UNIT: -8

REGISTERS and COUNTERS

6.1. Introduction to Registers:

- Flip-flop is a 1 bit memory cell which can be used for storing the digital data. To increase the storage capacity in terms of number of bits, we have to use a group of flip-flop. Such a group of flip-flop is known as a Register.
- The n-bit register will consist of n number of flip-flop and it is capable of storing an n-bit word.
- The binary data in a register can be moved within the register from one flip-flop to another.
- The registers that allow such data transfers are called as shift registers.

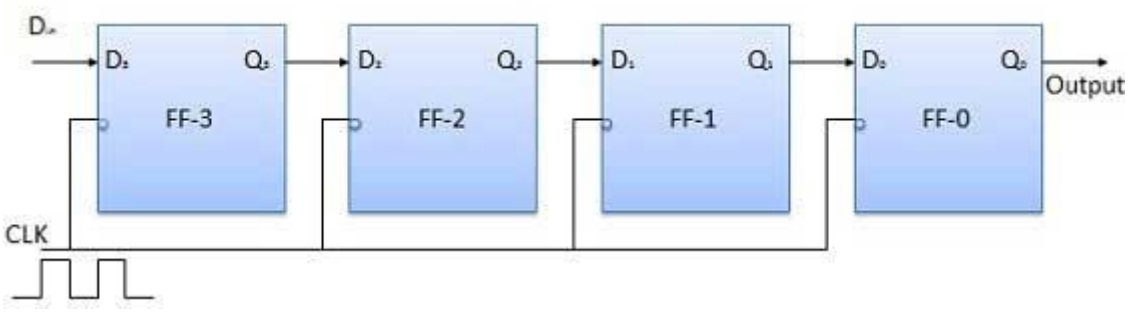
There are four modes of operations of a shift register.

- Serial Input Serial Output
- Serial Input Parallel Output
- Parallel Input Serial Output
- Parallel Input Parallel Output

1. Serial Input Serial Output:

- Let all the flip-flop be initially in the reset condition i.e. Q3 = Q2 = Q1 = Q0 = 0. If an entry of a four-bit binary number 1 1 1 1 is made into the register, this number should be applied to Din bit with the LSB bit applied first. The D input of FF-3 i.e. D3 is connected to serial data input Din. Output of FF-3 i.e. Q3 is connected to the input of the next flip-flop i.e. D2 and so on.

Block Diagram



Operation

Before application of clock signal, let Q3 Q2 Q1 Q0 = 0000 and apply LSB bit of the number to be entered to Din. So Din = D3 = 1. Apply the clock. On the first falling edge of clock, the FF-3 is set, and stored word in the register is Q3 Q2 Q1 Q0 = 1000.

Apply the next bit to Din. So Din = 1. As soon as the next negative edge of the clock hits, FF-2 will set and the stored word change to Q3 Q2 Q1 Q0 = 1100.

Apply the next bit to be stored i.e. 1 to Din. Apply the clock pulse. As soon as the third negative clock edge hits, FF-1 will be set and output will be modified to Q3 Q2 Q1 Q0 = 1110.

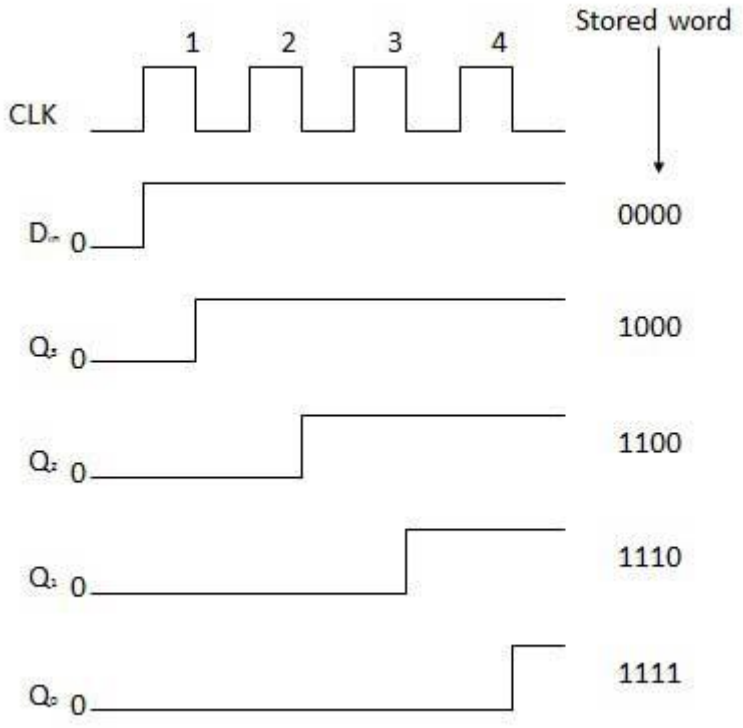
Similarly with Din = 1 and with the fourth negative clock edge arriving, the stored word in the register is Q3 Q2 Q1 Q0 = 1111.

Truth Table

	CLK	$D_{in} = Q_3$	$Q_3 = D_2$	$Q_2 = D_1$	$Q_1 = D_0$	Q_0
Initially			0	0	0	0
(i)	↓	1	1	0	0	0
(ii)	↓	1	1	1	0	0
(iii)	↓	1	1	1	1	0
(iv)	↓	1	1	1	1	1

→ Direction of data travel

Waveforms

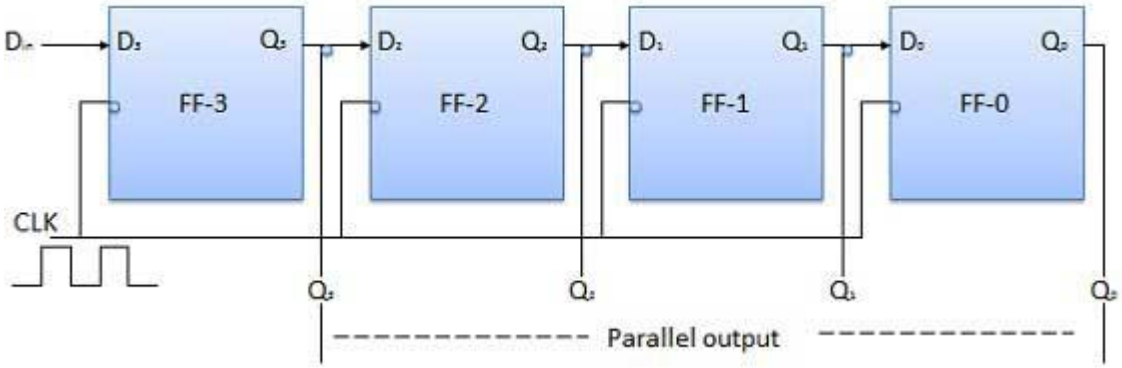


2. **Serial Input Parallel Output:**

- In such types of operations, the data is entered serially and taken out in parallel fashion.
- Data is loaded bit by bit.
- The outputs are disabled as long as the data is loading.

- As soon as the data loading gets completed, all the flip-flops contain their required data, the outputs are enabled so that all the loaded data is made available over all the output lines at the same time.
- 4 clock cycles are required to load a
- four-bit word.
- Hence the speed of operation of SIPO mode is same as that of SISO mode.

Block Diagram

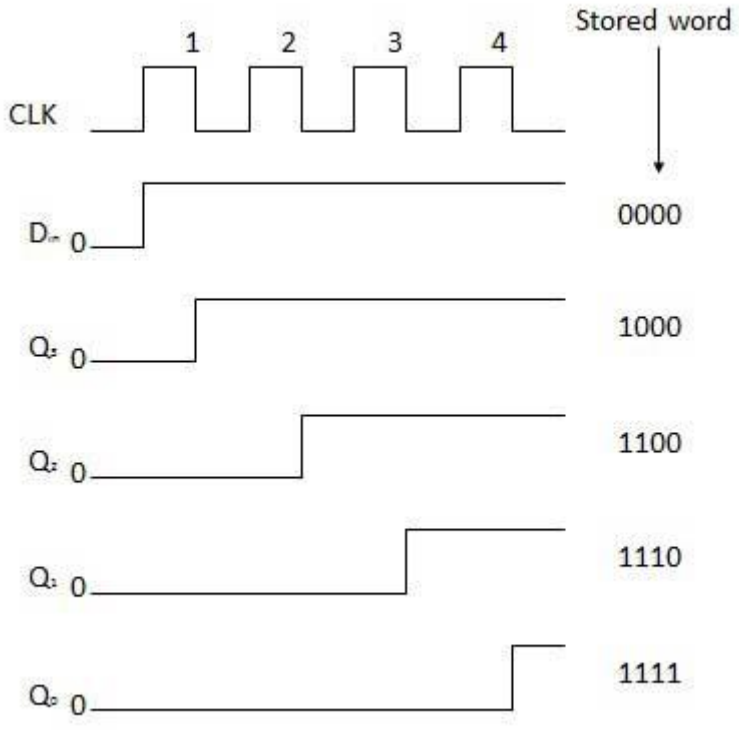


Truth Table:

	CLK	$D_{in} = Q_3$	$Q_3 = D_2$	$Q_2 = D_1$	$Q_1 = D_0$	Q_0
Initially			0	0	0	0
(i)	↓	1	1	0	0	0
(ii)	↓	1	1	1	0	0
(iii)	↓	1	1	1	1	0
(iv)	↓	1	1	1	1	1

→ Direction of data travel

Here the truth table is exact same as of SISO but the difference is here we will get output once the data gets shifted completely.



3. **Parallel Input Serial Output (PISO)**
   - Data bits are entered in parallel fashion.
   - The circuit shown below is a four bit parallel input serial output register.
   - Output of previous Flip Flop is connected to the input of the next one via a combinational circuit.
   - The binary input word B0, B1, B2, B3 is applied though the same combinational circuit.

There are two modes in which this circuit can work namely - shift mode or load mode.

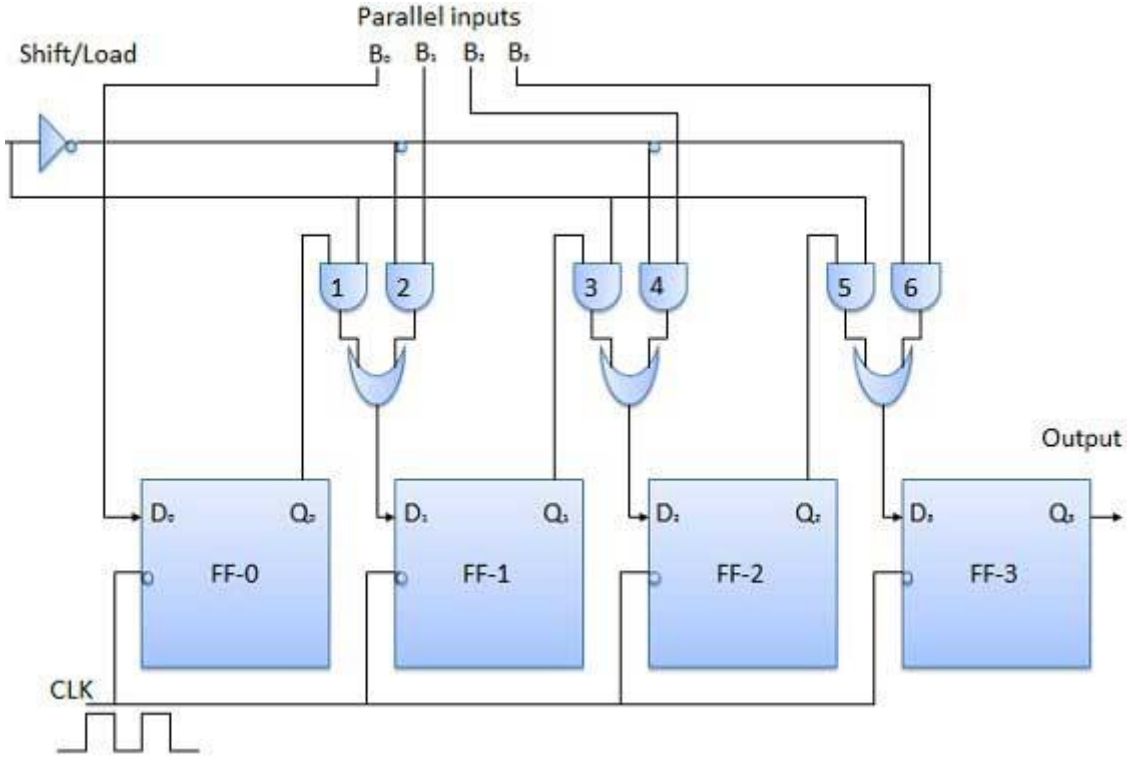
Load mode

When the shift/load bar line is low (0), the AND gate 2, 4 and 6 become active they will pass B1, B2, B3 bits to the corresponding flip-flops. On the low going edge of clock, the binary input B0, B1, B2, B3 will get loaded into the corresponding flip-flops. Thus parallel loading takes place.

Shift mode

When the shift/load bar line is low (1), the AND gate 2, 4 and 6 become inactive. Hence the parallel loading of the data becomes impossible. But the AND gate 1,3 and 5 become active. Therefore the shifting of data from left to right bit by bit on application of clock pulses. Thus the parallel in serial out operation takes place.

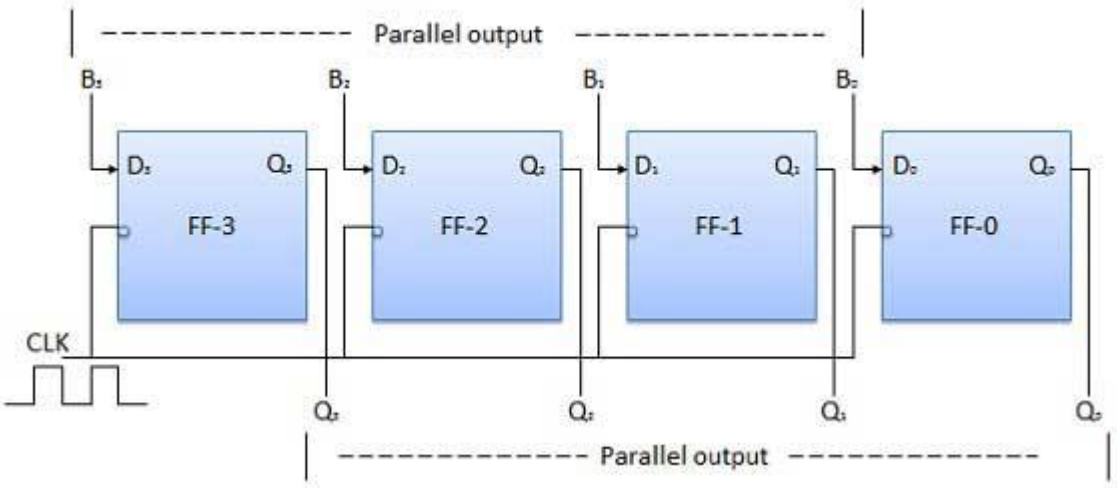
Block Diagram



4. Parallel Input Parallel Output (PIPO):

In this mode, the 4 bit binary input B0, B1, B2, B3 is applied to the data inputs D0, D1, D2, D3 respectively of the four flip-flops. As soon as a negative clock edge is applied, the input binary bits will be loaded into the flip-flops simultaneously. The loaded bits will appear simultaneously to the output side. Only clock pulse is essential to load all the bits.

Block Diagram



Truth table:

CLK	Q0	Q1	Q2	Q3
0	0	0	0	0
↓	1	1	1	1

Here the input is parallel that means all the data are entered at once and the output is also parallel that means as soon as the data are loaded into the Flipflop the output comes.

6.3. Introduction to Counters:

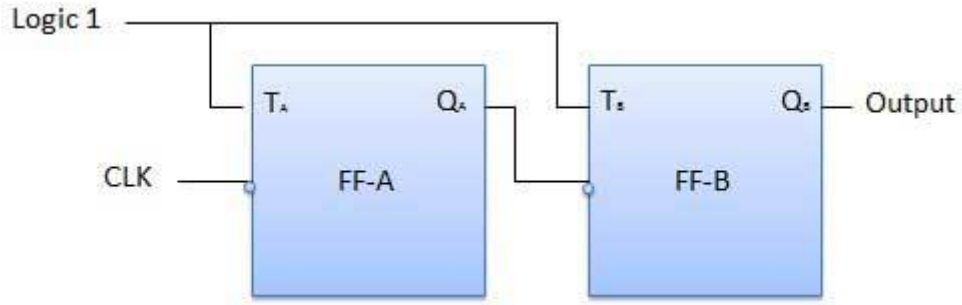
- Counter is a sequential circuit. A digital circuit which is used for a counting pulses is known counter. Counter is the widest application of flip-flops. It is a group of flip-flops with a clock signal applied.

Counters are of two types.
Asynchronous or ripple counters.
Synchronous counters.

Asynchronous or ripple counters

- The logic diagram of a 2-bit ripples up counter is shown in figure. The toggle (T) flip-flop is being used. But we can use the JK flip-flop also with J and K connected permanently to logic 1. External clock is applied to the clock input of flip-flop A and QA output is applied to the clock input of the next flip-flop i.e. FF-B.

Logical Diagram



Operation

S.N.	Condition	Operation
1	Initially let both the FFs be in the reset state	QBQA = 00 initially
2	After 1st negative clock edge	As soon as the first negative clock edge is applied, FF-A will toggle and QA will be equal to 1. QA is connected to clock input of FF-B. Since QA has changed from 0 to 1, it is treated as the positive clock edge by FF-B. There is no change in QB because FF-B is a negative edge triggered FF. QBQA = 01 after the first clock pulse.
3	After 2nd negative clock edge	On the arrival of second negative clock edge, FF-A toggles again and QA = 0. The change in QA acts as a negative clock edge for FF-B. So it will also toggle, and QB will be 1. QBQA = 10 after the second clock pulse.
4	After 3rd negative clock edge	On the arrival of 3rd negative clock edge, FF-A toggles again and QA become 1 from 0. Since this is a positive going change, FF-B does not respond to it and remains inactive. So QB does not change and continues to be equal to 1. QBQA = 11 after the third clock pulse.
5	After 4th negative clock edge	On the arrival of 4th negative clock edge, FF-A toggles again and QA becomes 1 from 0. This negative change in QA acts as clock pulse for FF-B. Hence it toggles to change QB from 1 to 0. QBQA = 00 after the fourth clock pulse.

Truth Table

Clock	Counter output		State number	Deciimal Counter output
	Q_B	Q_A		
Initially	0	0	—	0
1st	0	1	1	1
2nd	1	0	2	2
3rd	1	1	3	3
4th	0	0	4	0

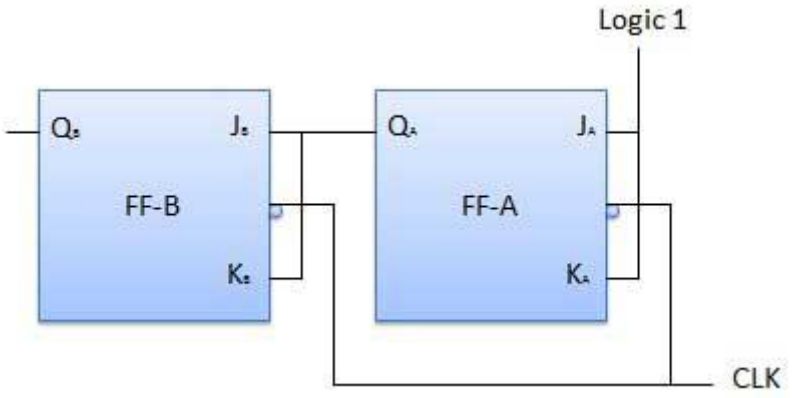
Synchronous counters

If the "clock" pulses are applied to all the flip-flops in a counter simultaneously, then such a counter is called as synchronous counter.

2-bit Synchronous up counter

The JA and KA inputs of FF-A are tied to logic 1. So FF-A will work as a toggle flip-flop. The JB and KB inputs are connected to QA.

Logical Diagram



Operation

S.N.	Condition	Operation
1	Initially let both the FFs be in the reset state	QBQA = 00 initially.
2	After 1st negative clock edge	As soon as the first negative clock edge is applied, FF-A will toggle and QA will change from 0 to 1. But at the instant of application of negative clock edge, QA , JB = KB = 0. Hence FF-B will not change its state. So QB will remain 0. QBQA = 01 after the first clock pulse.
3	After 2nd negative clock edge	On the arrival of second negative clock edge, FF-A toggles again and QA changes from 1 to 0. But at this instant QA was 1. So JB = KB= 1 and FF-B will toggle. Hence QB changes from 0 to 1. QBQA = 10 after the second clock pulse.
4	After 3rd negative clock edge	On application of the third falling clock edge, FF-A will toggle from 0 to 1 but there is no change of state for FF-B. QBQA = 11 after the third clock pulse.
5	After 4th negative clock edge	On application of the next clock pulse, QA will change from 1 to 0 as QB will also change from 1 to 0. QBQA = 00 after the fourth clock pulse.

Application of counters

- Frequency counters
- Digital clock
- Time measurement
- A to D converter
- Frequency divider circuits
- Digital triangular wave generator.

6.3. Difference between Synchronous and Asynchronous Counter:

S. No	Synchronous Counter	Asynchronous Counter
1.	In synchronous counter, all flip flops are triggered with same clock simultaneously.	In asynchronous counter, different flip flops are triggered with different clock, not simultaneously.
2.	Synchronous Counter is faster than asynchronous counter in operation.	Asynchronous Counter is slower than synchronous counter in operation.
3.	Synchronous Counter does not produce any decoding errors.	Asynchronous Counter produces decoding error.
4.	Synchronous Counter is also called Parallel Counter.	Asynchronous Counter is also called Serial Counter.
5.	Synchronous Counter designing as well implementation are complex due to increasing the number of states.	Asynchronous Counter designing as well as implementation is very easy.
6.	Synchronous Counter will operate in any desired count sequence.	Asynchronous Counter will operate only in fixed count sequence (UP/DOWN).
7.	Synchronous Counter examples are ring counter Johnson counter etc.	Asynchronous Counter examples are: Ripple UP counter, Ripple DOWN counter.
8.	In synchronous counter, propagation delay is less.	In asynchronous counter, there is high propagation delay.

Classification of Synchronous counters

Depending on the way in which the counting progresses, the synchronous or asynchronous counters are classified as follows −

Up counters
Down counters
Up/Down counters

Decade/BCD Synchronous Counter or MOD 10 Up Counter

Introduction:
The value of N can be different from power of 2. Also, the counting sequence may be random for example some cyclic code (8421, 2423 etc). The following method is applied for designing for mod N and any counting sequence.

Design for Mod-N counter:
The steps for the design are –

Step 1: Decision for number of flip-flops –

Example: If we are designing mod N counter and n number of flip-flops are required then n can be found out by this equation.

$$N <= 2^n$$

Here we are designing Mod-10 counter Therefore, N= 10 and number of Flip flops(n) required is

For n =3, 10<=8, which is false.

For n= 4,10<=16, which is true.

Therefore, number of FF required is 4 for Mod-10 counter.

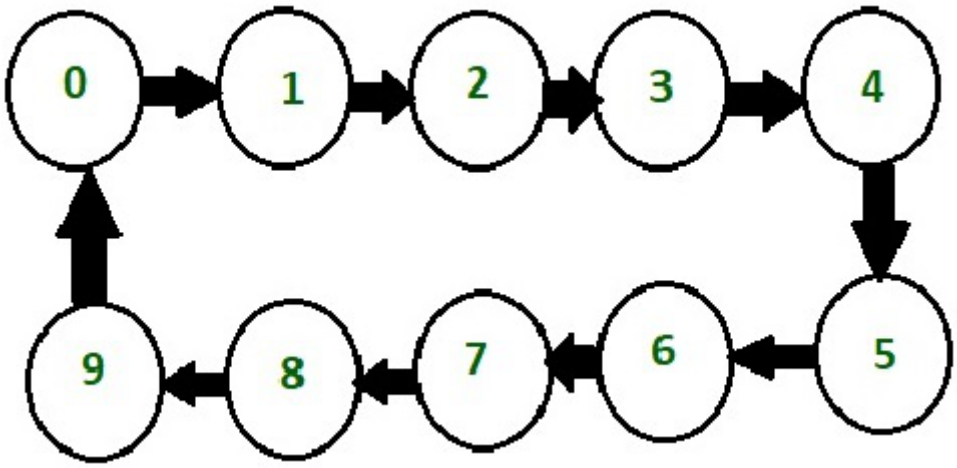
Step 2 : Write excitation table of Flip flops –

Here T FF is used

Previous state(Q_n)	Next state(Q_{n+1})	T
0	0	0
0	1	1
1	0	1
1	1	0

Excitation table of T FF.

Step 3: Draw state diagram and circuit excitation table –



Counting Sequence of Decade/BCD counter

A decade counter is called as mod -10 or divide by 10 counters. It counts from 0 to 9 and again reset to 0. It counts in natural binary sequence. Here 4 T Flip flops are used. It resets after $Q_3\ Q_2\ Q_1\ Q_0 = 1001$.

Circuit excitation table –

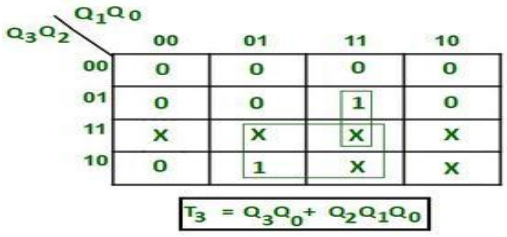
Here $Q_3\ Q_2\ Q_1\ Q_0$ are present states of four flip-flops and $Q^*_3\ Q^*_2\ Q^*_1\ Q^*_0$ is next counting state of 4 Flip flops. If there is a transition in current state i.e if Q3 value changes from 0 to 1 or 1 to 0 then there's corresponding T(toggle) bit is written as 1 otherwise 0.

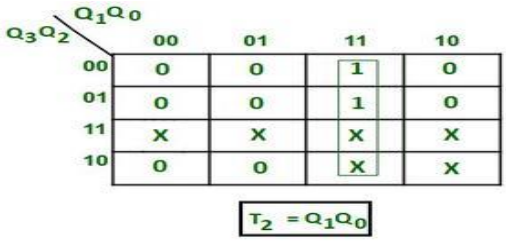
Q_3	Q_2	Q_1	Q_0	Q^*_3	Q^*_2	Q^*_1	Q^*_0	T_3	T_2	T_1	T_0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	1
0	0	1	1	0	1	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	1
0	1	0	1	0	1	1	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	1
0	1	1	1	1	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	1
1	0	0	1	0	0	0	1	1	0	0	1

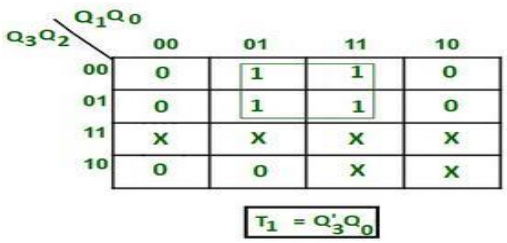
Circuit excitation table.

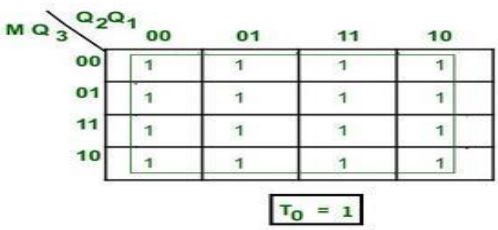
Step 4: Create Karnaugh map for each FF input in terms of flip-flop outputs as the input variable –

Simplify the K map –







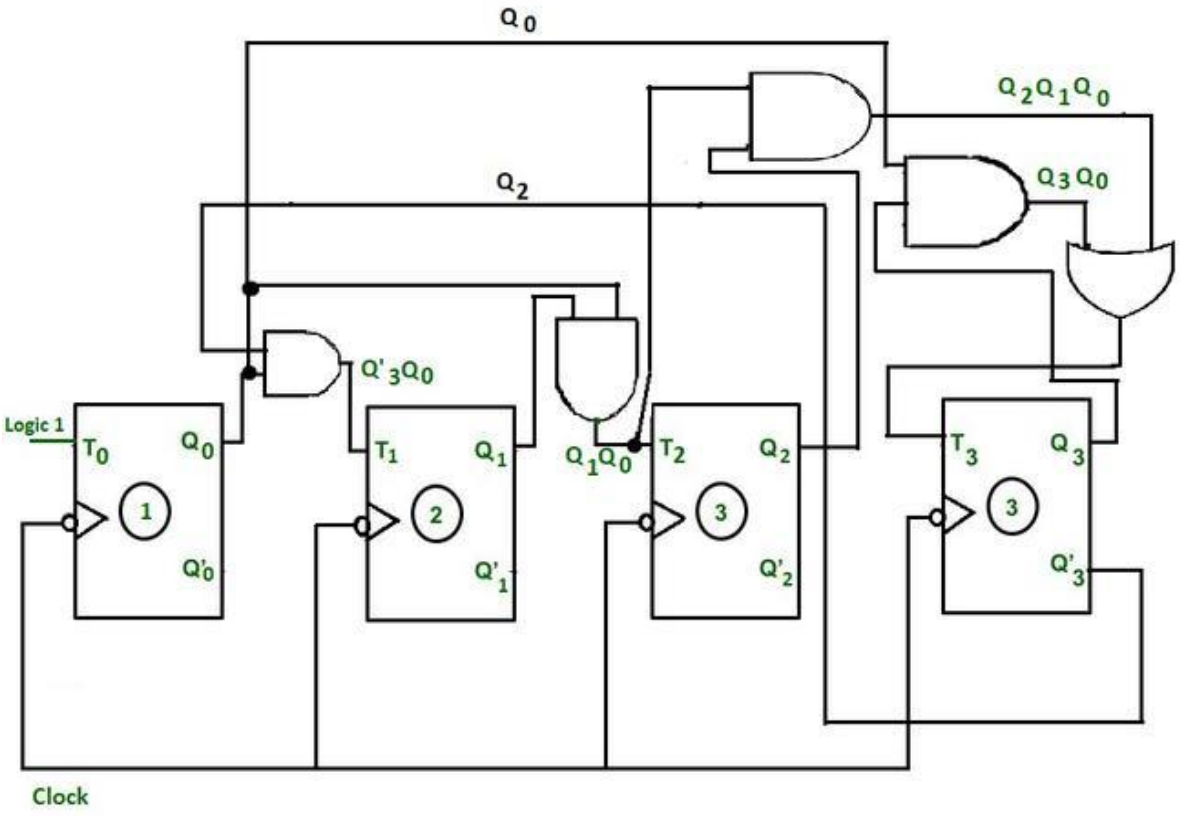


K map for finding minimal expressions.

Step 5: Create circuit diagram –

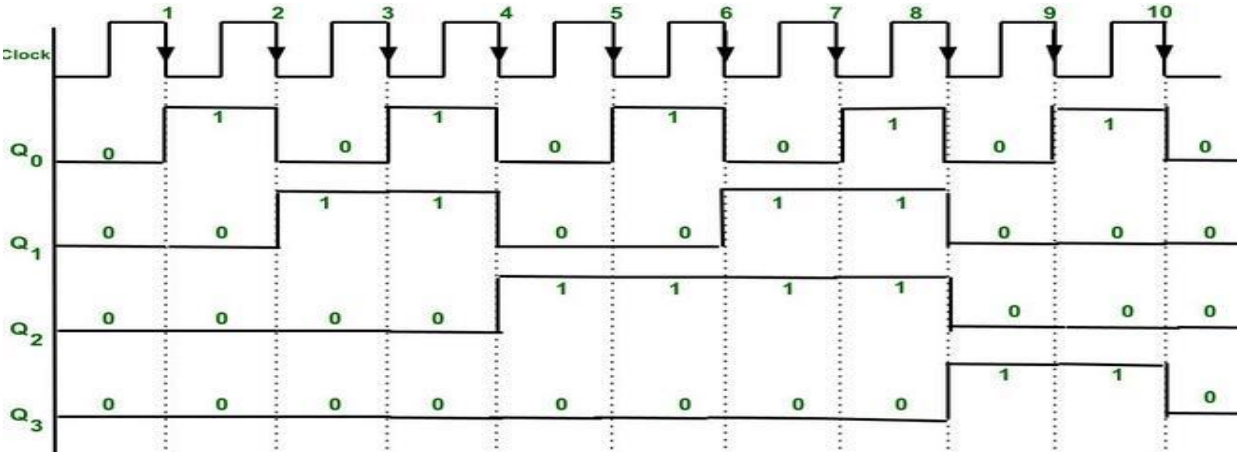
Here negative edge triggered clock is used for toggling purpose.

- The clock is provided to every Flip flop at same instant of time.
- The toggle(T) input is provided to every Flip flop according to the simplified equation of K map.



Circuit diagram

Timing diagram:



Timing diagram of synchronous Decade counter

Explanation:

- Initially Q3 Q2 Q1 Q0 are 0 0 0 0**.**
- The sequence of counter can be verified from the timing diagram. At every falling edge of the clock output Q_0 toggles because T_0 is connected to logic 1**.**
- T_1 becomes 1 only when expression T1 = Q'_3Q_0 becomes 1 also if clock falling edge occurs (because there is negative edge triggering) then the output state of T_1 i.e., Q_1 will change.
- T_2 becomes 1 only when expression T2 = Q_1Q_0 becomes 1 also if clock falling edge occurs then the output state Q2 will change.
- T_3 becomes 1 only when expression T1 = $Q_3Q_0 + Q_2Q_1Q_0$ resultant becomes 1 also if clock falling edge occurs (because there is negative edge triggering) then the state of Q3 will change.
- We get Output as Q_3(MSB) Q_2 Q_1 Q_0(LSB).
- After 10th falling edge the output state of all the FFs again becomes 0 0 0 0

Ring Counter:

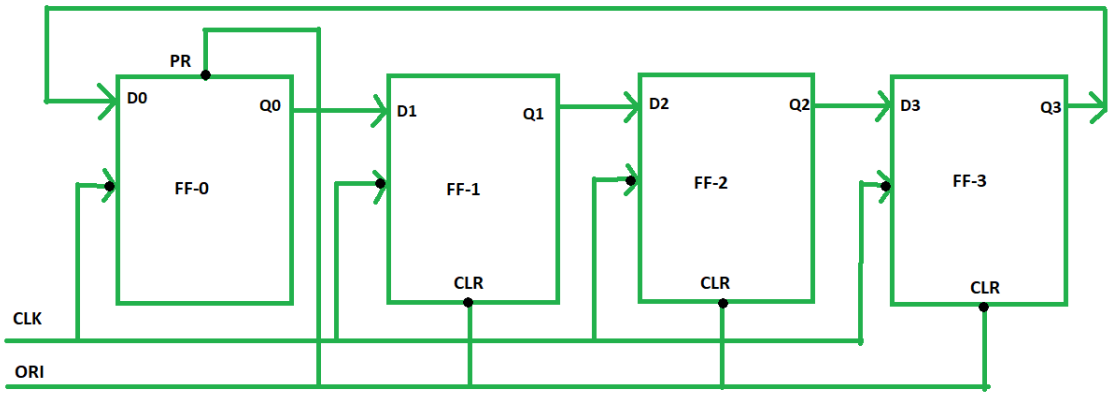
- A ring counter is a special type of application of the Serial IN Serial OUT Shift register.
- The only difference between the shift registers and the ring counter is that the last flip flop outcome is taken as the output in the shift register.
- But in the ring counter, this outcome is passed to the first flip flop as an input. All of the remaining things in the ring counter are the same as the shift register.

In the Ring counter

No. of states in Ring counter = No. of flip-flop used

- Below is the block diagram of the 4-bit ring counter.
- Here, we use 4 D flip flop The same clock pulse is passed to the clock input of all the flip flops as a synchronous counter.
- The Overriding input (ORI) is used to design this circuit.

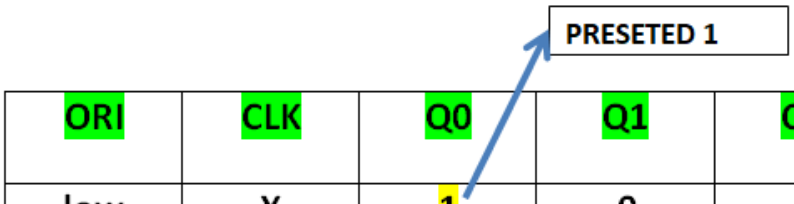
The Overriding input is used as **clear** and **pre-set.**



Ring Counter

In this diagram, we can see that the clock pulse (CLK) is applied to all the flip-flops simultaneously. Therefore, it is a Synchronous Counter. Also, here we use Overriding input (ORI) for each flip-flop. Preset (PR) and Clear (CLR) are used as ORI. When PR is 0, then the output is 1. And when CLR is 0, then the output is 0. Both PR and CLR are active low signal that always works in value 0.

PR = 0, Q = 1
CLR = 0, Q = 0
These two values are always fixed. They are independent of the value of input D and the Clock pulse (CLK). Working – Here, ORI is connected to Preset (PR) in FF-0 and it is connected to Clear (CLR) in FF-1, FF-2, and FF-3. Thus, output Q = 1 is generated at FF-0, and the rest of the flip-flop generates output Q = 0. This output Q = 1 at FF-0 is known as Pre-set 1 which is used to form the ring in the Ring Counter.

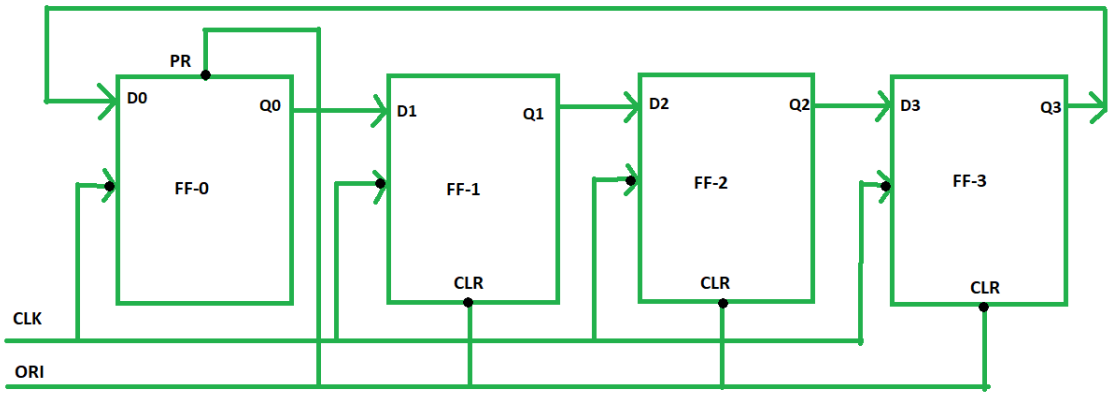


ORI	CLK	Q0	Q1	Q2	Q3
low	X	1	0	0	0
high	low	0	1	0	0
high	low	0	0	1	0
high	low	0	0	0	1
high	low	1	0	0	0

This Preseted 1 is generated by making ORI low and that time Clock (CLK) becomes don't care. After that ORI is made to high and apply low clock pulse signal as the Clock (CLK) is negative edge triggered. After that, at each clock pulse, the Preseted 1 is shifted to the next flip-flop and thus forms a Ring. From the above table, we can say that there are 4 states in a 4-bit Ring Counter.
4 states are:
1 0 0 0
0 1 0 0
0 0 1 0
0 0 0 1
In this way can design a 4-bit Ring Counter using four D flip-flops.
Types of Ring Counter:
There are two types of Ring Counter:

1. Straight Ring Counter:
It is also known as One hot Counter.
In this counter, the output of the last flip-flop is connected to the input of the first flip-flop.

The main point of this Counter is that it circulates a single one (or zero) bit around the ring.

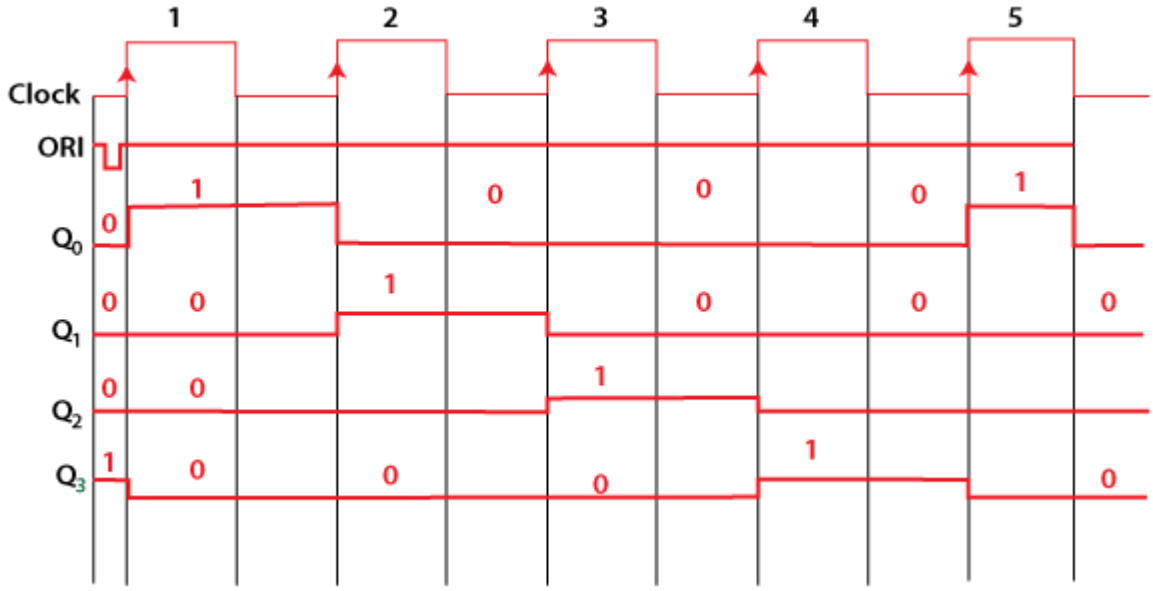


Straight Ring Counter

Here, we use Preset (PR) in the first flip-flop and Clock (CLK) for the last three flip-flops.

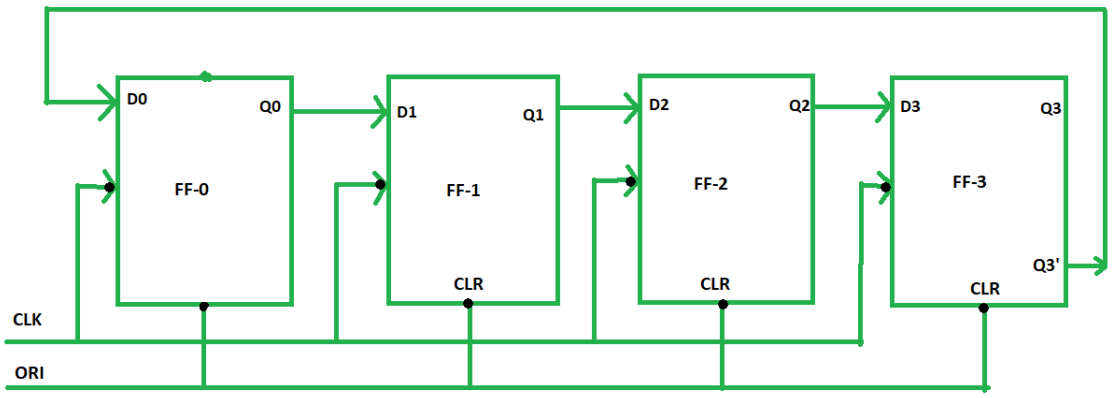
ORI	Clk	Q_0	Q_1	Q_2	Q_3
Low	X	1	0	0	0
High	Low	0	1	0	0
High	Low	0	0	1	0
High	Low	0	0	0	1
High	Low	1	0	0	0

Waveform/Timing Diagram/Signal Diagram:



2. Johnson Counter (Switch Tail/Twisted Ring Counter):

- The Johnson counter is similar to the Ring counter.
- The only difference between the Johnson counter and the ring counter is that the outcome of the last flip flop is passed to the first flip flop as an input.
- But in Johnson counter, the inverted outcome Q' of the last flip flop is passed as an input.The remaining work of the Johnson counter is the same as a ring counter.
- The Johnson counter is also referred to as the Creeping counter.



Twisted Ring Counter

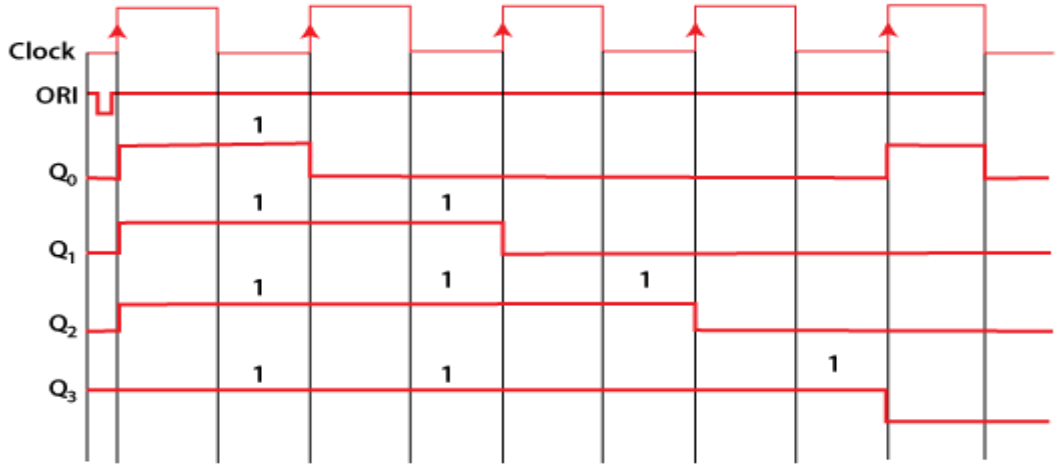
4-bit Johnson counter

CP	Q0	Q1	Q2	Q3
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	1	1	1

The above table state that:

- The counter produces the output 0000 when there is no clock input passed (0).
- The counter produces the output 1000 when the 1st clock pulse is passed to the flip flops.
- The counter produces the output 1100 when the 2nd clock pulse is passed to the flip flops.
- The counter produces the output 1110 when the 3rd clock pulse is passed to the flip flops.
- The counter produces the output 1111 when the 4th clock pulse is passed to the flip flops.
- The counter produces the output 0111 when the 5th clock pulse is passed to the flip flops.
- The counter produces the output 0011 when the 6th clock pulse is passed to the flip flops.
- The counter produces the output 0001 when the 7th clock pulse is passed to the flip flops.

Timing diagram



Advantages

- The number of flip flops in the Johnson counter is equal to the number of flip flops in the ring counter, and the Johnson counter counts twice the number of states the ring counter can count.
- The Johnson counter can also be designed by using D or JK flip flop.
- The data is count in a continuous loop in the Johnson ring counter.
- The circuit of the Johnson counter is self-decoding.

Disadvantages

- The Johnson counter is not able to count the states in a binary sequence.
- In the Johnson counter, the unutilized states are greater than the states being utilized.
- The number of flip flops is equal to one half of the number of timing signals.
- It is possible to design the Johnson counter for any number of timing sequences.

Design of 3-bit UP synchronous Counter/MOD-8 UP Counter:

Steps to follow while designing any synchronous counter:

1. First find out the number of flip flops:
    - N number of Flip flop(FF) required for N bit counter.
    - For 3-bit counter we require 3 FF.
    - Maximum count = 2n-1, where n is a number of bits.
    - For n= 3, Maximum count = 7.
2. Find the type of Flip flop to use, check the question properly, it may be T, D or JK.
3. Draw the State diagram
4. Draw the Circuit excitation table
5. Find the expression using K-maps
6. Draw the Circuit diagram
7. Draw the waveform

No need to explain the process but write the figure and steps clearly while solving the synchronous counter problem.

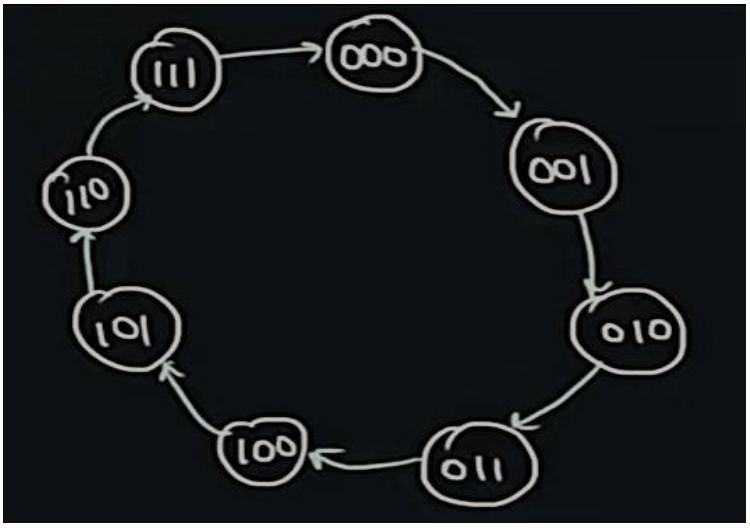


Fig: State diagram

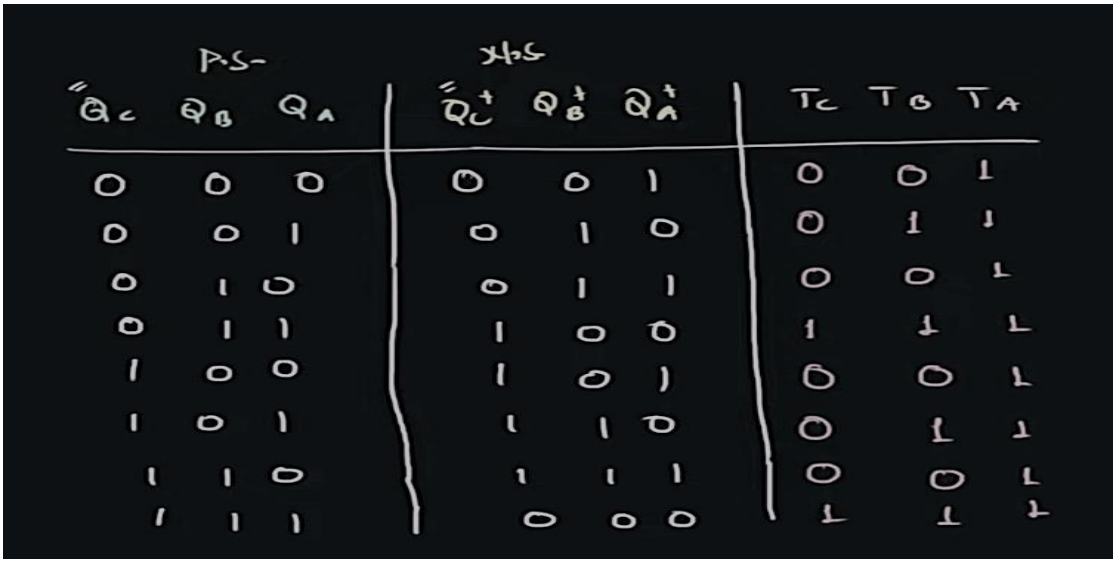
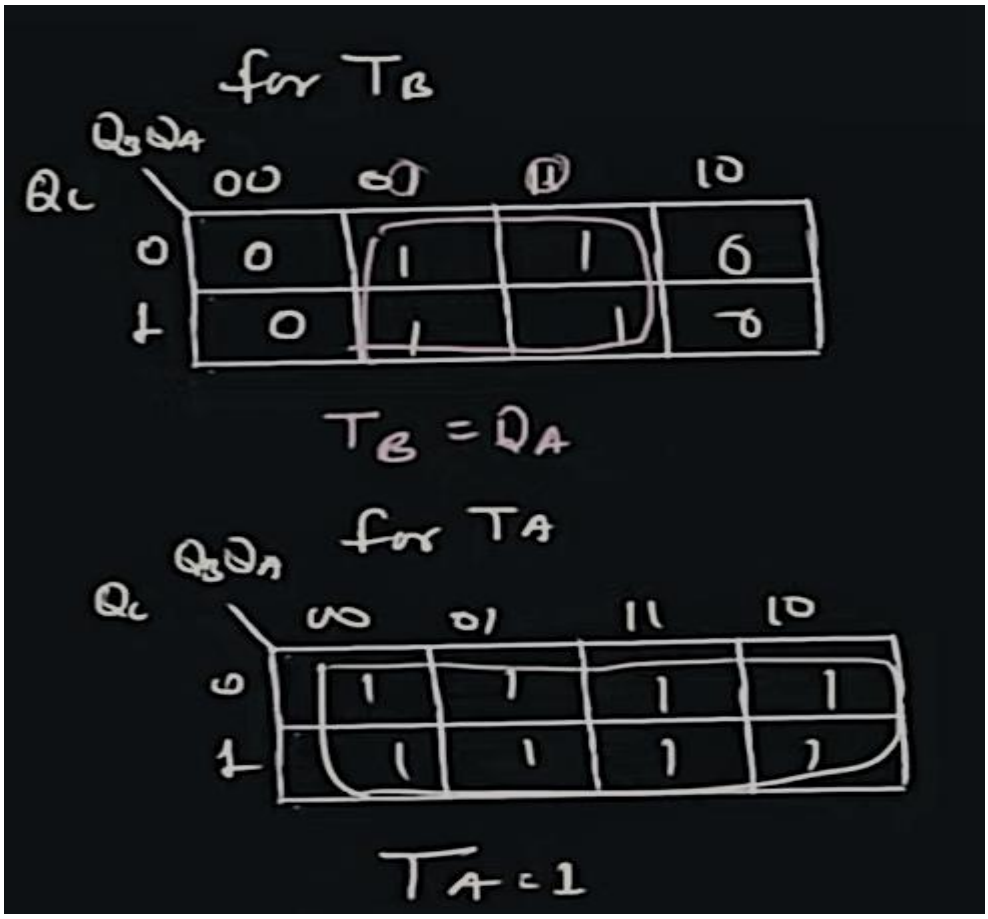
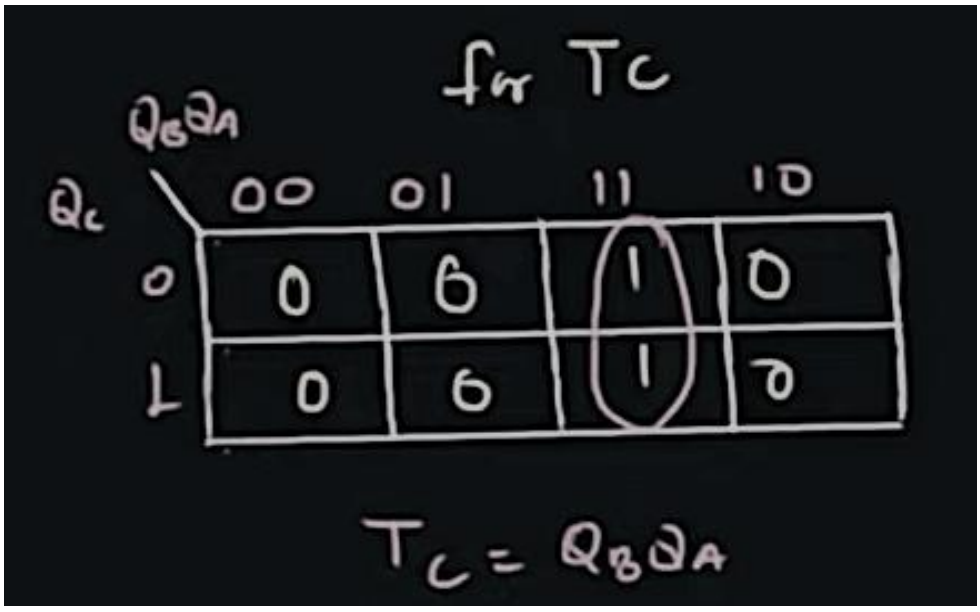


Fig: State table

K-Maps for Ta, Tb and Tc:





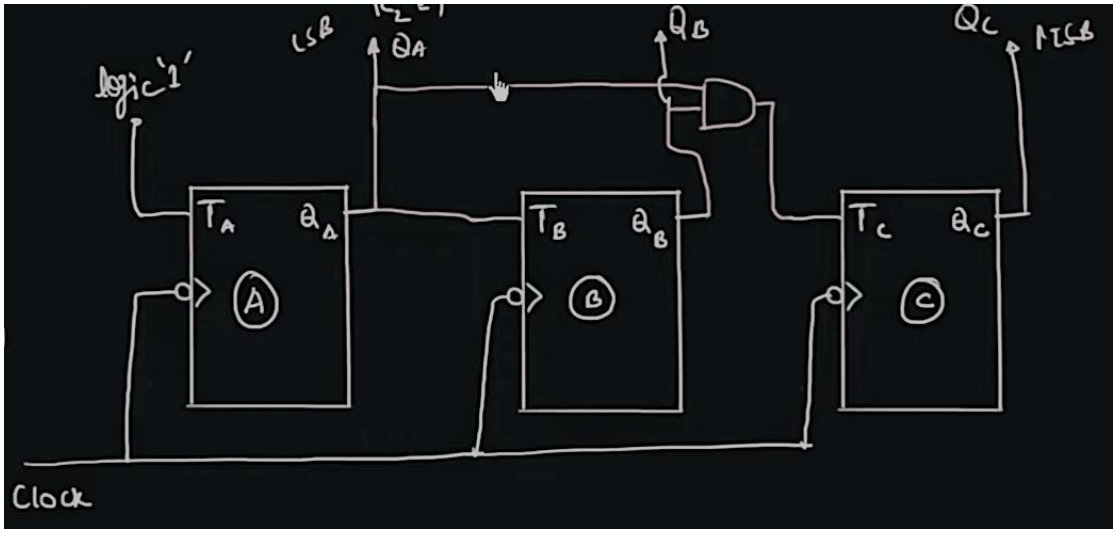


Fig: Logic Diagram

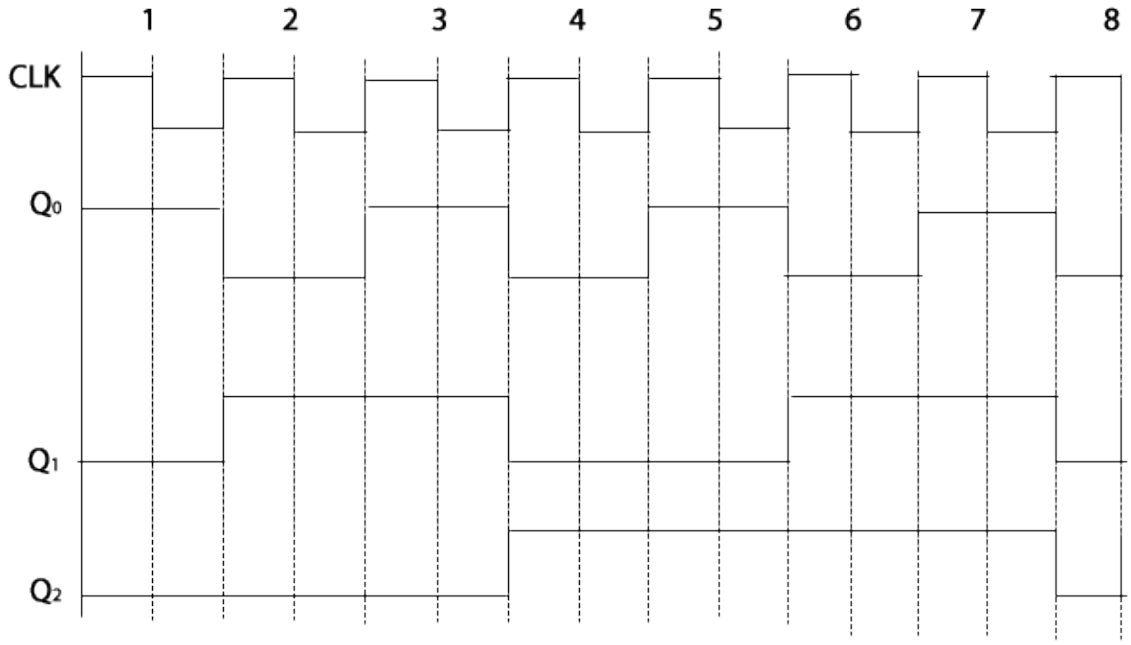


Fig: Waveform/Timing Diagram

Note:

Q_0, Q_1,Q_2 are Q_A,Q_B,Q_C Respectively.

Design of 3-bit Down Synchronous Counter/MOD-8 Down Counter:

- In synchronous counter clock is provided to all the flip-flops simultaneously.
- Circuit becomes complex as the number of states increases.
- Speed is high.

Design: The steps involve in design are

1. **Decide the number of Flip flops –**
   - N number of Flip Flop (FF) required for N bit counter.
   - For 3-bit counter we require 3 FF.
   - Maximum count = 2n-1, where n is a number of bits.
   - For n= 3, Maximum count = 7.
   - Here T FF is used.
2. **Remember the Excitation table of T flip flop**
3. **Draw State diagram and circuit excitation table –**
   Number of states = 2n, where n is number of bits.

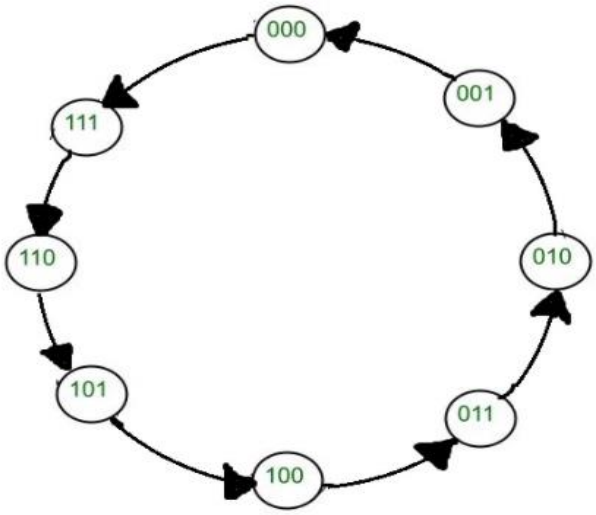


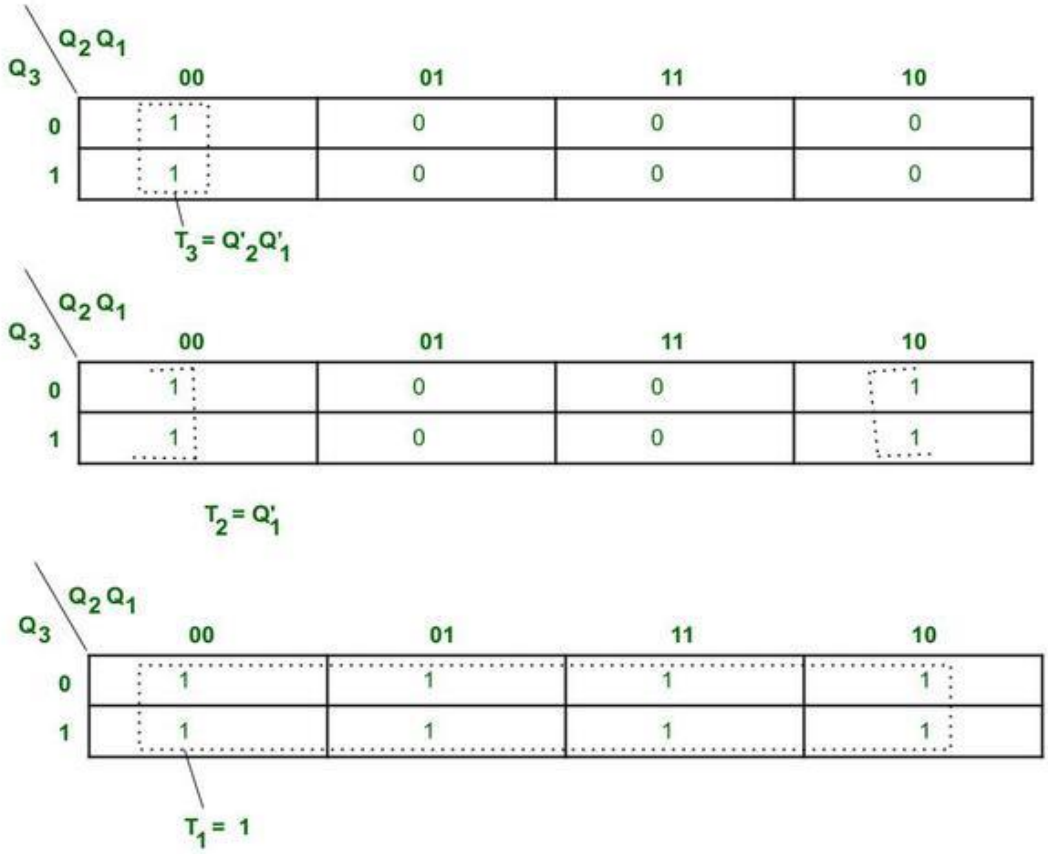
Fig: State Diagram

Previous state			Next state					
Q_3	Q_2	Q_1	Q^*_3	Q^*_2	Q^*_1	T_3	T_2	T_1
0	0	0	1	1	1	1	1	1
0	0	1	0	0	0	0	0	1
0	1	0	0	0	1	0	1	1
0	1	1	0	1	0	0	0	1
1	0	0	0	1	1	1	1	1
1	0	1	1	0	0	0	0	1
1	1	0	1	0	1	0	1	1
1	1	1	1	1	0	0	0	1

Fig: Circuit Excitation Table

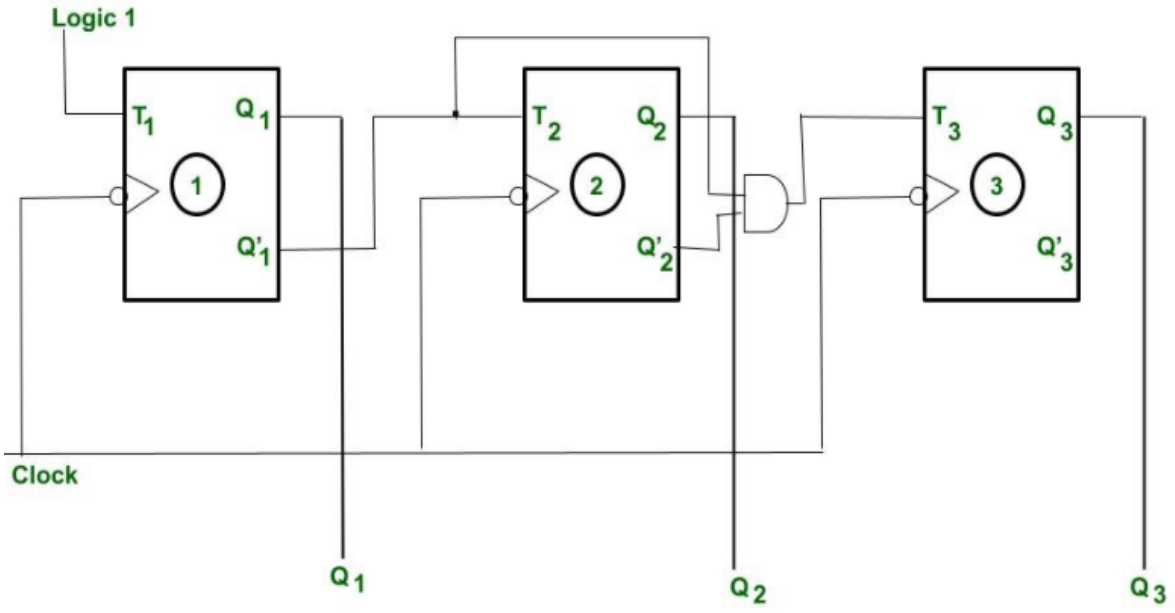
- Here T = 1, then there is output state (next state changes from previous state) changes i.e. Q changes from 0 to 1 or 1 to 0
- T= 0 then, there is no state output state changes i.e Q remains same

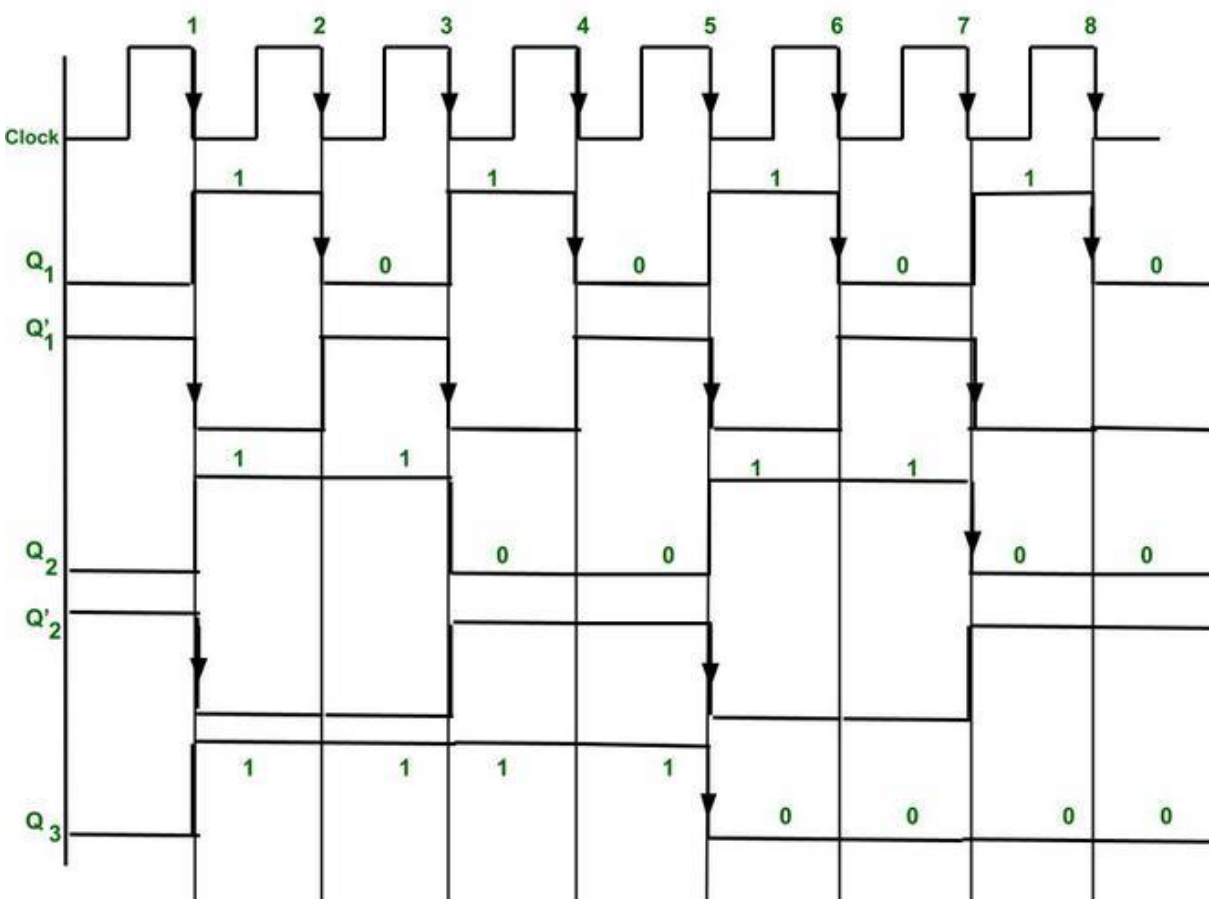
4. Find simplified equation using k map –



K map for 3-bit synchronous down counter

5. Create circuit diagram –





Timing diagram of 3-bit synchronous Down counter.

Explanation: **[No need to write this in examination, this is for understanding purpose only]**

Here -ve edge triggered clock is used for toggling purpose.

Previous state(Q_n)	T	Next state(Q_{n+1})
0	0	0
0	1	1
1	0	1
1	1	0

As we see from characteristics table when T = 1, then toggling takes place and T =0 then it stores the output state.

Initially Q3 = 0, Q2= 0, Q1= 0.

In simplified equation of K map we get T1 = 1, therefore Flip flop 1 output Q1 is toggle for every negative edge(because clock is negative edge triggered). Flip-flop(FF) 2 toggle input(T2) is connected to Q'1. Therefore, Flip Flop 2 output state Q2 is toggle only when there is clock falling edge (i.e -ve edge triggering) and Q'1 =1.

Similarly, Flip flop 3 toggle input(T) is connected to Q'2 and Q'1. Therefore, Flip flop 3 output is toggle when there is clock falling edge and Q'2=1 and Q'1 = 1 .(as you can see from timing diagram)
Therefore, we get output (as down counting Q3(MSB) Q2 Q1(LSB) after 8th -ve edge triggered clock the output of the three Flip flops again becomes Q3 = 0, Q2 = 0, Q1 =0.
We get output (state changes) after every -ve edge clock pulse.
By 3 Flip flop we get output as 23-1= 7 to 0.

Design of Synchronous 3 bit Up/Down Counter:

1. Decide the number and type of FF –

- Here we are performing 3 bit or mod-8 Up or Down counting, so 3 Flip Flops are required, which can count up to 23-1 = 7.
- Here T Flip Flop is used.

2. **Write excitation table of Flip Flop –**

Previous state(Q_n)	Next state(Q_{n+1})	T
0	0	0
0	1	1
1	0	1
1	1	0

Excitation table of T FF

3. Decision for Mode control input M –

- When M=0 ,then the counter will perform up counting.
- When M=1 ,then the counter will perform down counting.

4. Draw the state transition diagram and circuit excitation table –

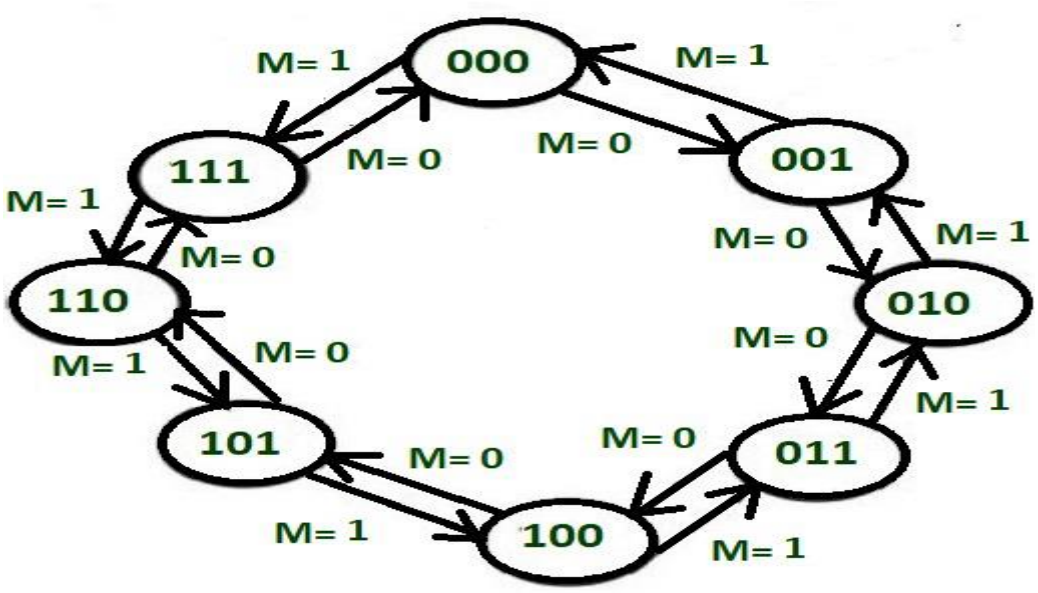


Fig: State transition /diagram for 3 bit up/down counting.

5. Circuit excitation table –

- The circuit excitation table represents the present states of the counting sequence and the next states after the clock pulse is applied and input T of the flip-flops.
- By seeing the transition between the present state and the next state, we can find the input values of 3 Flip Flops using the Flip Flops excitation table.
- The table is designed according to the required counting sequence.

M	Q_3	Q_2	Q_1	Q_3^*	Q_2^*	Q_1^*	T_3	T_2	T_1
0	0	0	0	0	0	1	0	0	1
0	0	0	1	0	1	0	0	1	1
0	0	1	0	0	1	1	0	0	1
0	0	1	1	1	0	0	1	1	1
0	1	0	0	1	0	1	0	0	1
0	1	0	1	1	1	0	0	1	1
0	1	1	0	1	1	1	0	0	1
0	1	1	1	0	0	0	1	1	1
1	0	0	0	1	1	1	1	1	1
1	0	0	1	0	0	0	0	0	1
1	0	1	0	0	0	1	0	1	1
1	0	1	1	0	1	0	0	0	1
1	1	0	0	0	1	1	1	1	1
1	1	0	1	1	0	0	0	0	1
1	1	1	0	1	0	1	0	1	1
1	1	1	1	0	1	0	0	0	1

Fig: Circuit excitation table

If there is a change in the output state of a flip flop (i.e. 0 to 1 or 1 to 0), then the corresponding T value becomes 1 otherwise 0.

6. Find a simplified equation using k map –

MQ3 \ Q2Q1	00	01	11	10
00	0	0	1	0
01	0	0	1	0
11	1	0	0	0
10	1	0	0	0

$T_3 = M'Q_2Q_1 + MQ'_2Q'_1$

MQ3 \ Q2Q1	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	1	0	0	1
10	1	0	0	1

$T_2 = M'Q_1 + MQ'_1$

MQ3 \ Q2Q1	00	01	11	10
00	1	1	1	1
01	1	1	1	1
11	1	1	1	1
10	1	1	1	1

$T_1 = 1$

7. **Create a circuit diagram –**

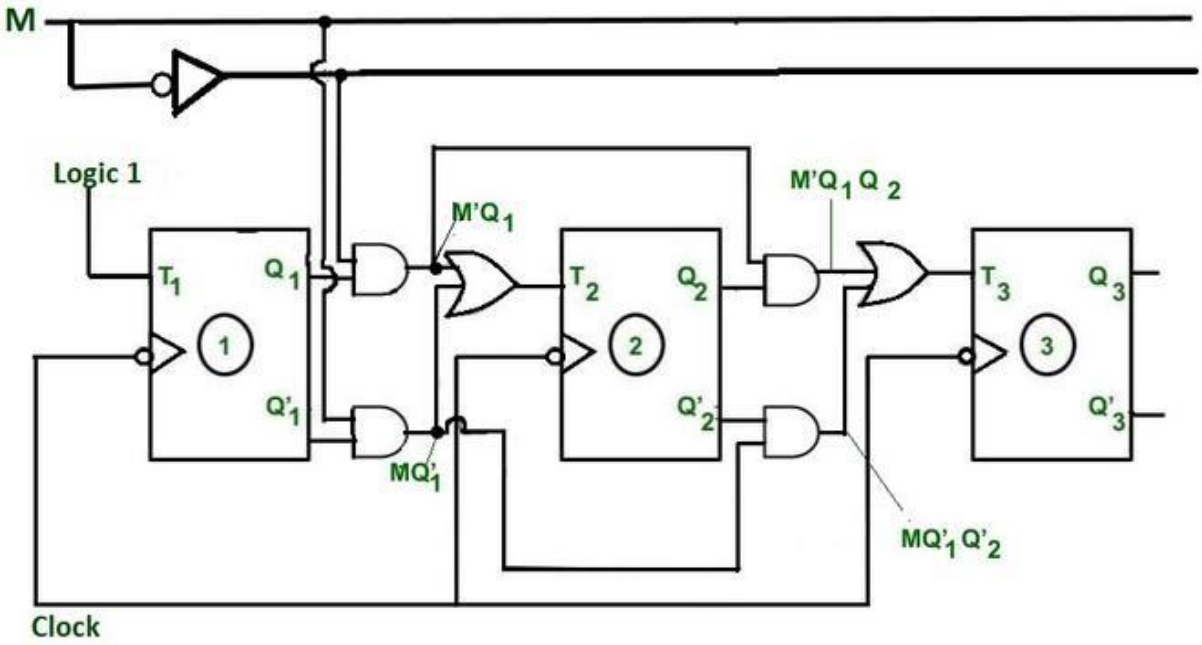


Fig: 3 bit synchronous up/down counter.

8. Timing Diagram –

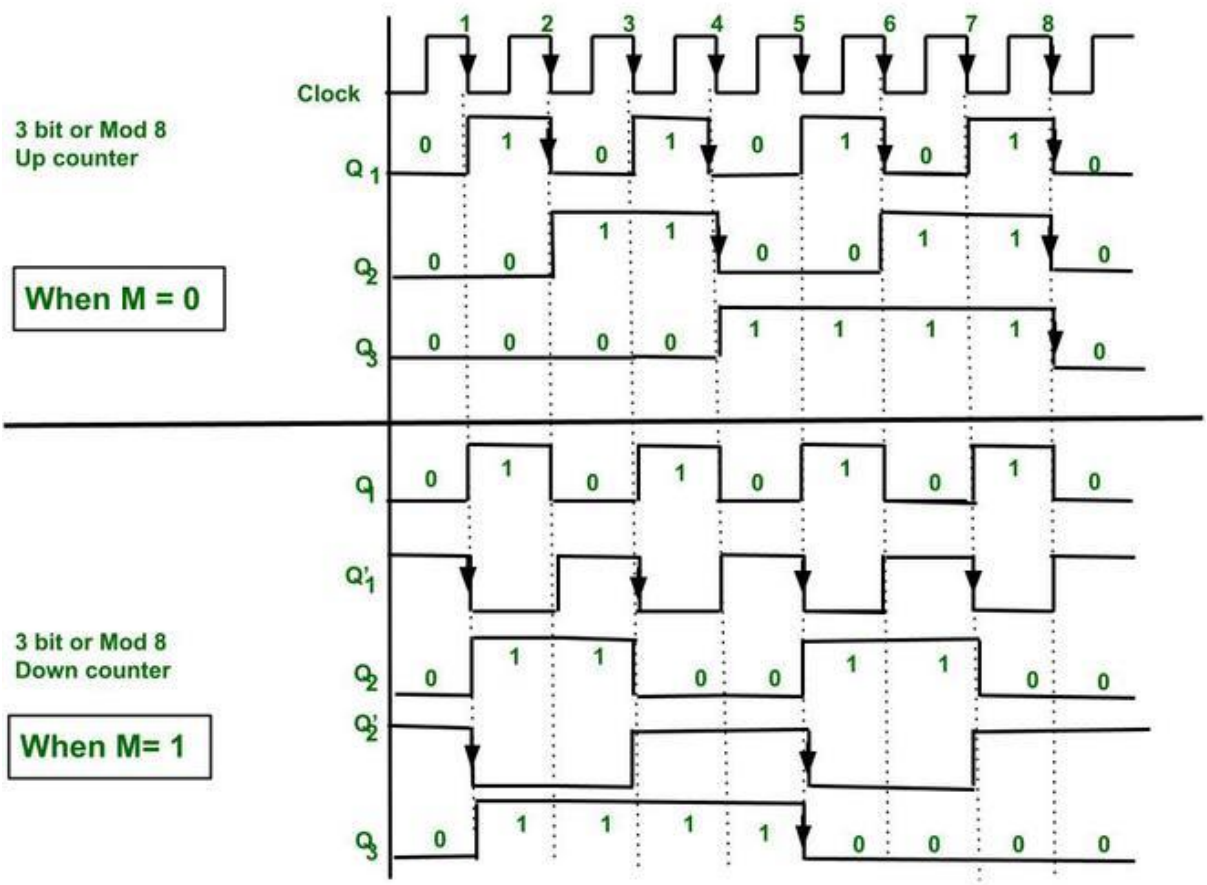


Fig: Timing diagram /Waveform for 3 bit synchronous Up/Down counter

Explanation: [No need to write in examination]

Here -ve edge triggered clock pulse is used for toggling purpose.

Previous state(Q_n)	T	Next state(Q_{n+1})
0	0	0
0	1	1
1	0	1
1	1	0

Characteristics table of T FF

After every falling edge, when T = 1, the output state of Flip Flop will toggle.

Initially Q3 = 0 , Q2= 0 , Q1= 0.

Case 1 : When M=0 ,then M'= 1

T3 = M'Q2Q1 + MQ'2Q'1 = Q2Q1.

T2 = M'Q1 + MQ'1= 1.Q1= Q1.

T1= 1.

Because T1= 1, therefore FF1 output state toggles for every falling edge.

The output state of FF 2 will toggle when Q1 = 1 and the falling edge of the clock pulse occurs.

The output state of FF 3 will toggle only when Q2. Q1= 1 and the falling edge of the clock pulse occurs.
In this way, after every falling edge, state transition takes place and we can get our desired counting sequence.

Case 2 : When M=1 ,then M' =0
T3 = M'Q2Q1+MQ'2Q'1 = Q'2Q'1
T2 = M'Q1+ MQ'1= 1.Q1= Q'1.
T1= 1.
Because T1= 1, therefore FF1 output state toggles for every falling edge.
The output state of FF 2 will toggle when Q'1 = 1 and the falling edge of the clock pulse occurs.
The output state of FF 3 will toggle only when Q'2. Q'1= 1 and the falling edge of the clock pulse occurs.
In this way, after every falling edge, state transition takes place and we can get our desired counting sequence.

Types of Asynchronous /Ripple Counter:

Up Asynchronous Counter
Down Asynchronous Counter
Up/Down Asynchronous Counter

Design of 3-bit Asynchronous /Ripple Up Counter/ MOD-8 Up Asynchronous /Ripple Counter:

This counter consists of $2^3 = 8$ count states (000, 001, 010, 011, 100, 101, 110, 111).
The counter counts the incoming pulses starting from 0 to 7.

No of negative edge of Clock	Qc	Q_B	Q_A
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Fig: Truth Table

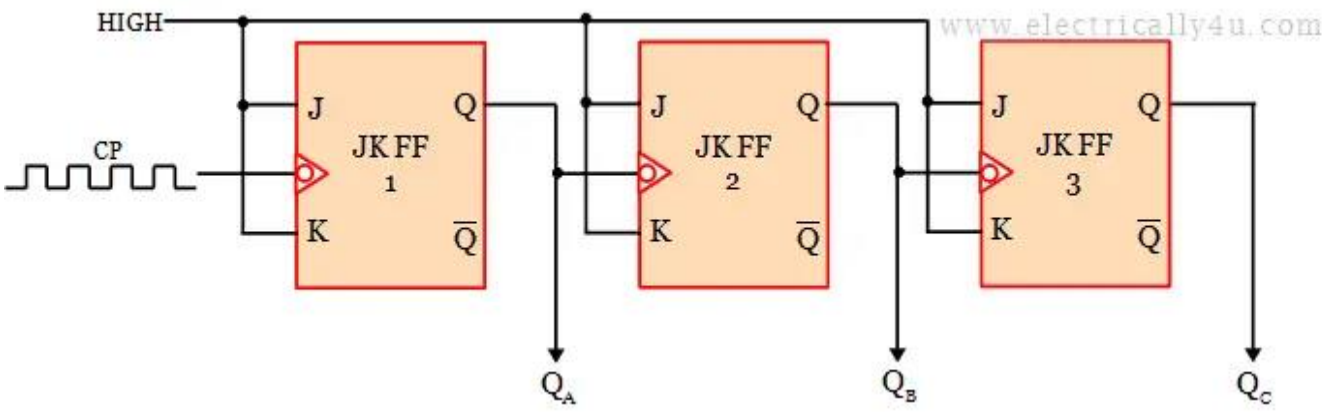


Fig: Circuit Diagram

- The above circuit shows the circuit diagram of a 3-bit asynchronous up counter, in which the clock pulse is given as clock input for JK FF1.
- For the other flip-flops, the clock input is fed from the output of previous flip-flops.
- The clock pulse count is noted at the output of each flip-flop ($Q_CQ_BQ_A$).

Operations:

- At the falling edge of each clock pulse, the output of JF FF1 toggles. For each logic HIGH output ($Q_A = 1$) of JK FF1, at its falling edge, JK FF2 will toggle the output (Q_B). Similarly, for each logic HIGH output ($Q_B = 1$) of JK FF2, JK FF3 will toggle the output (Q_C).

The below figure shows the timing diagram of the 3-bit ripple counter, which shows the change of state of each flip-flop during each clock pulse. In this type, the counter resets to 000, after counting up to 111.

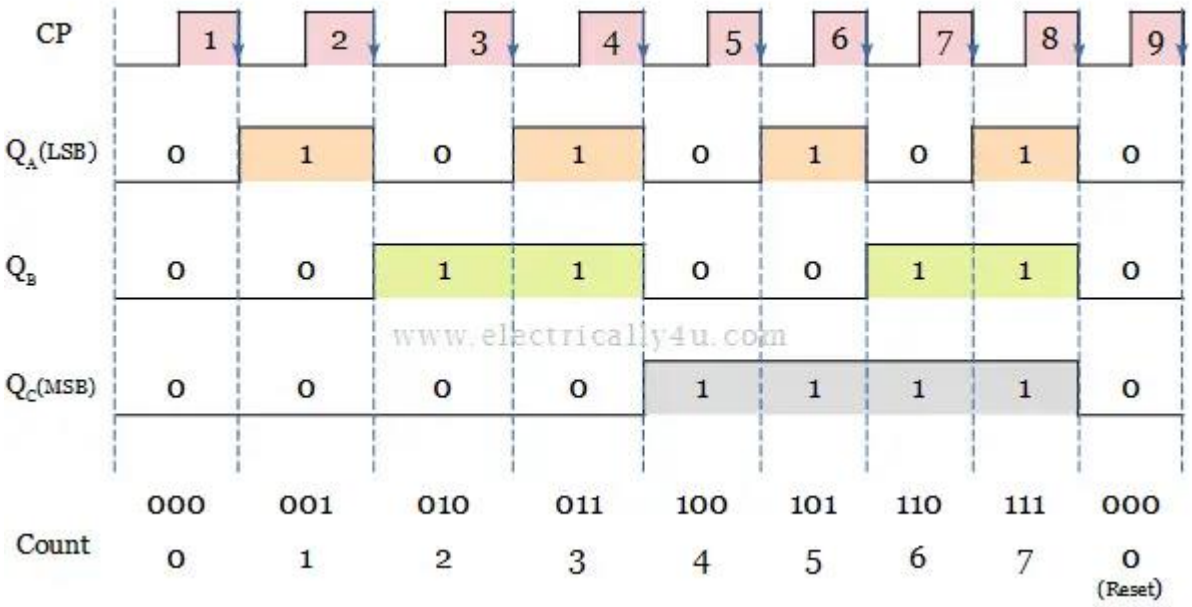


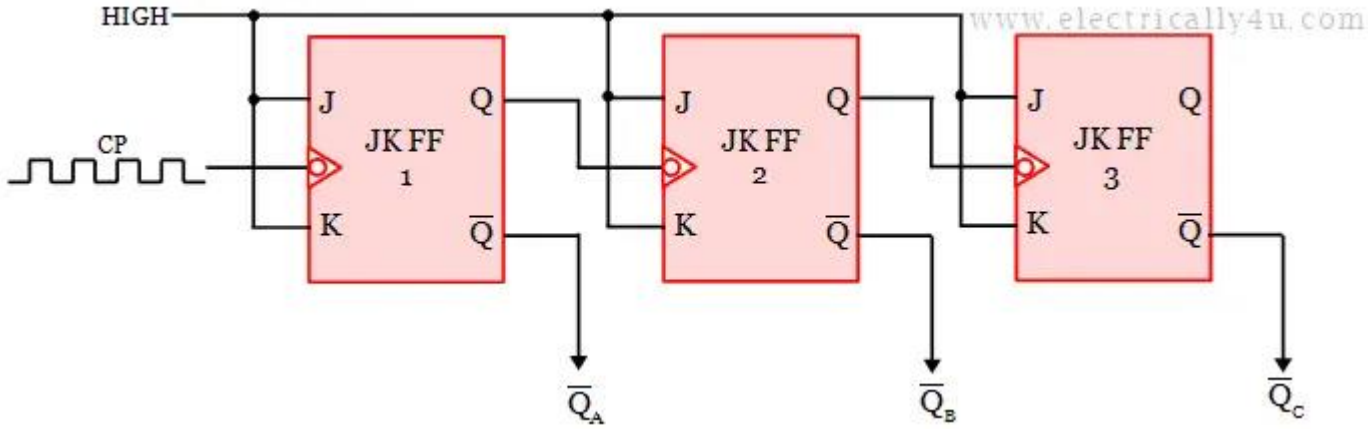
Fig: Timing/Waveform diagram

Design of 3-bit asynchronous/Ripple down counter/MOD-8 Down Asynchronous /Ripple Counter:

- The down counter will count the clock pulses from maximum value to zero.
- In other words, for each clock pulse, the count value is decremented.

No of negative edge of Clock	Q_C	Q_B	Q_A
0	0	0	0
1	1	1	1
2	0	1	1
3	1	0	1
4	0	0	1
5	1	1	0
6	0	1	0
7	1	0	0

The below diagram shows the 3-bit asynchronous down counter.
Since it is a 3-bit counter, 3 negative edge-triggered flip-flops are used.
The clock pulse input is given only to the first flip-flop.
The clock input of the remaining flip-flops is triggered by the Q output of the previous flip-flop.
Since it is down counter, the 3-bit count value is measured from the (Q_C'Q_B' Q_A').



Operation:

- The operation is the same as that of the 3-bit asynchronous up counter.
- If the output is taken at the normal Q output of each flip flop, then it is an up counter. If the output is taken at the complemented output (Q') of each flip flop, it is said to be the down counter.
- The change of state of each flip flop with respect to the clock pulse and the count value is shown in the below timing diagram.

- In this diagram, the output waveform Q_C, Q_B, Q_A represents the normal Q output of JK FF3, JK FF2 and JK FF1 respectively. In the below waveform, Q_C', Q_B', Q_A' represents the complemented output from of JK FF3, JK FF2 and JK FF1 respectively.

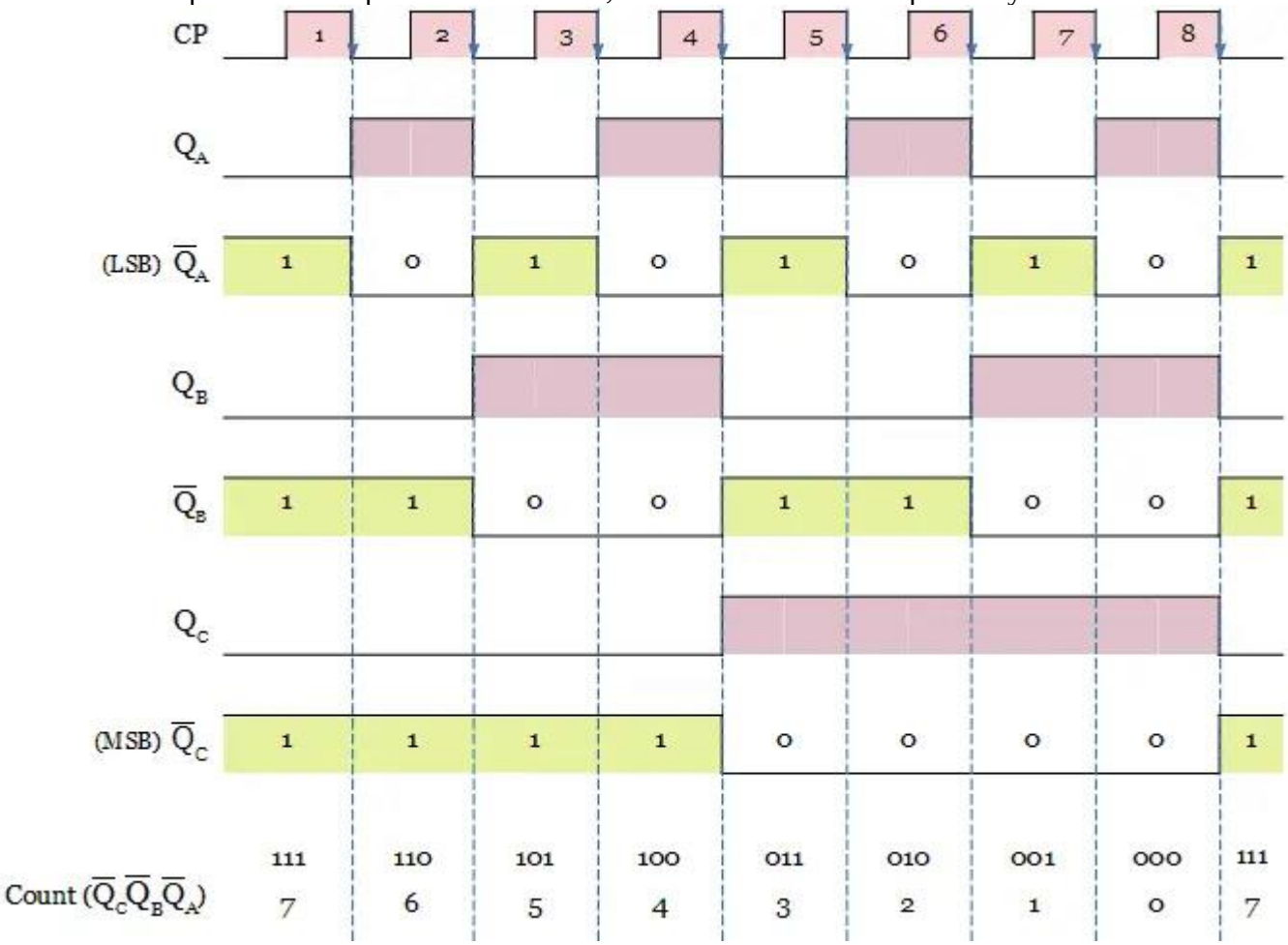


Fig: Timing/Waveform Diagram

The count value can be observed from the complemented outputs. The counter value starts from 111 and decrements its value for each clock pulse. After reaching 000, the counter resets to the maximum value (111) and starts to decrement again for the next clock pulse.

Design of 3 bit Asynchronous up/down counter/Bidirectional Counter:

It is used more than separate up or down counter.

In this a mode control input (say M) is used for selecting up and down mode.

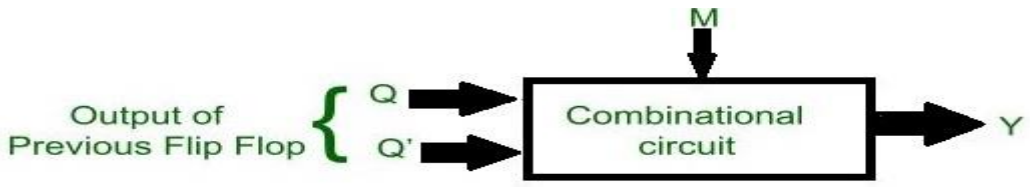
A combinational circuit is required between each pair of flip-flops to decide whether to do up or do down counting.

For n = 3, i.e. for 3-bit counter

Maximum count = $\mathbf{2^n - 1}$ and number of states are $\mathbf{2^n}$.

Steps involve in design are:

Step 1: Decision for Mode control input –



Decision for mode control input

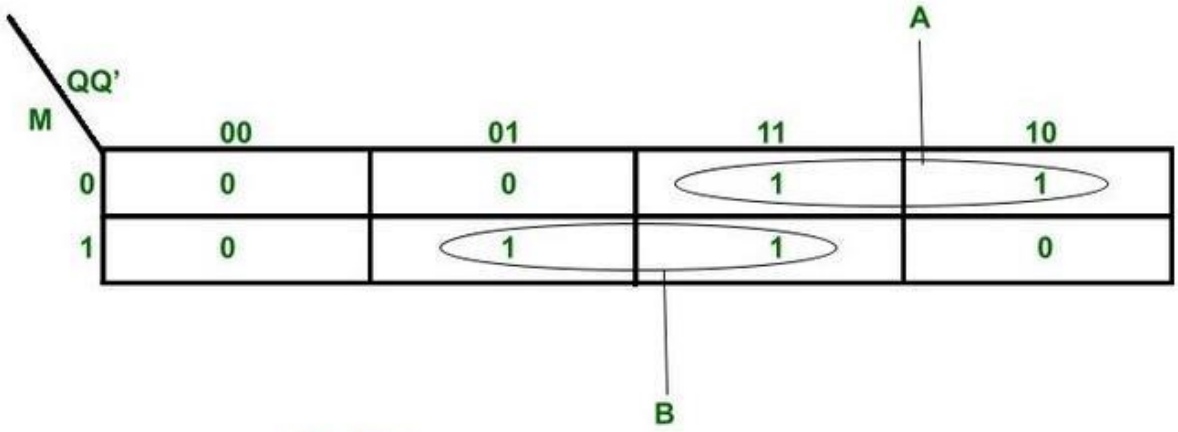
When M = 0, then Y= Q, therefore it will perform Up counting (As discussed above).
When M = 1, then Y= Q' therefore it will perform Down counting (As discussed above).
Combinational circuit is required for deciding mode control (i.e whether counter will perform Up counting or Down counting).
So, the all-possible combinations are –

Y = Q when M= 0
Y = Q' when M=1

M	Q	Q'	Y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Here Y is the output function

K-map for finding output Y that will be given as clock to next FF.

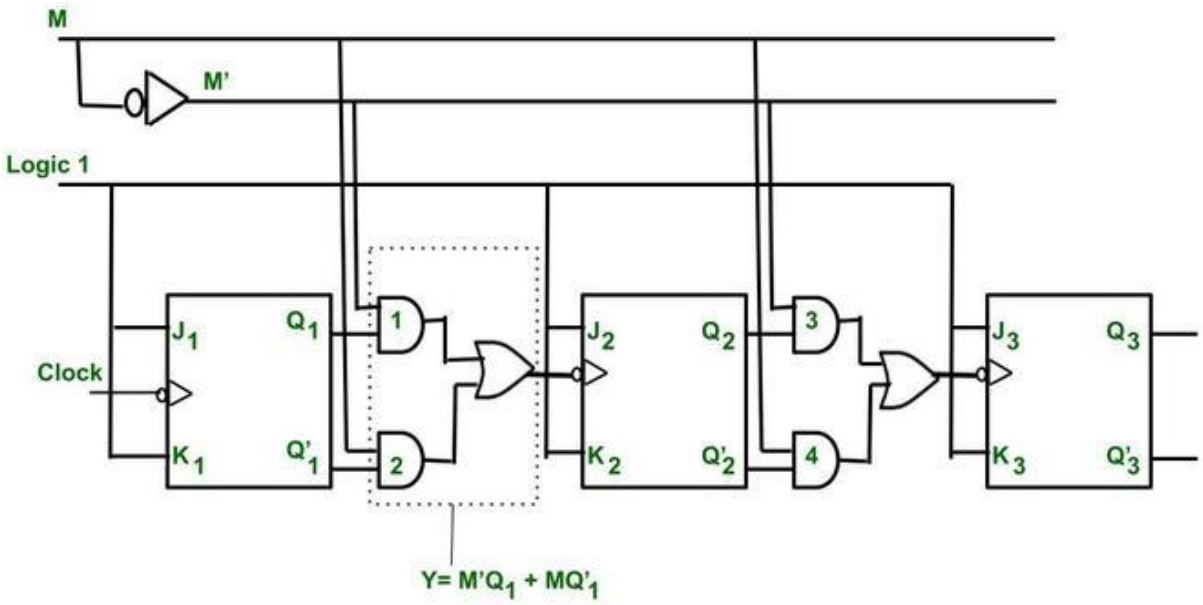


Y = A+B

Y= M'Q + MQ'

K map for finding Y

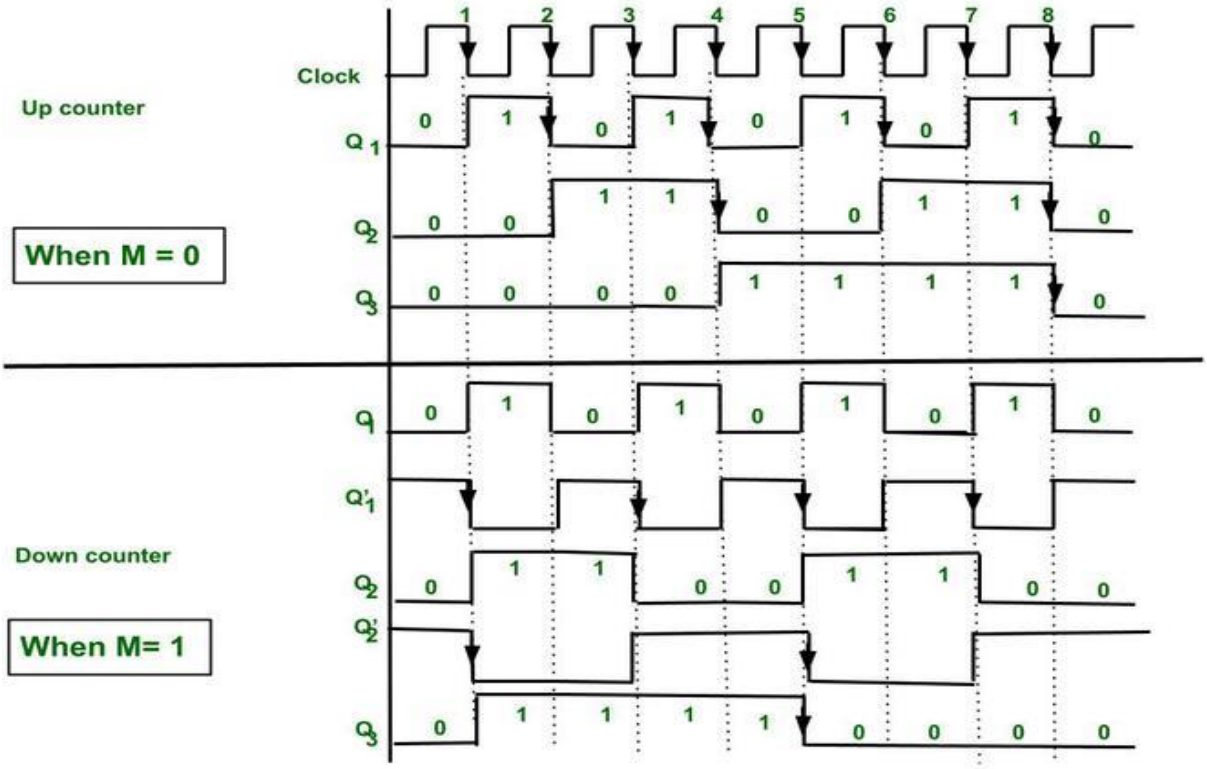
Step 2 : **Insertion of Combinational logic between every pair of FFs –**



Up/Down Counter

Timing diagram:

Initially $Q_3 = 0$, $Q_2 = 0$, $Q_1 = 0$.



Timing diagram for 3 bit asynchronous up/down counter

Case 1 – When M=0, then M' =1.

Put this in Y**= M'Q + MQ'= Q** So Q is acting as clock for next FFs.

Therefore, the counter will act as Up counter.

Explanation of Up counter –

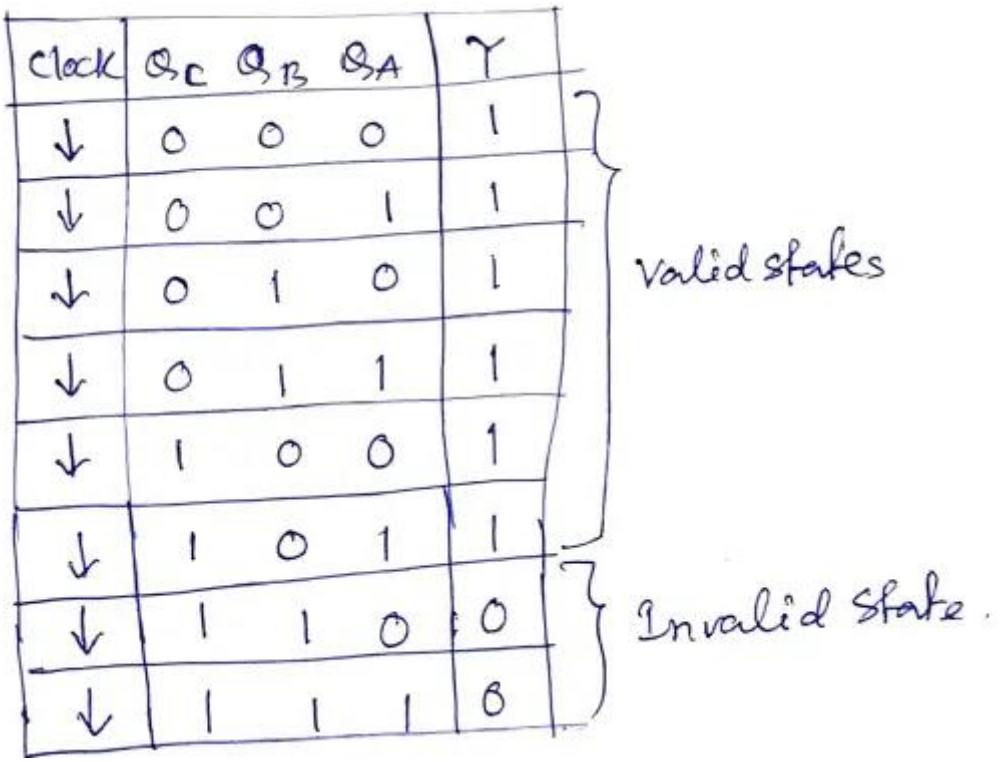
The 1st FF is connected to logic 1. Therefore, it will toggle for every falling edge.
The 2nd FF input is connected to Q_1. Therefore, it changes its state when Q_1= 1 and there is falling edge of clock.
Similarly, 3rd FF is connected to Q_2. Therefore, it changes its state when Q_2= 1 and there is falling edge of clock.
By this we can generate counting states of Up counter.
After every 8th falling edge, the counter is again reaching to state 0 0 0.
Therefore, it is also known as divide by 8 circuit or mod 8 counter.

Case 2 – When M=1, then M' =0.
Put this in **Y= M'Q + MQ'= Q'**. So, Q' is acting as clock for next FFs.
Therefore, the counter will act as Down counter.

Explanation of Down counter –
The 1st FF is connected to logic 1. Therefore, it will toggle for every falling edge.
The 2nd FF input is connected to Q'_1. Therefore, it changes its state when Q'_1= 1 and there is falling edge of clock.
Similarly, 3rd FF is connected to Q'_2. Therefore, it changes its state when Q'_2= 1 and there is falling edge of clock.
By this we can generate counting states of down counter. After every 8th falling edge, the counter is again reaching to state 0 0 0. Therefore, it is also known as divide by 8 circuit or mod 8 counter.

Design of MOD-6 Ripple Counter:

Step1: Truth table

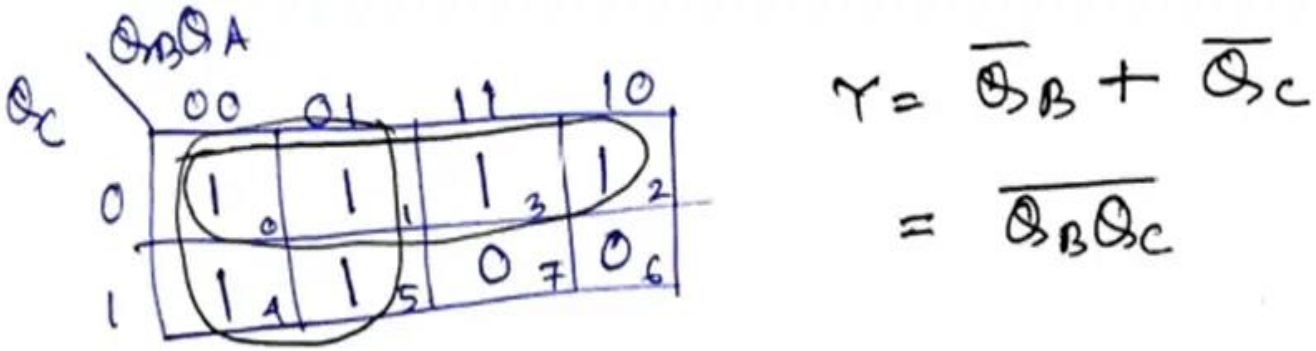


Clock	Q_C	Q_B	Q_A	Y
↓	0	0	0	1
↓	0	0	1	1
↓	0	1	0	1
↓	0	1	1	1
↓	1	0	0	1
↓	1	0	1	1
↓	1	1	0	0
↓	1	1	1	0

Valid states

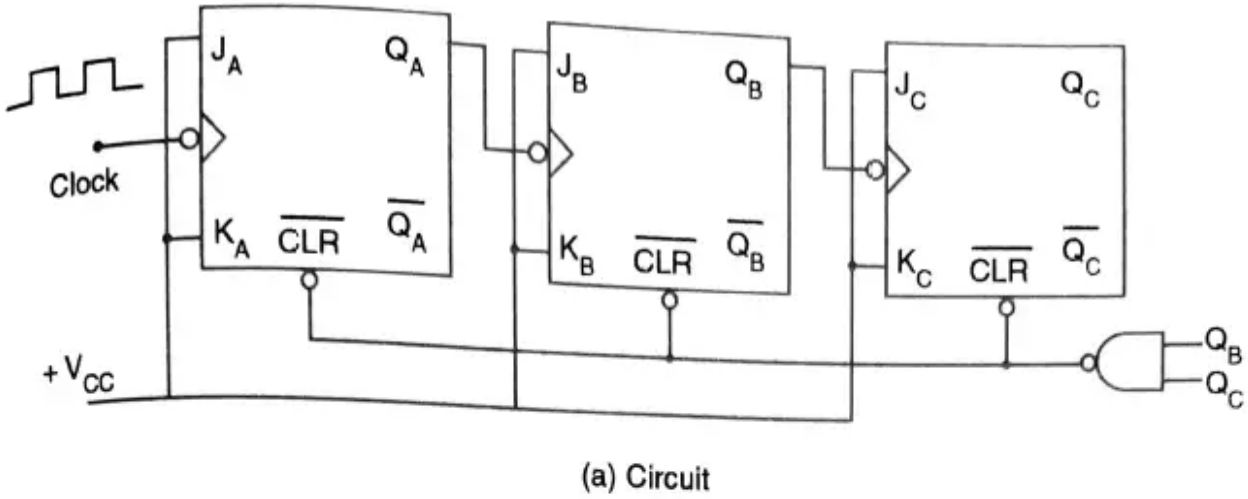
Invalid State.

Step 2: K-map

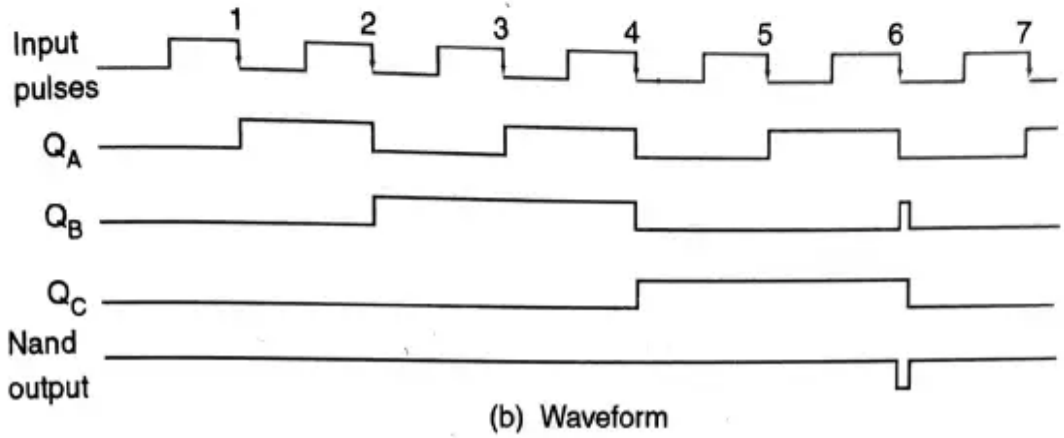


K-Map for resetting logic circuit to 0 after encounting 5 during counting

Step 3: Circuit diagram and Timing Diagram:



(a) Circuit



(b) Waveform