Create State Transition Tables For The Following(a) A Synchronous Counter That Counts From 0 To 7, Then

Create State Transition Tables For The Following(a) A Synchronous Counter That Counts From 0 To 7, Then

Designing a synchronous counter that counts from 0 to 7 involves understanding how to represent states, determine transition conditions, and formulate comprehensive state transition tables. These tables serve as a foundational tool in digital logic design, allowing engineers to visualize how the counter progresses through each state in a synchronized manner. By carefully analyzing the requirements, selecting appropriate flip-flops, and establishing clear state definitions, one can develop an accurate and functional state transition table that guides the implementation of the counter circuit.

Understanding Synchronous Counters

Definition and Characteristics

A synchronous counter is a sequential circuit where all flip-flops are triggered simultaneously by a common clock signal. Unlike asynchronous counters, which change state one after another, synchronous counters ensure that all state transitions occur in unison, reducing propagation delays and improving performance.

Counting Sequence

For this particular counter, the goal is to count from 0 to 7 repeatedly, which implies an 3-bit binary counter because 2^3 = 8 states (0 through 7). After reaching 7, the counter resets to 0, creating a cyclic counting sequence.

State Representation and Encoding

Binary Encoding of States

Each state corresponds to a unique 3-bit binary number, as follows:

- 0: 000
- 1: 001
- 2: 010
- 3: 011
- 4: 100
- 5: 101
- 6: 110
- 7: 111

State Variables

Let’s denote the flip-flops as:

- Q2 for the most significant bit (MSB)
- Q1 for the middle bit
- Q0 for the least significant bit (LSB)

Each state of the counter is then represented as (Q2, Q1, Q0).

Design Approach for the State Transition Table

Step 1: List All States

Identify all possible states from 0 to 7, corresponding to binary values:

1. 000 (0)

2. 001 (1)
3. 010 (2)
4. 011 (3)
5. 100 (4)
6. 101 (5)
7. 110 (6)
8. 111 (7)

Step 2: Determine Next States

Since the counter counts from 0 to 7 repeatedly, the next state is simply the current state + 1, with the transition from 7 back to 0.

Step 3: Establish State Transitions

For each current state, determine the next state based on the counting sequence, considering the reset after 7.

Constructing the State Transition Table

Table Format Overview

The table will have the following columns:

- **Current State**: (Q2, Q1, Q0)
- **Next State**: (Q2', Q1', Q0')
- **Input Conditions**: For a simple counter, inputs are often not required; transitions are driven by clock pulses.

State Transition Table for Counting 0 to 7

Current State (Q2 Q1 Q0)	Next State (Q2' Q1' Q0')
000	001
001	010
010	011
011	100
100	101
101	110
110	111
111	000

Deriving Flip-Flop Input Equations

Choosing Flip-Flops

For a 3-bit counter, common choices include JK, D, or T flip-flops. The selection depends on simplicity and availability. For illustrative purposes, we'll assume JK flip-flops, as they offer straightforward toggle behavior.

State Transition Equations

To determine the required inputs for each flip-flop, analyze the transition from current to next state:

- Q2 (MSB): toggles when the lower bits overflow from 110 to 111, and from 111 to 000.
- Q1: toggles when Q0 transitions from 1 to 0 during the count.
- Q0: toggles on every clock pulse, since it counts 0 to 1 repeatedly.

JK Flip-Flop Excitation Tables

- J = Q_next AND NOT Q + NOT Q_next AND Q (for toggling)
- K = same as J for toggling

Applying these principles, we derive the required flip-flop input equations for each transition.

Final State Transition Table with Flip-Flop Inputs

Incorporating Input Conditions

The complete table includes the current state, next state, and the required J and K inputs for each flip-flop:

Current State (Q2 Q1 Q0)	Next State (Q2' Q1' Q0')	J2, K2	J1, K1	J0, K0
000	001	0, X	0, X	1, X
001	010	0, X	1, X	1, X
010	011	0, X	0, X	1, X
011	100	1, X	1, X	1, X
100	101	0, X	0, X	1, X
101	110	0, X	1, X	1, X
110	111	1, X	1, X	1, X
111	000	1, X	1, X	0, X