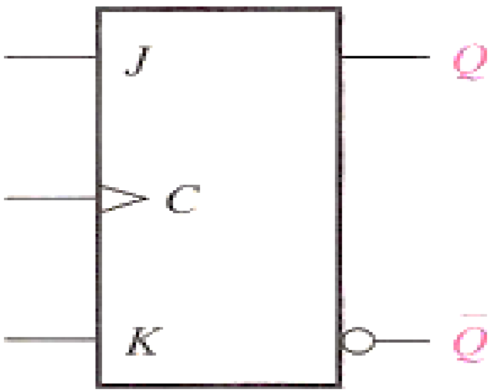
EEE 3101: Digital Logic and Circuits

Counters

Course Teacher: Nafiz Ahmed Chisty

Associate Professor, Department of EEE & CoE
Head (UG), Department of EEE
Faculty of Engineering
Room# DNG03, Ground Floor, D Building
Email: chisty@aiub.edu
Website: http://engg.aiub.edu/
Website: www.nachisty.com

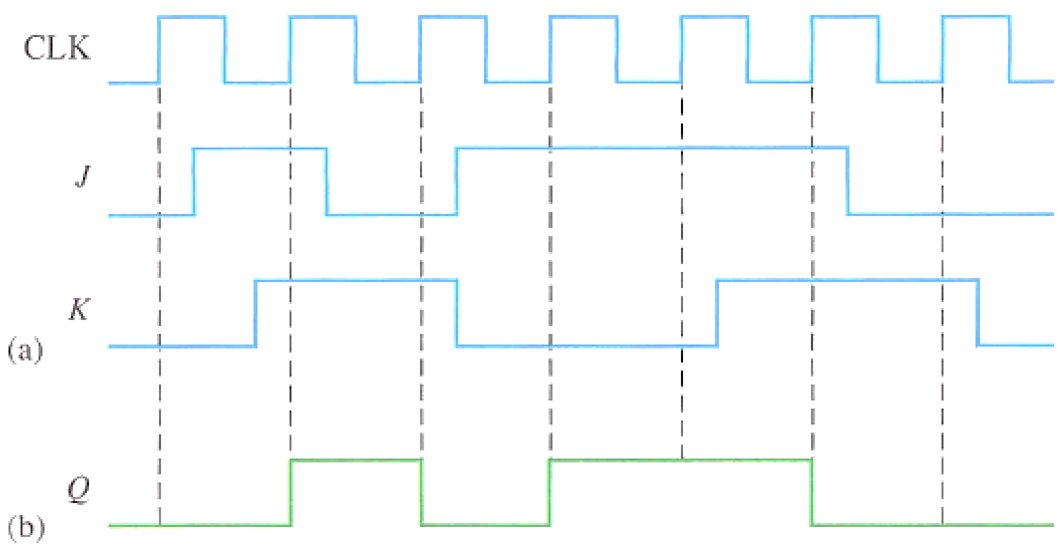
Review



INPUTS			OUTPUTS		COMMENTS
J	K	CLK	Q	$\overline{Q}$	
0	0	↑	Q_0	$\overline{Q}_0$	No change
0	1	↑	0	1	RESET
1	0	↑	1	0	SET
1	1	↑	$\overline{Q}_0$	Q_0	Toggle

↑ = clock transition LOW to HIGH

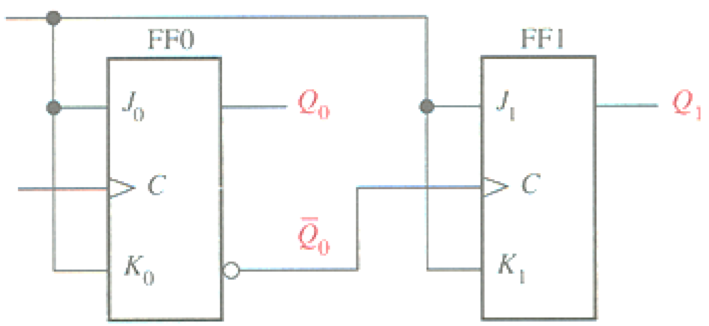
Q_0 = output level prior to clock transition



Counters

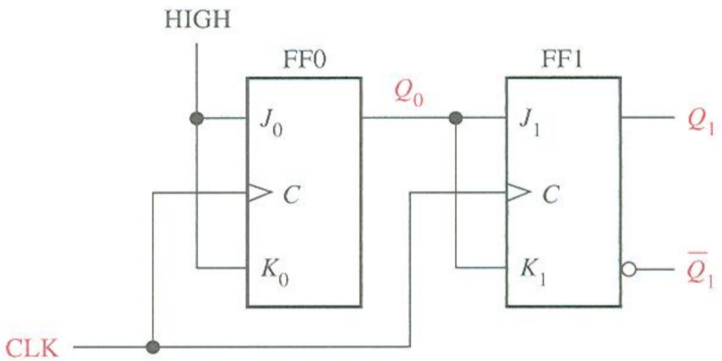
Asynchronous Counter

Flip-flops(FF) **do not** have a **common clock pulse**



Synchronous Counter

Flip-flops (FF) have a **common clock pulse**



Nafiz Ahmed Chisty| Head (UG) & Associate Professor, Dept. of EEE, FE, AIUB|

Asynchronous Counter

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**

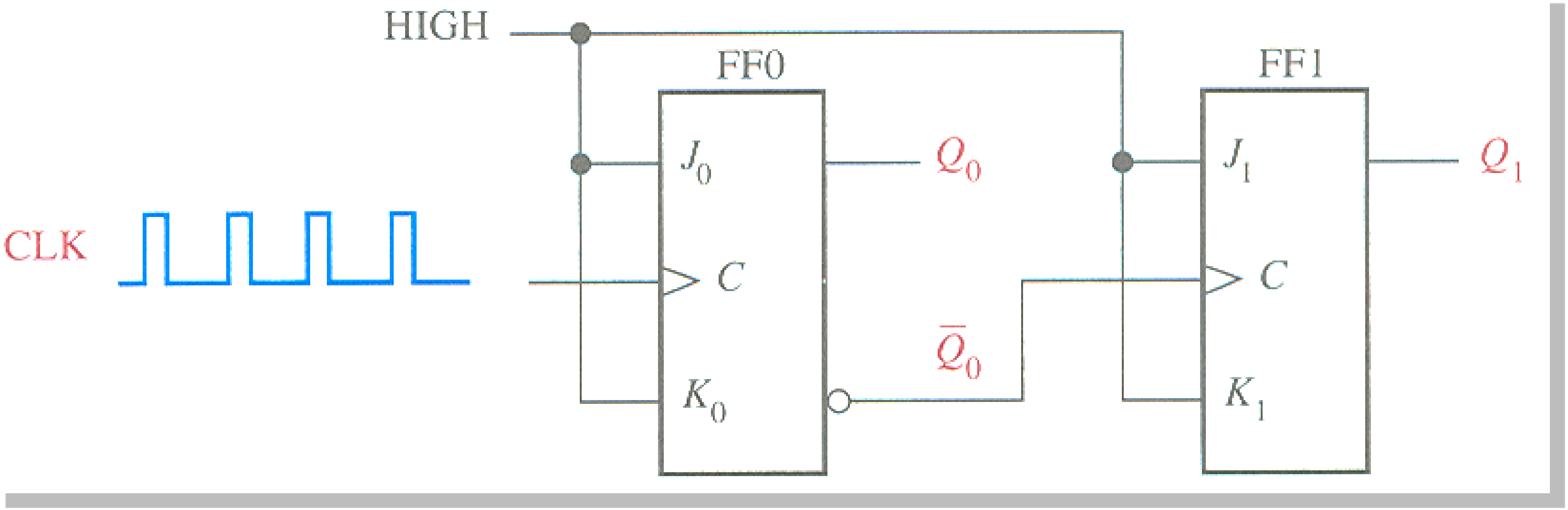
Asynchronous Counter Operation

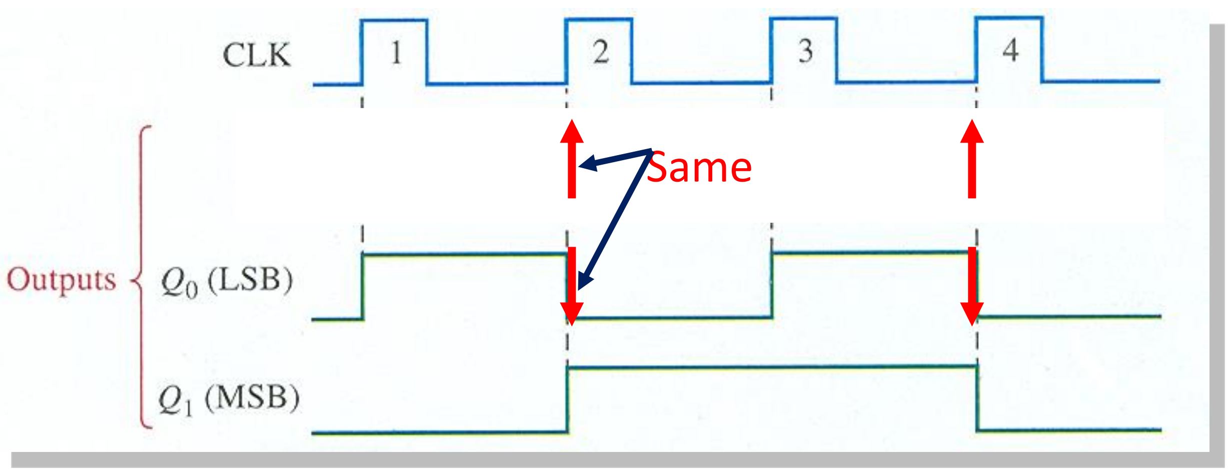
- Asynchronous binary counter
  - 2-bit asynchronous binary counter
  - 3-bit asynchronous binary counter
  - 4-bit asynchronous binary counter

- Asynchronous decade counter
- Asynchronous Modulus Twelve counter

Nafiz Ahmed Chisty| Head (UG) & Associate Professor, Dept. of EEE, FE, AIUB|

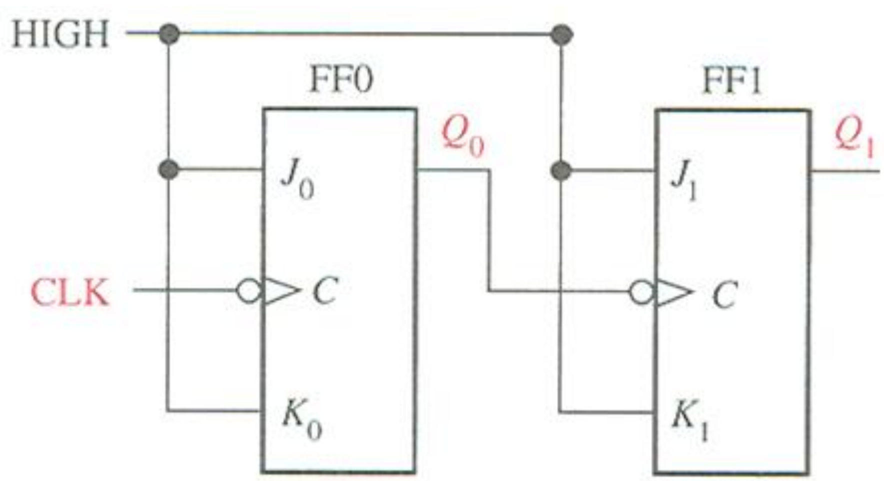
2-bit asynchronous binary counter

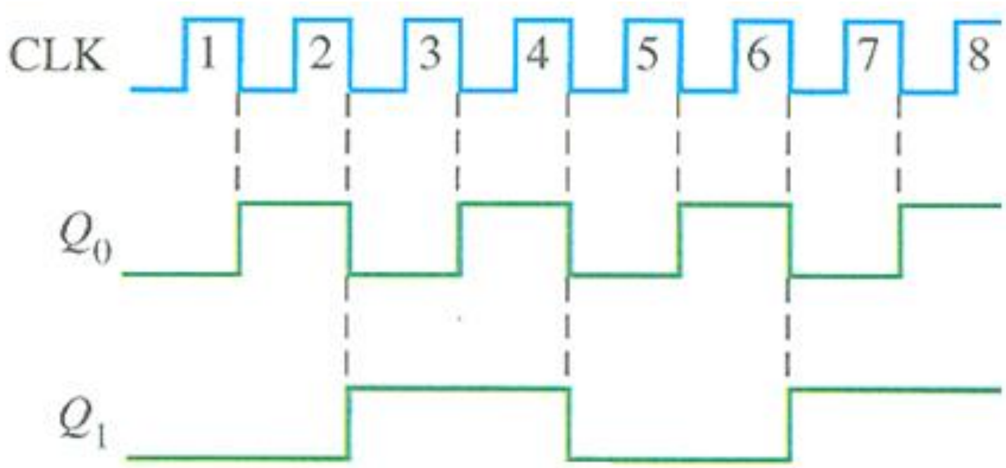




CLOCK PULSE	Q_1	Q_0
Initially	0	0
1	0	1
2	1	0
3	1	1
4 (recycles)	0	0

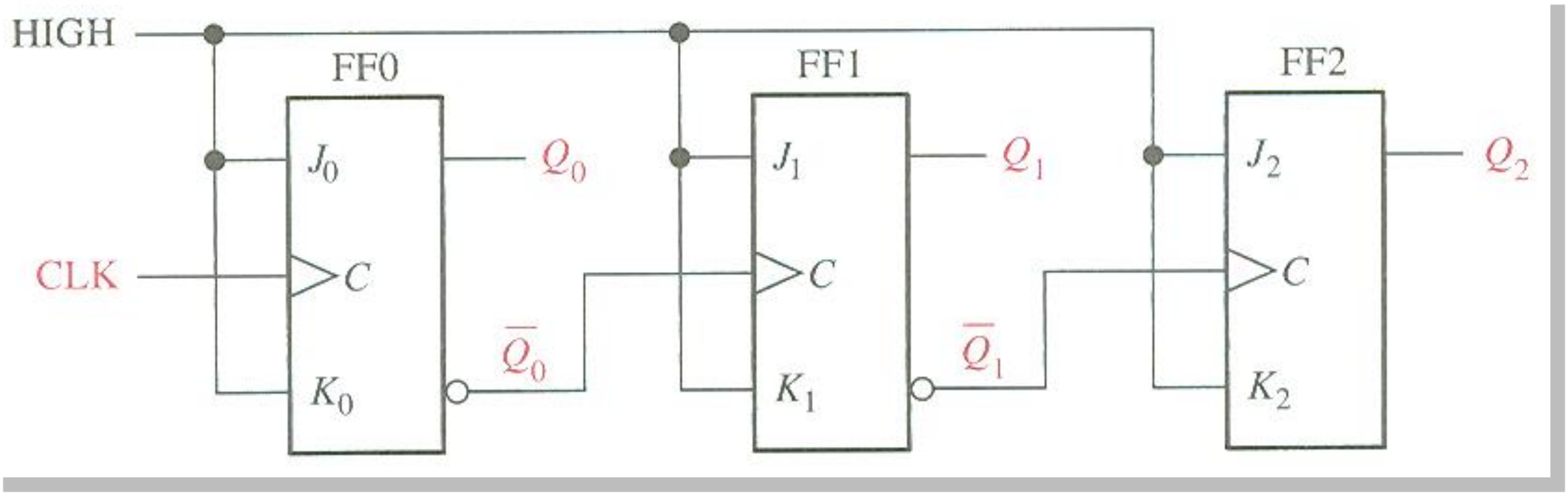
2-bit asynchronous binary counter



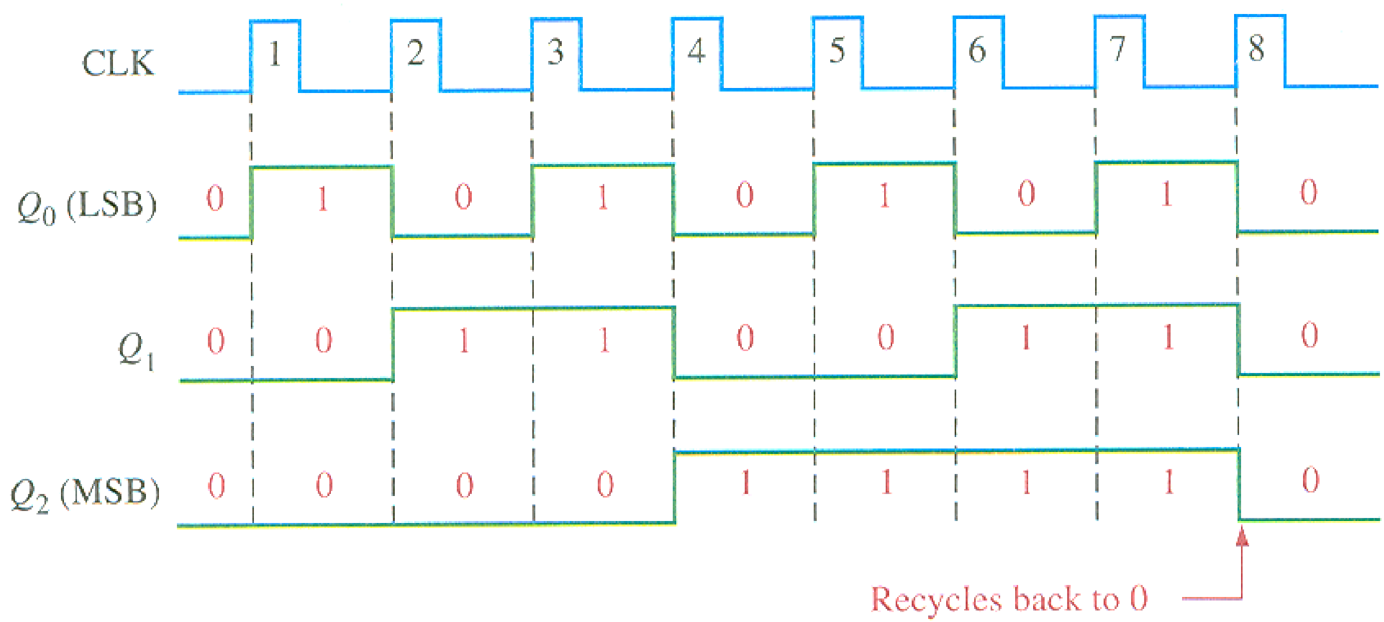


CLOCK PULSE	Q_1	Q_0
Initially	0	0
1	0	1
2	1	0
3	1	1
4 (recycles)	0	0

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



Modulus of a Counter

The **modulus** of a counter is the number of unique states through which the counter will sequence. The maximum possible number of states (maximum modulus) of a counter is 2^n, where n is the number of flip-flops in the counter. Counters can be designed to have a number of states in their sequence that is less than the maximum of 2^n. This type of sequence is called a *truncated sequence*.

Asynchronous decade (MOD-10) counter

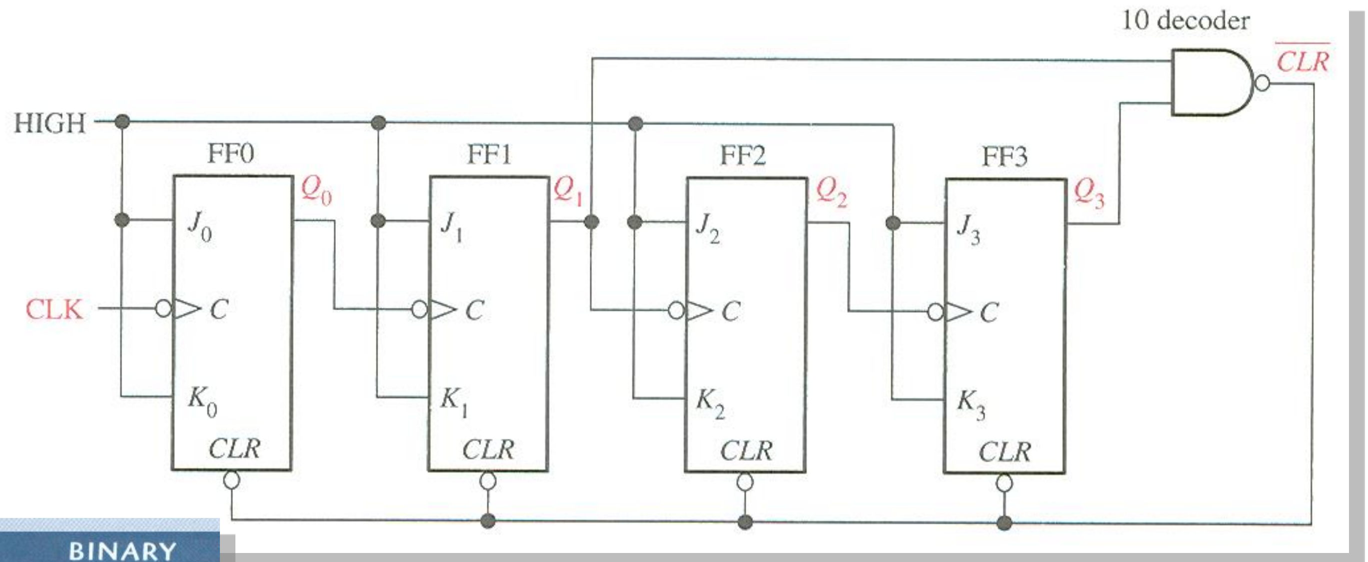
One common modulus for counters with truncated sequences is ten (called MOD10). Counters with ten states in their sequence are called **decade** counters. A decade counter with a count sequence of zero (0000) through nine (1001) is a BCD decade counter because its ten-state sequence produces the BCD code. This type of counter is useful in display applications in which BCD is required for conversion to a decimal readout.

DECIMAL	BINARY
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

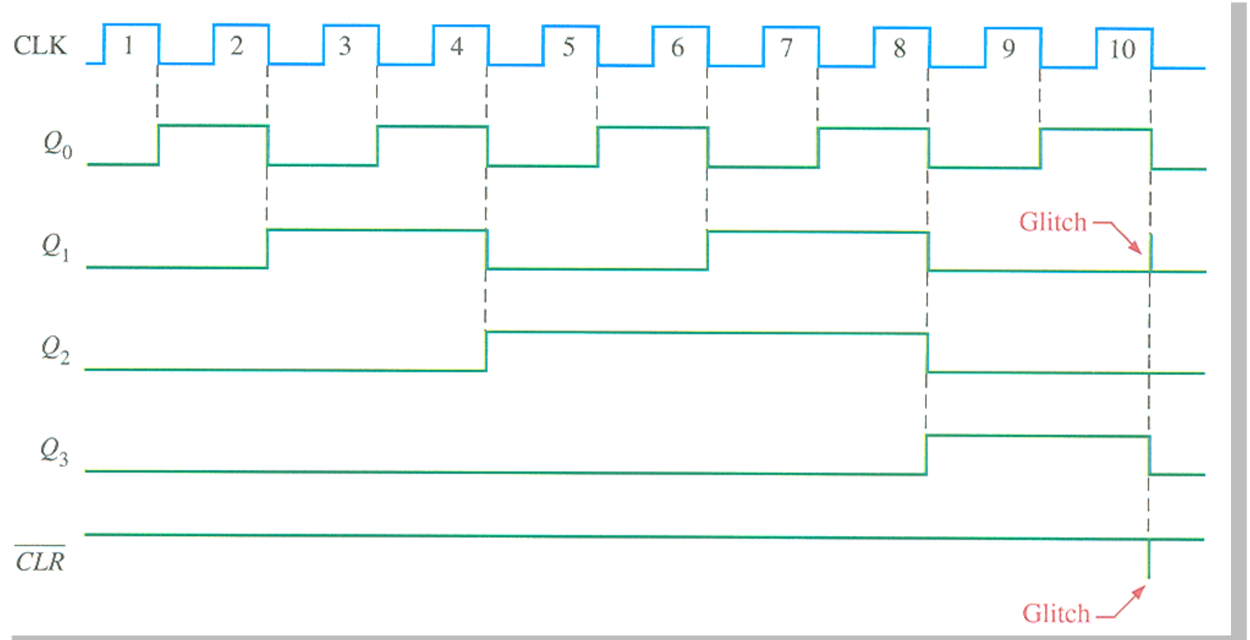
Not wanted. We want 0000.
So Decode this state.

To obtain a truncated sequence, it is necessary to force the counter to recycle before going through all of its possible states. For example, the BCD decade counter must recycle back to the 0000 state after the 1001 state. A decade counter requires four flip-flops (three flip-flops are insufficient because $2^3 = 8$).

Asynchronous decade counter



DECIMAL	BINARY
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111



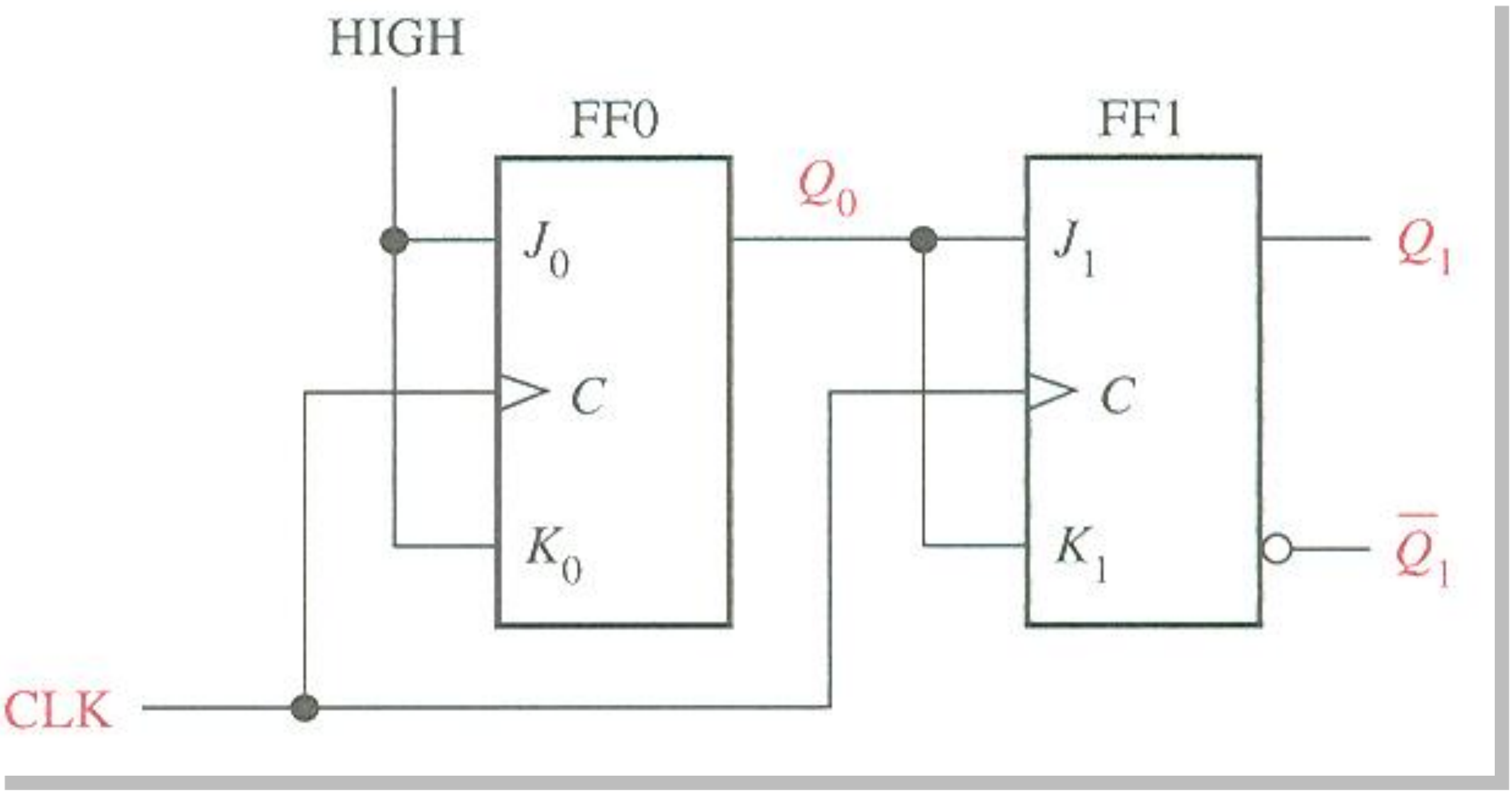
Synchronous Counter

A Synchronous counter is one in which the flip-flops(FF) within the counter are clocked at the same time by a common clock pulse.

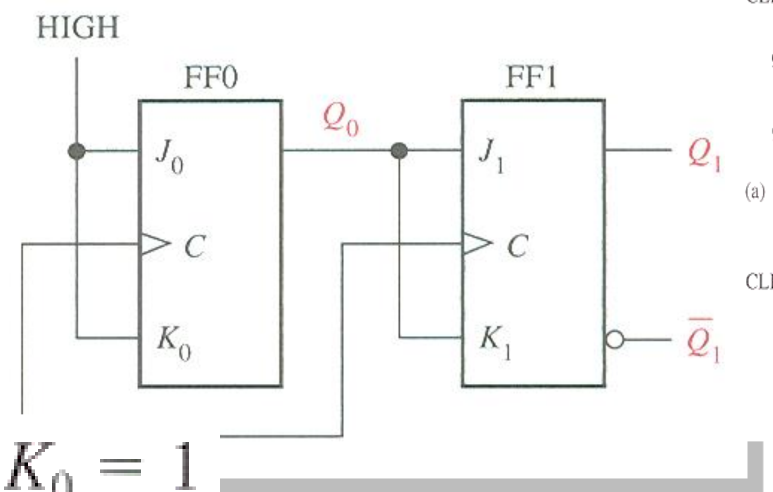
Synchronous Counter Operation

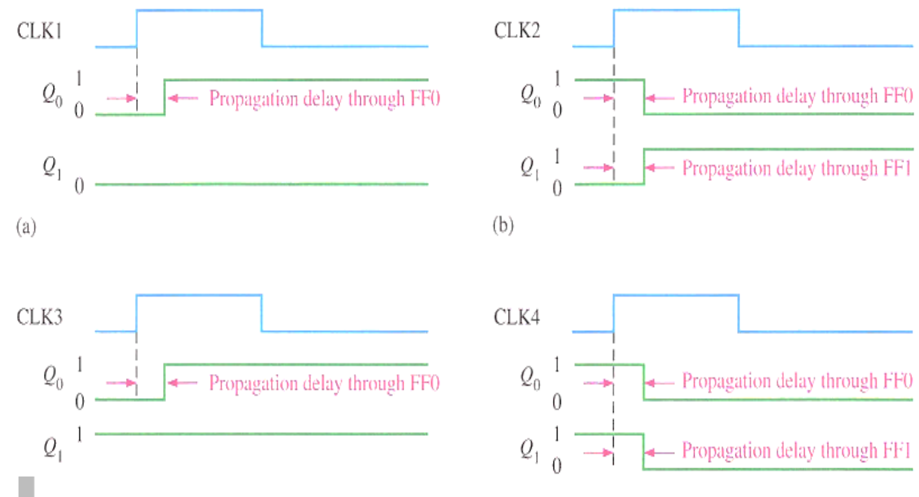
- Synchronous binary counters
  - 2-bit counter
  - 3-bit counter
  - 4-bit counter
- Synchronous BCD Decade counter
- Irregular sequence counter

2-bit synchronous binary counter



2-bit synchronous binary counter

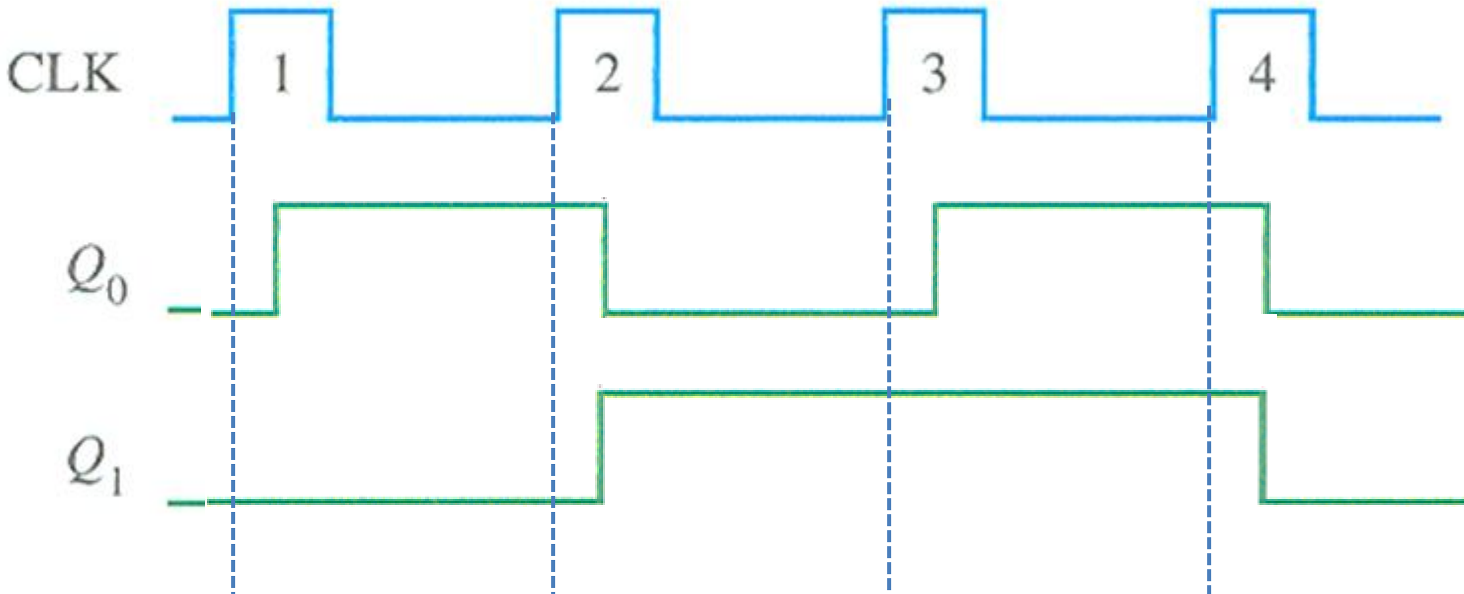




$$J_0 = K_0 = 1$$

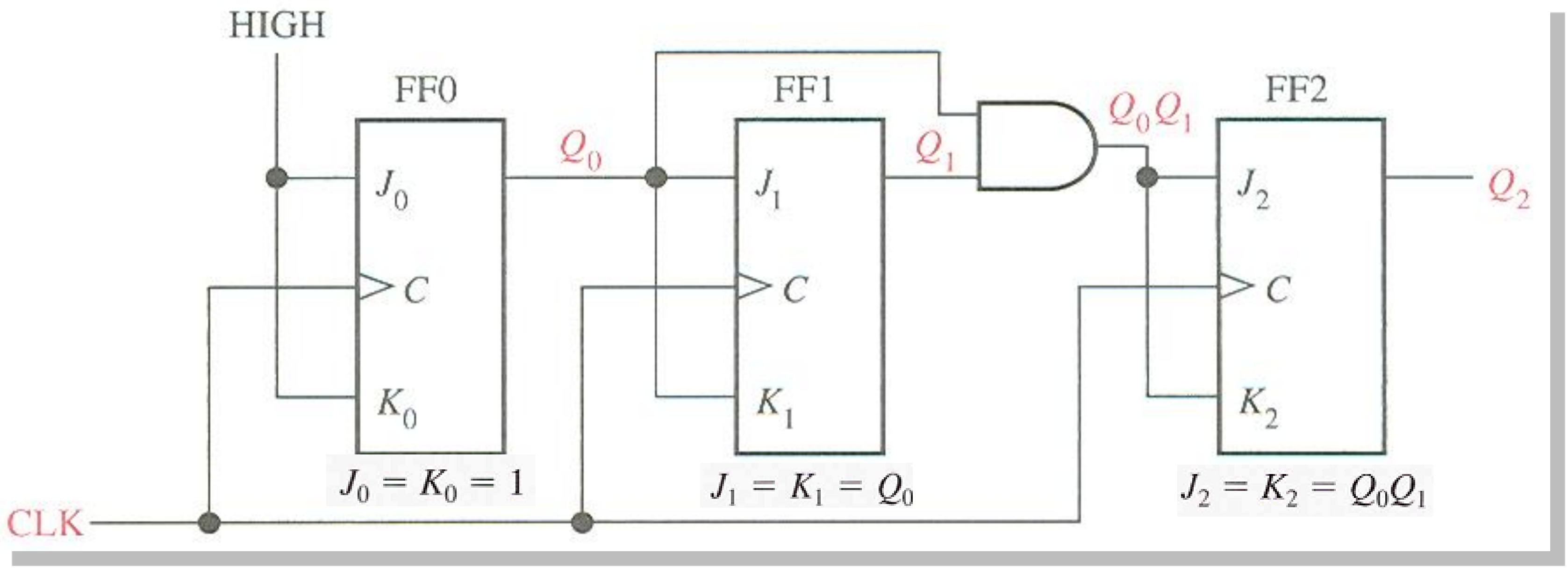
$$J_1 = K_1 = Q_0$$

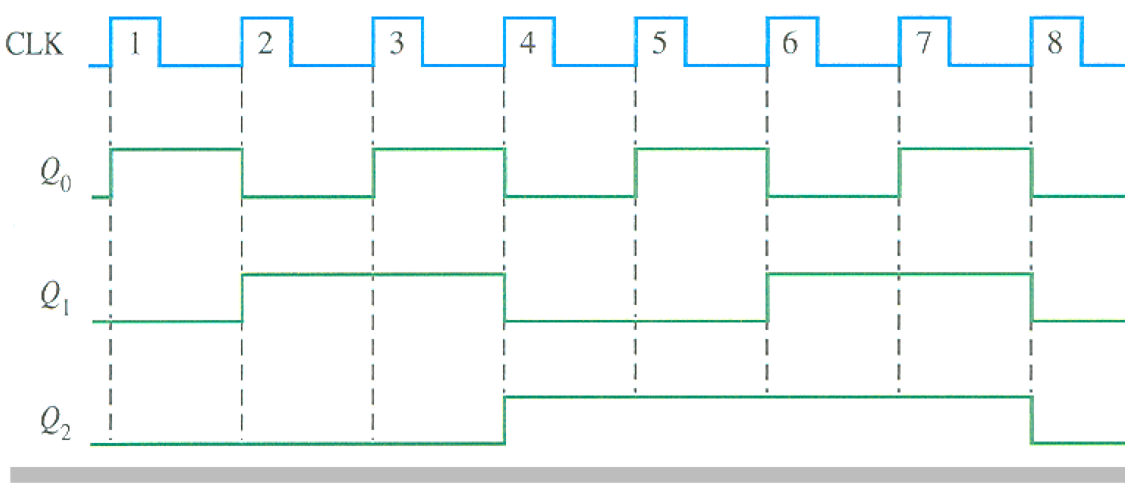
Q_1	Q_0
0	0
0	1
1	0
1	1
0	0



Nafiz Ahmed Chisty| Head (UG) & Associate Professor, Dept. of EEE, FE, AIUB|

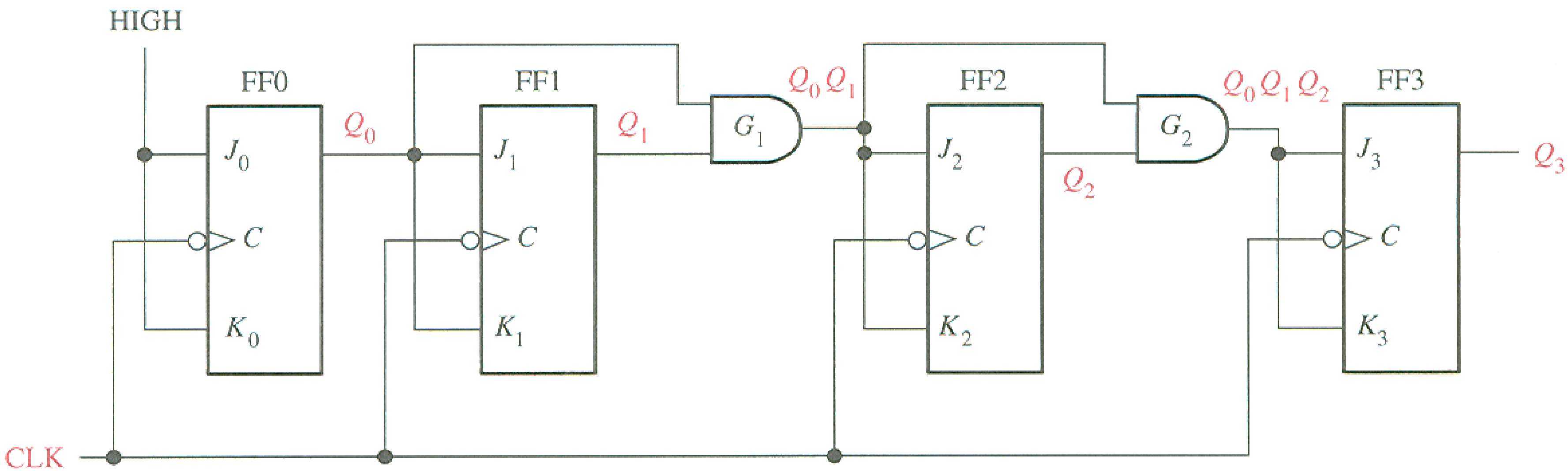
3-bit synchronous binary counter





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

4-bit synchronous binary counter



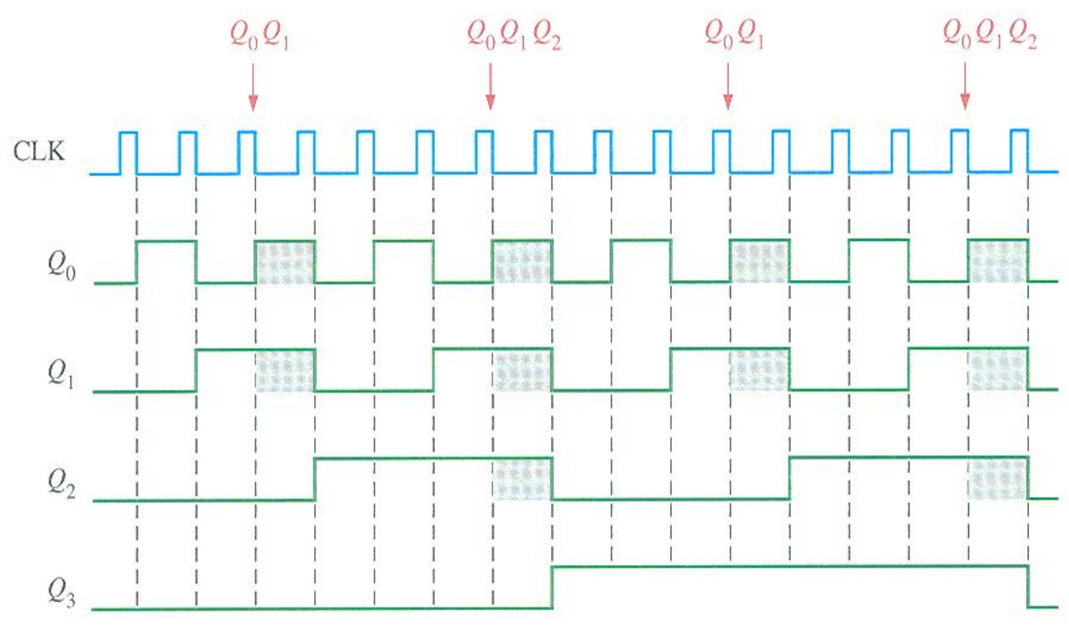
DECIMAL	BINARY
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111
0	0000

$$J_0 = K_0 = 1$$

$$J_1 = K_1 = Q_0$$

$$J_2 = K_2 = Q_0Q_1$$

$$J_3 = K_3 = Q_0Q_1Q_2$$



Synchronous Decade counter (MOD 10)

4-bit synchronous binary counter

Q_0: $J_0 = K_0 = 1$

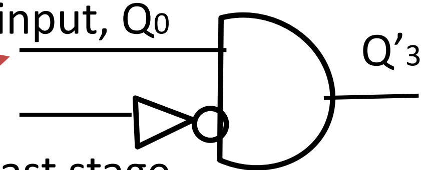
Q_1: $J_1 = K_1 = Q_0$

Q_2: $J_2 = K_2 = Q_0Q_1$

Q_3: $J_3 = K_3 = Q_0Q_1Q_2$

Forcefully Stable

Normal input, Q_0

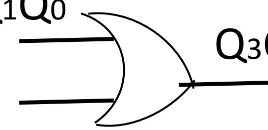


Q'_3Q_0

Decode last stage before recycle, Q_3Q_0

Forcefully Toggle

Normal input, $Q_2Q_1Q_0$



$Q_3Q_0+Q_2Q_1Q_0$

Decode last stage before recycle, Q_3Q_0

Naturally Zero (unchanged)

		$Q_3Q_2Q_1Q_0$
9	Last Valid	1001
10	Invalid	1010
0	Recycle	0000

Forcefully Toggle

Forcefully Stable

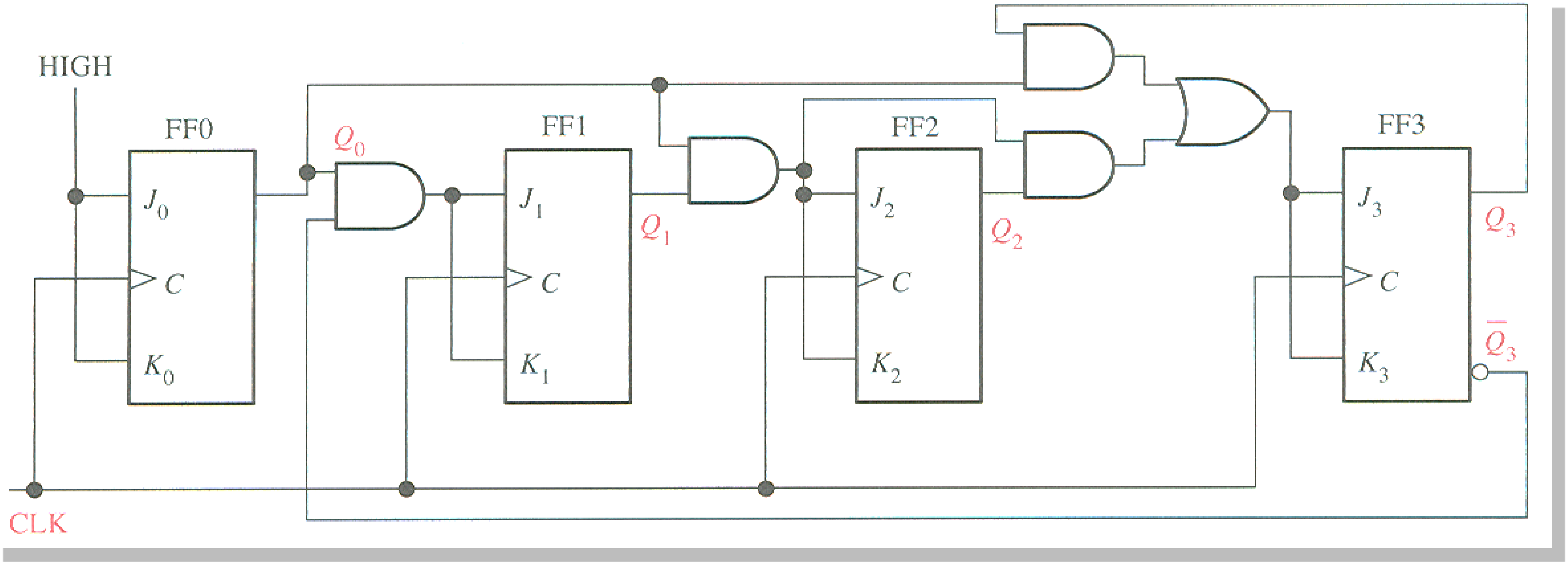
$J_0 = K_0 = 1$

$J_1 = K_1 = Q_0\overline{Q_3}$

$J_2 = K_2 = Q_0Q_1$

$J_3 = K_3 = Q_0Q_1Q_2 + Q_0Q_3$

Synchronous BCD decade counter

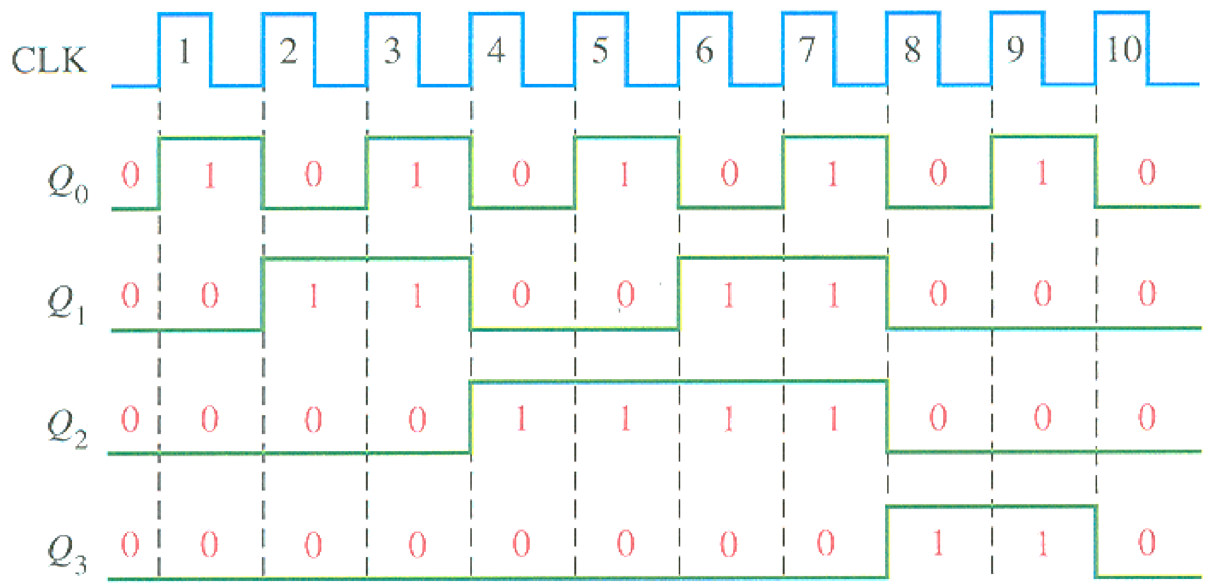


$$J_0 = K_0 = 1$$

$$J_1 = K_1 = Q_0\overline{Q}_3$$

$$J_2 = K_2 = Q_0Q_1$$

$$J_3 = K_3 = Q_0Q_1Q_2 + Q_0Q_3$$

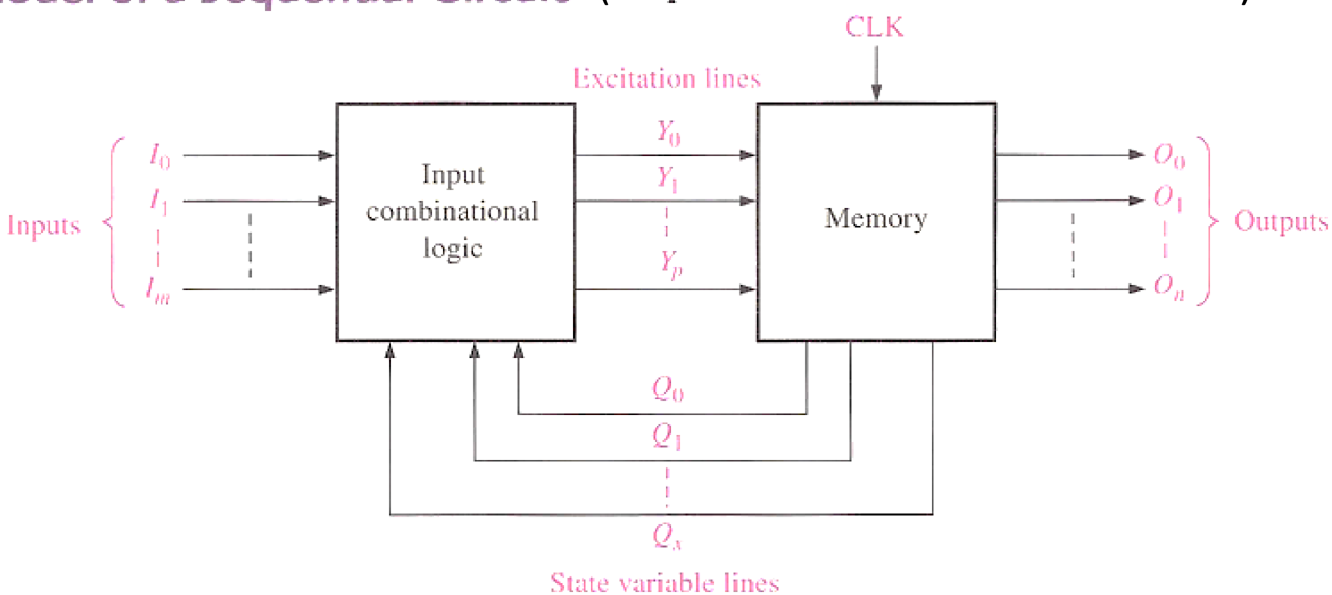


Nafiz Ahmed Chisty| Head (UG) & Associate Professor, Dept. of EEE, FE, AIUB|

DESIGN OF SYNCHRONOUS COUNTERS

In this section, you will see how sequential circuit design techniques can be applied specifically to counter design. In general, sequential circuits can be classified into two types: (1) those in which the output or outputs depend only on the present internal state (called *Moore circuits*) and (2) those in which the output or outputs depend on both the present state and the input or inputs (called *Mealy circuits*). This section is

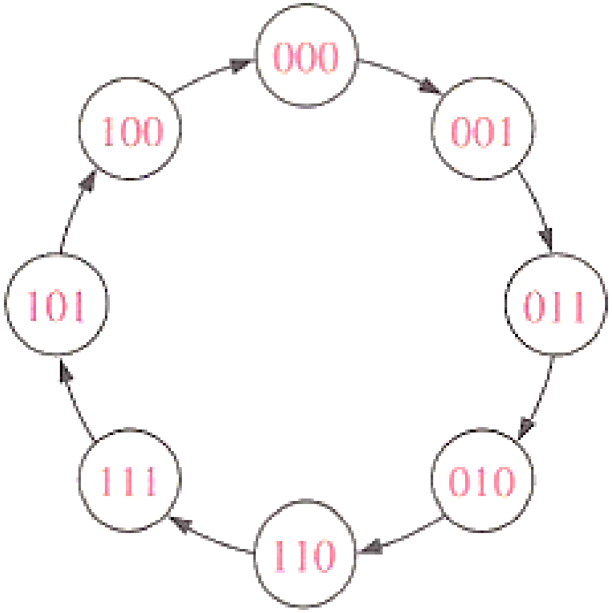
General Model of a Sequential Circuit (sequential circuit or state machine)



Not all sequential circuits have input and output variables as in the general model just discussed. However, all have excitation variables and state variables. Counters are a special case of clocked sequential circuits. In this section, a general design procedure for sequential circuits is applied to synchronous counters in a series of steps.

Step 1: State Diagram

The first step in the design of a counter is to create a state diagram. A state diagram shows the progression of states through which the counter advances when it is clocked. As an example, Figure below shows a state diagram for a basic 3-bit Gray code counter.



Step 2: Next-State Table

Once the sequential circuit is defined by a state diagram, the second step is to derive a nextstate 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. The next-state table is derived from the state diagram and is shown in Table 8-7 for the 3-bit Gray code counter. Q, is the least significant bit.

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

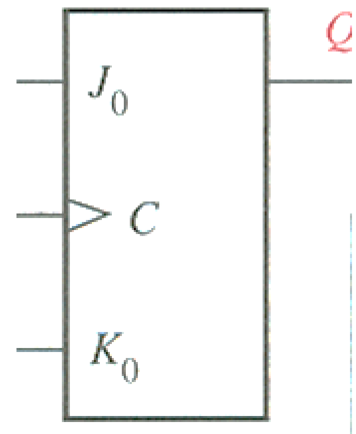
Nafiz Ahmed Chisty| Head (UG) & Associate Professor, Dept. of EEE, FE, AIUB|

Step 3: Flip-Flop Transition Table

Table 8–8 is a transition table for the J-K flip-flop. All possible output transitions are listed by showing the output of the flip-flop going from present states to next states. Q_N is the present state of the flip-flop (before a clock pulse) and Q_{N+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).

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"



To design the counter, the transition table is applied to each of the flip-flops in the counter, based on the next-state table (Table 8–7). For example, for the present state 000, Qo goes from a present state of 0 to a next state of 1. To make this happen, J, must be a 1 and you don't care what K_0 is (J_0 = 1, K_0 = X), as you can see in the transition table (Table 8-8). Next, Q_1 is 0 in the present state and remains a 0 in the next state. For this transition, J_1 = 0 and K_1 = X. Finally, Q_2 is 0 in the present state and remains a 0 in the next state. Therefore, J_2 = 0 and K_2 = X. This analysis is repeated for each present state in Table 8–7.

Step 4: Karnaugh Maps

Karnaugh maps can be used to determine the logic required for the *J* and *K* inputs of each flip-flop in the counter. There is a Karnaugh map for the *J* input and a Karnaugh map for the *K* input of each flip-flop. In this design procedure, each cell in a Karnaugh map represents one of the present states in the counter sequence listed in Table 8–7.

From the *J* and *K* states in the transition table (Table 8–8) a 1, 0, or X is entered into each present state cell on the maps depending on the transition of the *Q* output for a particular flip-flop. To illustrate this procedure, two sample entries are shown for the J_0 and the K_0 inputs to the least significant flip-flop (Q_0) in Figure 8–29.

J_0 map

Q_2Q_1 \ Q_0	0	1
00		
01		
11		
10		X

K_0 map

Q_2Q_1 \ Q_0	0	1
00		
01		
11		
10		1

Q_2Q_1 \ Q_0	0	1	
00	1	X	Q_2Q_1
01	0	X	
11	1	X	Q_2Q_1
10	0	X	

J_0 map

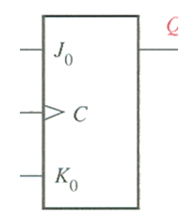
Q_2Q_1 \ Q_0	0	1	
00	X	0	
01	X	1	$\overline{Q}_2Q_1$
11	X	0	
10	X	1	$Q_2\overline{Q}_1$

K_0 map

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

Present State Q_2	Q_1	Q_0	Next State 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

For Q_0:



Nafiz Ahmed Chisty| Head (UG) & Associate Professor, Dept. of EEE, FE, AIUB|

The completed Karnaugh maps for all three flip-flops in the counter are shown in Figure 8–30. The cells are grouped as indicated and the corresponding Boolean expressions for each group are derived.

For Q_2:

Q_2Q_1 \ Q_0	0	1
00	0	0
01	1	0
11	X	X
10	X	X

$Q_1\bar{Q}_0$

J_2 map

Q_2Q_1 \ Q_0	0	1
00	X	X
01	X	X
11	0	0
10	1	0

$\bar{Q}_1\bar{Q}_0$

K_2 map

For Q_1:

Q_2Q_1 \ Q_0	0	1
00	0	1
01	X	X
11	X	X
10	0	0

$\bar{Q}_2Q_0$

J_1 map

Q_2Q_1 \ Q_0	0	1
00	X	X
01	0	0
11	0	1
10	X	X

Q_2Q_0

K_1 map

For Q_0:

Q_2Q_1 \ Q_0	0	1
00	1	X
01	0	X
11	1	X
10	0	X

$\bar{Q}_2\bar{Q}_1$

Q_2Q_1

J_0 map

Q_2Q_1 \ Q_0	0	1
00	X	0
01	X	1
11	X	0
10	X	1

$\bar{Q}_2Q_1$

$Q_2\bar{Q}_1$

K_0 map

Step 5: Logic Expressions for Flip-Flop Inputs

From the Karnaugh maps of Figure 8–30 you obtain the following expressions for the J and K inputs of each flip-flop:

$$J_0 = Q_2Q_1 + \overline{Q}_2\overline{Q}_1 = \overline{Q_2 \oplus Q_1}$$

$$K_0 = Q_2\overline{Q}_1 + \overline{Q}_2Q_1 = Q_2 \oplus Q_1$$

$$J_1 = \overline{Q}_2Q_0$$

$$K_1 = Q_2Q_0$$

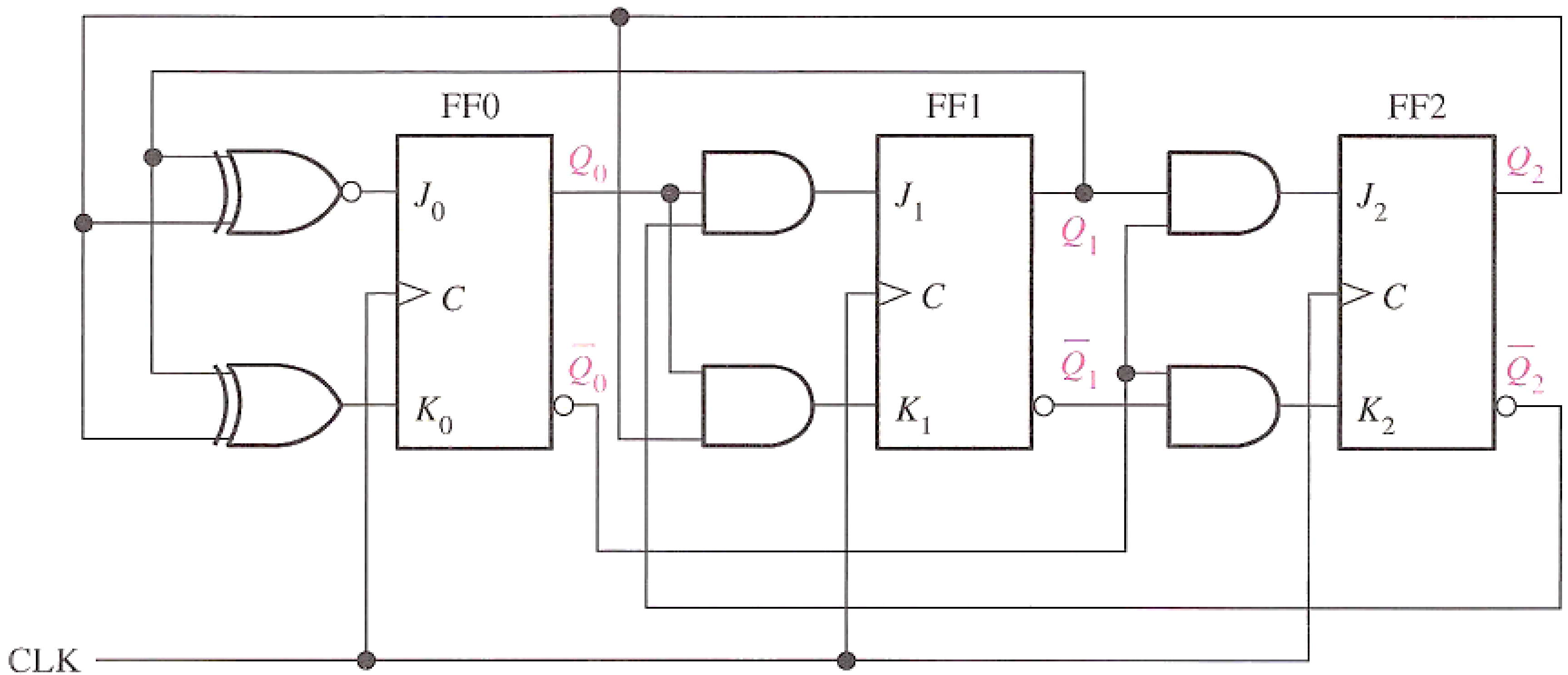
$$J_2 = Q_1\overline{Q}_0$$

$$K_2 = \overline{Q}_1\overline{Q}_0$$

Implement in front of the flip-flops

Step 6: Counter Implementation

The final step is to implement the combinational logic from the expressions for the *J* and *K* inputs and connect the flip-flops to form the complete 3-bit Gray code counter as shown in Figure 8–31.



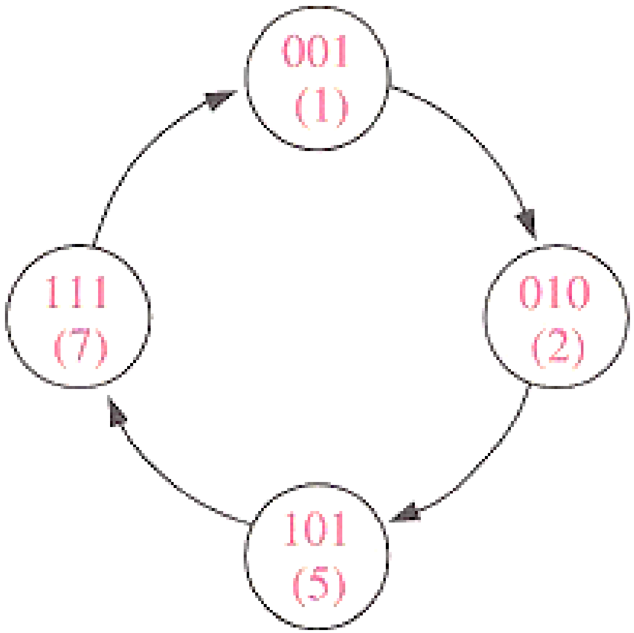
Three-bit Gray code counter.

A summary of steps used in the design of this counter follows. In general, these steps can be applied to any sequential circuit.

1. Specify the counter sequence and draw a state diagram.
2. Derive a next-state table from the state diagram.
3. Develop a transition table showing the flip-flop inputs required for each transition. The transition table is always the same for a given type of flip-flop.
4. Transfer the *J* and *K* states from the transition table to Karnaugh maps. There is a Karnaugh map for each input of each flip-flop.
5. Group the Karnaugh map cells to generate and derive the logic expression for each flip-flop input.
6. Implement the expressions with combinational logic, and combine with the flip-flops to create the counter.

EXAMPLE

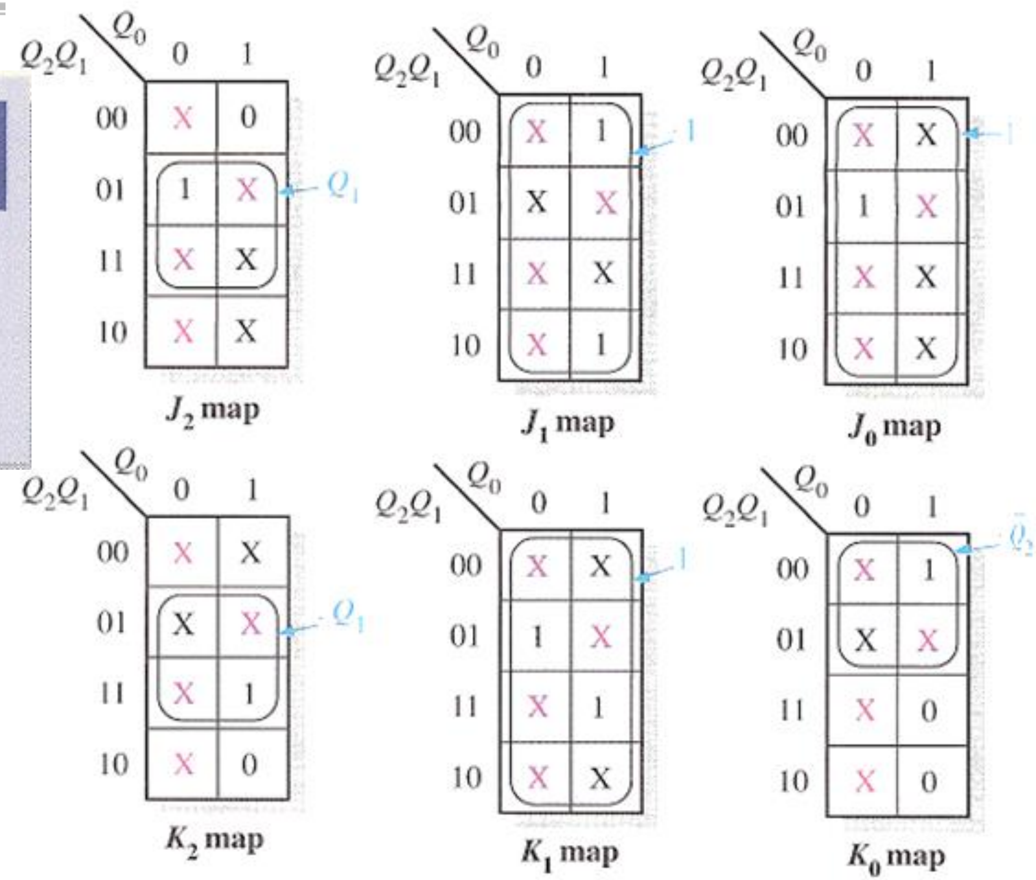
Design a counter with the irregular binary count sequence shown in the state diagram of Figure 8–32. Use J-K flip-flops.



Nafiz Ahmed Chisty| Head (UG) & Associate Professor, Dept. of EEE, FE, AIUB|

PRESENT STATE			NEXT STATE		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0
0	0	1	0	1	0
0	1	0	1	0	1
1	0	1	1	1	1
1	1	1	0	0	1

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

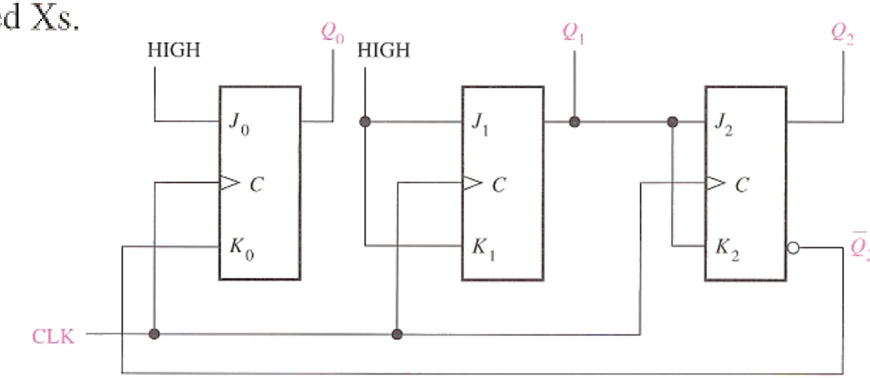


The J and K inputs are plotted on the present-state Karnaugh maps in Figure 8–33. Also “don’t cares” can be placed in the cells corresponding to the invalid states of 000, 011, 100, and 110, as indicated by the red Xs.

$$J_0 = 1, K_0 = \overline{Q_2}$$

$$J_1 = K_1 = 1$$

$$J_2 = K_2 = Q_1$$



Reference:

[1] Thomas L. Floyd, "Digital Fundamentals" 11th edition, Prentice Hall.
[2] M. Morris Mano, "Digital Logic & Computer Design" Prentice Hall.
[3] Mixed contents from Vahid And Howard.

Thanks