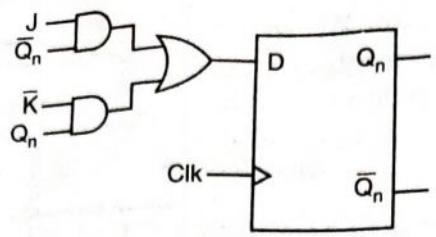


4.21. REGISTER

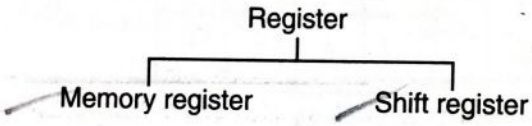
It is a group of flip flops and is used for storing binary information. Flip-flop stores one bit of information and is called as binary cell or 1-bit memory is called as binary cell or 1-bit memory cell. In this way—

An '*n*' bit register is basically a group of a n-flip-flops and is capable of storing *n*-bit information.

Applications of register

— It is used in memories
— Used in microprocesors.

Register can be classified into two categories



Memory Register

A register that is used to store binary information is known as memory register.

Shift Register

A register that is used to shift binary information either to right or to the left is called as shift register.

Shift Register

Shift Register is a type of *Sequential circuit* (means which has memory) that is formed by combination of flip flops and is capable of *shifting data from left to right or right to left.*

Shift register performs two basic functions
(*i*) Shifting of Data (Transfer of Data)
(*ii*) Storage function

Shifting of Data

It is defined as the movement of data from stage to stage (left to right or right to left) within the register or into or out of the register upon the application of clock.

Storage Function

It is defined as the number of bits (1's or 0's) of digital data it can retain. The storage function will depends upon the number of flip flops which are connected to form register *e.g.* if three flip flops are connected together, then register is capable of storing 3-bits. So we can say that if '*N' flip flops* are there, then it can store '*N' bits*. Due to this storage feature, it is an important memory device. Let us take an *e.g.* showing the concept of storage feature :

Case I. *When '1' bit is applied to the input* and clock pulse is applied that stores the '1' and hence flip flop is said to be set. When '1' on the input is removed, the flip flop will remains in the *set state*, storing the '1'. It is shown as below:

So when D = '1', then Q = '1' means '1' bit is stored when clock occurs or remains a '1' if already in that state.

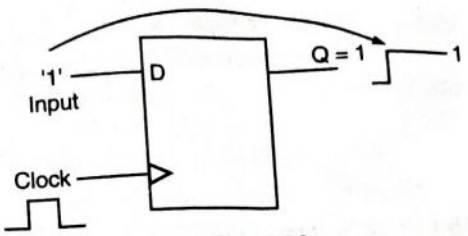


Fig. 4.82.

Case II *When '0' bit is applied to the input* and clock pulse is applied that stores the '0' and hence flip flop is said to be in *Reset state*. When '0' on the input is removed, the flip flop will remains in the *Reset* state, storing the '0' bit. It is shown as below:

So when D = '0', then Q = '0' means '0' bit is stored when clock occurs or remains a '0' if already in that state.

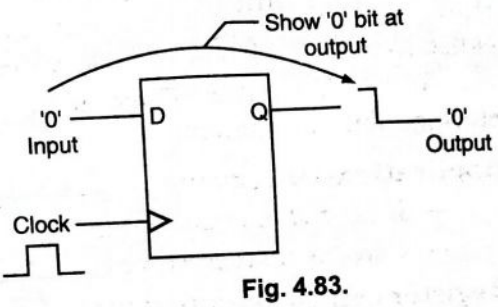


Fig. 4.83.

4.22 BASIC CONCEPTS

It deals with the functions performed by the shift Register.

Shift Register are commonly used for temporary storage of data with in the digital system. Registers are formed by flip flops. *The shifting of data is the process of movement of bits into or out of the register with the help of clock input.* The classification of data shifting are:

(*i*) Serial shift right
(*ii*) Serial shift left
(*iii*) Parallel shift in
(*iv*) Parallel shift out
(*v*) Rotate right
(*vi*) Rotate left.

Generally shifting involves two process *i.e.*

— Serial shift
— Parallel shift

Serial shift means bits are transferred/shifted one by one according to clock pulse *e.g.* for four bits, '4' clock pulses are required to shift.

Parallel shift means bits are transferred simultaneous means if '4' bits are there, then '4' bits are transferred at the same time and in single clock pulse.

Various data Shifting Methods

(*i*) Serial Shift Right

Here if we have '3' bits, then right shift means the movement of data in right direction as shown in figure 4.84 (a).

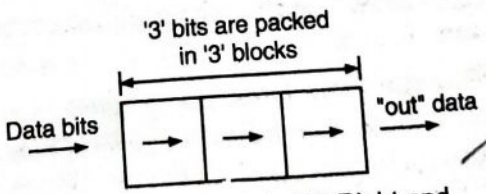


Fig. 4.84. (a) Serial shift Right and than out

Here arrows indicate type of movement (serial or parallel) the movement of bits in right direction, the '3' blocks are used to indicate '3' bits.

(ii) Serial Shift Left

Here if were have '3' bits, then shift left means the movement of data in left direction as shown in figure 4.85 (*b*).

'3' bits are packed in '3' blocks.

Here the arrow indicate the direction and type of data movement (serial or parallel)

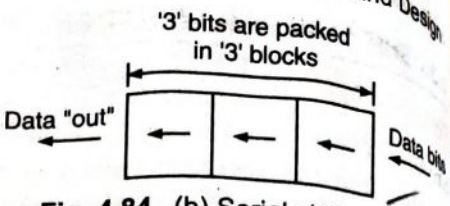


Fig. 4.84. (b) Serial shift left and then out

(iii) Parallel shift in

Here the data is entered parallelly means simultaneously. It is represented as in figure 4.85.

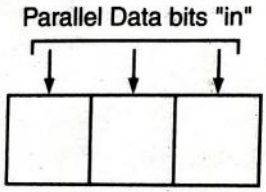


Fig. 4.85.

Here '3' bits are entered in parallel order.

(iv) Parallel shift out

Here the data bits are out in parallel mode. It is shown as follows:

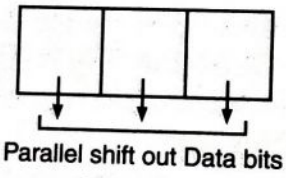


Fig. 4.86.

Here '3' bits are out in parallel mode. The arrow indicate the direction and type of data movement.

(v) Rotate right

Here the data bits are shifted in right direction and keeps on circulating as shown in figure 4.71.

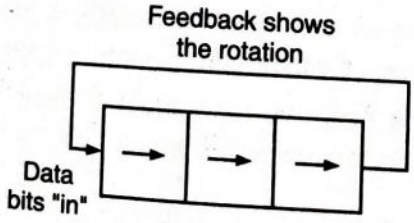


Fig 4.87. Rotate right

Here the feedback is used to rotate the bits and the arrow indicate the direction of movement.

(vi) Rotate left

Here the data bits are shifted in left direction and keeps on circulating as shown in fig.

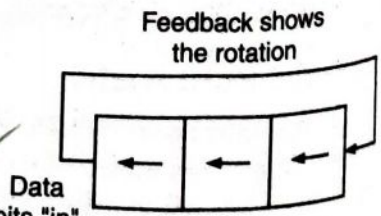


Fig. 4.88. Rotate left

Here the feedback is used to rotate the bits in left direction and arrows are used to indicate the direction of movement.

4.23 SHIFT LEFT AND SHIFT RIGHT CONCEPTS

These concepts are used for shifting the data into the register.

Shift Left

Here the data bits are shifted into the left direction. It is shown as follows:

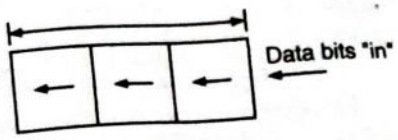


Fig. 4.89. Shift left

Here the '3' bits are shifted in left direction as shown by arrow.

Left us take an e.g. 1.0 $\overset{(MSB)}{\underset{1}{D_2}}$ $\underset{1}{D_1}$ $\overset{(LSB)}{\underset{0}{D_0}}$ means '3' bits i.e. 110 are shifted in left, then in first clock cycle 'C_1', D_0 is passed i.e. 'O' is fed i.e. $D_0 = 0$

Then in next (second) clock cycle 'C_2', D_1 is fed into the flip flop *i.e.* $D_1 = 1$

Then in third clock cycle 'C_3', D_2 is fed into flip flop i.e., $D_2 = 1$.

So, we can say that '3' clock pulses are required to fed the '3' data bits if it is serial shift left and data should be outered or 'in' from least significant bit (LSB or D_0 as in example).

4.23.1. Shift Left Register Circuit

Shift left register can be formed using *D* and JK flip flop.

Let us draw the shift left register using *D* flip flop first, *i.e.*

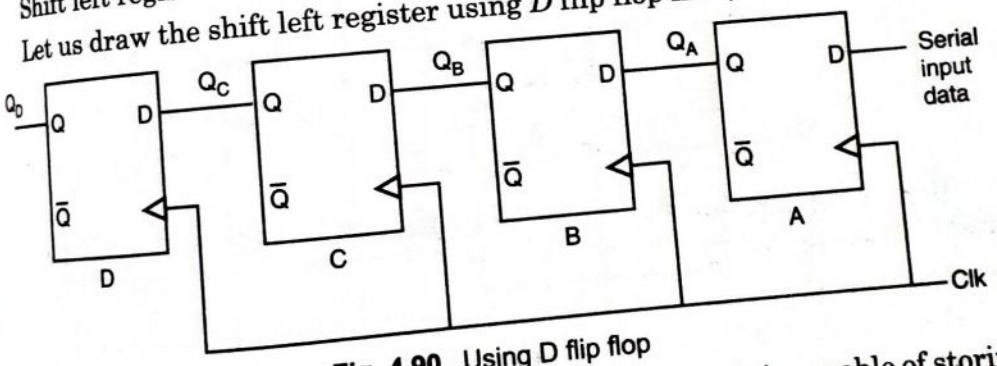


Fig. 4.90. Using D flip flop

Here '4' *D* flip flops are connected together and hence it is capable of storing '4' bits. Suppose we want to pass 1100 into the register, it begins with right most bit. '0' bit is applied to serial input data, then flip flop A is *reset*, thus storing '0' bit means Q_A is '0'. Next '0' is applied to the serial input, making D = '0'/or flip flop A and $D = 0$ for flip flop B, because the input of flip flop B is connected to Q_A output.

When the second clock pulse occurs, the '0' data input is shifted to the flip flop A and the '0' in the flip flop A is shifted to flip flop B. The '0' in the binary number is now applied at the serial input line and the third clock pulse is now applied. Then at third clock pulse '1' is applied at flip flop A and the '0' stored in flip flop is shifted to flip flop B and the '0' stored in flip flop B is shifted to flip flop C. The last bit *i.e.* '1' is now applied at the serial input line and fourth clock pulse is applied. This '1' enters into A flip flop and the '1' stored in A flip flop is shifted to 'B' flip flop and the '0' stored in 'B' flip flop is shifted to C flip flop and the '0' stored in C flip-flop is shifted to *D* flip flop. Thus entry of '4' data bits in shift left is carried out.

Operation of shift – Left Register

	Q_D	Q_C	Q_B	Q_A	Serial output at Q_D
Initial value	0	0	0	0	0
1st Clock pulse	0	0	0	0	0
2nd Clock pulse	0	0	0	0	0
3rd Clock pulse	0	0	0	1	0
4th Clock pulse	0	0	1	1	0

The timing diagram or waveform is as shown below.

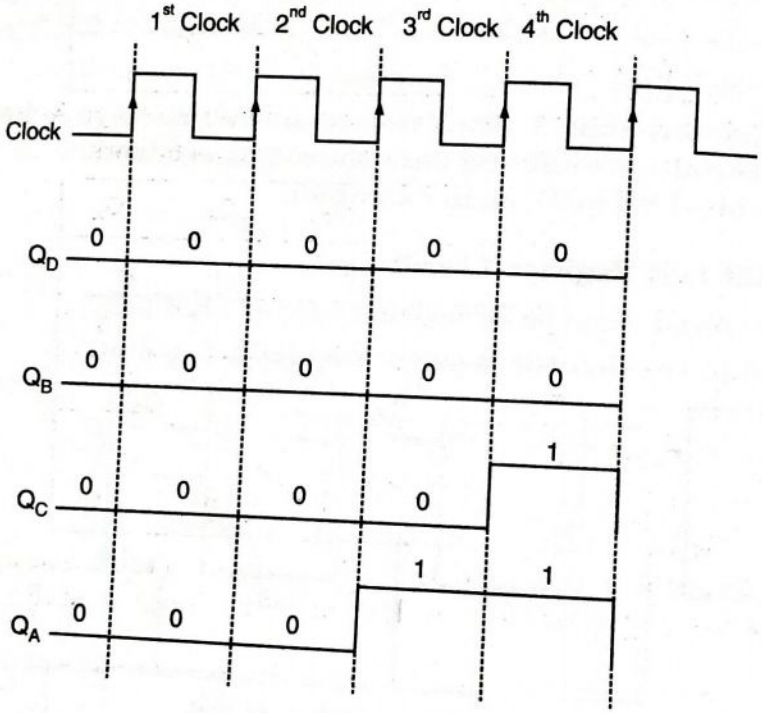


Fig. 4.91.

The designing of serial left register using JK flip flop is shown as follows:

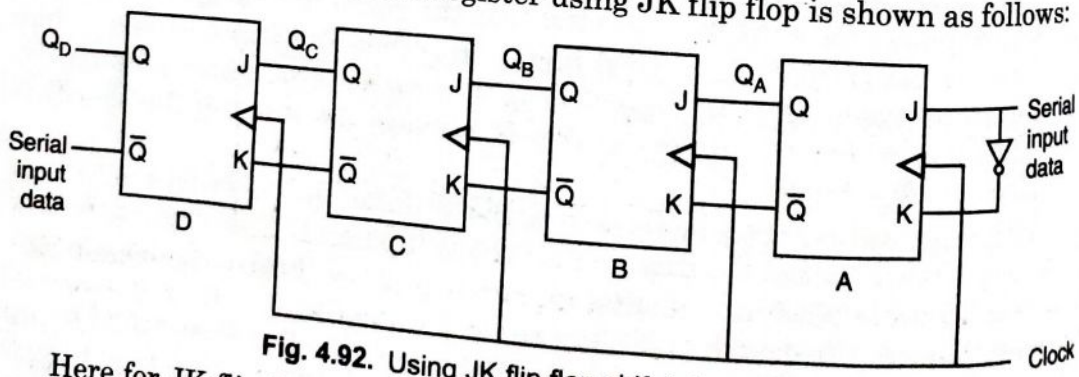


Fig. 4.92. Using JK flip flop shift left register

Here for JK flip flop, it require connection to both J–K inputs. Input data are connected to the J and K in puts of the right most (LSB) of flip flop. In *order to input '1', the inputs at J = 1 & K = 0* (as NOT gate is connected to K input) and

for input '0', J = '0' & K = '1' should be applied and when clock pulse is applied, the data will be shifted bit by bit to the left.

e.g. If all stages are reset and steady '1' is applied at the serial input data connected to stage A. The data after four shift pulses are given as follows. Here the logical- 1 input enters into stage A & then shifts left to stage D after four shift pulses as showing.

	Q_D	Q_C	Q_B	Q_A
Initial value	0	0	0	0
1st Clock pulse	0	0	0	1
2nd Clock pulse	0	0	1	1
3rd Clock pulse	0	1	1	1
4th Clock pulse	1	1	1	1

In case 2, the shifting of alternate 0 and 1 data into stage A, starting with all stages are set i.e. 1111. These stages are shown as follows:

	Q_D	Q_C	Q_B	Q_A
Initial value	1	1	1	1
1st Clock pulse	1	1	1	0
2nd Clock pulse	1	1	0	1
3rd Clock pulse	1	0	1	0
4th Clock pulse	0	1	0	1

In **case 3,** consider starting with 0110 as shown in table and applying four more shift pulses but placing logical 'O' at serial input to stage A as shown below:

	Q_D	Q_C	Q_B	Q_A
Initial value	0	1	1	0
1st Clock pulse	1	1	0	0
2nd Clock pulse	1	0	0	0
3rd Clock pulse	0	0	0	0
4th Clock pulse	0	0	0	0

Note: In order to shift the bits, it always start with least significant bit (LSB) *i.e.* A is LSB & D is MSB in the particular example.

4.23.2. Shift Right

Here the data bits are shifted into the right direction. It is shown as follows:

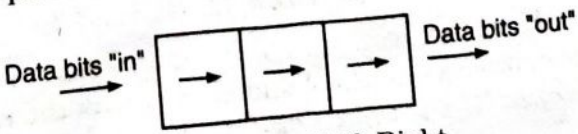


Fig. 4.93. Shift Right

Here the three bits are shifted in right direction as shown in fig. (by using arrows) . Let us take an example i.e. $\underset{1}{\overset{(MSB)}{D_2}}$ $\underset{0}{D_1}$ $\underset{1}{\overset{(LSB)}{D_0}}$. Here in first clock cycle, the D_2 is fed *i.e.* D_2 = '1'

In second clock cycle, the D_1 is fed *i.e.*, D_1 = '0'.

Then in third clock cycle, the D_0 is fed *i.e.* D_0 = '1'

So, we can say that '3' clock pulses are required to fed the '3' data bits if it is serial shift right and data should be entered or "in" through MSB (Most significant bit i.e. D_2).

Shift Right Circuit

It can be formed using *D* or JK flip flop. The circuit formed using *D* flip flop is shown below:

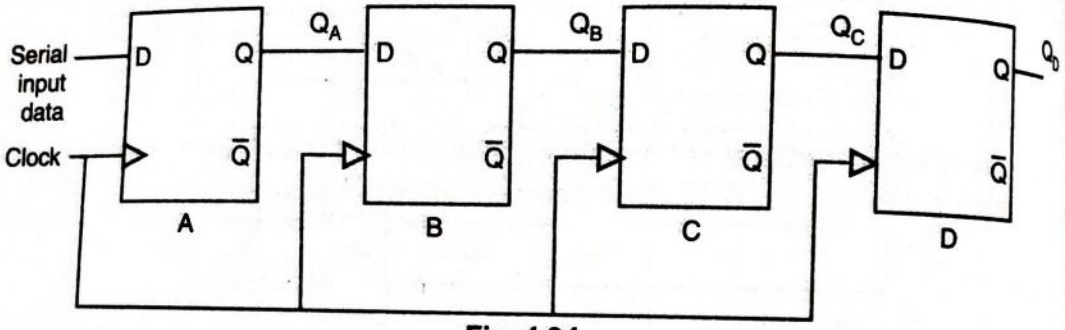


Fig. 4.94.

In this shift register, the least significant bit (LSB) of Data input is fed first means least significant bit (LSB) acts as serial Input data. The clock pulse is applied to all the flip flop simultaneously.

Let us take an *e.g. i.e.* 1011 is passed, then LSB bit *i.e.* '1' is passed to serial input data & at 'A' flip flop, this bit '1' is passed to *D* and at first clock pulse, flip flop A is *set,* thus storing the '1'. Then next bit *i.e.* '1' is applied to serial input making *D* = 1 for flip flop A & *D* = 1 for flip flop B because the input of B flip flop is connected to Q_A output.

When the second clock pulse arrives, the '1' on the data input is shifted to the flip flop A & then '1' on A is shifted to B and '0' bit is applied at data input and the third clock pulse is now applied. The '0' is entered in flip flop A & the '1' stored in '*A*' is shifted to B and the value stored on B i.e. '1' is shifted to 'C' flip flop. Then fourth clock pulse is applied and '1' bit is passed to data input and the '0' stored in flip flop A is now shifted to *B* and the '1' stored in flip flop *B* is now shifted to flip flop *C* and the '1' stored in '*C*' is shifted to '*D*' flip flop.

The operation is given as follows:

	Q_A	Q_B	Q_C	Q_D	Serial output at Q_D
Initial value	0	0	0	0	0
1st Clock pulse	1	0	0	0	0
2nd Clock pulse	1	1	0	0	0
3rd Clock pulse	0	1	1	0	0
4th Clock pulse	1	0	1	1	1

Waveform : The operation is shown in the table, from there we can draw the waveform of outputs Q_A, Q_B, Q_C & Q_D as shown below:

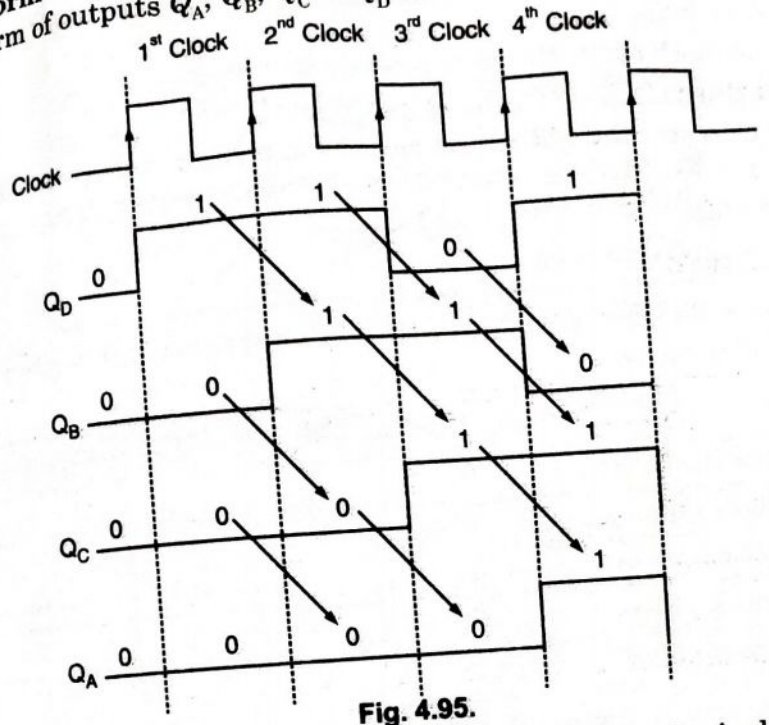


Fig. 4.95.

The designing of the serial right register using J-K flip flop is shown as below:

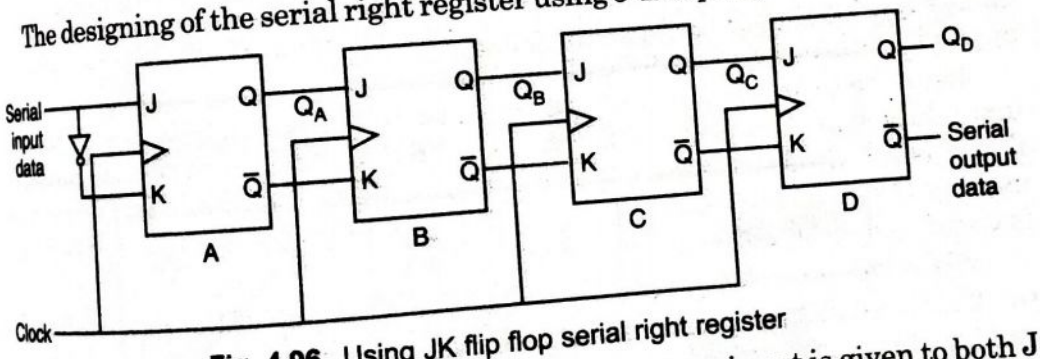


Fig. 4.96. Using JK flip flop serial right register

Here the data is entered from the LSB bit. The data input is given to both J-K inputs. In order to pass input '1', the J = '1' & K = '0' is passed and to pass input '0', the value of J = '0' and Ks = '1' .

The truth table showing the data input 1101 is given as below:

	Q_A	Q_B	Q_C	Q_D
Initial value	0	0	0	0
1st Clock pulse	1	0	0	0
2nd Clock pulse	0	1	0	0
3rd Clock pulse	1	0	1	0
4th Clock pulse	1	1	0	1

The truth table showing the action of shifting all logical - 1 inputs into initially reset shift register as below :

	Q_A	Q_B	Q_C	Q_D
Initial value	0	0	0	0
1st Clock pulse	1	0	0	0
2nd Clock pulse	0	1	0	0
3rd Clock pulse	1	0	1	0
4th Clock pulse	1	1	0	1

Suppose we want to pass 1011 to JK base shift right shift register, the truth table is shown as follows:

	Q_A(LSB)	Q_B	Q_C	Q_D (MSB)
Initial condition	0	0	0	0
Ist clock pulse	1	0	0	0
2nd clock pulse	1	1	0	0
3rd clock pulse	0	1	1	0
4th clock pulse	1	0	1	1

Here when 1011 is applied at J then K is 0100 because NOT gate is applied to K.

So, the waveform is shown as below.

	Q_A	Q_B	Q_C	Q_D
Initial value	0	0	0	0
1st Clock pulse	1	0	0	0
2nd Clock pulse	1	1	0	0
3rd Clock pulse	1	1	1	0
4th Clock pulse	1	1	1	1

Fig. 4.97.

4.24 TYPES OF SHIFT REGISTER

In shift register the data is shifted in and shifted out from the register. The shifting process involved to two process i.e. *Serial or parallel shift.* According to these combination shift register are of following types: