

Ripple (Asynchronous) Counters

Sparsh Mittal



What is a counter?

A register that goes through a prescribed sequence of states upon the application of input pulses is called a *counter*

A counter that follows the binary number sequence is called a *binary counter*.

An n -bit binary counter consists of n flip-flops and can count in binary from 0 through $2^n - 1$.

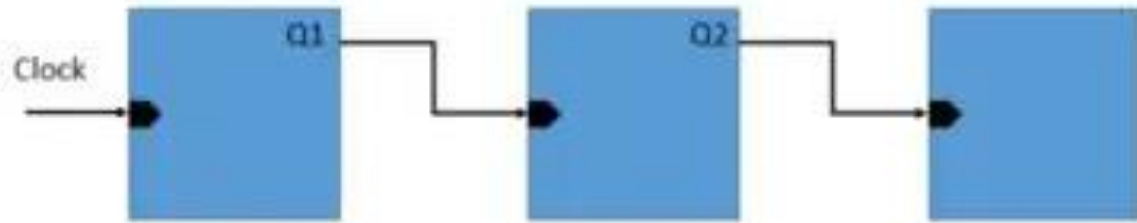
Ripple vs synchronous counters

Two types of counters: ripple (asynchronous) and synchronous.

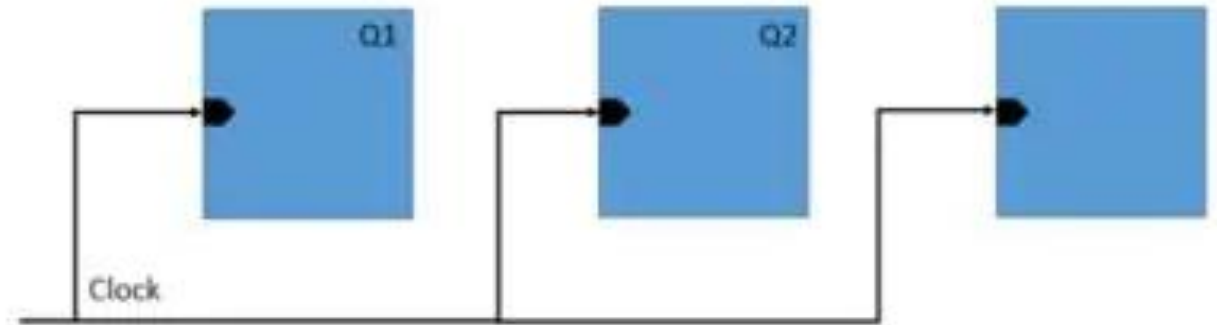
Ripple: a flip-flop output transition serves as a source for triggering other flip-flops.

In other words: C input of some or all flip-flops are triggered, not by the common clock pulses, but rather by the transition that occurs in other flip-flop outputs.

Synchronous: C inputs of all flip-flops receive the common clock.



Ripple (asynchronous)



Synchronous

Background: How to design complementing flip-flop

1. From a JK flip-flop with the J and K inputs tied together
2. From a T flip-flop.
3. Use a D flip-flop with the complement output connected to the D input. In this way, the D input is always the complement of the present state, and the next clock pulse will cause the flip-flop to complement.



Asynchronous (ripple) counter

Binary ripple counter

It has a series connection of complementing flip-flops, with the output of each flip-flop connected to the C input of the next higher order flip-flop.

The flip-flop holding the least significant bit receives the incoming count pulses.

Lets observe the pattern of bit-flips

Pattern of flips

A0 is complemented with each count pulse input.

Every time A0 goes from 1 to 0, it complements A1.

Every time that A1 goes from 1 to 0, it complements A2.

Every time that A2 goes from 1 to 0, it complements A3.

Based on this observation, we design the circuit.

Binary Count Sequence (0 to 8)

A_3	A_2	A_1	A_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

Circular counting from 0 → 15 → 0

Four-bit binary ripple counter

D Flip-Flop

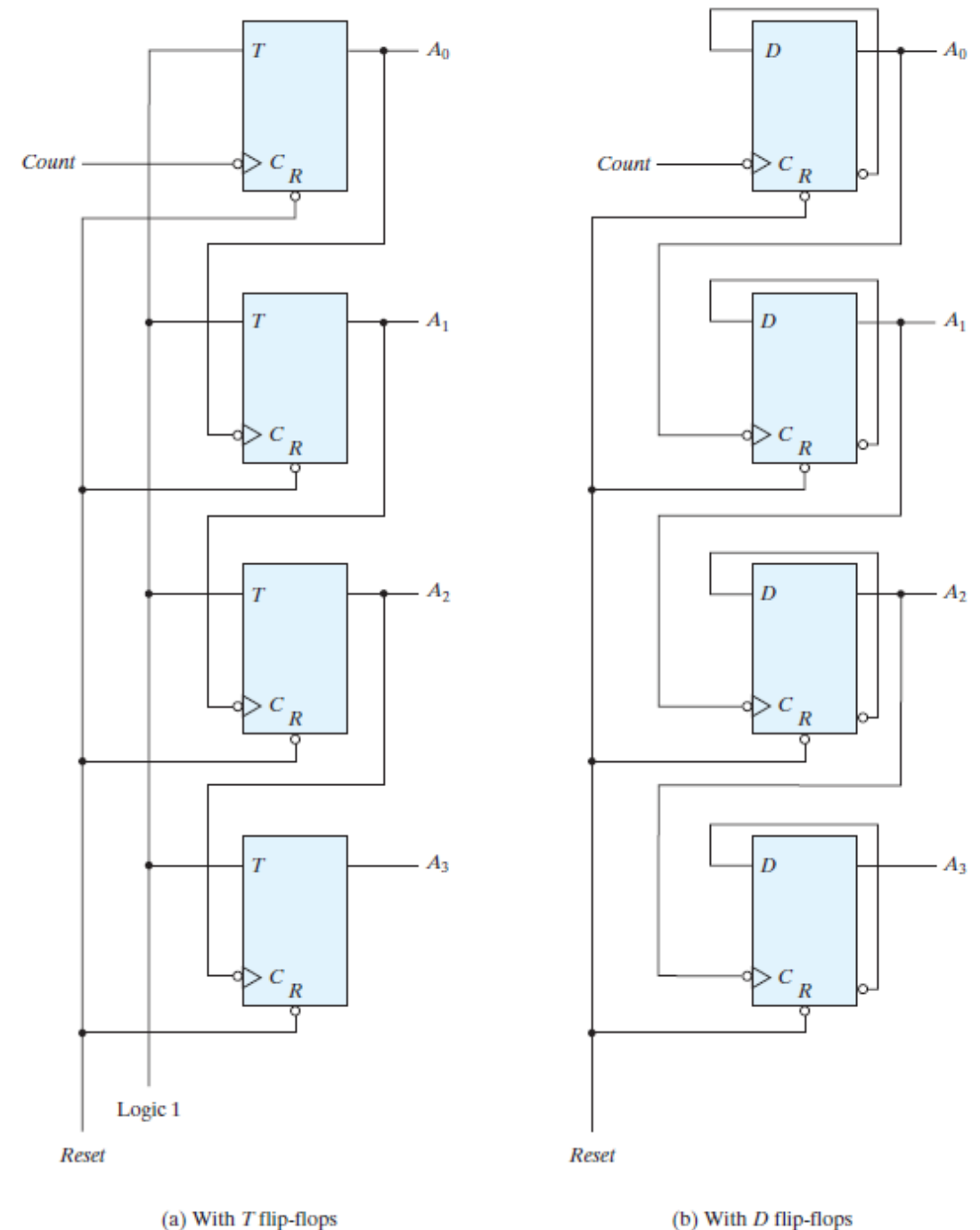
D	$Q(t + 1)$
0	0 Reset
1	1 Set

T Flip-Flop

T	$Q(t + 1)$
0	$Q(t)$ No change
1	$Q'(t)$ Complement

Output of each flip-flop is connected to the C input of the next flip-flop.

In (a), Each flip-flop complements if the signal in its C input goes through a negative transition (negative edge triggered, i.e., when output of previous flip-flop goes from 1 to 0).



Signal propagation in ripple fashion

Consider transition from 0011 to 0100. A0 is complemented with the count pulse.

Since A0 goes from 1 to 0, it triggers A1 and complements it.

As a result, A1 goes from 1 to 0, which in turn complements A2, changing it from 0 to 1.

A2 does not trigger A3, because A2 produces a positive transition and the flip-flop responds only to negative transitions.

Thus, the count from 0011 to 0100 is achieved by changing the bits one at a time.

Binary countdown counter

Counts from 15 to 0.

LSB is complemented with every count pulse.

Any other bit is complemented if its previous LSB goes from 0 to 1.

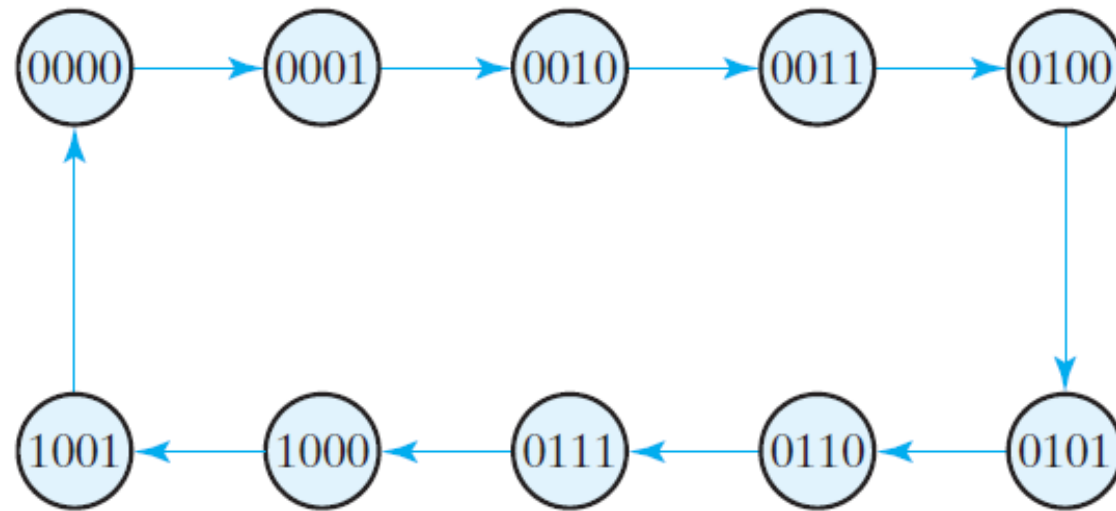
Therefore, a countdown counter looks the same as the binary ripple counter shown earlier, except that all flip-flops trigger on the positive clock-edge. (Bubble in the C inputs must be absent.)



BCD Ripple Counter

BCD Ripple Counter

A decimal counter is similar to a binary counter, except that the state after 1001_2 (9) is 0000_2 (0).



Counting sequence

Q1 changes state after each clock pulse.

Q2 complements every time **Q1** goes from 1 to 0, as long as $Q8 = 0$.

When **Q8** becomes 1, **Q2** remains at 0.

Q4 complements every time **Q2** goes from 1 to 0.

Q8 remains at 0 as long as **Q2** or **Q4** is 0.

When both **Q2** and **Q4** become 1, **Q8** complements when **Q1** goes from 1 to 0.

Q8 is cleared on the next transition of **Q1**.

Clock	Counter Output				State Number	BCD Number
	Q8	Q4	Q2	Q1		
Initially	0	0	0	0	—	0000
1 st	0	0	0	1	1	0001
2 nd	0	0	1	0	2	0010
3 rd	0	0	1	1	3	0011
4 th	0	1	0	0	4	0100
5 th	0	1	0	1	5	0101
6 th	0	1	1	0	6	0110
7 th	0	1	1	1	7	0111
8 th	1	0	0	0	8	1000
9 th	1	0	0	1	9	1001
10 th	0	0	0	0	0	0000

BCD Ripple Counter

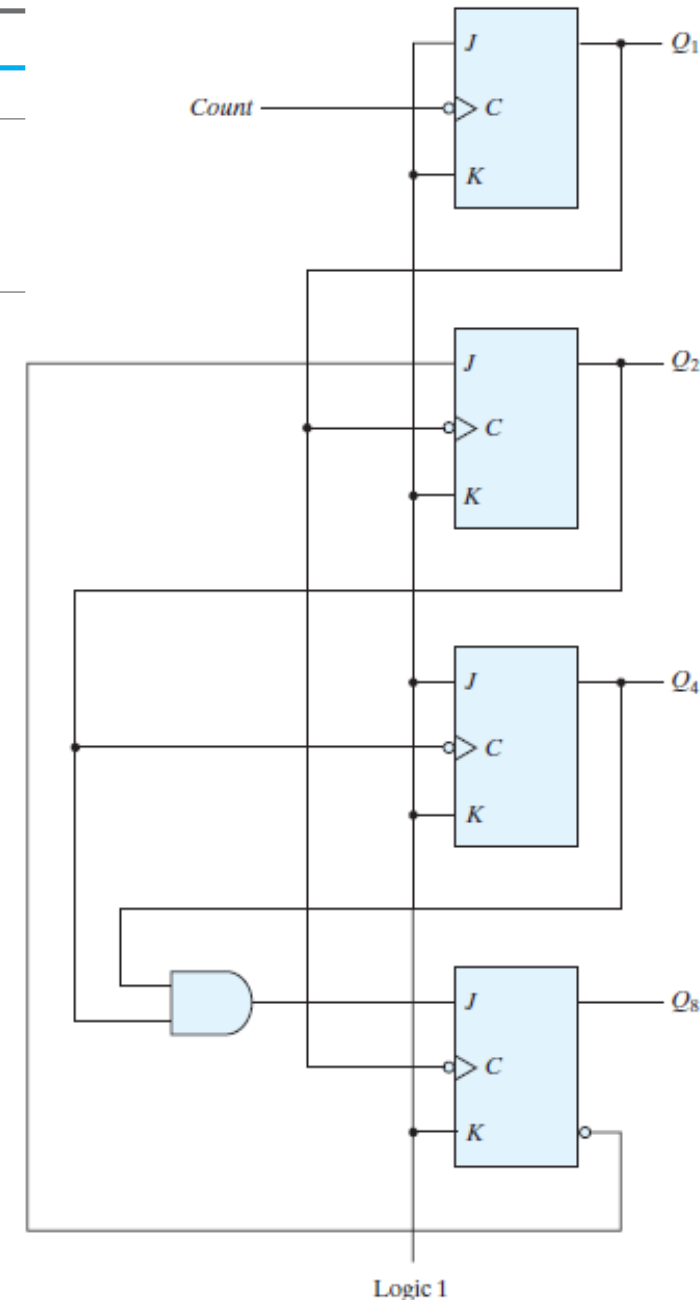
Table 5.1
Flip-Flop Characteristic Tables

JK Flip-Flop				
J	K	$Q(t + 1)$		
0	0	$Q(t)$	No change	
0	1	0	Reset	
1	0	1	Set	
1	1	$Q'(t)$	Complement	

Output of Q_1 is applied to the C inputs of both Q_2 and Q_8 and output of Q_2 is applied to the C input of Q_4 .

The J and K inputs are connected either to a permanent 1 signal or to outputs of other flip-flops.

Remember that when the C input goes from 1 to 0, flip-flop is set if $J = 1$, is cleared if $K = 1$, is complemented if $J = K = 1$, and is left unchanged if $J = K = 0$.



BCD Ripple Counter

Table 5.1

Flip-Flop Characteristic Tables

<i>JK</i> Flip-Flop			
<i>J</i>	<i>K</i>	<i>Q(t + 1)</i>	
0	0	<i>Q(t)</i>	No change
0	1	0	Reset
1	0	1	Set
1	1	<i>Q'(t)</i>	Complement

Q1 changes state after each clock pulse.

Q2 complements every time **Q1** goes from 1 to 0, as long as $Q8 = 0$.

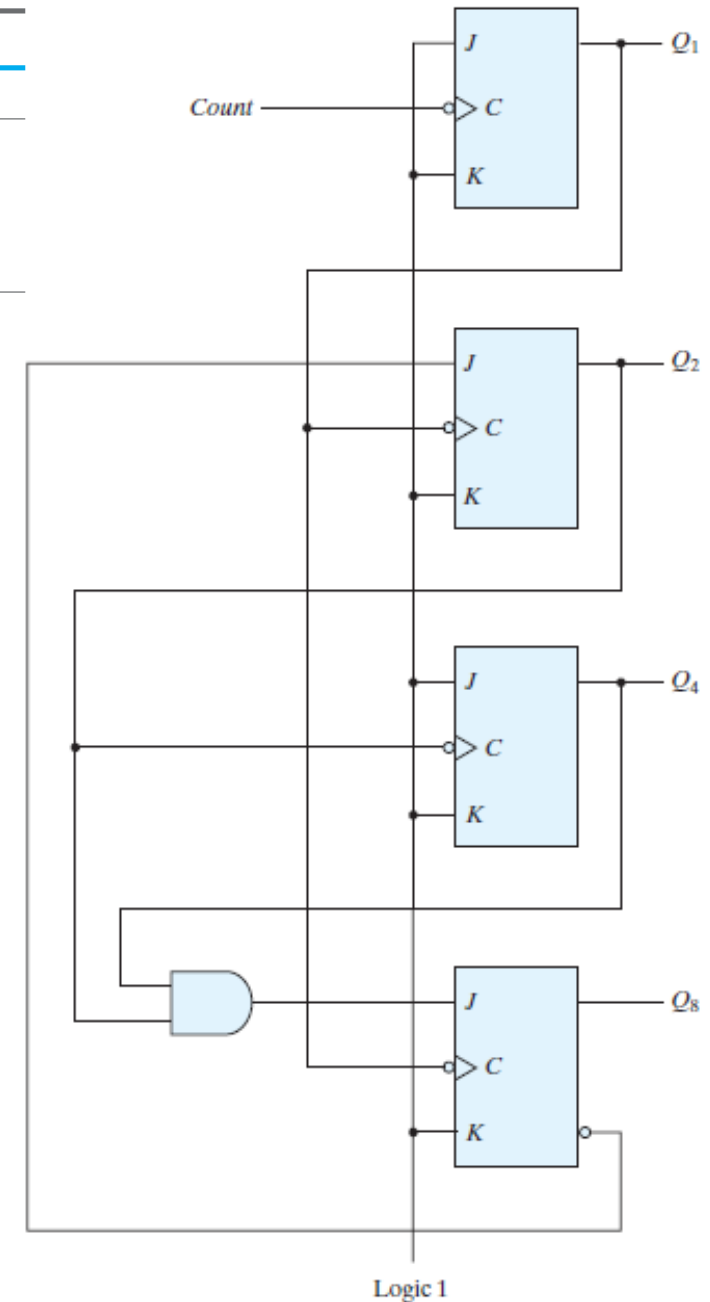
When Q_8 becomes 1, Q_2 remains at 0.

Q4 complements every time **Q2** goes from 1 to 0.

Q_8 remains at 0 as long as Q_2 or Q_4 is 0.

When both $Q2$ and $Q4$ become 1, $Q8$ complements when $Q1$ goes from 1 to 0.

Q_8 is cleared on the next transition of Q_1 .



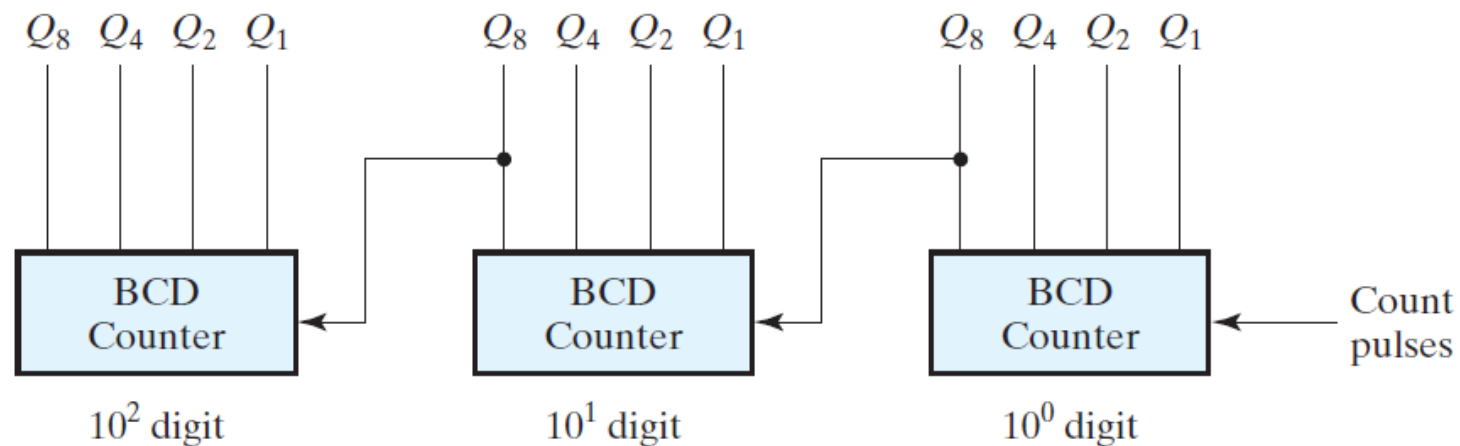
Multi-decade counter

BCD counter shown previously was a decade counter (counts from 0 to 9).

To count from 0 to 999, we need a three-decade counter.

The inputs to second and third decades come from Q_8 of 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 decade goes from 9 to 0.



Block diagram of a three-decade decimal BCD counter

Limitation of asynchronous counter

They suffer from “Propagation Delay” in which the timing signal is delayed a fraction through each flip-flop.

In **synchronous counter**, the external clock signal is connected to the clock input of EVERY individual flip-flop within the counter so that all of the flip-flops are clocked together simultaneously at the same time giving a fixed time relationship.

==> Changes in the output occur in “synchronisation” with the clock signal.

==> no ripple effect and no propagation delay.

==> The maximum operating frequency of this counter is much higher than that for a similar asynchronous counter.