UNIT 12

Registers and Counters

This chapter includes:

12.1 Registers and Register Transfers

12.2 Shift Registers

12.3 Design of Binary Counters

12.4 Counters for Other Sequences

12.5 Counter Design using S-R and J-K Flip-Flops

12.6 Derivation of Flip-Flop Input Equations-Summary

Learning Objectives

1. Explain the operation of registers. Show how to transfer data between registers using a tri-state bus.

2. Explain the operation of shift registers, show how to build them using flip-flops, and analyze their operation. Construct a timing diagram for a shift register.

3. Explain the operation of binary counters, show how to build them using flip-flops and gates, and analyze their operation.

Learning Objectives

4. Given the present state and desired next state of a flip-flop, determine the required flip-flop inputs.

5. Given the desired counting sequence for a counter, derive the flip-flop input equations.

6. Explain the procedures used for deriving flip-flop input equations.

7. Construct a timing diagram for a counter by tracing signals through the circuit.

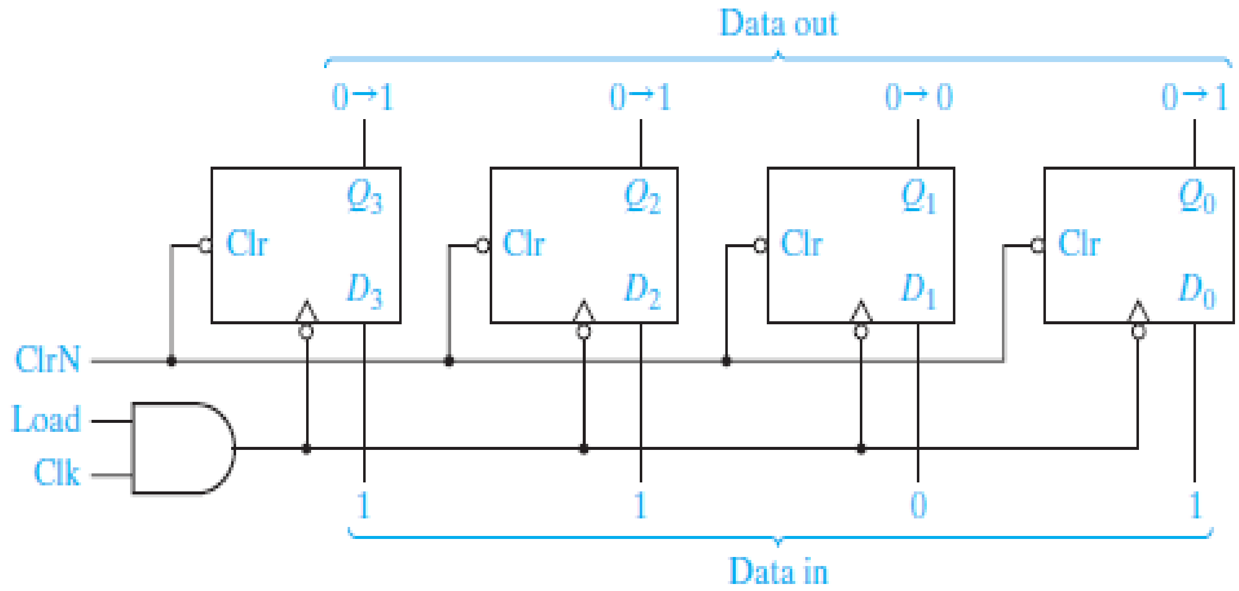
Registers and Register Transfers

Introduction:

- A **register** consists of a group of flip-flops with a common clock input. Registers are commonly used to store and shift binary data.
- A **counter** is usually constructed from two or more flip-flops which change states in a prescribed sequence when input pulses are received.

Registers and Register Transfers

4-Bit D Flip-Flop Registers with Data, Load, Clear and Clock Inputs:



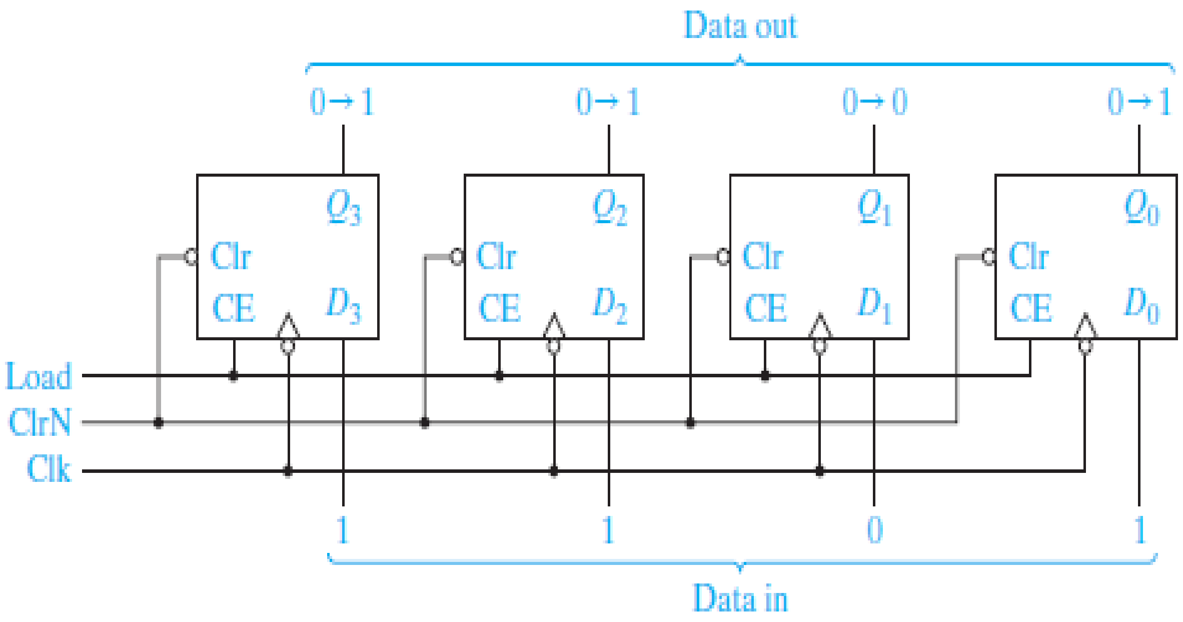
(a) Using gated clock

FIGURE 12-1 4-Bit D Flip-Flop Registers with Data, Load, Clear, and Clock Inputs

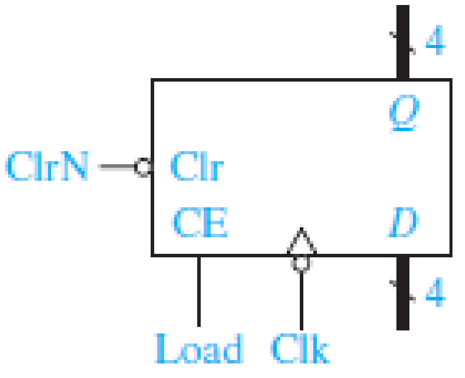
© Cengage Learning 2014

Registers and Register Transfers

4-Bit D Flip-Flop Registers with Data, Load, Clear and Clock Inputs (continued):



(b) With clock enable



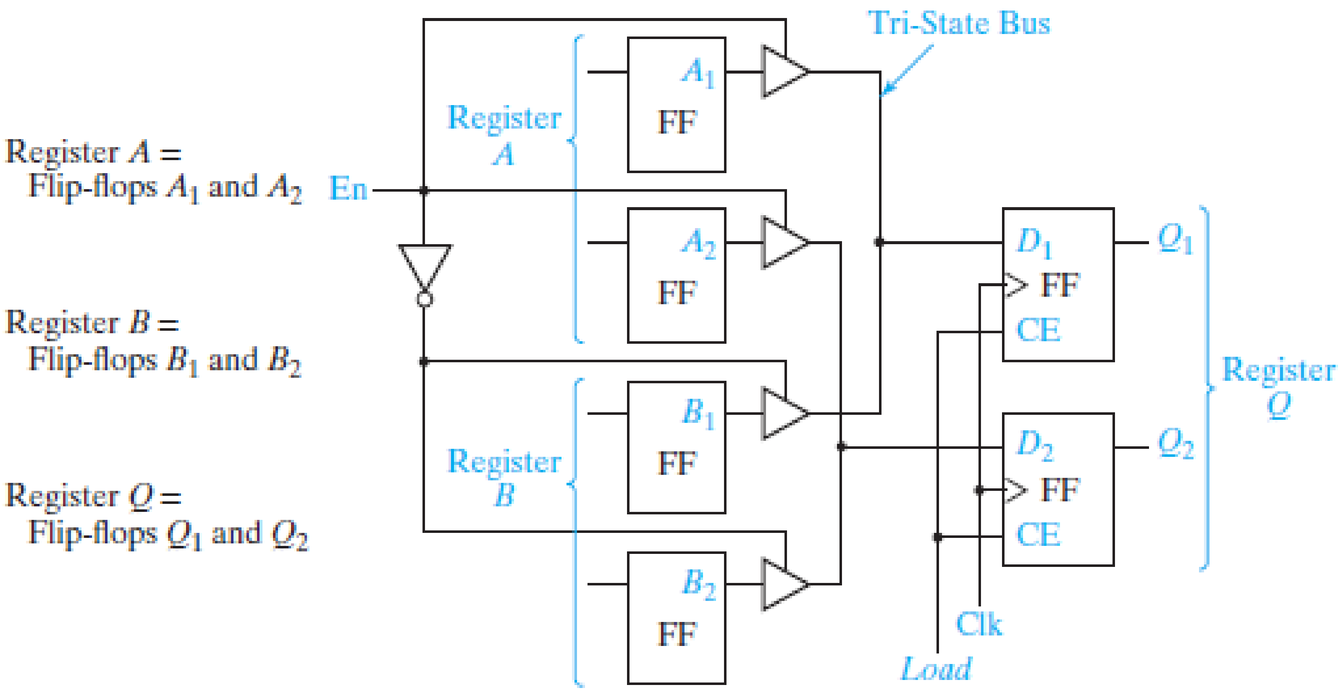
(c) Symbol

Registers and Register Transfers

Data Transfer Between Registers:

FIGURE 12-2 Data Transfer Between Registers

© Cengage Learning 2014



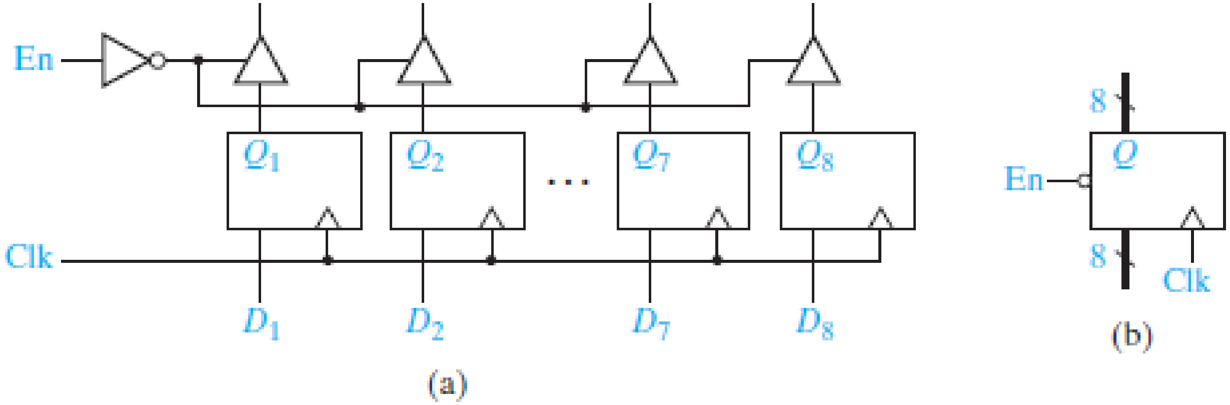
Registers and Register Transfers

8-Bit Register with Tri-State Output:

- Figure 12-3(a) shows an integrated circuit register that contains eight D flip-flops with tri-state buffers at the flip-flop outputs.
- These buffers are enabled when En = 0. A symbol for this 8-bit register is shown in Figure 12-3(b).

FIGURE 12-3 Logic Diagram for 8-Bit Register with Tri-State Output

© Cengage Learning 2014

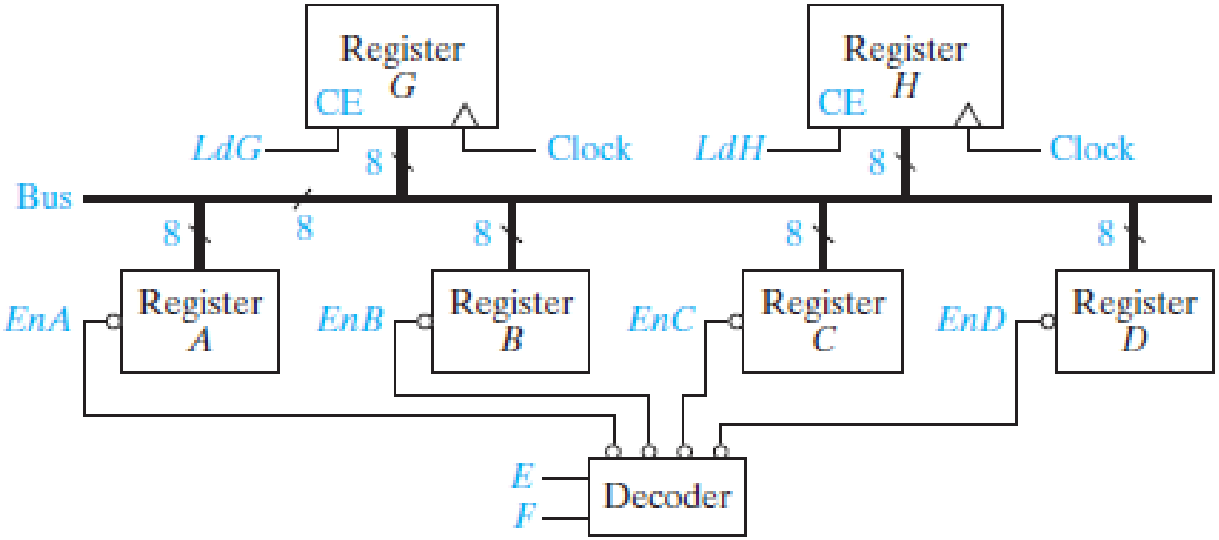


Registers and Register Transfers

Data Transfer Using a Tri-State Bus:

FIGURE 12-4
Data Transfer Using a Tri-State Bus

© Cengage Learning 2014



Operation is as follows:

If $EF = 00$, A is stored in G (or H).

If $EF = 01$, B is stored in G (or H).

If $EF = 10$, C is stored in G (or H).

If $EF = 11$, D is stored in G (or H).

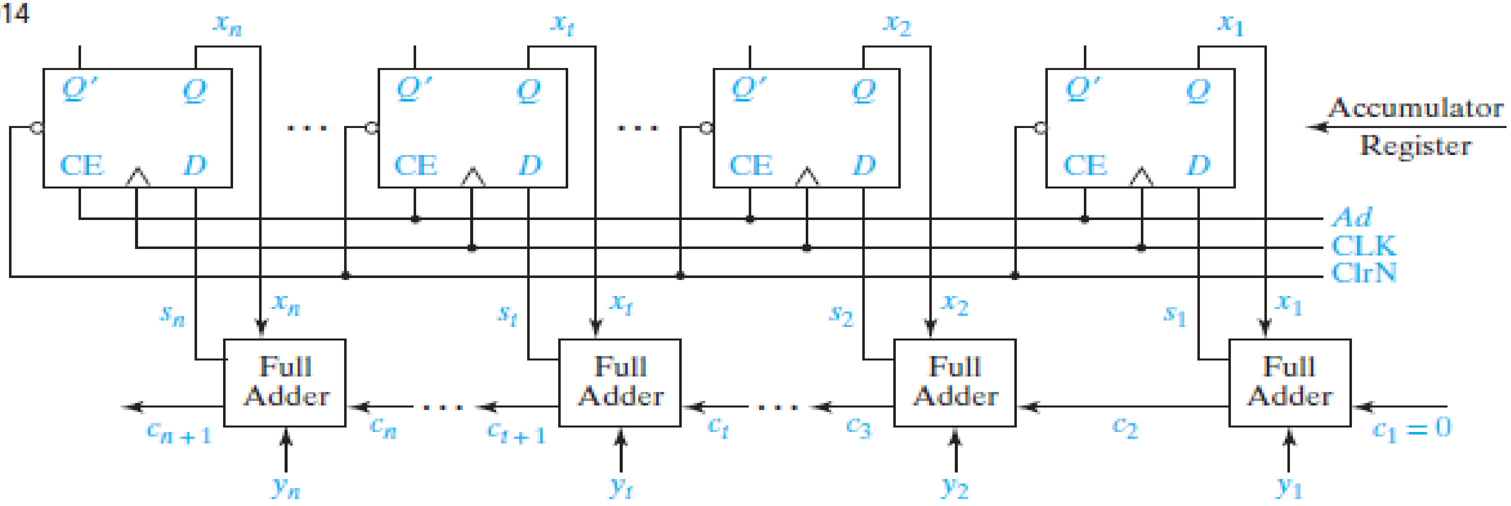
Registers and Register Transfers

Parallel Adder with Accumulator:

- It is frequently desirable to store one number in a register of flip-flops (called an **accumulator**) and add a second number to it, leaving the result stored in the accumulator. See figure 12-5 below:

FIGURE 12-5 *n*-Bit Parallel Adder with Accumulator

© Cengage Learning 2014



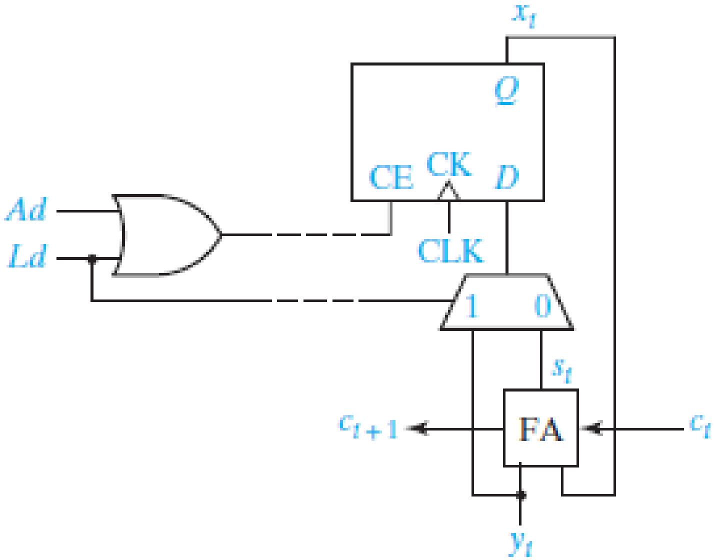
Registers and Register Transfers

Adder Cell with Multiplexer:

- Figure 12-6 shows a typical cell of the adder where the accumulator flip-flop can either be loaded directly from y_i or from the sum output (s_i).

FIGURE 12-6
Adder Cell with Multiplexer

© Cengage Learning 2014



Shift Registers

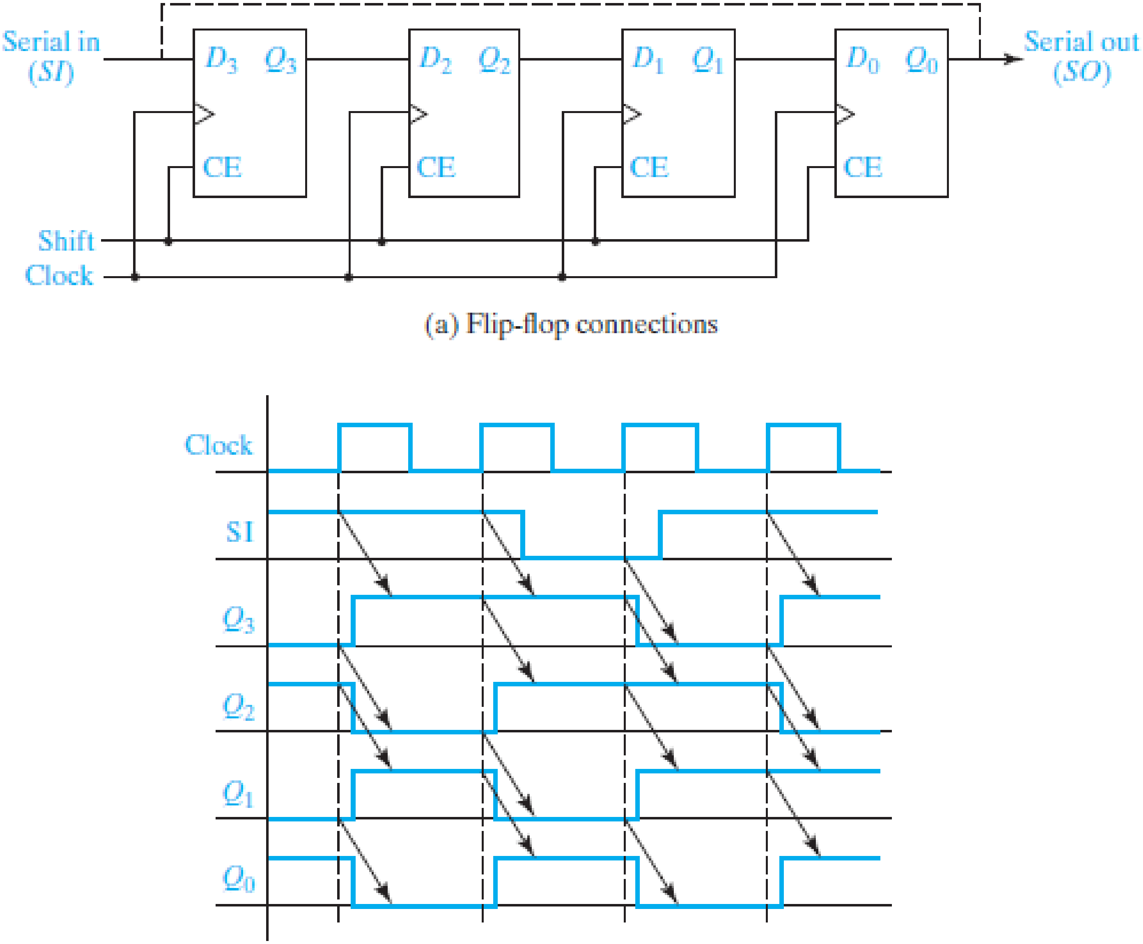
Shift Registers:

- A shift register is a register in which binary data can be stored, and this data can be shifted to the left or right when a shift signal is applied.
- Shifts can be linear or cyclic.
- The figure on the next slide shows a 4-bit right-shift register with serial input and output constructed from D flip-flops.
- When Shift=1, the clock is enabled and shifting occurs on the rising clock edge. When Shift=0, no shifting occurs and the data in the register is unchanged.

Shift Registers

FIGURE 12-7 Right-Shift Register

© Cengage Learning 2014



(a) Flip-flop connections

(b) Timing diagram

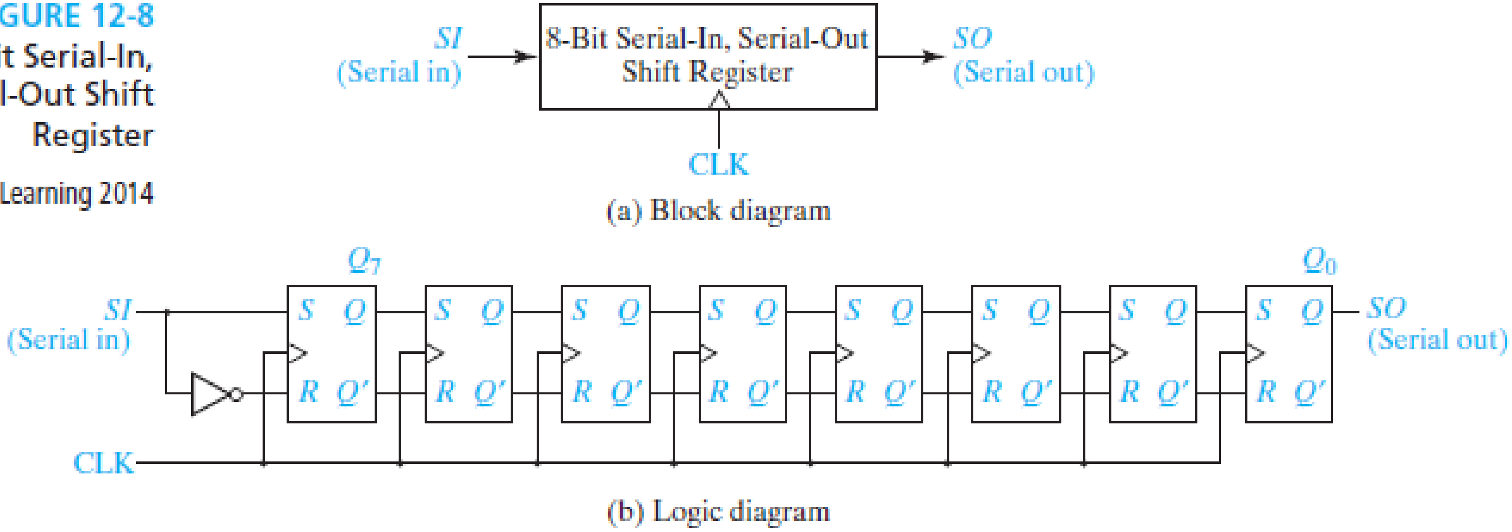
Shift Registers

Serial In/ Serial Out Registers:

- **Serial in** means that data is shifted into the first flip-flop one bit at a time, and the flip-flops cannot be loaded in parallel.
- **Serial out** means that data can only be read out of the last flip-flop and the outputs from the other flip-flops are not connected to terminals of the integrated circuit.

Shift Registers

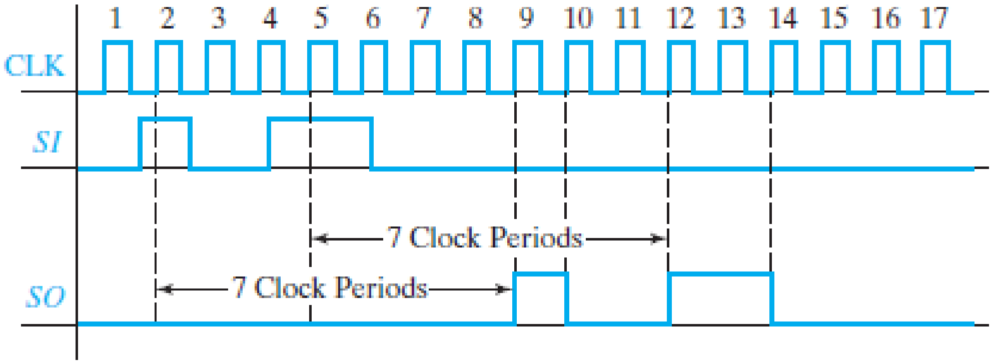
FIGURE 12-8
8-Bit Serial-In, Serial-Out Shift Register



(a) Block diagram

(b) Logic diagram

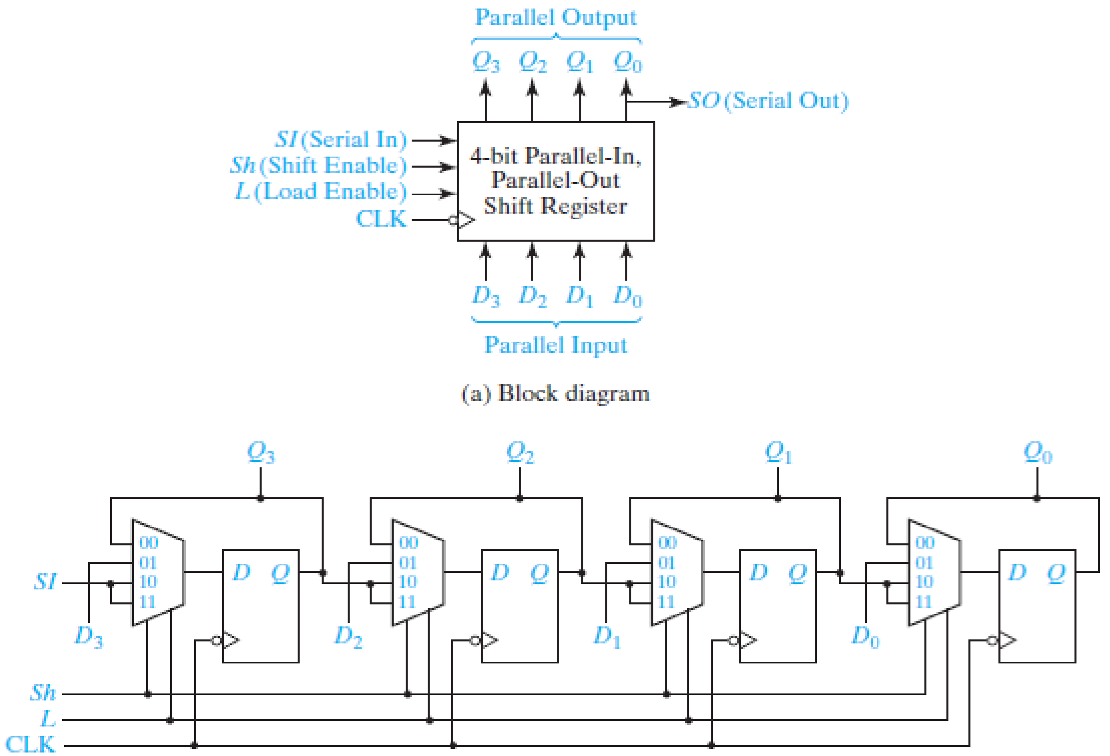
FIGURE 12-9
Typical Timing Diagram for Shift Register of Figure 12-8



Shift Registers

FIGURE 12-10 Parallel-In, Parallel-Out Right-Shift Register

© Cengage Learning 2014



(a) Block diagram

(b) Implementation using flip-flops and MUXes

.The next-state equations for the flip-flops are

$$Q_3^+ = Sh' \cdot L' \cdot Q_3 + Sh' \cdot L \cdot D_3 + Sh \cdot SI \qquad (12\text{-}1)$$
$$Q_2^+ = Sh' \cdot L' \cdot Q_2 + Sh' \cdot L \cdot D_2 + Sh \cdot Q_3$$
$$Q_1^+ = Sh' \cdot L' \cdot Q_1 + Sh' \cdot L \cdot D_1 + Sh \cdot Q_2$$
$$Q_0^+ = Sh' \cdot L' \cdot Q_0 + Sh' \cdot L \cdot D_0 + Sh \cdot Q_1$$

Shift Registers

TABLE12-1
Shift Regi
Operat

© Cengage Learning

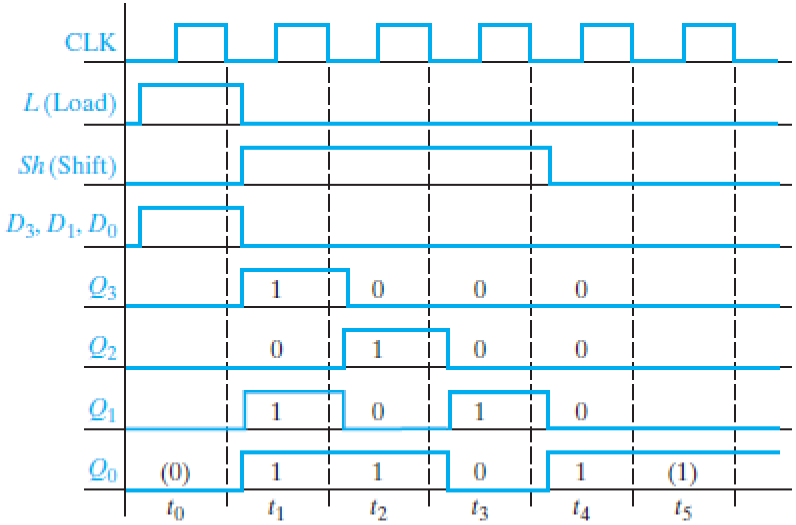
Inputs | Next State |

Action

No change
Load
Right shift

FIGURE 12-11
Timing Diagram for
Shift Register

© Cengage Learning 2014



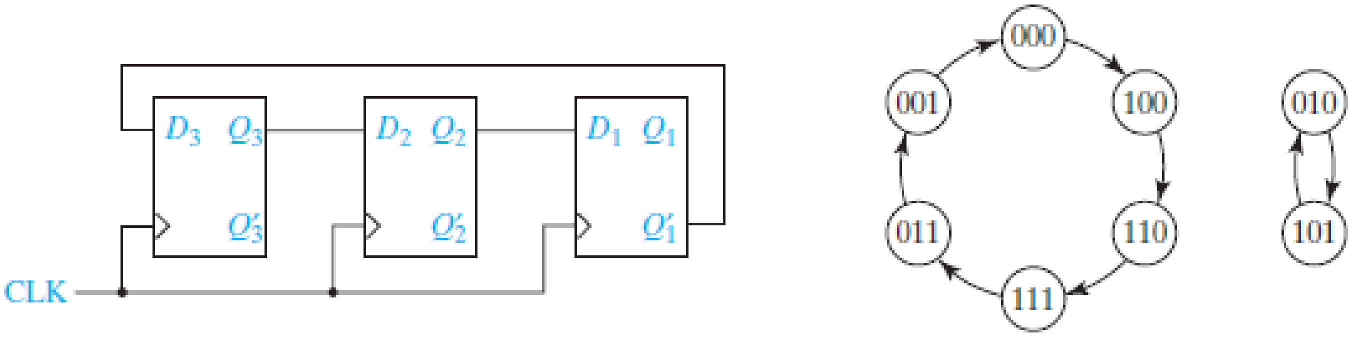
Shift Registers

Johnson Counter:

- A circuit that cycles through a fixed sequence of states is called a **counter**.
- A shift register with inverted feedback is called a **Johnson counter** or a **twisted ring counter**.

FIGURE 12-12 Shift Register with Inverted Feedback

© Cengage Learning 2014



(a) Flip-flop connections

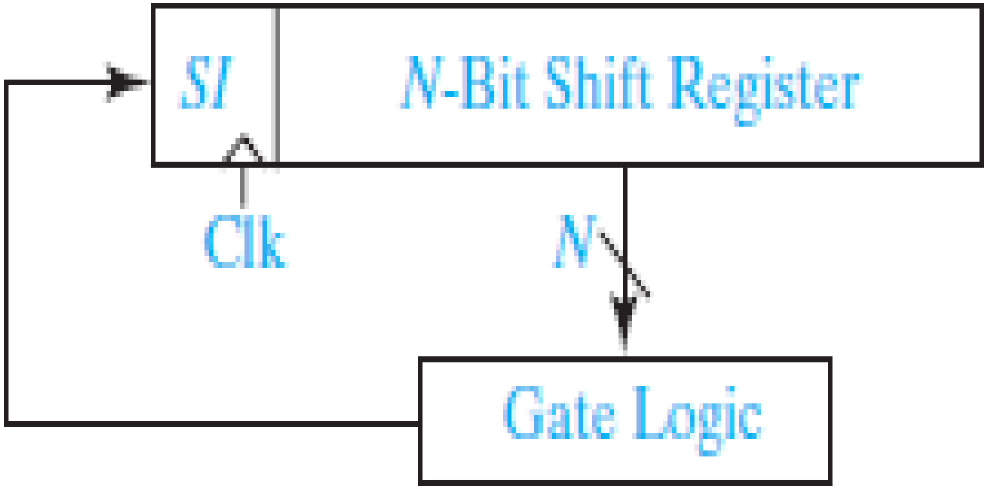
(b) Transition graph

Shift Registers

General Form of a Shift Register Counter:

FIGURE 12-13 General Shift Register Counter

© Cengage Learning 2014



Design of Binary Counters

Synchronous and Ripple Counters:

- For synchronous counters, the operation of the flip-flops is synchronized by a common clock pulse so that when several flip-flops must change state, the state changes occur simultaneously.
- Ripple counters are those in which the state change of one flip-flop triggers another flip-flop. These will not be focused on in this chapter.

Design of Binary Counters

Binary Counters Using 3 T Flip-Flops to Count Clock Pulses:

❖We assume that all the flip-flops change state a short time following the rising edge of the input pulse.

❖The state of the counter is determined by the states of the individual flip-flops; for example, if flip-flop C is in state 0, B in state 1, and A in state 1, the state of the counter is 011.

❖Initially, assume that all flip-flops are set to the 0 state. When a clock pulse is received, the counter will change to state 001; when a second pulse is received, the state will change to 010, etc.

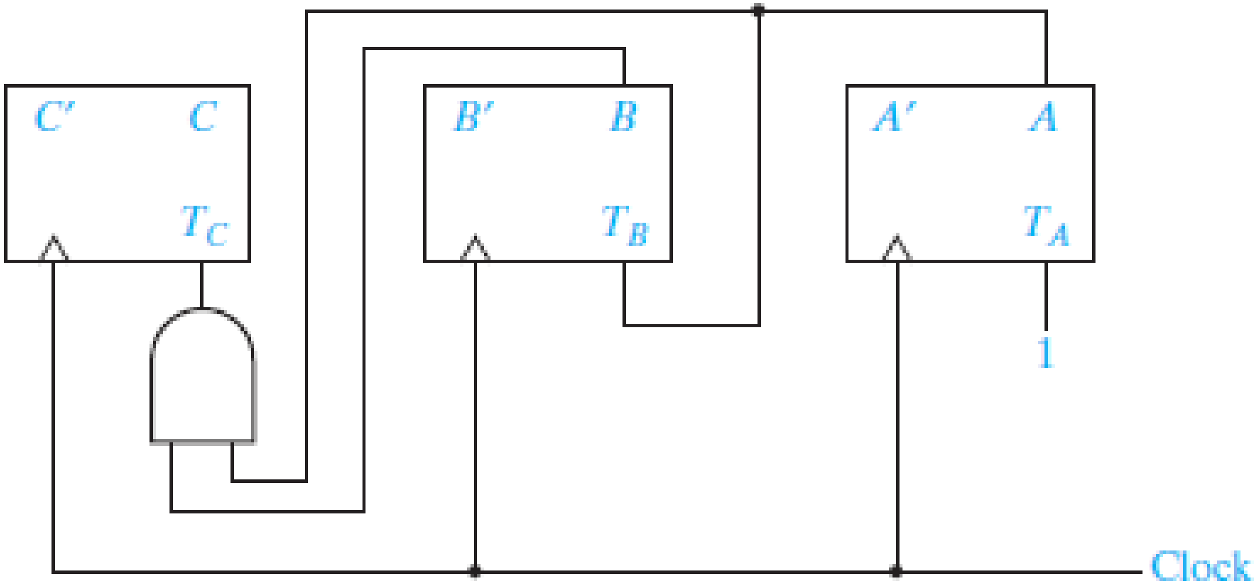
❖When 111 is reached, the counter resets to the 000 state.

❖See next slide for figure.

Design of Binary Counters

FIGURE 12-14 Synchronous Binary Counter

© Cengage Learning 2014



Design of Binary Counters

Design of Binary Counter Using Transition Tables:

❖This table shows the present state of flip-flops C, B, and A (before a clock pulse is received) and the corresponding next state (after the clock pulse is received).

❖A third column in the table is used to derive the inputs for T_C, T_B, and T_A. Whenever the entries in the A and A^+ columns differ, flip-flop A must change state and T_A must be 1.

❖T_C, T_B, and T_A are now derived from the table as functions of C, B, and A.

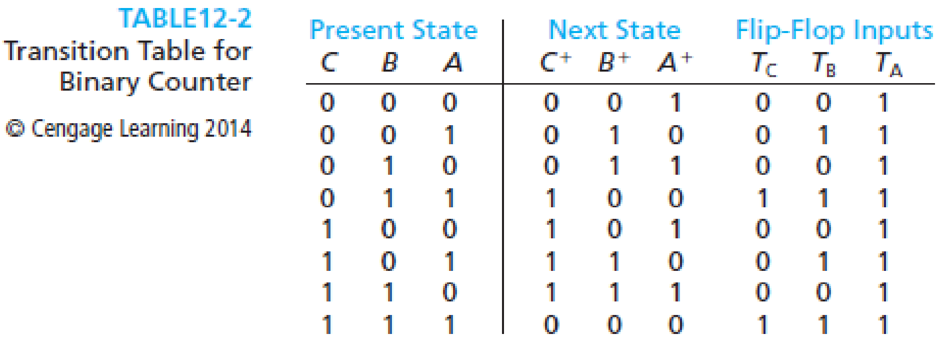


TABLE12-2
Transition Table for Binary Counter

© Cengage Learning 2014

Present State			Next State			Flip-Flop Inputs		
C	B	A	C^+	B^+	A^+	T_C	T_B	T_A
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

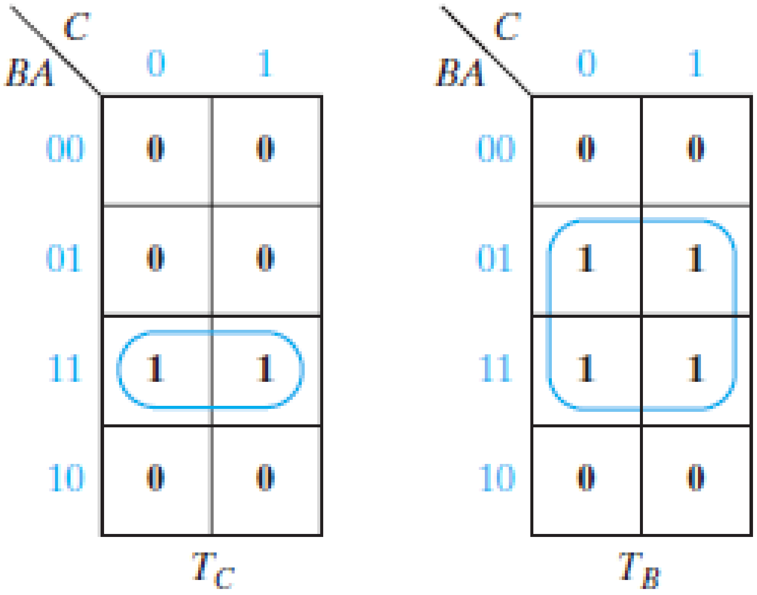
Design of Binary Counters

Karnaugh Maps for Binary Counters:

- Figure 12-15 shows the Karnaugh maps for T_C and T_B, from which T_C = BA and T_B = A. These equations yield the same circuit derived previously for Figure 12-14.

FIGURE 12-15 Karnaugh Maps for Binary Counter

© Cengage Learning 2014



BA \ C	0	1
00	0	0
01	0	0
11	1	1
10	0	0

T_C

BA \ C	0	1
00	0	0
01	1	1
11	1	1
10	0	0

T_B

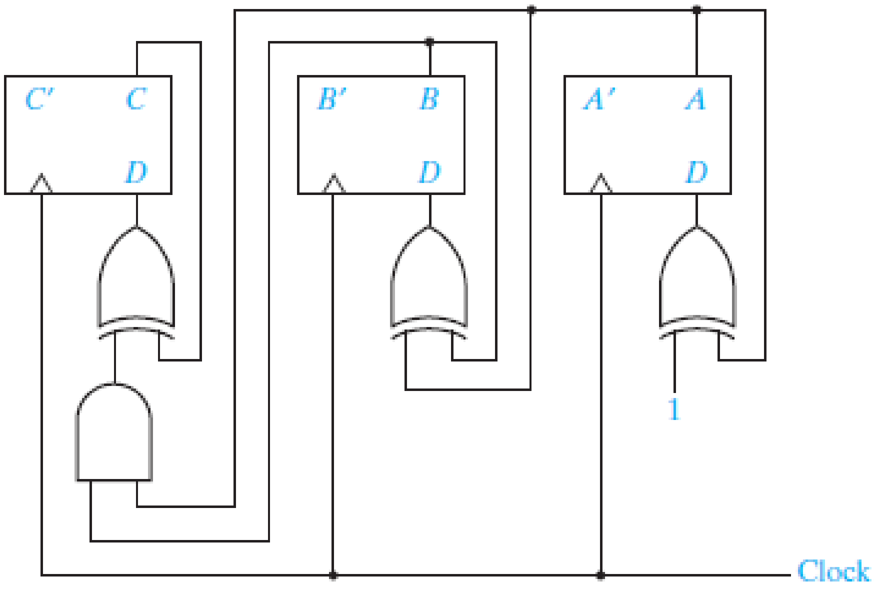
Design of Binary Counters

Binary Counter with D Flip-Flops:

❖We must convert each D flip-flop to a T flip-flop by adding an XOR (exclusive-OR) gate, as shown in Figure 11-28(b). Figure 12-16 shows the resulting counter circuit.

FIGURE 12-16 Binary Counter with D Flip-Flops

© Cengage Learning 2014



Design of Binary Counters

We can also derive the D flip-flop inputs for the binary counter starting with its transition table (Table 12-2). For a D flip-flop, $Q^+ = D$. By inspection of the table, $Q_A{}^+ = A'$, so $D_A = A'$. The maps for Q_B^+ and Q_C^+ are plotted in Figure 12-17. The D input equations derived from the maps are

$$
\begin{aligned}
D_A &= A^+ = A' \\
D_B &= B^+ = BA' + B'A = B \oplus A \qquad (12\text{-}2) \\
D_C &= C^+ = C'BA + CB' + CA' = C'BA + C(BA)' = C \oplus BA
\end{aligned}
$$

which give the same logic circuit as was obtained by inspection.

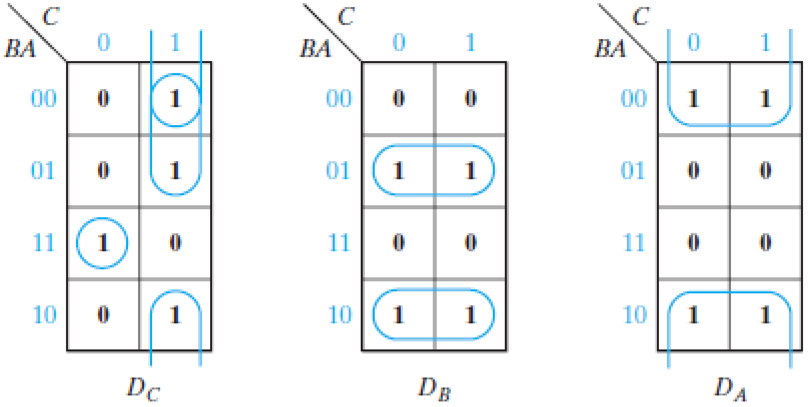


FIGURE 12-17 Karnaugh Maps for D Flip-Flops

© Cengage Learning 2014

D_C:

BA \ C	0	1
00	0	1
01	0	1
11	1	0
10	0	1

D_B:

BA \ C	0	1
00	0	0
01	1	1
11	0	0
10	1	1

D_A:

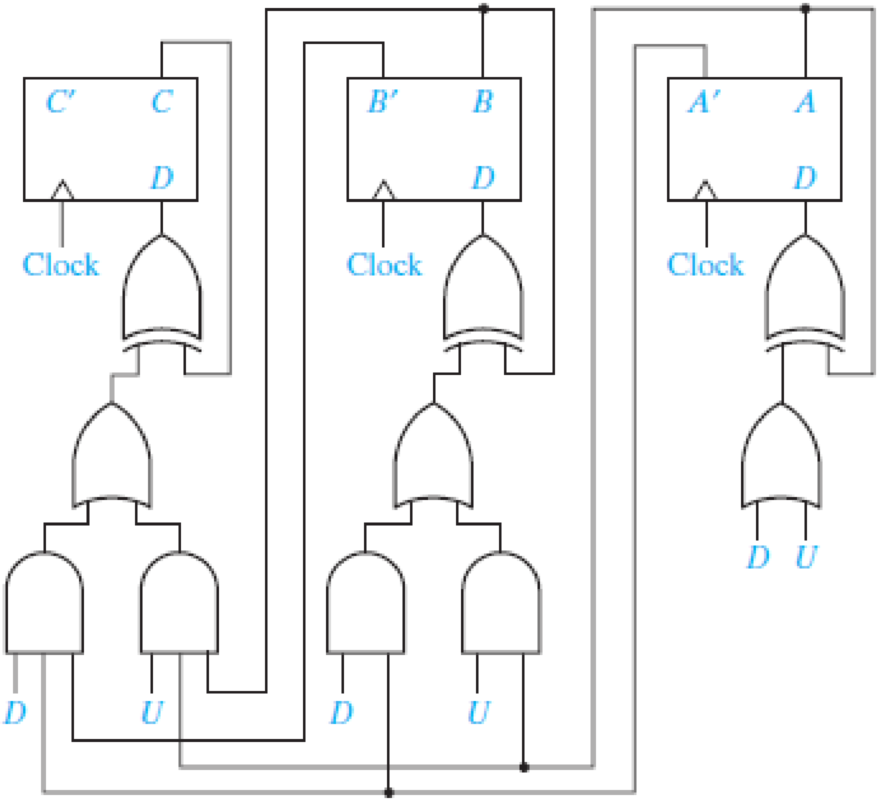
BA \ C	0	1
00	1	1
01	0	0
11	0	0
10	1	1

Design of Binary Counters

Binary Up-Down Counter:

FIGURE 12-19
Binary Up-Down Counter

© Cengage Learning 2014

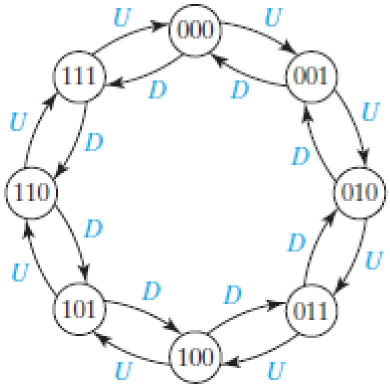


Design of Binary Counters

Up-Down Counter Transition Graph, Table and Logic Equations::

FIGURE 12-18 Transition Graph and Table for Up-Down Counter

© Cengage Learning 2014



	$C^+B^+A^+$	
CBA	U	D
000	001	111
001	010	000
010	011	001
011	100	010
100	101	011
101	110	100
110	111	101
111	000	110

The up-down counter can be implemented using D flip-flops and gates, as shown in Figure 12-19. The corresponding logic equations are

$$D_A = A^+ = A \oplus (U + D)$$

$$D_B = B^+ = B \oplus (UA + DA')$$

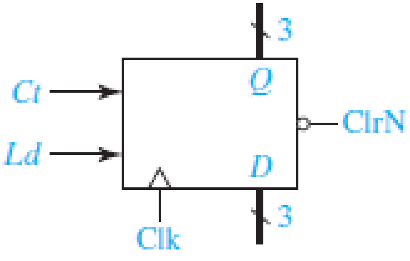
$$D_C = C^+ = C \oplus (UBA + DB'A')$$

Design of Binary Counters

Loadable Counter with Count Enable:

FIGURE 12-20
Loadable Counter with Count Enable

© Cengage Learning 2014



(a)

ClrN	Ld	Ct	C^+	B^+	A^+	
0	X	X	0	0	0	
1	1	X	D_C	D_B	D_A	(load)
1	0	0	C	B	A	(no change)
1	0	1	Present state + 1			

(b)

The next-state equations for the counter of Figure 12-21 are

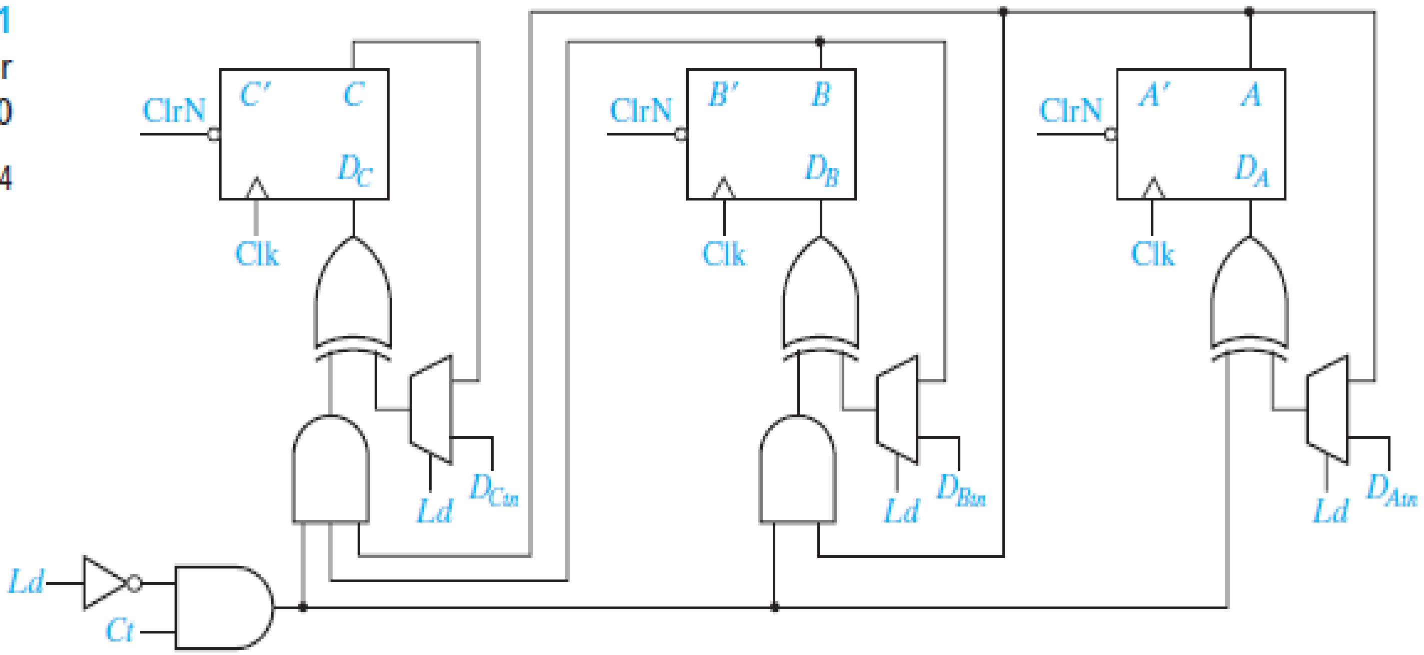
$$A^+ = D_A = (Ld' \cdot A + Ld \cdot D_{Ain}) \oplus Ld' \cdot Ct$$
$$B^+ = D_B = (Ld' \cdot B + Ld \cdot D_{Bin}) \oplus Ld' \cdot Ct \cdot A$$
$$C^+ = D_C = (Ld' \cdot C + Ld \cdot D_{Cin}) \oplus Ld' \cdot Ct \cdot B \cdot A$$

When $Ld = 0$ and $Ct = 1$, these equations reduce to $A^+ = A'$, $B^+ = B \oplus A$, and $C^+ = C \oplus BA$, which are the equations previously derived for a 3-bit counter.

Design of Binary Counters

FIGURE 12-21
Circuit for
Figure 12-20

© Cengage Learning 2014



Counters for Other Sequences

Counter for Other Sequences (Example):

- We will design a counter for the transition table shown in Table 12-3 using T Flip-Flops.

FIGURE 12-22
Transition Graph for Counter

© Cengage Learning 2014

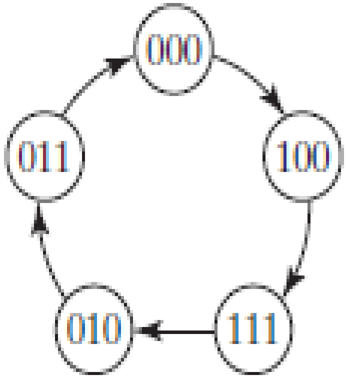


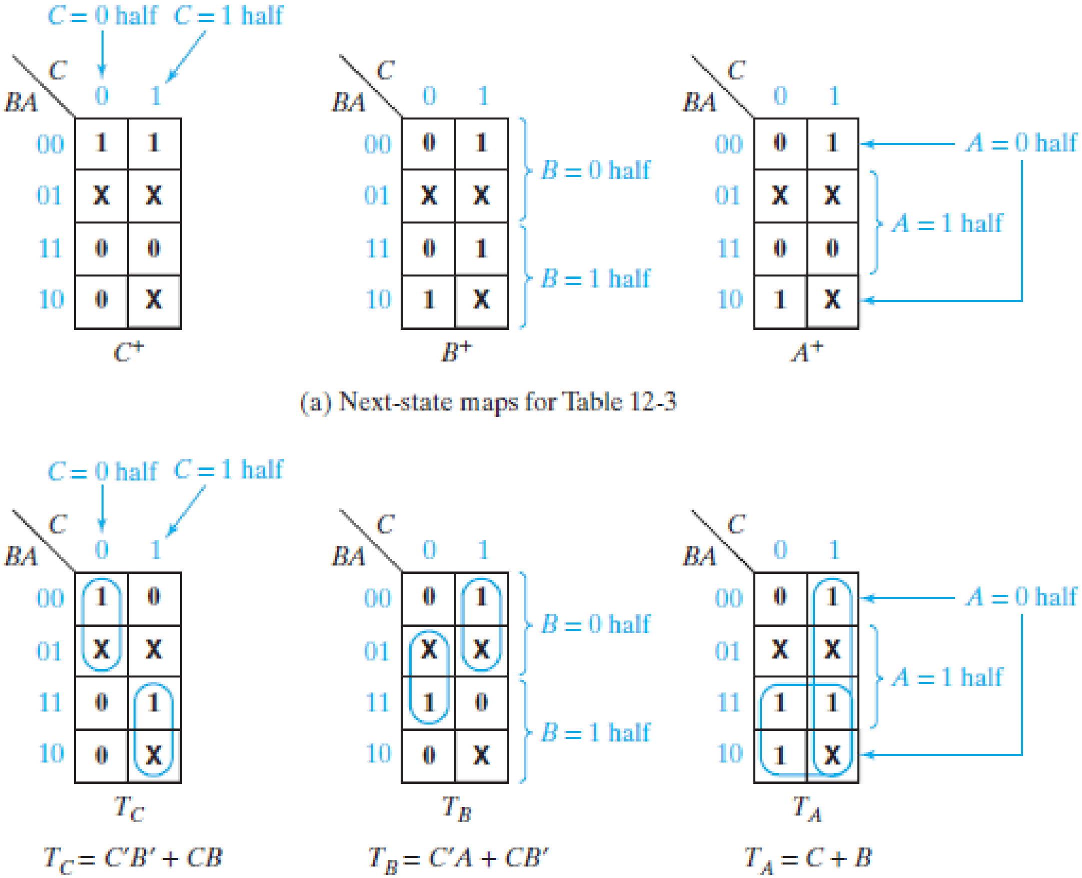
TABLE 12-3
Transition Table for Figure 12-22

© Cengage Learning 2014

C	B	A	C^+	B^+	A^+
0	0	0	1	0	0
0	0	1	–	–	–
0	1	0	0	1	1
0	1	1	0	0	0
1	0	0	1	1	1
1	0	1	–	–	–
1	1	0	–	–	–
1	1	1	0	1	0

Counters for Other Sequences

FIGURE 12-23



C^+

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

B^+

BA \ C	0	1
00	0	1
01	X	X
11	0	1
10	1	X

A^+

BA \ C	0	1
00	0	1
01	X	X
11	0	0
10	1	X

(a) Next-state maps for Table 12-3

T_C

BA \ C	0	1
00	1	0
01	X	X
11	0	1
10	0	X

$T_C = C'B' + CB$

T_B

BA \ C	0	1
00	0	1
01	X	X
11	1	0
10	0	X

$T_B = C'A + CB'$

T_A

BA \ C	0	1
00	0	1
01	X	X
11	1	1
10	1	X

$T_A = C + B$

(b) Derivation of T inputs

Counters for Other Sequences

Input for T Flip-Flop:

- T=1 whenever a state of change is required.

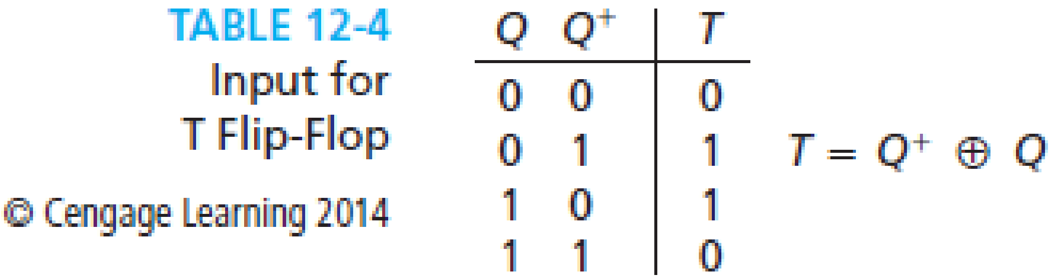


TABLE 12-4 Input for T Flip-Flop

© Cengage Learning 2014

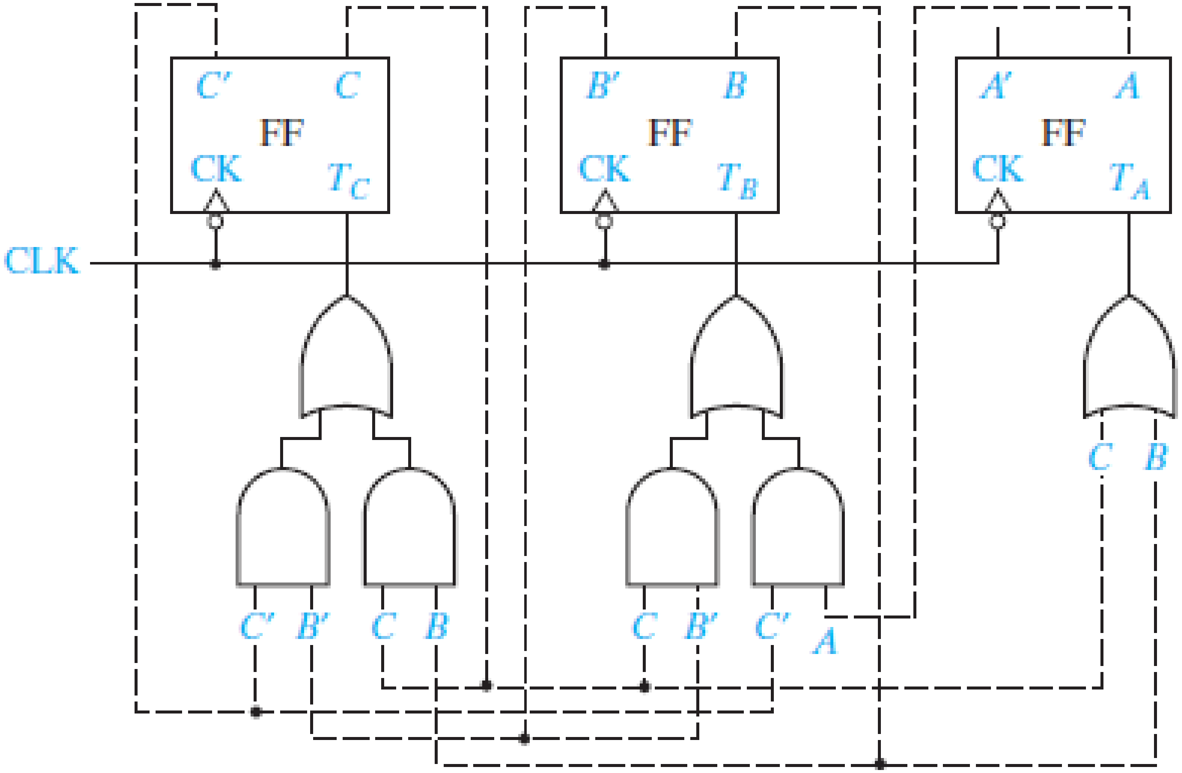
Q	Q^+	T
0	0	0
0	1	1
1	0	1
1	1	0

$$T = Q^+ \oplus Q$$

Counters for Other Sequences

FIGURE 12-24
Counter Using
T Flip-Flops

© Cengage Learning 2014



Counters for Other Sequences

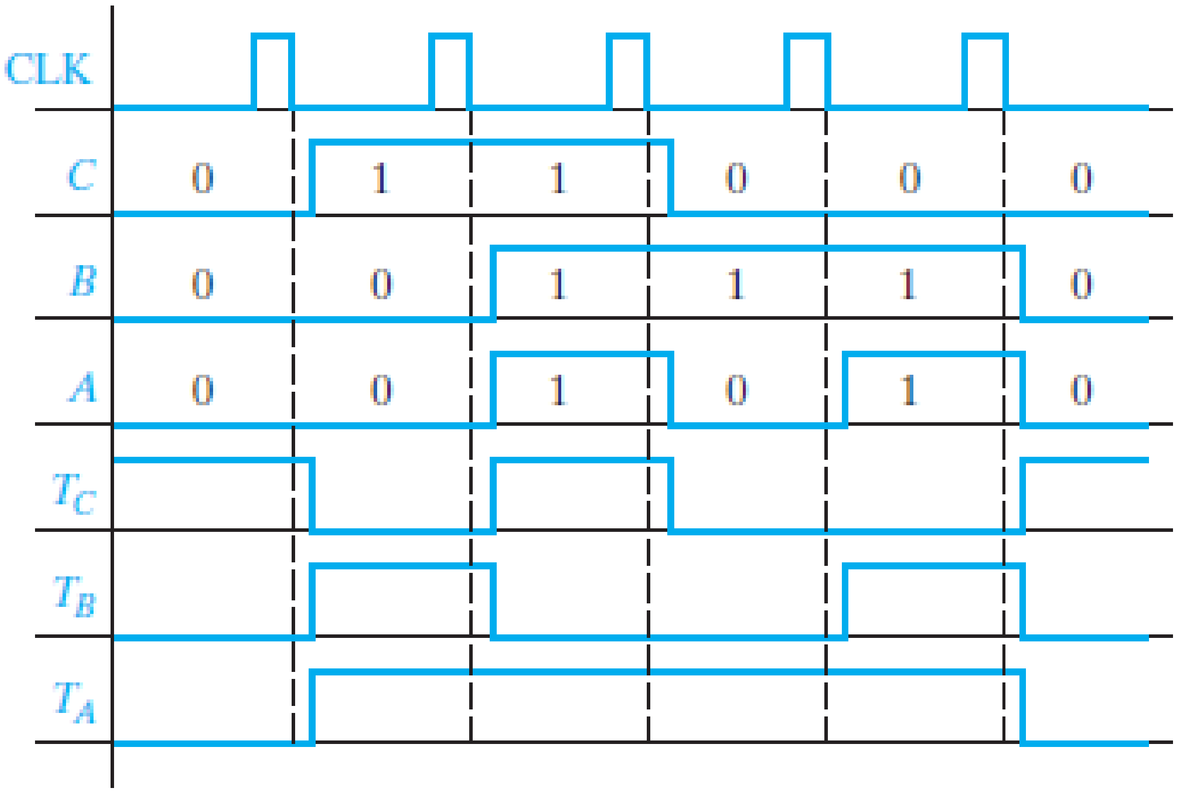
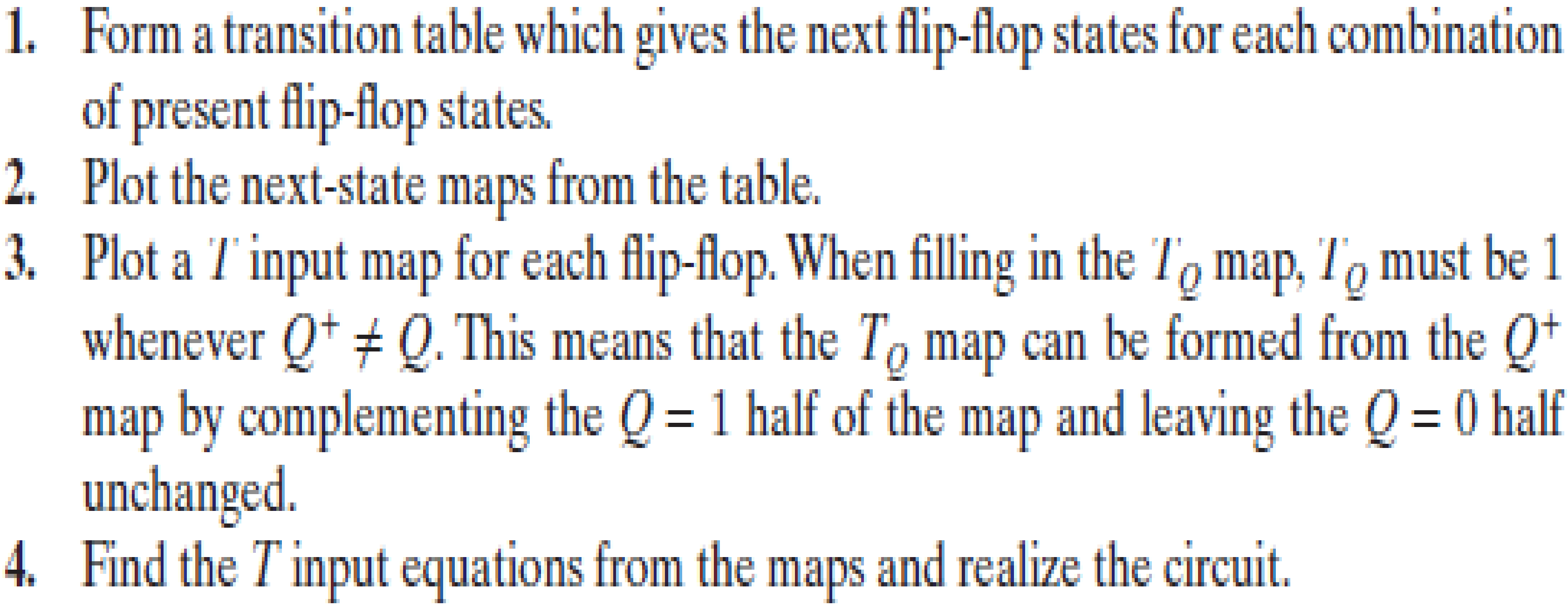


FIGURE 12-25 Timing Diagram for Figure 12-24

© Cengage Learning 2014

Counters for Other Sequences

Procedure for Designing a Counter Using T Flip-Flops:



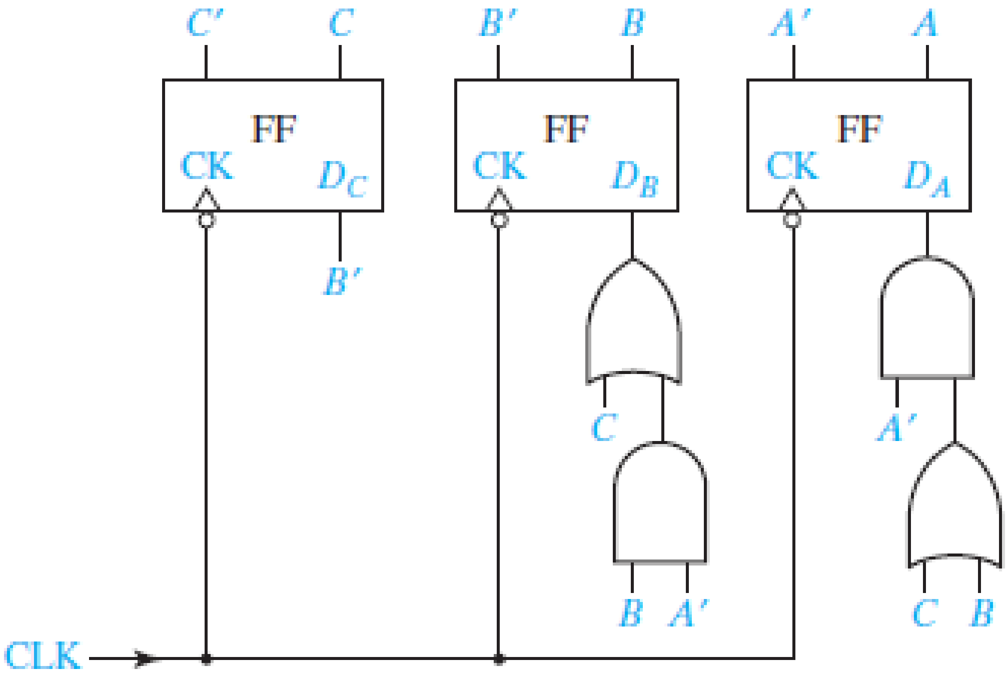
1. Form a transition table which gives the next flip-flop states for each combination of present flip-flop states.
2. Plot the next-state maps from the table.
3. Plot a T input map for each flip-flop. When filling in the T_Q map, T_Q must be 1 whenever $Q^+ \neq Q$. This means that the T_Q map can be formed from the Q^+ map by complementing the $Q = 1$ half of the map and leaving the $Q = 0$ half unchanged.
4. Find the T input equations from the maps and realize the circuit.

Counters for Other Sequences

Counter Design using D Flip-Flops:

FIGURE 12-27 Counter of Figure 12-22 Using D Flip-Flops

© Cengage Learning 2014



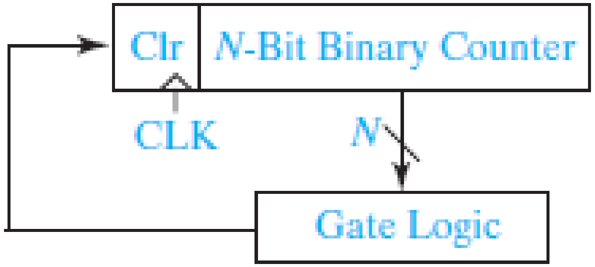
Counters for Other Sequences

Binary Counter with Clear:

- Counters and shift registers with clear, preset, or parallel load capability can also be used to generate nonbinary count cycles.
- Consider a binary counter with a clear input as shown Figure 12-28.

FIGURE 12-28 Binary Counter with Clear

© Cengage Learning 2014



Synchronous Clear:

$$\text{Clr} = Q_3Q_0$$

Asynchronous Clear:

$$\text{Clr} = Q_3Q_1$$

Counters for Other Sequences

Binary Counter with Parallel Load:

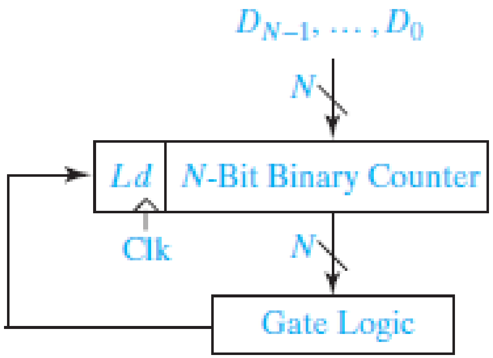


FIGURE 12-29 Binary Counter with Parallel Load

© Cengage Learning 2014

The counter must cycle through states 3 to 12. The logic must generate Ld when the counter is in state 12 and the parallel inputs must be 0011.

$$D_3 = 0, D_2 = 0, D_1 = 1, D_0 = 1$$

$$Ld = Q_3Q_2$$

States 0,1,2,13,14,15 are don't-cares.

Counter Design Using S-R and J-K Flip-Flops

Procedure for Counter Design Using S-R Flip-Flops:

Instead of deriving an input equation for each D or T flip-flop, the S and R input equations must be derived.

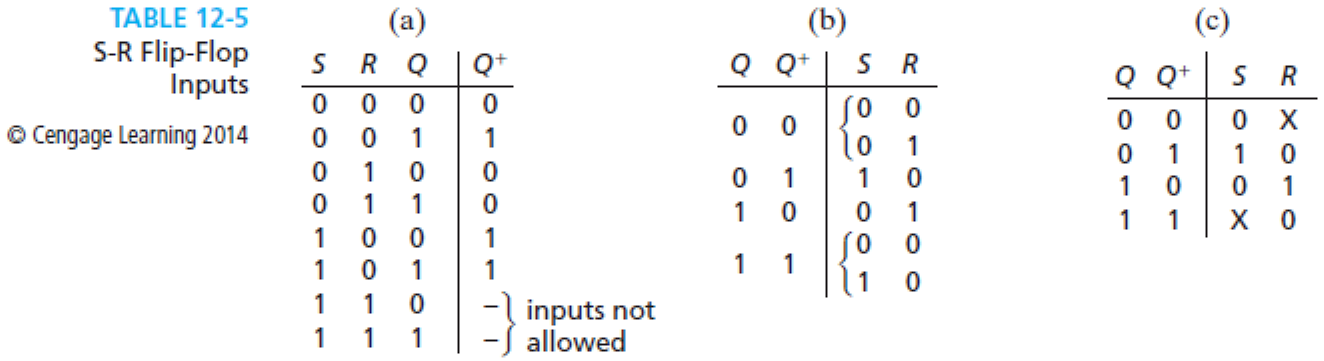


TABLE 12-5
S-R Flip-Flop
Inputs

© Cengage Learning 2014

(a)

S	R	Q	Q⁺	
0	0	0	0	
0	0	1	1	
0	1	0	0	
0	1	1	0	
1	0	0	1	
1	0	1	1	
1	1	0	–	inputs not allowed
1	1	1	–	

(b)

Q	Q⁺	S	R
0	0	0 0	0 1
0	1	1	0
1	0	0	1
1	1	0 1	0 0

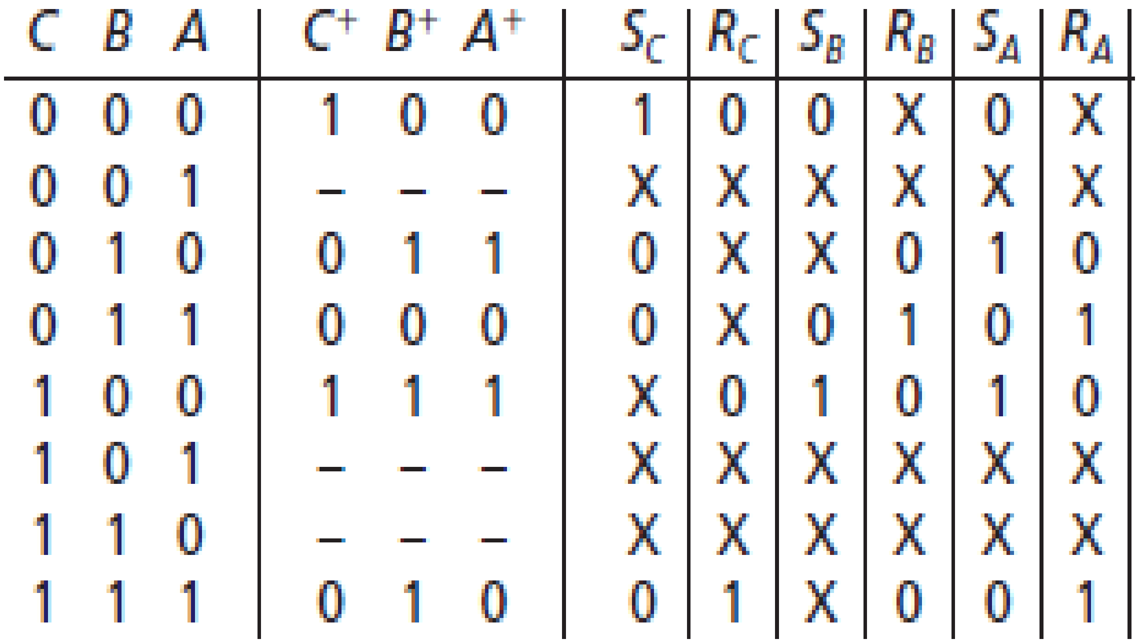
(c)

Q	Q⁺	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

Counter Design Using S-R and J-K Flip-Flops

TABLE 12-6

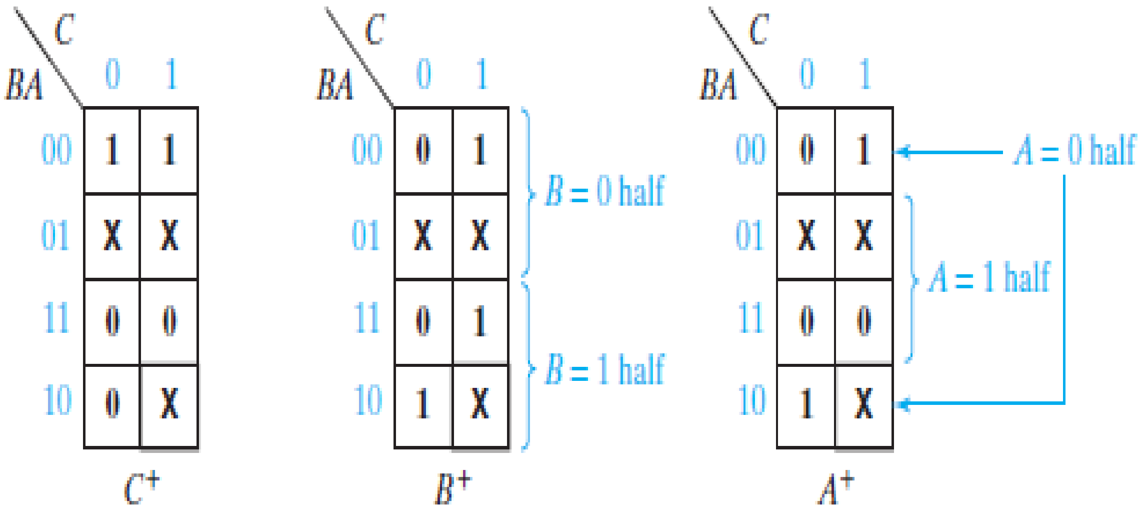
© Cengage Learning 2014



C	B	A	C^+	B^+	A^+	S_C	R_C	S_B	R_B	S_A	R_A
0	0	0	1	0	0	1	0	0	X	0	X
0	0	1	–	–	–	X	X	X	X	X	X
0	1	0	0	1	1	0	X	X	0	1	0
0	1	1	0	0	0	0	X	0	1	0	1
1	0	0	1	1	1	X	0	1	0	1	0
1	0	1	–	–	–	X	X	X	X	X	X
1	1	0	–	–	–	X	X	X	X	X	X
1	1	1	0	1	0	0	1	X	0	0	1

Counter Design Using S-R and J-K Flip-Flops

Counter Using S-R Flip Flops:



C^+

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

B^+

BA \ C	0	1
00	0	1
01	X	X
11	0	1
10	1	X

A^+

BA \ C	0	1
00	0	1
01	X	X
11	0	0
10	1	X

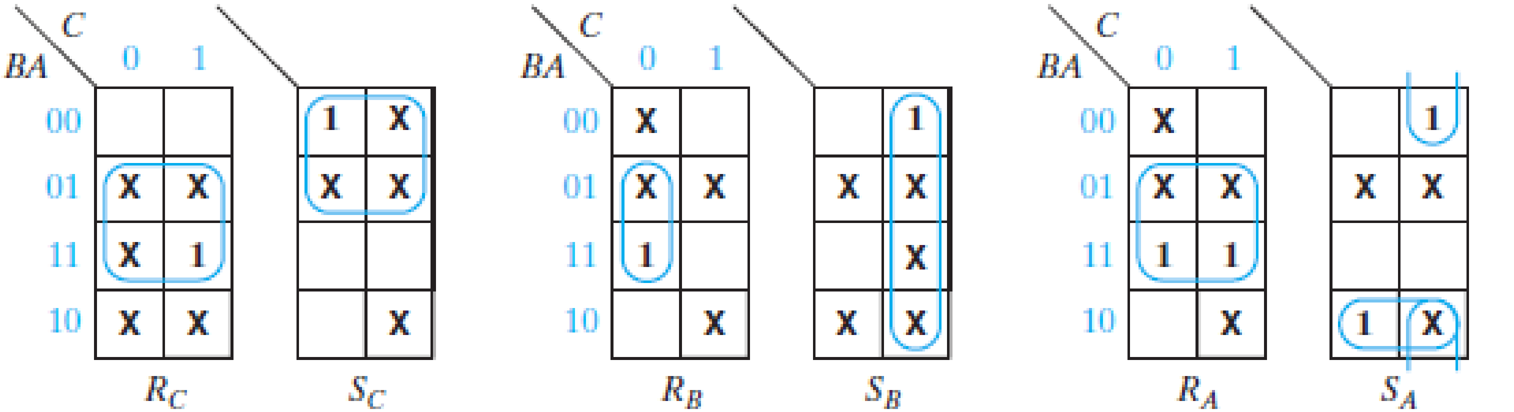
(a) Next-state maps

FIGURE 12-30 Counter of Figure 12-22 Using S-R Flip-Flops

© Cengage Learning 2014

Counter Design Using S-R and J-K Flip-Flops

Counter Using S-R Flip Flops (continued):



BA \ C	0	1
00		
01	X	X
11	X	1
10	X	X

$R_C = A$

BA \ C	0	1
00	1	X
01	X	X
11		
10		X

$S_C = B'$

BA \ C	0	1
00	X	
01	X	X
11	1	
10		X

$R_B = C'A$

BA \ C	0	1
00		1
01	X	X
11		X
10	X	X

$S_B = C$

BA \ C	0	1
00	X	
01	X	X
11	1	1
10		X

$R_A = A$

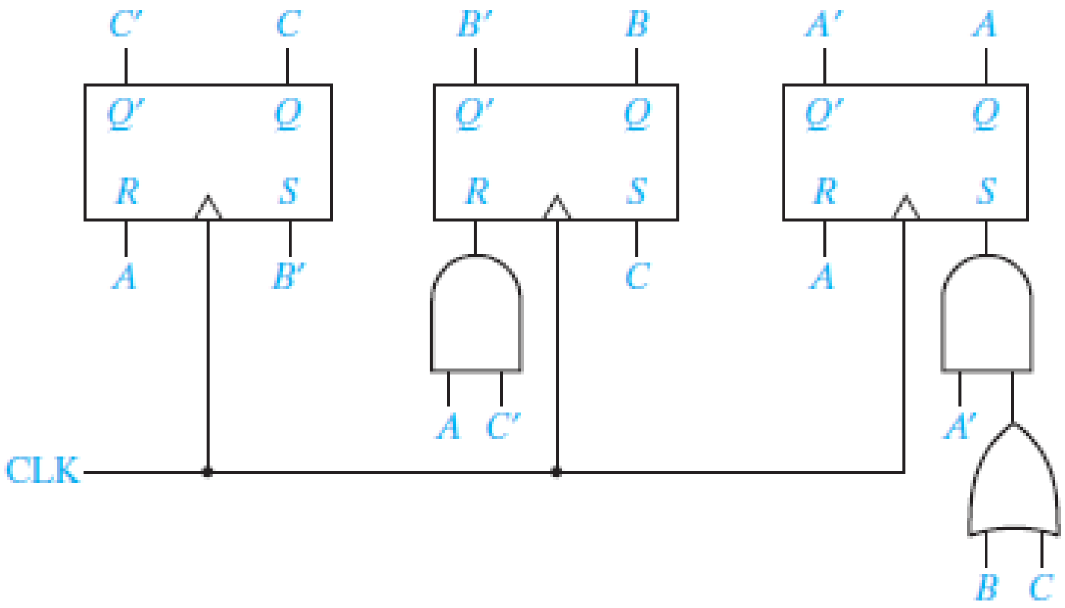
BA \ C	0	1
00		1
01	X	X
11		
10	1	X

$S_A = CA' + BA'$
$= A'(C + B)$

(b) S-R flip-flop equations

Counter Design Using S-R and J-K Flip-Flops

Counter Using S-R Flip-Flops (continued):



(c) Logic circuit

Counter Design Using S-R and J-K Flip-Flops

Procedure for Counter Design Using J-K Flip- Flops:

❖The procedure used to design a counter with J-K flip-flops is very similar to that used for S-R flip-flops, except that J and K can be 1 simultaneously, in which case the flip-flop changes state.

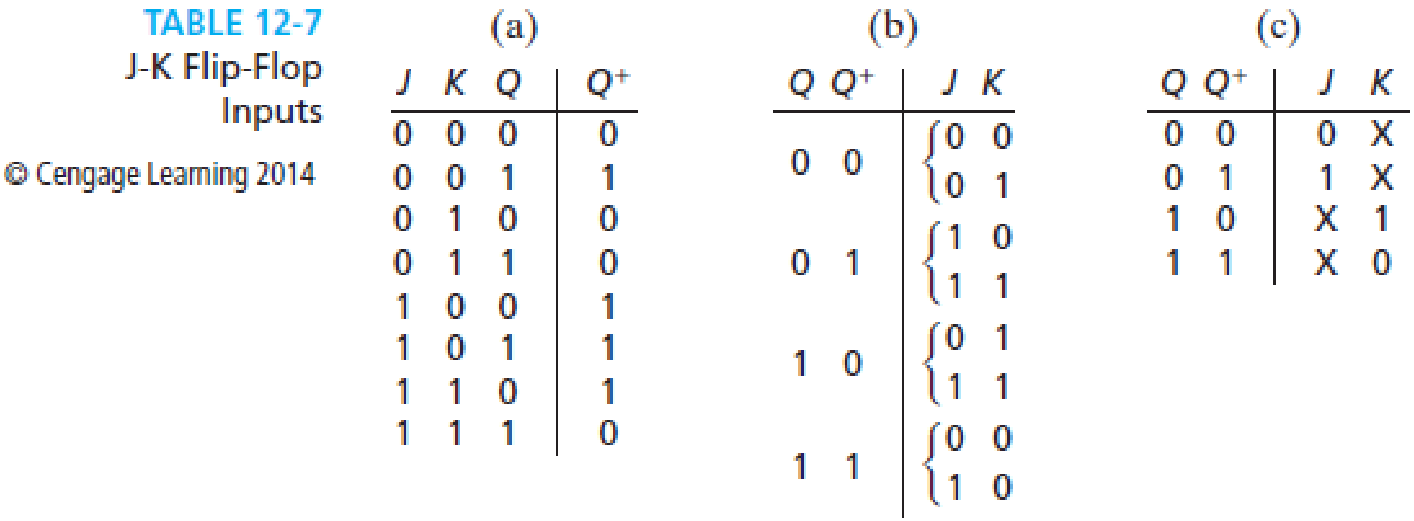


TABLE 12-7
J-K Flip-Flop Inputs

© Cengage Learning 2014

(a)

J	K	Q	Q^+
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

(b)

Q	Q^+	J	K
0	0	0 0	0 1
0	1	1 1	0 1
1	0	0 1	1 1
1	1	0 1	0 0

(c)

Q	Q^+	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Counter Design Using S-R and J-K Flip-Flops

J-K Flip-Flop Table:

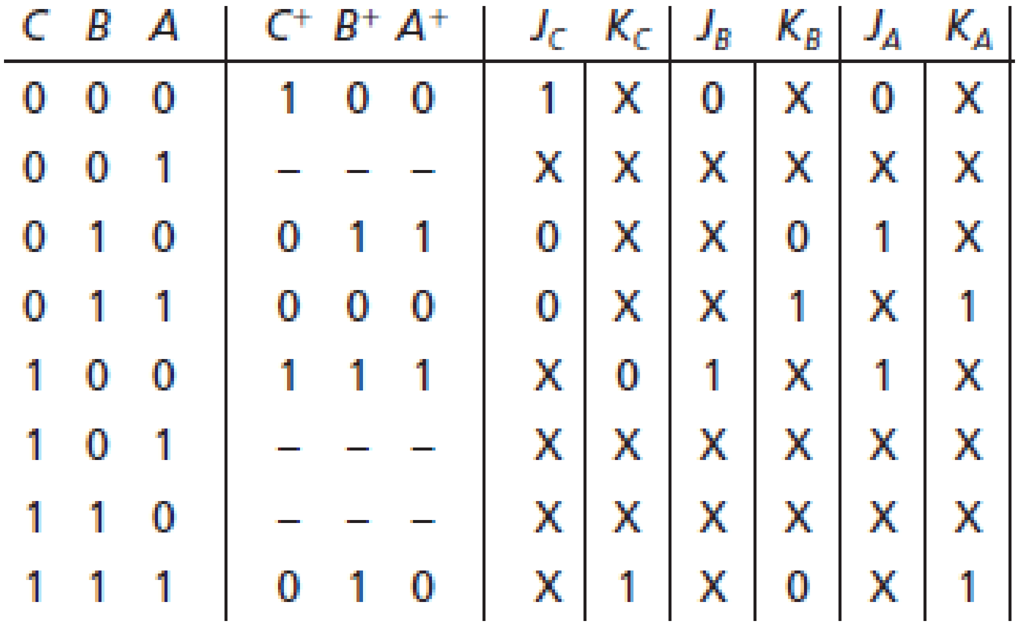


TABLE 12-8

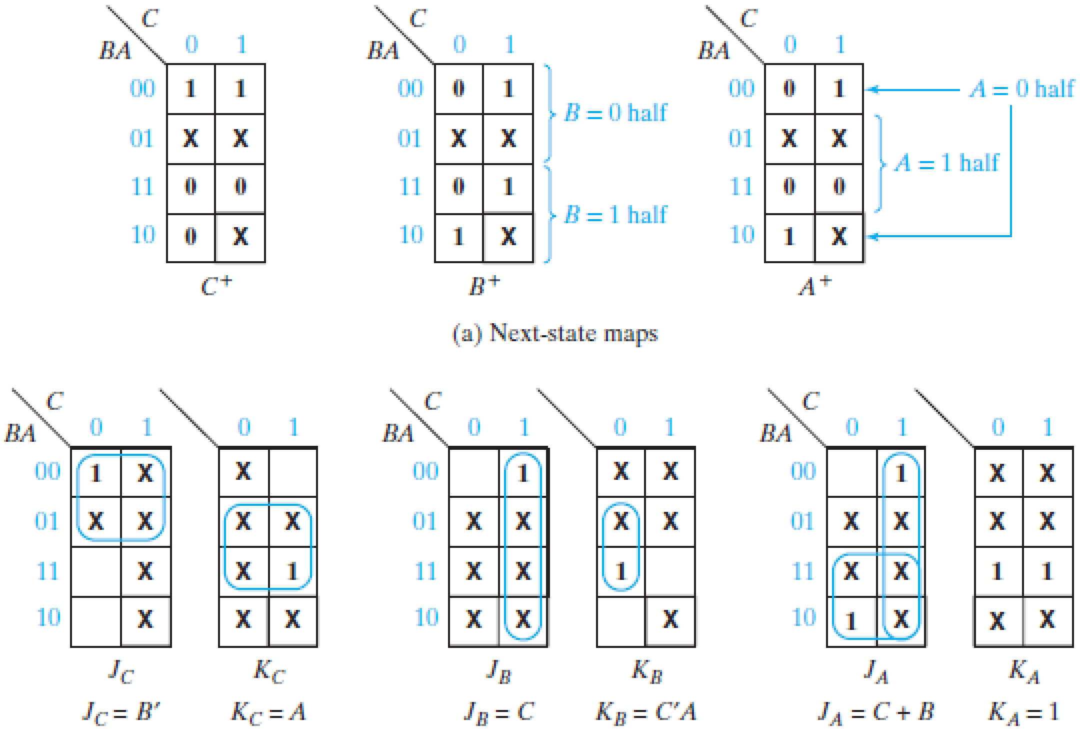
© Cengage Learning 2014

C	B	A	C^+	B^+	A^+	J_C	K_C	J_B	K_B	J_A	K_A
0	0	0	1	0	0	1	X	0	X	0	X
0	0	1	–	–	–	X	X	X	X	X	X
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	0	0	0	0	X	X	1	X	1
1	0	0	1	1	1	X	0	1	X	1	X
1	0	1	–	–	–	X	X	X	X	X	X
1	1	0	–	–	–	X	X	X	X	X	X
1	1	1	0	1	0	X	1	X	0	X	1

Counter Design Using S-R and J-K Flip-Flops

Counter Design Using J-K Flip-Flop:

FIGURE 12-31 Counter of Figure 12-22 Using J-K Flip-Flops



C^+

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

B^+

BA \ C	0	1
00	0	1
01	X	X
11	0	1
10	1	X

A^+

BA \ C	0	1
00	0	1
01	X	X
11	0	0
10	1	X

(a) Next-state maps

J_C

BA \ C	0	1
00	1	X
01	X	X
11		X
10		X

K_C

BA \ C	0	1
00	X	
01	X	X
11	X	1
10	X	X

J_B

BA \ C	0	1
00		1
01	X	X
11	X	X
10	X	X

K_B

BA \ C	0	1
00	X	X
01	X	X
11	1	
10		X

J_A

BA \ C	0	1
00		1
01	X	X
11	X	X
10	1	X

K_A

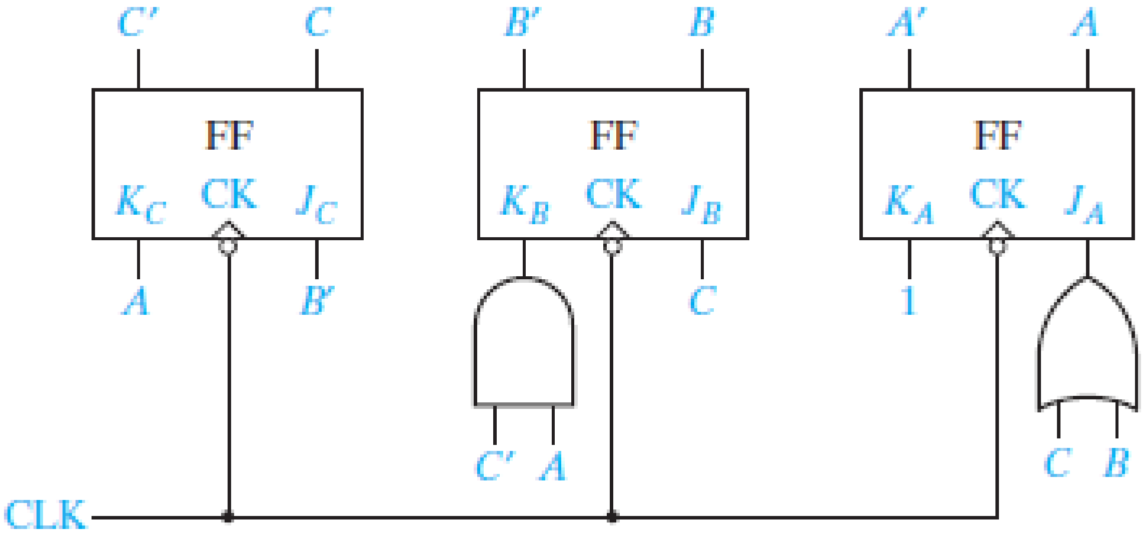
BA \ C	0	1
00	X	X
01	X	X
11	1	1
10	X	X

$J_C = B'$ $K_C = A$ $J_B = C$ $K_B = C'A$ $J_A = C + B$ $K_A = 1$

(b) J-K flip-flop input equations

Counter Design Using S-R and J-K Flip-Flops

Counter Design Using J-K Flip-Flop (continued):



(c) Logic circuit (omitting the feedback lines)

Derivation of Flip-Flop Input Equations- Summary

Summary:

- The input equation for the flip-flops in a sequential circuit may be derived from the next-state equations by using truth tables or by using Karnaugh maps.
- For the D flip-flop, the input is the same as the next state.
- For the T flip-flop, the input is 1 whenever a state change is required.
- For the S-R flip-flop, S is 1 whenever the flip-flop must be set to 1 and R is 1 when it must be reset to 0.
- For a J-K flip-flop, the J and K inputs are the same as S and R, respectively, except that when one input is 1 the other input is X.

Derivation of Flip-Flop Input Equations- Summary

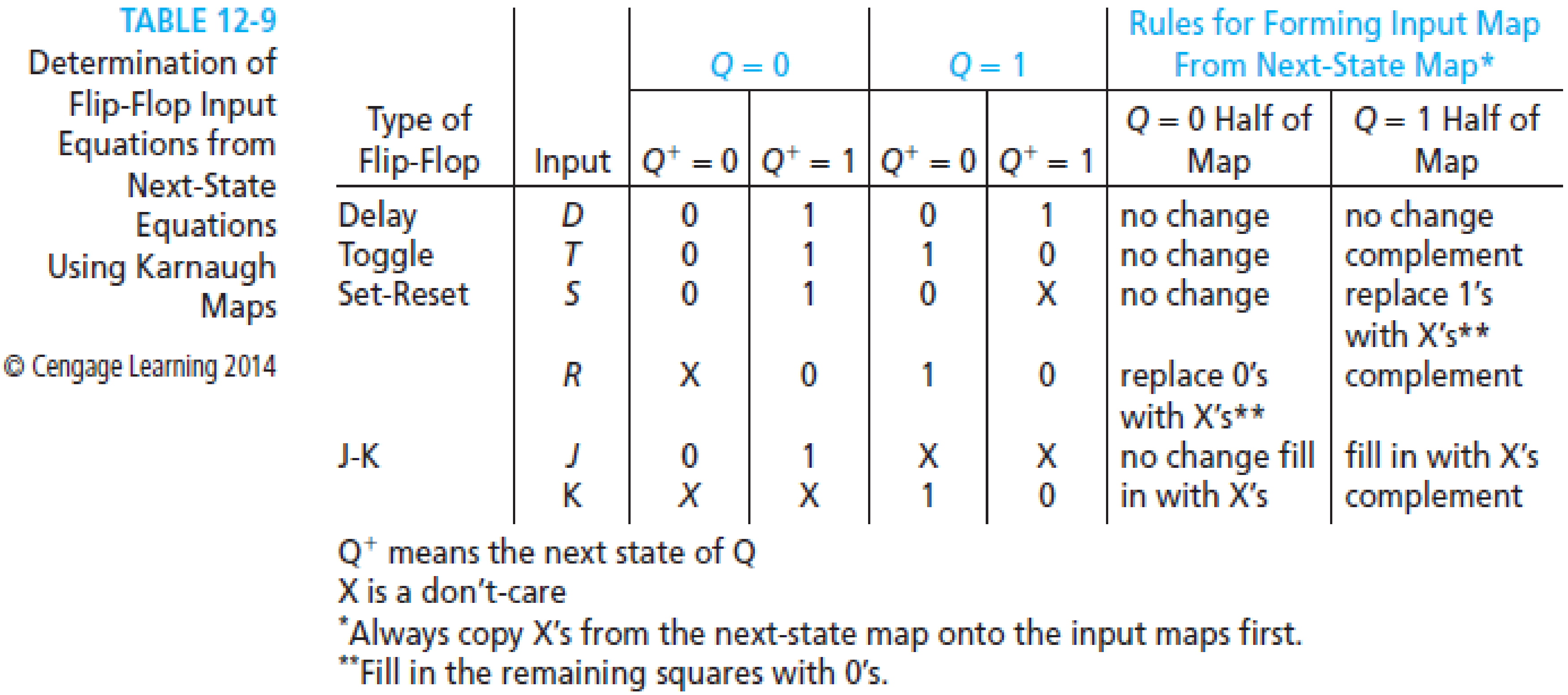


TABLE 12-9 Determination of Flip-Flop Input Equations from Next-State Equations Using Karnaugh Maps

© Cengage Learning 2014

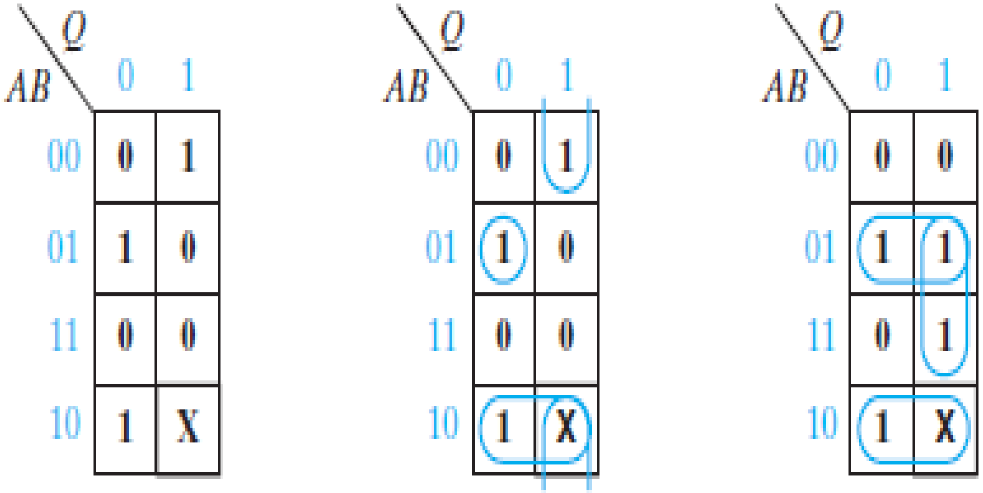
Type of Flip-Flop	Input	$Q = 0$		$Q = 1$		Rules for Forming Input Map From Next-State Map*	
		$Q^+ = 0$	$Q^+ = 1$	$Q^+ = 0$	$Q^+ = 1$	$Q = 0$ Half of Map	$Q = 1$ Half of Map
Delay	D	0	1	0	1	no change	no change
Toggle	T	0	1	1	0	no change	complement
Set-Reset	S	0	1	0	X	no change	replace 1's with X's**
	R	X	0	1	0	replace 0's with X's**	complement
J-K	J	0	1	X	X	no change fill	fill in with X's
	K	X	X	1	0	in with X's	complement

Q^+ means the next state of Q
X is a don't-care
*Always copy X's from the next-state map onto the input maps first.
**Fill in the remaining squares with 0's.

Derivation of Flip-Flop Input Equations- Summary

Example 1:

Example (illustrating the use of Table 12-9)



AB \ Q	0	1
00	0	1
01	1	0
11	0	0
10	1	X

Q^+

Next-state map

AB \ Q	0	1
00	0	1
01	1	0
11	0	0
10	1	X

$D = Q'A'B + QB' + AB'$

D input map

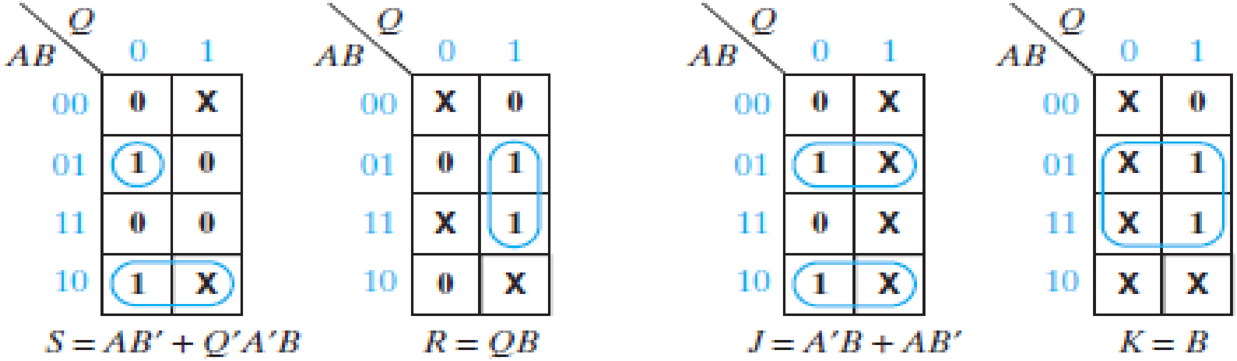
AB \ Q	0	1
00	0	0
01	1	1
11	0	1
10	1	X

$T = A'B + AB' + QB$

T input map

Derivation of Flip-Flop Input Equations- Summary

Example 1 (continued):



AB \ Q	0	1
00	0	X
01	1	0
11	0	0
10	1	X

$S = AB' + Q'A'B$

AB \ Q	0	1
00	X	0
01	0	1
11	X	1
10	0	X

$R = QB$

S-R input maps

AB \ Q	0	1
00	0	X
01	1	X
11	0	X
10	1	X

$J = A'B + AB'$

AB \ Q	0	1
00	X	0
01	X	1
11	X	1
10	X	X

$K = B$

J-K input maps

For the S-R flip-flop, note that when $Q = 0$, $R = X$ if $Q^+ = 0$; and when $Q = 1$, $R = 1$ if $Q^+ = 0$. Therefore, to form the R map from the Q^+ map, replace 0's with X's on the $Q = 0$ half of the map and replace 0's with 1's on the $Q = 1$ half (and fill in 0's for the remaining entries). Similarly, to form the S map from the Q^+ map, copy the 1's on the $Q = 0$ half of the map, and replace the 1's with X's on the $Q = 1$ half.