Unit II

Sequential Logic Design

1 Bit Memory Cell :-

- The basic digital memory circuit is known as "Flip-flop.

- It has two stable states namely logic 1 and logic 0 state. We can design it either using NOR gates or NAND gates.

- A flip-flop can be designed by using the fundamental circuit as shown in fig.

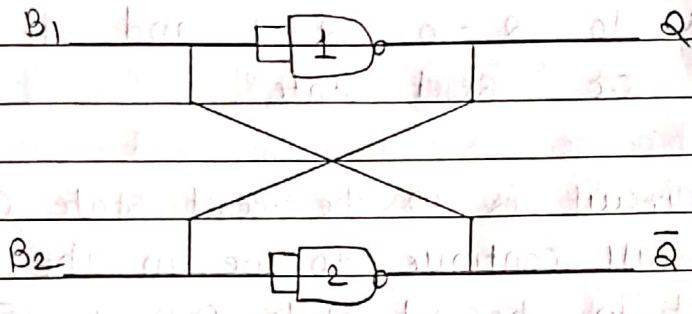


Fig: A cross coupled inverter as memory element.

NAND gates 1 and 2 are basically acting as inverter. Hence this circuit is called a cross coupled inverter output of gate 1 is connected to the input of gate 2 and output of gate 2 is connected to input of gate 1.

Operation :-

- Assume that output of gate 1 i.e $Q = 1$ hence $B_2 = 1$ As $B_2 = 1$, output of gate 2 i.e $\bar{Q} = 0$. This make $B_1 = 0$. Hence Q continues to be equal to 1.

- similarly we can demonstrate that if we start with $Q = 0$ then we end up obtaining $Q = 0$ & $\bar{Q} = 1$.

Conclusion :-

- The outputs of the circuit (Q and $\bar{Q}$) will always be complementary. That means if $Q = 0$ then $\bar{Q} = 1$ and vice versa. They will never be equal $Q = \bar{Q} = 0$ or 1 is an invalid state.

- This circuit has two stable states. one of them corresponds to $Q = 1$, $\bar{Q} = 0$ and it is called as "1 state" or "set state". Whereas the other corresponds to $Q = 0$, $\bar{Q} = 1$ and it is called as "0 state" or "Reset state".

- If the circuit is in the reset state ($Q = 0$, $\bar{Q} = 1$) then it will continue to be in the reset state and if it in the set state ($Q = 1$, $\bar{Q} = 0$) then it will continue to remain in the set state. This property of the circuit shows that it can store 1 bit of digital information. Therefore it is called as a "1 bit memory cell".

o Clocked S-R Flip-flop :-

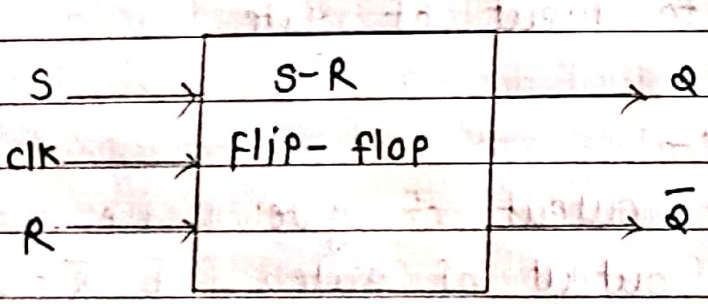
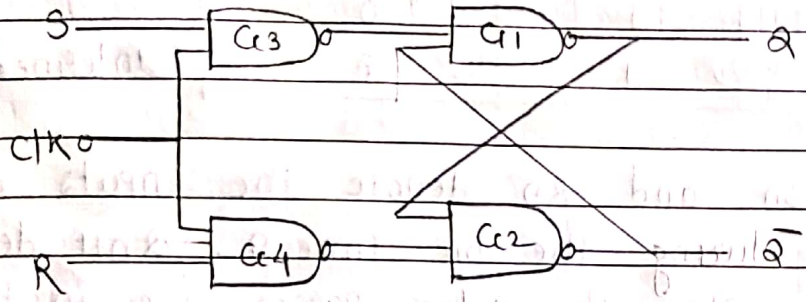


Fig :- Logic symbol of clocked S-R flip-flop

- The logic symbol of clocked S-R flip-flop is given in the fig. It is often required to set or reset the memory cell in synchronism with a train of pulses known as clock, such a circuit is referred to as a clocked set Reset (S-R) flip-flop.



Clocked SR flip-flop

- In this ckt if a clock pulse is present (CLK = 1) its operation is exactly the same as memory cell with provision for entering data.

- On the other hand, when the clock pulse is not present (CLK = 0) the gate G_3 and G_4 are inhibited i.e their outputs are 1 irrespective of the values of S or R. In other words, the circuit responds to the input S and R only when the clock is present.

- Assuming that the inputs do not change during the presence of clock pulse, we can express the operation of flip-flop in the form of truth table.

Table : Truth table of S-R flip-flop :-

Inputs		output	
Sn	Rn	Qn+1	
0	0	Qn	
1	0	1	
0	1	0	
1	1	?	(undefined)

- Here Sn and Rn denote the inputs and Qn the output during the bit time n. Qn+1 denotes the o/p Q after the pulse passes i.e in the bit time n+1.

- If Sn = Rn = 0 and the clock pulse is applied the output at the end of the clock pulse is same as the output before the clock pulse i.e Qn+1 = Qn

- If Sn = Rn = 1 when the clock is present the o/p of gate G_3 and G_4 are both 0. making one of the input of G_1 and G_2 NAND gate 0. Consequently Q and $\bar{Q}$ both with attain logic 1 which is inconsistant with our assumption of complementary outputs.

- Now when the clock pulse has passed away (Clk = 0) the outputs of G_3 & G_4 will rise from 0 to 1. Depending upon the propagation delays of the gates either the stable state $Q_{n+1} = 1$ ($\bar{Q}_{n+1} = 0$) or $Q_{n+1} = 0$ ($\bar{Q}_{n+1} = 1$) will result. That means the state of circuit is "undefined", "intermediate" or "ambiguous".

o J-K Flip-Flop :-

- The uncertainty in the state of an S-R f/f when $S_n = R_n = 1$ (fourth row of truth table) can be eliminated by converting it into a JK flip flop. The data inputs are J and K which are ANDed with $\bar{Q}$ and Q respectively to obtain S and R i/p i.e

$$S = J \cdot \bar{Q} \quad , \quad R = K \cdot Q$$

The logic symbol for J-K flip-flop is given by

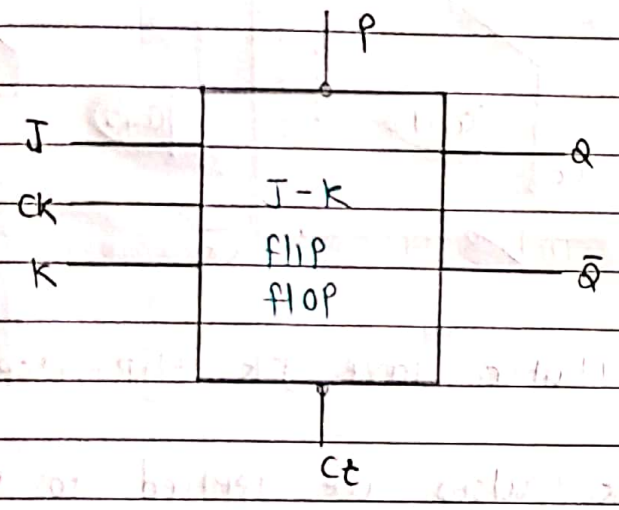


Fig:- Logic Symbol of J-K flip-flop

Table :- Truth table of J-K flip-flop :-

Inputs		output
J_n	K_n	Q_{n+1}
0	0	Q_n
1	0	1
0	1	0
1	1	$\bar{Q}_n$

Master slave (MS) J-K flip-flop :-

- A master slave J-K flip-flop is a cascade of two SR flip-flops with flb from the o/p of the second to the inputs of the first.

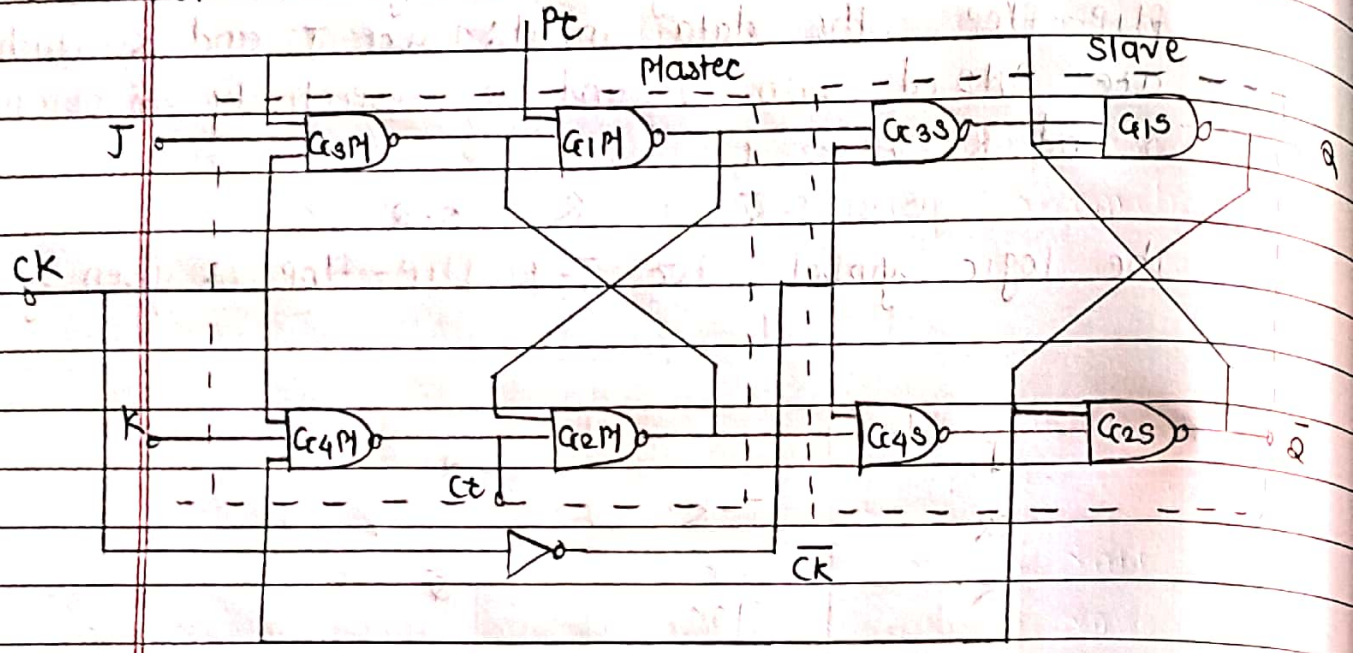


fig :- A Master slave JK flip-flop

Positive clock pulses are applied to the first flip-flop and the clock pulses are inverted before these are applied to the second flip flop.

- When CK = 1, the first flip-flop is enabled & the outputs Q_M and $\bar{Q}_M$ respond to the inputs J and K. At this time, the second flip flop is inhibited because its clock is low ($\because \overline{CK} = 0$). When CK goes Low ($\overline{CK} = 1$), the first flip flop is inhibited and the second flip flop is enabled because now its clock is high ($\because \overline{CK} = 1$). Therefore the outputs Q and $\bar{Q}$ follow the outputs Q_M and $\overline{Q_M}$ respectively.

since the second flip flop simply follows the first one, it is referred to as the slave and the first one as the master. Hence the configuⁿ is referred to as master slave flip flop.

- The Logic symbol for the master slave flip-flop is given in the fig.

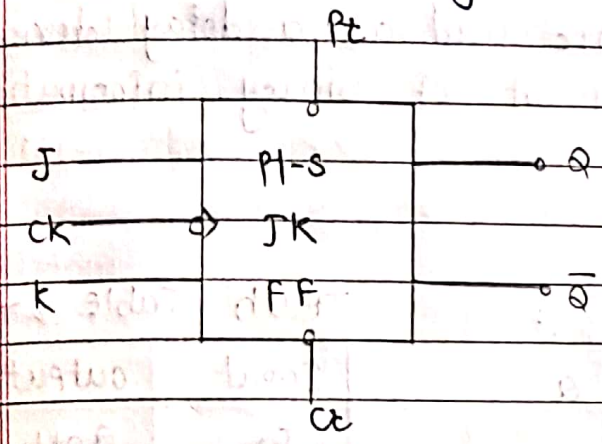


fig :- Logic symbol of Master slave JK ff

o D Flip-Flop :-

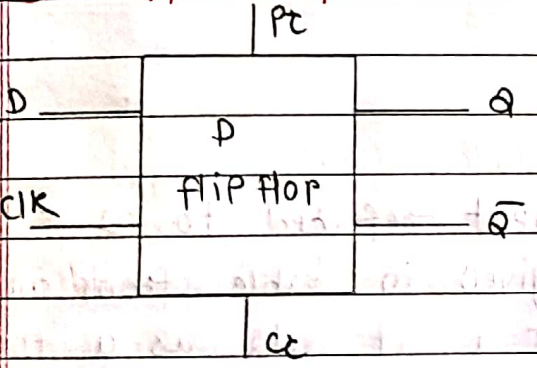


fig :- Logic Symbol

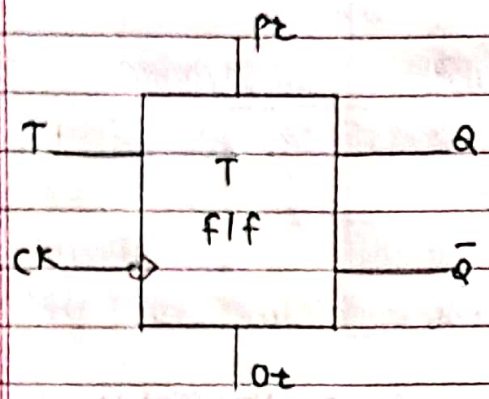
Truth Table :-

Input	output
Dn	Qn+1
0	0
1	1

- D flip-flop has only one input referred to as D input or data input.
- From truth table it is clear that the output Qn+1 at the end of the clock pulse equals the input Dn before the clock pulse.

- This is equivalent to saying that the input data appears at the output at the end of the clock pulse.
- Thus the transfer of data from the input to the output is delayed and hence the name delay (D) flip-flop. The D flf is either used as a delay (D) flf
- The D flf is either used as a delay device or as a latch to store 1 bit of binary information.

o T flip-flop :-



Logic symbol

Truth Table :-

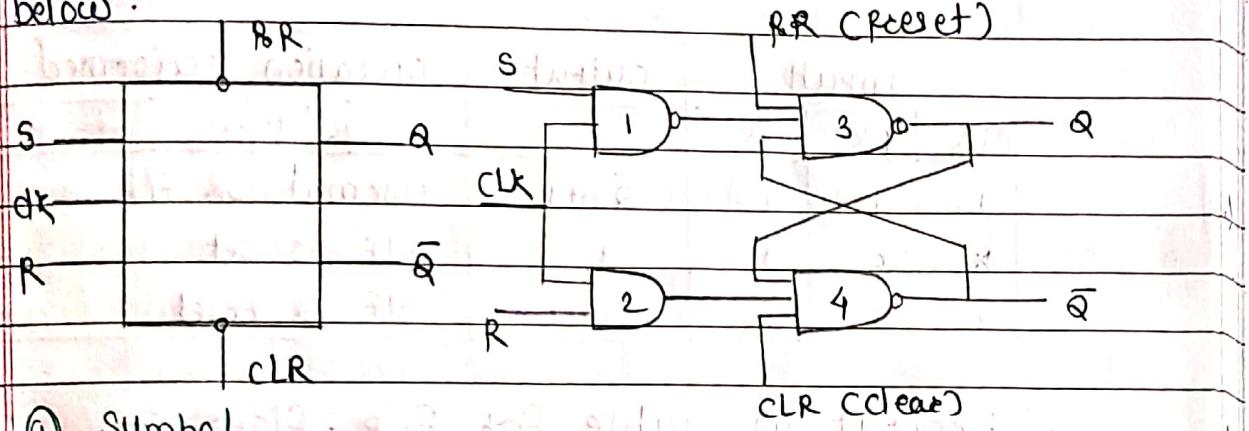
Input	output
Tn	Qn+1
0	Qn
1	$\bar{Q}n$

- It has only one input refered to as T-input It's truth table is given in table from which it is clear that if T=1 it acts as a toggle s/w. for every clock pulse, the output Q changes.

o Use of preset and clear terminals :-

- In the flip-flops when the power switch is turned on, the state of outputs Q & $\bar{Q}$ is uncertain. The outputs can be Q = 0, $\bar{Q}$ = 1 or other way round.

- But this uncertainty can not be tolerate in certain applications. In some applications it is necessary to initially set or reset the flip-flops.
- This can be practically achieved by adding two more inputs to a flip flop called as "preset" (PR) and "clear" (CLR) input.
- These inputs are called as direct or asynchronous inputs because we can apply them any time bet^n clock pulses without thinking about their synchro^n with the clock.
- The S-R flip-flop with preset and clear is shown below.



(a) Symbol

fig :- SR flip-flop with preset & clear.

Case-I PR = CLR = 1

If PR = CLR = 1 then both are inactive and SR flf operates in accordance with the truth table of conventional SR flip-flop.

Case-II PR = 0 and CLR = 1

As PR = 0 the output of gate 3 will be 1. That means Q = 1 Therefore all the three inputs to the gate 4 will be 1 and $\bar{Q}$ will become 0. Thus making PR = 0 will set the flf.

Case-III PR = 1 , CLR = 0

If PR = 1 and CLR = 0 then the output of gate 4 i.e $\bar{Q}$ = 1. Therefore all the three inputs to gate will be 1 and hence Q = 0. Thus making CLR = 0 will reset the flip flop.

Case-IV PR = 0 , CLR = 0

This condition should never be used because it leads to an uncertain state.

The operation of SR flip flop with preset & clear inputs is summarised in the table :-

Input			output	operation performed
CLK	PR	CLR		
1	1	1	$\bar{Q}_{n+1}$	Normal SR f/f
x	0	1	1	f/f is set
x	1	0	0	f/f is reset.

Excitation table for flip-flops :-

- The truth table of a flip-flop is also referred to as the characteristics table and specifies the operational characteristics of the flip-flop.
- In the design of sequential circuits, we usually come across situations in which the "present state" & the "next state" of the circuit are specified and we have to find the input conditions that must prevail to cause the desired transition of the state. By the present state and the next state we mean the state of the circuit prior to and after the clock pulse respectively.

- For example the output of S-R flip-flop before the clock pulse is $Q_n = 0$ and it is desired that the output does not change when the clock pulse is applied.

What input conditions (S_n & R_n values) must exist to achieve this?

From the truth table of SR flip-flop we obtain the following conditions

1. $S_n = R_n = 0$ (First row)
2. $S_n = 0$, $R_n = 1$ (Third row)

We conclude from the above conditions that the S_n input must be 0 whereas R_n input may be either 0 or 1 (don't care). similarly input cond^n can be found for all possible situations.

- A tabulation of these conditions is known as the excitation table.

It is very important and useful design aid for sequential circuit.

Table : Excitation table of Flip-Flops :-

Present state	Next state	SR FlF		JK FlF		T FlF	D FlF
		S_n	R_n	J_n	K_n	T_n	D_n
0	0	0	X	0	X	0	0
0	1	1	0	1	X	1	1
1	0	0	1	X	1	1	0
1	1	X	0	X	0	0	1

Conversion of Flip-Flops :-

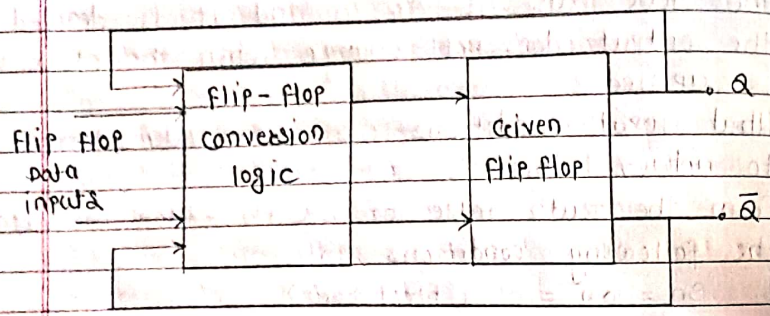


fig :- General model used to convert one type of flip-flop to another type.

- Consider the general model for conversion from one type of flip-flop to other type. In this we are required to design the combinational logic decoder (conversion logic) for converting new i/p definations into input codes which will cause the given flip-flop to perform as desired.

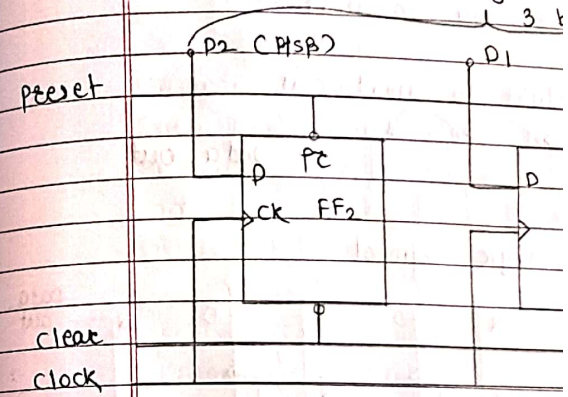
- To design the conversion logic we need to combine the excitation tables for both flip-flops and make a truth table with data inputs and Q as the inputs and the inputs of the given FF as the outputs.

- The conversion method of combinational logic design then follows as usual.

Applications of flip-

1) Register's :-

- A register is composed o
to store a group of bi
an N-bit word, the n
is N (one flip-flop f
register using 7474 posit
flop is shown in fig.



The bits to be stored as
which are clocked in at
clock pulse. In this reg
must be available in pa

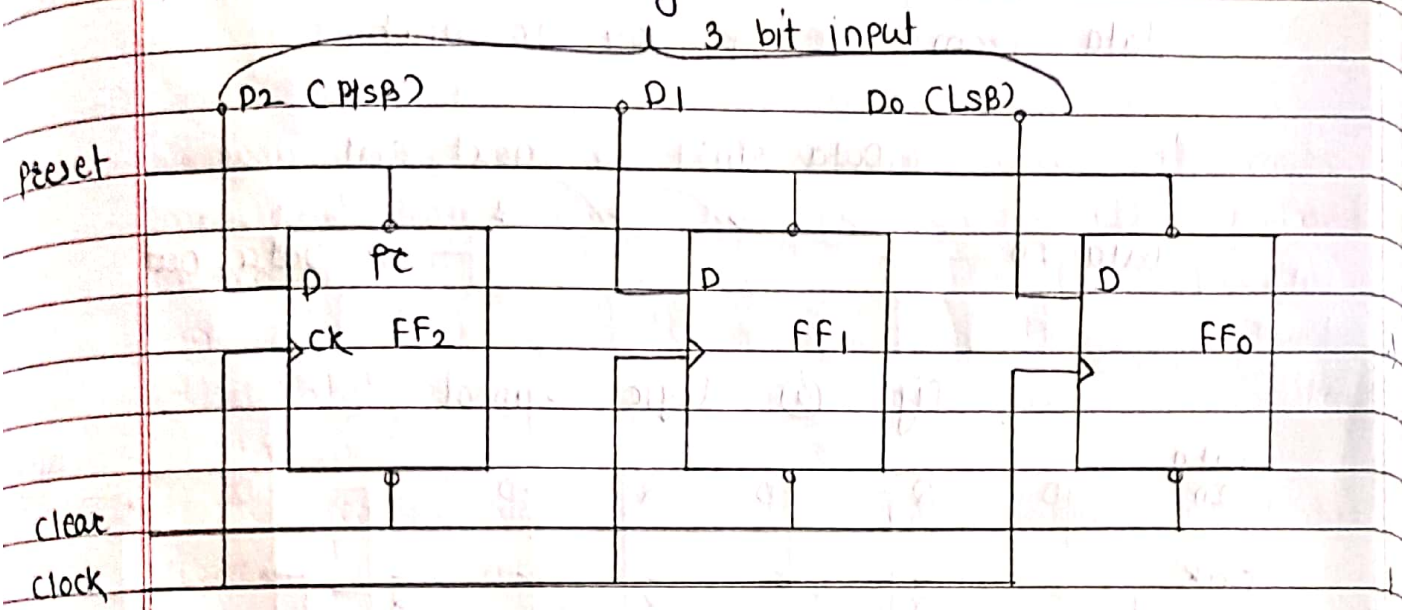
2) Shift registers :-

- shift register is a gr
the capability of stori
information.
- shift register performs
of binary data from

Applications of flip-flops :-

1) Registers :-

- A register is composed of a group of flip-flops to store a group of bits (word). For storing an N-bit word, the no. of flip-flops required is N (one flip-flop for each bit). A 3-bit register using 7474 positive edge triggered flip-flop is shown in fig.



The bits to be stored are applied at the D-input which are clocked in at the leading edge of the clock pulse. In this register, the data to be enter must be available in parallel form.

2) Shift registers :-

- shift register is a group of flip-flops that has the capability of storing and shifting the binary information.
- shift register performs the logic operation shifting of binary data from one flip-flop to next

flip- flop. The shifting of binary information from one position to another is very important logic operation in digital system design.

- In the shift register, the shifting operation is controlled by common clock pulse like register.

- All the flip- flops employed with shift register receive the clock pulse that helps to shift the data from one position to next.

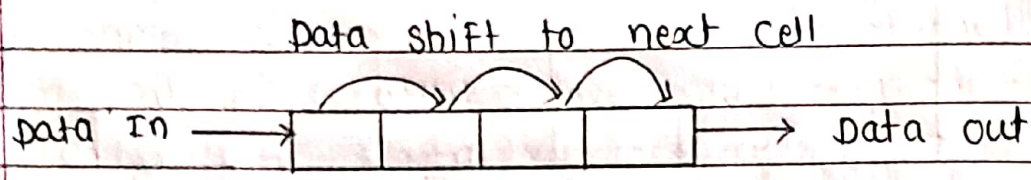
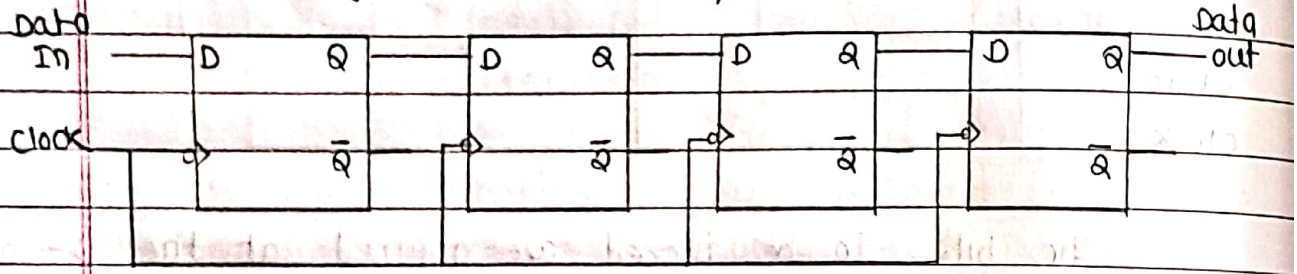


fig (a) Logic symbol



(b) Logic Diagram

fig :- 4 bit shif register.

- The logic diagram of shift register is consisting of four flip flops in which the output of first flf is connected as input for next flf. When the clock pulse is triggering the flf's the data is shifted from one flip - flop into next flip- flop.

The input is loaded through first flip-flop and output is obtained from last flip-flop.

- The commonly used shift registers are
  1 - Serial In Serial out shift register
  2 - Serial In parallel out shift register.
  3 - parallel In parallel out shift register
  4 - parallel In serial out shift register.
  5 - Bidirectional shift register.

3) Counter's :-

- Digital counters are often needed to count events for example, counting the no. of tablets filled in a vial. Electrical pulses corresponding to the event are produced using a transducer and these pulses are counted using counter.

- The counters are composed of flip-flops. A 3-bit counter consisting of three flip-flops is as shown below

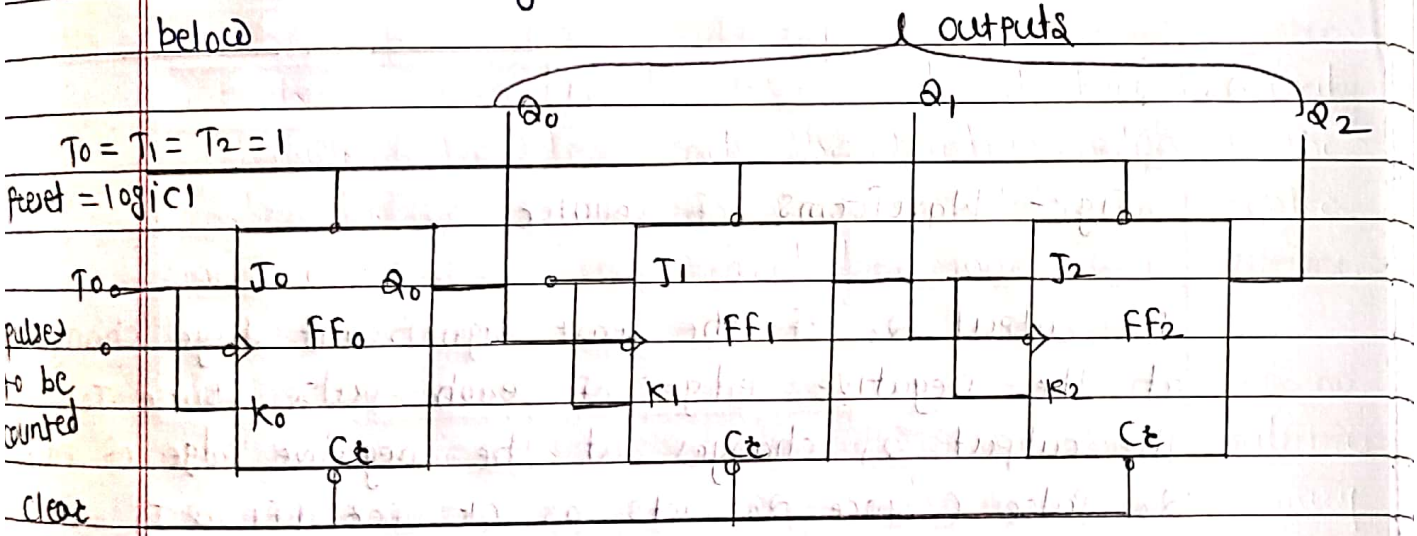


fig:- A 3 bit counter using flip-flops.

- A circuit with n flip-flops has 2^n possible states. Therefore 3 bit counter can count from decimal 0 to 7.
- The flip-flops used are 74107 J-K master-slave flip-flops used as T-type. The pulses to be counted are connected at the clock input of FF0. The Q_0 output of FF_0 is connected to the clock input of FF_1 and similarly Q_1 is connected to the clock input of FF_2.
- The flip-flops are cleared by applying logic 0 at the clear input terminal momentarily. For normal counting operation, it is to be maintained at logic 1. The pulses & output waveforms are given below.

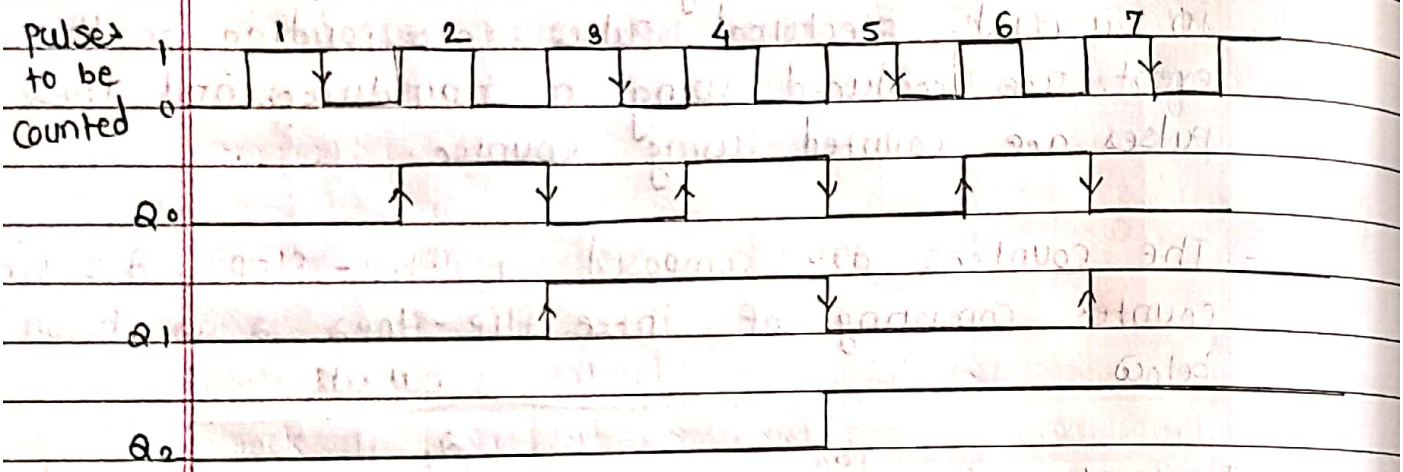
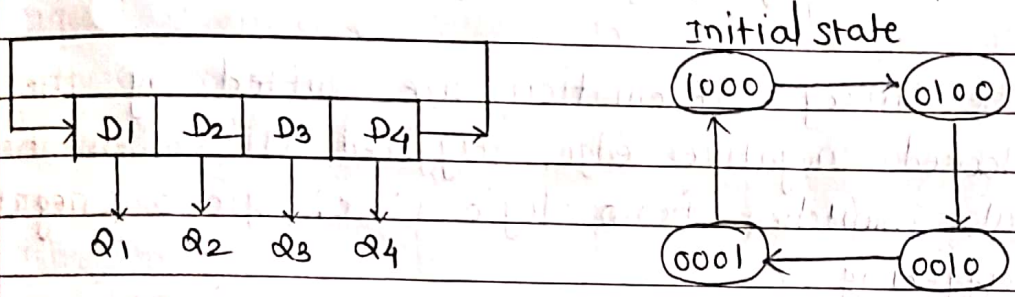


Fig:- Waveforms of counter.

The output Q_0 of the least significant stage change at the negative edge of each pulse (since $T_0=1$). The output Q_1 changes at the negative edge of each Q_0 pulse (since Q_0 acts as CK for FF1 & $T_1=1$ & the output Q_2 changes at the negative edge of each Q_1 pulse (since Q_1 acts as CK for FF_2 & $T_2=1$).

Ring Counter :-

- The design of ring counter is directly obtained from the serial in serial out shift register by connecting the output to the data input terminal hence it is also called as "circular shift register".
- In ring counter only one flip-flop is set to provide logic 1 output at any particular time, other flip-flops are cleared to reset logic 0. single logic 1 bit is shifted circularly through the serial connected flip-flops to produce the sequence of binary data.



ⓐ Logic symbol ⓑ sequence of binary bit shifting.

- In the logic symbol, the binary 4 bit shift register connected as a ring counter. The sequence of binary bit shifting is graphically shown in fig ⓑ.
- In the sequence initially the binary 4 bit ring counter is set into 1000. Then the logic sequence is shifted right in sequence as follows :-

1000 → 0100 → 0010 → 0001 → 1000 ... so on.

- The logic diagram that provides the above sequence i.e a 4 bit ring counter is shown in fig. It is constructed by the series circular connected four negative edge triggered flip-flops.

- Initially only one of the flip-flops is fixed into logic 1. Then for each clock cycle the bit 1 is shifted to right to get the binary sequence.

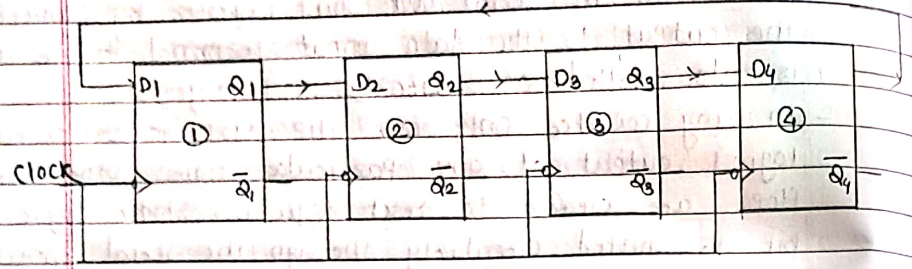


fig. Logic diagram

- The binary information are shifted by the commonly clocked negative edge triggered flf when the clock pulse switches from logic 1 to logic 0 (negative edge triggering).

Truth table of ring counter :-

Control i/p	output				States
clock	Q_1	Q_2	Q_3	Q_4	
0 → 1	1	0	0	0	Initial state
1 → 0	0	1	0	0	After 1st clock
1 → 0	0	0	1	0	After 2nd clock
1 → 0	0	0	0	1	After 3rd clock
1 → 0	1	0	0	0	After 4th clock

The timing diagram is as shown below.
The time sequence of each state is equal to one fourth of clock freq?

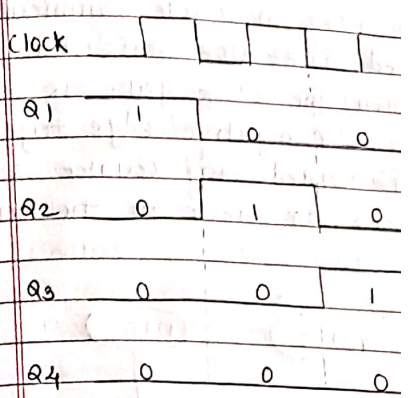


fig :- Timing diagram

o Twisted Ring Coun

- Johnson counter is a
inverted output of l
input for first flip-
as "twisted ring count

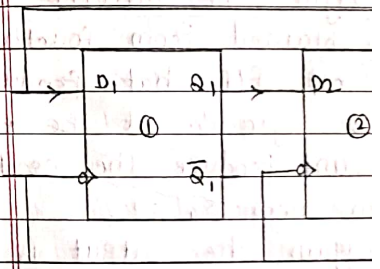


fig :- Logic d

The diagram is cor
common clocked fou
The inverted output

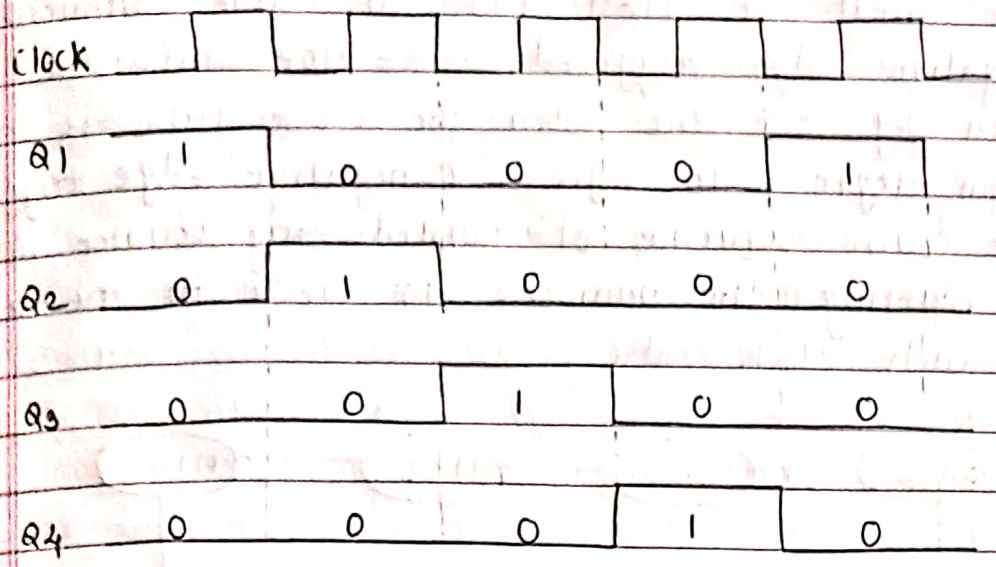


fig :- Timing diagram of 4 bit ring counter.

o Twisted Ring Counter (Johnson Counter) :-

- Johnson counter is a modified ring counter that the inverted output of last flip-flop is applied as input for first flip-flop. Hence it is also called as "twisted ring counter" or "switch tail ring counter".

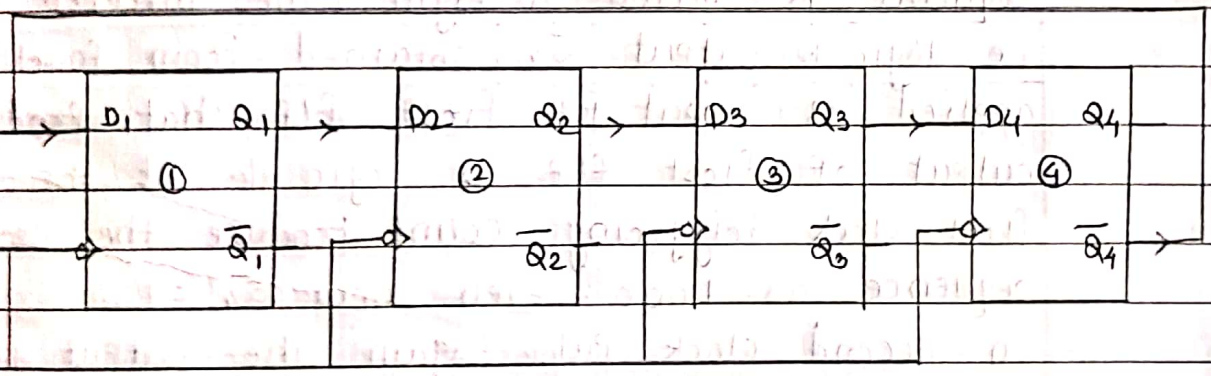


fig :- Logic diagram of Johnson counter.

The diagram is constructed with series connected common clocked four negative edge triggered flf. The inverted output of flf 4 is connected with

the input of first flf in circle commonly clock negative edge triggered flip-flop shifts the binary data by one bit when the clock pulse is switch from logic 1 to logic 0 (negative edge triggering). The count sequence of twisted ring counter is obtained by shifting the numbers in circle to the triggering of each clock pulse.

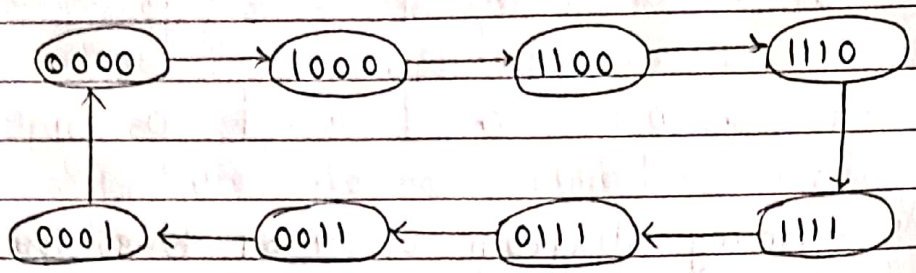


fig:- Logic sequence of twisted ring counter
[Johnsons counter]

In the 4-bit Johnson counter, initially the count sequence is cleared and set to 0000. When the first clock pulse triggers the flip-flops the count sequence is shifted to right. The inverted output i.e logic 1 output $\bar{Q}_4$ obtained from fourth flf is applied as input for first flf that produces the output of first flf as logic 1. At the end of first clock triggering count produce the next output sequence as 1000. Even now $\bar{Q}_4 = 1$.

- In second clock pulse again the output is shifted right to produce the output as 1100.
- In third clock triggering again the output is shifted to produce the output as 1110.
- After fourth clock pulse all the flf outputs are changed into logic 1 that switches the output $\bar{Q}_4$ to logic 0.

the input of first flf in circle commonly clock negative edge triggered flip-flop shifts the binary data by one bit when the clock pulse is switch from logic 1 to logic 0 (negative edge triggering). The count sequence of twisted ring counter is obtained by shifting the numbers in circle to the triggering of each clock pulse.

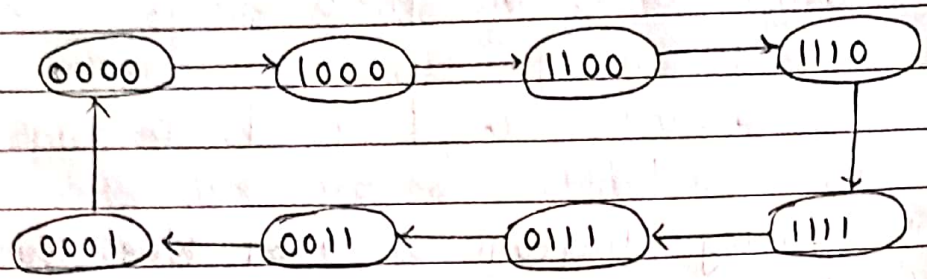


fig:- logic sequence of twisted ring counter
[Johnsons counter]

In the 4-bit Johnson counter, initially the count sequence is cleared and set to 0000. When the first clock pluse triggers the flip-flops the count sequence is shifted to right. The inverted output i.e logic 1 output $\bar{Q}_4$ obtained from fourth flf is applied as input for first flf that produces the output of first flf as logic 1. At the end of first clock triggering count produce the next output sequence as 1000. Even now $\bar{Q}_4 = 1$.

- In second clock pulse again the output is shifted right to produce the output as 1100.
- In third clock triggering again the output is shifted to produce the output as 1110.
- After fourth clock pulse all the flf outputs are changed into logic 1 that switches the output $\bar{Q}_4$ to logic 0.

- Then for the triggering of each clock pulse, the inverted output of flf 4 $\bar{Q}_4$ (logic 0) is shifted.

Truth table of ring counter :-

control input	outputs				states	preceding output
clock	Q_1	Q_2	Q_3	Q_4		
0 → 1	0	0	0	0	Inital state	$\bar{Q}_1 \bar{Q}_4$
1 → 0	1	0	0	0	After 1st clock	$\bar{Q}_1 \bar{Q}_2$
1 → 0	1	1	0	0	After 2nd clock	$Q_2 \bar{Q}_3$
1 → 0	1	1	1	0	After 3rd clock	$Q_3 \bar{Q}_4$
1 → 0	1	1	1	1	After 4th clock	$Q_1 Q_4$
1 → 0	0	1	1	1	After 5th clock	$\bar{Q}_1 Q_2$
1 → 0	0	0	1	1	After 6th clock	$\bar{Q}_2 Q_3$
1 → 0	0	0	0	1	After 7th clock	$\bar{Q}_3 Q_4$

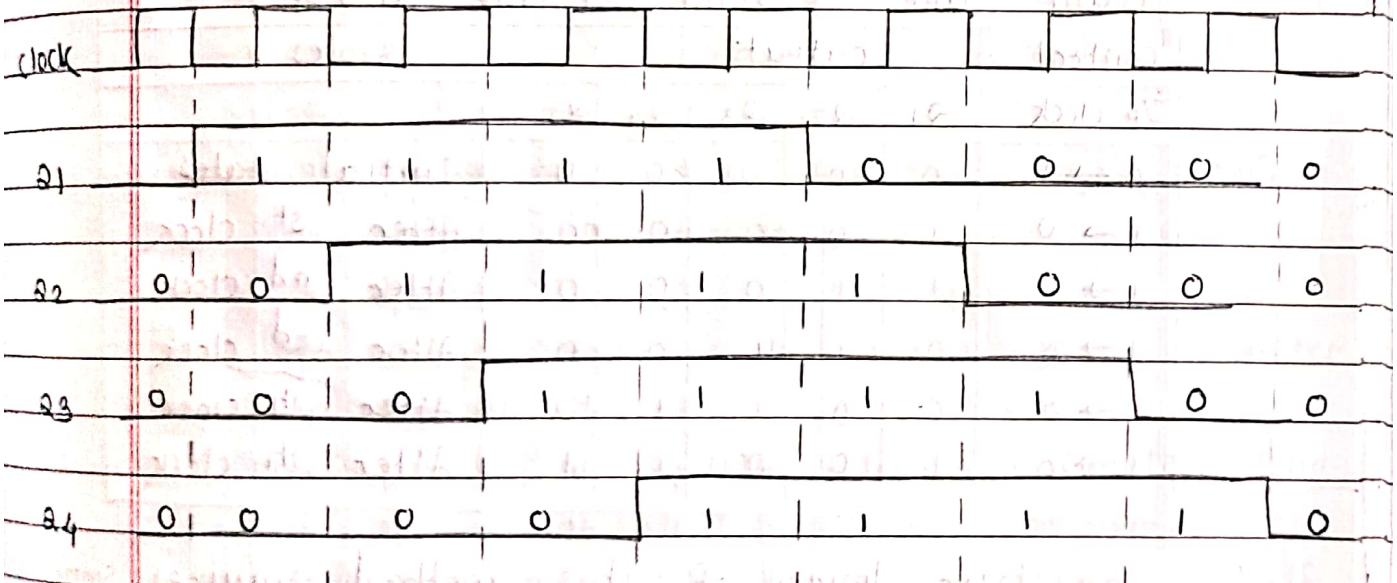


fig :- Timing diagram of 4 bit Johnson counter [Twisted ring counter]

4) Sequence Generator :-

- Sequence generator is a design that used to construct a logic diagram for any prescribed format of binary number.
- The design of sequence generator for any sequence of binary number can be obtained from shift registers.
- An n bit sequence generator design requires an n-bit shift register where n is a length of sequence.
- Consider the binary sequence 10011. The length of sequence is 5 (5 → no. of bits). The sequence of binary data can be loaded in direct to the shift registers.

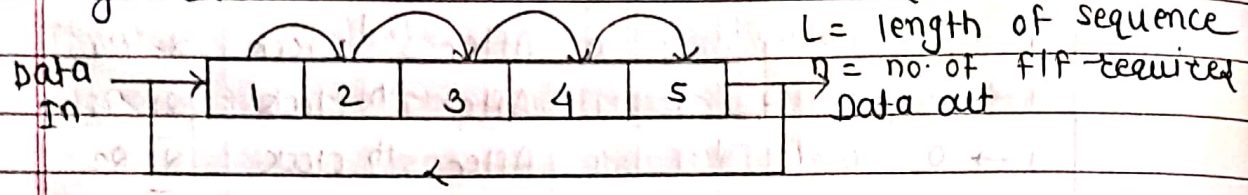


fig :- Logic symbol of sequence generated.

Truth table of 5-bit sequence counter :-

Control	outputs					states
I/P clock	Q1	Q2	Q3	Q4	Q5	
0 → 1	0	0	0	0	0	Initial states
1 → 0	1	0	0	0	0	After 1st clock
1 → 0	1	1	0	0	0	After 2nd clock
1 → 0	0	1	1	0	0	After 3rd clock
1 → 0	0	0	1	1	0	After 4th clock
1 → 0	1	0	0	1	1	After 5th clock

- The logic design of this method requires same no. of FF that are equal to the length of sequence.

5) Ripple Counters (Asynchronous Counter) :-

- Asynchronous counters are the MSI circuit that are used to count the binary numbers in prescribed by sequence. In asynchronous counter the flfs are not triggered with common clock pulse except the first (least significant) flip-flop others are triggered by the output of previous flf while the first one is triggered by the clock pulse. Hence asynchronous counter is also called as "Ripple counter" Basically it is classified into two types such as up counter and down counter.

i) Up - Counter :-

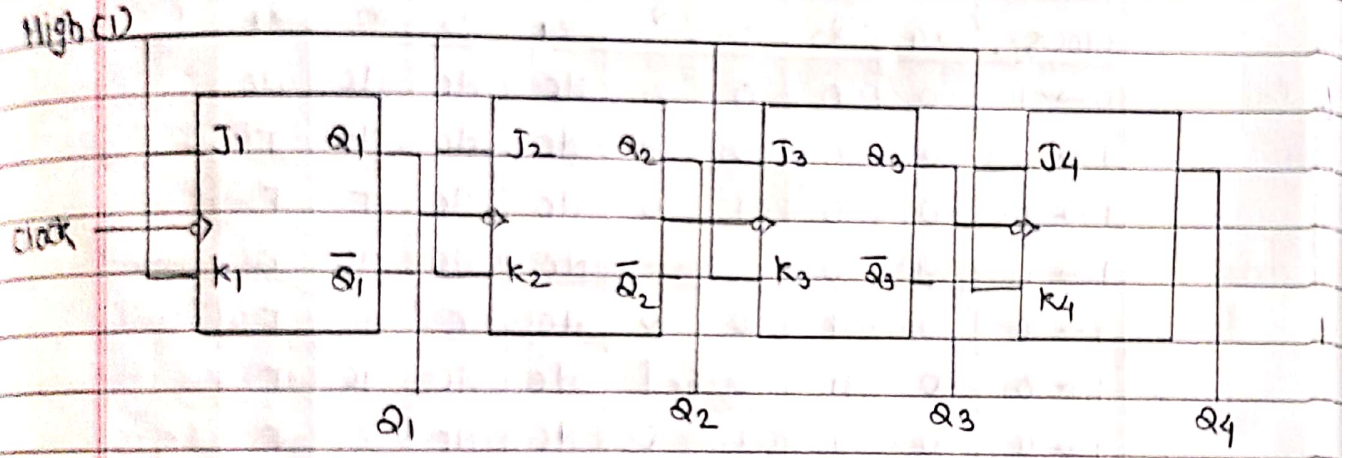


Fig:- Logic diagram of 4-bit Asynchronous (Ripple) Up - Counter.

- The logic diagram of 4-bit asynchronous up-counter is as shown in fig.
- It consists of four series connected complementing flf (JK or T) without common clock pulse. The clock pulse is applied only to the first (LSB) flf to drive it. All other flf are driven by the o/p of previous flf instead of clock pulse.

- The transition that occurs in the o/p of flf triggers the clock terminal of next flf Hence it is called as "Ripple up-counter".
- The i/p's of all the flf set into logic high (level). When inputs set into logic high the JK flf's are continuously present in the toggle condition which complements the outputs continuously. This cause to prevent the circuit from triggering of two adjacent flf simultaneously.

Truth table of 4-bit Asynchronous up-counter:-

Inputs	outputs				state change			
clock	Q_4	Q_3	Q_2	Q_1	Q_4	Q_3	Q_2	Q_1
0→1	0	0	0	0	NC	NC	NC	NC
1→0	0	0	0	1	NC	NC	NC	T
1→0	0	0	1	0	NC	NC	T	T
1→0	0	0	1	1	NC	NC	NC	T
1→0	0	1	0	0	NC	T	T	T
1→0	0	1	0	1	NC	NC	NC	T
1→0	0	1	1	0	NC	NC	T	T
1→0	0	1	1	1	NC	NC	NC	T
1→0	1	0	0	0	T	T	T	T
1→0	1	0	0	1	NC	NC	NC	T
1→0	1	0	1	0	NC	NC	T	T
1→0	1	0	1	1	NC	NC	NC	T
1→0	1	1	0	0	NC	T	T	T
1→0	1	1	0	1	NC	NC	NC	T
1→0	1	1	1	0	NC	NC	T	T
1→0	1	1	1	1	NC	NC	NC	T

Note :- NC → No change (0 to 1); T - toggle (complement of previous o/p) (1 to 0 change)

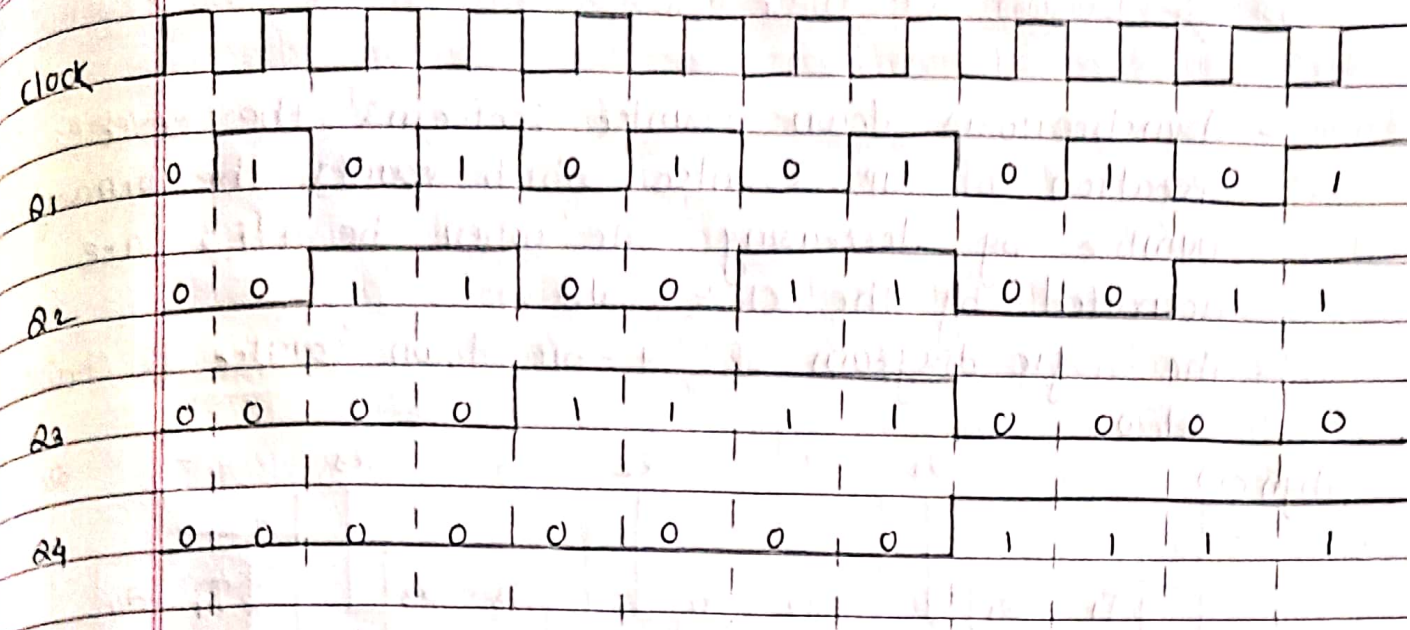


fig :- Timing diagram of 4 bit Asynchronous up- counter.

- The table shown above describes the logic operat^{n} of 4 bit asynchronous up counter with the switching state of f/f.
- The negative edge triggered f/f are used to perform this counter function i.e the f/f's are activated to outputs for the change of i/p.
- When the clock pulse is active low signal, the change of clock pulse from logic 1 to logic 0 trigger's the f/f 1.
- The change in o/p of f/f 1 from logic 1 to logic 0 triggers the f/f_2.
- The change in o/p of f/f_2 from logic 1 to logic 0 triggers the f/f_3. and so on.
- The timing diagram illustrates the logic change of 4 bit asynchronous up counter.

ii) Down Counter :-

- Asynchronous down counter performs the reverse operation of up counter which counts the binary number by decreasing one when the flf's are activated by the clock pulse.
- The logic diagram of 4-bit down counter is shown below.

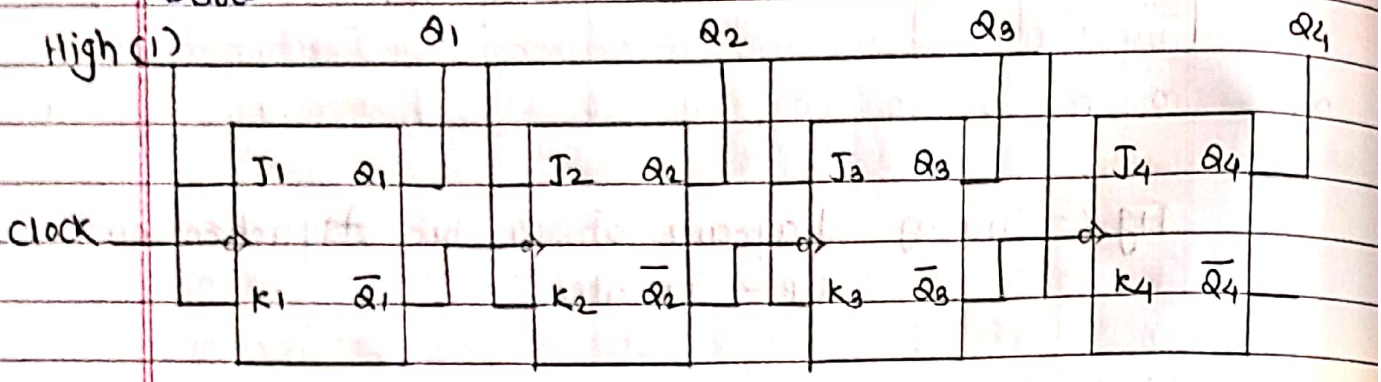


fig :- Logic diagram of 4- bit asynchronous down counter.

- The circuit is constructed using four series connected flfs similar to up counter.
- First flf is triggered by clock pulse, The remaining three are triggered by the inverted o/p of previous flf i.e $\bar{Q}$. It is only difference from up counter.
- The negative edge triggering of flf are used to construct a counter so the flf 2, 3, & 4 are enabled. When the o/p of flf 1, 2 and 3 i.e $\bar{Q}_1$, $\bar{Q}_2$ & $\bar{Q}_3$ are switched from logic 1 to logic 0 respectively.
- flf 1 is enabled when the clock pulse is switched from logic 1 to logic 0 due to the negative edge triggering.
- The inputs of all the flf are set into logic high (logic 1). When inputs are set into logic high the

JK flfs are operated with toggle condition which is complement of previous output. It prevents the circuit from triggering of two adjacent flf simultaneously.

Truth table 4 bit Asynchronous Down counter.

Input	output				state change			
clock	Q_4	Q_3	Q_2	Q_1	Q_4	Q_3	Q_2	Q_1
0 → 1	0	0	0	0	NC	NC	NC	NC
1 → 0	1	1	1	1	T	T	T	T
1 → 0	1	1	1	0	NC	NC	NC	T
1 → 0	1	1	0	1	NC	NC	T	T
1 → 0	1	1	0	0	NC	NC	NC	T
1 → 0	1	0	1	1	NC	T	T	T
1 → 0	1	0	1	0	NC	NC	NC	T
1 → 0	1	0	0	1	NC	NC	T	T
1 → 0	1	0	0	0	NC	NC	NC	T
1 → 0	0	1	1	1	T	T	T	T
1 → 0	0	1	1	0	NC	NC	NC	T
1 → 0	0	1	0	1	NC	NC	T	T
1 → 0	0	1	0	0	NC	NC	NC	T
1 → 0	0	0	1	1	NC	T	T	T
1 → 0	0	0	1	0	NC	NC	NC	T
1 → 0	0	0	0	1	NC	NC	T	T

NC → No change
otal change
→ Toggle
complement
previous o/p)
1 to 0 change)

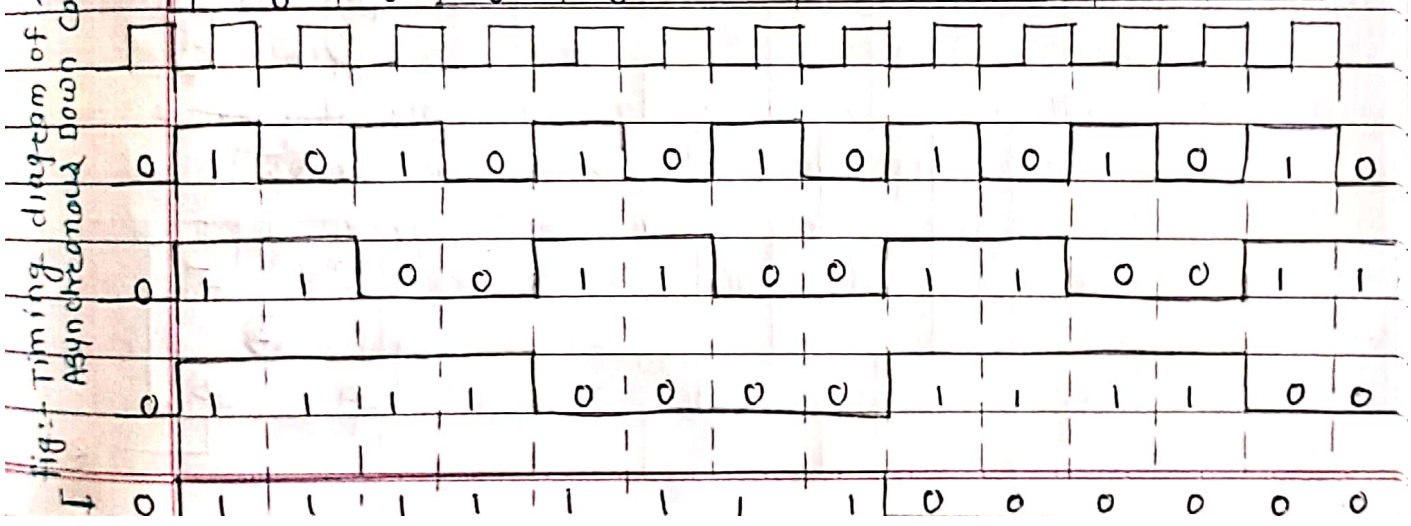


fig:- Timing diagram of 4-bit Asynchronous Down Counter

6) UP/ down Counter [Asynchronous UP/down counter]

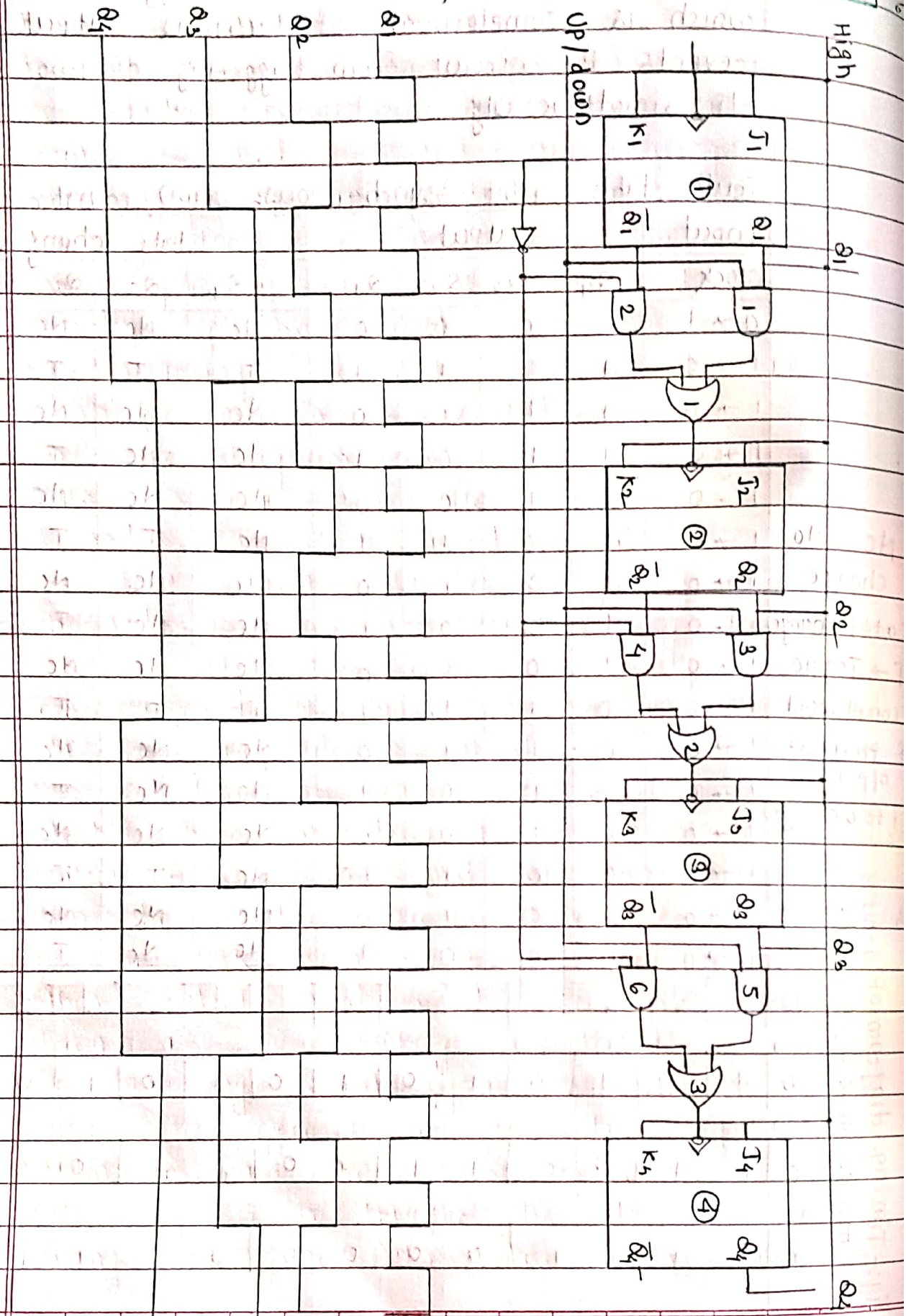


Table :- Truth table of Asynchronous up / down counter :-

Control i/p	UP / Down	outputs				state changed			
Clock		Q_4	Q_3	Q_2	Q_1	Q_4	Q_3	Q_2	Q_1
0 → 1	Initial state	0	0	0	0	NC	NC	NC	NC
1 → 0	0	1	1	1	1	T	T	T	T
1 → 0	0	1	1	1	0	NC	NC	NC	T
1 → 0	0	1	1	0	1	NC	NC	T	T
1 → 0	0	1	1	0	0	NC	NC	NC	T
1 → 0	0	1	0	1	1	NC	T	T	T
1 → 0	0	1	0	1	0	NC	NC	NC	T
1 → 0	0	1	0	0	1	NC	NC	T	T
1 → 0	0	1	0	0	0	NC	NC	NC	T
1 → 0	0	0	1	1	1	T	T	T	T
1 → 0	0	0	1	1	0	NC	NC	NC	T
1 → 0	0	0	1	0	1	NC	NC	T	T
1 → 0	0	0	1	0	0	NC	NC	NC	T
1 → 0	0	0	0	1	1	NC	T	T	T
1 → 0	0	0	0	1	0	NC	NC	NC	T
1 → 0	0	0	0	0	1	NC	NC	T	T
1 → 0	0	0	0	0	0	NC	NC	NC	T
1 → 0	1	0	0	0	1	NC	NC	NC	T
1 → 0	1	0	0	1	0	NC	NC	T	T
1 → 0	1	0	0	1	1	NC	NC	NC	T
1 → 0	1	0	1	0	0	NC	T	T	T
1 → 0	1	0	1	0	1	NC	NC	NC	T
1 → 0	1	0	1	1	0	NC	NC	T	T
1 → 0	1	0	1	1	1	NC	NC	NC	T
1 → 0	1	1	0	0	0	T	T	T	T
1 → 0	1	1	0	0	1	NC	NC	NC	T
1 → 0	1	1	0	1	0	NC	NC	T	T
1 → 0	1	1	0	1	1	NC	NC	NC	T
1 → 0	1	1	1	0	0	NC	T	T	T
1 → 0	1	1	1	0	1	NC	NC	NC	T
1 → 0	1	1	1	1	0	NC	NC	T	T
1 → 0	1	1	1	1	1	NC	NC	NC	T

7) Synchronous Counter :-

- A counter which performs count in either up or down is called up/down counter. It can't perform up and down counting simultaneously.
- In asynchronous up-down counter, an extra control single is used to select the counting operation increasingly by up counting or decreasingly by down counter.
- The progressing of up or down counting is fixed initially by the control signal up/down; Then by enforcing the clock pulses to the flf desired counting operation is executed.
- In logic diagram, it consists of series connected four JK flf with logic high common i/p signal J and K. The clock signal is called applied to the LSB (first) flf as triggering signal.
- The flf 2, 3, 4 are triggered by the o/p or previous flip flops 1, 2, 3 respectively. The flf are triggered by the o/p $\bar{Q}$ for up counting while flf are triggered by output $\bar{Q}$ for down counting
- The selection of o/p to trigger a next flf is selected the combination of AND gates with the control of up/down signal as shown in fig.
- Then the selected logic counting operation is directed to the clock terminal of flf the OR gate.
- When control signal is set into logic 1 i.e up/$\overline{down}$ = 1, up counting is selected & executed by increasing count by 1 for each applied clock pulse. The logic 1 i/p of control s/g up/$\overline{down}$ allow to pass the o/p Q_1, Q_2 & Q_3 the AND gate 1, 3 & 5 to trigger the flf 2, 3 & 4 resp. During this time AND gates 2, 4, & 6 are disable

by the control signal that passed through inverter.

- When the control signal is set into logic 0 i.e UP / $\overline{DOWN}$ = 0 down counting is selected & executed by decreasing count by 1 for each applied clock pulse. The logic 0 input control signal disables the AND gates 1, 3 & 5 and enables the AND gates 2, 4 & 6 th' on inverter. This set is allowing to pass the o/p $\overline{Q_1}$, $\overline{Q_2}$ and $\overline{Q_3}$ th' the enabled AND gates 2, 4 & 6 to trigger the flf 2, 3 & 4 respectively.

Definations :-

- Clock skew :-

Clock skew is a phenomenon in synchronous digital circuit systems in which the same sourced clock signal arrives at diff. components at diff. times due to gate or in more advanced semiconductor technology, wire signal propagation delay.

- Clock Jitter :-

Clock jitter describes how uniform the clock edges are in a digital signal. clock jitter will influence the skew of a clock overtime. If the clock is supposed to have an edge every m sec but they fall bet^n 0.9 and 1.1 ms this would be the jitter. This is a sporadic deviation for the specific clock.

o Lock out :-

In digital counters, the condition where in the counter enter to an unused state and rather than coming out of this unused state to a valid state or initial state, it switched to another invalid or unused state and gets stuck up in the cycle of unused states only is known as the lock out condition in the counter.