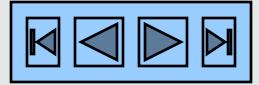




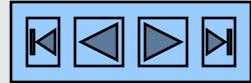
“COUNTERS”





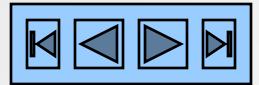
INTRODUCTION

- ✓ Sequential circuit that goes through predefined sequence of states upon application of triggering pulses.
- ✓ Group of flip-flops capable of counting a no. of clock pulses which have arrived at clock input of counter circuit
- ✓ Since pulses occur at known interval, counter can be used as an instrument for measuring time and therefore period or frequency
- ✓ Used for counting events in a circuit
- ✓ Used for generating timing sequence to control operations in digital circuit



ASYNCHRONOUS VS SYNCHRONOUS COUNTERS

ASYNCHRONOUS (RIPPLE)	SYNCHRONOUS
1st Flip-flop clocked by external clock pulse and then after successive flip-flops are triggered by the output of previous flip-flop	Every flip-flops are triggered by the clock simultaneously
Setting up time is cumulative delay time for each flip-flop	Setting time is simply equal to delay time of a single flip-flop
Have an overall count which 'ripples' through, meaning the overall operation is relatively slow	Faster as compared to Asynchronous
Very simple circuitry (require virtually no design)	Need designing, to determine how the present state of the circuit must be used to determine the next state thereby require more logic gates for their implementation



Up Counter



Down Counter



Up/Down Counter

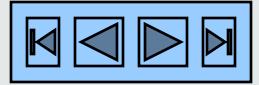


Mod-N Counter



Cascade Counter

Asynchronous



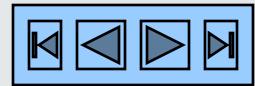
ASYNCHRONOUS COUNTER

- ✓ Also known as Ripple counter due to the way the FFs respond one after another in a kind of rippling effect. Each FF output drives the CLK input of the next FF
- ✓ FFs do not change states in exact synchronism with the applied clock pulses. There is delay between the responses of successive FFs.
- ✓ Ripple counters are the simplest type of binary counters because they require the fewest components to produce a given counting operation.

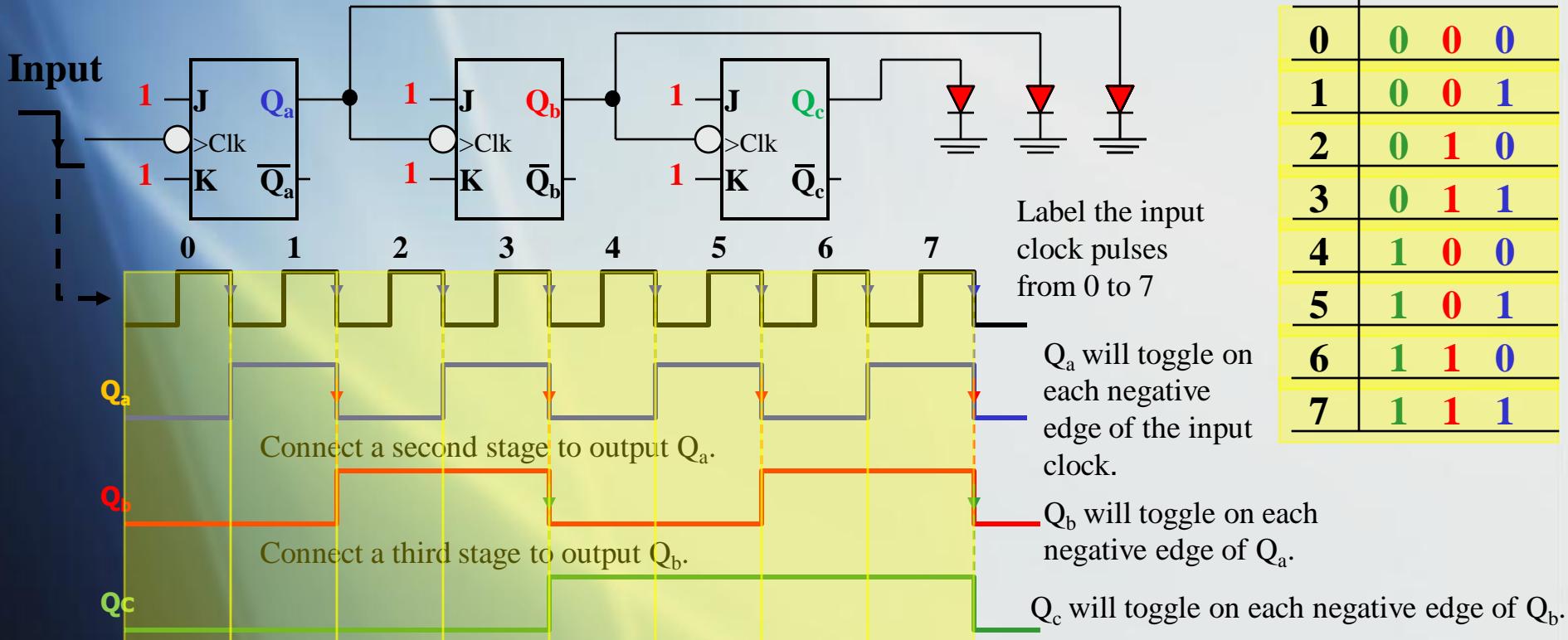
REMEMBER:

- N bit ---N flip-flop
- JK=1
- Clock asynchronous and Negative edge

THREE BIT UP RIPPLE COUNTER



3 JK flip flops connected in the toggle mode can be connected together to create a binary counter system. Start with one JK flip flop, apply a clock waveform and sketch the Q output response. Assume PRE and CLR has been disabled (=1) on all flip flops.

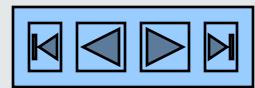


The speed at which the counter counts is controlled by input clock. 1 Hz input clock will display 0 to 7 count sequence on the LED's in 8 seconds. Each count state would last 1 sec..

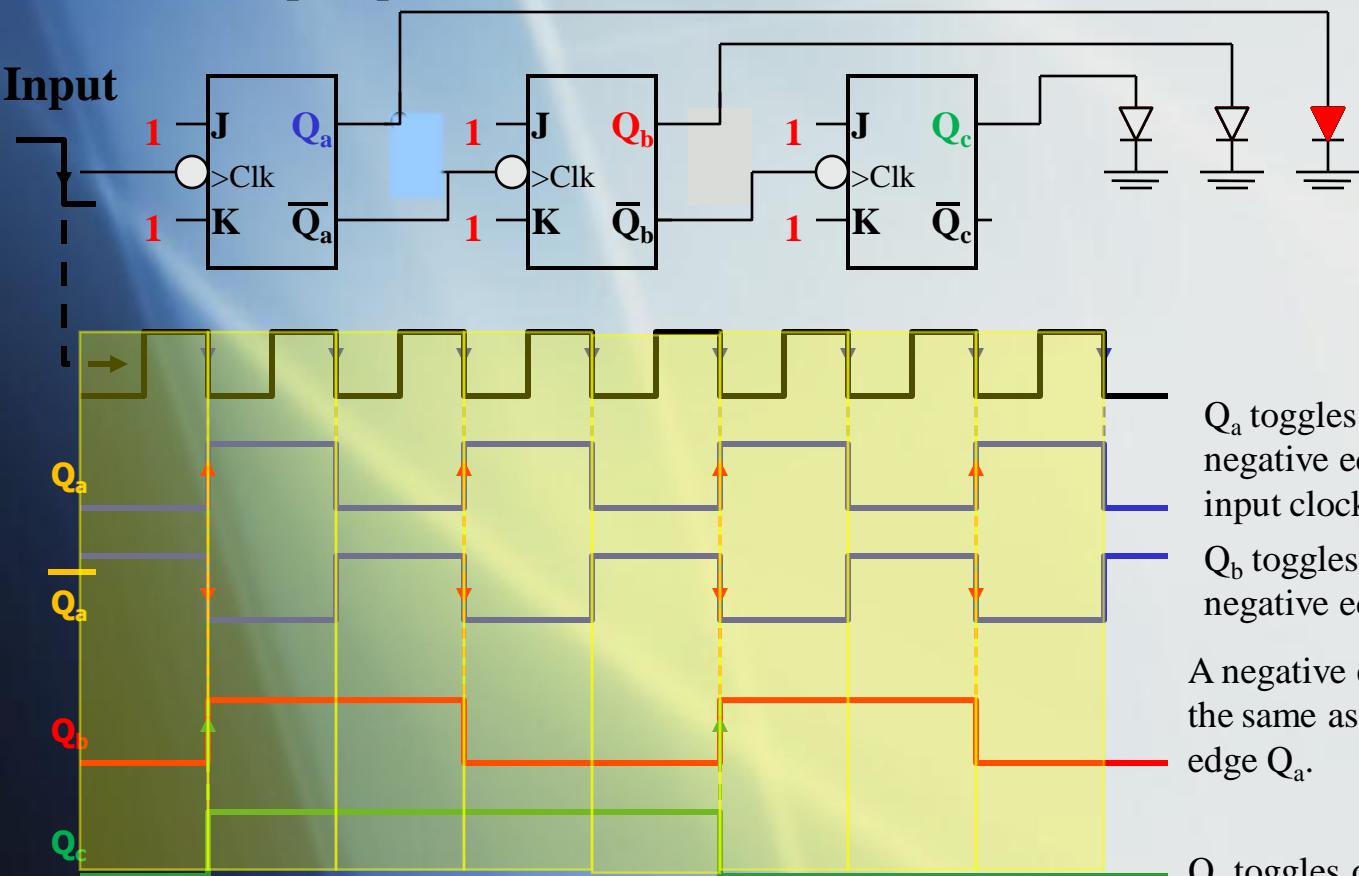
The table is a COUNT state table. The counter is called a MOD 8 counter because it has 8 different count states. The counter restarts at 0, 0, 0 after clock input 7.

Connect the flip flop outputs to 3 LED's and you will see a binary count sequence from 0 ... to ... 7.

THREE BIT DOWN RIPPLE COUNTER



To make a counter count backwards all you need to do is to connect the \overline{Q} to the Clock of the next flip flop.



In	Q _c	Q _b	Q _a
0	0	0	0
1	1	1	1
2	1	1	0
3	1	0	1
4	1	0	0
5	0	1	1
6	0	1	0
7	0	0	1

Q_a toggles on every negative edge of the input clock.

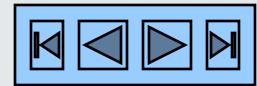
Q_b toggles on every negative edge of Q'_a .

A negative edge on Q'_a is the same as the positive edge Q_a .

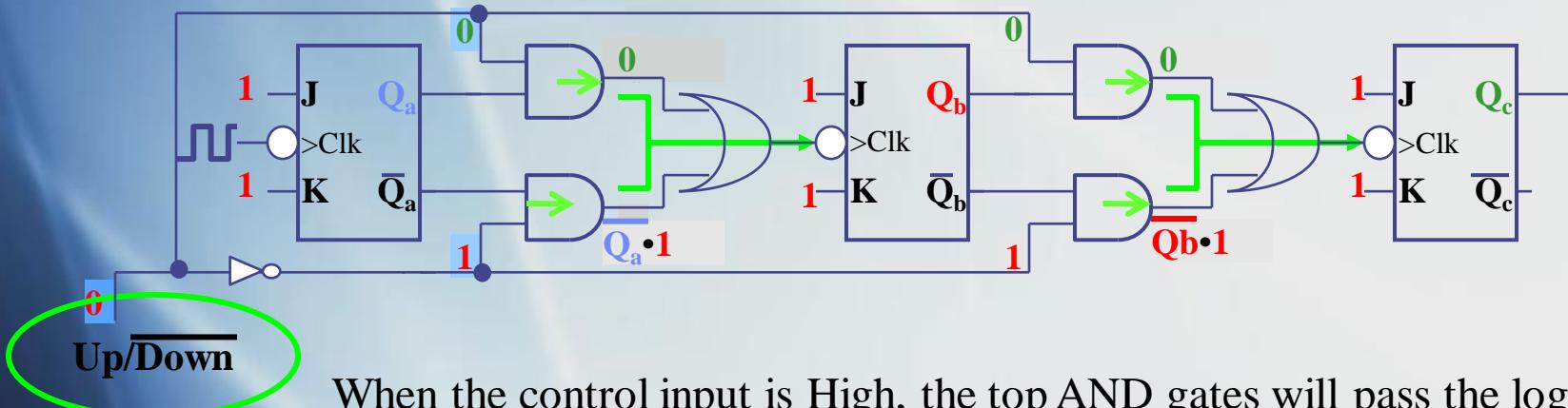
Q_c toggles on every negative edge of Q'_b . Which is the same as the positive edge of Q_b .

If you place the count states in a table you can see the down count sequence.

THREE BIT UP/DOWN RIPPLE COUNTER



This system combines the features of both an up and a down counter. The system has a count direction control input to select up counting or down counting.

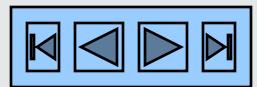


When the control input is High, the top AND gates will pass the logic levels from the Q outputs.

The bottom AND gates output 0. The OR gate outputs a $Q \cdot 1 + 0 = Q$. This connects Q to clock and the counter counts up or forward.

When the control input is low, the bottom AND gates pass the logic levels from the Q' outputs.

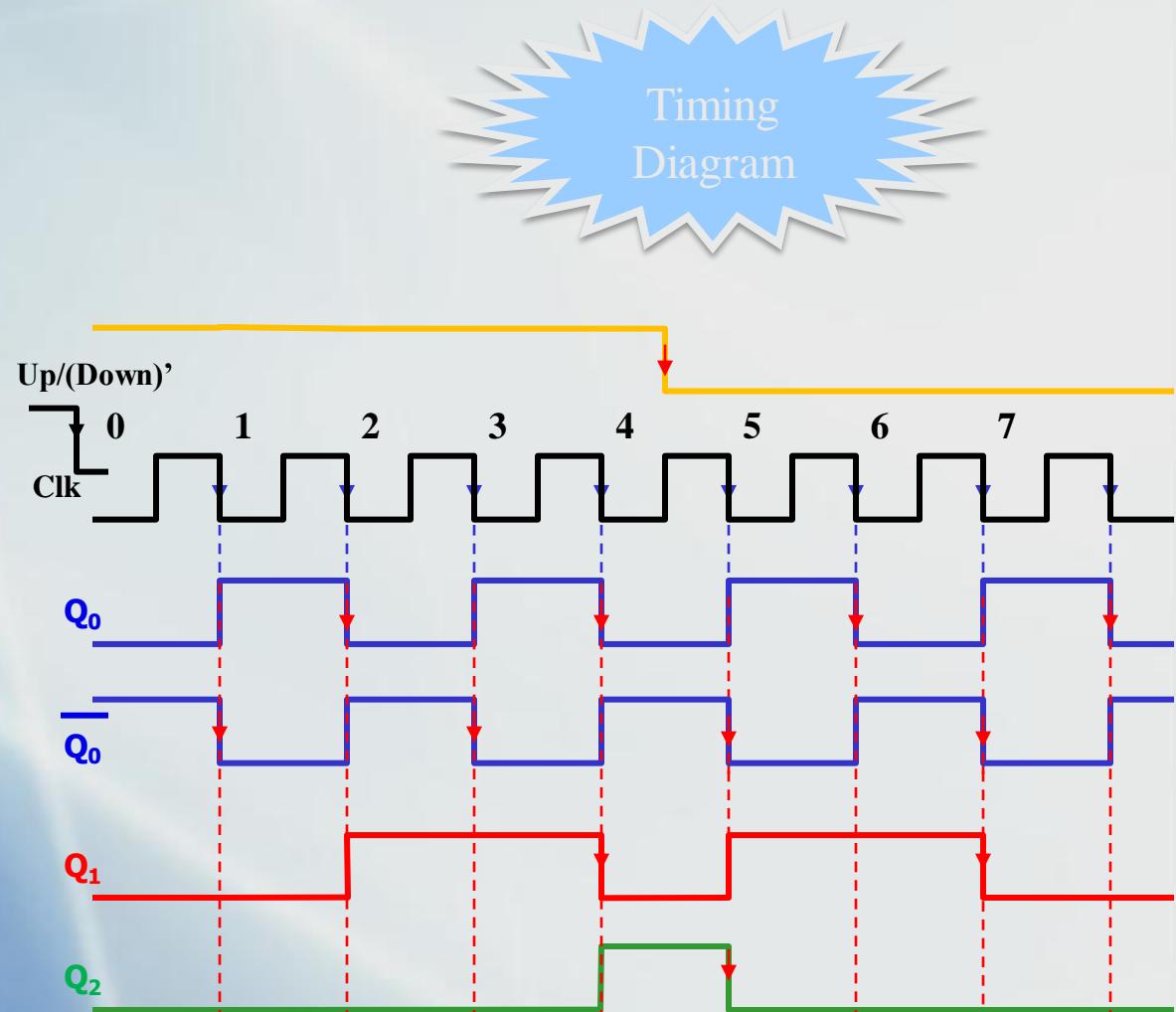
The top AND gates output 0. The OR gate outputs a $Q' \cdot 1 + 0 = Q'$. This connects Q' to clock and the counter counts down or backwards.

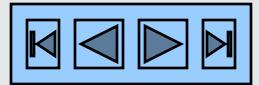


THREE BIT UP/DOWN RIPPLE COUNTER

Clk	Up/(down)'	Q_2	Q_1	Q_0
0	x	0	0	0
1	1 (Up)	0	0	1
2	1	0	1	0
3	1	0	1	1
4	1	1	0	0
5	0 (Down)	0	1	1
6	0	0	1	0
7	0	0	0	1

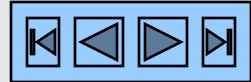
Characteristic Table



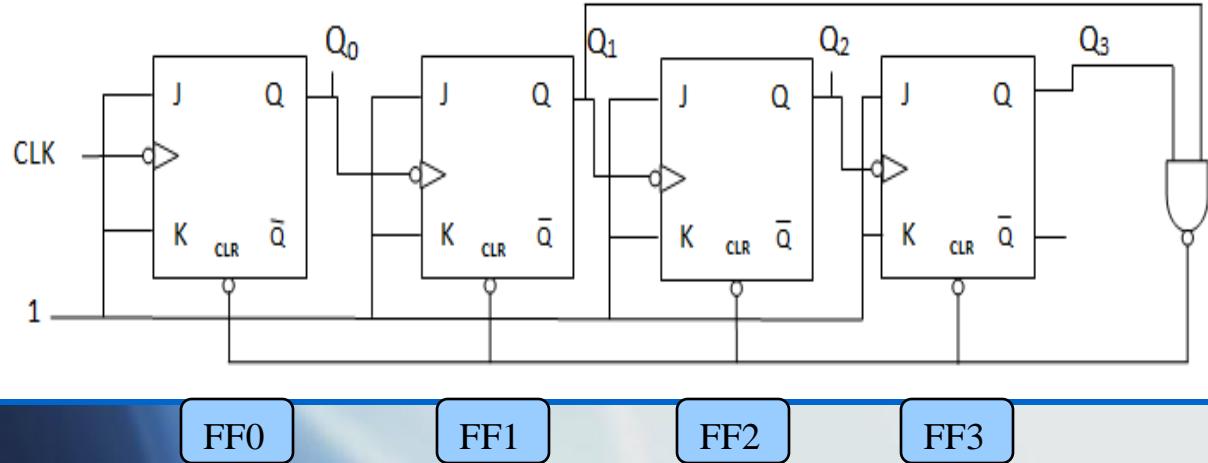


MOD-N ASYNCHRONOUS COUNTER (Divide by N)

- ✓ Basic counter is limited to MOD numbers that are equal to 2^N , where N is the number of FF's
- ✓ Required when desired to count to base N which is not a power of 2
- ✓ Counter can skip some states Eg: Decade Counter (Base=10) skips 11 to 15
- ✓ A feedback NAND gate is added feeding to CLEAR input in parallel to all the flip-flops such that at the count of N, all flip-flops are reset to zero



MOD-10 ASYNCHRONOUS COUNTER

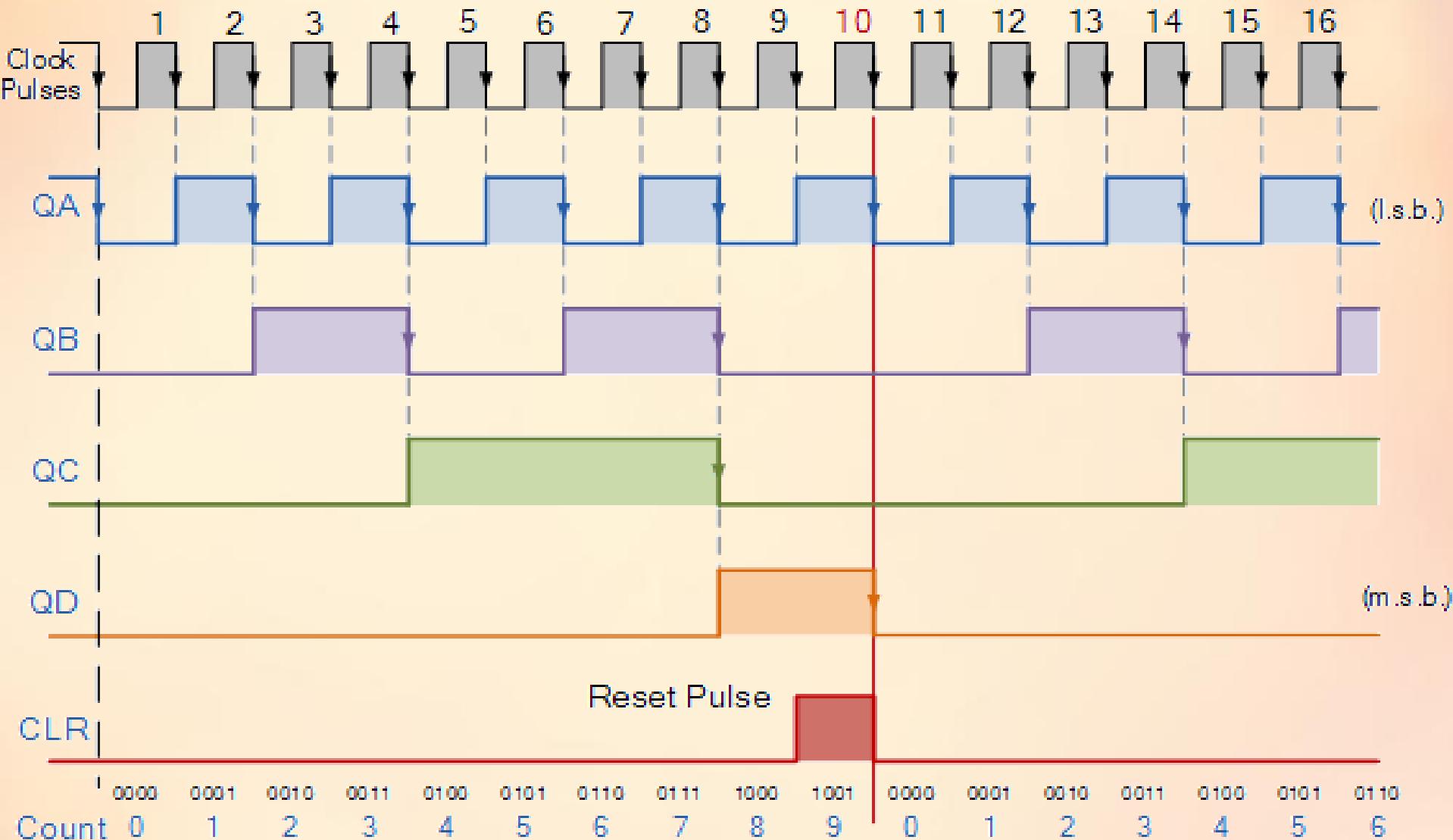


Clk	Q ₃	Q ₂	Q ₁	Q ₀
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

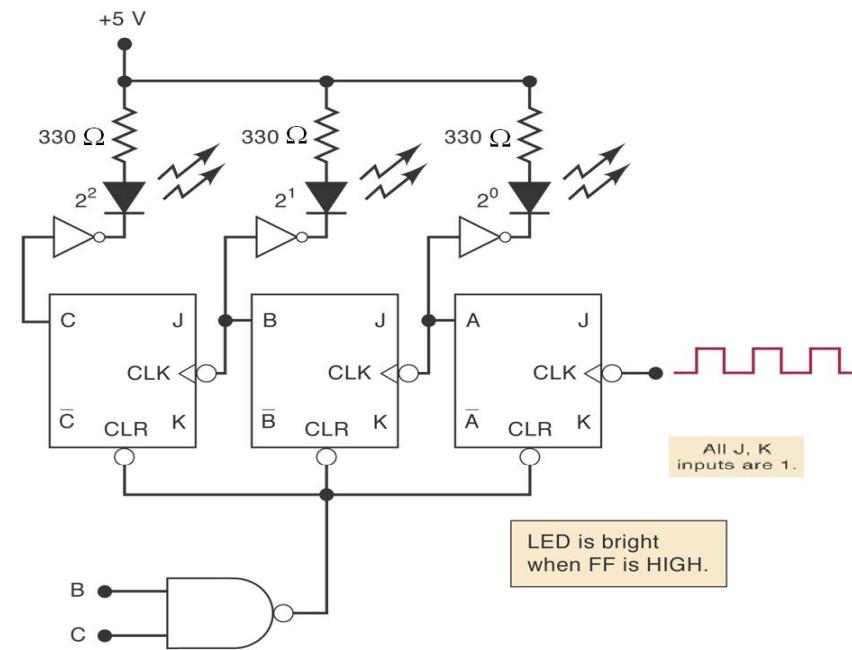
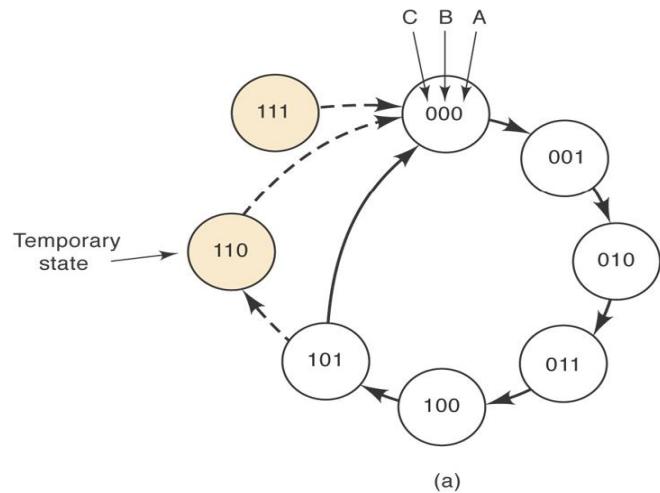
- ✓ Decade Counter, Divide by 10 Counter
- ✓ Counts from 0 (0000) to 9 (1001)
- ✓ (1001)----4 bits required, so 4 JK flip-flop; JK=1
- ✓ Clock asynchronously provided
- ✓ A feedback NAND gate is added feeding to CLEAR input in parallel to all the flip-flops such that at the count of 10, all flip-flops are reset to zero
- ✓ Characteristic table is shown alongside



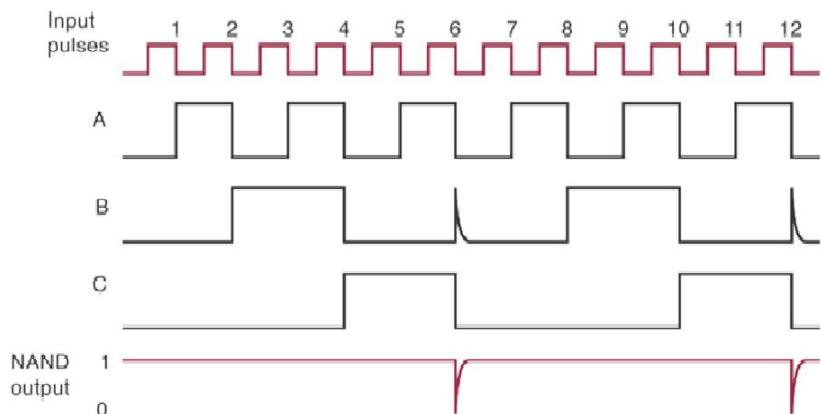
MOD-10 ASYNCHRONOUS COUNTER: Timing Diagram



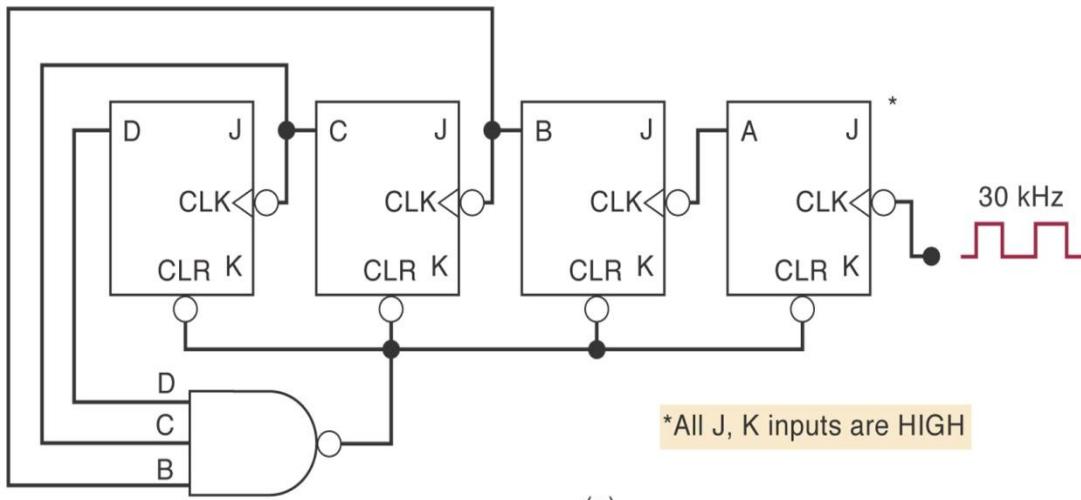
SOME OTHER MOD-N COUNTER



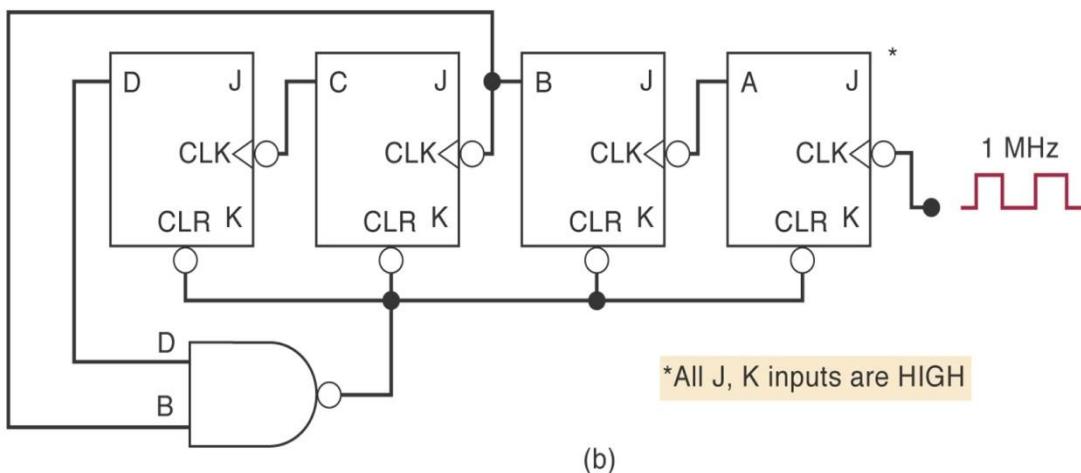
- ✓ MOD-6 asynchronous counter produced by clearing a MOD-8 counter when a count of six (110) occurs.
- ✓ State 110 will not be visible, because it lasts for few nanoseconds as the counter recycles to 000



SOME OTHER MOD-N COUNTER



(a)



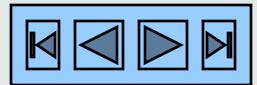
(b)

a) MOD-14 ripple counter:

$$\begin{aligned} \text{At D, freq} &= 30/14 \\ &= 2.14\text{kHz} \end{aligned}$$

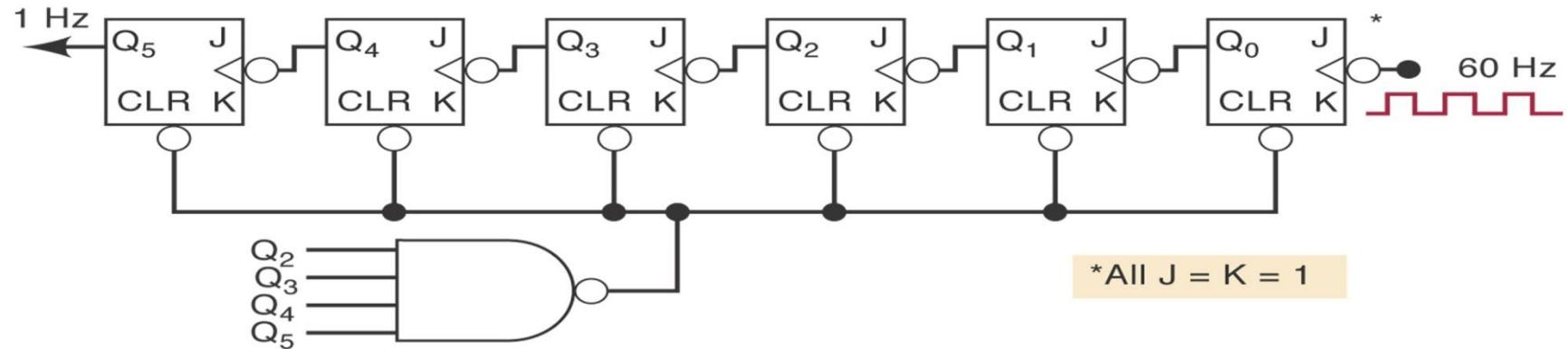
b) MOD-10 (decade) ripple counter:

$$\begin{aligned} \text{At D, freq} &= 1/10 \\ &= 0.1\text{MHz} \end{aligned}$$



SOME OTHER MOD-N COUNTER

MOD-60 ripple counter.



000000, ..., 111011, Reset at 60

0

59

111100, 111101, 111110, 111111

60

61

62

63

The count (111100) appears momentarily as a glitch

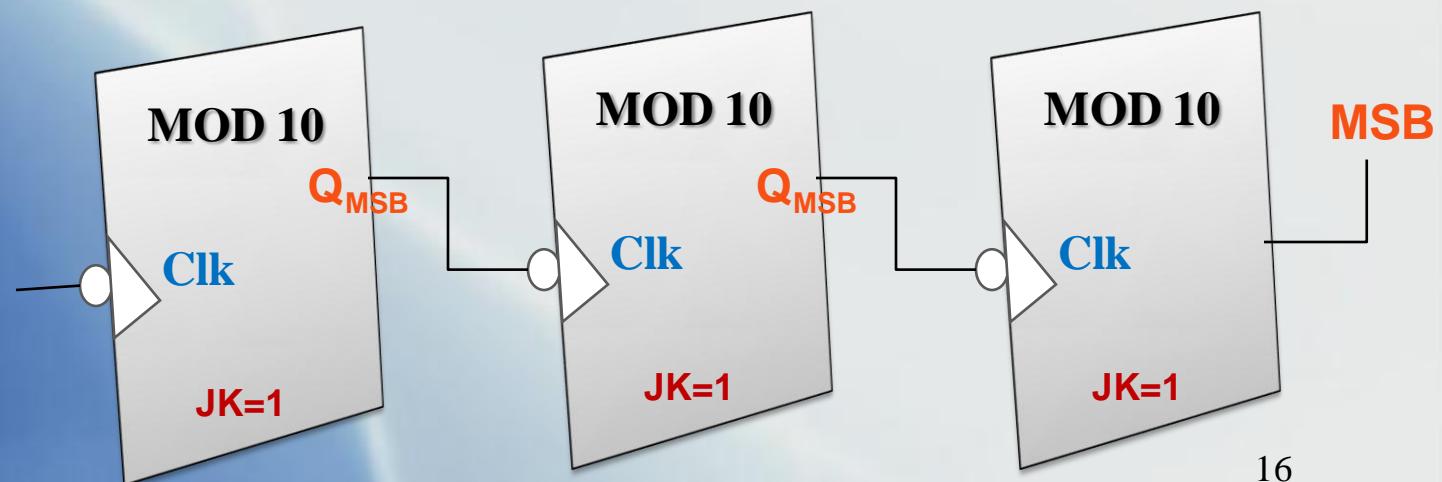
Skipped

ASYNCHRONOUS CASCADE COUNTER

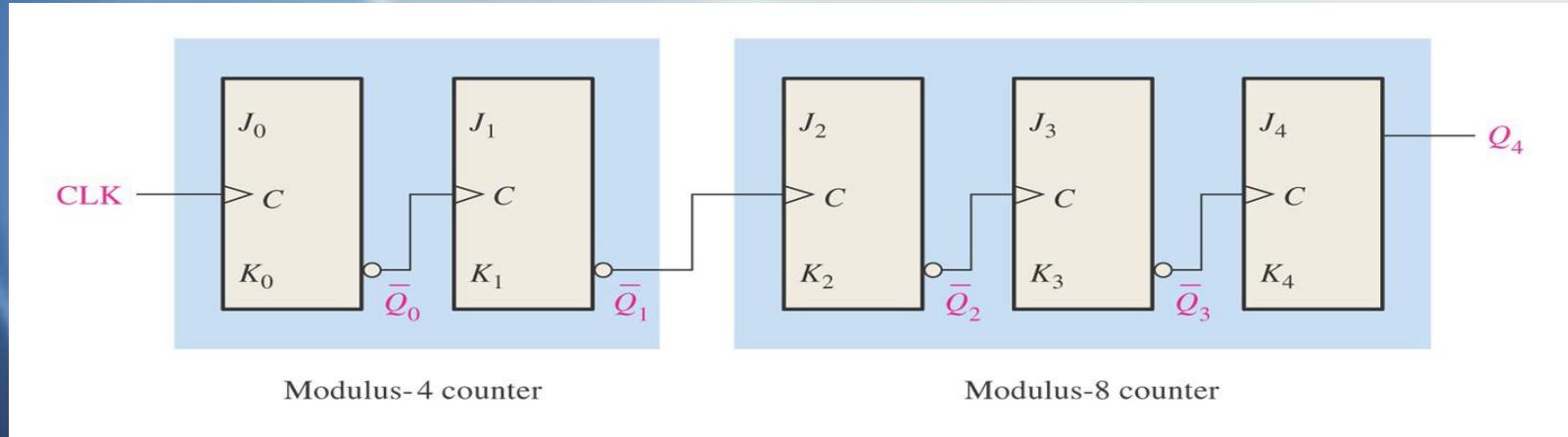
- ✓ Counters are cascaded or connected together in asynchronous way to achieve higher modulus operation
- ✓ The overall modulus of a Cascaded counter = Product of individual modulus of a counter
- ✓ Last stage of a counter drives the clock of the next counter

MOD 1000 COUNTER

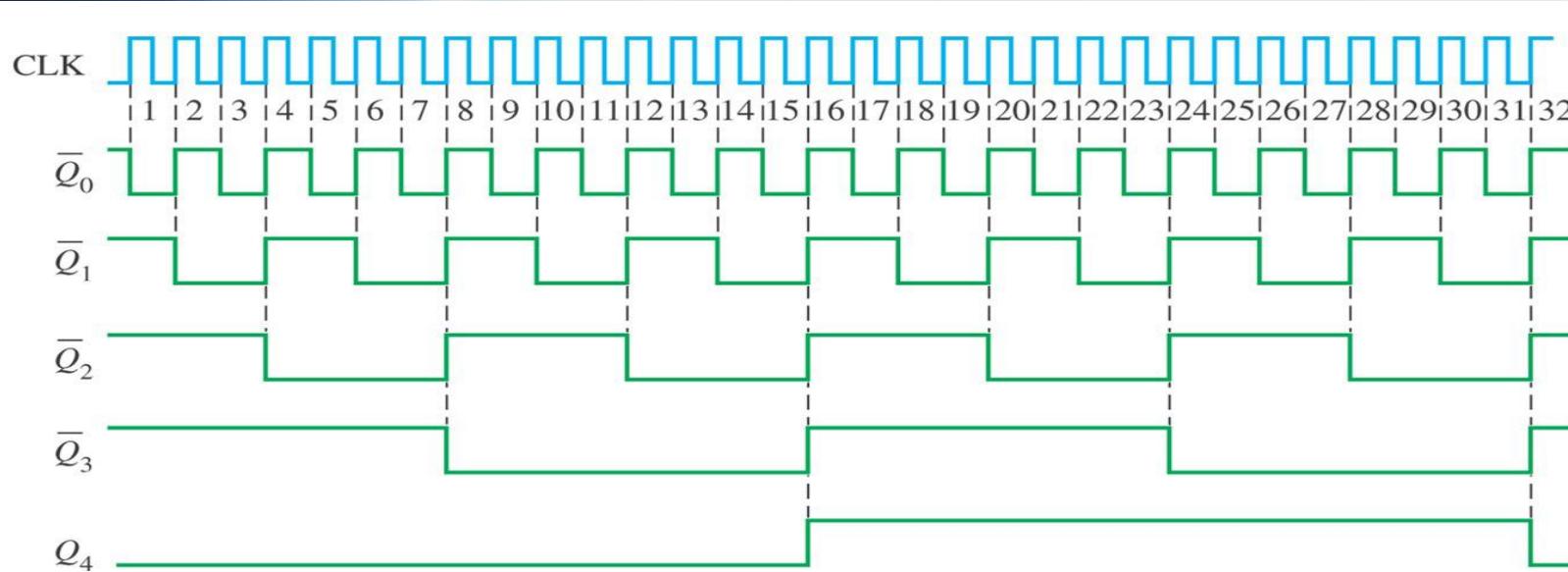
Mod 1000 =
 Mod 10 *
 Mod 10 *
 Mod 10



CASCADE COUNTER

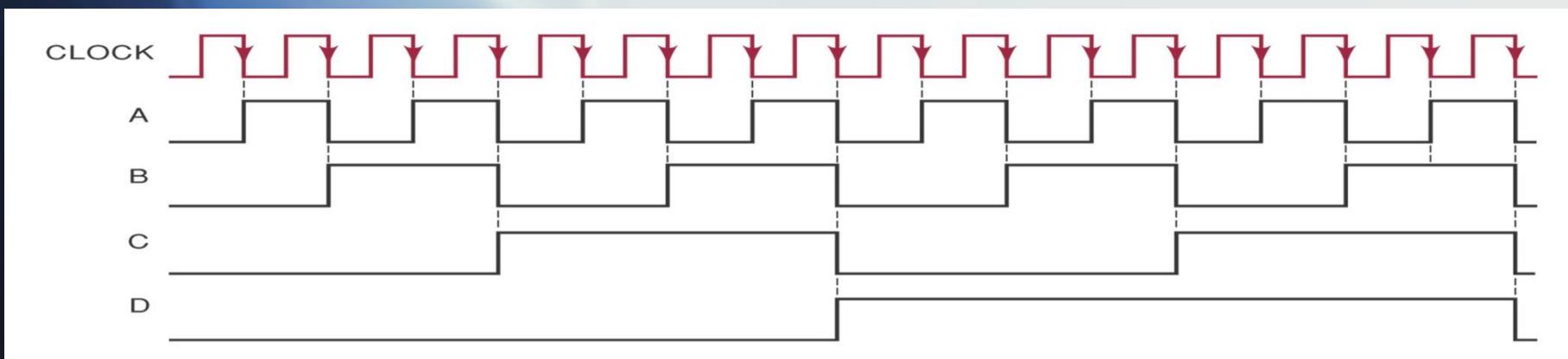


- ✓ Cascading a mod-4 and mod-8 counter yields a mod-32 counter.
- ✓ Note that the mod number is 2 raised to the number of outputlines $\Rightarrow 2^5 = 32$
- ✓ There are 32 unique states for this counter. This counter counts in binary.

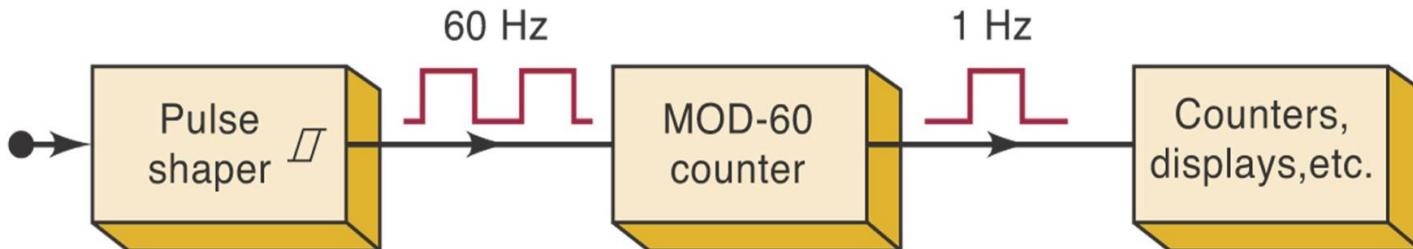


SOME POINTS TO REMEMBER

- ✓ Frequency division – each FF will have an output frequency of $\frac{1}{2}$ the input. The output frequency of the last FF of any counter will be the clock frequency divided by the MOD of the counter.
- ✓ 2 Bit FF----Counts 0-3 So, Mod 4 Counter, Divide frequency by 4
- ✓ 3 Bit FF----Counts 0-7 So, Mod 8 Counter, Divide frequency by 8
- ✓ 4 Bit FF----Counts 0-15 So, Mod 16 Counter, Divide frequency by 16
- ✓ Timing diagram for 4 bit FF, Output A(LSB)---Clk frequency/2
B---Clk frequency/4 C---Clk frequency/8 D (MSB)---Clk frequency/16



FREQUENCY DIVISION CONTINUED



Example generating clock from power line

MOD-60 counter

$$2^5 = 32 < 60, 2^6 = 64 > 60$$

- 6 FF's are needed, but the circuitry can be modified to do 60 and becomes MOD-60

SYNCHRONOUS COUNTER

Up
counter

Cascade
counter

Down
counter

Mod N
counter

Up/Down
counter



SYNCHRONOUS COUNTER

- ✓ Output bits change state simultaneously, with no ripple
- ✓ Synchronous refers to events that have a fixed time relationship with each other and receive clock pulse from a common source i.e. all flip-flops are simultaneously clocked by an external clock.
- ✓ Synchronous counters can operate at much higher frequencies than asynchronous counters so are faster than asynchronous because of the simultaneous clocking.

DESIGN STEPS

- State Diagram
- Next State Table
- Transition/ Excitation table
- K-Map
- Diagram

REMEMBER:

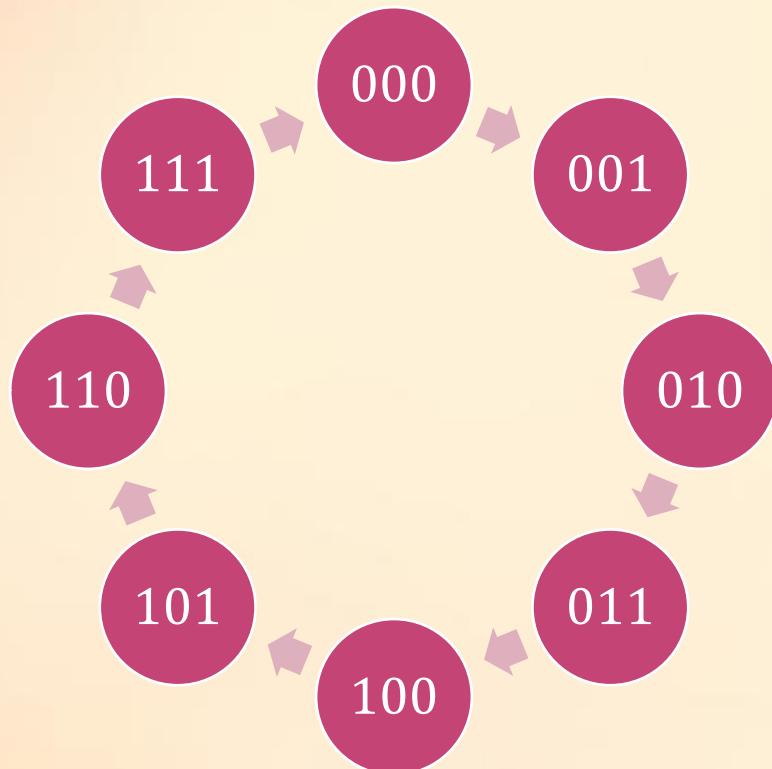
- N bit ---N flip-flop
- JK FF
- Clock synchronous and Negative edge

THREE BIT UP SYNCHRONOUS COUNTER



- ✓ 3 BIT—3 FF, JK or T Flip-flop, Clock synchronous and -ve edge
- ✓ Counts from 0 to 7

Step 1: State Diagram



Step 2: Next State Table

Previous			Next		
Q ₂	Q ₁	Q ₀	Q ₂	Q ₁	Q ₀
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	0



Excitation table for T FF

Q_p	Q_n	T
0	0	0
0	1	1
1	0	1
1	1	0

Excitation table for SR FF

Q_p	Q_n	S	R
0	0	0	x
0	1	1	0
1	0	0	1
1	1	x	0

Excitation table for D FF

Q_p	Q_n	D
0	0	0
0	1	1
1	0	0
1	1	1

Excitation table for JK FF

Q_p	Q_n	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

THREE BIT UP SYNCHRONOUS COUNTER



Step 3: Transition/Excitation Table

Q_p	Q_n	T
0	0	0
0	1	1
1	0	1
1	1	0

For T_0

Q_{p0}	Q_{n0}	T_0
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1

For T_1

Q_{p1}	Q_{n1}	T_1
0	0	0
0	1	1
1	1	0
1	0	1
0	0	0
0	1	1
1	1	0
1	0	1

For T_2

Q_{p2}	Q_{n2}	T_2
0	0	0
0	0	0
0	0	0
0	1	1
1	1	0
1	1	0
1	0	1

THREE BIT UP SYNCHRONOUS COUNTER



Step 4: K-Map

For T_0

		$Q_1 Q_0$	00	01	11	10
		0	1	1	1	1
Q_2	0	1	1	1	1	

$$T_0 = 1$$

For T_1

		$Q_1 Q_0$	00	01	11	10
		0	0	1	1	0
Q_2	0	0	1	1	0	

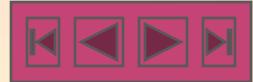
$$T_1 = Q_0$$

For T_2

		$Q_1 Q_0$	00	01	11	10
		0	0	0	1	0
Q_2	0	0	0	1	0	

$$T_2 = Q_1 Q_0$$

THREE BIT UP SYNCHRONOUS COUNTER

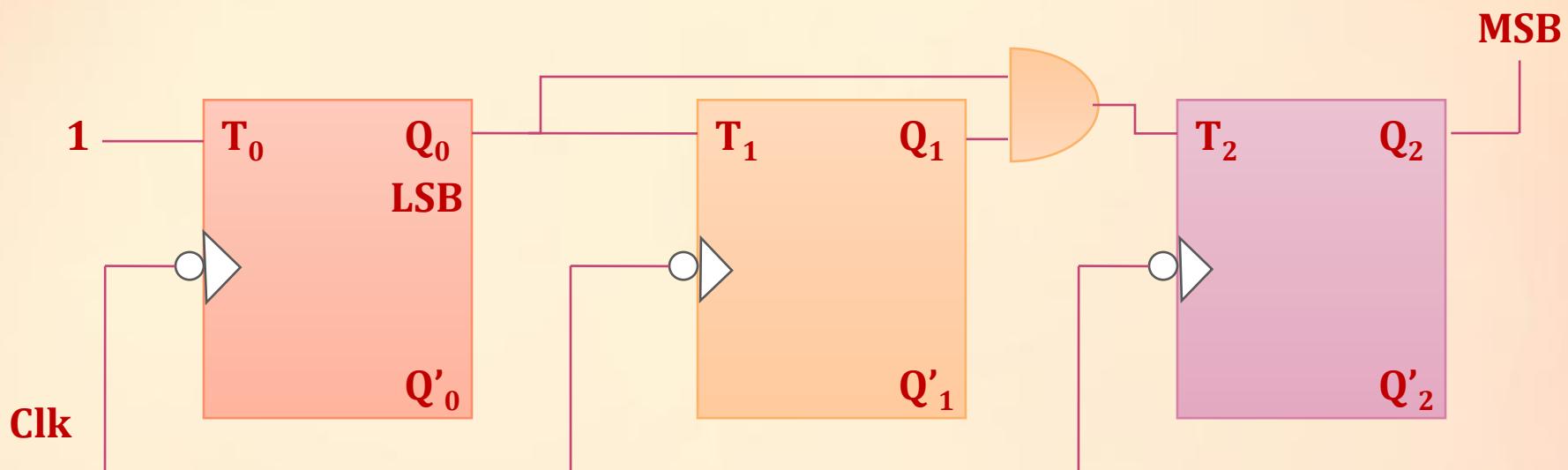


Step 5: Diagram

$$T_0 = 1$$

$$T_1 = Q_0$$

$$T_2 = Q_1 Q_0$$

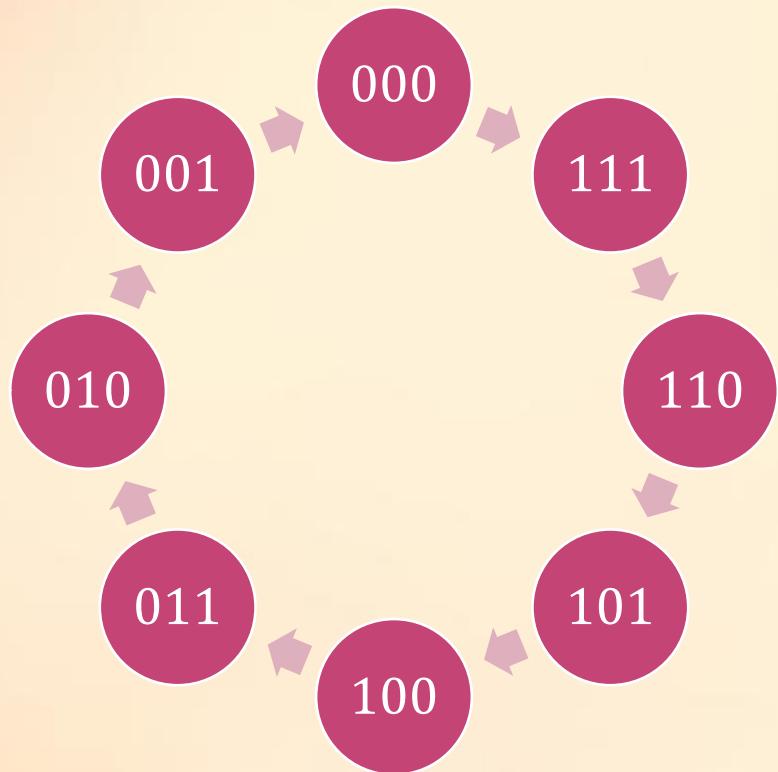


THREE BIT DOWN SYNCHRONOUS COUNTER



- ✓ 3 BIT—3 FF, JK or T Flip-flop, Clock synchronous and -ve edge
- ✓ Counts from 0, 7 to 0

Step 1: State Diagram



Step 2: Next State Table

Previous			Next		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0
0	0	0	1	1	1
0	0	1	0	0	0
0	1	0	0	0	1
0	1	1	0	1	0
1	0	0	0	1	1
1	0	1	1	0	0
1	1	0	1	0	1
1	1	1	1	1	0

THREE BIT DOWN SYN. COUNTER



Step 3: Transition/Excitation Table

Q_p	Q_n	T
0	0	0
0	1	1
1	0	1
1	1	0

For T_0

Q_{p0}	Q_{n0}	T_0
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1

For T_1

Q_{p1}	Q_{n1}	T_1
0	1	1
0	0	0
1	0	1
1	1	0
0	1	1
0	0	0
1	0	1
1	1	0

For T_2

Q_{p2}	Q_{n2}	T_2
0	1	1
0	0	0
0	0	0
0	0	0
1	0	1
1	1	0
1	1	0
1	1	0

THREE BIT DOWN SYN. COUNTER



Step 4: K-Map

For T_0

		$Q_1 Q_0$	00	01	11	10
		0	1	1	1	1
Q_2	0	1	1	1	1	1

$$T_0 = 1$$

For T_1

		$Q_1 Q_0$	00	01	11	10
		0	1	0	0	1
Q_2	0	1	0	0	1	1

$$T_1 = Q'_0$$

For T_2

		$Q_1 Q_0$	00	01	11	10
		0	1	0	0	0
Q_2	0	1	0	0	0	0

$$T_2 = Q'_1 Q'_0$$

THREE BIT DOWN SYN. COUNTER

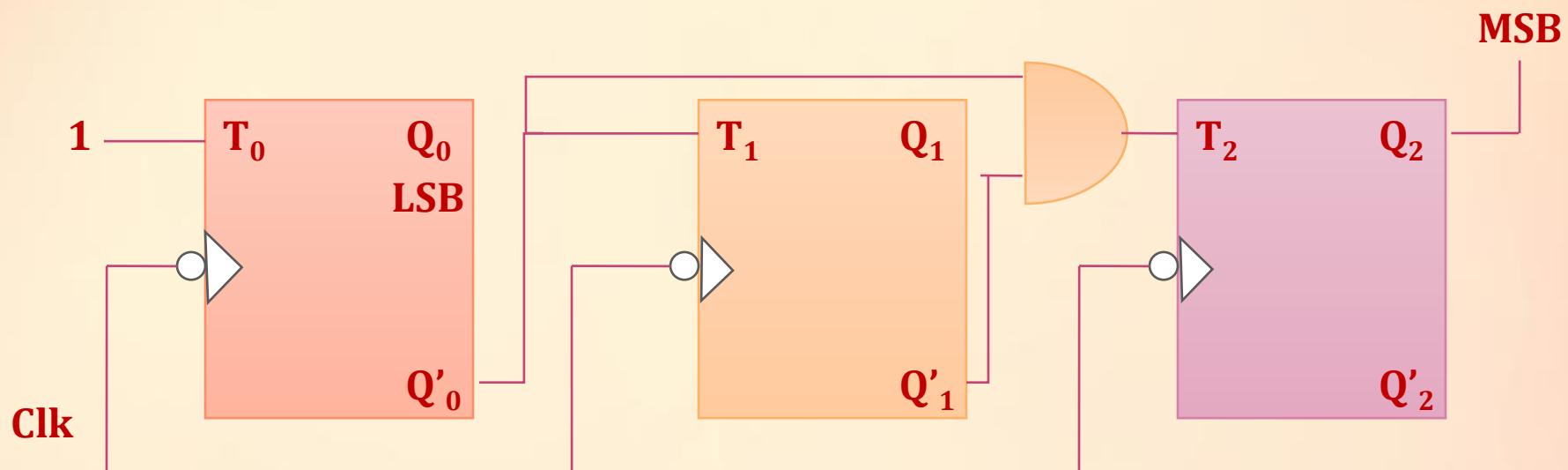


Step 5: Diagram

$$T_0 = 1$$

$$T_1 = Q'_0$$

$$T_2 = Q'_1 Q'_0$$

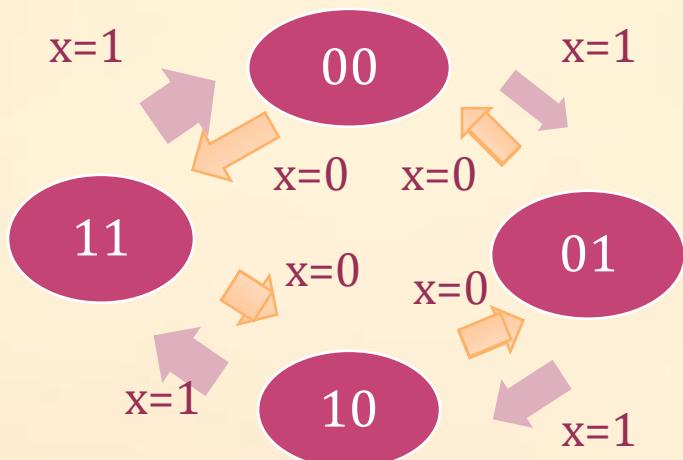


TWO BIT UP/DOWN SYN. COUNTER



- ✓ 2 BIT—2 FF, JK or T Flip-flop, Clock synchronous and -ve edge
- ✓ Counts from 0 to 3 (upwards when $x=1$) or 3 to 0 (downwards when $x=0$)

Step 1: State Diagram



Step 2: Next State Table

Previous		Next			
		X=0		X=1	
Q ₁	Q ₀	Q ₁	Q ₀	Q ₁	Q ₀
0	0	1	1	0	1
0	1	0	0	1	0
1	0	0	1	1	1
1	1	1	0	0	0

TWO BIT UP/DOWN SYN. COUNTER



Step 3: Transition/Excitation Table

Q_p	Q_n	T
0	0	0
0	1	1
1	0	1
1	1	0

At X=0

Q_{p0}	Q_{n0}	T_0
0	1	1
1	0	1
0	1	1
1	0	1

For T_0

Q_{p1}	Q_{n1}	T_1
0	1	1
0	0	0
1	0	1
1	1	0

For T_1

At X=1

Q_{p0}	Q_{n0}	T_0
0	1	1
1	0	1
0	1	1
1	0	1

For T_0

Q_{p1}	Q_{n1}	T_1
0	0	0
0	1	1
1	1	0
1	0	1

For T_1

TWO BIT UP/DOWN SYN. COUNTER



Step 4: K-Map

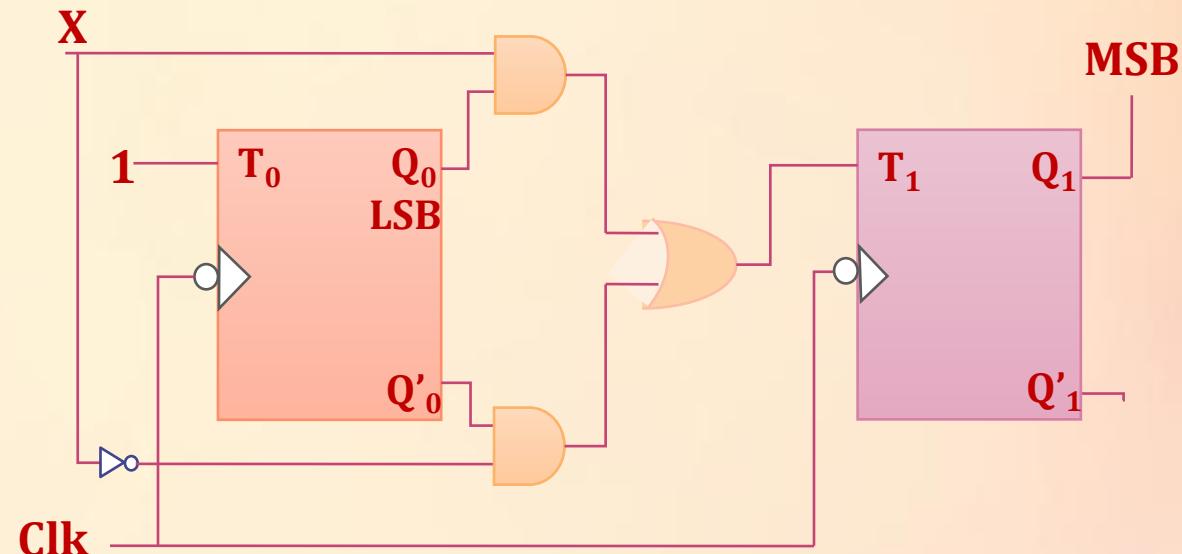
For T_0

X	$Q_1 Q_0$	00	01	11	10
0	1	1	1	1	1
1	1	1	1	1	1

Step 5: Diagram

$$T_0 = 1$$

$$T_1 = X'Q'_0 + XQ_0$$



For T_1

X	$Q_1 Q_0$	00	01	11	10
0	1	0	0	1	0
1	0	1	1	0	0

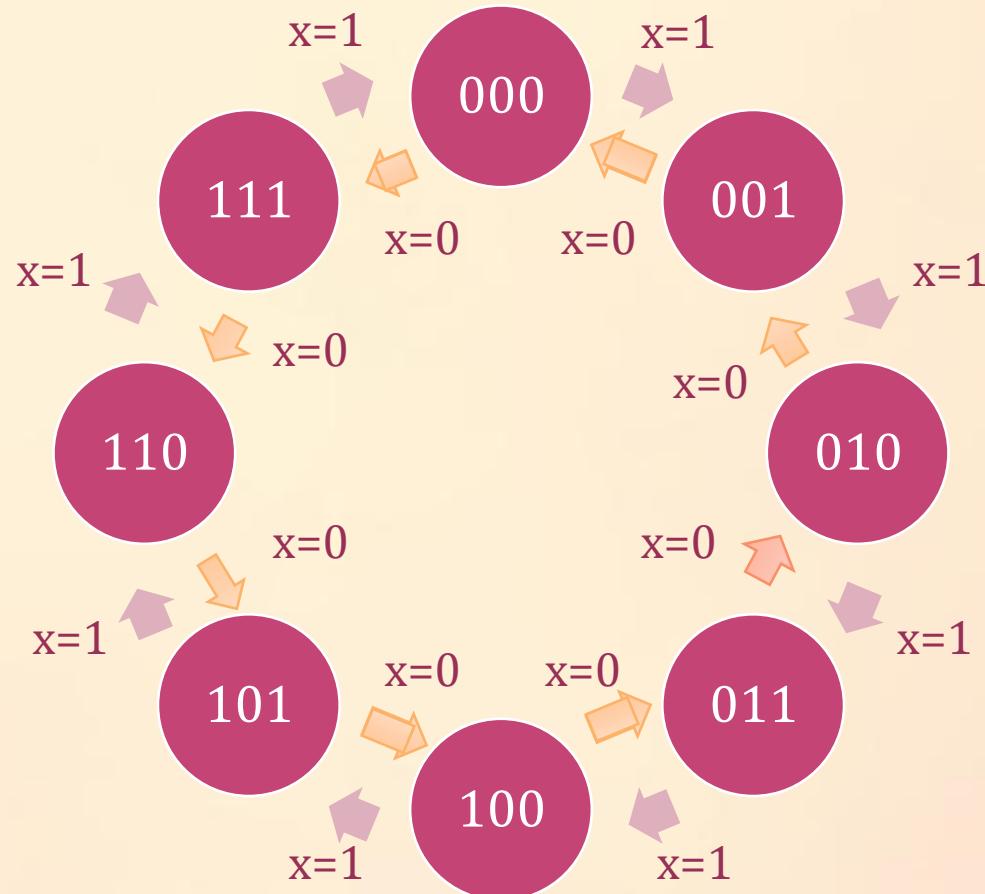
$$T_1 = X'Q'_0 + XQ_0$$

THREE BIT UP/DOWN SYN. COUNTER



- ✓ 3 BIT—3 FF, JK or T Flip-flop, Clock synchronous and -ve edge
- ✓ Counts from 0 to 7 (upwards when $x=1$) or 7 to 0 (downwards when $x=0$)

Step 1: State Diagram



THREE BIT UP/DOWN SYN. COUNTER



Step 2: Next State Table

Previous			Next					
			X=0			X=1		
Q ₂	Q ₁	Q ₀	Q ₂	Q ₁	Q ₀	Q ₂	Q ₁	Q ₀
0	0	0	1	1	1	0	0	1
0	0	1	0	0	0	0	1	0
0	1	0	0	0	1	0	1	1
0	1	1	0	1	0	1	0	0
1	0	0	0	1	1	1	0	1
1	0	1	1	0	0	1	1	0
1	1	0	1	0	1	1	1	1
1	1	1	1	1	0	0	0	0

THREE BIT UP/DOWN SYN. COUNTER



Step 3: Transition/Excitation Table

At X=0

Q_p	Q_n	T
0	0	0
0	1	1
1	0	1
1	1	0

For T_0

Q_{p0}	Q_{n0}	T_0
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1

For T_1

Q_{p1}	Q_{n1}	T_1
0	1	1
0	0	0
1	0	1
1	1	0
0	1	1
0	0	0
1	0	1
1	1	0

For T_2

Q_{p2}	Q_{n2}	T_2
0	1	1
0	0	0
0	0	0
0	0	0
1	0	1
1	1	0
1	1	0
1	1	0

THREE BIT UP/DOWN SYN. COUNTER



Step 3: Transition/Excitation Table

At X=1

Q_p	Q_n	T
0	0	0
0	1	1
1	0	1
1	1	0

For T_0

Q_{p0}	Q_{n0}	T_0
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1

For T_1

Q_{p1}	Q_{n1}	T_1
0	0	0
0	1	1
1	1	0
1	0	1
0	0	0
0	1	1
1	1	0
1	0	1

For T_2

Q_{p2}	Q_{n2}	T_2
0	0	0
0	0	0
0	0	0
0	1	1
1	1	0
1	1	0
1	0	1

THREE BIT UP/DOWN SYN. COUNTER



Step 4: K-Map

For T_0

xQ_2	Q_1Q_0	00	01	11	10
00		1	1	1	1
01		1	1	1	1
11		1	1	1	1
10		1	1	1	1

$T_0 = 1$

For T_1

xQ_2	Q_1Q_0	00	01	11	10
00		1	0	0	1
01		1	0	0	1
11		0	1	1	0
10		0	1	1	0

$$T_1 = X'Q'_1Q'_0 + XQ_1Q_0$$

For T_2

xQ_2	Q_1Q_0	00	01	11	10
00		1	0	0	0
01		1	0	0	0
11		0	0	1	0
10		0	0	1	0

$$T_2 = X'Q'_1Q'_0 + XQ_1Q_0$$

THREE BIT UP/DOWN SYN. COUNTER

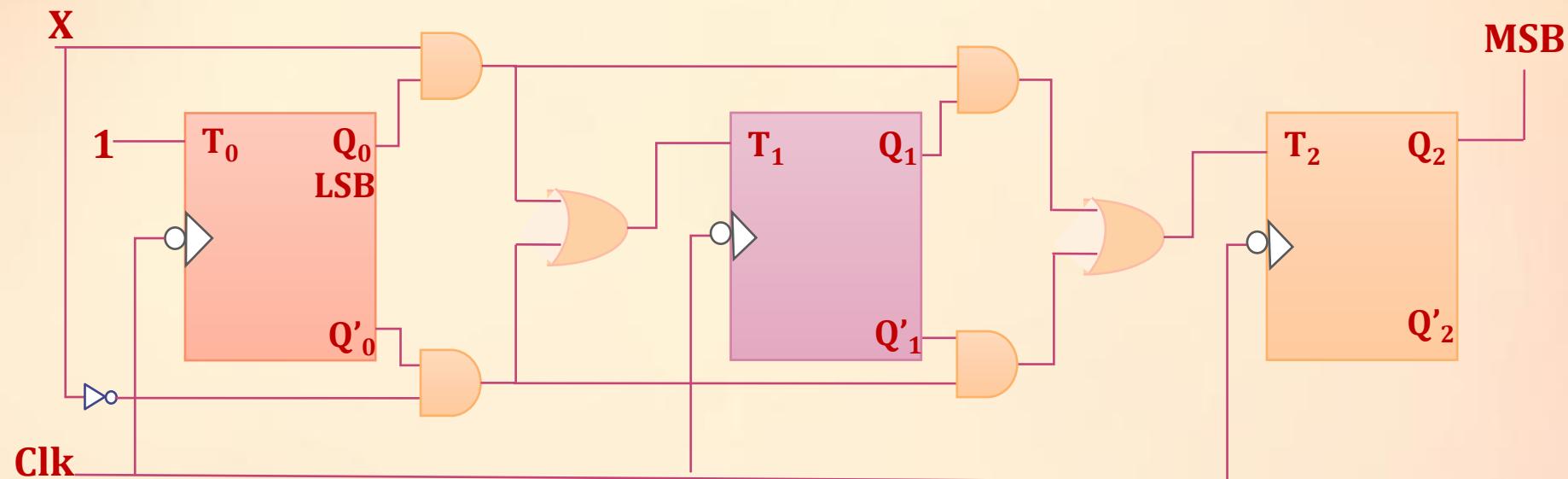


Step 5: Diagram

$$T_0 = 1$$

$$T_1 = X'Q'_0 + XQ_0$$

$$T_2 = X'Q'_1Q'_0 + XQ_1Q_0$$



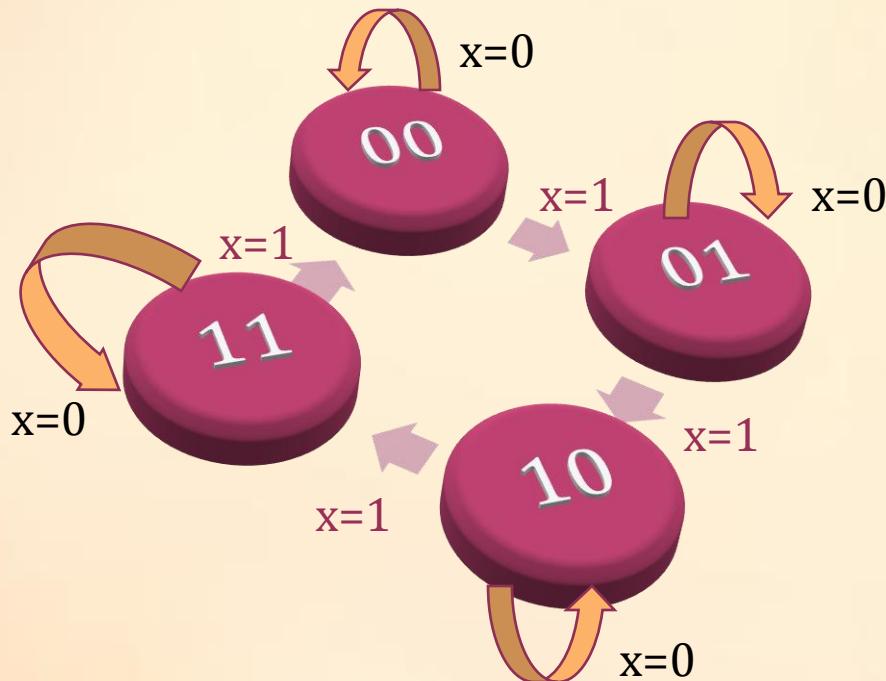
2 BIT SYN. COUNTER WITH CONDITIONS



- ✓ 2 BIT—2 FF, Clock synchronous and -ve edge
- ✓ Counts from 0 to 3 (upwards when $x=1$)
- ✓ Remains in same state (when $x=0$)

Using D FF

Step 1: State Diagram



Step 2: Next State Table

Previous		Next			
		x=0		x=1	
Q_1	Q_0	Q_1	Q_0	Q_1	Q_0
0	0	0	0	0	1
0	1	0	1	1	0
1	0	1	0	1	1
1	1	1	1	0	0

2 BIT SYN. COUNTER WITH CONDITIONS



Step 3: Transition/Excitation Table

Q_p	Q_n	D
0	0	0
0	1	1
1	0	0
1	1	1

At X=0

Q_{p0}	Q_{n0}	D_0
0	0	0
1	1	1
0	0	0
1	1	1

For D_0

Q_{p1}	Q_{n1}	D_1
0	0	0
0	0	0
1	1	1
1	1	1

For D_1

At X=1

Q_{p0}	Q_{n0}	D_0
0	1	1
1	0	0
0	1	1
1	0	0

For D_0

Q_{p1}	Q_{n1}	D_1
0	0	0
0	1	1
1	1	1
1	0	0

For D_1

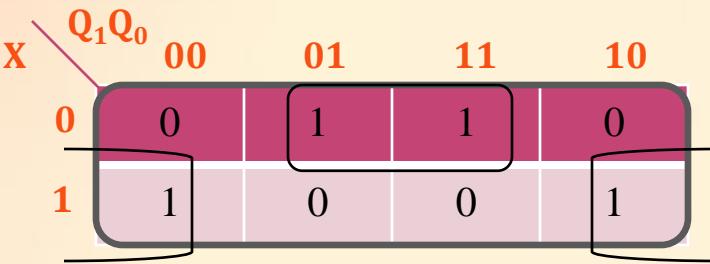
2 BIT SYN. COUNTER WITH CONDITIONS



Step 5: Diagram

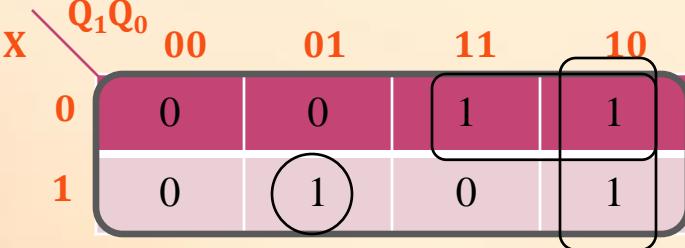
Step 4: K-Map

For D_0



$$D_0 = X'Q_0 + XQ'_0$$

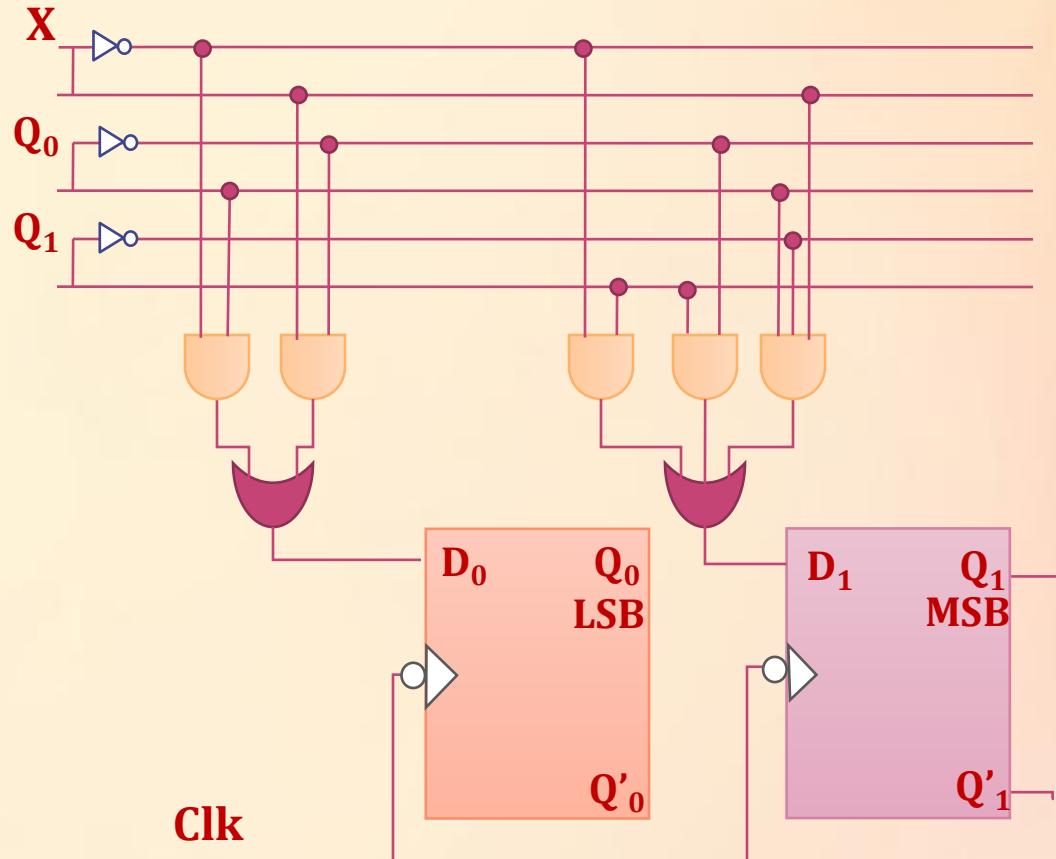
For D_1



$$D_1 = X'Q_1 + Q_1Q'_0 + XQ'_1Q_0$$

$$D_0 = X'Q_0 + XQ'_0$$

$$D_1 = X'Q_1 + Q_1Q'_0 + XQ'_1Q_0$$

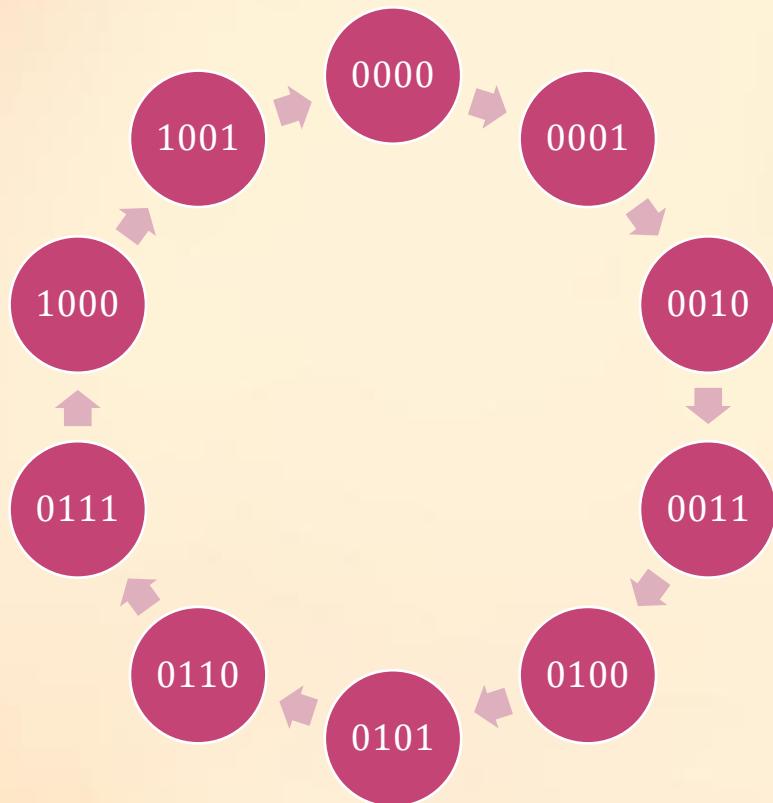


MOD 10 SYNCHRONOUS COUNTER



- ✓ Counts from 0 to 9 (1001)
- ✓ 4 BIT—4 FF, JK or T Flip-flop, Clock synchronous and –ve edge

Step 1: State Diagram



Step 2: Next State Table

Previous				Next			
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0

MOD 10 SYN. COUNTER



Step 3: Transition/Excitation Table

Q_p	Q_n	T
0	0	0
0	1	1
1	0	1
1	1	0

For T_0

Q_{p0}	Q_{n0}	T_0
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1
0	1	1
1	0	1

For T_1

Q_{p1}	Q_{n1}	T_1
0	0	0
0	1	1
1	1	0
1	0	1
0	0	0
0	1	1
1	1	0
1	0	1
0	0	0
0	0	0

For T_2

Q_{p2}	Q_{n2}	T_2
0	0	0
0	0	0
0	0	0
0	1	1
1	1	0
1	1	0
1	1	0
1	0	1
0	0	0
0	0	0

For T_3

Q_{p3}	Q_{n3}	T_3
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	1	1
1	1	0
1	0	1
0	0	0
1	0	1

MOD 10 SYN. COUNTER



Step 4: K-Map

For T_0

		$Q_1 Q_0$	00	01	11	10
		$Q_3 Q_2$	00	01	11	10
$Q_3 Q_2$	$Q_1 Q_0$	00	1	1	1	1
00	01	00	1	1	1	1
01	11	01	x	x	x	x
11	10	11	1	0	x	x
10	10	10	x	x	x	x

$T_0 = 1$

For T_1

		$Q_1 Q_0$	00	01	11	10
		$Q_3 Q_2$	00	01	11	10
$Q_3 Q_2$	$Q_1 Q_0$	00	0	1	1	0
00	01	01	0	1	1	0
01	11	11	x	x	x	x
11	10	10	0	0	x	x
10	10	10	0	0	x	x

$$T_1 = Q'_3 Q_0$$

For T_2

		$Q_1 Q_0$	00	01	11	10
		$Q_3 Q_2$	00	01	11	10
$Q_3 Q_2$	$Q_1 Q_0$	00	0	0	1	0
00	01	01	0	0	1	0
01	11	11	x	x	x	x
11	10	10	0	0	x	x
10	10	10	0	0	x	x

$T_2 = Q_1 Q_0$

For T_3

		$Q_1 Q_0$	00	01	11	10
		$Q_3 Q_2$	00	01	11	10
$Q_3 Q_2$	$Q_1 Q_0$	00	0	0	0	0
00	01	01	0	0	1	0
01	11	11	x	x	x	x
11	10	10	x	x	x	x
10	10	10	0	1	x	x

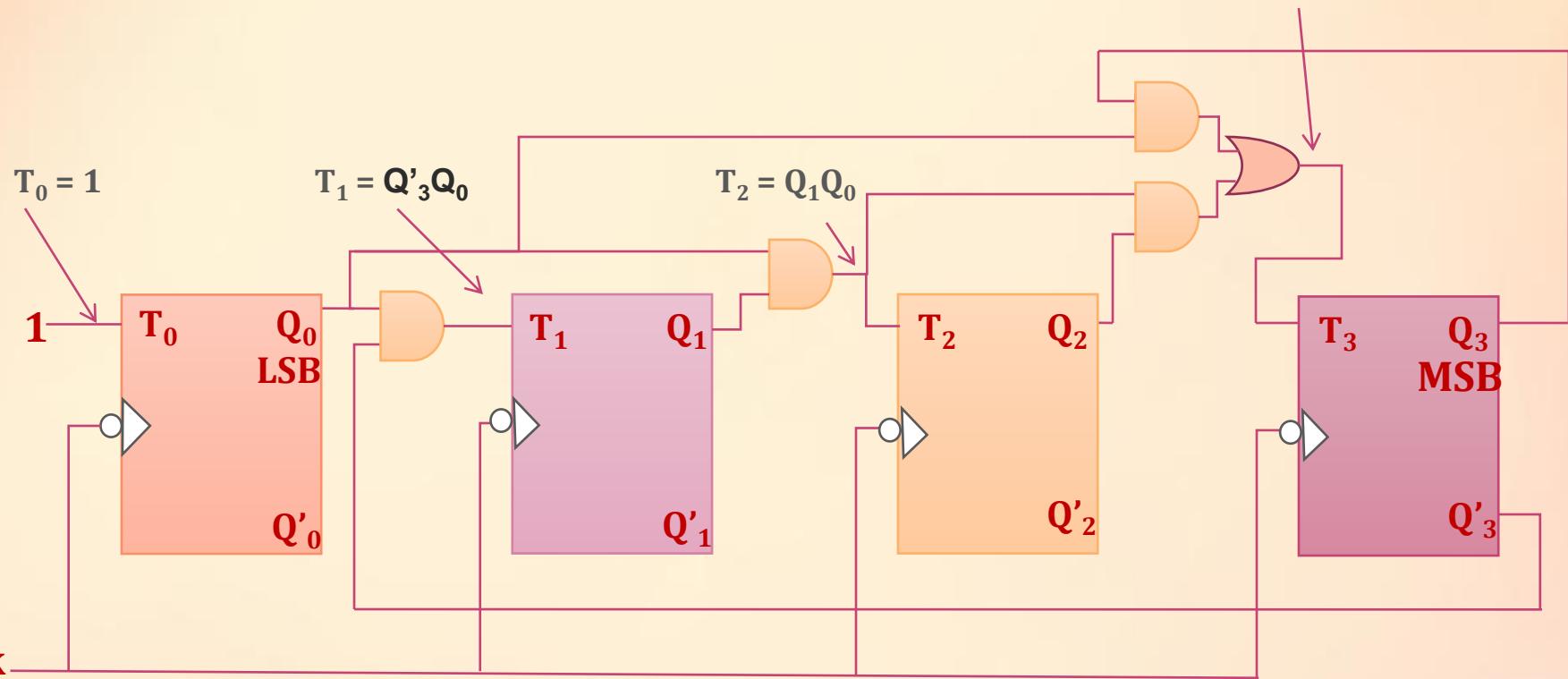
$T_3 = Q_2 Q_1 Q_0 + Q_3 Q_0$

MOD 10 SYN. COUNTER



Step 5: Diagram

$$T_3 = Q_2 Q_1 Q_0 + Q_3 Q_0$$

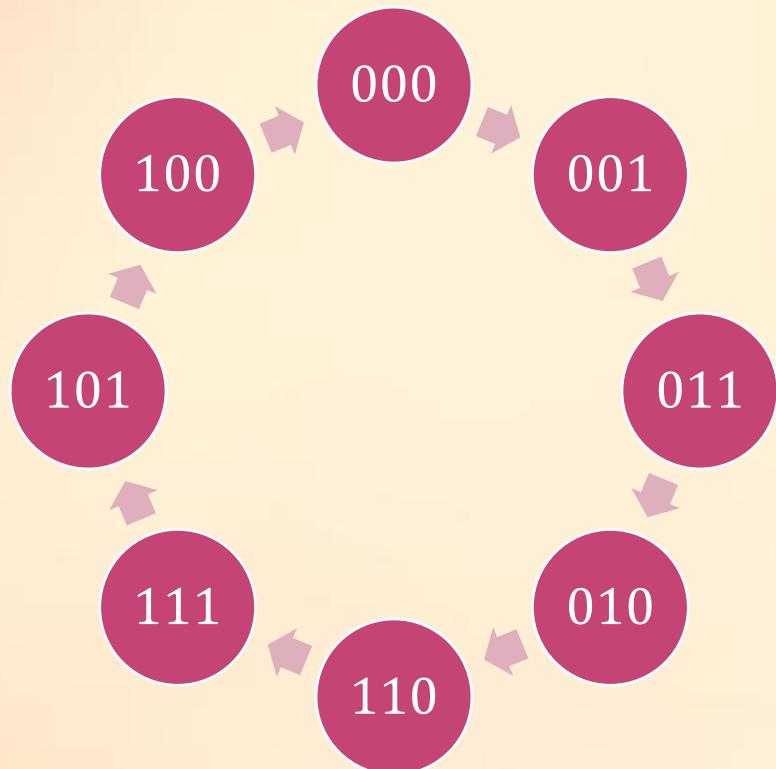


THREE BIT GRAY SYN. COUNTER



- ✓ 3 BIT—3 FF, Using JK Flip-flop, Clock synchronous and -ve edge
- ✓ Counts 0,1,3,2,6,7,5,4

Step 1: State Diagram



Step 2: Next State Table

Previous			Next		
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

THREE BIT GRAY SYN. COUNTER



Step 3: Transition/Excitation Table

Q_p	Q_n	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

For J_0, K_0

Q_{p0}	Q_{n0}	J_0	K_0
0	1	1	x
1	1	x	0
1	0	x	1
0	0	0	x
0	1	1	x
1	1	x	0
1	0	x	1
0	0	0	x

For J_1, K_1

Q_{p1}	Q_{n1}	J_1	K_1
0	0	0	x
0	1	1	x
1	1	x	0
1	1	x	0
1	1	x	0
1	0	x	1
0	0	0	x
0	0	0	x

For J_2, K_2

Q_{p2}	Q_{n2}	J_2	K_2
0	0	0	x
0	0	0	x
0	0	0	x
0	1	1	x
1	1	x	0
1	1	x	0
1	1	x	0
1	0	x	1

Step 4: K-Map



For J_0

		$Q_1 Q_0$	00	01	11	10
		0	1	x	x	0
		1	0	x	x	1
			$J_0 = Q'_2 Q'_1 + Q_2 Q_1$			
			$J_0 = Q_2 \text{ xnor } Q_1$			

For K_0

		$Q_1 Q_0$	00	01	11	10
		0	x	0	1	x
		1	x	1	0	x
			$K_0 = Q'_2 Q_1 + Q_2 Q'_1$			
			$K_0 = Q_2 \text{ xor } Q_1$			

For J_1

		$Q_1 Q_0$	00	01	11	10
		0	0	1	x	x
		1	0	0	x	x
			$J_1 = Q'_2 Q_0$			

For K_1

		$Q_1 Q_0$	00	01	11	10
		0	x	x	0	0
		1	x	x	1	0
			$K_1 = Q_2 Q_0$			

For J_2

		$Q_1 Q_0$	00	01	11	10
		0	0	0	0	1
		1	x	x	x	x
			$J_2 = Q_1 Q'_0$			

For K_2

		$Q_1 Q_0$	00	01	11	10
		0	x	x	x	x
		1	1	0	0	0
			$K_2 = Q'_1 Q'_0$			

THREE BIT GRAY SYN. COUNTER



Step 5: Diagram

$$J_0 = Q'_2 Q'_1 + Q_2 Q_1$$

$$J_0 = Q_2 \text{ xnor } Q_1$$

$$K_0 = Q'_2 Q_1 + Q_2 Q'_1$$

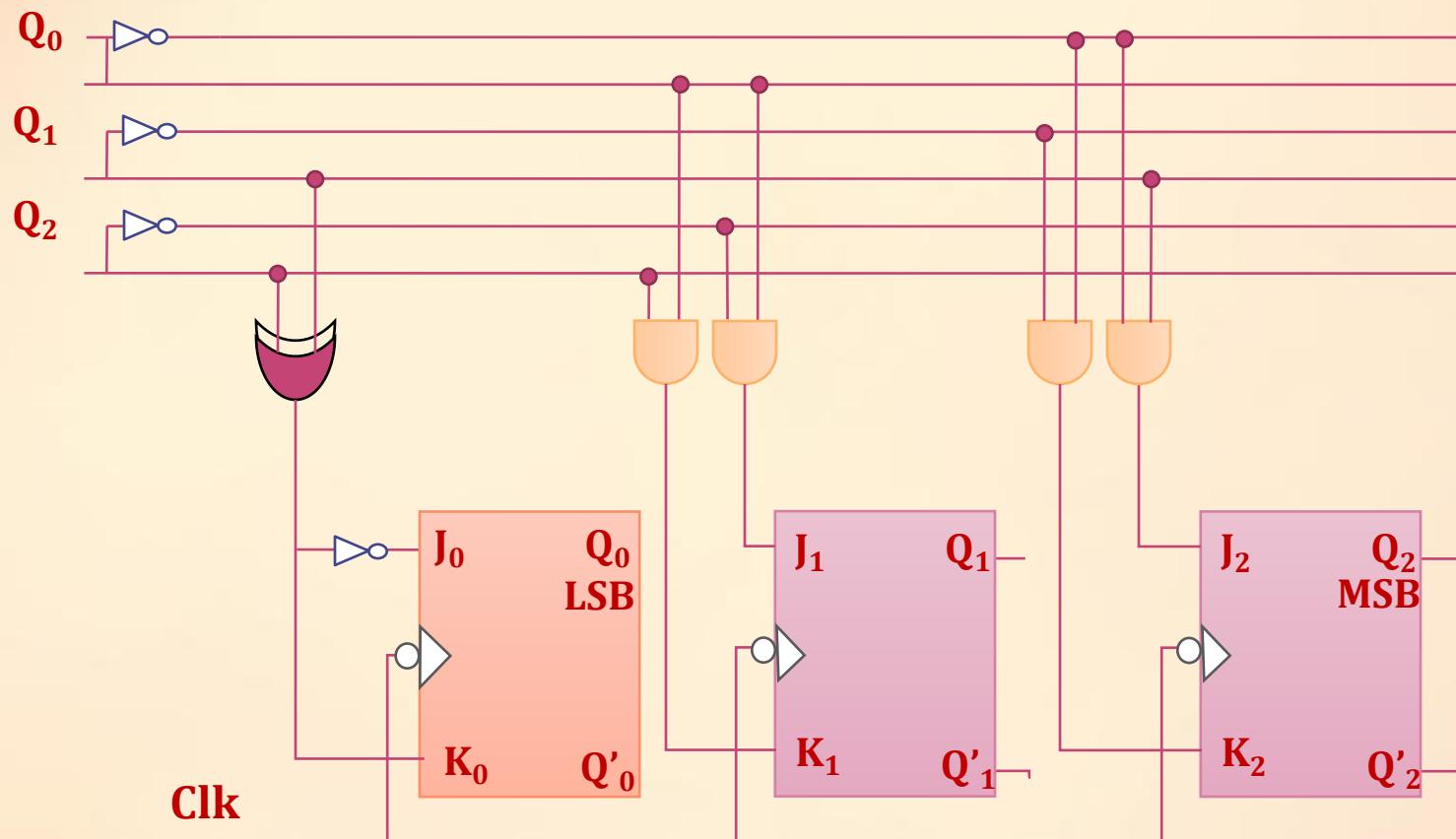
$$K_0 = Q_2 \text{ xor } Q_1$$

$$J_1 = Q'_2 Q_0$$

$$K_1 = Q_2 Q_0$$

$$J_2 = Q_1 Q'_0$$

$$K_2 = Q'_1 Q'_0$$



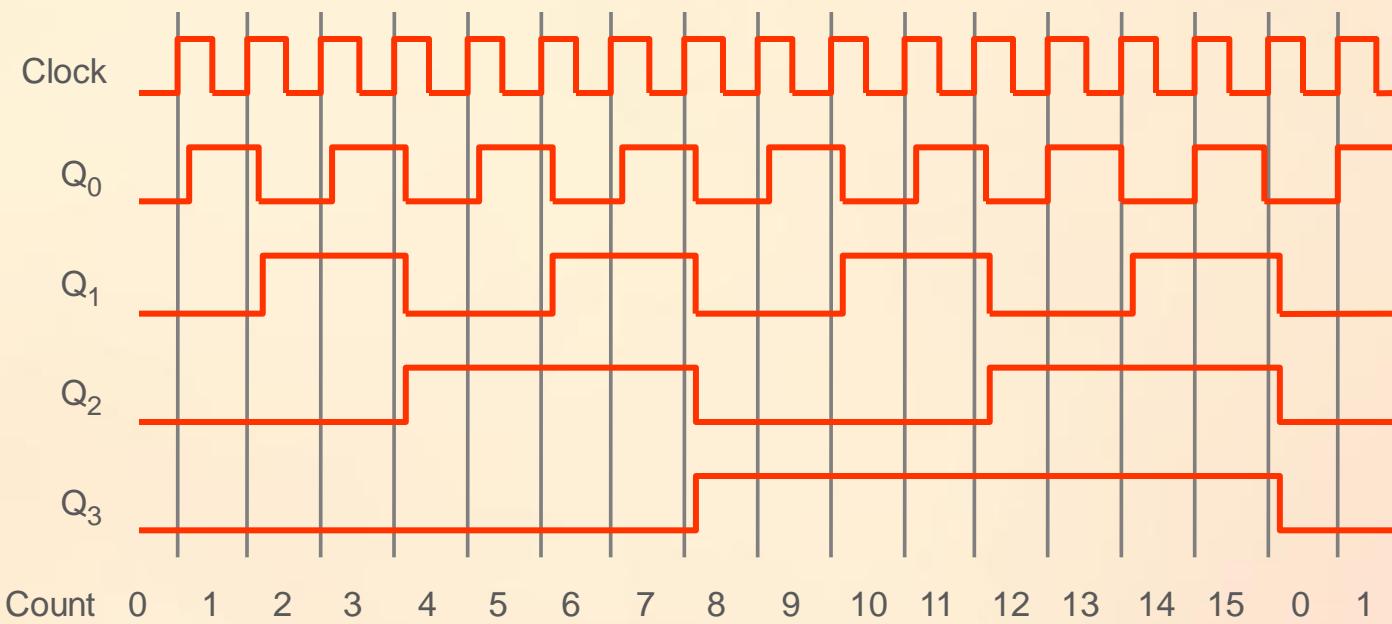
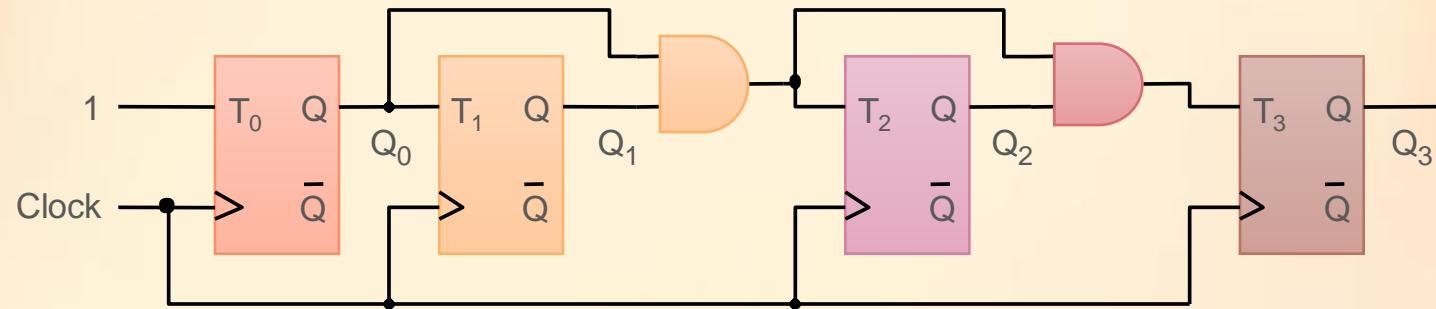
N BIT UP SYN. COUNTER –DIRECT WAY



Clock cycle	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1

$T_0 \text{ or } J_0K_0 = 1$	Always Toggle
$T_1 \text{ or } J_1K_1 = Q_0$	Toggle when $Q_0 = 1$
$T_2 \text{ or } J_2K_2 = Q_1Q_0$	Toggle when $Q_1Q_0 = 1$
$T_3 \text{ or } J_3K_3 = Q_2Q_1Q_0$	Toggle when $Q_2Q_1Q_0 = 1$
...	

4 BIT UP SYN. COUNTER –DIRECT WAY



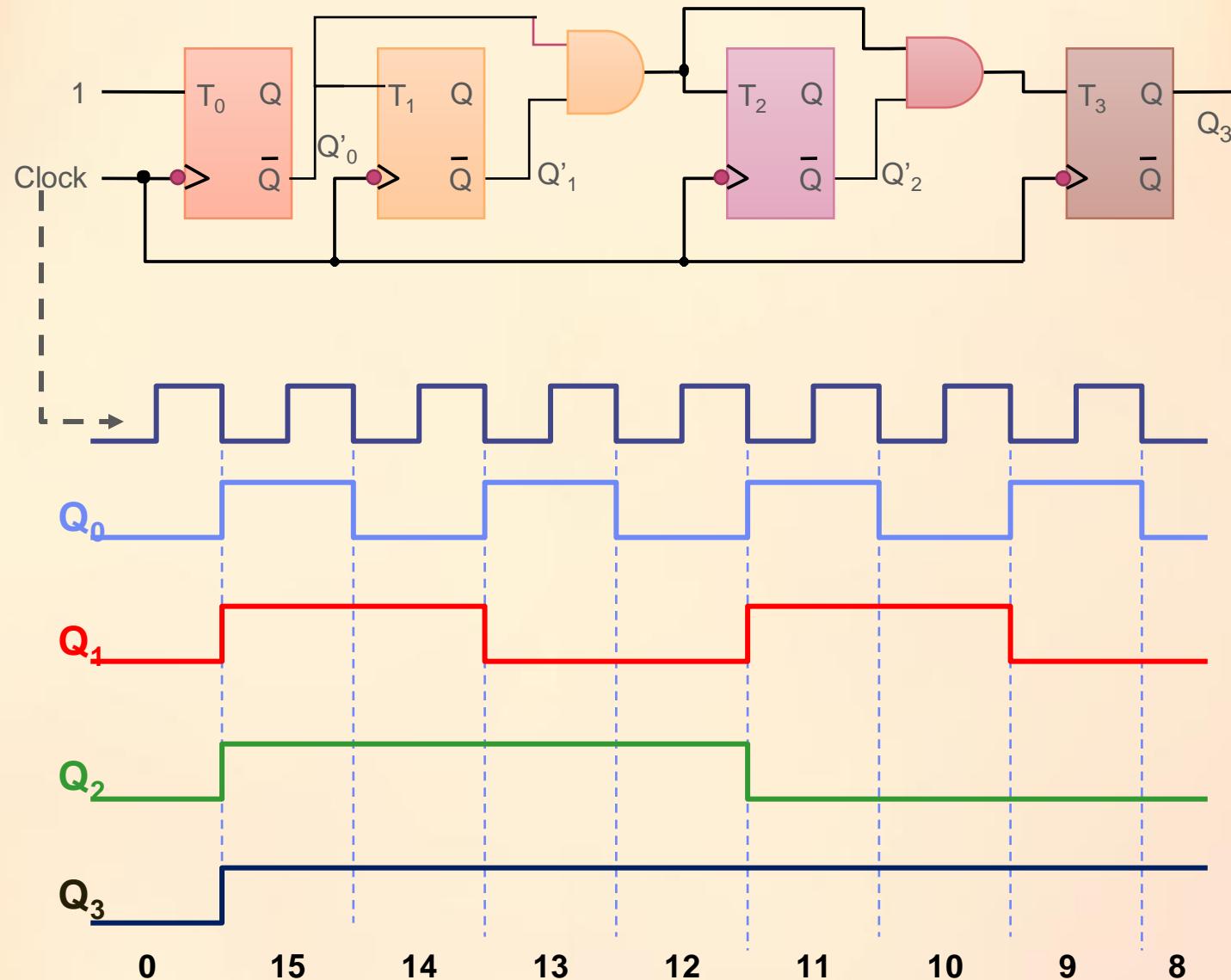
N BIT DOWN SYN. COUNTER –DIRECT WAY



Clock cycle	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	1	1	1	1
2	1	1	1	0
3	1	1	0	1
4	1	0	0	0
5	1	0	1	1
6	1	0	1	0
7	1	0	0	1
8	0	0	0	0
9	0	1	1	1
10	0	1	1	0
11	0	1	0	1

$T_0 \text{ or } J_0K_0 = 1$	Always Toggle
$T_1 \text{ or } J_1K_1 = Q'_0$	Toggle when $Q_0 = 0$ $Q'_0 = 1$
$T_2 \text{ or } J_2K_2 = Q'_1Q'_0$	Toggle when $Q_1Q_0 = 0$ $Q'_1Q'_0 = 1$
$T_3 \text{ or } J_3K_3 = Q'_2Q'_1Q'_0$	Toggle when $Q_2Q_1Q_0 = 0$ $Q'_2Q'_1Q'_0 = 1$
...	

4 BIT DOWN SYN. COUNTER –DIRECT WAY

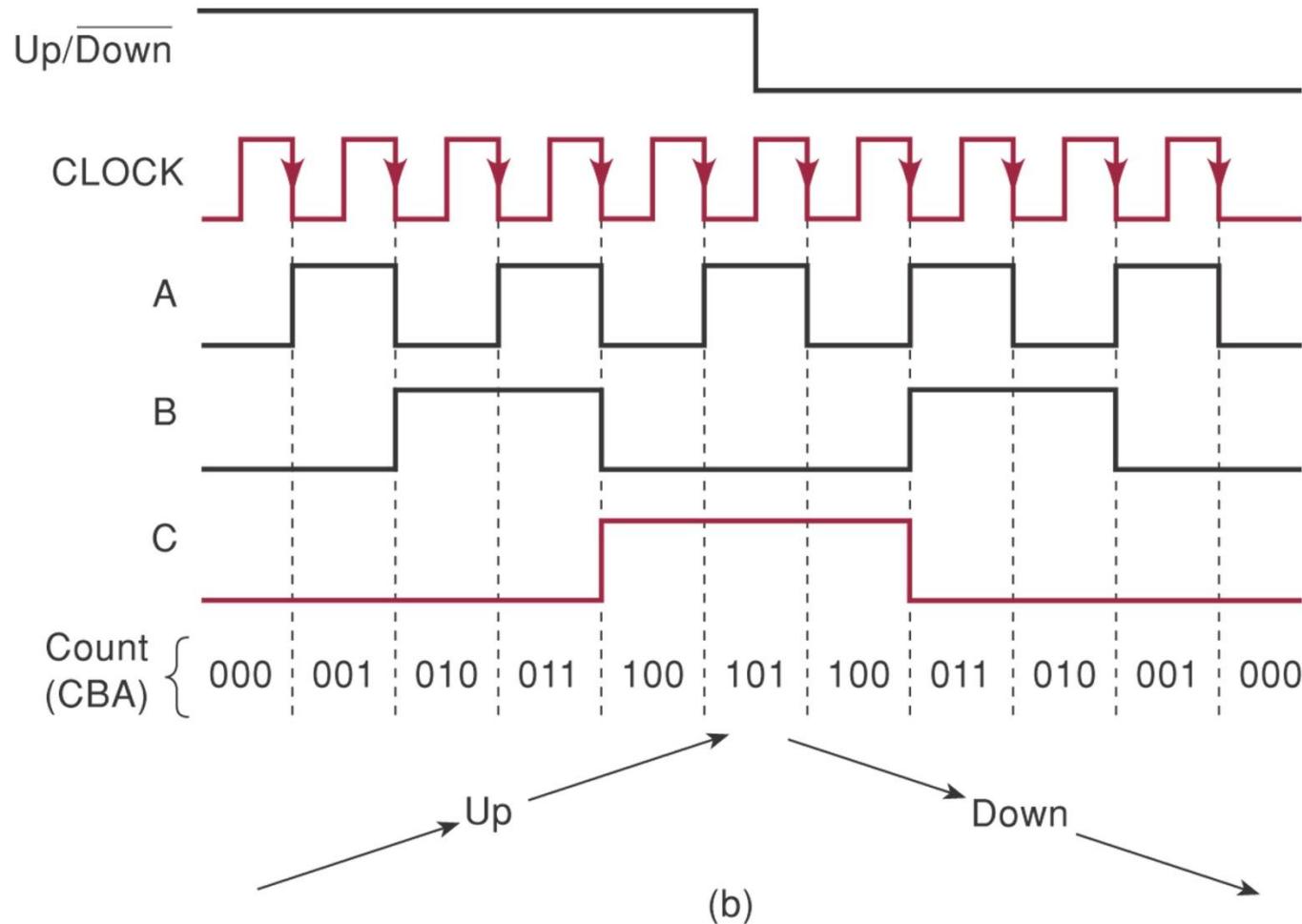


N BIT UP/DOWN SYN. COUNTER –DIRECT



Clock cycle	Up/(down)'	Q ₂	Q ₁	Q ₀		
0	x	0	0	0		
1	1	0	0	1	UP	DOWN
2	1	0	1	0	T ₀ or J ₀ K ₀ = 1	T ₀ or J ₀ K ₀ = 1
3	1	0	1	1	T ₁ or J ₁ K ₁ = Q ₀	T ₁ or J ₁ K ₁ = Q' ₀
4	1	1	0	0	T ₂ or J ₂ K ₂ = Q ₁ Q ₀	T ₂ or J ₂ K ₂ = Q' ₁ Q' ₀
5	1	1	0	1	T ₃ or J ₃ K ₃ = Q ₂ Q ₁ Q ₀	T ₃ or J ₃ K ₃ = Q' ₂ Q' ₁ Q' ₀
6	0	1	0	0
7	0	0	1	1		
8	0	0	1	0		
9	0	0	0	1		
10	0	0	0	0		

3 BIT UP/DOWN SYN. COUNTER –DIRECT



The counter counts up when the control input Up/Down = 1; it counts down when the control input Up/Down = 0.

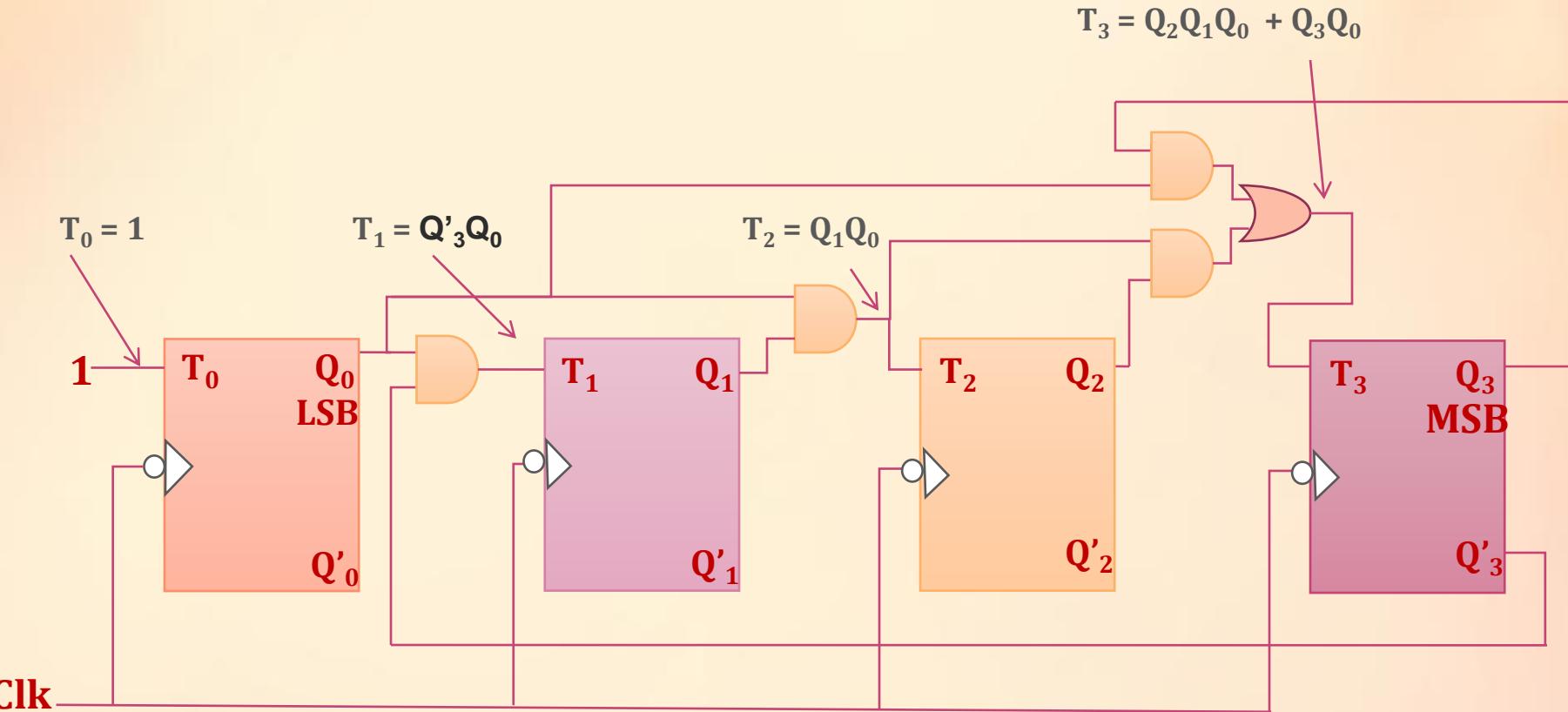
MOD 10 SYN. COUNTER –DIRECT WAY



Clock cycle	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	0	0	0	0
11	0	0	0	1

$T_0 \text{ or } J_0K_0 = 1$	Always Toggle
$T_1 \text{ or } J_1K_1 = Q_0Q'_3$	Toggle when $Q_0 = 1 \text{ & } Q_3 = 0$ (i.e. $Q'_3 = 1$)
$T_2 \text{ or } J_2K_2 = Q_1Q_0$	Toggle when $Q_1Q_0 = 1$
$T_3 \text{ or } J_3K_3 = Q_2Q_1Q_0 + Q_0Q_3$	Toggle when $Q_2Q_1Q_0 = 1$ or $Q_0Q_3 = 1$

MOD 10 SYN. COUNTER –DIRECT WAY

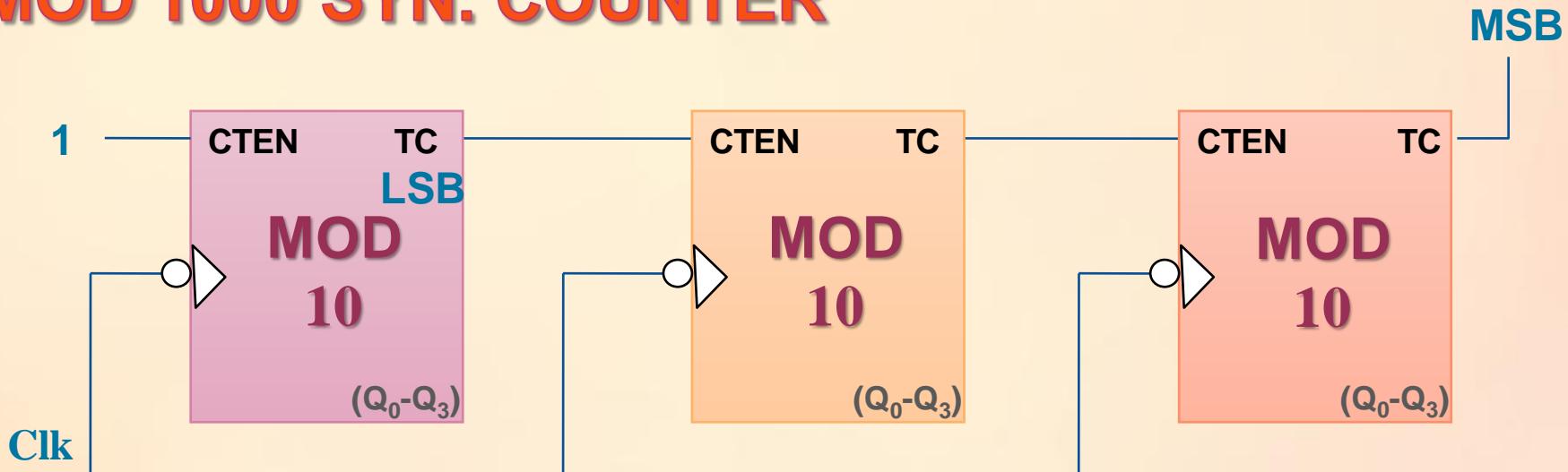


CASCADE SYNCHRONOUS COUNTER



- ✓ Counters are cascaded or connected together in synchronous way to achieve higher modulus operation
- ✓ The overall modulus of a Cascaded counter = Product of individual modulus of a counter
- ✓ All the individual counters are triggered by the clock simultaneously
- ✓ Two inputs CTEN----Counter enable and TC-----Terminal count are used. TC of one counter triggers CTEN of next counter

MOD 1000 SYN. COUNTER



APPLICATION OF COUNTERS



- Wide variety of applications in digital logic, computer circuits, microcontrollers, timers and others
 - ✓ Count the number of times an event takes place (in industries, no. of telephone calls)
 - ✓ In computers, counting the no. of instructions while executing a program and in Microprocessors
 - ✓ Frequency conversion or division of clock frequency
 - ✓ Speed measurement
 - ✓ Frequency and time measurement
 - ✓ Control the number of steps in a sequence of fixed actions (a sequencer)
 - ✓ Signal Generators used for waveform generation
 - ✓ Automatic parking control
 - ✓ Digital clock
 - ✓ Parallel-to-Serial Data Conversion
 - ✓ A/D Converters



: 05

