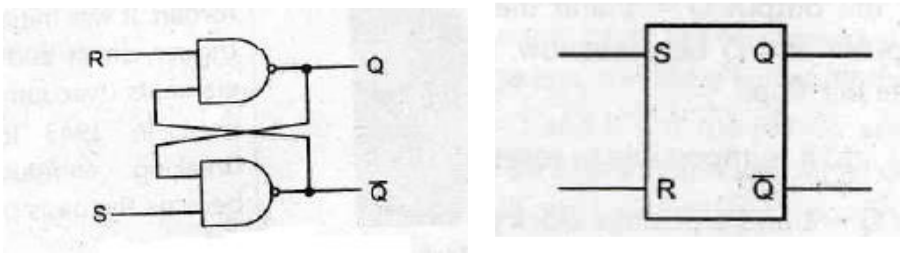
FLIP-FLOPS

The basic digital memory circuit is known as FLIP-FLOP. It has two stable states which are known as the 1 state and the 0 state. It can be obtained using NAND or NOR gates

R-S FLIP-FLOP



(a) (b)

Fig.1: (a) R-S FLIP-FLOP (b) Symbol

An R-S Flip-flop can be constructed by using two NAND gates and connecting output of one to the one of the inputs of other as shown in Fig.1(a). Fig.1(b)shows symbol of R-S flip-flop.

Case1: If S=R=0. The S=0 will force $\bar{Q}$ =1 and R=0 will force Q =1 Therefore Q and $\bar{Q}$ are equal which is meaningless as Q and $\bar{Q}$ are compliments of each other. Hence this combination is not allowed.

Case2: If S=0 and R=1 .When S=0, it means that one of the inputs to the lower NAND gate is 0 which makes $\overline{Q} = 1$. Now the inputs to the upper NAND gates are 1 and 1, which make $\overline{Q} = 0$. The flip-flop is reset to 0.Thus for S=0 and R=1, irrespective of the previous output, Q=0 and $\overline{Q} = 1$.

Case3: If S=1 and R=0 .When R=0, it means that one of the inputs to the upper NAND gate is 0 which makes Q=1. Now the inputs to the lower NAND gates are 1 and 1, which make $\overline{Q} = 0$. The flip-flop is set to 1.Thus for S=1 and R=0, irrespective of the previous output, Q=1 and $\overline{Q} = 0$.

Case4: If S=1 and R=1 .The output will not change and the flip-flop will either be set or reset.

Inputs		Outputs		
S	R	Q_{n+1}	Q_{n+1}	
0	0	1	1	Invalid
0	1	0	1	Reset
1	0	1	0	Set
1	1	Q_n	Q_n	No change, Same as previous state

Table: Truth Table of R-S Flip-Flop

R-S FLIP-FLOP with PREST AND CLEAR inputs

In the FLIP -FLOP, when the power is switched on, the state of the circuit is uncertain. IT may come to set (Q=1) or reset (Q=0) state. In many applications it is desired to initially set or reset the FLIP -FLOP, i.e. the initial state of the FLIP- FLOP is to be assigned. This is done by using the direct or asynchronous inputs, referred to as *preset* (Pr) and *clear* (Cr) inputs. These inputs may be applied at any time between clock pulses and are not in synchronism with the clock. An S-R FLIP –FLOP with preset and clear is shown in Fig.1

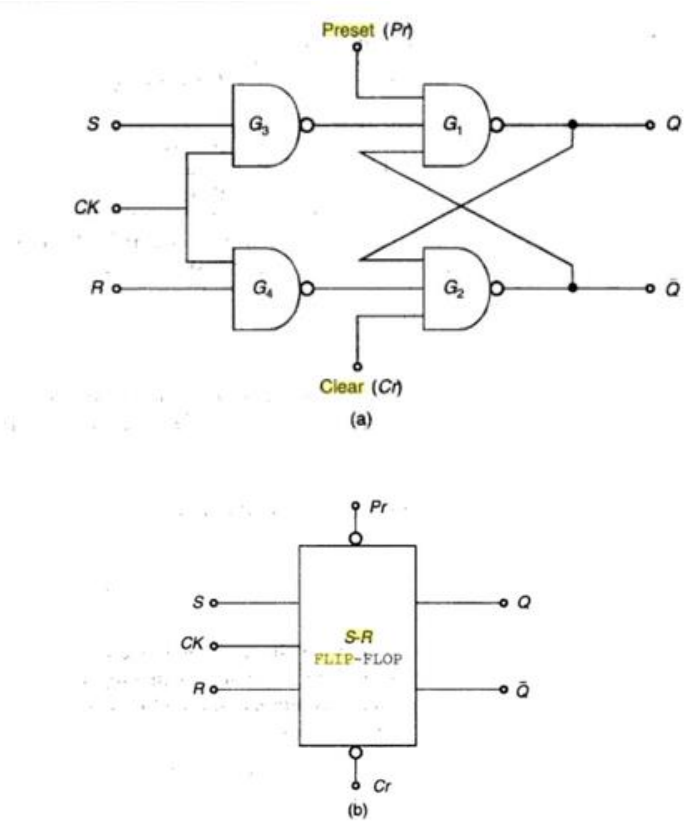


Fig.1: [a] An S-R FLIP –FLOP with preset and clear [b] Its logic symbol

If Pr=Cr=1, the circuit operates in accordance with the truth table

Inputs		Outputs
S	R	Q_{n+1}
0	0	Q_n
1	0	1
0	1	0
1	1	?

If Pr=0 and Cr=1, the output of G_1(Q)will be 1.Consequently, all the three inputs to G_2 will be 1 which will make $\bar{Q}$ =0, Hence, making Pr=0 sets the flip-flop.

Similarly, if Pr=1 and Cr=0, the flip-flop is reset.

The condition Pr=0 and Cr=0 must not be used, since this leads to an uncertain state.

In the logic symbol of fig.1 (b), bubbles are used for Pr and Cr inputs, which mean these are active low, i.e. the intended function is performed when the signal is applied to Pr or Cr is LOW. The operation of Fig.1 is summarized in following table.

Inputs			Output	Operation performed
CK	Cr	Pr	Q	
1	1	1	Q_{n+1}	Normal flip-flop
0	0	1	0	Clear
0	1	0	1	preset

Que: Using logic diagram and truth table, explain clocked R-S flip-flop with preset and clear inputs.

J-K FLIP-FLOP

The uncertainty in the state of an S-R FLIP-FLOP when S=R= 1 can be eliminated by converting it into a J-K FLIP-FLOP. The data inputs are J and K which are ANDed with $\bar{Q}$ and Q respectively, to obtain S and R inputs, i.e.

$$S = J.\bar{Q}$$

$$R = K.Q$$

A J-K FLIP-FLOP thus obtained is shown in Fig.1.Its truth table is given in Table 1(a) which is reduced to Table 1(b) for convenience. Table 1(a) has been prepared for all the possible combination s of J and K inputs, and for each combination both the states of the outputs have been considered.

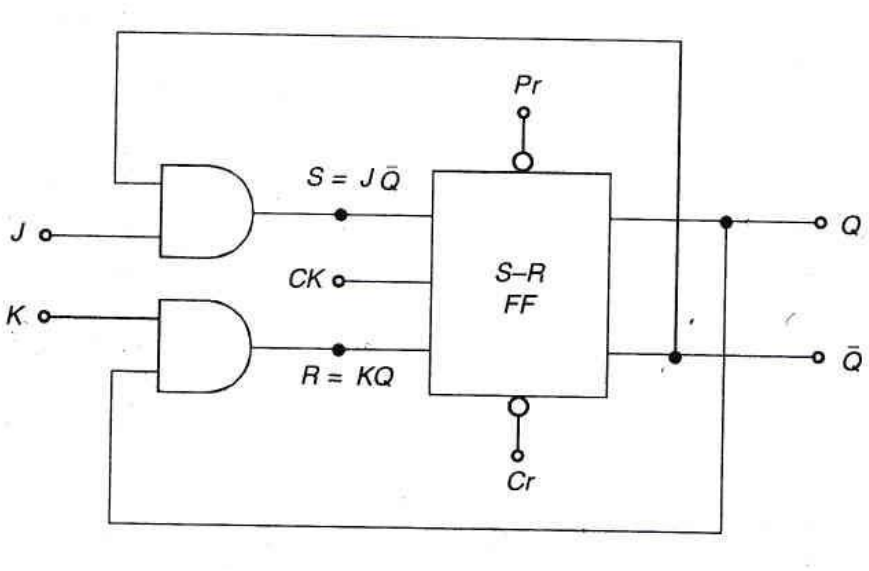


Fig.1: An S-R FLIP-FLOP converted into J-K FLIP-FLOP

Data inputs		*Outputs*		*Inputs to S–R FF*		*Output* Q_{n+1}	
J_n	K_n	Q_n	$\bar{Q}_n$	S_n	R_n		
0	0	0	1	0	0	0	$= Q_n$
0	0	1	0	0	0	1	
1	0	0	1	1	0	1	$= 1$
1	0	1	0	0	0	1	
0	1	0	1	0	0	0	$= 0$
0	1	1	0	0	1	0	
1	1	0	1	1	0	1	$= \bar{Q}_n$
1	1	1	0	0	1	0	

Table 1(a): Truth Table for Fig.1

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

Table 1(b): Truth Table of J-K FLIP-FLOP

It is not necessary to use the AND Gates of Fig.1, since the same function can be performed by adding an extra input terminal to each NAND gate G_3 and G_4 of S-R FLIP –FLOP . With this modification incorporated, we obtain the J-K FLIP-FLOP using NAND Gates as shown in Fig.2.

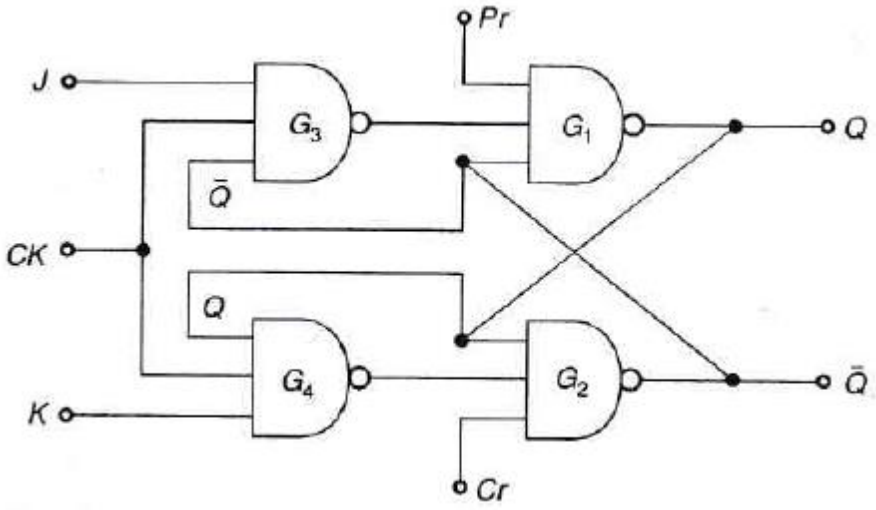


Fig.2: A J-K FLIP-FLOP using NAND Gates

The logic symbol of J-K FLIP-FLOP shown in Fig.3

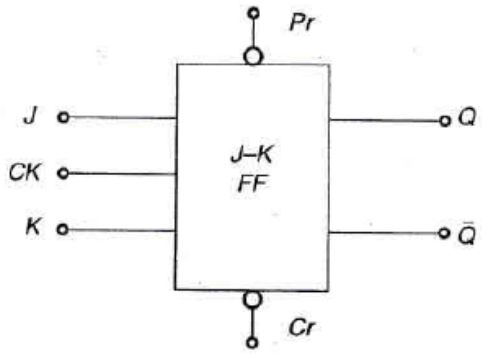


Fig.3: Logic symbol of J-K FLIP-FLOP

D TYPE FLIP-FLOP

D type FLIP-FLOP has only one input referred as D-input or data input.Fig1 shows D type FLIP-FLOP. Its truth table is given in Table 1 from which it is clear that the output Q_{n+1} at the end of clock pulse equals the input D_n before the clock pulse.

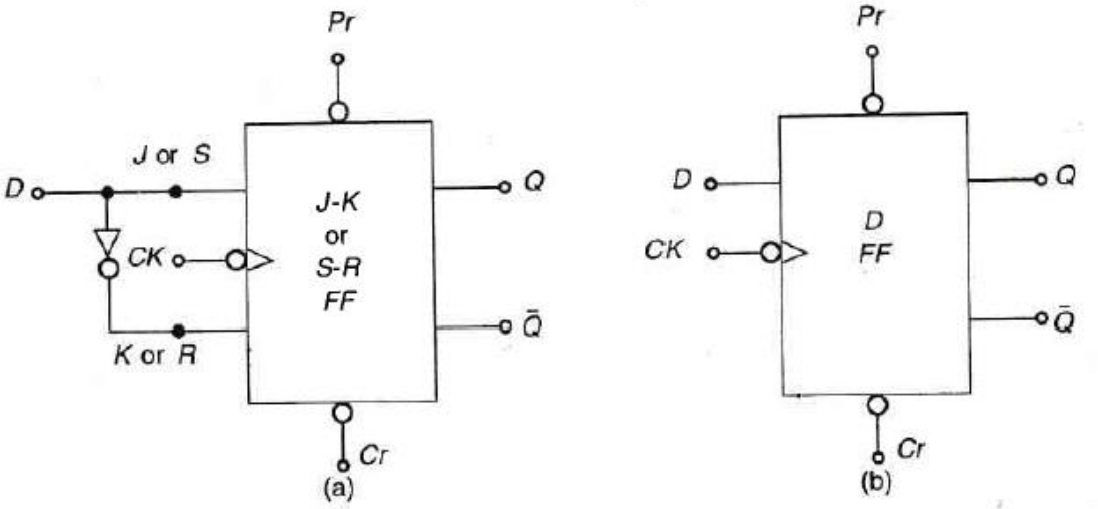


Fig1: (a) A J-K FLIP-FLOP or S-R FLIP-FLOP converted into a D type FLIP-FLOP

(b) D type FLIP_FLOP logic symbol

Input D_n	Output Q_{n+1}
0	0
1	1

Table 1: Truth Table of D type FLIP-FLOP

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 the name *delay* FLIP-FLOP. The d- type FLIP-FLOP is either used as a delay device or as a latch to store 1-bit of binary information.

T- TYPE FLIP-FLOP

In a J-K FLIP-FLOP, if J=K , the resulting FLIP-FLOP is referred to as a T-type FLIP-FLOP and is shown in Fig.1. It has only one input, referred to as T-input. Its truth table is given in Table 1 from which it is clear that if T=1 it acts as a toggle switch. For every clock pulse, the output Q changes.

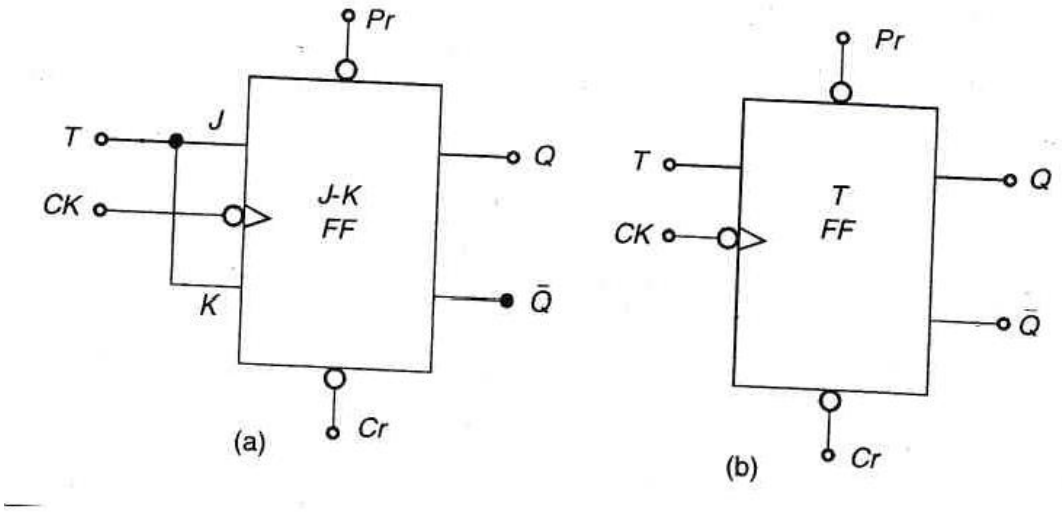


Fig.1: (a) A J-K FLIP-FLOP converted into T-type FLIP-FLOP

(b) Logic symbol

Input T_n	Output Q_{n+1}
0	Qn
1	$\bar{Q}n + 1$

Table 1: Truth Table of T type FLIP- FLOP

An S-R FLIP- FLOP cannot be converted into a T type FLIP- FLOP since S=R=1 is not allowed.