Chapter No:8
Counters

TOPICS

1. Asynchronous Counters
2. Synchronous Counters
3. Cascaded Counters
4. Counter Decoding

Counters are classified into two broad categories according to the way they are clocked: asynchronous and synchronous.

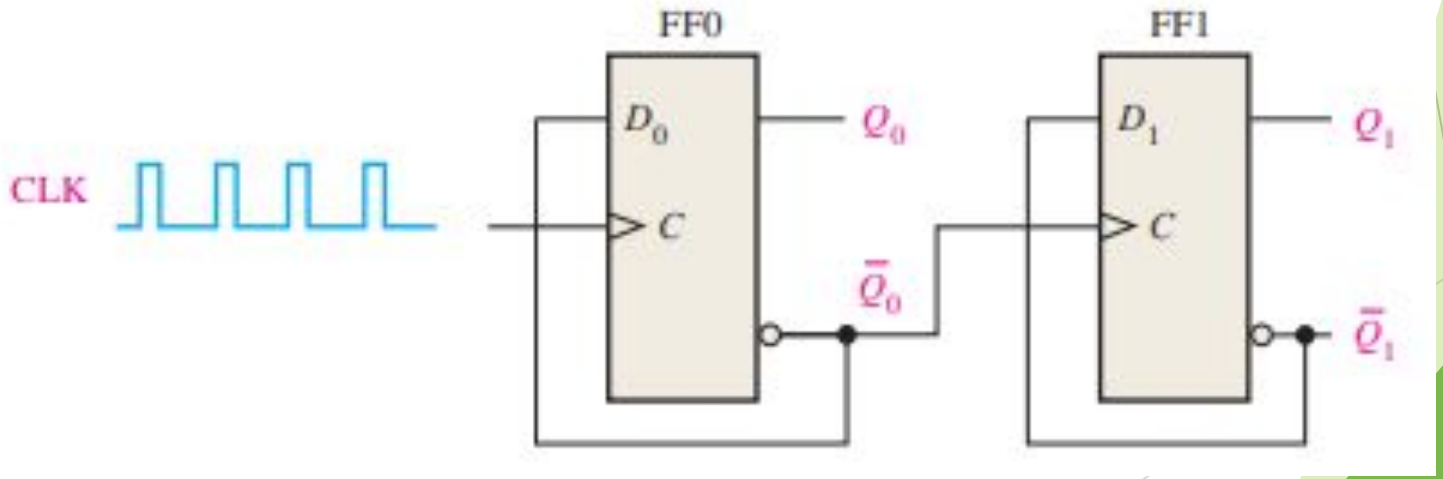
In asynchronous counters, commonly called ripple counters, the first flip-flop is clocked by the external clock pulse and then each successive flip-flop is clocked by the output of the preceding flip-flop.

In synchronous counters, the clock input is connected to all of the flip-flops so that they are clocked simultaneously.

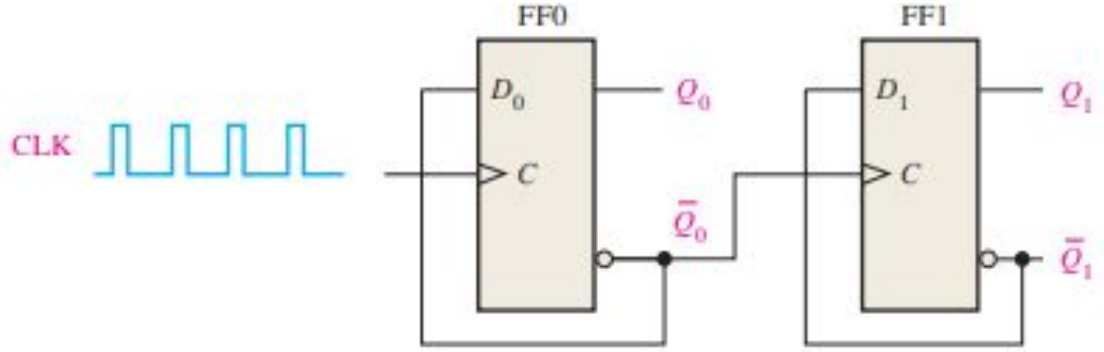
Asynchronous Counters

An asynchronous counter is one in which the flip-flops (FF) within the counter do not change states at exactly the same time because they do not have a common clock pulse.

A 2-Bit Asynchronous Binary Counter

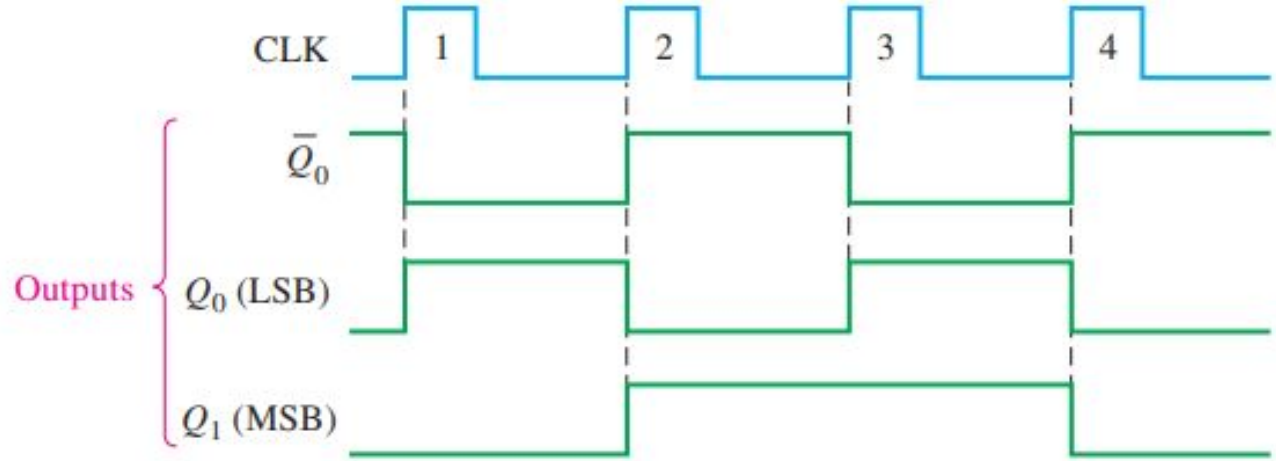


Asynchronous Counters



Both flip-flops are connected for toggle operation (D = Q) and are assumed to be initially RESET (Q LOW).

The positive-going edge of CLK1 (clock pulse 1) causes the Q0 output of FF0 to go HIGH.

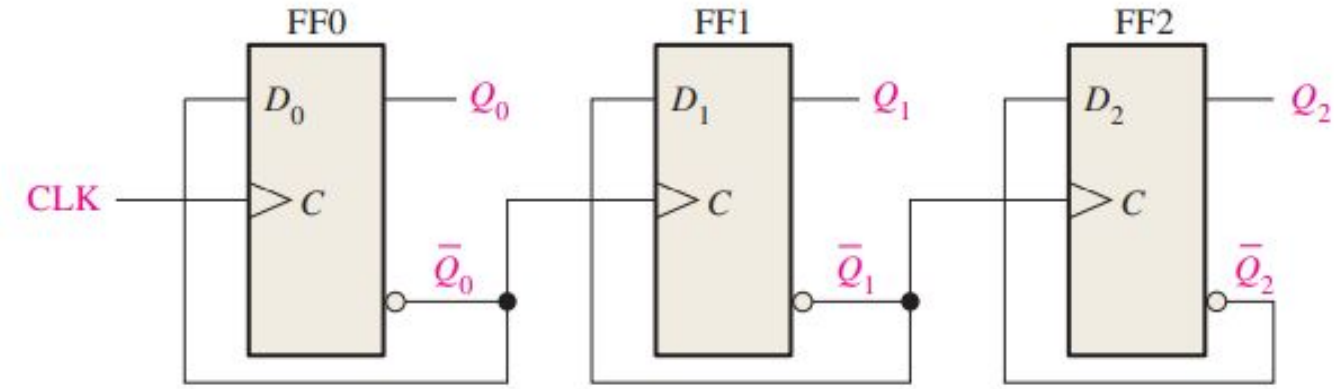


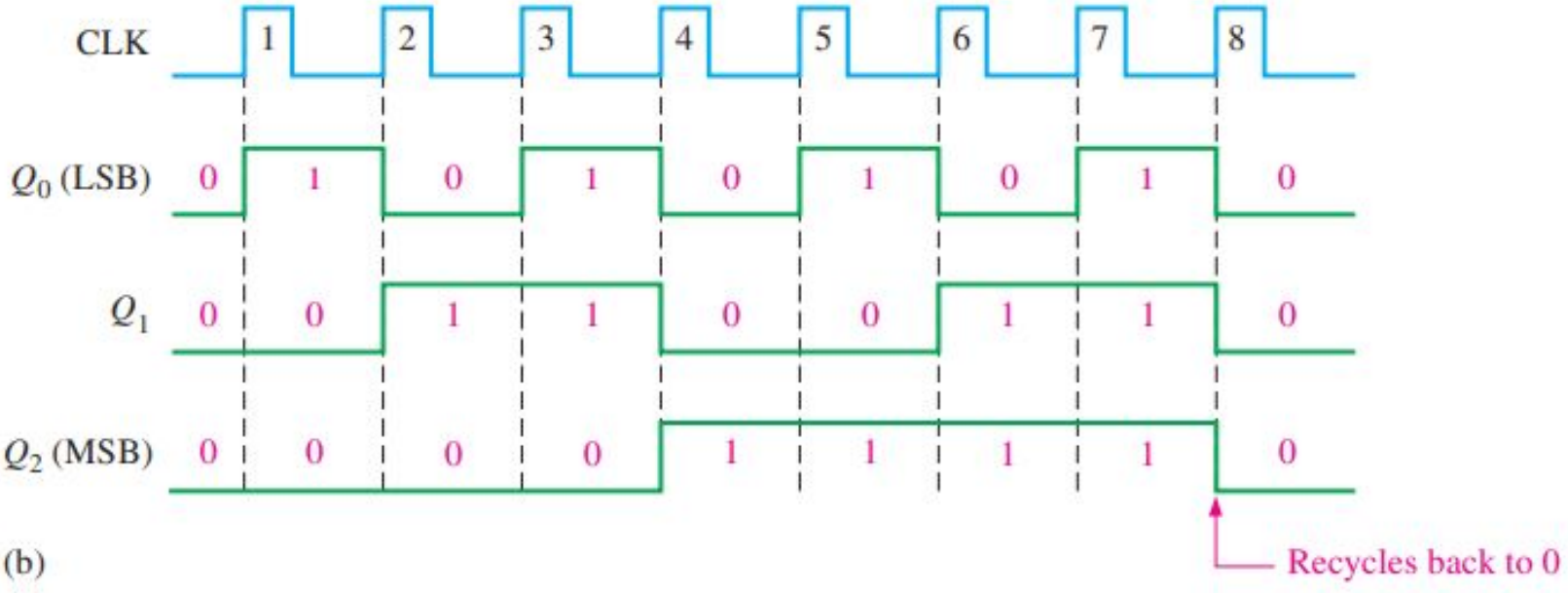
Asynchronous Counters

Clock Pulse	Q_1	Q_0
Initially	0	0
1	0	1
2	1	0
3	1	1
4 (recycles)	0	0

The term recycle is commonly applied to counter operation; it refers to the transition of the counter from its final state back to its original state.

A 3-Bit Asynchronous Binary Counter





(b)

Propagation Delay

Asynchronous counters are commonly referred to as ripple counters for the following reason:

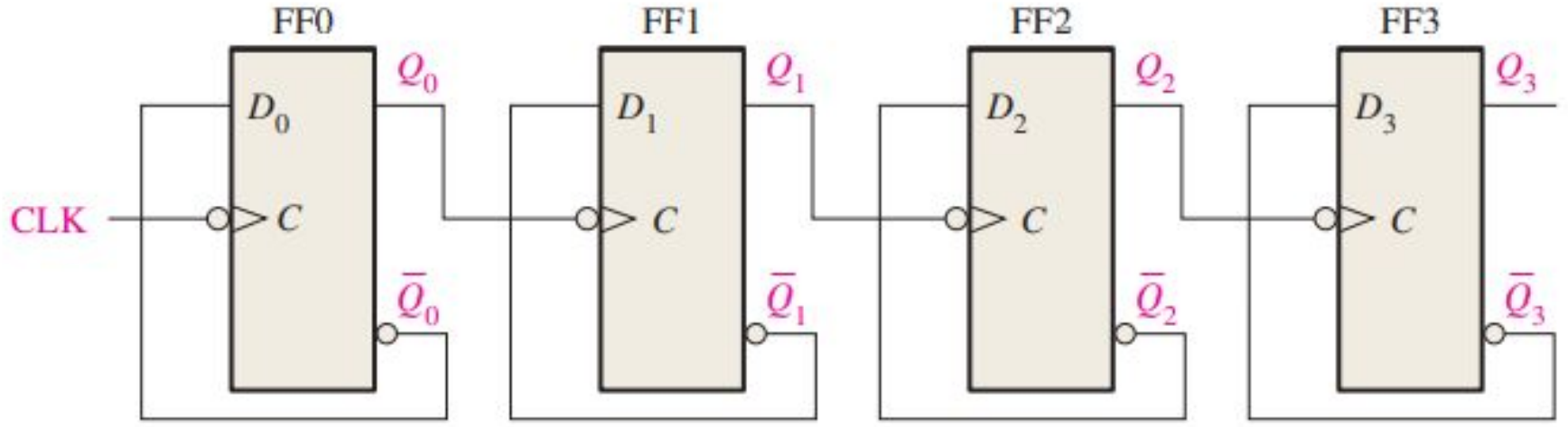
The effect of the input clock pulse is first "felt" by FF0. This effect cannot get to FF1 immediately because of the propagation delay through FF0.

Then there is the propagation delay through FF1 before FF2 can be triggered. Thus, the effect of an input clock pulse "ripples" through the counter, taking some time, due to propagation delays, to reach the last flip-flop.

EXAMPLE

A 4-bit asynchronous binary counter is shown in Figure 9-8(a). Each D flip-flop is negative edge-triggered and has a propagation delay for 10 nanoseconds (ns).

Determine the total propagation delay time from the triggering edge of a clock pulse until a corresponding change can occur in the state of Q3. Also determine the maximum clock frequency at which the counter can be operated.



Solution

For the total delay time, the effect of CLK8 or CLK16 must propagate through four flip-flops before Q3 changes, so

$$t_{p(tot)} = 4 \times 10\,\text{ns} = \mathbf{40\,ns}$$

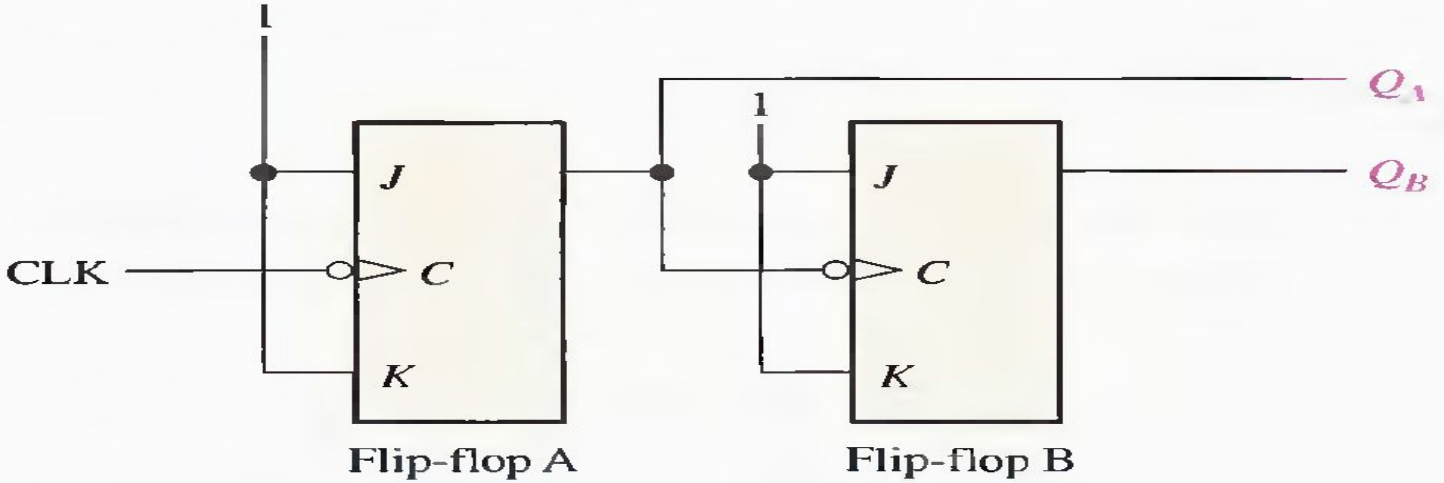
The maximum clock frequency is

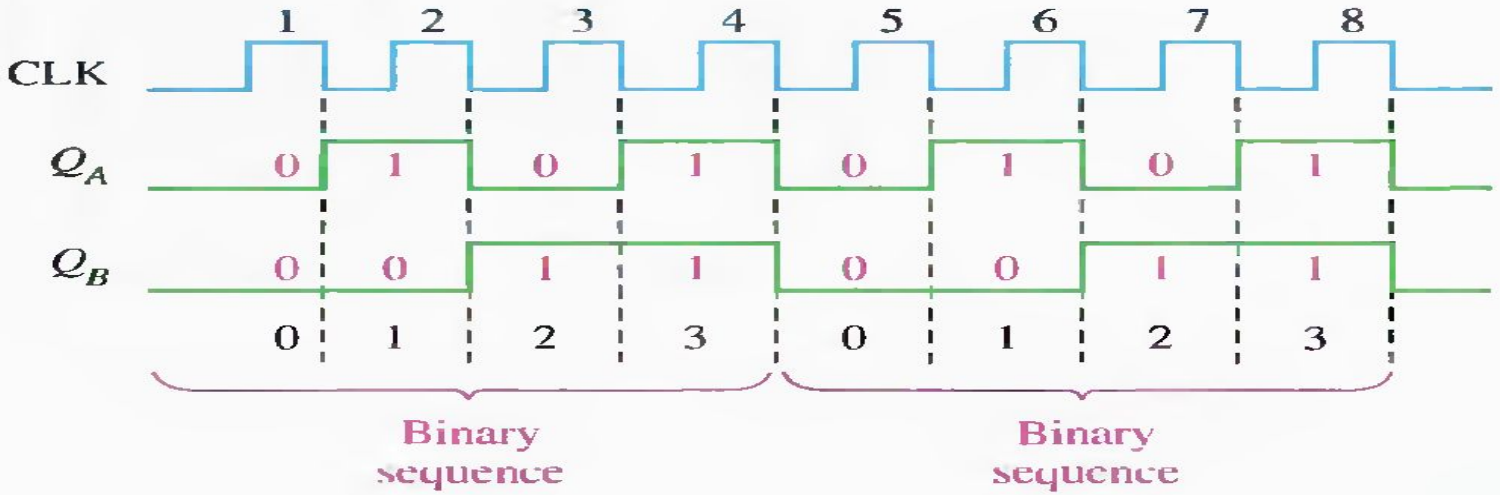
$$f_{\max} = \frac{1}{t_{p(tot)}} = \frac{1}{40\,\text{ns}} = \mathbf{25\,MHz}$$

The counter should be operated below this frequency to avoid problems due to the propagation delay.

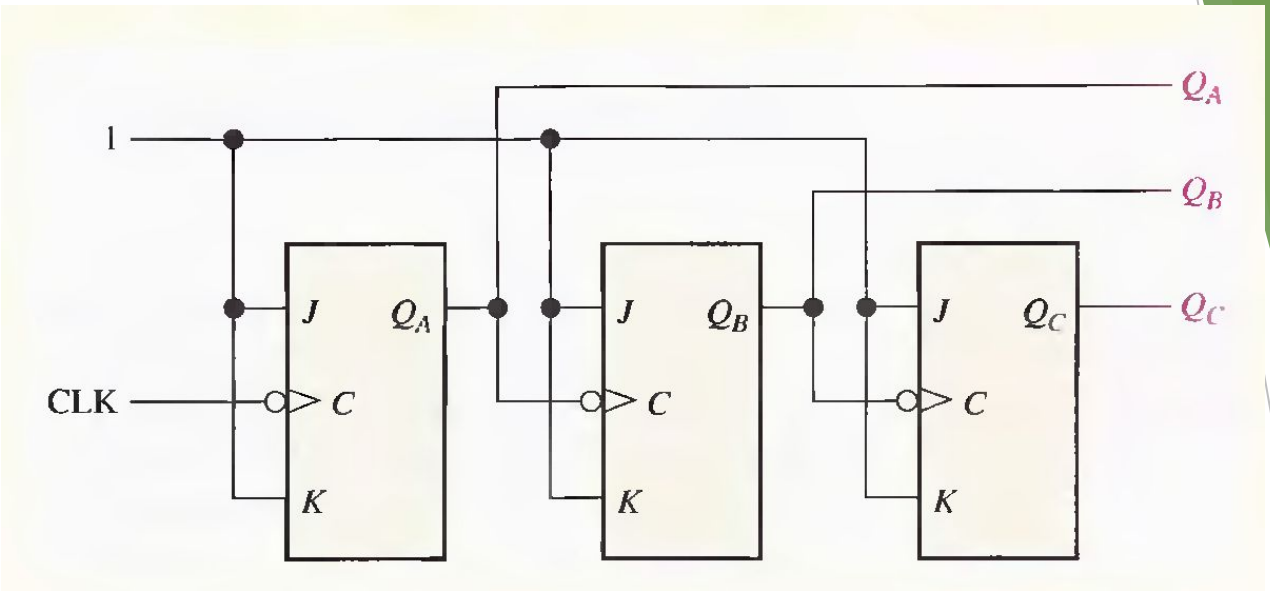
Counting

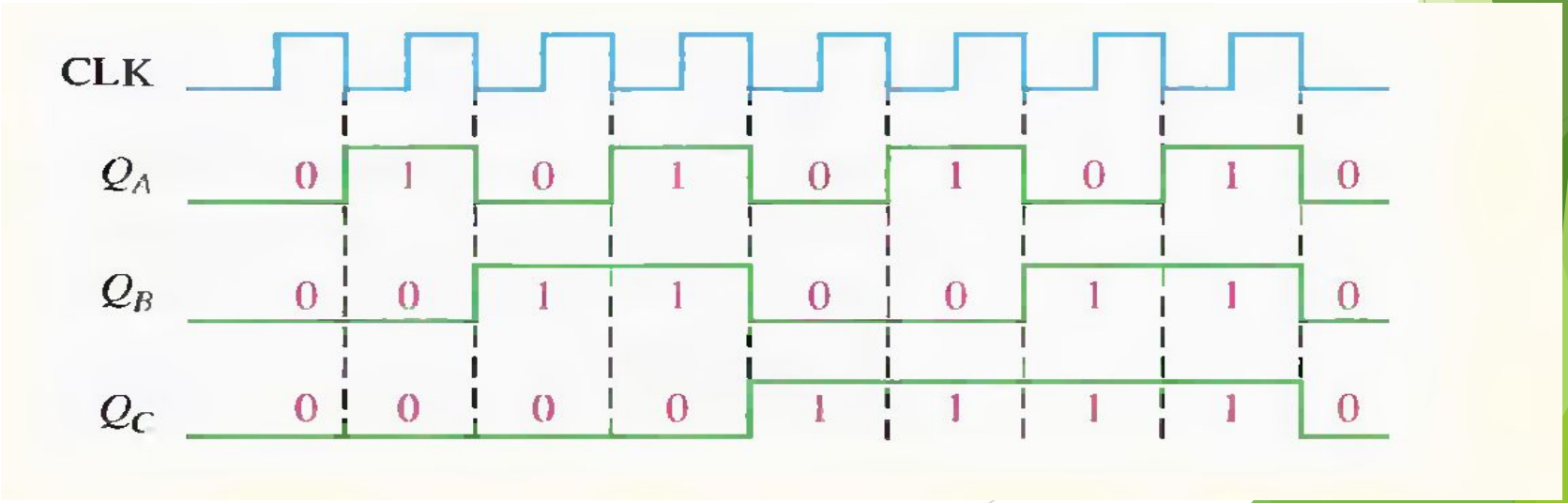
Flip-flops used to generate a binary count sequence. Two repetitions (00, 01, 10, 11) are shown.



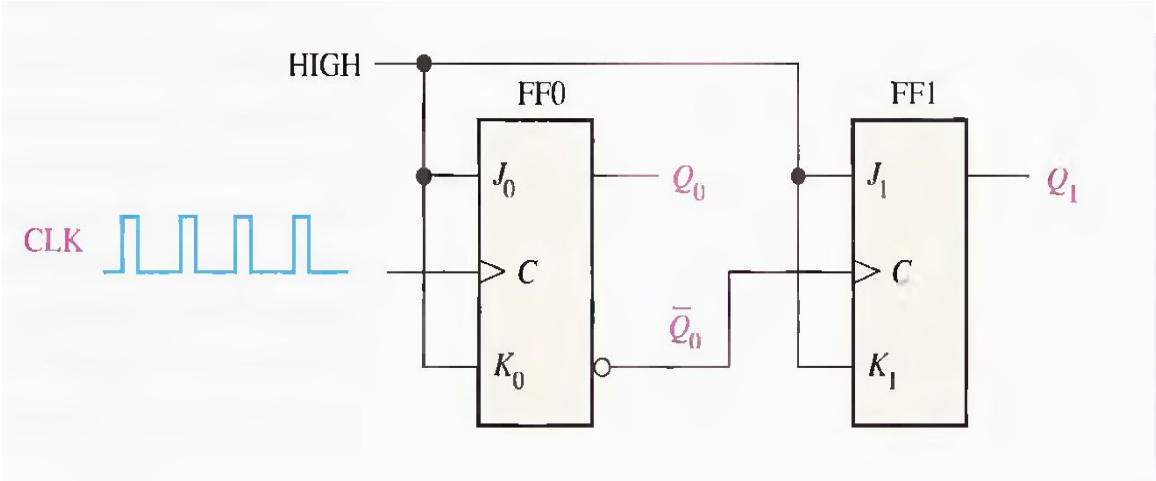


Determine the output waveforms in relation to the clock for Q_A, Q_B, and Q_C.

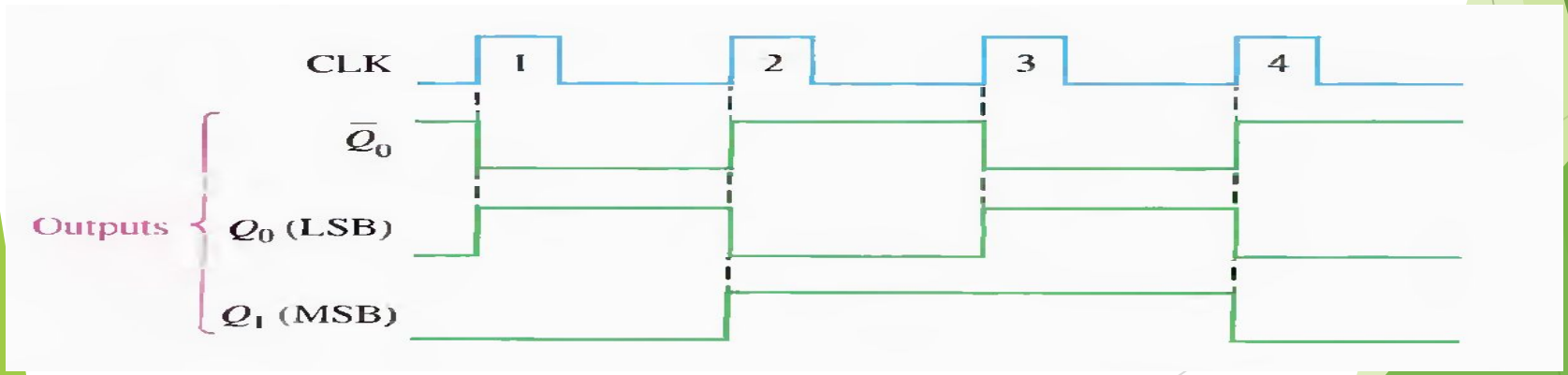




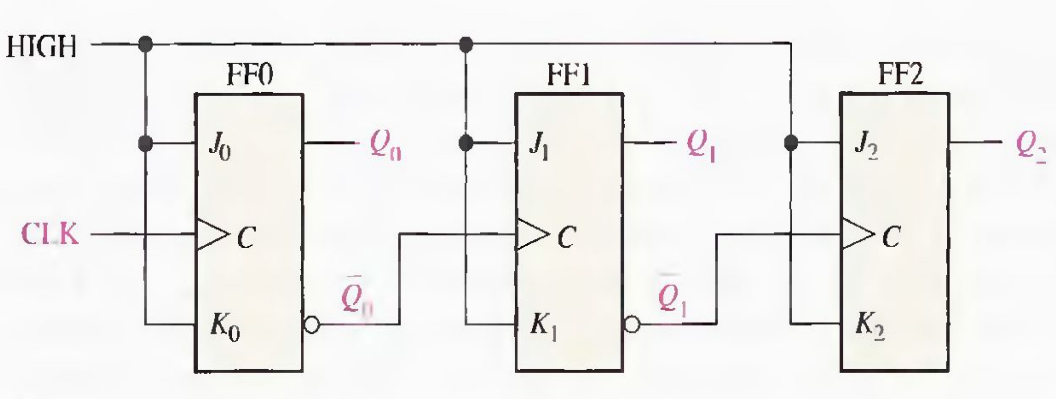
A 2-Bit Asynchronous Binary Counter



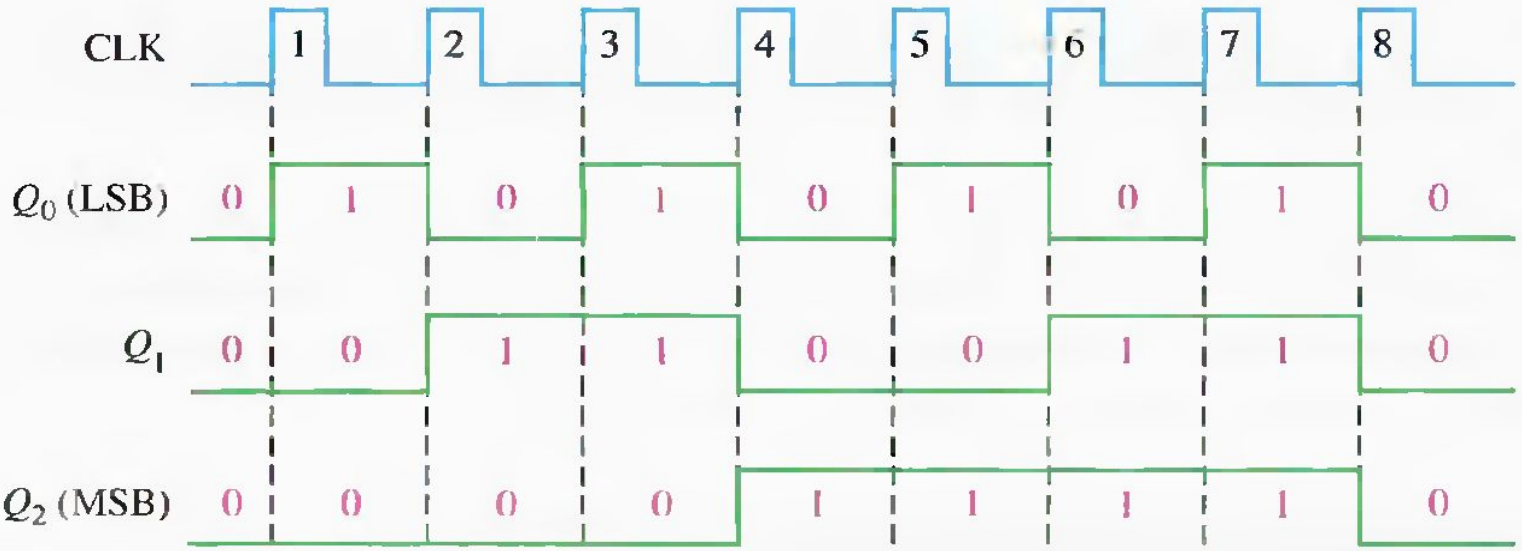
CLOCK PULSE	Q_1	Q_0
Initially	0	0
1	0	1
2	1	0
3	1	1
4 (recycles)	0	0



A 3-Bit Asynchronous Binary Counter



CLOCK PULSE	Q_2	Q_1	Q_0
Initially	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8 (recycles)	0	0	0



(b)

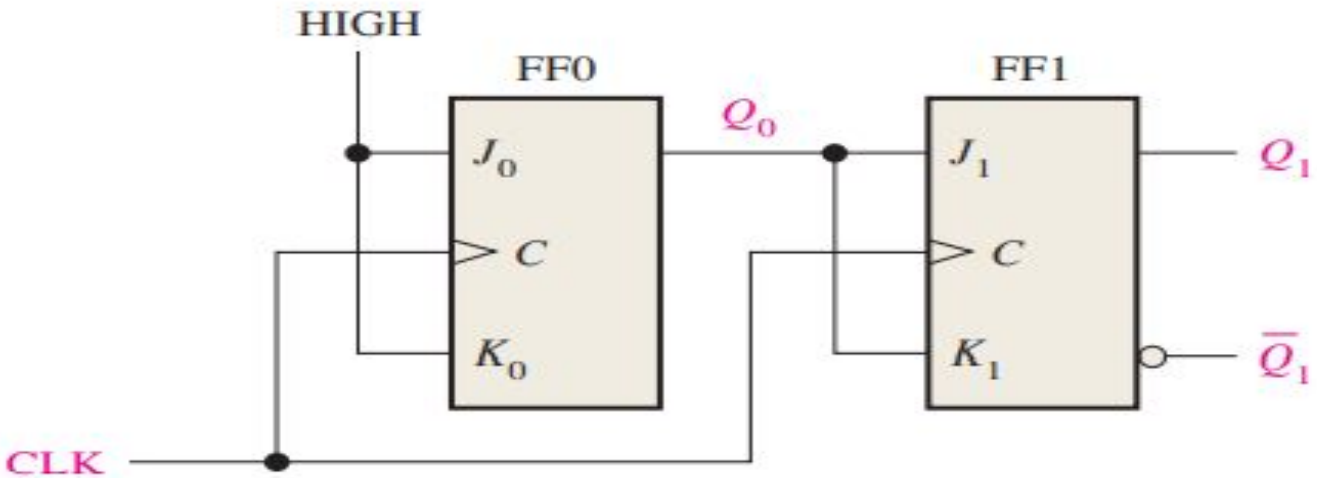
Synchronous Counters

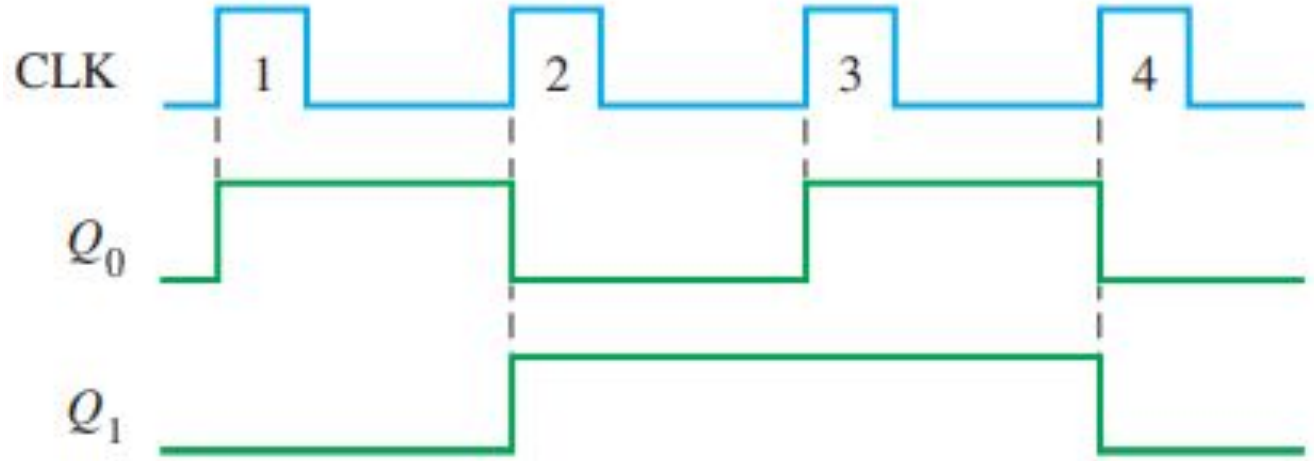
A synchronous counter is one in which all the flip-flops in the counter are clocked at the same time by a common clock pulse.

J-K flip-flops are used to illustrate most synchronous counters.

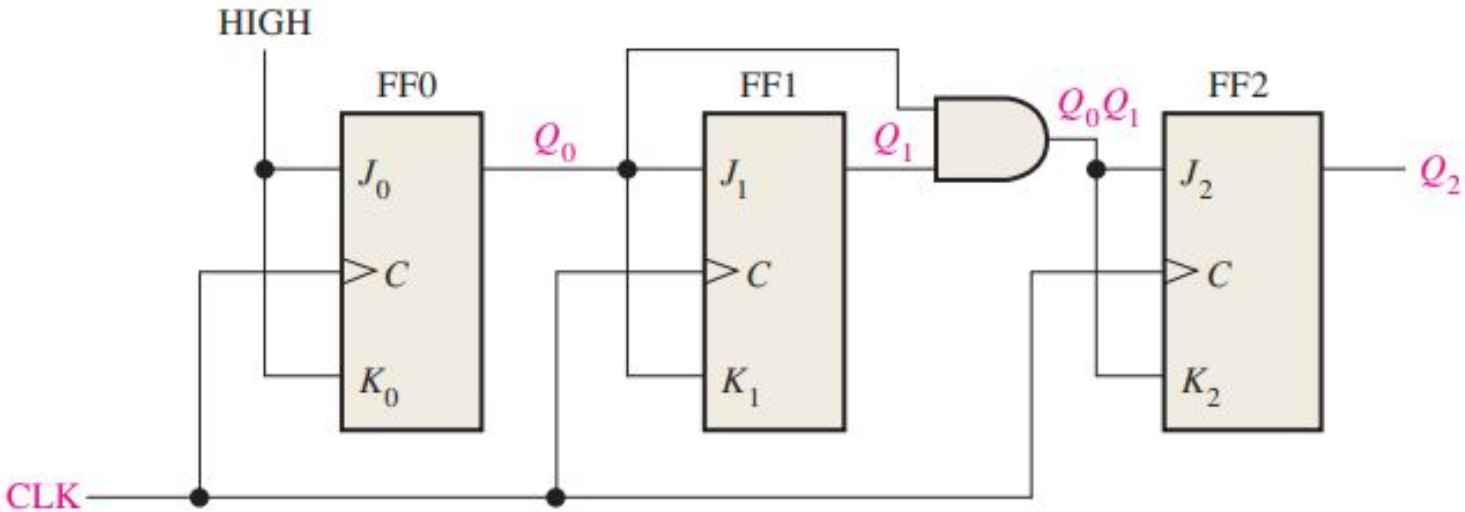
D flip-flops can also be used but generally require more logic because of having no direct toggle or no-change states

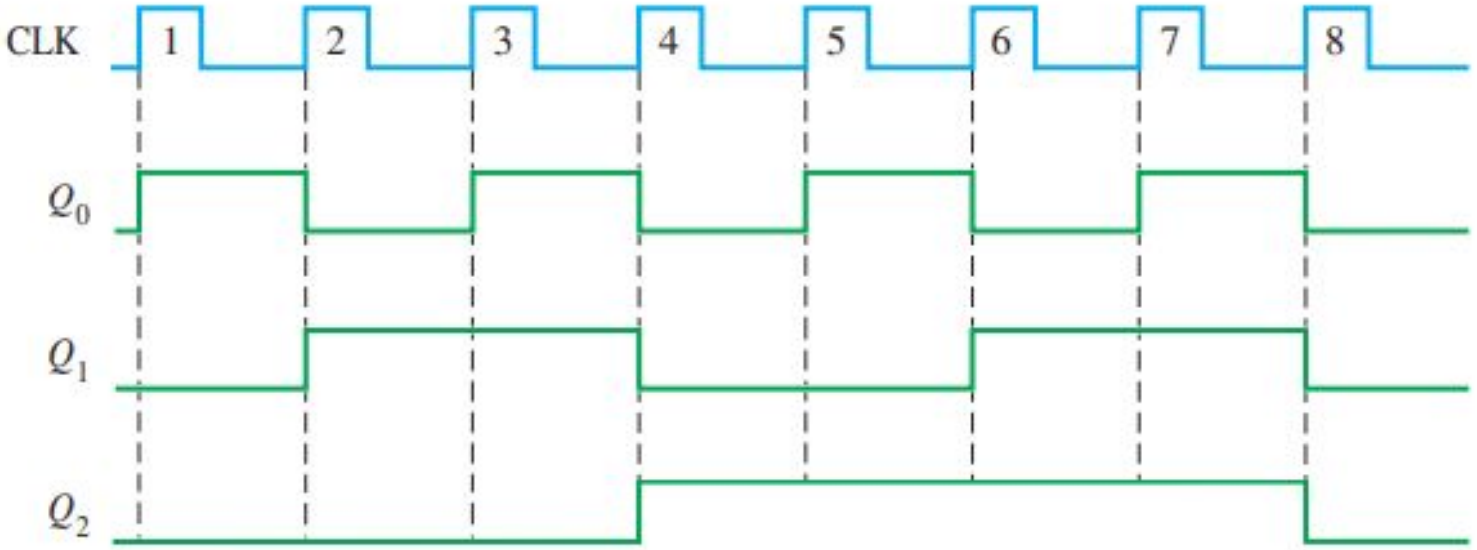
A 2-Bit Synchronous Binary Counter (RESET INITIALLY)





A 3-Bit Synchronous Binary Counter

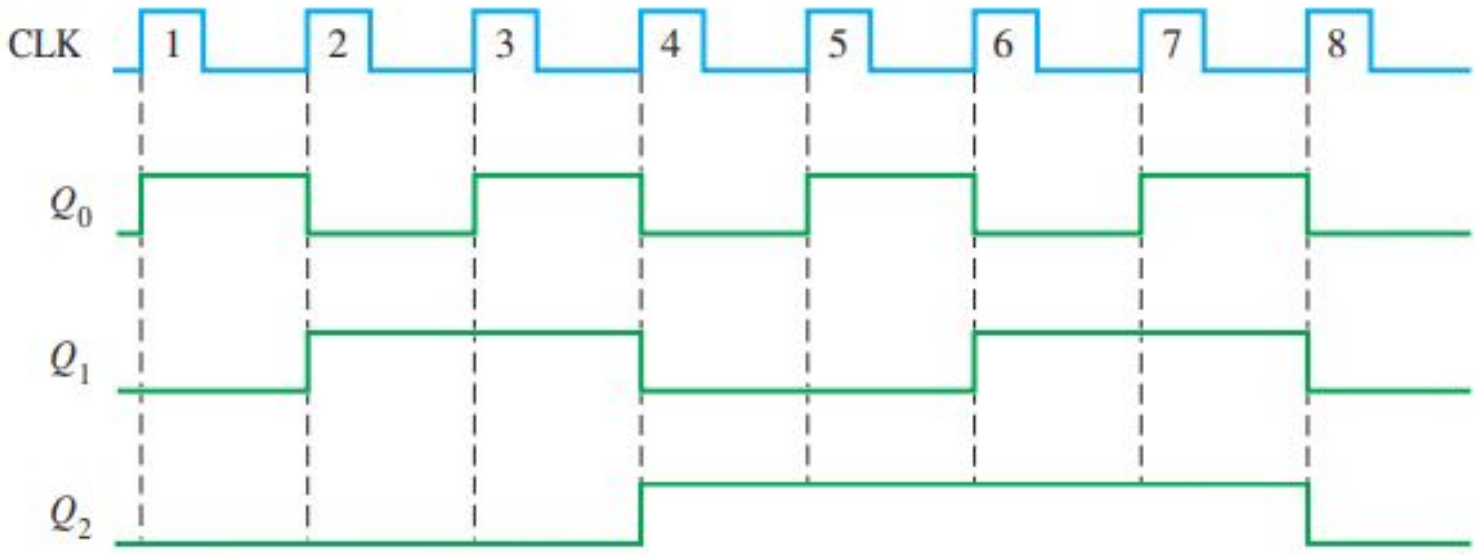




A 3-Bit Synchronous Binary Counter

Clock Pulse	Outputs			*J-K* Inputs						At the Next Clock Pulse		
	Q_2	Q_1	Q_0	J_2	K_2	J_1	K_1	J_0	K_0	FF2	FF1	FF0
Initially	0	0	0	0	0	0	0	1	1	NC*	NC	Toggle
1	0	0	1	0	0	1	1	1	1	NC	Toggle	Toggle
2	0	1	0	0	0	0	0	1	1	NC	NC	Toggle
3	0	1	1	1	1	1	1	1	1	Toggle	Toggle	Toggle
4	1	0	0	0	0	0	0	1	1	NC	NC	Toggle
5	1	0	1	0	0	1	1	1	1	NC	Toggle	Toggle
6	1	1	0	0	0	0	0	1	1	NC	NC	Toggle
7	1	1	1	1	1	1	1	1	1	Toggle	Toggle	Toggle
										Counter recycles back to 000.		

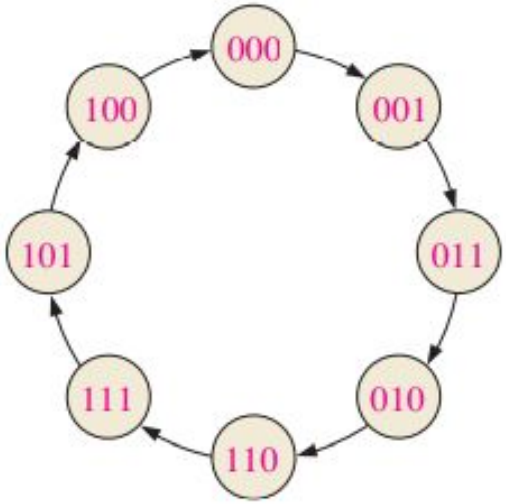
*NC indicates *No Change*.



Design of Synchronous Counters (Gray Code Example)

Step 1: State Diagram

A state diagram shows the progression of states through which the counter advances when it is clocked.



Step 2: Next-State Table

Once the sequential circuit is defined by a state diagram, the second step is to derive a next-state table, which lists each state of the counter (present state) along with the corresponding next state.

The next state is the state that the counter goes to from its present state upon application of a clock pulse.

Present State			Next State		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0
0	0	0	0	0	1
0	0	1	0	1	1
0	1	1	0	1	0
0	1	0	1	1	0
1	1	0	1	1	1
1	1	1	1	0	1
1	0	1	1	0	0
1	0	0	0	0	0

Step 3: Flip-Flop Transition Table

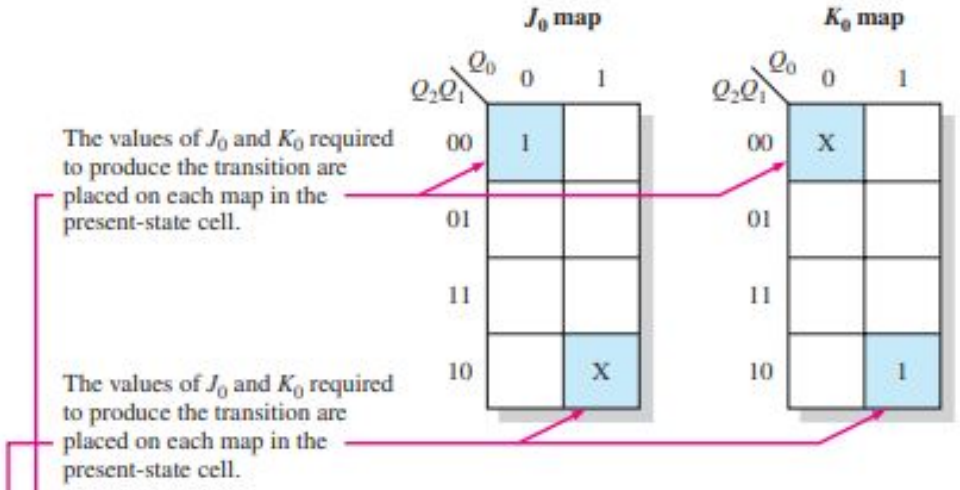
Output Transitions			Flip-Flop Inputs	
Q_N		Q_{N+1}	J	K
0	⟶	0	0	X
0	⟶	1	1	X
1	⟶	0	X	1
1	⟶	1	X	0

Q_N: present state
Q_{N+1}: next state
X: "don't care"

All possible output transitions are listed by showing the Q output of the flip-flop going from present states to next states. QN is the present state of the flip-flop (before a clock pulse) and QN + 1 is the next state (after a clock pulse). For each output transition, the J and K inputs that will cause the transition to occur are listed. An X indicates a "don't care" (the input can be either a 1 or a 0).

Step 4: Karnaugh Maps

In this design procedure, each cell in a Karnaugh map represents one of the present states in the counter sequence



Output Transitions		Flip-Flop Inputs	
Q_N	Q_{N+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

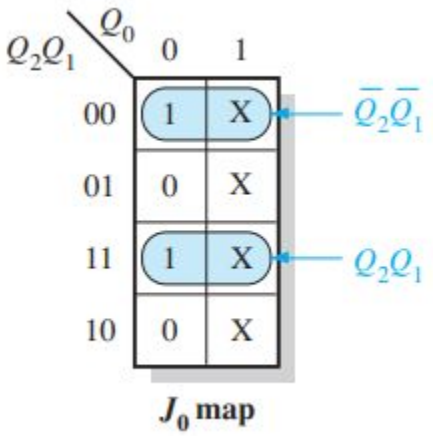
Flip-flop transition table

Present State			Next State		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0
0	0	0	0	0	1
0	0	1	0	1	1
0	1	1	0	1	0
0	1	0	1	1	0
1	1	0	1	1	1
1	1	1	1	0	1
1	0	1	1	0	0
1	0	0	0	0	0

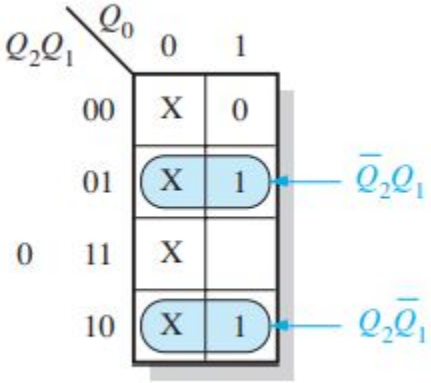
Next-state table

For the present state 000, Q_0 makes a transition from 0 to 1 to the next state.

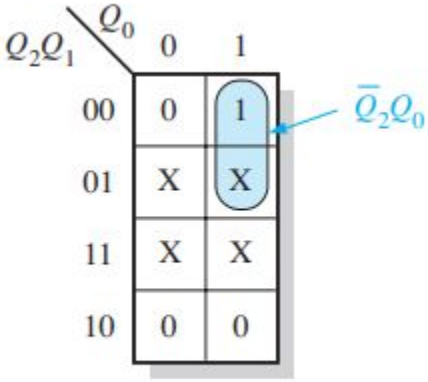
For the present state 101, Q_0 makes a transition from 1 to 0 to the next state.



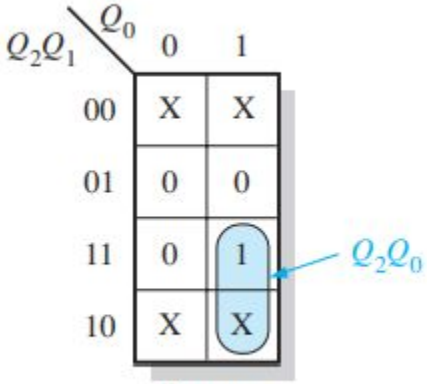
J_0 map



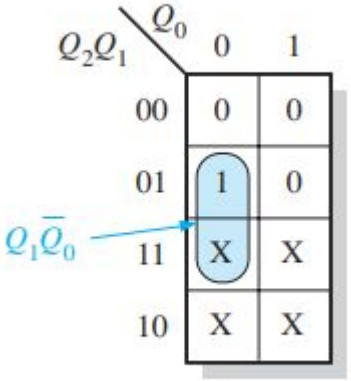
K_0 map



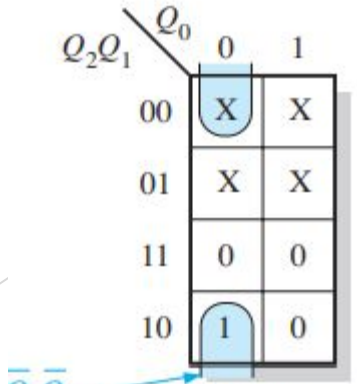
J_1 map



K_1 map



J_2 map



K_2 map

Step 5: Logic Expressions for Flip-Flop Inputs

$$J_0 = Q_2Q_1 + \overline{Q}_2\overline{Q}_1 = \overline{Q_2 \oplus Q_1}$$
$$K_0 = Q_2\overline{Q}_1 + \overline{Q}_2Q_1 = Q_2 \oplus Q_1$$
$$J_1 = \overline{Q}_2Q_0$$
$$K_1 = Q_2Q_0$$
$$J_2 = Q_1\overline{Q}_0$$
$$K_2 = \overline{Q}_1\overline{Q}_0$$

Step 6: Counter Implementation