❖ COMBINATIONAL CIRCUIT

- ✓ A combinational circuit is a connected arrangement of logic gates with a set of inputs and outputs.
- ✓ As shown in the below figure n binary input variable comes from an external source; the m binary output variable go to an external destination, and in between there is an interconnection of logic gates.
- ✓ A combinational circuit transforms binary information from the given input data to the required output data.
- ✓ Combinational circuit are employed in digital computer for generating binary control decisions and for providing digital components required for data processing.

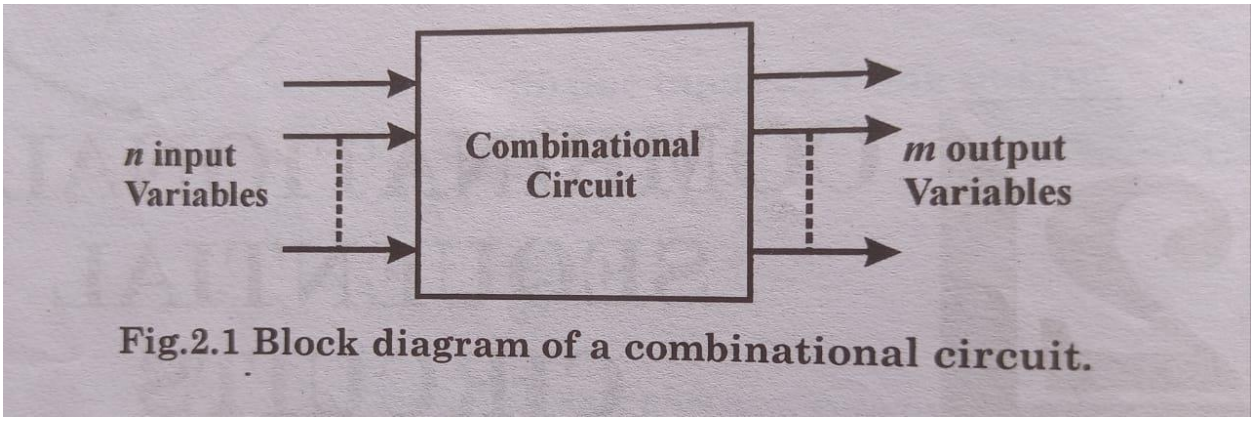

Fig.2.1 Block diagram of a combinational circuit.

❖ ARITHMETIC CIRCUITS

- ✓ Digital computer perform a variety of arithmetic operations, like addition, substraction etc.

[A] Half-Adder

- ✓ A combinational circuit that perform the arithmetic addition of two bits is called a half-adder.
- ✓ This circuit need two binary input X and Y and two output variable S(sum) and C (Carry).
- ✓ The truth table shows that when both input X and Y is high then output C produced 1- carry, while for other cases it will not generate any carry bit.
- ✓ The S output represents the least significant bit of the sum.

The Boolean functions for the two outputs can be obtained directly from the truth table as follow :

Sum S = X' Y + X Y' = X⊕Y (Ex-OR gate)
Carry C = XY (AND gate)

Table- 2.1 : Truth table for Half Adder

Inputs		Outputs	
X	Y	S Sum	C Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

The logic diagram is shown in the Fig-2.2. It consists of an Ex-OR gate and an AND gate.

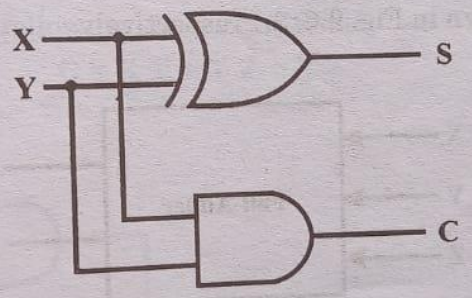

Fig. 2.2 Logic diagram of Half Adder

[b] Full-Adder

A full-adder is a combinational circuit that performs the arithmetic sum of three input bits. It consists of three inputs and two outputs. When we want to add two binary numbers, each having two or more bits, the LSB (Least Significant Bits) can be added by using a half-adder.

Two of the input variables, denoted by A and B, represent the two significant bits to be added and third input C, represents the carry from the previous lower significant position.

Table - 2.2 : Truth Table of Full Adder

Inputs			Outputs	
X	Y	Z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

The block diagram of the full-adder is shown in Fig-2.3 and logic diagram for sum and carry output is shown in Fig. 2.6, 2.7 respectively.

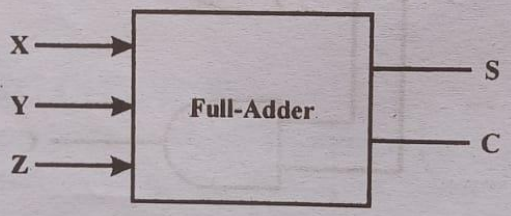

Fig. 2.3 Block diagram of Full Adder

➢ **K-MAP for Sum :**

From the table-2, it is clear for decimal value 1, 2, 4 and 7 s-output is high. Therefore mark respective cell with '1' as shown in the K-map.

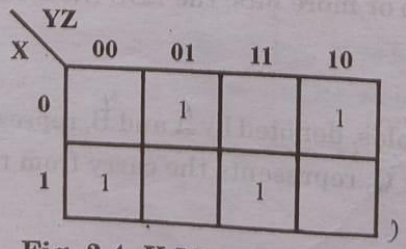

Fig. 2.4 K-Map for sum

One can't create any pair in the above K-map, threfore equation is as follows:

$$S = X' Y' Z + X' Y Z' + X Y' Z' + X Y Z$$

K-MAP for Carry:

Carry output is high for decimal value 3,5,6 and 7 (see table-2). So K-map for carry can be drawn as shown below. In this map it is possible to create three pairs,

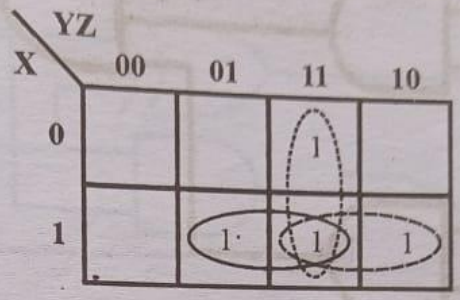

Fig. 2.5 K-Map for carry

From the above map the following equation is found,

$$C = X Y + Y Z + X Z$$

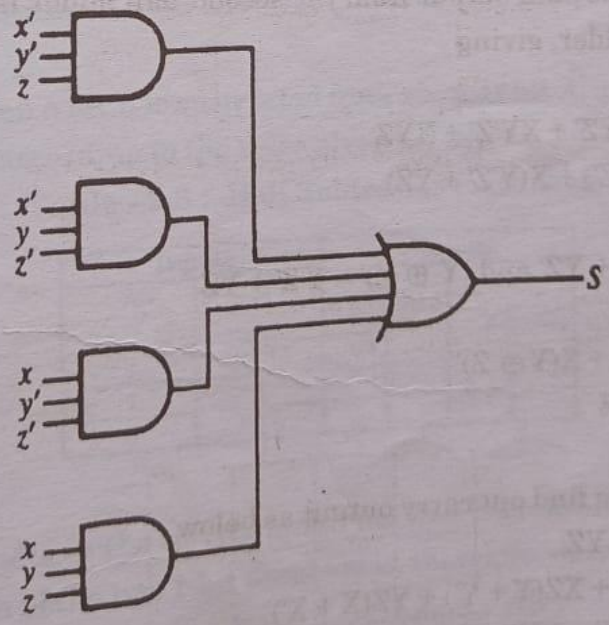

Fig. 2.6 Logic diagram for sum output

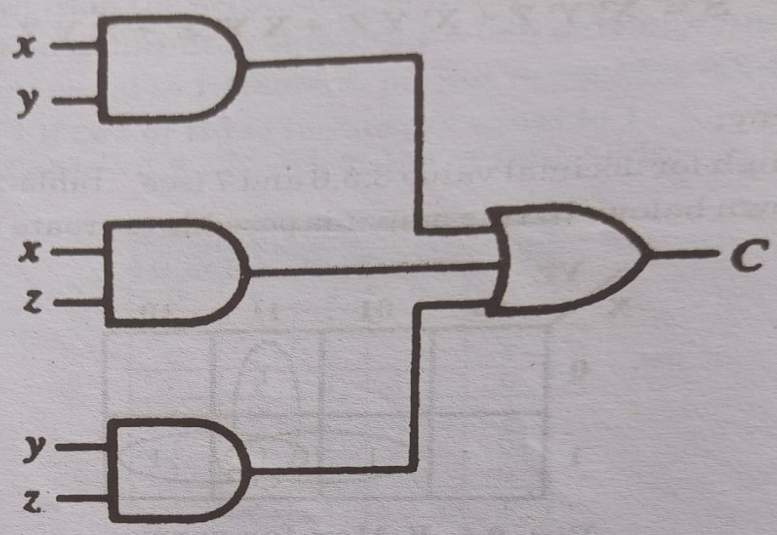

Fig. 2.7 Logic diagram for carry output

[c] Half - Subtractor

A half-Subfactor is an arithmetic circuit that substracts one bit from the other. It is used to subtract the LSB of the subtrahend from the LSB of the minuend when one binary number is subtracted from the other.

We know that when a bit B is sustracted from another bit A, a different bit (d) and a borrow bit (b) result according to the rules given below:

Table - 2.3 : Half Subtractor truth table

Inputs		Outputs	
A	B	d	b
0	0	0	0
1	0	1	0
1	1	0	0
0	1	1	1

A circuit that produces the correct difference and borrow bits in response to every possible combination of the two 1-bit numbers is, therefore, described by

$$d = A\overline{B} + B\overline{A} = A \oplus B \text{ and } b = \overline{A}B$$

❖ FLIP- FLOP(FF)

- ✓ A flip flop is a binary cell capable of storing one bit of information. i.e. they contain either 1 or 0.
- ✓ It is a kind of an electronic circuit which has two stable states, therefore it is also known as **bistable multivibrator**.
- ✓ Today, the term flip-flop has come to generally denote non-trasparenet devices, while the simpler transparent ones are often referred to as **latches**
- ✓ A flip-flop usually controlled by one or two control signal and/or gate.
- ✓ The output often includes the component as well as normal output.

➢ Application of Flip- Flop

- ✓ A single flip-flop can be used to store one- bit, or binary digit of data
- ✓ Flip flops are the fundamental of shift registers and counters.
- ✓ Static RAM is built out of flip flops.
- ✓ Any one of the flip flop types can be used to build any of the others.
- ✓ The data contain in several flip-flop may represent the state of sequencer, the value of counter, or any other piece of information.
- ✓ The flip-flop remember the machine previous state, and digital logic uses that state to calculate the next state.

❖ TYPES OF FLIP-FLOPS

1) R – S Flip Flop

- ✓ The most fundamental flip-flop is the simple RS flip flop or (**RS latch**), where R and S stand for Set and Reset.
- ✓ It can be constructed from a pair of cross-coupled NOR or NAND logic gates
- ✓ The stored bit is present on the output marked Q.
- ✓ In the figure R-S flip flop using two NOR gate is displayed.
- ✓ As per figure it has two input names Reset(R) and Set(S) and two output named Q and Q'.
- ✓ To analyzed the operation of the circuit, we must remember that output of a NOR gate is 0 if any input is 1 and its output is 1 only when all inputs are 0
- ✓ The first input in truth table is R=0,S=0. Since a zero at the input of a NOR gate has no effect on its output, the flip flop remain in its present state that is, Q remains unchanged.

- ✓ The second input condition, R=0 and S=1 forces the output of second NOR gate to be low. Both inputs to NOR gate(1) are now low and the NOR gate output must be high.
- ✓ Third condition is R=1 and S=0 forces the output of NOR gate(1) is low. This cause the both inputs of gate(2) at 0. Therefore output of gate(2) is high.

✓ The last condition in the table is R=1 and S=1 is forbidden, and one cannot predict what would be the value of Q and Q'

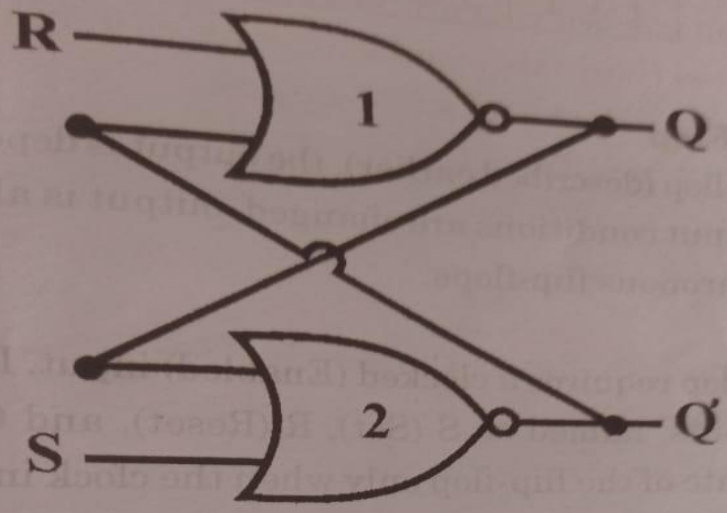

Fig.2.12(a)RS FF Logic Diagram.

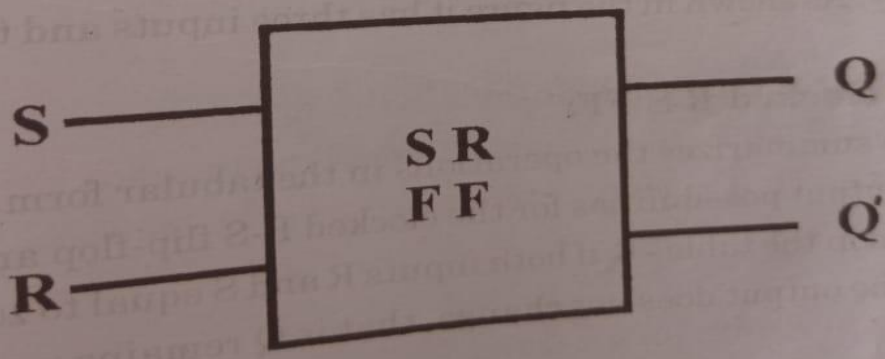

Fig.2.12(b) Block Diagram of RS FF

Table - 2.5 : Truth table for RS FF

R	S	Q	Comments
0	0	1	No Change
0	1	1	SET
1	0	0	Reset
1	1	?	Forbidden

2) Clocked R-S Flip Flop

- ✓ In the simple R-S flip flop, the output is depends on input conditions that is at any time the input conditions are changed, output is also changed. Therefore they are called as asynchronous flip-flops.
- ✓ The clocked R-S flip flop required a clocked(Enabled)input.
- ✓ It means that this type of flip-flop has three inputs, named as S(Set), R(Reset), and C(Clock).
- ✓ Its S and R inputs will control the state of the flip-flop only when the clock input is high.
- ✓ When the clock input is low, the input become ineffective and no change of state can take place.
- ✓ This flip-flop is also known as Gated R-S flip flop or synchronous R-S flip flop

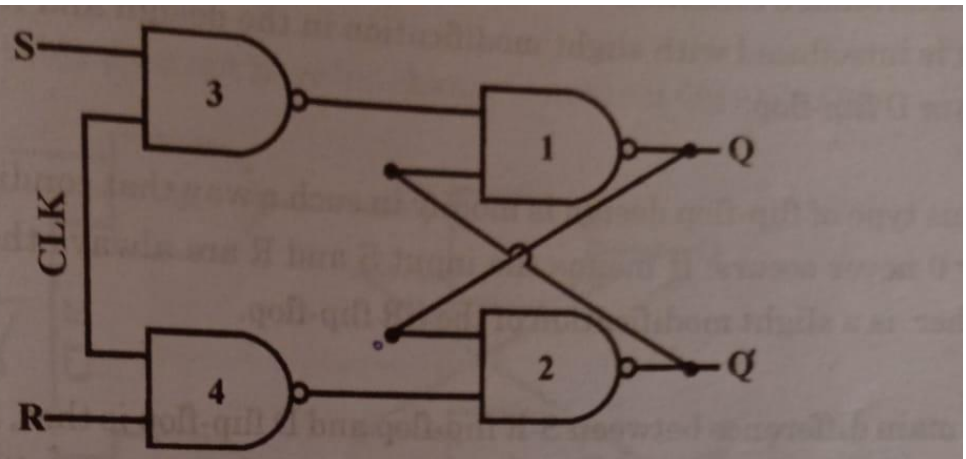

Fig.2.13 (a) Logic Diagram of SR - FF with Clock

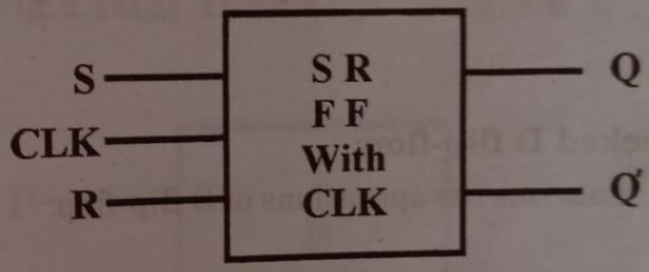

Fig.213(b) Block Diagram ofRS FF with Clock

Table - 2.6 : Truth table for RS FF with Clock

R	S	Clocked	Output Q
0	0	1 (High)	No Change
0	1	1 (High)	1 (SET)
1	0	1 (High)	0 (RESET)
1	1	1 (High)	Indeterminate

- **Operation of clocked R-S FF :**

✓ If both input R and S equal to zero during the clock transition the output does not change, that is Q remain unchanged.

✓ The second input conditions, R=0 and S=1 forces the output of first NAND gate to be high, that is Q=1 is said to SET.

✓ The third input condition in truth table, R=1 and S=0 forces the output of second NAND gate to be high.

- ✓ An intermediate condition occurs when clocked input is high, and both input S=1, R=1.

3) D Flip-Flop(Data FF)

- ✓ The main disadvantage of R-S flip-flop is when both inputs at high level it produced forbidden conditions.
- ✓ To eliminate this undesirable condition, the new kind of flip-flop is introduced with slight modification in the design and it is known as Delay flip-flop or D Flip-flop.
- ✓ In this type of flip-flop design is modify in such a way that conditions R=S=1 and R=S=0 never occurs.
- ✓ It means the input S and R are always the complemented of each other. It is a slight modification of the SR flip flop.
- ✓ The main difference between S-R flip flop and D flip-flop is that, D flip-flop has only one input other than clock input.

➢ **Operation of clocked D flip-flop**

- ✓ When D=0 and clocked is high then it makes next state Q(t+1) low, i.e. reset the flip-flop.
- ✓ While, when D value is high with high clock pulse it causes the flip-flop to set.
- ✓ In other words we can say that output is follows D input when clock pulse is high.
- ✓ We can summarized above discussion as that next state Q(t+1) is independent of present state Q(t). This means that
  Q(t+1) = D

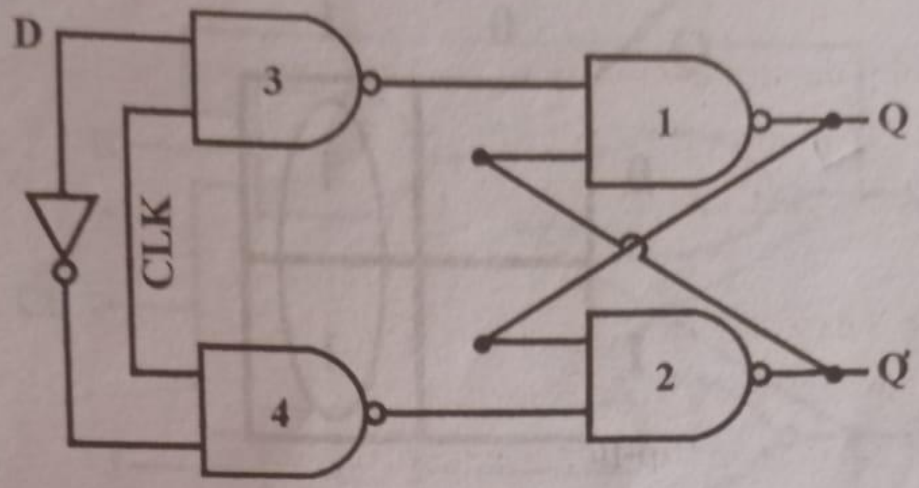

Fig.2.14(a) D-FF Logic Diagram

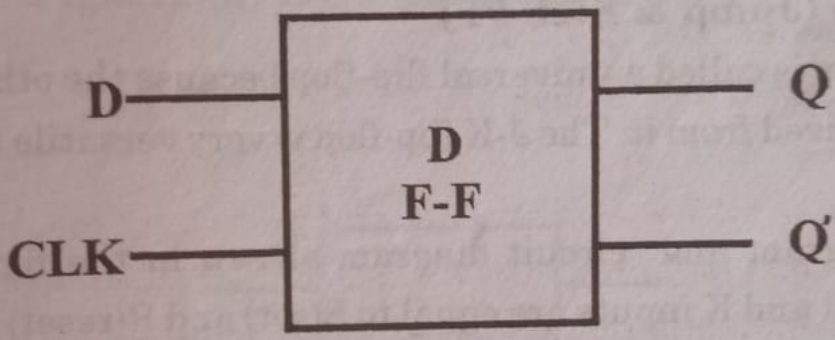

Fig.2.14(b) D-FF Block Diagram

Table - 2.7 : Truth Table for D-FF

Clock

Q(t) Present State	D	Q(t+1) Next State
0	0	0
0	1	1
1	0	0
1	1	1

➢ **K-MAP for D-Flip Flop**

As shown in the map one pair is possible, hence it removed variable Q which changed from unprimed to prime.

Hence solution is Q(t+1) = D

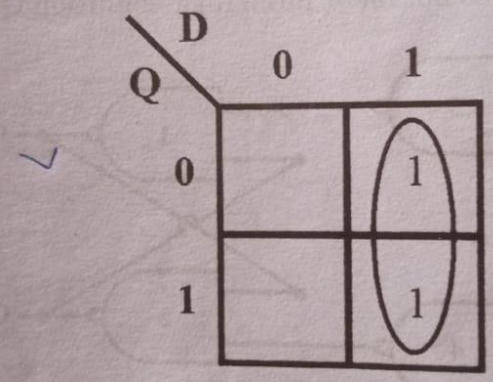

Fig. 2.15 : K-Map for D-FF

4) J-K Flip Flop (Jump & Kick FF)

- ✓ The J-K FF is called a universal flip-flop because the other flip-flop like D, R-S, and T can be derived from it.
- ✓ The J-K flip-flop is most widely used flip-flop.
- ✓ The J and K inputs are equal to S(set) and R(Reset) inputs of the R-S flip-flop.

➢ **Working of J-K Flip flop**

- ✓ As shown in truth table, if both input J and K equal to zero, then no change of state take place even if a clock pulse is applied.
- ✓ The second condition J=0 and K=1, causes flip-flop resets.
- ✓ When J=1 and K=0, the flip-flop sets.
- ✓ When both the inputs J=K=1, the flip-flop switches to its compliment state, i.e. if Q=1, it switches to Q=0 and vice versa.

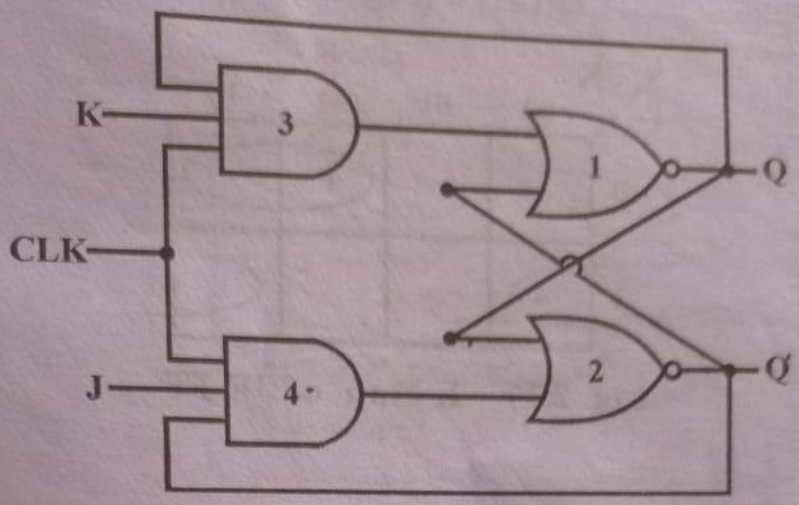

Fig.2.16 (a) Logic Diagram of JK FF

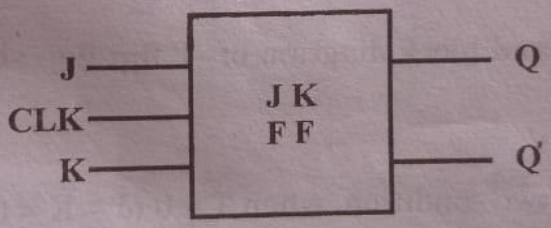

Fig.2.16 (b) Block Diagram of JK FF

Table - 2.8 : Truth Table for JK FF

J	K	Q(t+1)	
0	0	Q(t)	No change
0	1	0	Clear to 0
1	0	1	Set to 1
1	1	Q'(t)	Complement

5) T – Flip – Flop(Toggle FF)

- ✓ This flip flop has a single control input, labeled as T for toggle.
- ✓ These flip-flop are not widely available, but it is easy to construct from J-K flip flop.
- ✓ T flip flop is obtained by combining J and K inputs of J-K flip flop.
- ✓ This input is then referred as T or Toggle.
- ✓ T- FF, has only two condition, When T = 0(J=K=0), clock transition does not change the state of flip flop but when T = 1, clock transition compliments or toggles the state of flip-flop.
- ✓ It means when :

  T = 0 : Q (t+1) = Q(t)

  T = 1 : Q (t+1) = Q'(t)

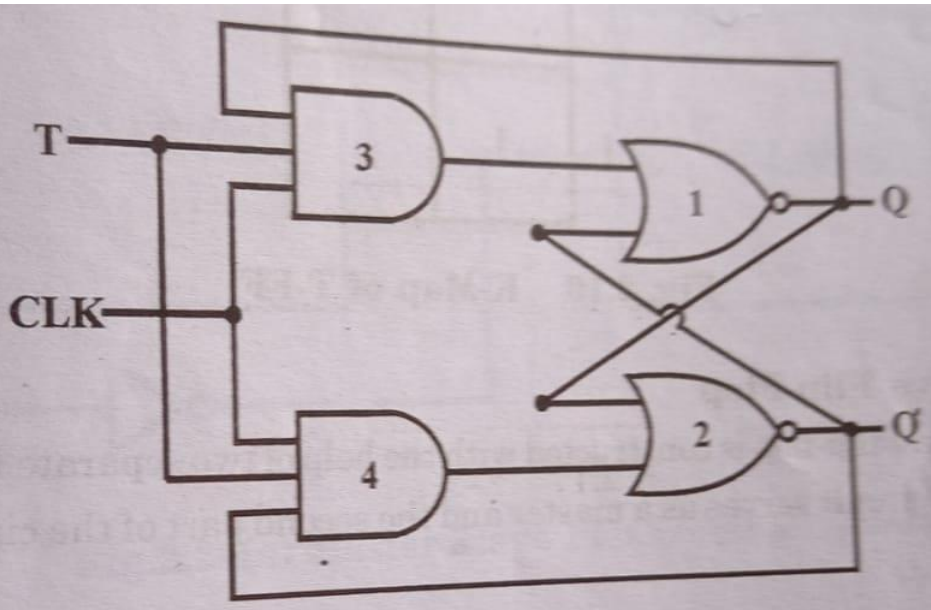

Fig.2.18 (a) Logic Diagram of T-FF

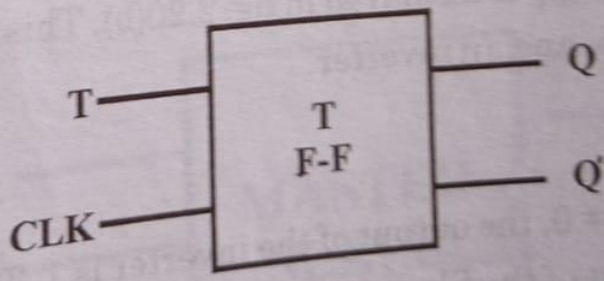

Fig.2.18 (b) Block Diagram of T-FF

Table - 2.9 : Truth Table for T-FF

Q(t)	T	Q(t+1)
0	1	1
0	0	0
1	0	1
1	1	0

➢ **K – Map for T- FF**

✓ As shown in below figure, bot cell marked '1' are diagonally opposite, therefore they are not considered adjacent. The function obtained from K-Map is Q(t+1)= TQ' + T'Q

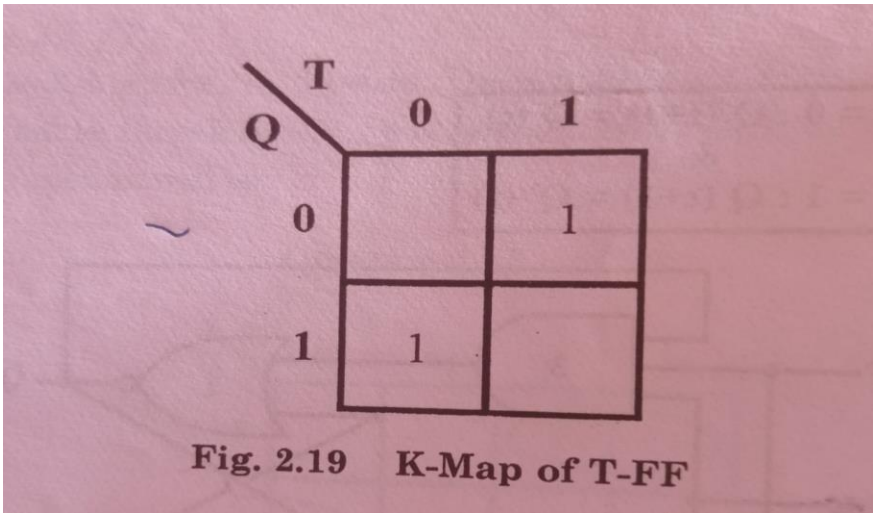

Fig. 2.19 K-Map of T-FF

6) Master Slave Flip – Flop

✓ A Master – Slave flip-flop is constructed with the help of two separate flip-flop circuits.

✓ First pair of circuit servers as a master and the second part of the circuit served as a slave flip-flop.

➢ **Operation**

✓ When clock pulse CP=0, the output of the inverter is 1. This is applied to Slave flip-flop. Since the clock input of the slave is now 1. The flip-flop is enabled and output Q is equal to Y, While Q' is equal to Y'. The master flip-flop will remain during this time as the clock pulse CP=0.

✓ When the clock pulse again goes to 1, the information then at the external J & K input is transmitted to the master flip-flop and thus Y and Y' get the values corresponding to the inputs of J-K values.

✓ The Slave flip-flop, however is isolated as long as the clock pulse is at its 1 level, because the output of the inverter is zero. When the clock input of the slave is 1, the flip- flop is enabled and output Q is equal to Y and Q' is equal to Y'.

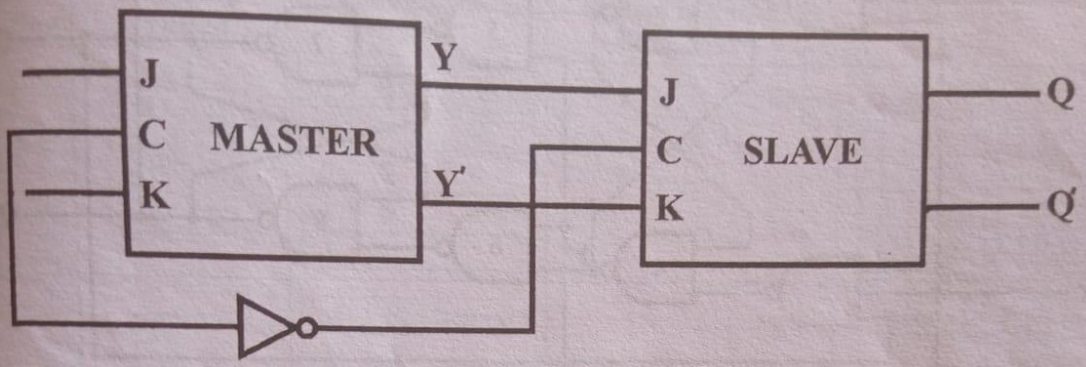

Fig.2.20(a) Master Slave FF Circuit Diagram.

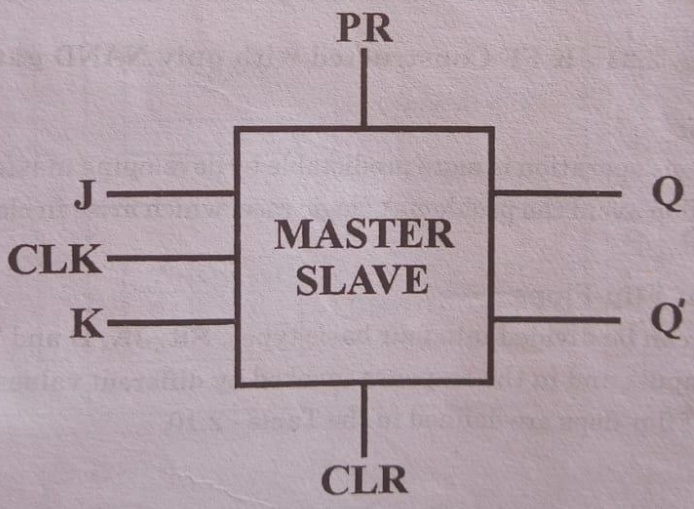

Fig.2.20(b) Block Diagram of Master Slave FF