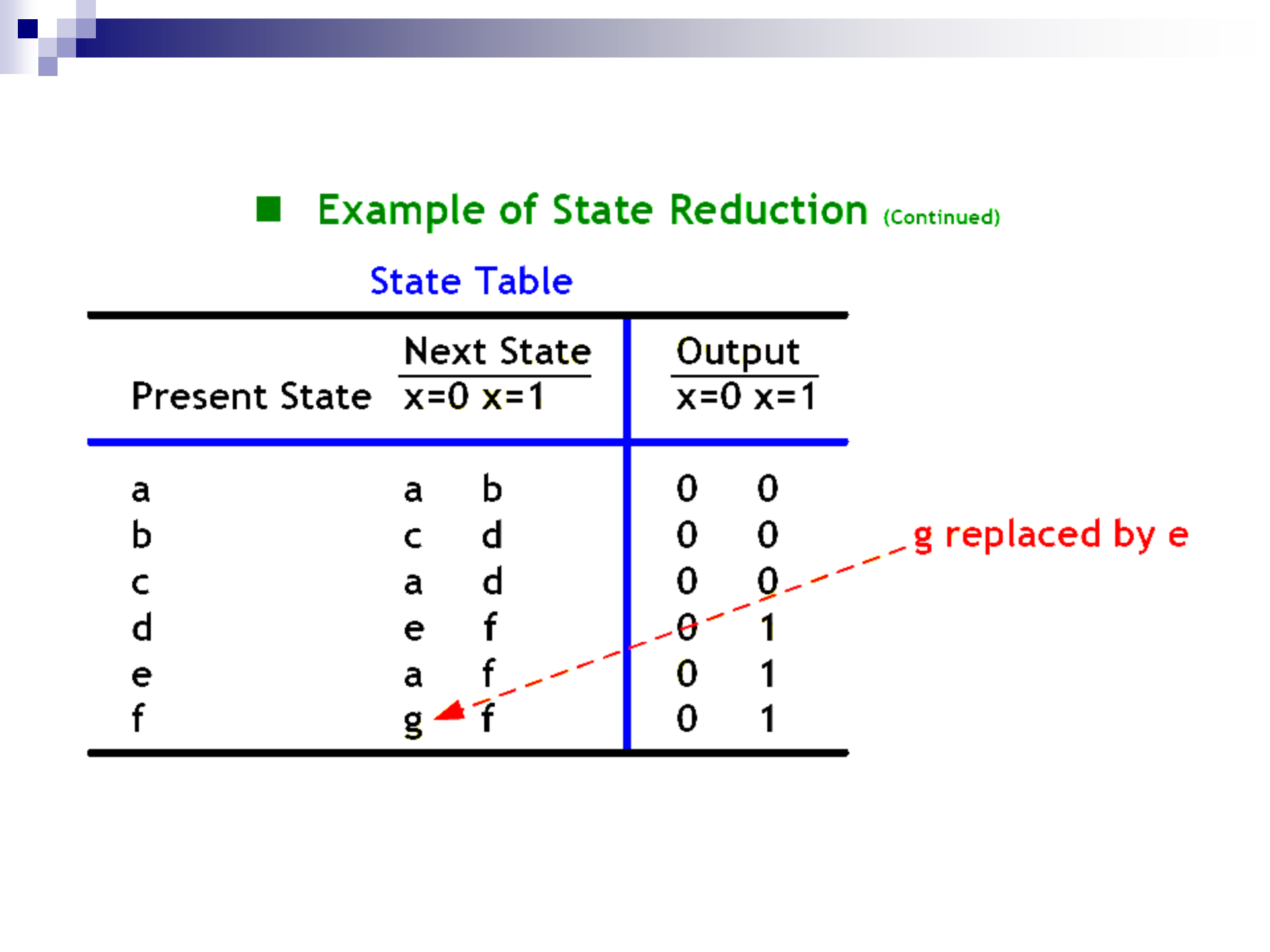
State Table

Present State	Next State x=0	Next State x=1	Output x=0	Output x=1	
a	a	b	0	0	
b	c	d	0	0	
c	a	d	0	0	
d	e	f	0	1	
e	a	f	0	1	Equivalent State
f	g	f	0	1	
g	a	f	0	1	Removed State

Example of State Reduction (Continued)

State Table

Present State	Next State x=0	Next State x=1	Output x=0	Output x=1
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	g	f	0	1



Example of State Reduction (Continued)

State Table

Present State	Next State x=0	Next State x=1	Output x=0	Output x=1	
a	a	b	0	0	
b	c	d	0	0	
c	a	d	0	0	
d	e	f	0	1	← Equivalent State
e	a	f	0	1	
f	e	f	0	1	← Removed State

Example of State Reduction (Continued)

State Table

Present State	Next State x=0	Next State x=1	Output x=0	Output x=1
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1

f replaced by d

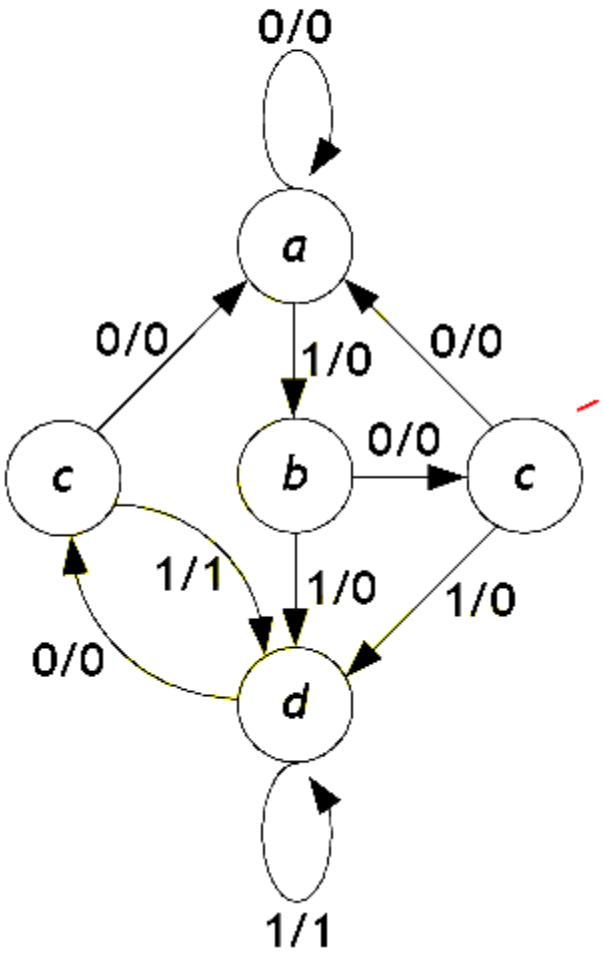
Example of State Reduction (Continued)

State Table

Present State	Next State x=0	Next State x=1	Output x=0	Output x=1
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	d	0	1
e	a	d	0	1

Example of State Reduction (Continued)

State Diagram



Initial state a, I/O sequence:

State	a	a	b	c	d	e	d	d	e	d	e	a
Input	0	1	0	1	0	1	1	0	1	0	0	
Output	0	0	0	0	0	1	1	0	1	0	0	

State Assignment

Possible Binary State Assignments

State	Assignment 1 Binary	Assignment 2 Gray Code	Assignment 3 One-hot
a	000	000	00001
b	001	001	00010
c	010	011	00100
d	011	010	01000
e	111	110	10000

Reduced State Table with Binary Assignment 1

Present State	Next State x=0	Next State x=1	Output x=0	Output x=1
000	000	001	0	0
001	010	011	0	0
010	000	011	0	0
011	100	011	0	1
100	000	011	0	1

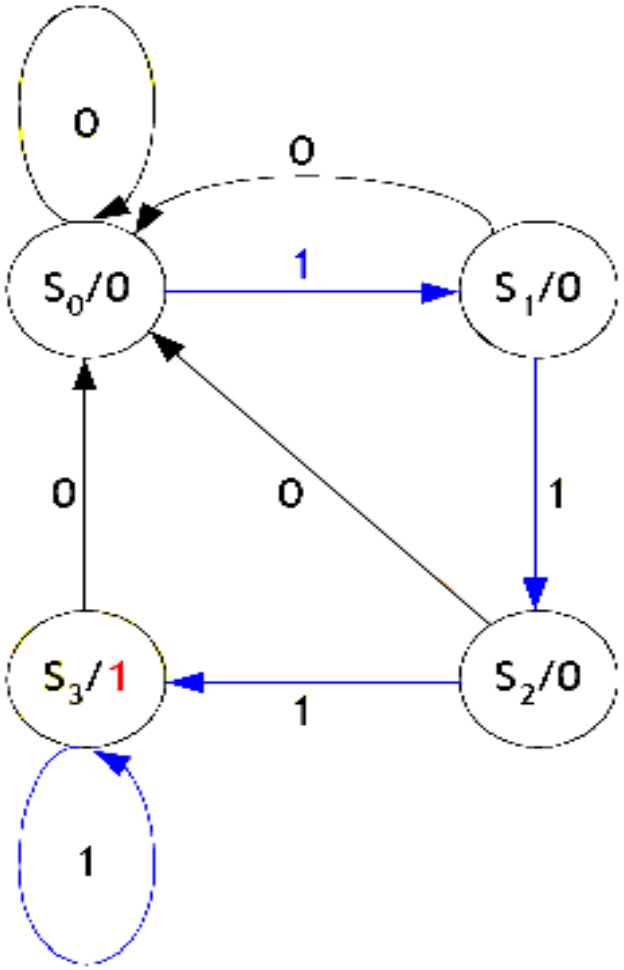
Design Procedure for Synchronous Sequential Circuits

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

Design Example: Problem Description

Design a circuit that detects three or more consecutive 1's in a string of bits coming through an input line.

State Diagram



State Reduction

- No State Reduction is possible for this example

- **Assign Binary Code to the States**

$S_0 \Rightarrow 00\ (AB)$

$S_1 \Rightarrow 01\ (AB)$

$S_2 \Rightarrow 10\ (AB)$

$S_3 \Rightarrow 11\ (AB)$

Synthesis using D flip-flop

- Most easy way since

  Q(t+1)= D

List State Table with D FFs

State Table

	Present State AB	Input x	Next State AB	Output y
m_0	00	0	00	0
m_1	00	1	01	0
m_2	01	0	00	0
m_3	01	1	10	0
m_4	10	0	00	0
m_5	10	1	11	0
m_6	11	0	00	1
m_7	11	1	11	1

- Find the Next State and Output Equations

$$A(t+1) = D_A(A,B,x) = \Sigma(3,5,7)$$

$$B(t+1) = D_B(A,B,x) = \Sigma(1,5,7)$$

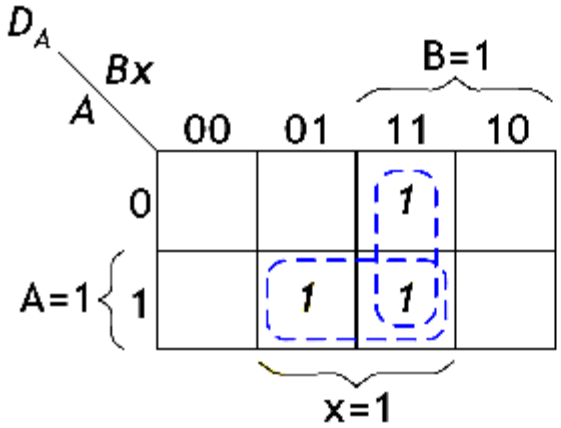
$$y = (A,B,x) = \Sigma(6,7)$$

$$D_A = Ax + Bx$$

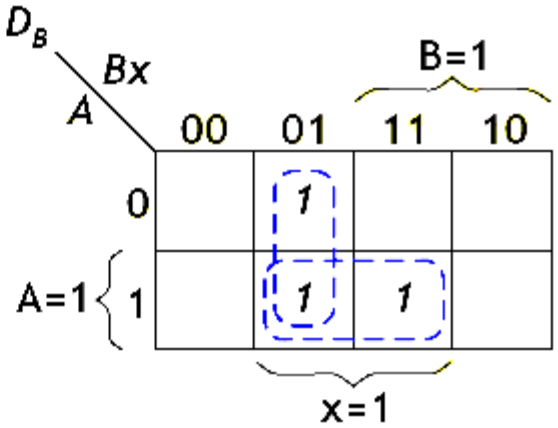
$$D_B = Ax + B'x$$

$$y = AB$$

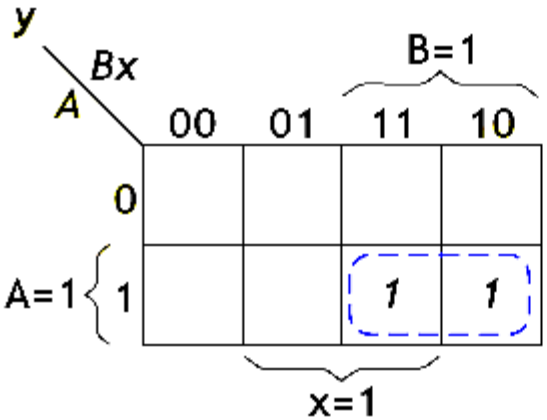
Logic Simplification



$D_A = Ax + Bx$

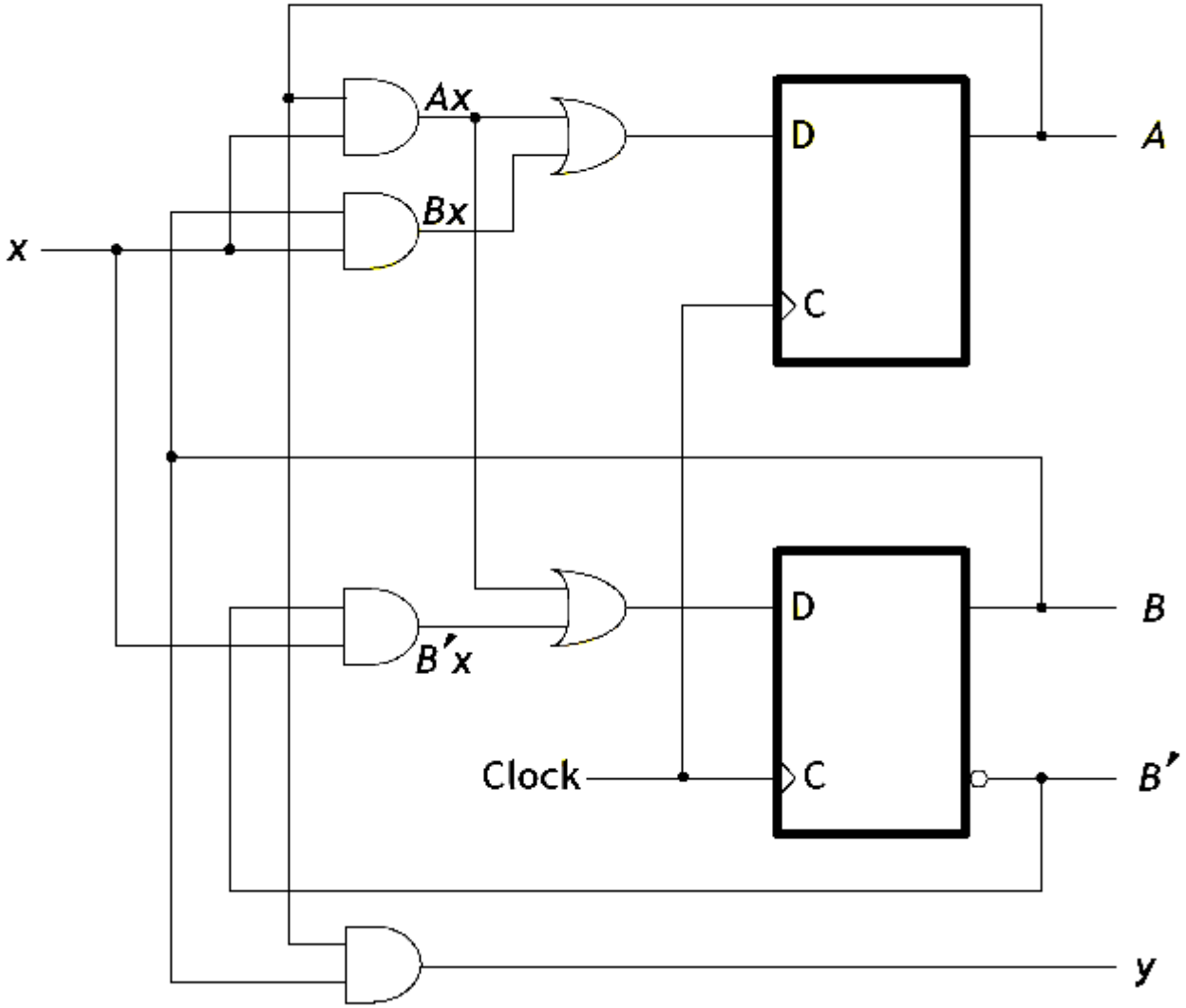


$D_B = Ax + B'x$



$y = AB$

Draw the Logic Diagram of Sequential Circuits



Excitation Tables

A table lists the required inputs for a given change of state. $Q(t) \Rightarrow Q(t+1)$

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

Characteristic Table

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

Excitation Table

Excitation Tables (Continued)

T Flip-Flop

Characteristic Table

T	$Q(t+1)$	
0	$Q(t)$	No Change
1	$Q'(t)$	Complement

Excitation Table

$Q(t)$	$Q(t+1)$	T
0	0	0
0	1	1
1	0	1
1	1	0

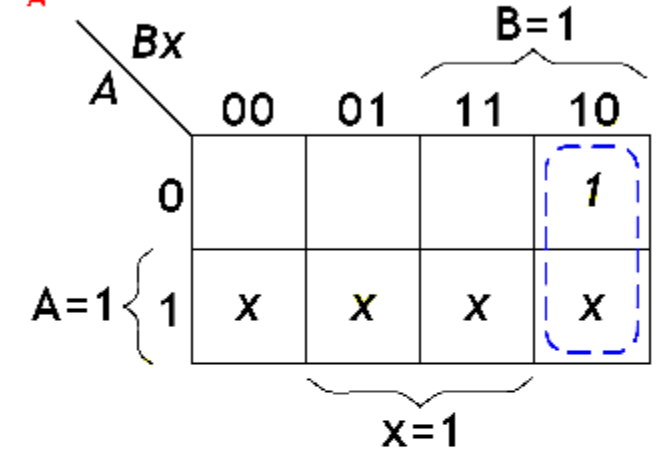
Synthesis Using JK Flip-Flops

State Table

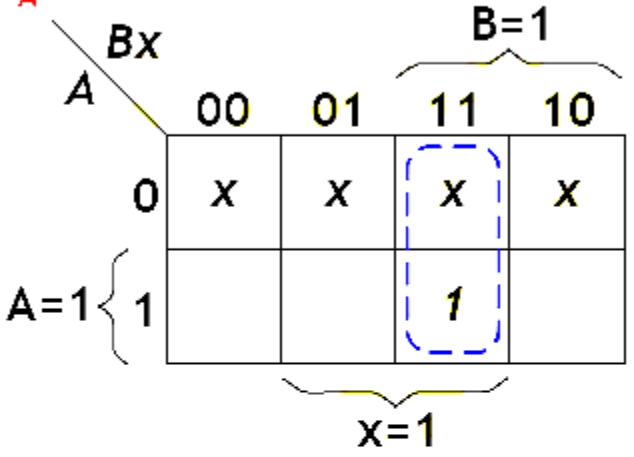
Present State AB	Input x	Next State AB	Flip-Flop Input $J_A\ K_A\ J_B\ K_B$
00	0	00	0 x 0 x
00	1	01	0 x 1 x
01	0	10	1 x x 1
01	1	01	0 x x 0
10	0	10	x 0 0 x
10	1	11	x 0 1 x
11	0	11	x 0 x 0
11	1	00	x 1 x 1

K-Map Logic Simplification

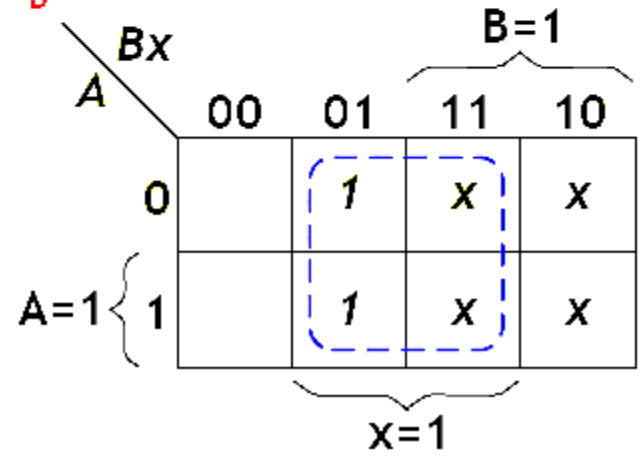
$J_A = Bx'$



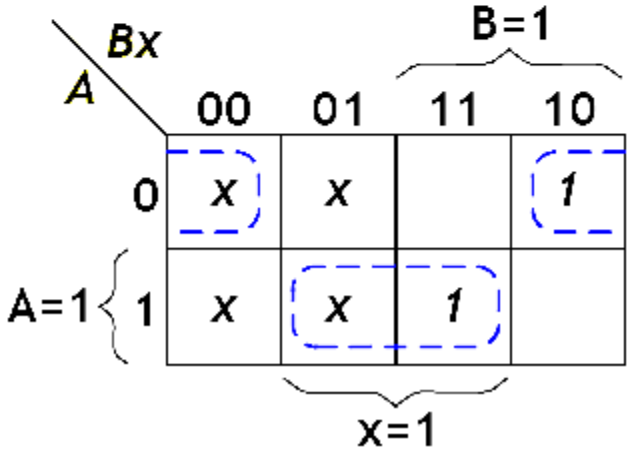
$K_A = Bx$



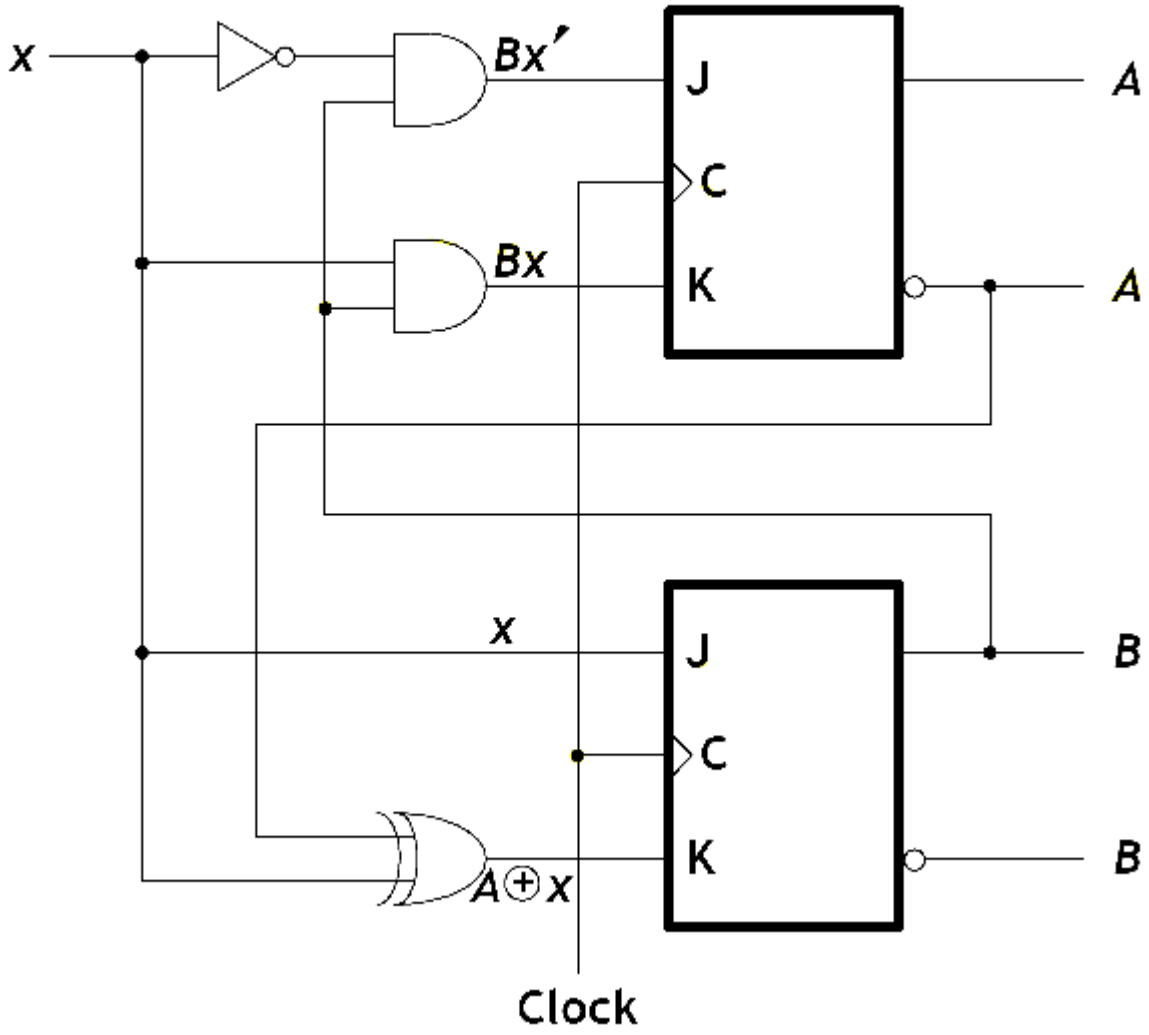
$J_B = x$



$K_B = (A \oplus x)'$



Draw the Sequential Circuits with JK FFs



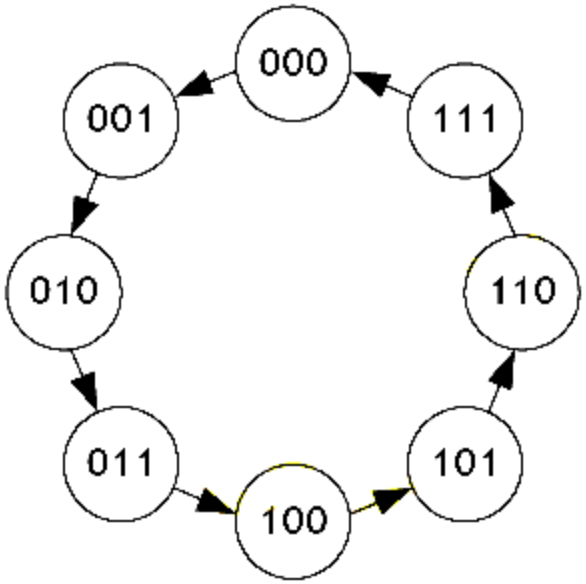
Synthesis Using T Flip-Flops

Design Example: 3-bit Binary Counter Using T FFs

1. N-bit binary counter consists of n flip-flops
2. N-bit binary counter can count 0 to 2^n-1
3. 3-bit counter 0 to 7
4. 3-bit counter internal states 000 to 111

State Diagram and State Table of 3-bit Binary Counter

State Diagram



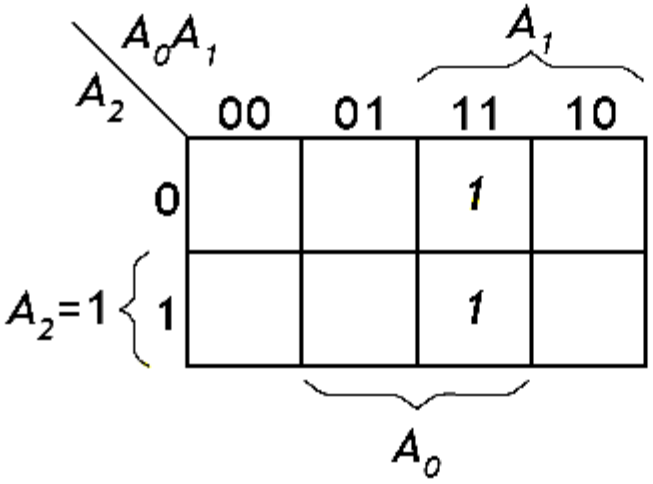
State Table

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

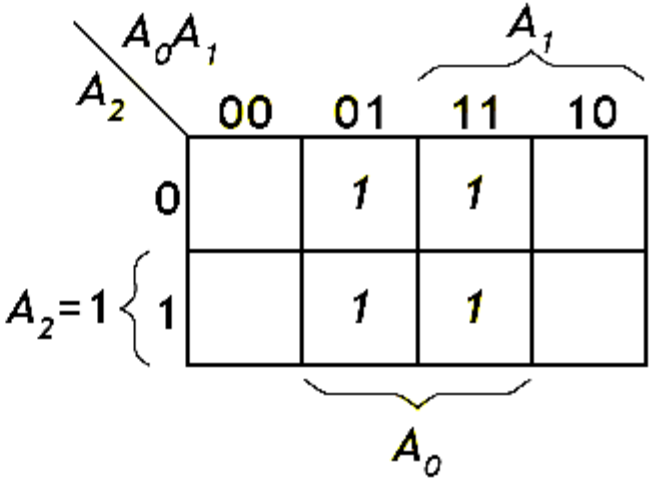
Refer to T-FF Excitation Table

K-Map Logic Simplification for 3-bit Binary Counter

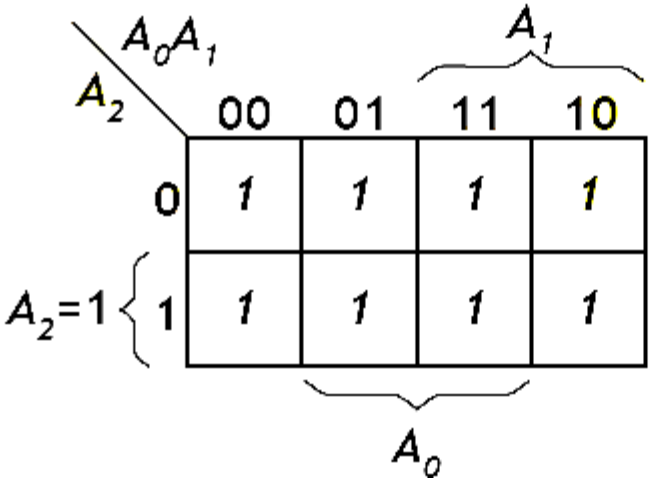
$T_{A2} = A_1A_0$



$T_{A1} = A_0$



$T_{A0} = 1$



- Draw the 3-bit Binary Counter Circuits with T FFs