Counters

Presented by Nabanita Das

Counters

- Counters are sequential circuits which "count" through a specific state sequence. It consists of set of flipflops which stores and sometimes displays the number of times a particular event or process has occurred depending upon on a clock or sometime other pulse.
- It is used for counting pulses. Counters can count up, count down, or count through other fixed sequences.
- Two distinct types are in common usage:

✓ **Asynchronous Counters (Ripple or Serial Counters)**

- Clock connected to the flip-flop clock input on the LSB bit flip-flop
- For all other bits, a flip-flop output is connected to the clock input.
- Output change is delayed more for each bit toward the MSB.

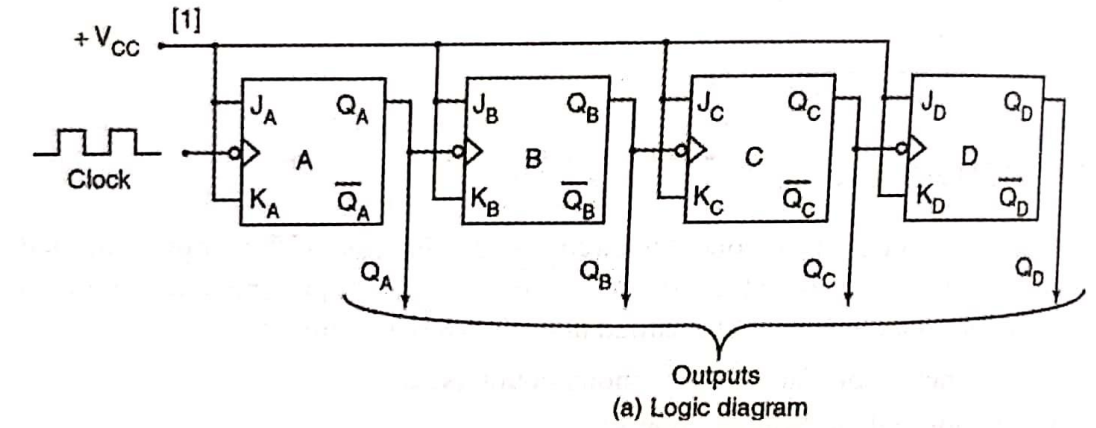
✓ **Synchronous Counters (Parallel Counter)**

- Clock is directly connected to the flip-flop clock inputs
- Logic is used to implement the desired state sequencing

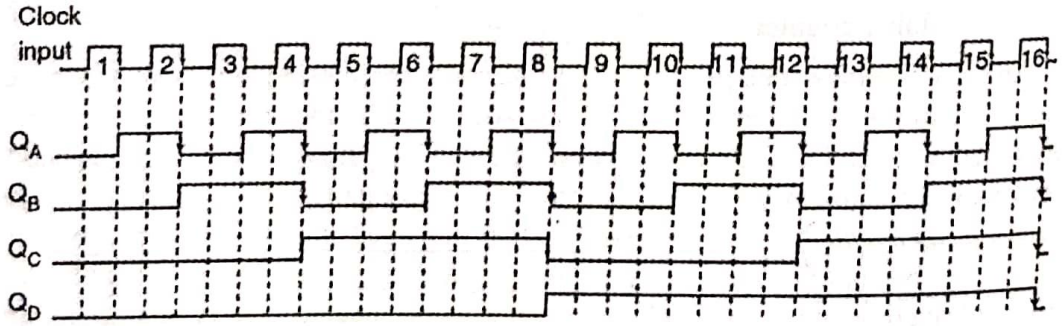
Asynchronous (Ripple) Counter

- In an Asynchronous counter each flip-flop is triggered by the output from the previous flip-flop.
- The settling time of this counter is cumulative sum of individual settling times of flip-flops.
- It is also known as serial counter.
- In an Asynchronous counter the first flip-flop is clocked by the external clock pulse, and then each successive flip-flop is clocked by the Q or Q' output of the previous flip-flop.
- Asynchronous counters are also called *ripple-counters* because of the way the clock pulse ripples it way through the flip-flops.
- The overall propagation delay time of the counter is the sum of all the individual delays of flip-flops.

Asynchronous (Ripple) Counter or UP Counter



(a) Logic diagram



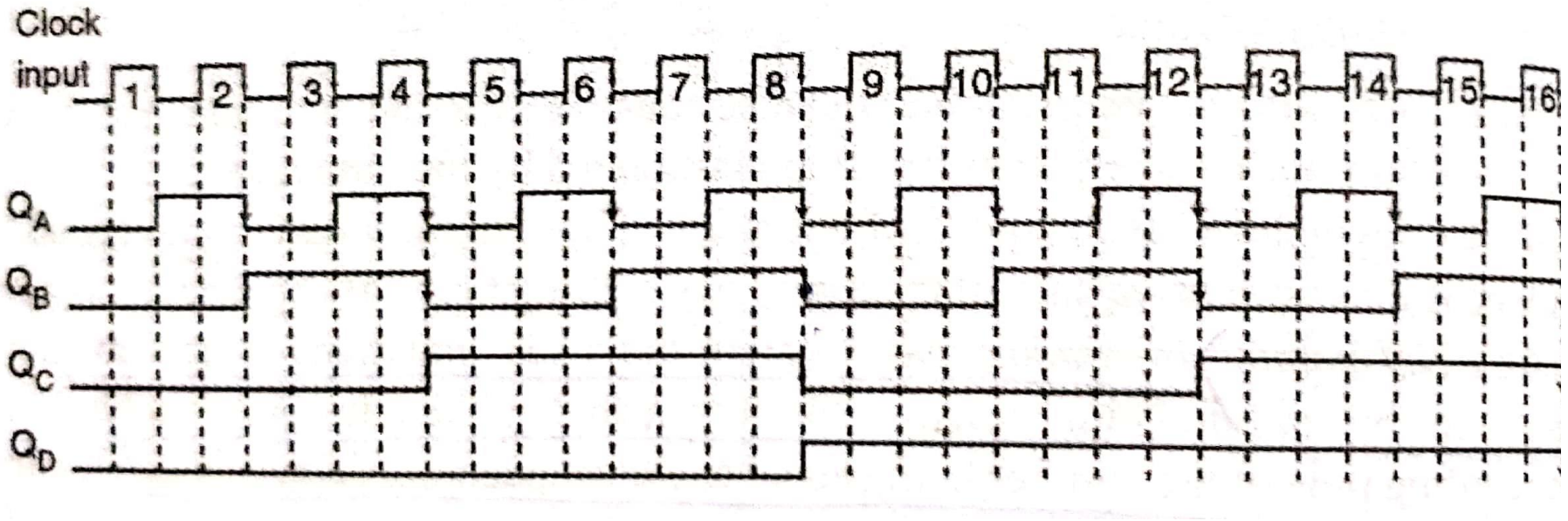
(b) Wave forms

Fig. 8.1 4-bit binary ripple counter

Q_D	Q_C	Q_B	Q_A
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	1	1	1

Ripple Counter Timing Diagram

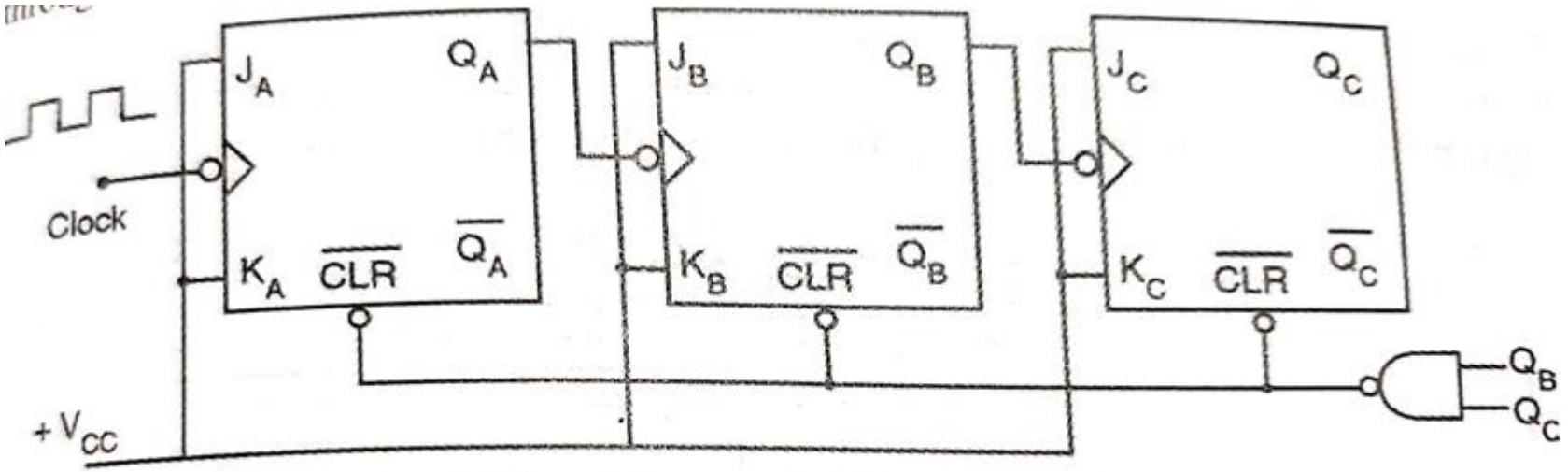
The ideal count sequence for the ripple counter yields the timing diagram below



Q_D	Q_C	Q_B	Q_A
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	1	1	1

Asynchronous (Ripple) Counter with modulus < 2^n

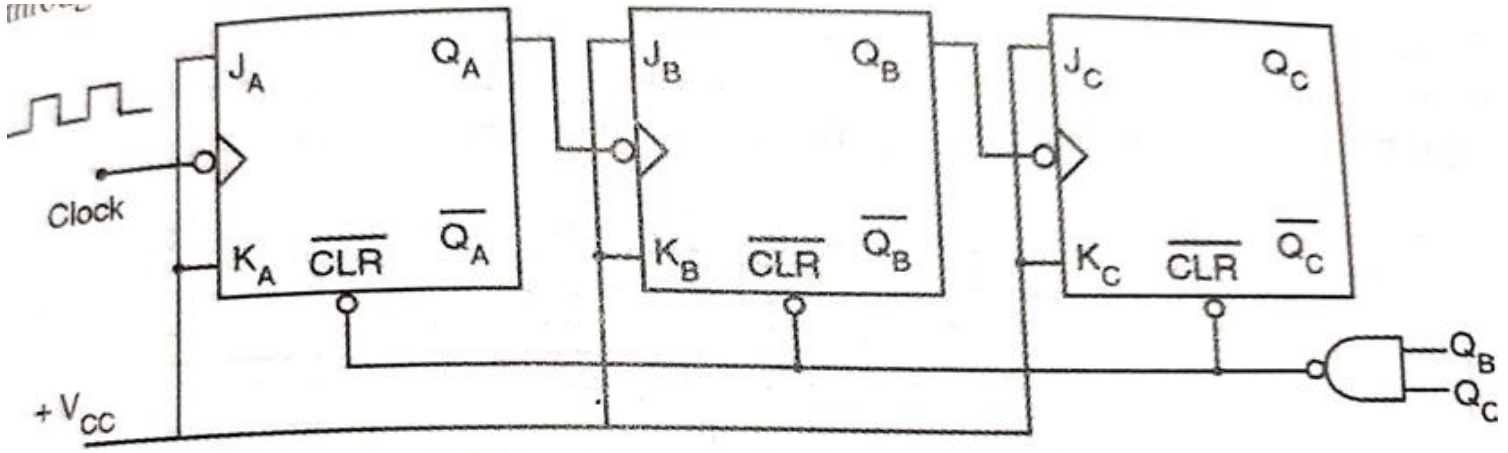
- Basic ripple counter is limited to a MOD-number that is, where n is the number of flip-flops equal to 2^n.
- This basic counter is modified to produce MOD-number less than 2^n by skip the states.
- For this purpose NAND gates is used. Without NAND gate it uses as MOD 8 counter and using NAND gates it uses as MOD 6 counter.



Asynchronous (Ripple) Counter with modulus < 2^n contd.

- NAND gate output is connected to CLEAR input of each flip-flops.
- When NAND gate output is HIGH it will have no effect on the counter; when the output is low it will clear all the flip-flops, counter goes to 000 state.
- The output of the counter Q_B and Q_C are given input of the NAND gates when $Q_B=Q_C=1$ NAND gates goes low. It occurs when counter goes to 101 to 110.

Q_C	Q_B	Q_A
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1



Asynchronous Down Counter

- A down counter using n flip-flops counts downward from maximum count of $(2^n - 1)$

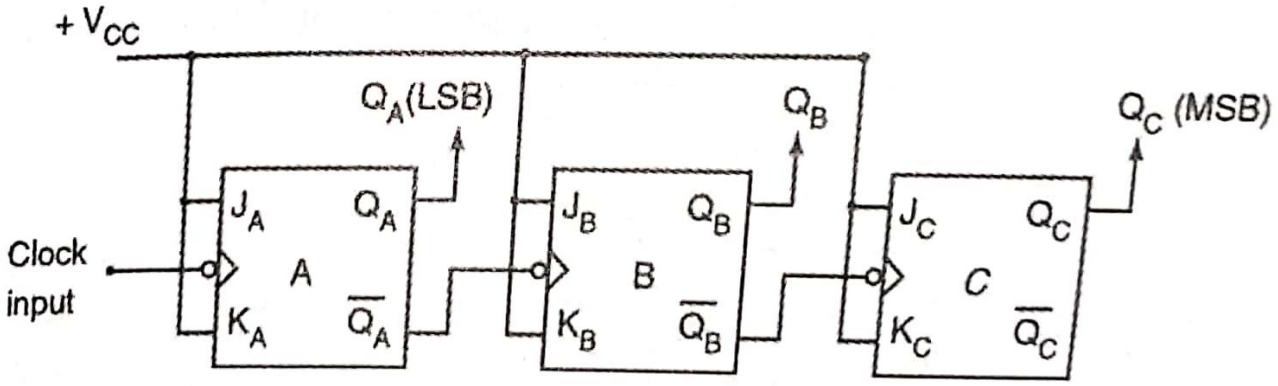


Fig. 8.8 MOD-8 down counter

State	Q_C	Q_B	Q_A
7	1	1	1
6	1	1	0
5	1	0	1
4	1	0	0
3	0	1	1
2	0	1	0
1	0	0	1
0	0	0	0
7	1	1	1

Asynchronous UP-Down Counter

8.7 UP-DOWN COUNTER

The UP-DOWN counter is a combinaiton of the up-counter and the down-counter. As the UP-DOWN counter has the capability of counting upwards as well as downwards, it is also called *Multimode counter*. In an UP-counter, each flip-flop is triggered by the *normal* output of the preceding flip-flop; in a DOWN-counter, each flip-flop is triggered by the inverted output of the preceding flip-flop. In both the counters, the first flip-flop is triggered by the input pulses.

A 4-bit UP-DOWN counter whose operation is controlled by the UP and DOWN control inputs is shown in Fig. 8.10. The counting sequence of UP/DOWN counter in the two modes of counting is given in Table 8.5.

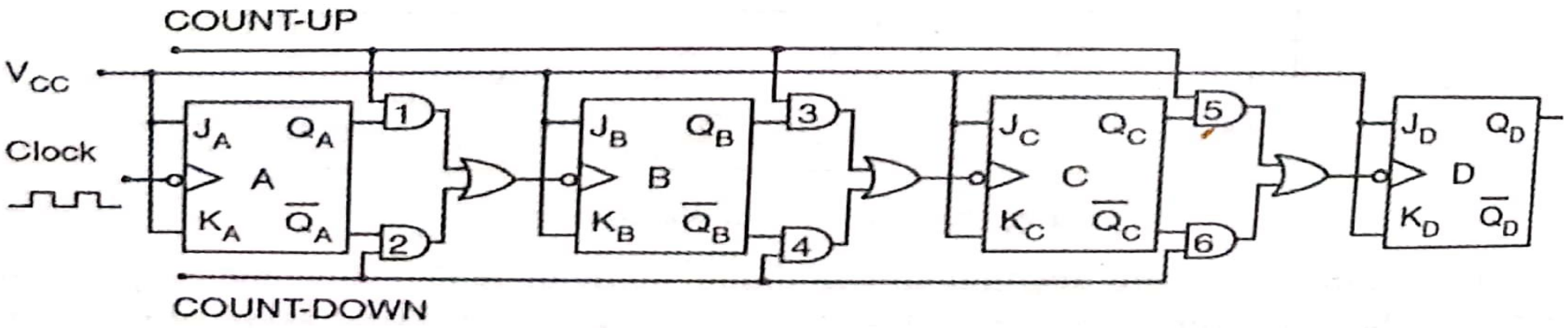


Fig. 8.10 Asynchronous 4-bit UP-DOWN counter

Propagation Delay in Ripple Counter

- A major problem with ripple counters arises from the propagation delay of the flip-flops constituting the counter.
- The effective propagation delay in a ripple counter is equal to the sum of propagation delays due to different flip-flops. If **t_{pd}** is the propagation delay in each flip-flop, then, in a counter with N flip-flops having a modulus of less than or equal to 2^n, $T_{\text{clock}} \geq n \times t_{pd}$

Thus, the maximum frequency that can be used in an asynchronous counter is

$$\frac{1}{f_{\text{clock}}} \geq n \times t_{pd}$$

$$\frac{1}{n \times t_{pd}} \geq f_{\text{clock}}$$

Thus, f_{clock} should be less than or equal to $1/(n \times t_{pd})$. So, the maximum clock frequency that can be applied in an n-bit asynchronous counter is

$$f_{\max} = \frac{1}{n \times t_{pd}}$$

For example, a 3-bit counter having flip-flops with identical $t_{pd} = 50\text{ns}$ will have a maximum input frequency limit of

$$f_{\max} = \frac{1}{3 \times 50\text{ns}} = 6.67\text{MHz}$$

Synchronous UP Counter (Parallel Counter)

- Ripple Counter drawbacks are

I. Speed operation is low because the propagation delay time of all flip-flop is cumulative & total time is the product of the total numbers of flip-flops & the propagation delay of single flip-flop.

II. It glitches (small problem or fault) at the decoding gate outputs.

III. This problem is eliminated by applying clock pulses to all the flip-flops simultaneously which is done by synchronous counter.

IV. The flip-flop B changes its state when QA=1 in the negative transition at clock input. Similarly, the flip-flop C changes its state when QA= QB=1 and the flip-flop D changes its state when QA= QB= QC=1.

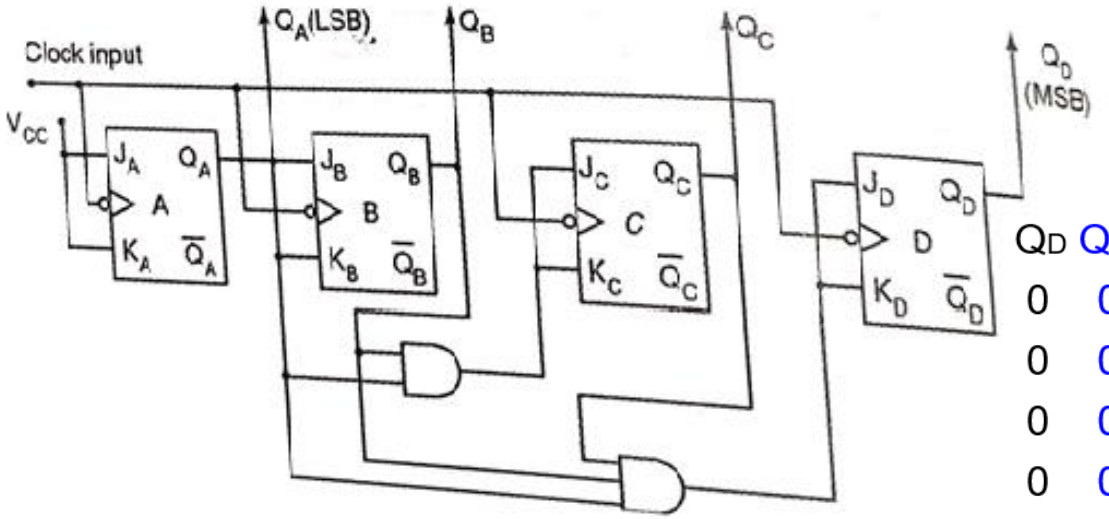


Fig. 8.11 4-bit synchronous counter

Q_D	Q_C	Q_B	Q_A
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	1	1	1

Propagation Delay in Synchronous Counter

- The time taken by one flip-flop to toggle plus the time for the new logic levels to propagate through a single AND gate to reach the J & K inputs of the following flip-flops.
- Total delay= Propagation delay of one flip-flop + Propagation delay of AND Gate

  = tp+ tg

  The maximum frequency of operation of synchronous counter is

  fmax= 1/ tp+ tg

Synchronous DOWN Counter

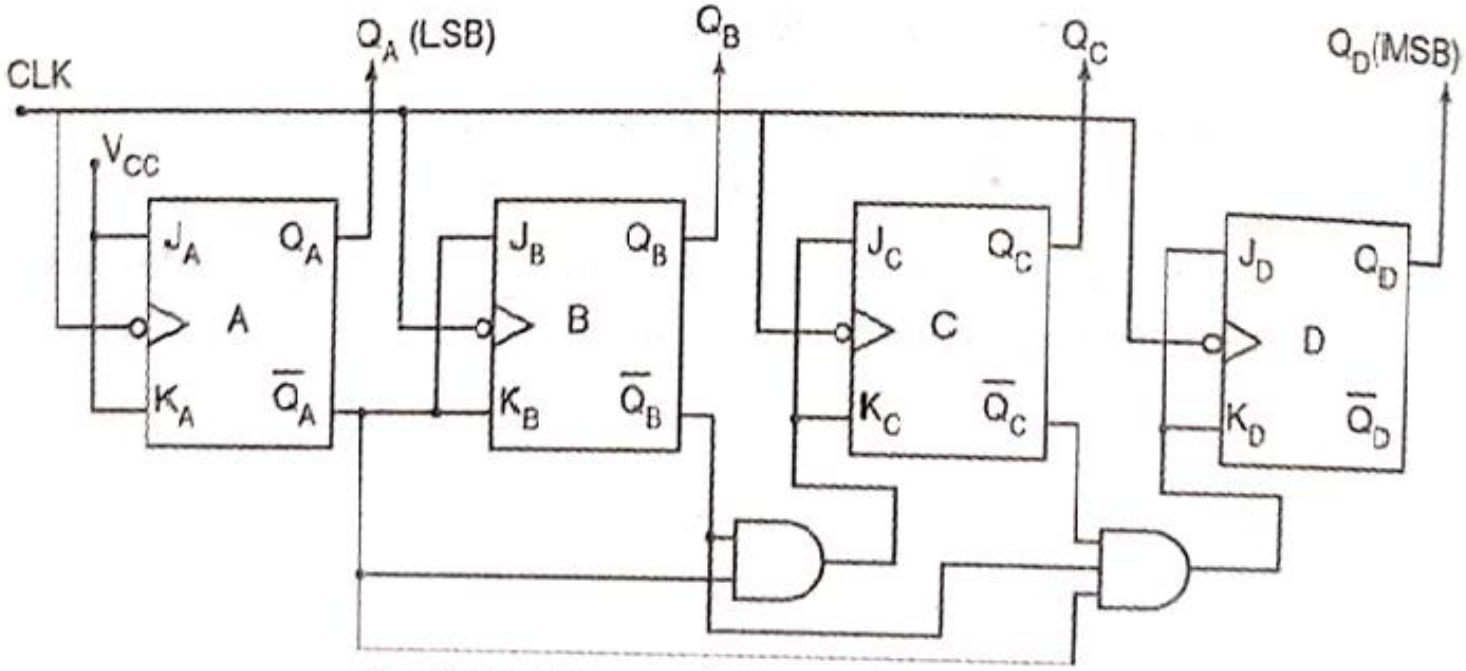


Fig. 8.13 4-bit synchronous down counter

State	Q_D	Q_C	Q_B	Q_A
15	1	1	1	1
14	1	1	1	0
13	1	1	0	1
12	1	1	0	0
11	1	0	1	1
10	1	0	1	0
9	1	0	0	1
8	1	0	0	0
7	0	1	1	1
6	0	1	1	0
5	0	1	0	1
4	0	1	0	0
3	0	0	1	1
2	0	0	1	0
1	0	0	0	1
0	0	0	0	0
15	1	1	1	1

Synchronous UP-DOWN Counter

8.12 SYNCHRONOUS UP / DOWN COUNTER

To form a parallel UP/DOWN counter, the control inputs (COUNT-UP and COUNT-DOWN) are used to allow either the normal output or the inverted output of one flip-flop to the J and K inputs of the following flip-flop. A MOD-8 UP/DOWN counter which will count from 000 to 111 when the COUNT-UP = 1 and COUNT-DOWN = 0, or from 111 to 000 when the COUNT-DOWN = 1 and COUNT-UP = 0, is shown in Fig. 8.14.

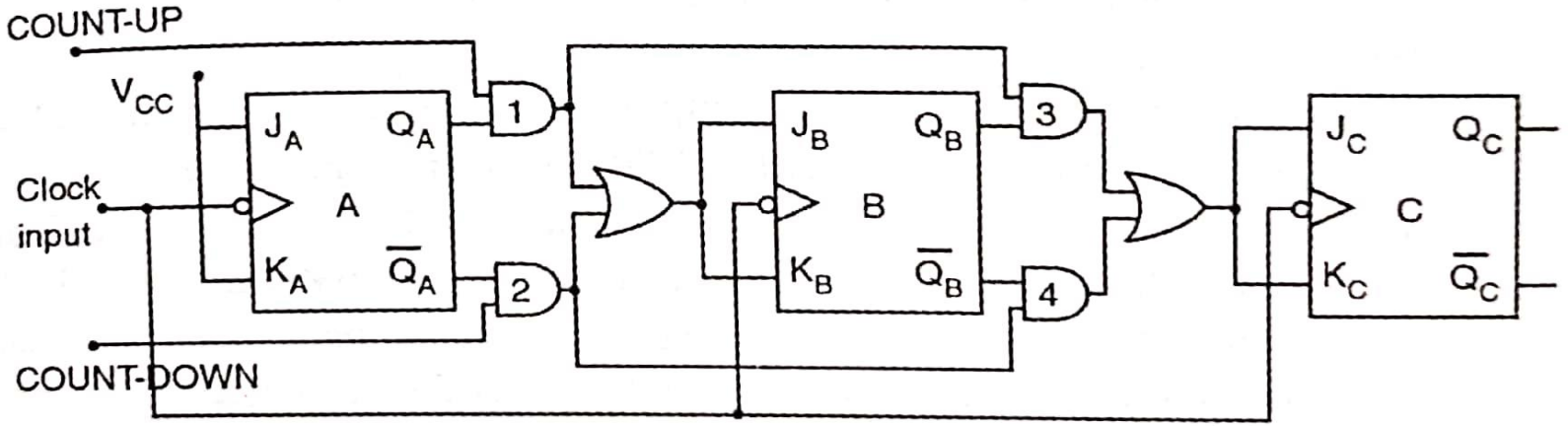


***Fig. 8.14** Parallel UP/DOWN counter (MOD-8)*

A logical 1 on the COUNT-UP line while COUNT-DOWN = 0 enables AND gates 1 and 3 and disables gates 2 and 4. This allows the Q_A and Q_B outputs through the AND gates to the J and K inputs of the following flip-flops, so that the counter counts up as pulses are applied. The reverse action takes place when COUNT-UP = 0 and COUNT-DOWN = 1.

Design of a Synchronous Counter

The steps involved in the design of synchronous counter are listed below:

Step 1 From the word description of the problem, draw a state diagram which describes the operation of the counter.

Step 2 From the above state diagram, obtain Present State – Next State (PS-NS) table of the counter and check the same to ascertain whether it has any equivalent states. Any two states are said to be equivalent, if and only if their next states are one and the same. In such a case, one of the equivalent states can be eliminated from the state table. Thus, in this step, the state table is modified in such a way that there is no redundant state in it.

Step 3 Make a state assignment and document the same in the above state table.

Step 4 Decide the type of memory element to be used in the counter design and then obtain the excitation table from PS–NS table using the application table of the flip-flop.

Step 5 Draw the excitation maps for various excitation inputs of flip-flops and simplify the excitation functions.

Step 6 Draw the schematic diagram of the counter.

Design MOD Synchronous Counter

- Design synchronous counter of any MOD number other than the power 2 is more difficult than asynchronous counter. The design is simplified by using K-Map.
- **Design of MOD 3 Counter**

MOD 3 counter has *THREE* states.

Flipflop required $2^n \geq N \geq 2^{n-1}$ where n is the number of flip-flops & N is the number of states.

For N=3, we need n=2 flip-flops

Design MOD Synchronous Counter contd.