CS/ECE/EEE/INSTR F215:Digital Design

Lecture 29: *Counters*

Thu, 18 Nov 2021

BITS Pilani
Hyderabad Campus

Dr. R. N. Ponnalagu, EEE

Desire for success > Fear for Failure

is the formula for success

Counters

- A sequential circuit that goes through prescribed sequence of states upon application of input pulses.
- Input pulses may be clock pulses.
- Sequence of states may follow binary sequence or any other sequence of states.
- A counter that follows binary sequence – Binary Counter
- An n-bit binary counter has n-FFs and can count in binary from 0 through 2^n -1.
- Types of counters: Ripple or Asynchronous counters and Synchronous counters

Ring Counter, Johnson Counter – Other counters

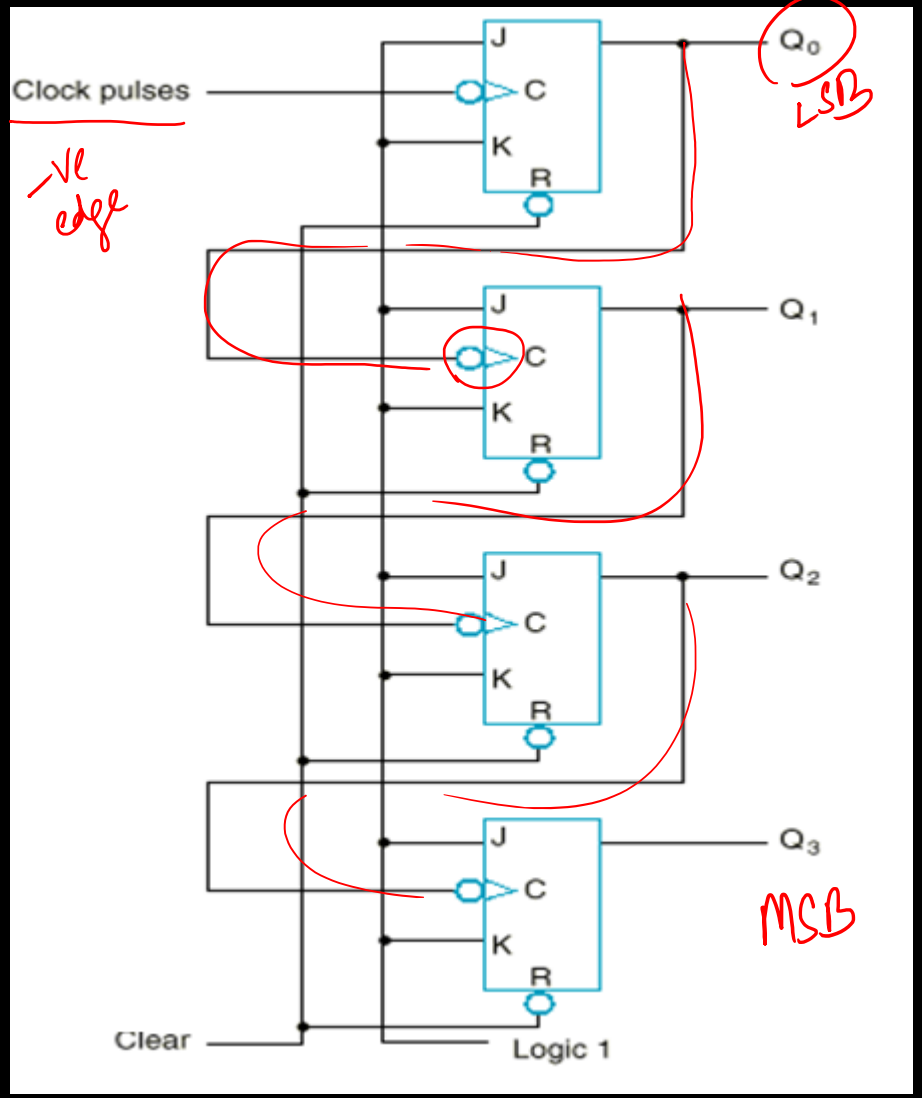
Ripple Counter – Asynchronous Counters

The FF output transition serves as a source for triggering other FFs. No common clock.

Synchronous Counters

All FFs receive the common clock pulse, and the change of state is determined from the present state.

Ripple Counter – Asynchronous Counters



In Asynchronous counter, ext. clock pulse is applied only to the flip flop (LSB).
Instead of the ext. clock pulse, the output of first flip-flop acts as a clock pulse to the next flip flop, whose output is used as a clock to the next in line flip-flop and so on.

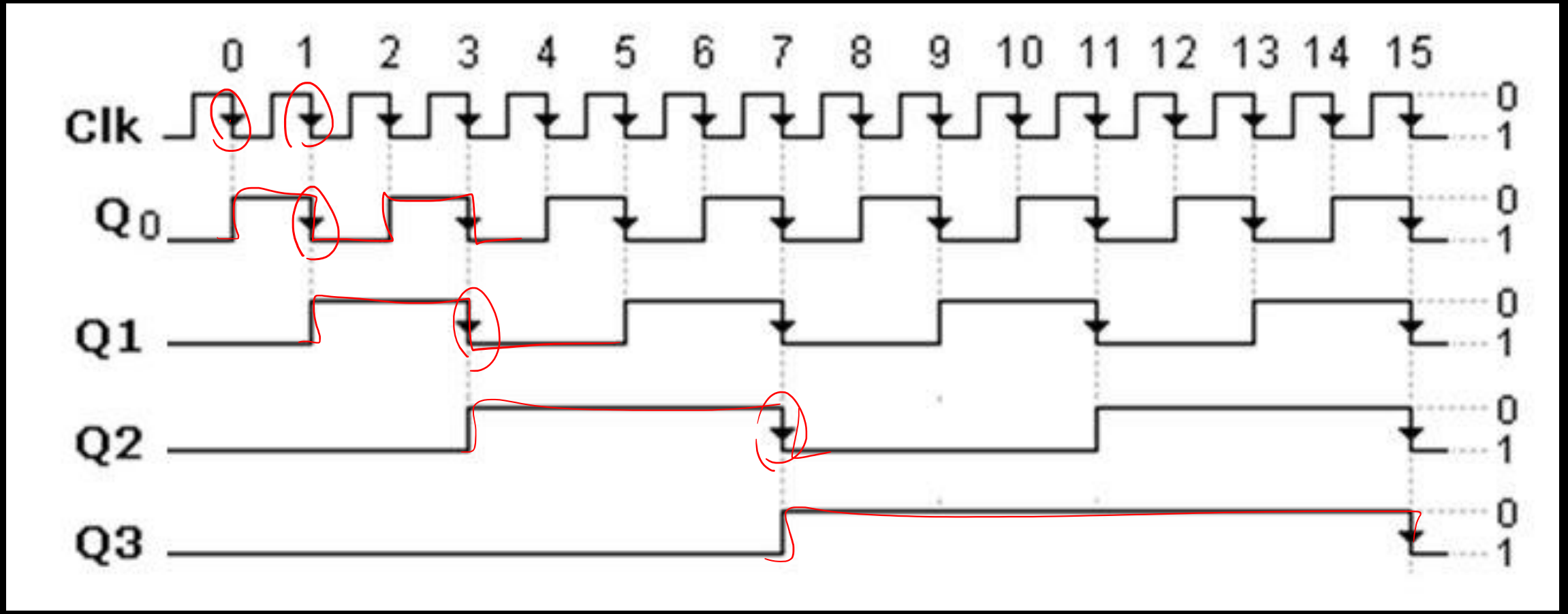
- The output of each FF is connected to the Clock input of the next FF in sequence.
- The FF holding the least significant bit receives the incoming clock pulses.
- The J and K inputs of all FFs are connected to a permanent logic 1.

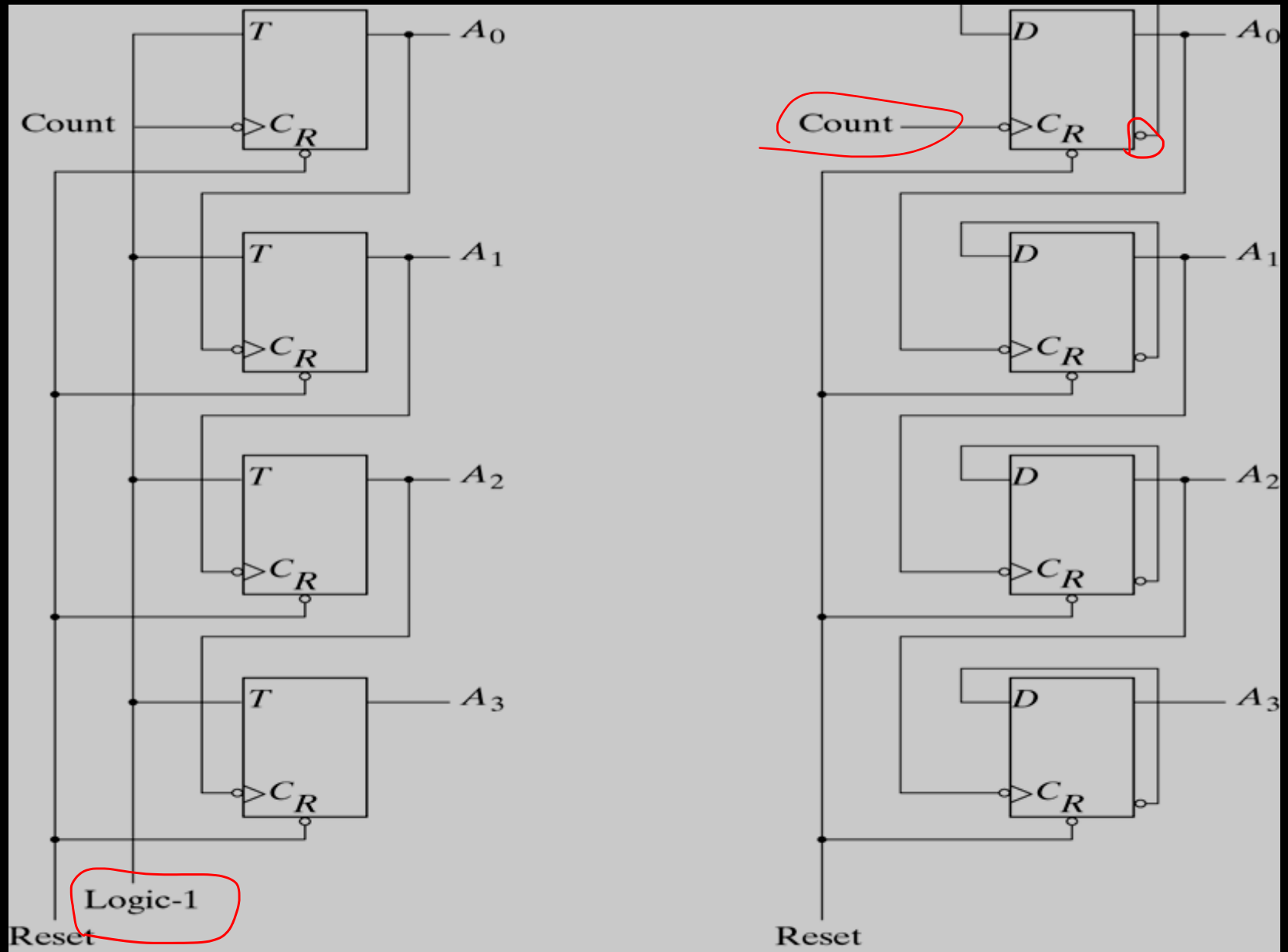
- Operation:
- The least significant bit (Q_0) is complemented with each negative-edge clock pulse input.
- Every time that Q_0 goes from 1 to 0, Q_1 is complemented.
- Every time that Q_1 goes from 1 to 0, Q_2 is complemented.
- Every time that Q_2 goes from 1 to 0, Q_3 is complemented, and so on.

Upward Counting Sequence

Q_3	Q_2	Q_1	Q_0
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

Timing diagram





4-Bit Binary Ripple Counter

Design a Binary Down Counter

Approach:

Use direct Set (S) signals (PRESET) instead of direct Reset (R), in order to start at 1111.

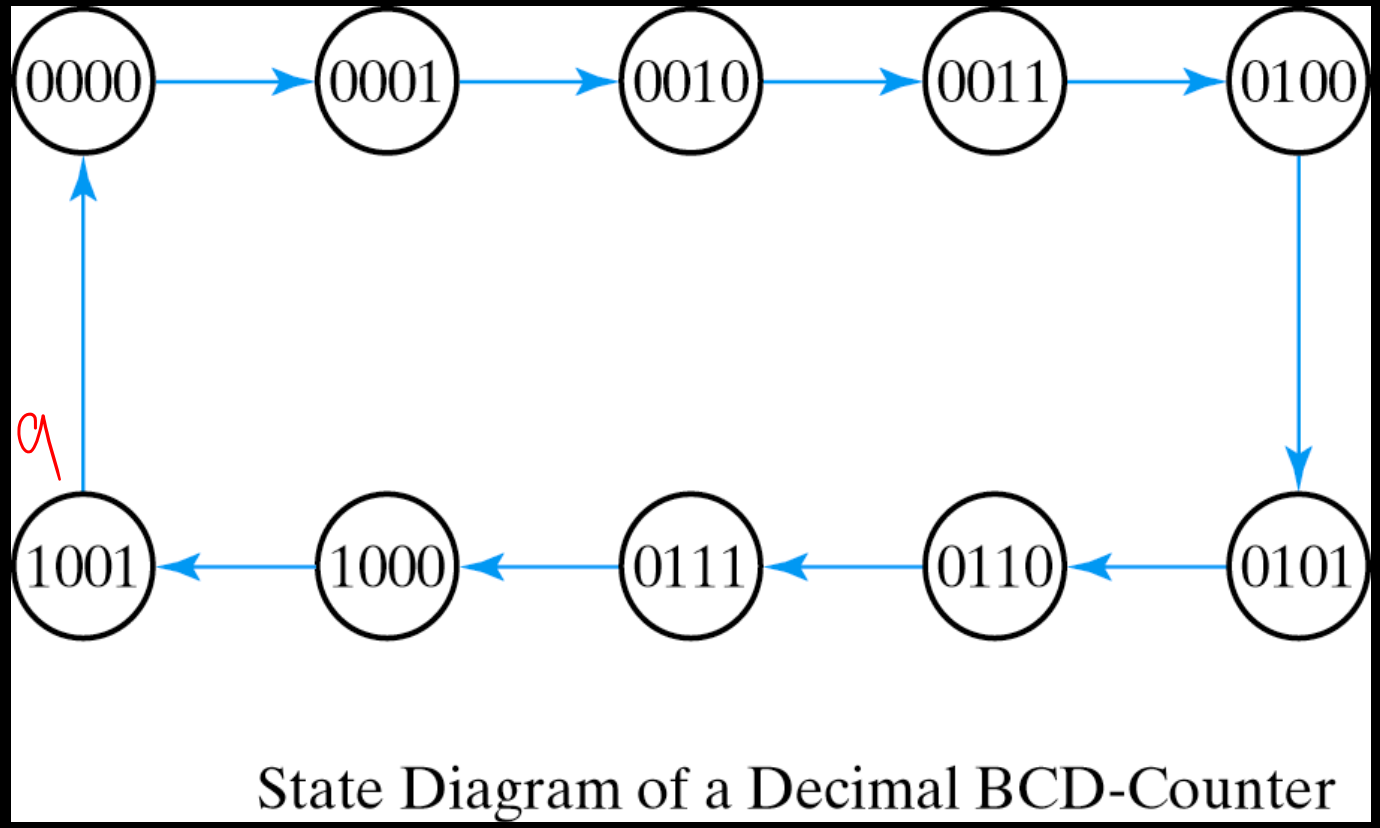
- Method 1:Change edge-triggering to positive
- Method 2: Connect the complement output of each FF to the C (CLOCK) input of the next FF in the sequence

decade counter

10

0

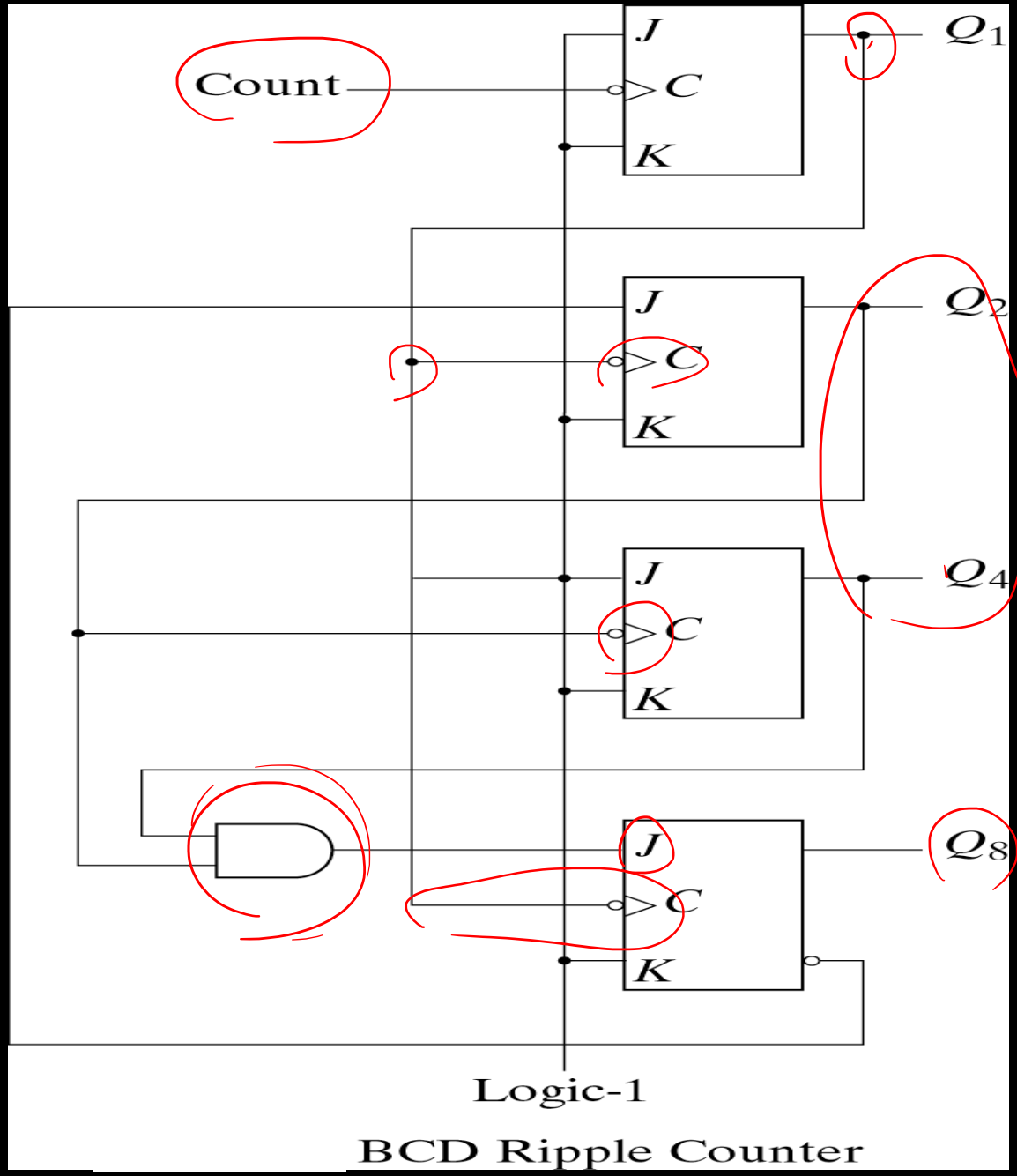
9



State Diagram of a Decimal BCD-Counter

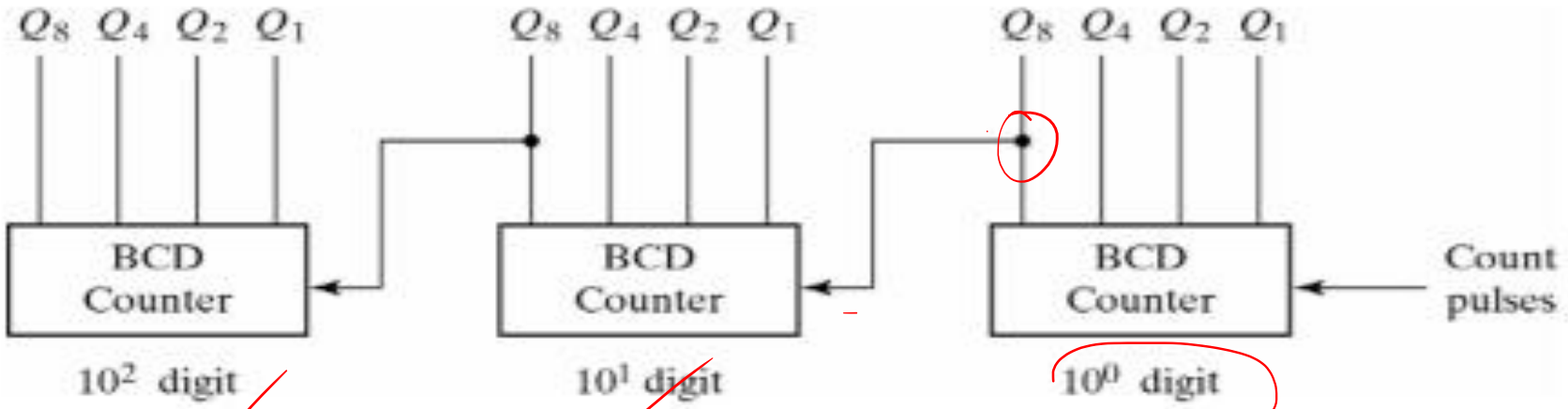
- Q1 complemented on every clock edge
- Q2 complemented when Q1 goes from 1 to 0. as long as Q8=0, when Q8= 1, Q2 remains at 0
- Q4 complemented when Q2 goes from 1 to 0
- Q8 cleared (remains at zero) as long as Q4 or Q2 is 0 and Q8 complemented when Q4Q2 = 11 and Q1 goes from 1 to 0.

Q8	Q4	Q2	Q1
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
0	0	0	0



BCD Ripple Counter

A multiple decade counter can be constructed by connecting BCD counters in cascade. A three-*decade* counter is shown below:



The inputs to the second and third decades come from Q_8 of the previous decade. When Q_8 in one decade goes from 1 to 0, it triggers the count for the next higher-order decade while its own goes from 9 to 0.

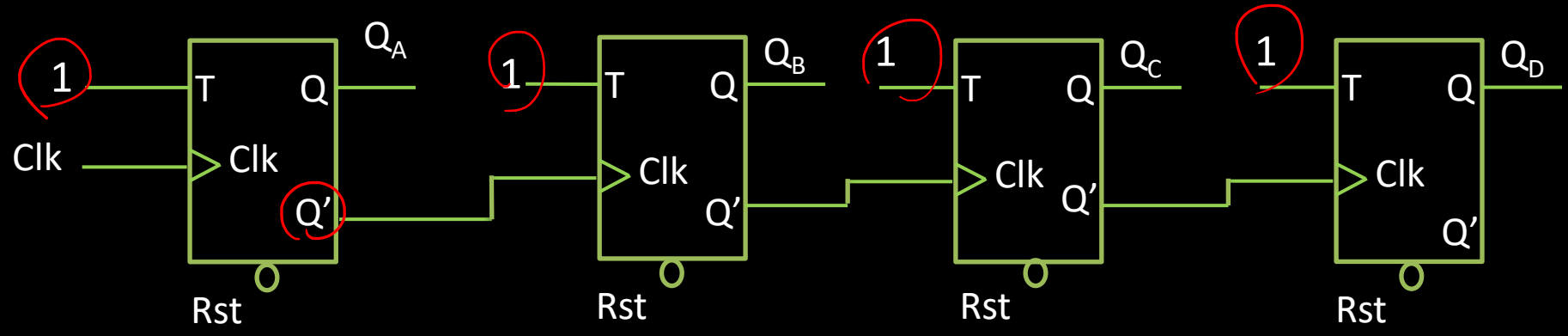
Modulo-n counter

A counter that goes through a repeated sequence of ‘n’ states.
Ex. Mod-3 counter will go through 3 states 00 to 01 to 10 to 00

divide by N counter

Mod Ripple counter

Can you design mod 10 counter counts till 9



Counts from 0000→1111

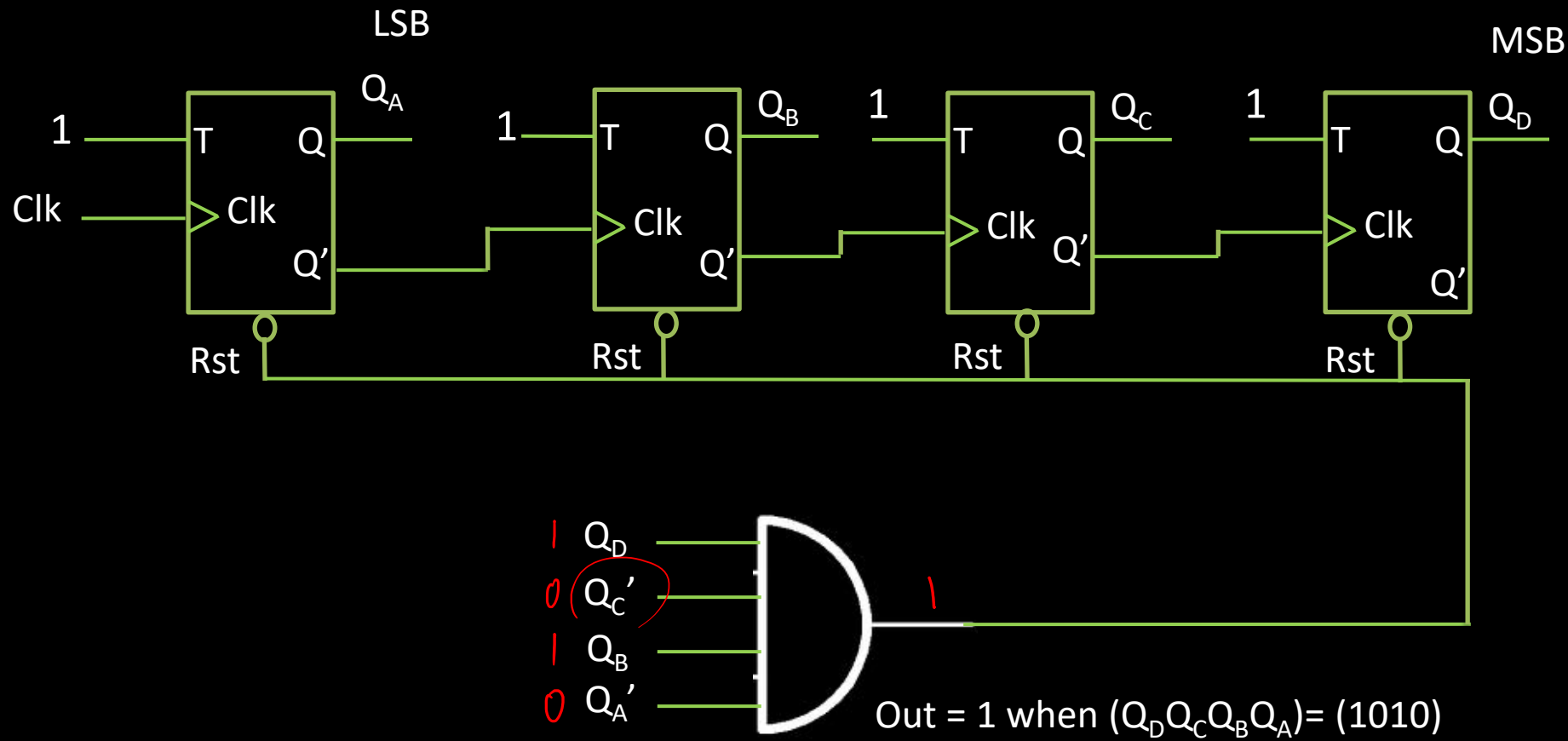
As soon as 1010 reached all flip flops should be reset

Ripple Counters

mod-12
0 to 11
1100
0000

Mod Ripple counter

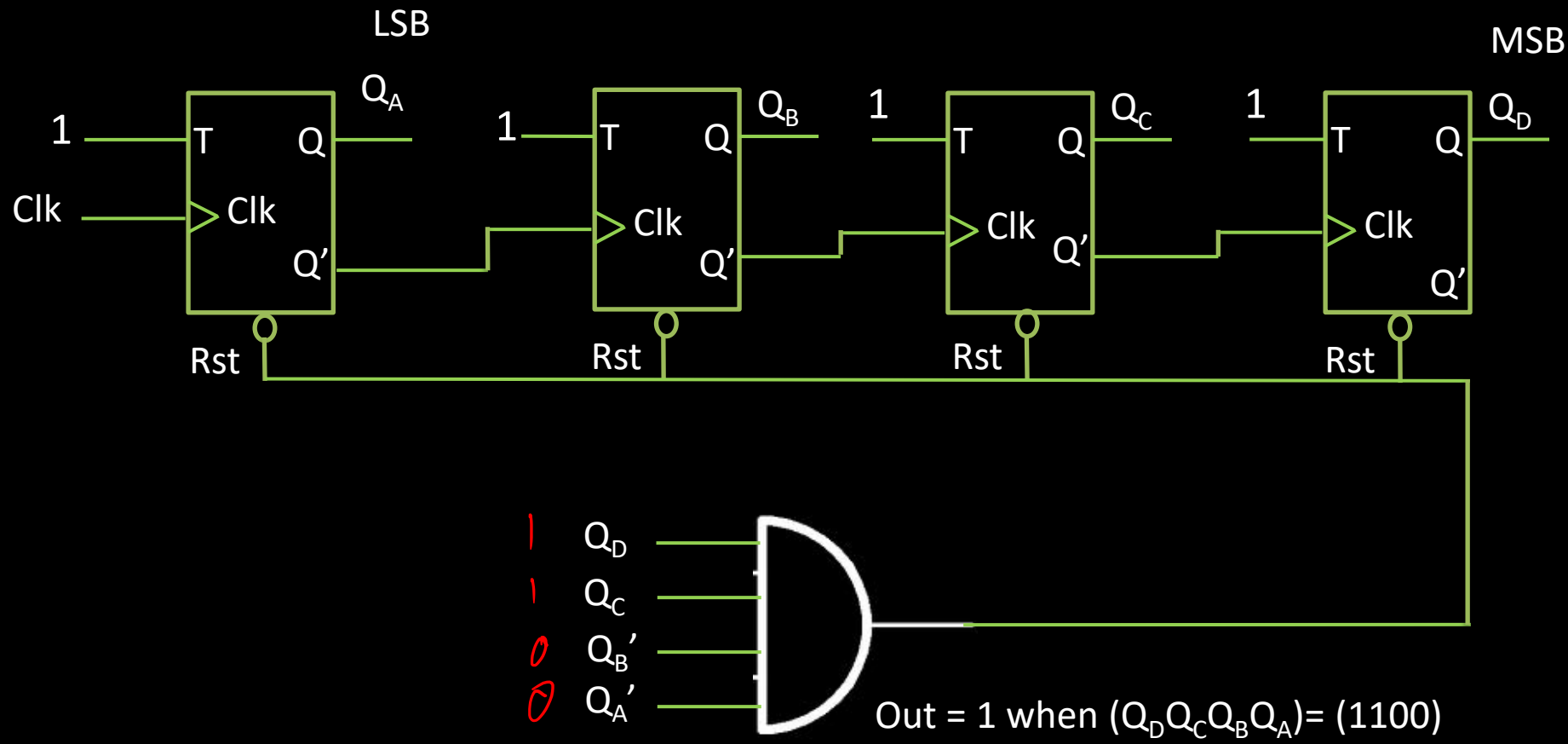
Mod 10 counter counts till 9



Ripple Counters

Mod Ripple counter

Can you design Mod 12 counter



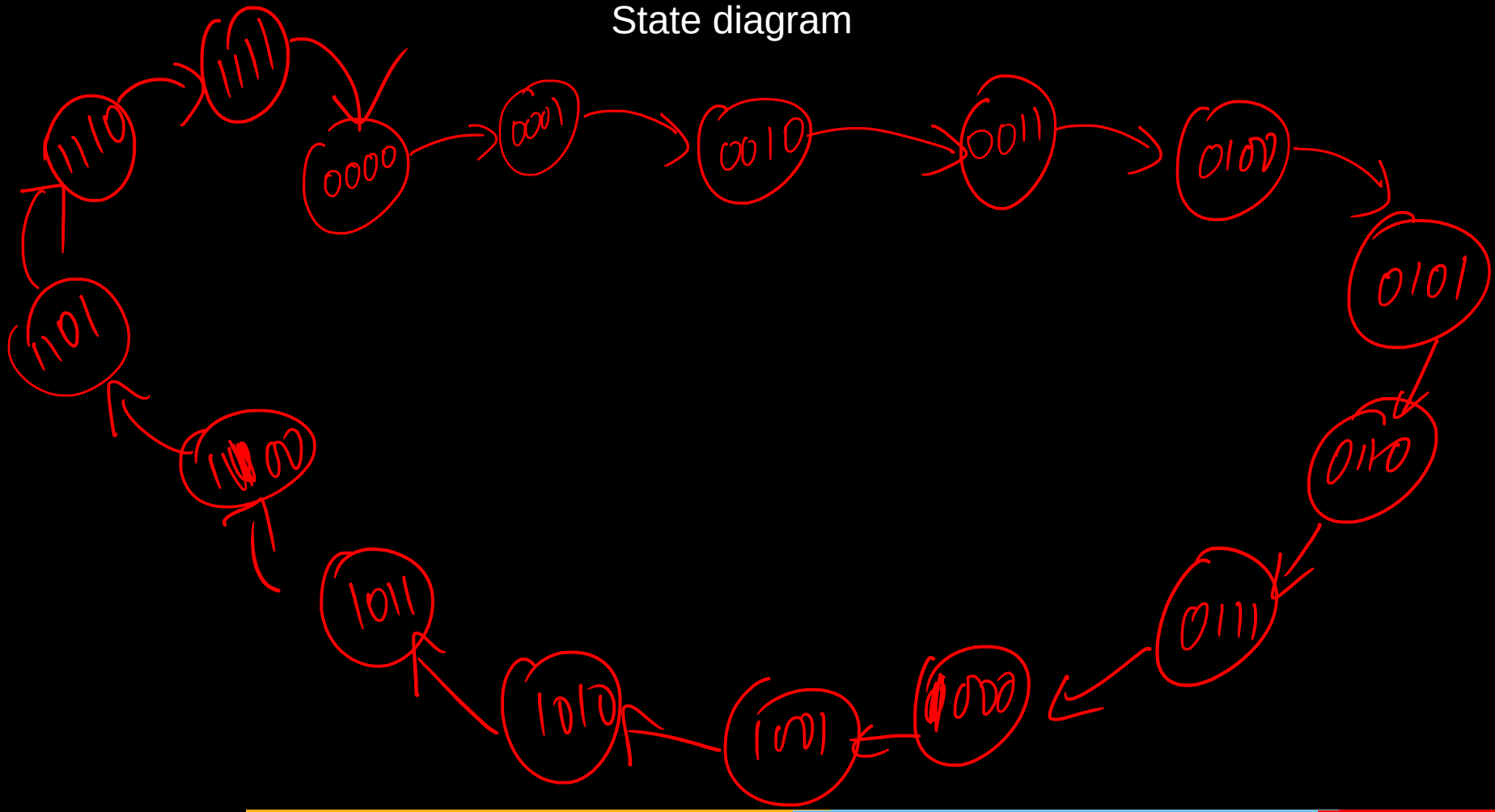
Synchronous Counters

- The design procedure for a synchronous counter is the same as any other synchronous sequential circuit.
- The primary inputs of the circuit are the CLK and any control signals (EN, Load, etc).
- The primary outputs are the FF outputs (present state).
- Most efficient implementations usually use T-FFs or JK-FFs. (complementing FFs)

We will examine JK, T and D flip-flop designs.

J-K Flip Flop Design of a 4-bit Binary Up Counter

State diagram



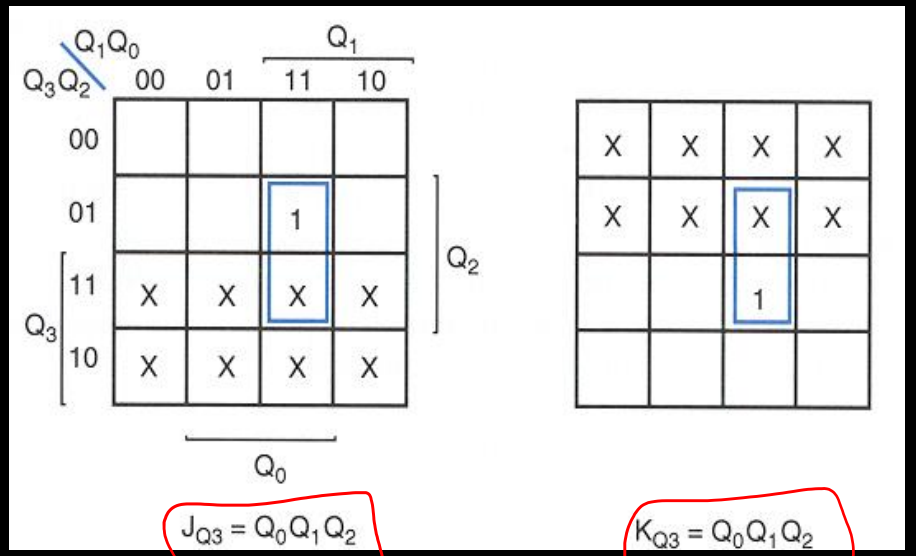
J-K Flip Flop Design of a 4-bit Binary Up Counter

Present state				Next state				Flip-flop inputs							
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	J_{Q3}	K_{Q3}	J_{Q2}	K_{Q2}	J_{Q1}	K_{Q1}	J_{Q0}	K_{Q0}
0	0	0	0	0	0	0	1	0	×	0	×	0	×	1	×
0	0	0	1	0	0	1	0	0	×	0	×	1	×	×	1
0	0	1	0	0	0	1	1	0	×	0	×	×	0	1	×
0	0	1	1	0	1	0	0	0	×	1	×	×	1	×	1
0	1	0	0	0	1	0	1	0	×	×	0	0	×	1	×
0	1	0	1	0	1	1	0	0	×	×	0	1	×	×	1
0	1	1	0	0	1	1	1	0	×	×	0	×	0	1	×
0	1	1	1	1	0	0	0	1	×	×	1	×	1	×	1
1	0	0	0	1	0	0	1	×	0	0	×	0	×	1	×
1	0	0	1	1	0	1	0	×	0	0	×	1	×	×	1
1	0	1	0	1	0	1	1	×	0	0	×	×	0	1	×
1	0	1	1	1	1	0	0	×	0	1	×	×	1	×	1
1	1	0	0	1	1	0	1	×	0	×	0	0	×	1	×
1	1	0	1	1	1	1	0	×	0	×	0	1	×	×	1
1	1	1	0	1	1	1	1	×	0	×	0	×	0	1	×
1	1	1	1	0	0	0	0	×	1	×	1	×	1	×	1

Synchronous Binary Counters:

J-K Flip Flop Design of a Binary Up Counter

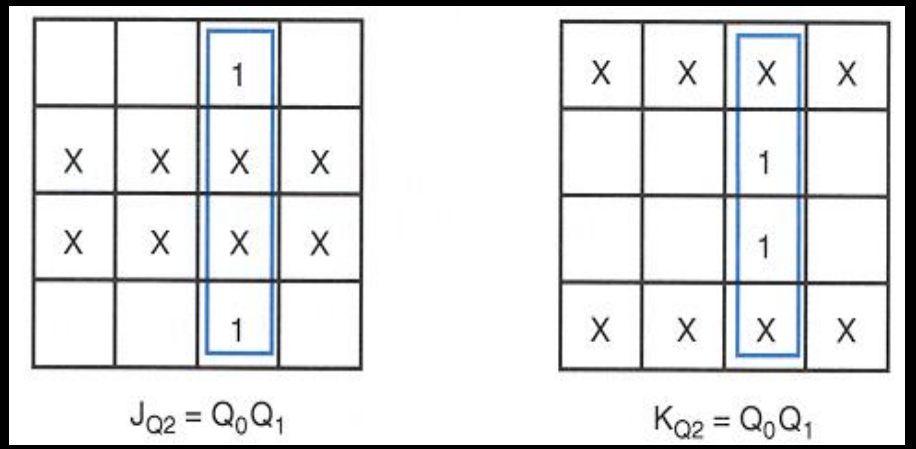
Present state				Next state					
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	J_{Q3}	K_{Q3}
0	0	0	0	0	0	0	1	0	X
0	0	0	1	0	0	1	0	0	X
0	0	1	0	0	0	1	1	0	X
0	0	1	1	0	1	0	0	0	X
0	1	0	0	0	1	0	1	0	X
0	1	0	1	0	1	1	0	0	X
0	1	1	0	0	1	1	1	0	X
0	1	1	1	1	0	0	0	1	X
1	0	0	0	1	0	0	1	X	0
1	0	0	1	1	0	1	0	X	0
1	0	1	0	1	0	1	1	X	0
1	0	1	1	1	1	0	0	X	0
1	1	0	0	1	1	0	1	X	0
1	1	0	1	1	1	1	0	X	0
1	1	1	0	1	1	1	1	X	0
1	1	1	1	0	0	0	0	X	1



Synchronous Binary Counters:

J-K Flip Flop Design of a Binary Up Counter

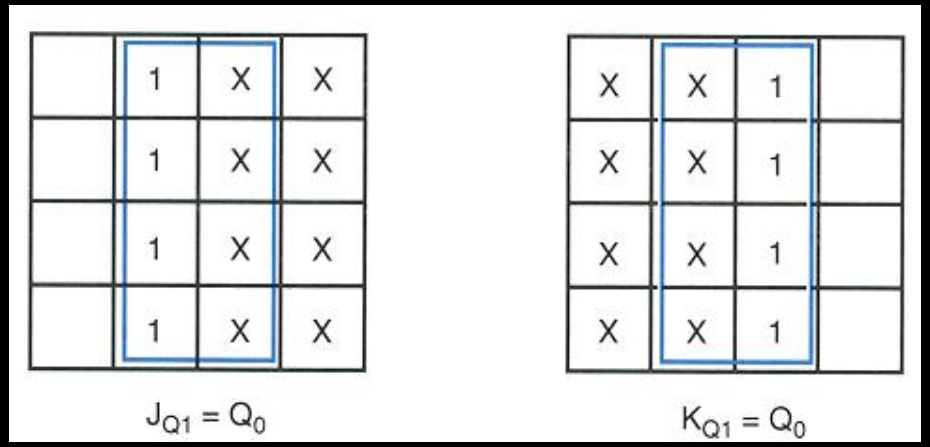
Present state				Next state				Flip-flo	
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	J_{Q2}	K_{Q2}
0	0	0	0	0	0	0	1	0	X
0	0	0	1	0	0	1	0	0	X
0	0	1	0	0	0	1	1	0	X
0	0	1	1	0	1	0	0	1	X
0	1	0	0	0	1	0	1	X	0
0	1	0	1	0	1	1	0	X	0
0	1	1	0	0	1	1	1	X	0
0	1	1	1	1	0	0	0	X	1
1	0	0	0	1	0	0	1	0	X
1	0	0	1	1	0	1	0	0	X
1	0	1	0	1	0	1	1	0	X
1	0	1	1	1	1	0	0	1	X
1	1	0	0	1	1	0	1	X	0
1	1	0	1	1	1	1	0	X	0
1	1	1	0	1	1	1	1	X	0
1	1	1	1	0	0	0	0	X	1

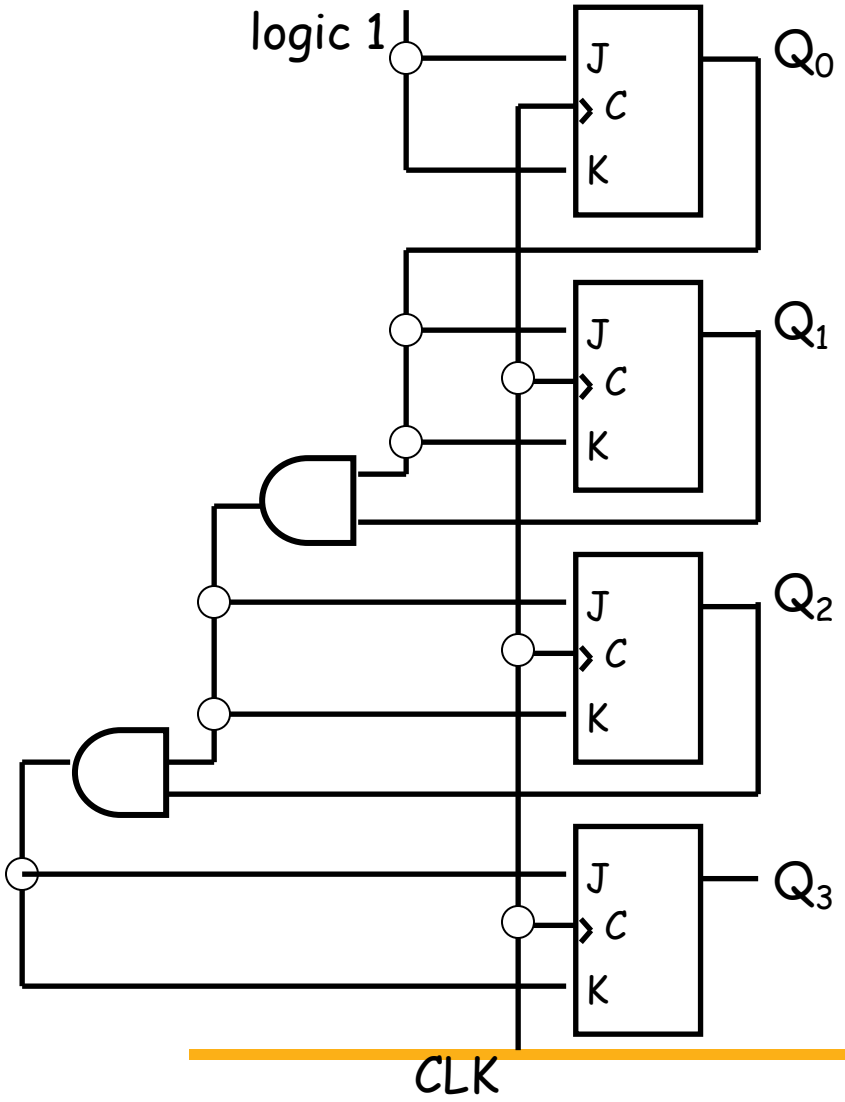


Synchronous Binary Counters:

J-K Flip Flop Design of a Binary Up Counter

Present state				Next state				p inputs	
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	J_{Q1}	K_{Q1}
0	0	0	0	0	0	0	1	0	X
0	0	0	1	0	0	1	0	1	X
0	0	1	0	0	0	1	1	X	0
0	0	1	1	0	1	0	0	X	1
0	1	0	0	0	1	0	1	0	X
0	1	0	1	0	1	1	0	1	X
0	1	1	0	0	1	1	1	X	0
0	1	1	1	1	0	0	0	X	1
1	0	0	0	1	0	0	1	0	X
1	0	0	1	1	0	1	0	1	X
1	0	1	0	1	0	1	1	X	0
1	0	1	1	1	1	0	0	X	1
1	1	0	0	1	1	0	1	0	X
1	1	0	1	1	1	1	0	1	X
1	1	1	0	1	1	1	1	X	0
1	1	1	1	0	0	0	0	X	1





$J_{Q0} = 1$
$K_{Q0} = 1$

$J_{Q1} = Q_0$
$K_{Q1} = Q_0$

$J_{Q2} = Q_0 Q_1$
$K_{Q2} = Q_0 Q_1$

$J_{Q3} = Q_0 Q_1 Q_2$
$K_{Q3} = Q_0 Q_1 Q_2$