SNS COLLEGE OF TECHNOLOGY
Coimbatore-35
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A++' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

19ECB231 – DIGITAL ELECTRONICS

II YEAR/ III SEMESTER

UNIT 4 – DESIGN OF SEQUENTIAL CIRCUITS

TOPIC –STATE MINIMIZATION & ASSIGNMENT

State Reduction and Assignment

- The analysis of sequential circuits starts from a circuit diagram and culminates in a state table or diagram.

- The design of a sequential circuit starts from a set of specifications and culminates discusses certain properties of sequential circuits that may be used to reduce the number of gates and flip-flops during the design.

State Reduction

- The reduction of the number of flip-flops in a sequential circuit is referred to as the state-reduction problem. State-reduction algorithms are concerned with procedures for reducing the number of states in a state table, while keeping the external input-output requirements unchanged.

- Since m flip-flops produce 2^m states, a reduction in the number of states may result in a reduction in the number of flip-flops. An unpredictable effect in reducing the number of flip-flops is that sometimes the equivalent circuit may require more combinational gates.

State Reduction

Example :

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	

Initial point

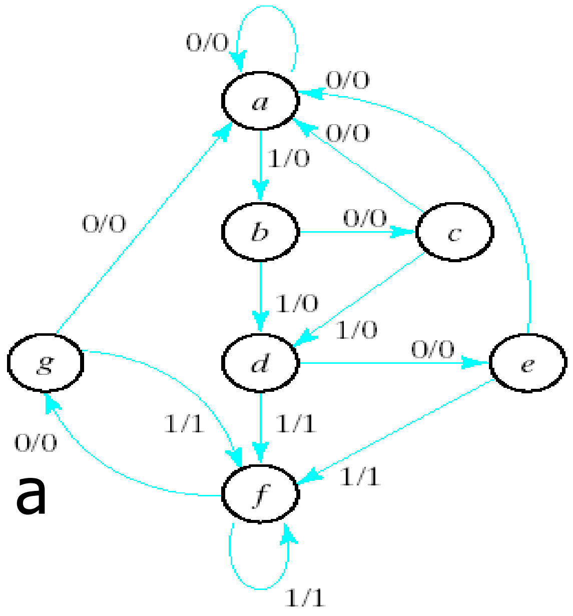


Fig. 5-22 State Diagram

State Reduction

We now proceed to reduce the number of states for this example. First, we need the state table; it is more convenient to apply procedures for state reduction using a table rather than a diagram. The state table of the circuit is listed in Table 5-6 and is obtained directly from the state diagram.

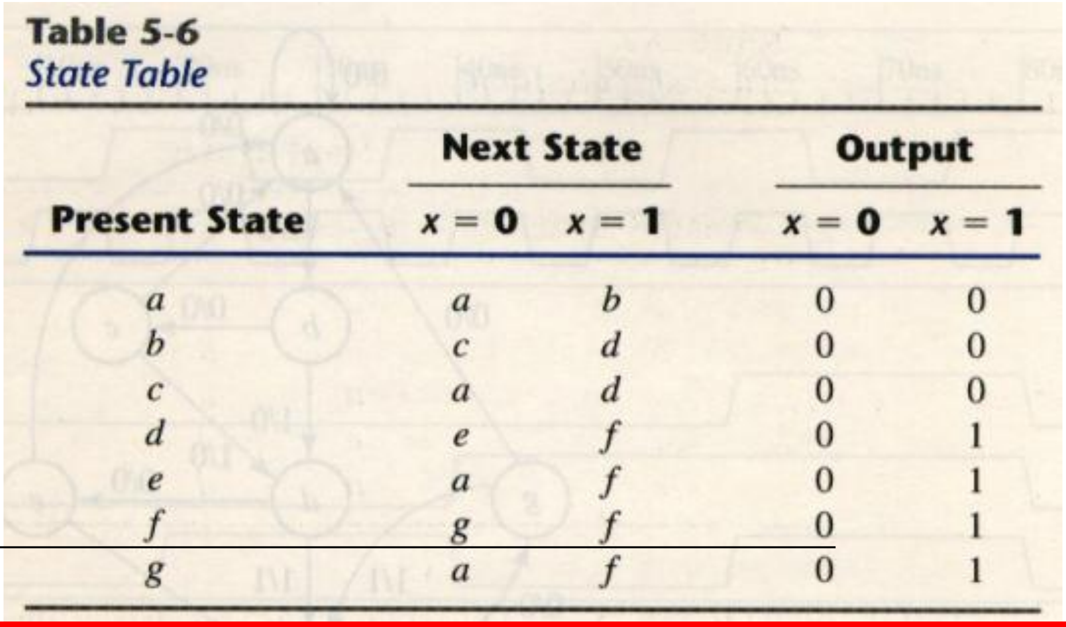


Table 5-6
State Table

Present State	Next State		Output	
	$x = 0$	$x = 1$	$x = 0$	$x = 1$
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
g	a	f	0	1

State Reduction

States g and e are two such states: they both go to states a and f and have outputs of 0 and 1 for x=0 and x=1, respectively. Therefore, states g and e are equivalent and one of these states can be removed. The procedure of removing a state and replacing it by its equivalent is demonstrated in Table 5-7. The row with present g is removed and state g is replaced by state e each time it occurs in the next-state columns.

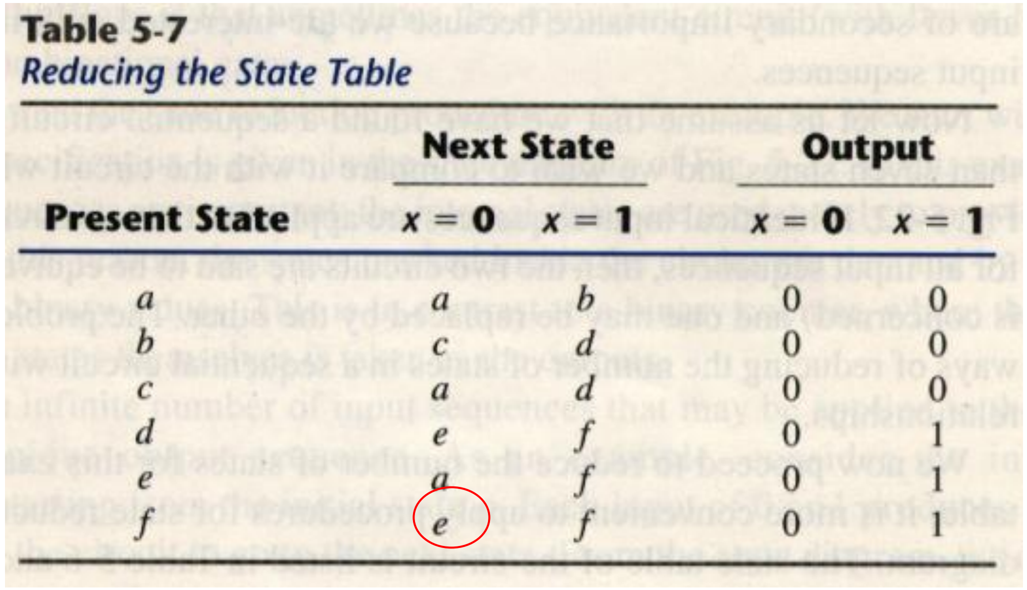


Table 5-7
Reducing the State Table

	Next State		Output	
Present State	$x = 0$	$x = 1$	$x = 0$	$x = 1$
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	e	f	0	1

State Reduction

Present state f now has next states e and f and outputs 0 and 1 for x=0 and x=1, respectively. The same next states and outputs appear in the row with present state d. Therefore, states f and d are equivalent and state f can be removed and replaced by d. The final reduced table is shown in Table 5-8. The state diagram for the reduced table consists of only five states and is shown in Fig. 5-23.

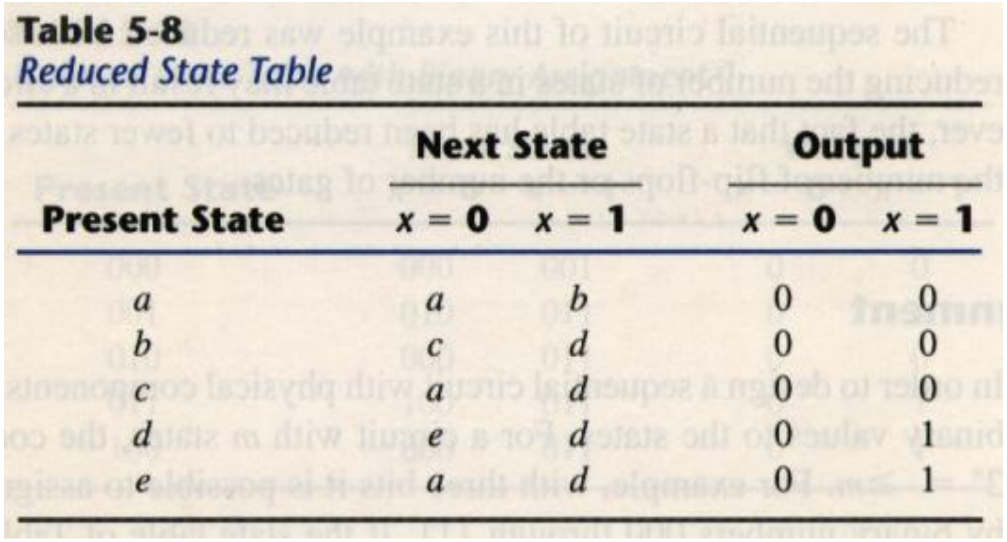
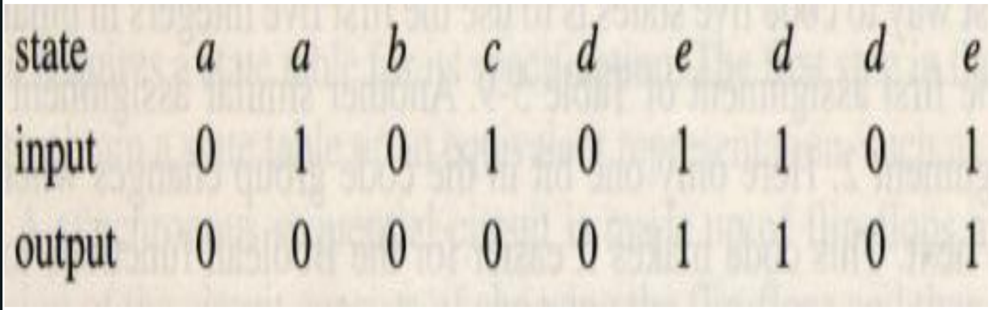


Table 5-8
Reduced State Table

	Next State		Output	
Present State	$x = 0$	$x = 1$	$x = 0$	$x = 1$
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

State Reduction



state	a	a	b	c	d	e	d	d	e
input	0	1	0	1	0	1	1	0	1
output	0	0	0	0	0	1	1	0	1

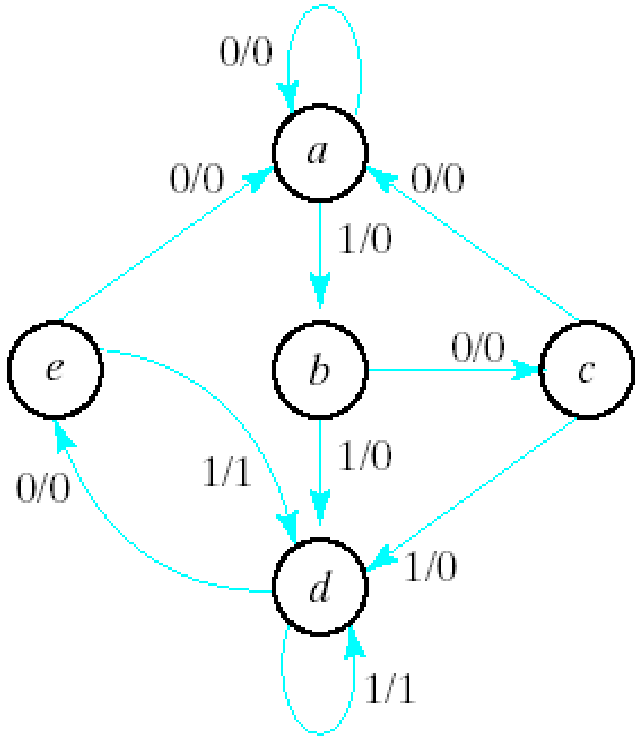


Fig. 5-23 Reduced State Diagram

State Assignment

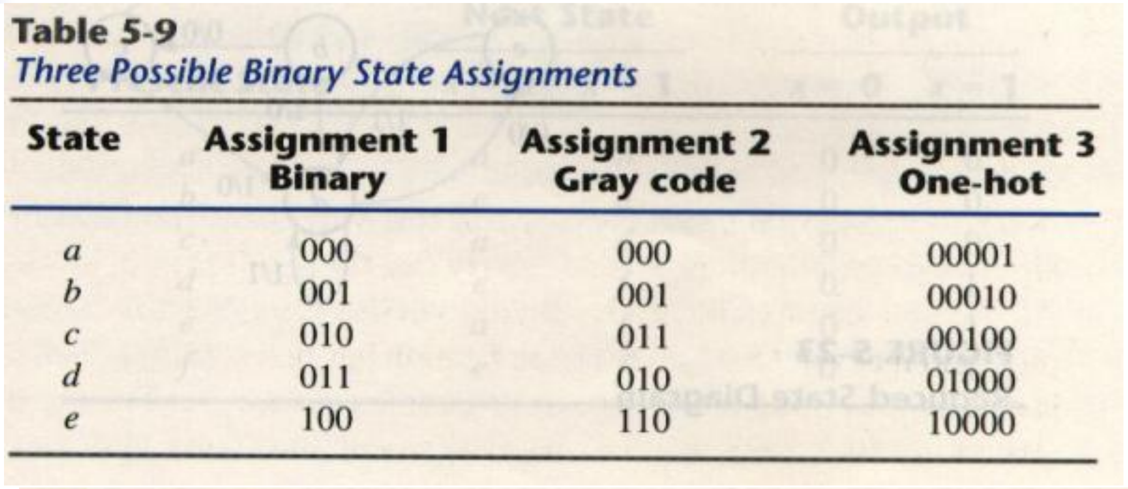


Table 5-9
Three Possible Binary State Assignments

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	100	110	10000

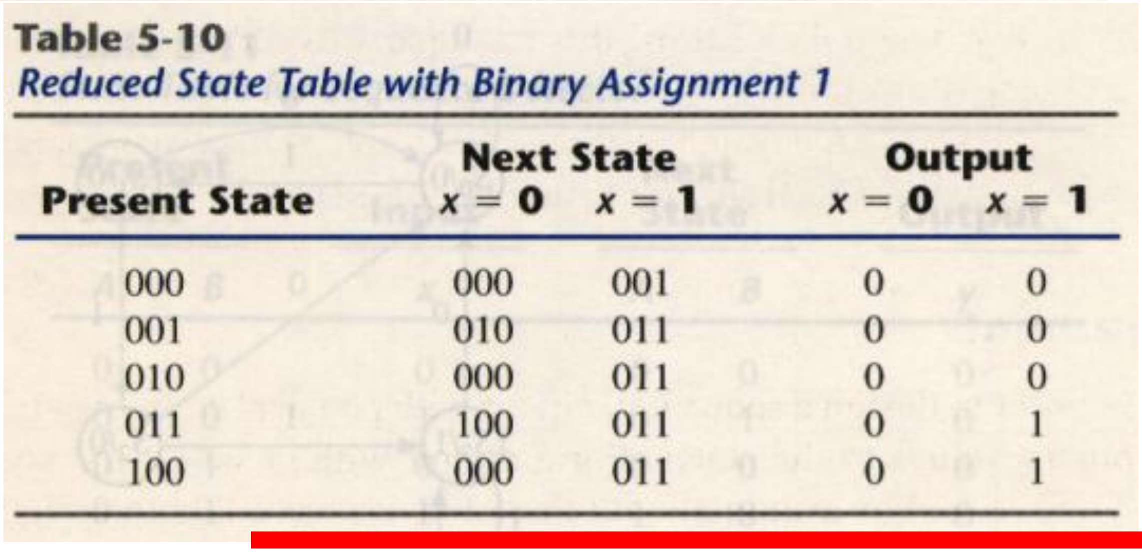


Table 5-10
Reduced State Table with Binary Assignment 1

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

5-7 Design Procedure

The procedure for designing synchronous sequential circuits can be summarized by a list of recommended steps.

1. From the word description and specifications of the desired operation, derive a state diagram for the circuit.
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 to be used.
6. Derive the simplified flip-flop input equations and output equations.
7. Draw the logic diagram.

Design Procedure

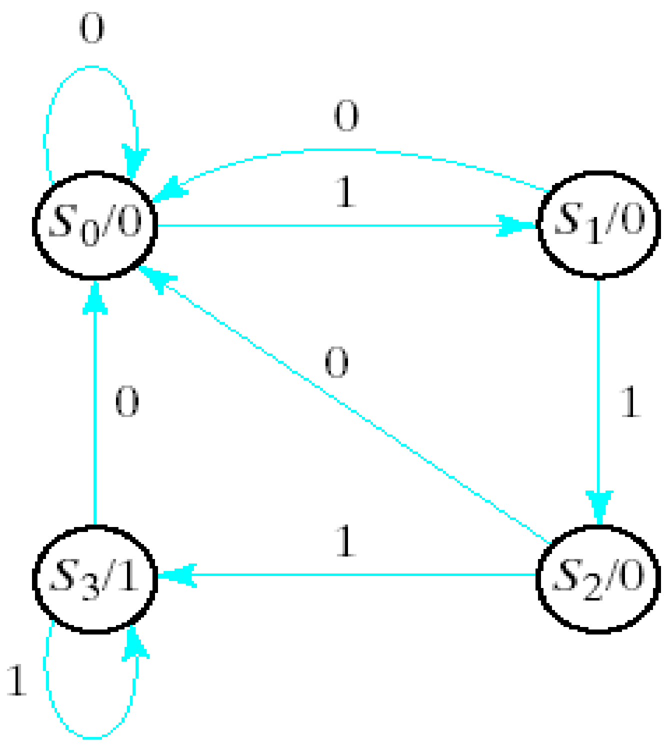


Fig. 5-24 State Diagram for Sequence Detector

Synthesis Using D Flip-Flops

$$A(t+1) = D_A(A, B, x) = \sum (3, 5, 7)$$
$$B(t+1) = D_B(A, B, x) = \sum (1, 5, 7)$$
$$y(A, B, x) = \sum (6, 7)$$

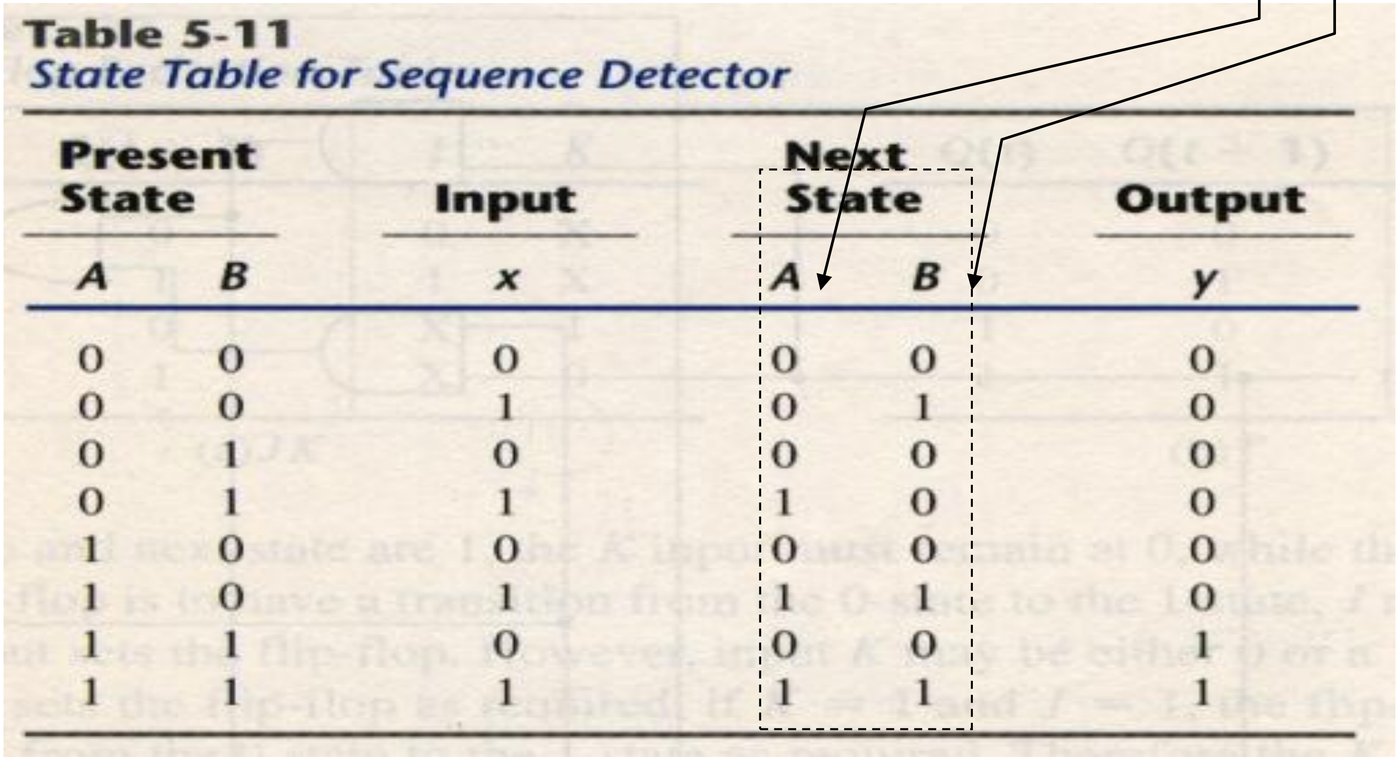


Table 5-11
State Table for Sequence Detector

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	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

Synthesis Using D Flip-Flops

$D_A = Ax + Bx$

$D_B = Ax + B`x$

$y = AB$

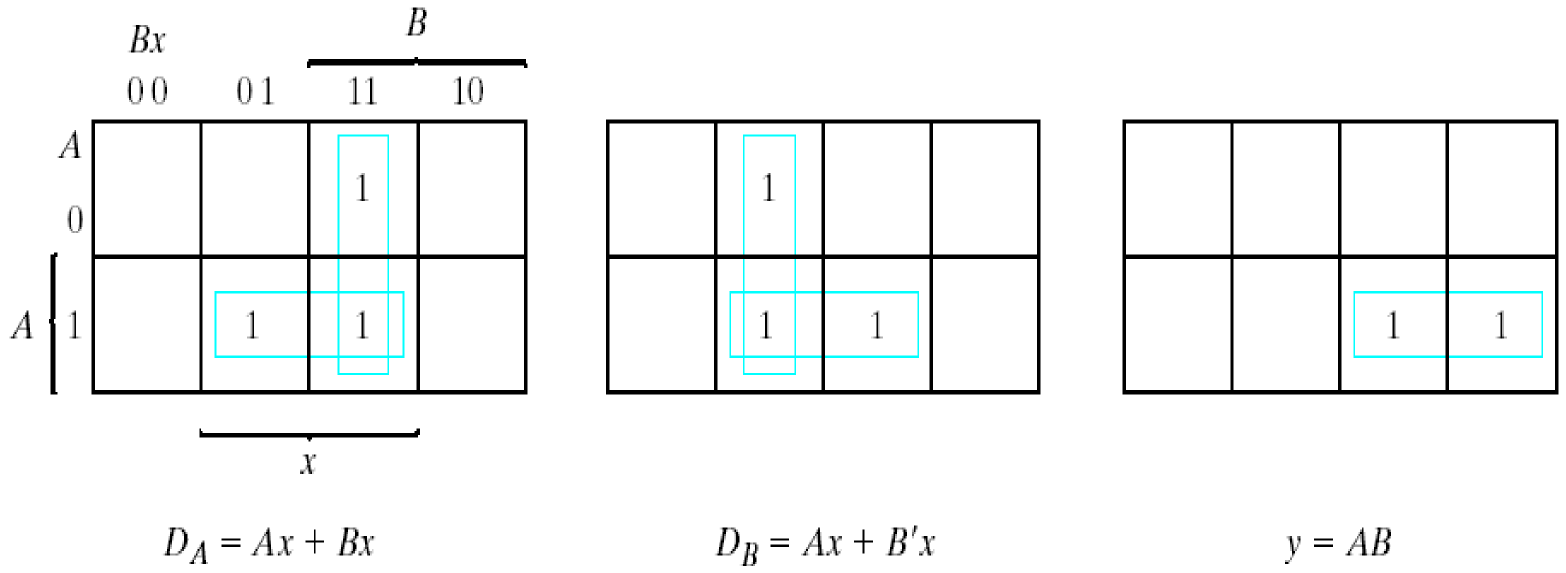


Fig. 5-25 Maps for Sequence Detector

Synthesis Using D Flip-Flops

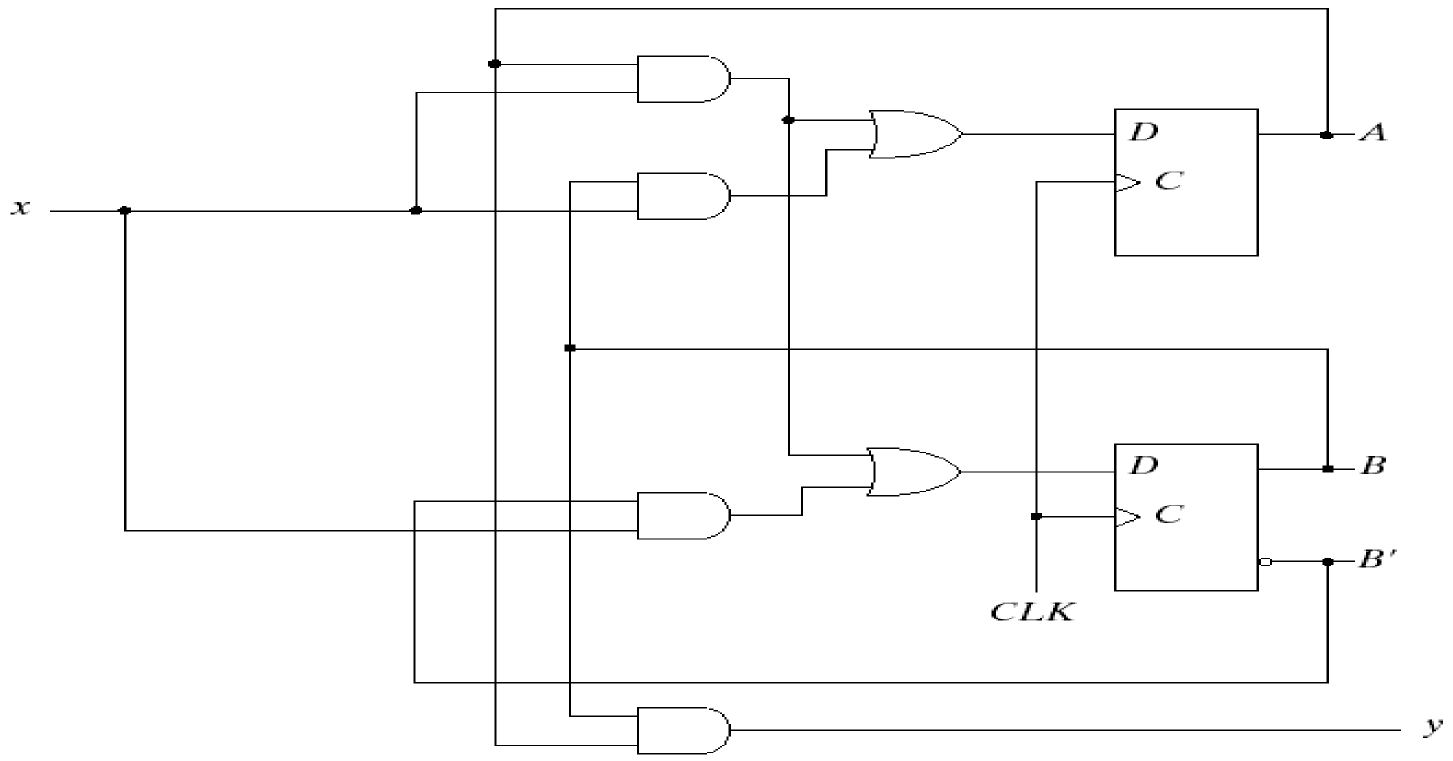


Fig. 5-26 Logic Diagram of Sequence Detector

Synthesis Using JK Flip-Flops

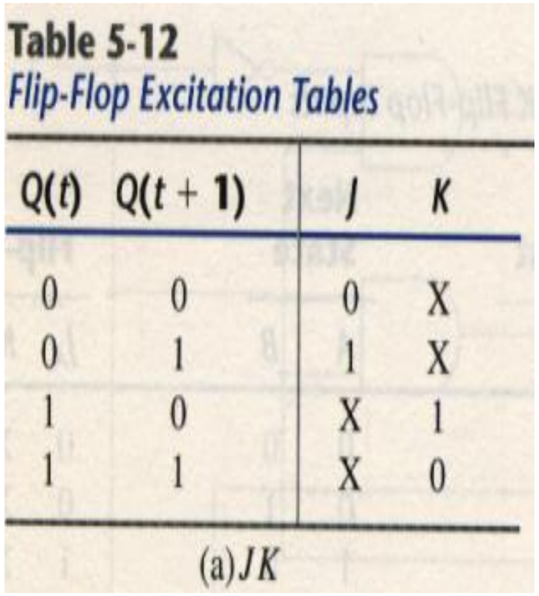


Table 5-12
Flip-Flop Excitation Tables

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

(a) JK

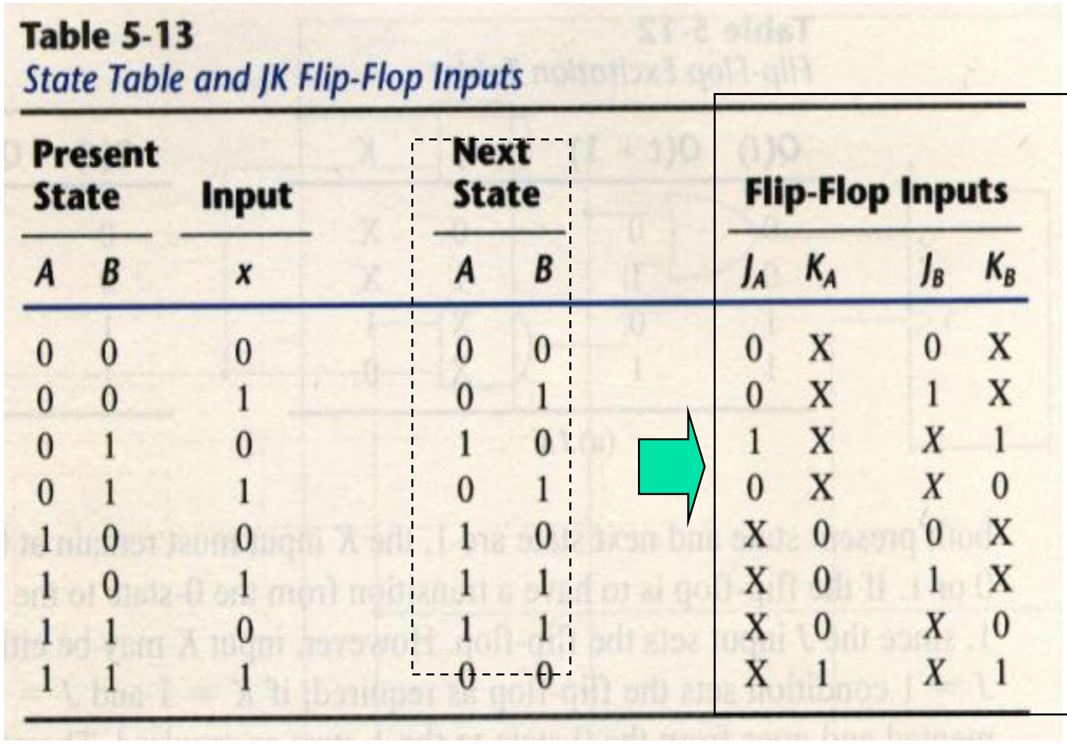
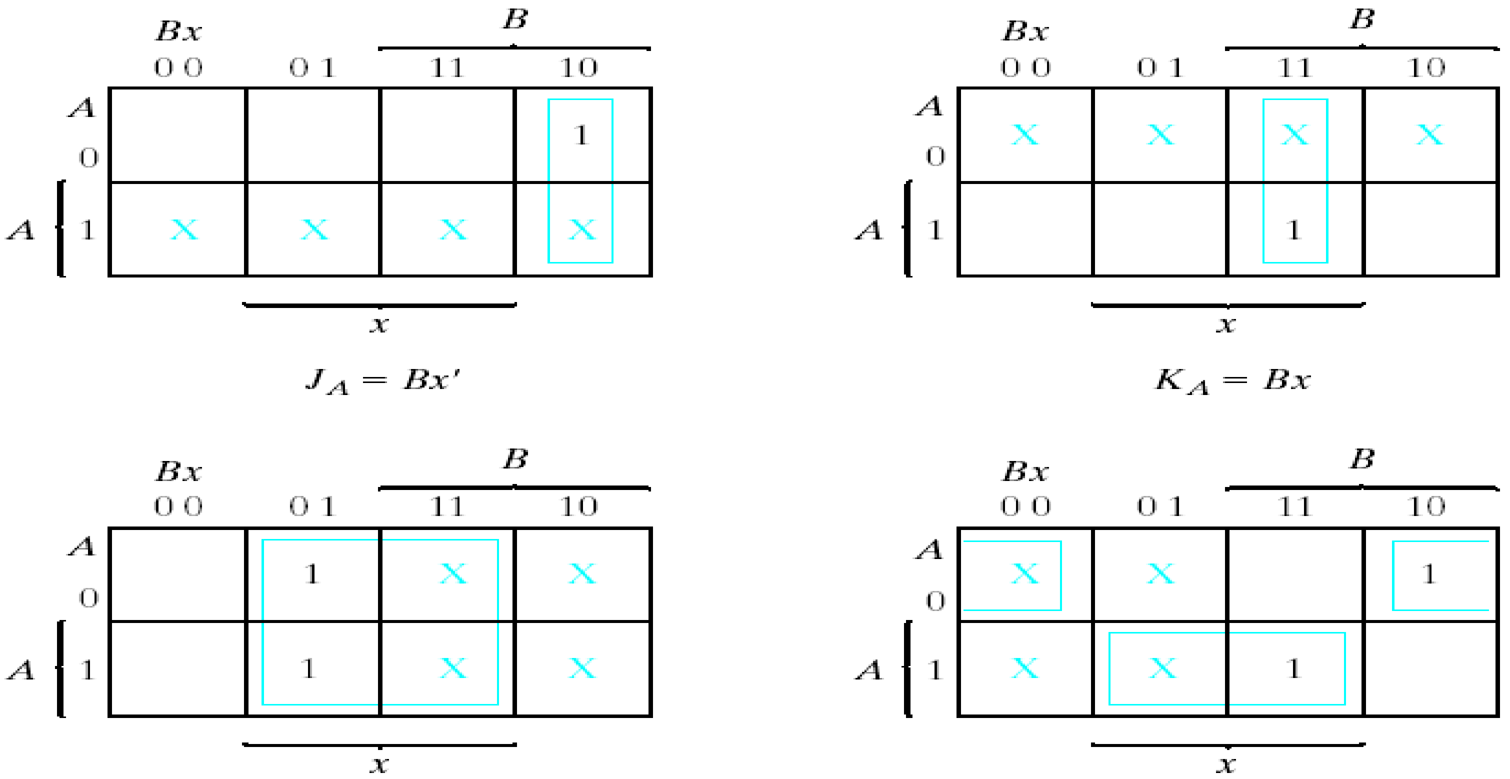


Table 5-13
State Table and JK Flip-Flop Inputs

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

Ref. Table 5-1

Synthesis Using JK Flip-Flops



$J_A = Bx'$

$K_A = Bx$

$J_B = x$

$K_B = (A \oplus x)'$

ice Hall, Inc.

Fig. 5-27 Maps for J and K Input Equations

Synthesis Using JK Flip-Flops

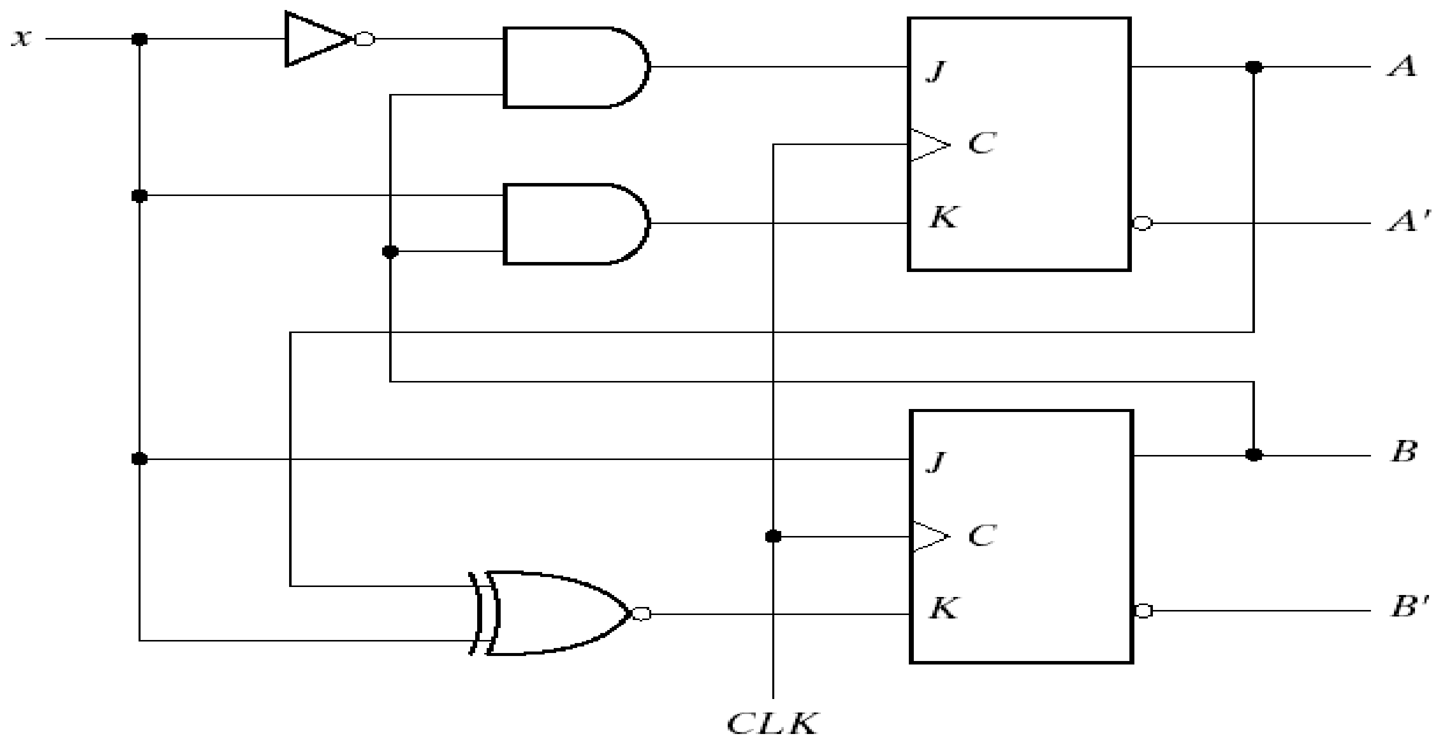
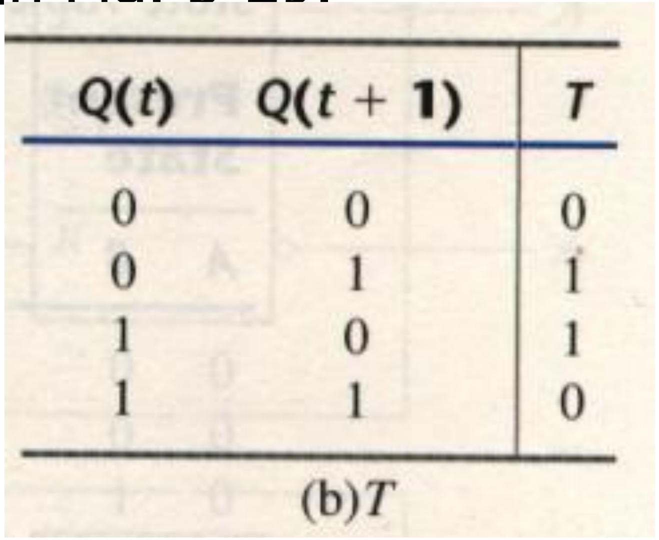


Fig. 5-28 Logic Diagram for Sequential Circuit with *JK* Flip-Flops

Synthesis Using T Flip-Flops

The synthesis using T flip-flops will be demonstrated by designing a binary counter. An n-bit binary counter consists of n flip-flops that can count in binary from 0 to 2^n-1. The state diagram of a 3-bit counter is shown in Fig. 5-29.



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

(b) T

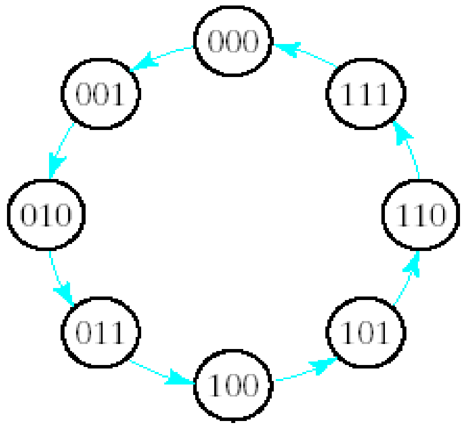


Fig. 5-29 State Diagram of 3-Bit Binary Counter

Synthesis Using T Flip-Flops

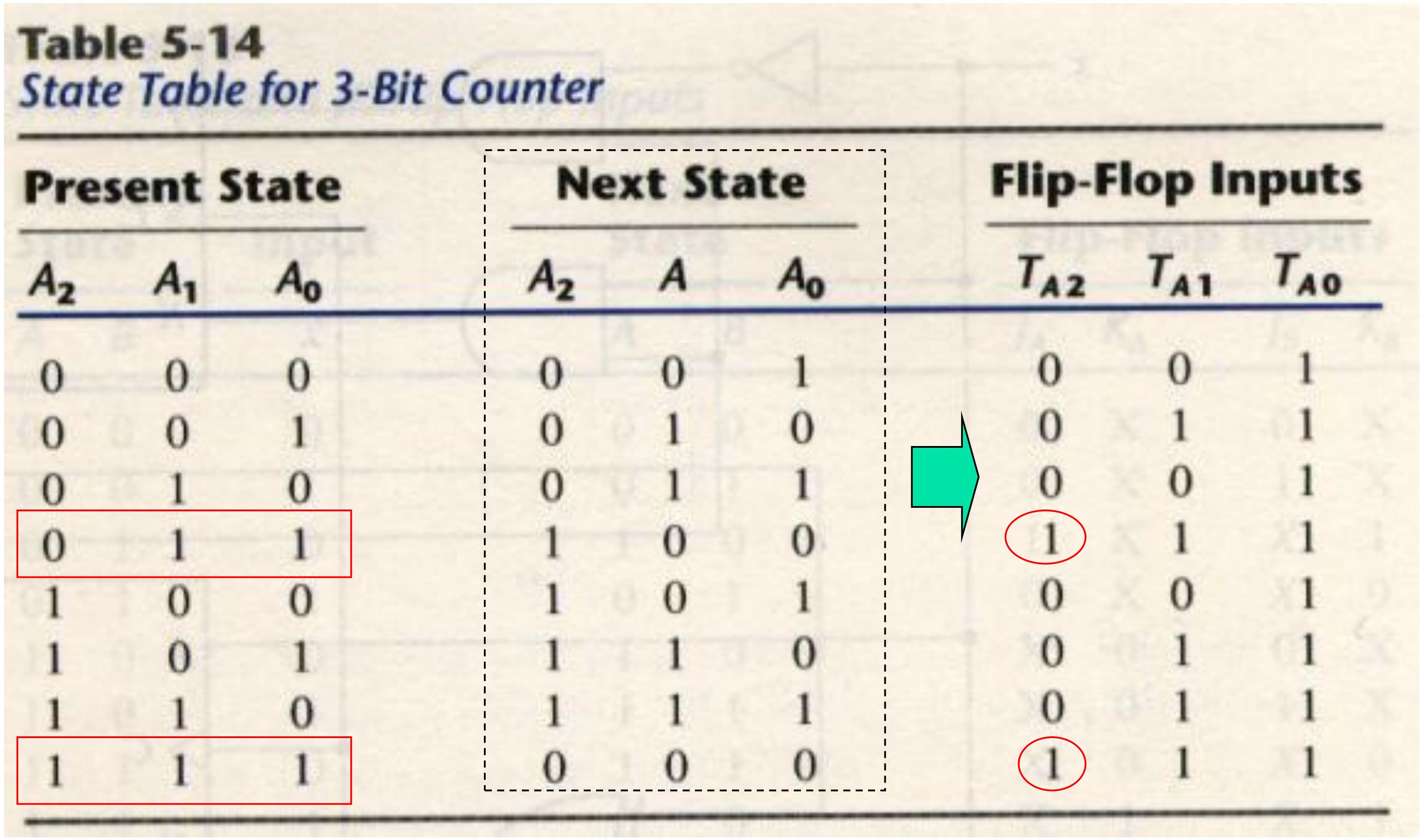
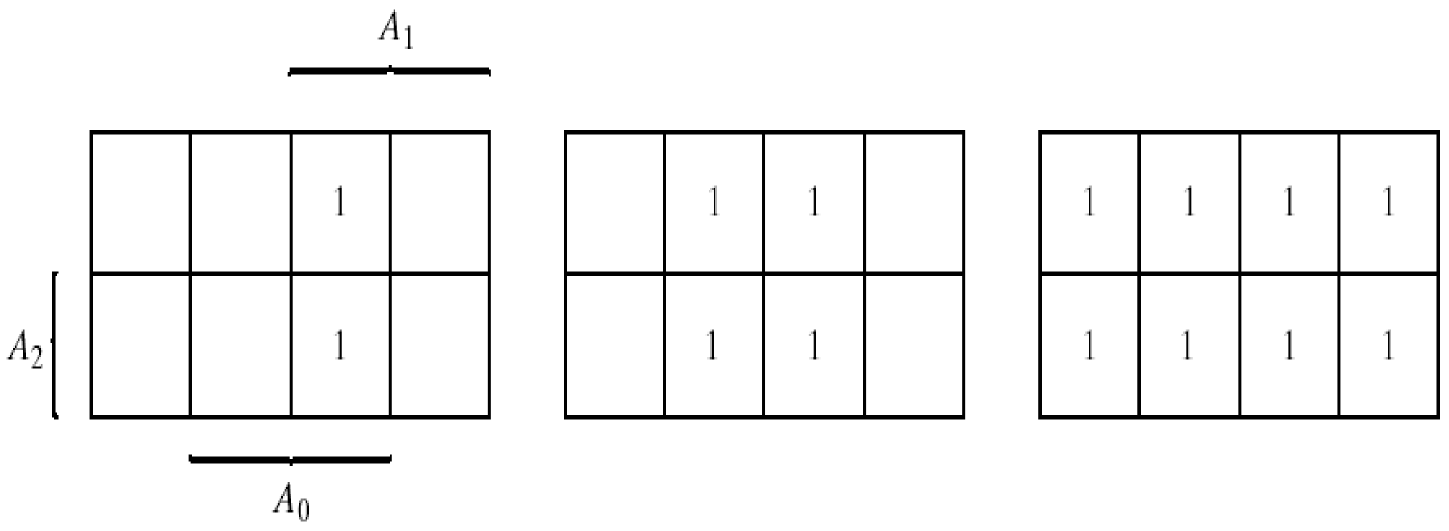


Table 5-14
State Table for 3-Bit Counter

Present State			Next State			Flip-Flop Inputs		
A_2	A_1	A_0	A_2	A	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	1	1
1	1	1	0	0	0	1	1	1

Synthesis Using T Flip-Flops



		1	
		1	

$T_{A2} = A_1 A_0$

	1	1	
	1	1	

$T_{A1} = A_0$

1	1	1	1
1	1	1	1

$T_{A0} = 1$

Fig. 5-30 Maps for 3-Bit Binary Counter

Synthesis Using T Flip-Flops

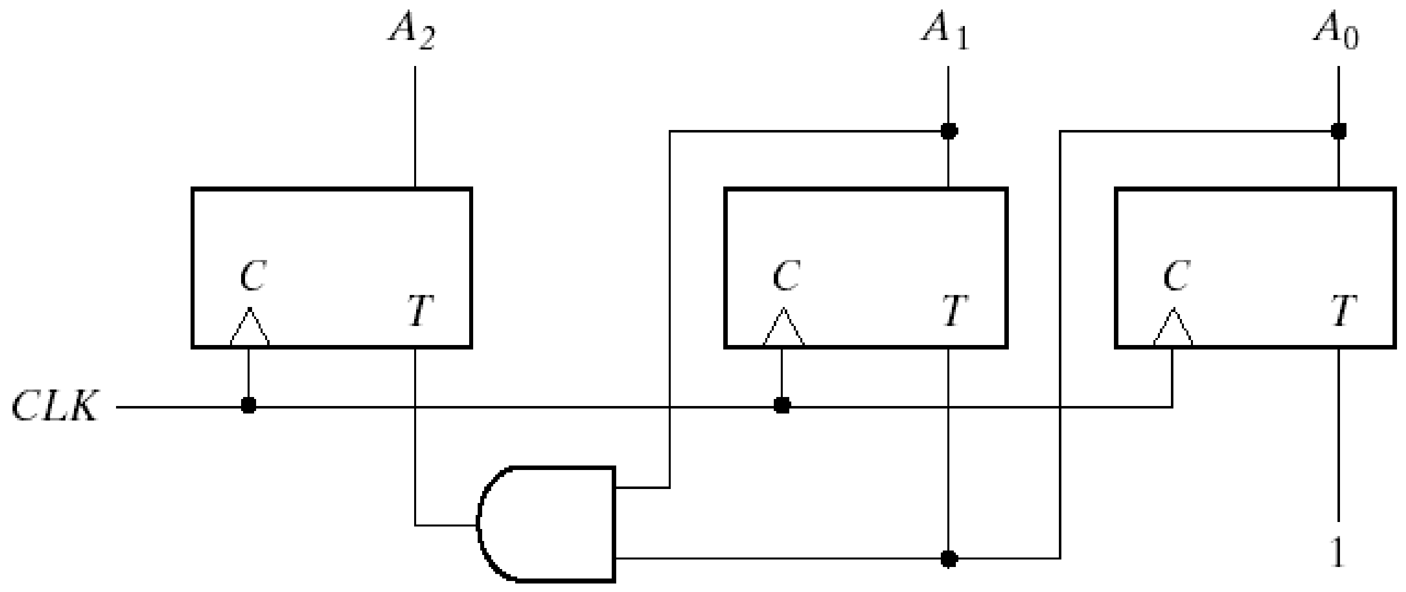


Fig. 5-31 Logic Diagram of 3-Bit Binary Counter

THANK YOU