SYNCHRONOUS COUNTERS

Synchronous digital counters have a common clock signal that controls all flip-flop stages. Since a common clock controls all flip-flops simultaneously, there are no cumulative delays that result when a clock signal must ripple through the stages, as in the case of asynchronous counters.

Synchronous counters can count up and down, and can be designed to produce special purpose count sequences of nonconsecutive numbers.

Synchronous Counter Analysis

Synchronous counters can be analyzed by using a procedure similar to the analysis of asynchronous counters to classify fully or define the counter operation.

Analyze a synchronous counter circuit by proceeding through the following steps:

1. *Verify that the counter system clock is common to all flip-flop stages and that the circuit is in fact a synchronous counter.*
2. *Determine the number of stages of the counter* by counting the flip-flops or outputs.
3. *Determine the type of flip-flops and the input logic function for each stage.* For reference, recall the present state-present input-next state table for each flip-flop.
4. *Construct a present input-present state-next state table for the counter to determine the inputs to the flip-flops and the resulting outputs.*
5. *Analyze the counter using the present input-present state-next state table to determine the complete count sequence.* Continue the analysis until the count sequence begins to repeat.
6. *Determine the modulus of the counter.*
7. *Construct a state transition diagram to describe the counter operation.*
8. *Graph the output waveforms produced by the counter.*

The key to the synchronous counter analysis is the present state-present input-next state table that serves as a truth table description of the counter operation as it progresses with each clock pulse. To complete this table properly, we need to define correctly the flip-flops of each stage by their own present input-present state-next state table.

EXAMPLE (12): Synchronous Counter Analysis

Problem: Analyze the counter circuit shown in Figure (16).

Solution: The circuit in Figure (16) is a three-stage synchronous counter since a common clock signal controls all three J-K flip-flop stages of the counter. The J-K flip-flop operation is defined in Table (l).

The present input-present state-next table describing the counter operation is constructed with the initial present state of the flip-flops assumed to be a logic LOW. The J and K inputs can be specified according to their logic equations:

Present states: Q_C, Q_B, $Q_A = 0, 0, 0$.

Present inputs: $J_C = K_C = Q_A Q_B$.
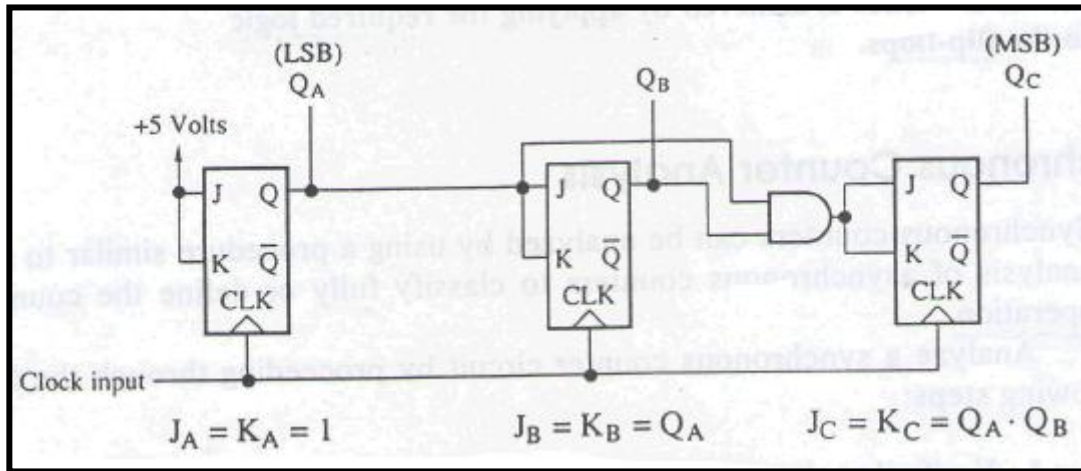 $J_B = K_B = Q_A$.
                    $J_A = K_A = 1$.

Figure (16): Example (12) Synchronous Counter.

Table (1): J-K Flip-Flop Operation.

Present Input		Present State	Next State	
J	K	Q_N	Q_{N+1}	
0	0	0	0	Hold
0	0	1	1	
0	1	0	0	Reset (Clear)
0	1	1	0	
1	0	0	1	Set
1	0	1	1	
1	1	0	1	Toggle
1	1	1	0	

The first entries in the present input-present state-next state table are shown in Table (2). The next output state is determined by the present inputs and present state of each flip-flop. Given the initial input conditions-the Q_A output will toggle and the Q_B and Q_c outputs will remain unchanged. The resulting 0, 0, 1 output state is recorded in the table as the next state.

Table (2): Example (12) Counter Analysis.

Present State			Present Input						Next State		
Q_C	Q_B	Q_A	J_C	K_C	J_B	K_B	J_A	K_A	Q_C	Q_B	Q_A
0	0	0	0	0	0	0	1	1	0	0	1

The analysis continues for all possible count states that can be generated 3 by a 3-bit counter. The next count state is determined by each present state 1 and the present inputs. The analysis of each present state and the resulting next state is shown in Table (3).

The counter sequence resulting from the analysis is defined in the state transition diagram shown in Figure (17), and the waveforms resulting from this

2

circuit are shown in Figure (18). Each output changes on the positive edge of the input clock. Note that the glitches occurring with the asynchronous counter do not occur with the synchronous counter.

Table (3): Example (12) Counter Analysis.
Present Input-Present State-Next State Table Completed

Present State			Present Input						Next State		
Q_C	Q_B	Q_A	J_C	K_C	J_B	K_B	J_A	K_A	Q_C	Q_B	Q_A
0	0	0	0	0	0	0	1	1	0	0	1
0	0	1	0	0	1	1	1	1	0	1	0
0	1	0	0	0	0	0	1	1	0	1	1
0	1	1	1	1	1	1	1	1	1	0	0
1	0	0	0	0	0	0	1	1	1	0	1
1	0	1	0	0	1	1	1	1	1	1	0
1	1	0	0	0	0	0	1	1	1	1	1
1	1	1	1	1	1	1	1	1	0	0	0

The counter classification is summarized as follows:
- MOD 8.
- Divide by 8.
- Three-bit.
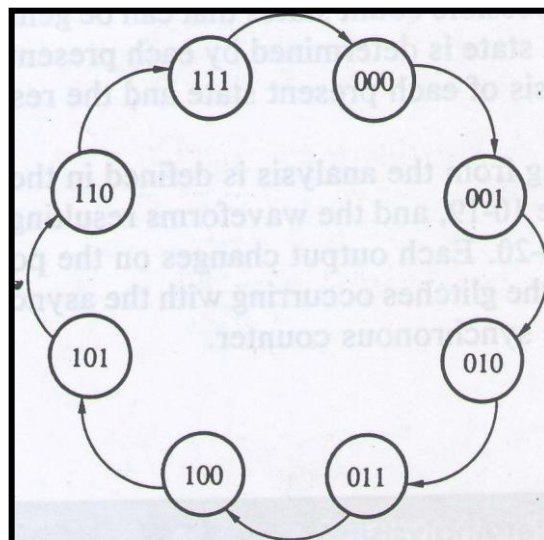- Synchronous.
- Binary up counter.

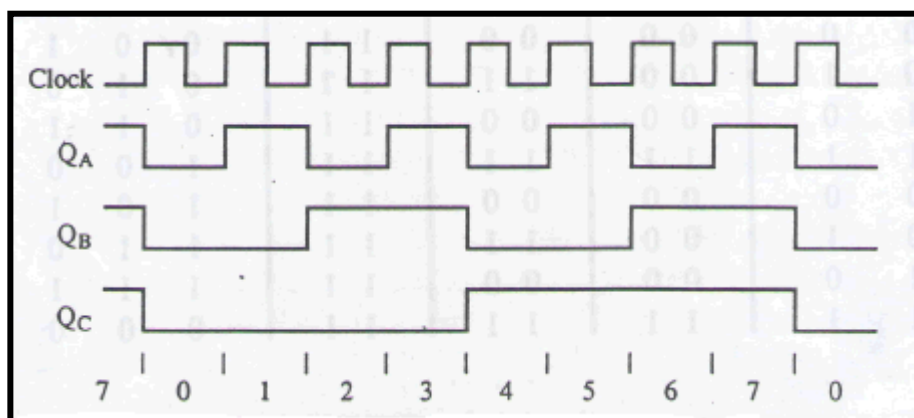Figure (17): Example (12) State Transition Diagram.

Figure (18): Example (12) Output Waveforms.

Synchronous counters can produce a count sequence that is not in counting order. Digital security locks might need a counter that generates a random sequence of numbers to serve as a password that must be matched by an appropriate input. Other applications that can use a non-serial count feature are ones where the counter output is decoded to enable certain circuits within a digital system. An example of such a counter is featured in Example (13).

EXAMPLE (13): Synchronous Counter Analysis
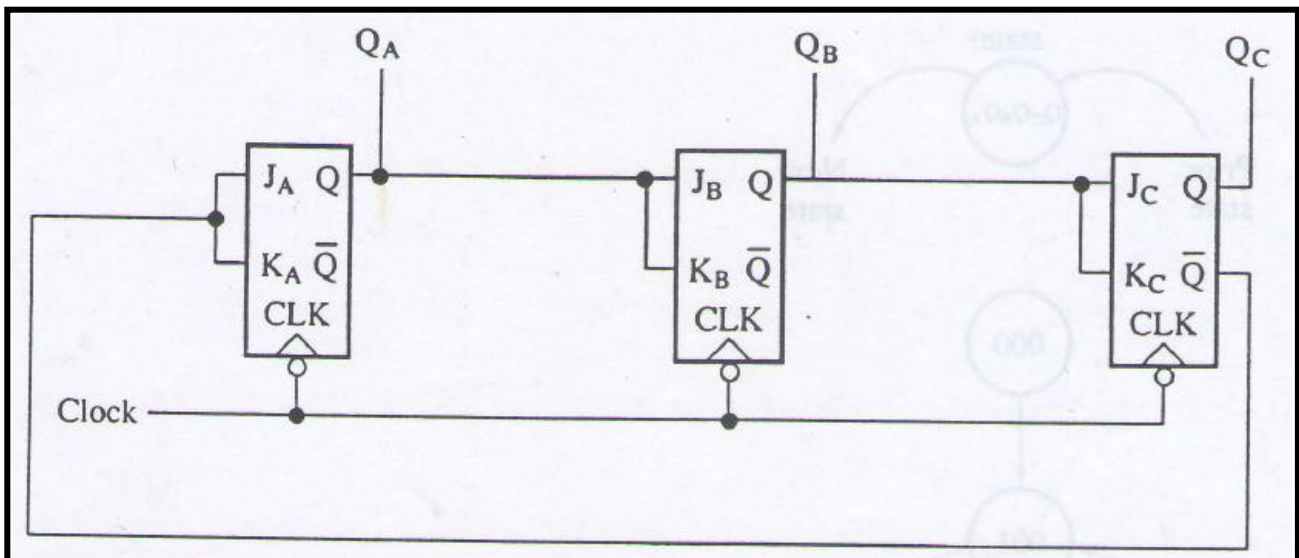***Problem:*** Determine the count sequence of the circuit shown in Figure (19).

Figure (19): Example (13) Synchronous Counter Analysis.

Solution: Analyze the counter by completing a present input-present state-next state table to determine the operation of the counter for all possible output states. The counter is a 3-bit counter. All 2^3 count states must be analyzed in the table. Therefore, set up the present states on the table in counting order, from 0 0 0 to 1 1 1, and determine their next state as a result of the inputs to each flip-flop. Present inputs:

$J_A = K_A = Q_C$.
$J_B = K_B = Q_A$.
$J_C = K_C = Q_B$.

Complete the analysis by graphing the state transition diagram. The present input-present state-next state table is shown in Table (4). The state transition diagram is shown in Figure (20).

From the state transition shown in Figure (20), it can be seen that this is a MOD-3 counter that has a "***lockout loop***". A ***lockout loop***, usually an undesirable situation for most applications, is a repetitive count sequence that is not part of the intended modulus of the counter. In this example the *lockout loop* can cause the counter to remain in the 1, 0, 0 state *forever* once it gets into that state. If this state is not desirable in a digital system, the counter should be redesigned to be ***self-starting*** so that all possible states lead to the count sequence.

Table (4): Example (13) Counter Analysis.
Present Input-Present State-Next State Table Completed

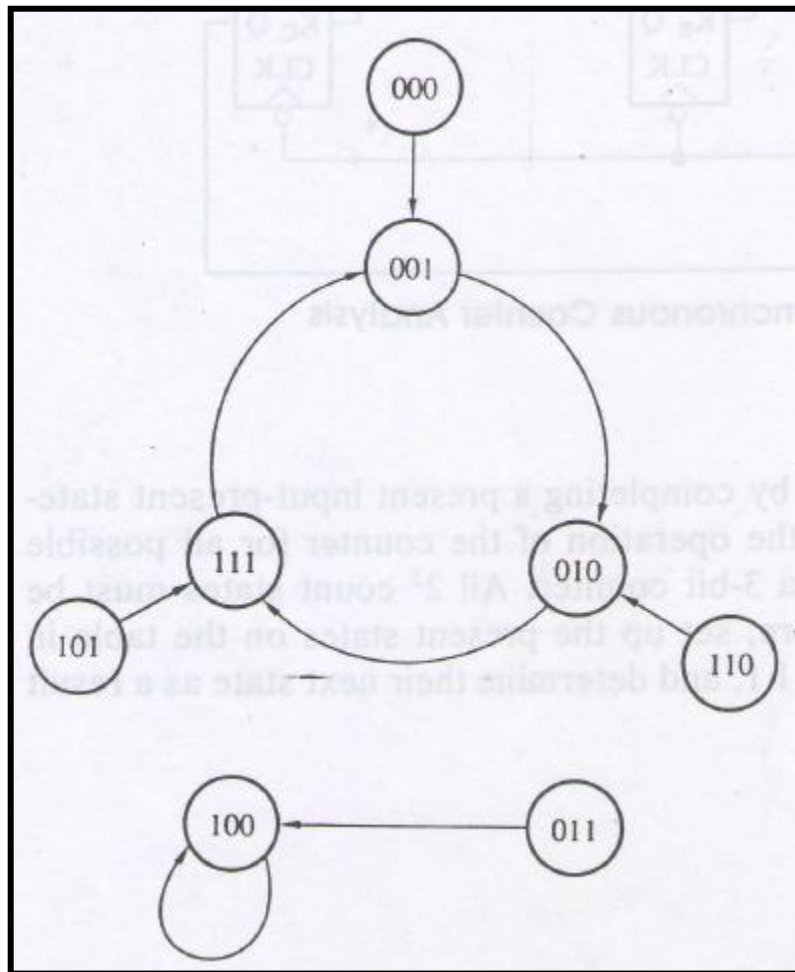| Present State			Present Input						Next State		
Q_C	Q_B	Q_A	J_C	K_C	J_B	K_B	J_A	K_A	Q_C	Q_B	Q_A
0	0	0	0	0	0	0	1	1	0	0	1
0	0	1	0	0	1	1	1	1	0	1	0
0	1	0	1	1	0	0	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	0	0
1	0	0	0	0	0	0	0	0	1	0	0
1	0	1	0	0	1	1	0	0	1	1	1
1	1	0	1	1	0	0	0	0	0	1	0
1	1	1	1	1	1	1	0	0	0	0	1

Figure (20): Example (13) State Transition Diagram.

Synchronous Counter Design

Synchronous counter design follows a systematic procedure to specify the count sequence required and to determine the input logic functions to obtain the desired count sequence. The flip-flop *excitation table* describes the input conditions that produce the output state from each individual J-K flip-flop. The excitation table for J-K flip-flops is shown in Table (5).

Table (5): J-K Excitation Table.

Present State Q_N	Next State Q_N	Inputs J	K	Condition
0	0	0	x	No Change, Reset
0	1	1	x	Toggle, Set
1	0	x	1	Toggle, Reset
1	1	x	0	No Change, Set

With synchronous counters, *the next output state from each flip-flop is determined by the present state and the present inputs* applied to that stage. The J-K excitation table forms the basis of synchronous counter design. An excitation table for the counter is constructed by specifying the present state and next state for each flip-flop in the order of the count sequence. For each transition, the necessary J and K inputs to produce the required sequence are listed. A logic function for each J and K input is then derived from counter excitation table. Example (14) illustrates a synchronous counter design.

EXAMPLE (14): Synchronous Counter Design

Problem: Design a MOD-5, 3-bit synchronous counter to count in the following sequence: 2, 3, 5, 1, 7. The counter must be self-starting with the count states of 0, 4, and 6 leading directly to 2.

Solution: Follow these procedures:
1. *Create a state transition diagram.*
2. *Create a counter excitation table* by listing the present state and next state sequence.
3. *Expand the table* by adding the J and K input states for each flip-
4. *Determine the logic functions for the J and K inputs as a function of the present states.*
5. *Analyze the counter to verify the design.*
6. *Construct and test the counter.*

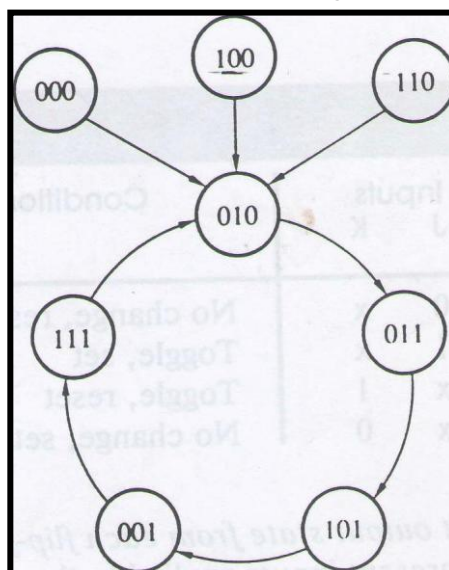The state transition diagram for the counter is shown in Figure (21).

Figure (21): Example (14) State Transition Diagram.

6

The present state-next state table is constructed, as shown in Table (6). The present state-next state table is expanded by adding the required J and K input to obtain the desired state transitions on each clock pulse. The expanded table is the counter excitation table, as shown in Table (7).

To determine the logic function required at each J and K input K-maps can be used to simplify the logic equations as functions of the present states of Q_C Q_B and Q_A Note that the present states listed in the excitation table below.

Table (6): Present State-Next State Table for Example (14).

Present State			Next State		
Q_C	Q_B	Q_A	Q_C	Q_B	Q_A
0	0	0	0	1	0
1	0	0	0	1	0
1	1	0	0	1	0
0	1	0	0	1	1
0	1	1	1	0	1
1	0	1	0	0	1
0	0	1	1	1	1
1	1	1	0	1	0

Table (7): Counter Excitation Table for Example (14).

Present State			Next State			Present Input					
Q_C	Q_B	Q_A	Q_C	Q_B	Q_A	J_C	K_C	J_B	K_B	J_A	K_A
0	0	0	0	1	0	0	X	1	X	0	X
1	0	0	0	1	0	X	1	1	X	0	X
1	1	0	0	1	0	X	1	X	0	0	X
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	0	1	1	X	X	1	X	0
1	0	1	0	0	1	X	1	0	X	X	0
0	0	1	1	1	1	1	X	1	X	X	0
1	1	1	0	1	0	X	1	X	0	X	1

Table (7), do not follow counting order, so be careful when filling in the K-maps. The K-maps and resulting logic functions for the J and K inputs are shown in Figure (22).

The resulting circuit to produce the required count sequence is shown in Figure (23).

The counter design should be analyzed to verify if it is correct, using the steps followed in Example (13). Finally, the counter circuit should be built and tested in the laboratory.

The synchronous counter design in Example (14) can be built with fewer additional logic gates required for the J and K inputs if the design requirements are slightly modified.

$Q_C Q_B$ \ Q_A	0	1
0 0	0	1
0 1	$\overline{0}$	1
1 1	x	x
1 0	x	x

$J_C = Q_A$ $K_C = 1$ by inspection of table

(a)

$Q_C Q_B$ \ Q_A	0	1
0 0	1	1
0 1	x	x
1 1	x	x
1 0	1	0

$J_B = \overline{Q_C Q_A}$

$Q_C Q_B$ \ Q_A	0	1
0 0	x	x
0 1	0	1
1 1	0	0
1 0	x	x

$K_B = \overline{Q_C} Q_A$

(b)

$Q_C Q_B$ \ Q_A	0	1
0 0	0	x
0 1	1	x
1 1	0	x
1 0	0	x

$J_A = \overline{Q_C} Q_B$

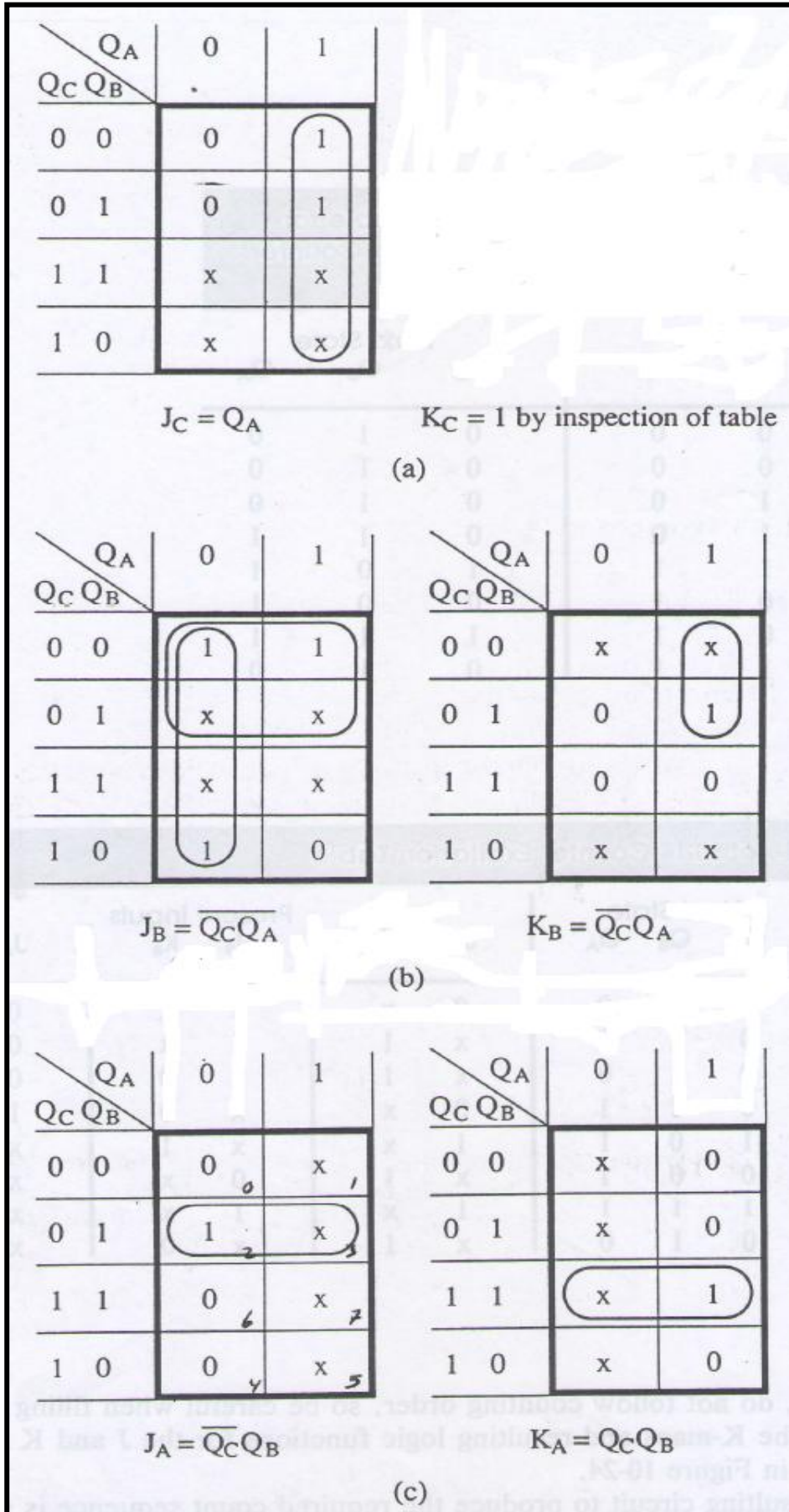| $Q_C Q_B$ \ Q_A	0	1
0 0	x	0
0 1	x	0
1 1	x	1
1 0	x	0

$K_A = Q_C Q_B$

(c)

Figure (22): Example (14):
(a) J_C and K_C Inputs; (b) J_B and K_B Inputs; (c) J_A and K_A Inputs.
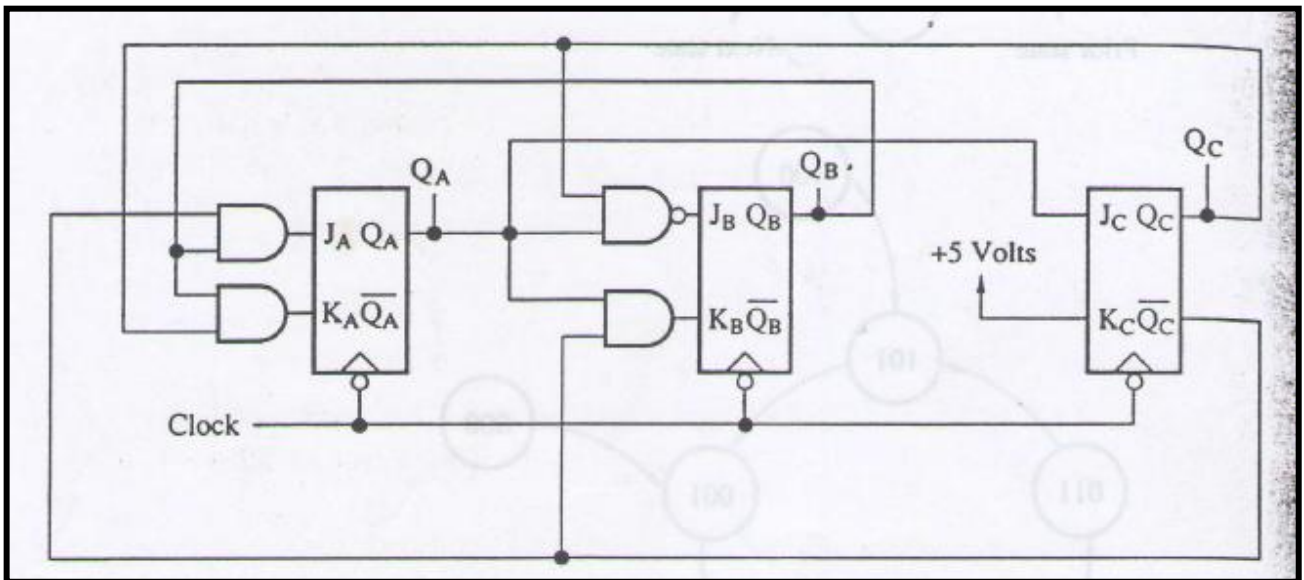
Figure (23): Example (14) Synchronous Counter.

Synchronous Counter Design Modification
_H.W.:_ Design a MOD-5, 3-bit synchronous counter to count in the following sequence: 2, 3, 5, 1, 7. To modify the design and minimize the additional gates required for the J and K inputs, route the unused states to their next natural count state. In other words, route 0 to 1, 4 to 5, and 6 to 7.

Section Self-Test
1. A synchronous counter has a common clock input to all flip-flop stages.
2. A synchronous counter of n flip-flop stages always has a modulus of 2^n.
3. A synchronous counter must count binary numbers in sequence, either up or down.

ANSWERS: (Not necessarily in the order of the questions).
     • True. • False. • False.