MODULE 2

SEQUENTIAL LOGIC CIRCUITS

Sequential circuits are constructed using combinational logic and a number of memory elements with some or all of the memory outputs fed back into the combinational logic forming a feedback path or loop.

Sequential circuit = Combinational logic + Memory Elements

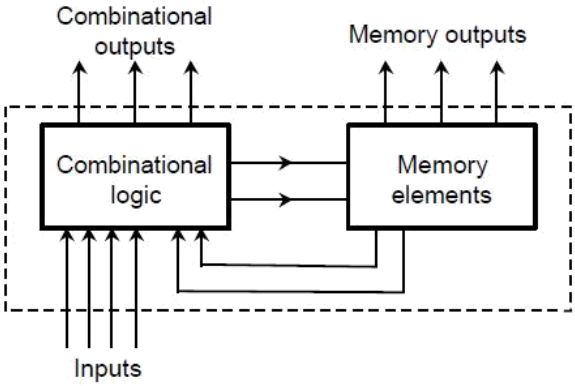


Fig 3.45 sequential circuit

- There are two types of sequential circuits:
  - *synchronous*: outputs change only at specific time
  - *asynchronous*: outputs change at any time

A state variable in a sequential circuit represents the single-bit variable Q stored in a memory element in circuit.

– Each memory element may be in state 0 or state 1 depending on the current value stored in the memory element.

• The State of A sequential Circuit:

– The collection of all state variables (memory element stored values) that at any time contain all the information about the **past necessary to account for the circuit's future behavior.**

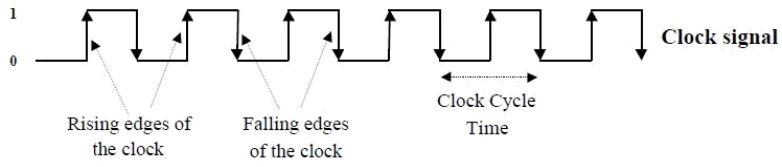
– A sequential circuit that contains n memory elements could be in one of a maximum of 2n states at any given time depending on the stored values in the memory elements.

– Sequential Circuit State transition: A change in the stored values in memory elements thus changing the sequential circuit from one state to another.

Clock Signals & Synchronous Sequential Circuits

• **A clock signal** is a periodic square wave that indefinitely switches values from 0 to 1 and 1 to 0 at fixed intervals.

- Clock cycle time or clock period: The time interval between two consecutive rising or falling edges of the clock.



Synchronous Sequential Circuits: Sequential circuits that have a clock signal as one of its inputs:

– All state transitions in such circuits occur only when the clock value is either 0 or 1 or happen at the rising or falling edges of the clock depending on the type of memory elements used in the circuit.

Sequential Circuit Memory Elements: Latches, Flip-Flops

• Latches and flip-flops are the basic single-bit memory elements used to build sequential circuit with one or two inputs/outputs, designed using individual logic gates and feedback loops.

• **Latches:**

– The output of a latch depends on its current inputs and on its previous inputs and its change of state can happen at any time when its inputs change.

• **Flip-Flop:**

– The output of a flip-flop also depends on current and previous input but the change in output (change of state or state transition) occurs at specific times determined by a clock input.

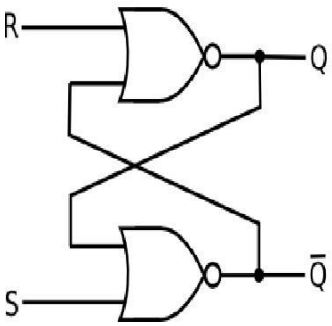
S-R Latch

• An S-R (set-reset) latch can be built using two NOR gates forming a feedback loop.

• The output of the S-R latch depends on current as well as previous inputs or state, and its state (value stored) can change as soon as its inputs change.

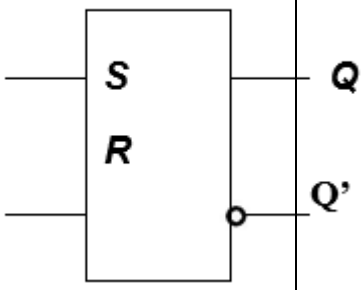
- When Q is HIGH, the latch is in SET state.

When Q is LOW, the latch is in RESET state.



Truth Table for S–R Flip-Flop

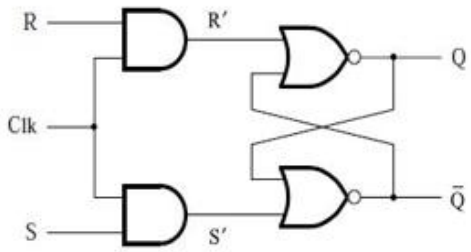
R	S	Q	Action
0	0	Last value	No change
0	1	1	Set
1	0	0	Reset
1	1	-	Invalid condition



S-R Flip flop

- Since the S-R latch is responsive to its inputs at all times an enable line C is used to disable or enable state transitions.

- Behaves similar to a regular S-R latch when enable C=1



(a) Circuit

Clk	S	R	Q(t + 1)
0	x	x	Q(t) (no change)
1	0	0	Q(t) (no change)
1	0	1	0
1	1	0	1
1	1	1	x

(b) Characteristic table

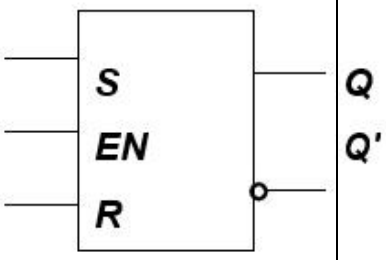


Fig 3.48 gated SR flip flop

S = 0, R = 0; this is the normal resting state of the circuit and it has no effect of the output states. **Q and Q'** will remain in whatever state they were in prior to the occurrence of this input condition. It works in **HOLD (no change)** mode operation. • **S = 0, R = 1;** this will reset **Q to 0**, it works in ***RESET*** mode operation.

S = 1, R = 0; this will set **Q to 1**, it works in **SET** mode operation.

S = 1, R = 1; this condition tries to set and reset the NOR gate latch at the same time, and it produces **Q = ¯ = 0.** This is an unexpected condition and are not used.

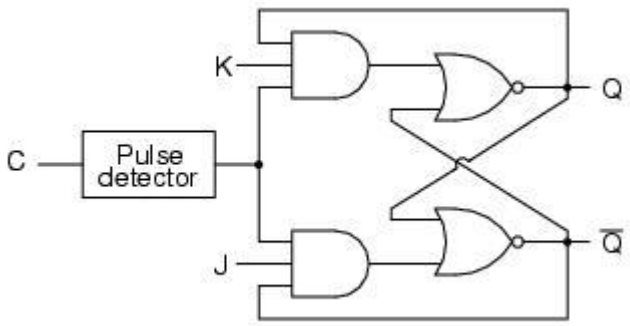
Since the two outputs should be inverse of each other. If the inputs are returned to 1 simultaneously, the output states are unpredictable. This input condition should not be used and when circuits are constructed, the design should make this condition SET=RESET = 1 never arises.

JK Flip Flop

Another types of Flip flop is JK flip flop.

- It differs from the RS flip flops when J=K=1 condition is not indeterminate but it is defined to give a very useful changeover (toggle) action.
- Toggle means that **Q** and Q ¯ will switch to their opposite states.
- The JK Flip flop has clock input Cp and two control inputs J and K.

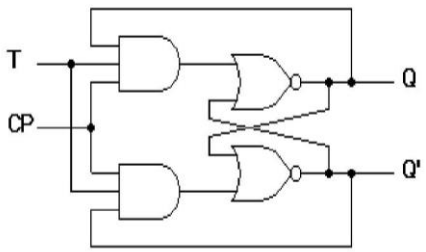
Operation of Jk Flip Flop is completely described by truth table



C	J	K	Q	$\overline{Q}$
⎍	0	0	latch	latch
⎍	0	1	0	1
⎍	1	0	1	0
⎍	1	1	toggle	toggle
x	0	0	latch	latch
x	0	1	latch	latch
x	1	0	latch	latch
x	1	1	latch	latch

T Flip Flop

The T flip flop has only the Toggle and Hold Operation. If Toggle mode operation. The output will toggle from 1 to 0 or vice versa.



Symbol

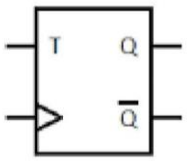


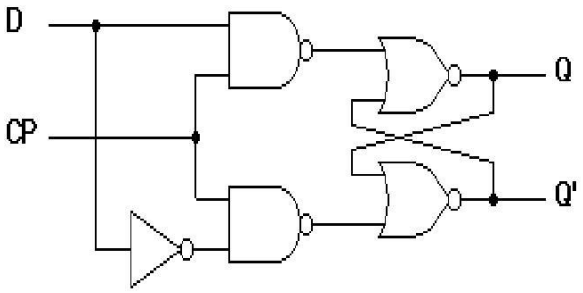
Table of truth:

T	Q	$\overline{Q}$
0	Q	$\overline{Q}$
1	$\overline{Q}$	Q
0	$\overline{Q}$	Q
1	Q	$\overline{Q}$

D Flip Flop

- Also Known as Data Flip flop
- Can be constructed from RS Flip Flop or JK Flip flop by addition of an inverter.
- Inverter is connected so that the R input is always the inverse of S (or J input is always complementary of K).

The D flip flop will act as a storage element for a single binary digit (Bit).



D Flip-flop

Symbol

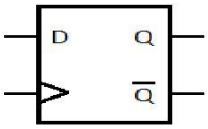


Table of truth:

clk	D	Q	$\overline{Q}$
0	0	Q	$\overline{Q}$
0	1	Q	$\overline{Q}$
1	0	0	1
1	1	1	0

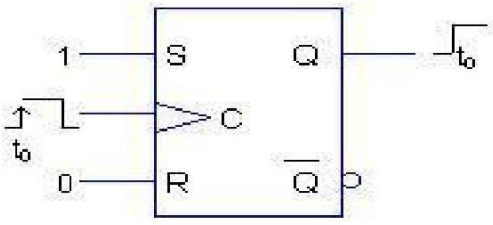
EDGE TRIGGERED FLIP FLOP

- Edge triggered flip-flop changes only when the clock C changes
- **The three basic types are introduced here: S-R, J-K and D.**

Edge-triggered S-R flip-flop

The basic operation is illustrated below, along with the truth table for this type of flip-flop. The operation and truth table for a negative edge-triggered flip-flop are the same as those for a positive except that the falling edge of the clock pulse is the triggering edge.

Inputs			Outputs		
S	R	C	Q	Q'	Comments
0	0	↑	Q	Q'	No change
0	1	↑	0	1	RESET
1	0	↑	1	0	SET
1	1	↑	?	?	Invalid



Edge-triggered J-K flip-flop

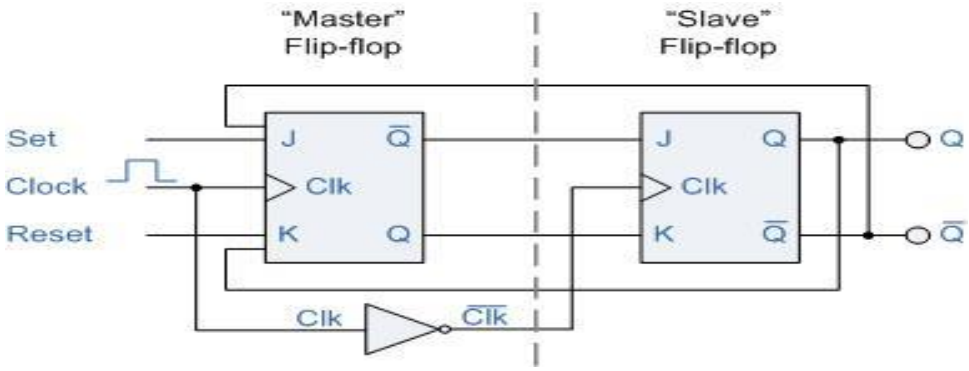
The J-K flip-flop works very similar to S-R flip-flop. The only difference is that this flip-flop has NO invalid state. The outputs toggle (change to the opposite state) when both J and K inputs are HIGH

Edge-triggered D flip-flop

The operations of a D flip-flop is much simpler. It has only one input addition to the clock. It is very useful when a single data bit (0 or 1) is to be stored. If there is a HIGH on the D input when a clock pulse is applied, the flip-flop SETs and stores a 1. If there is a LOW on the D input when a clock pulse is applied, the flip-flop RESETs and stores a 0. The truth table below summarize the operations of the positive edge-triggered D flip-flop. As before, the negative edge-triggered flip-flop works the same except that the falling edge of the clock pulse is the triggering edge.

MASTER-SLAVE FLIP FLOP

- Is designed using two separate flip flops. Out of these, one acts as the master and the other as a slave. The figure of a master-slave J-K flip flop is shown below.



From the above figure you can see that both the J-K flip flops are presented in a series connection. The output of the master J-K flip flop is fed to the input of the slave J-K flip flop. The output of the slave J-K flip flop is given as a feedback to the input of the master J-K flip flop. The clock pulse [Clk] is given to the master J-K flip flop and it is sent through a NOT Gate and thus inverted before passing it to the slave J-K flip flop.

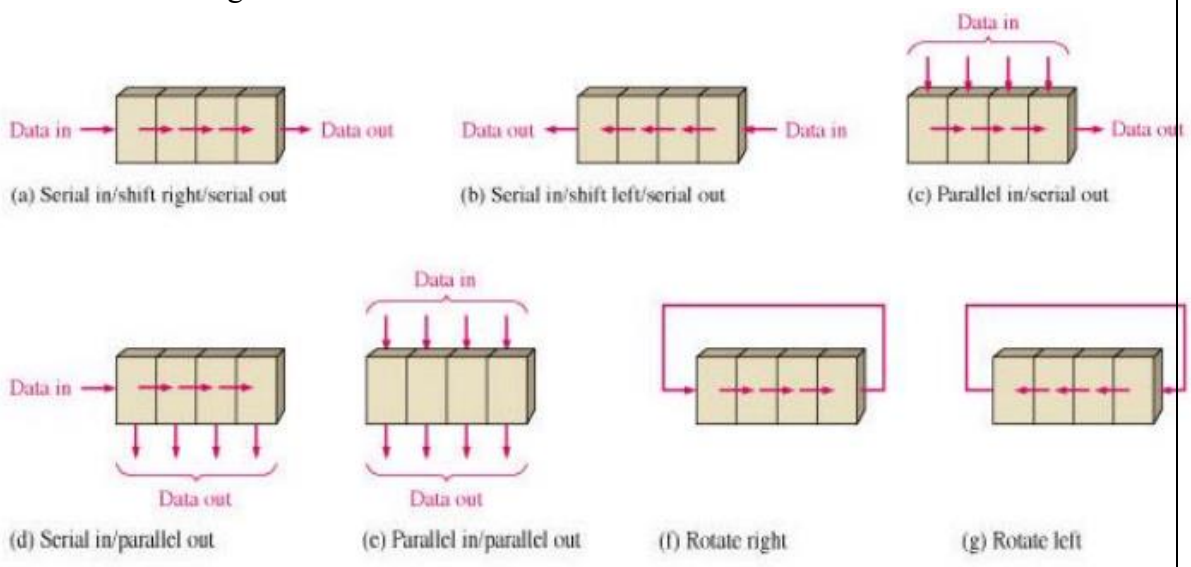
When Clk=1, the master J-K flip flop gets disabled. The Clk input of the master input will be the opposite of the slave input. So the master flip flop output will be recognized by the slave flip flop only when the Clk value becomes 0. Thus, when the clock pulse males a transition from 1 to 0, the locked outputs of the master flip flop are fed through to the inputs of the slave flip-flop making this flip flop edge or pulse-triggered.

REGISTERS

Registers are groups of flip-flops, where each flip-flop is capable of storing one bit of information. An n-bit register is a group of n flip-flops. The basic function of a register is to hold information in a digital system and make it available to the logic elements for the computing process. Registers consist of a finite number of flip-flops. Since each flip-flop is capable of storing either a "0" or a "1", there is a finite number of 0-1 combinations that can be stored into a register. Each of those combinations is known as state or content of the register. With flip-flops we can store data bitwise but usually data does not appear as single bits. Instead it is common to store data words of n bit with typical word lengths of 4, 8, 16, 32 or 64 bit. Thus, several flip-flops are combined to form a register to store whole data words. Registers are synchronous circuits thus all flip-flops are controlled by a common clock line. As registers are often used to collect serial data they are also called accumulators.

Shift Register

The Shift Register is another type of sequential logic circuit that can be used for the storage or the transfer of data in the form of binary numbers. This sequential device loads the data present on its inputs and then moves or “shifts” it to its output once every clock cycle, hence the name “shift register”.

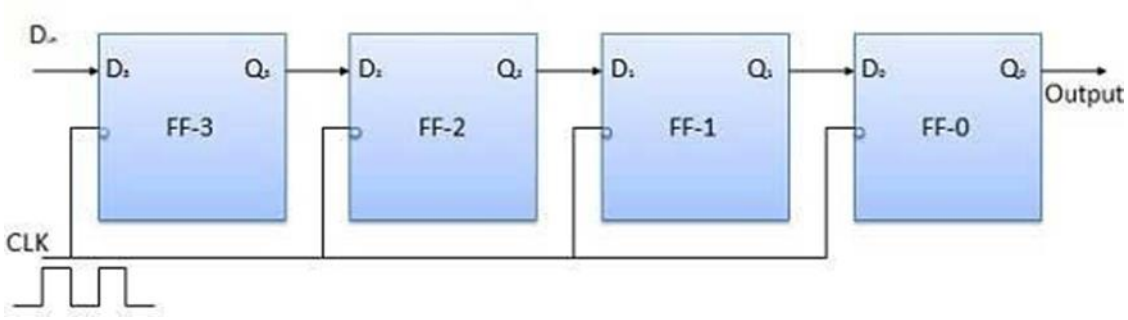


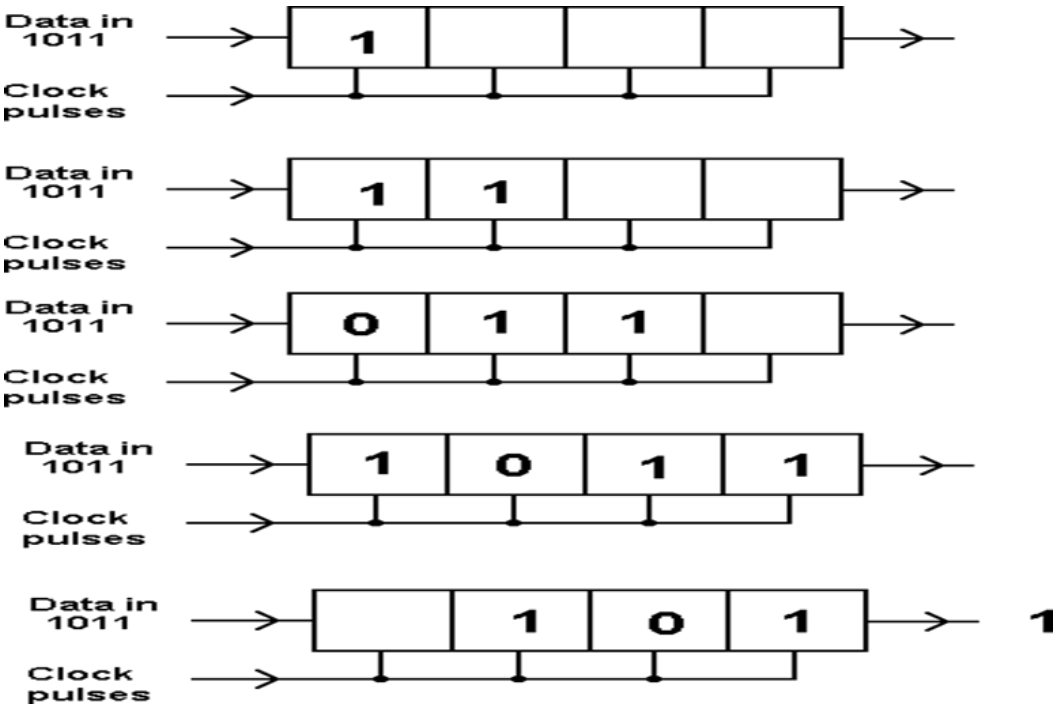
Shift registers operate in one of four different modes with the basic movement of data through a shift register being:

- Serial-in to Serial-out (SISO)
- Serial-in to Parallel-out (SIPO)
- Parallel-in to Serial-out (PISO)
- Parallel-in to parallel-out (PIPO)

Serial-in to Serial-out (SISO)

- The data is shifted serially "IN" and "OUT" of the register, one bit at a time in either a left or right direction under clock control.
- It has only three connections, the serial input, which determines what enters the left hand flip- flop, the serial output, which is taken from the output of the right hand flip-flop and the sequencing clock signal. The logic circuit diagram below shows a generalized serial-in serial-out shift register.

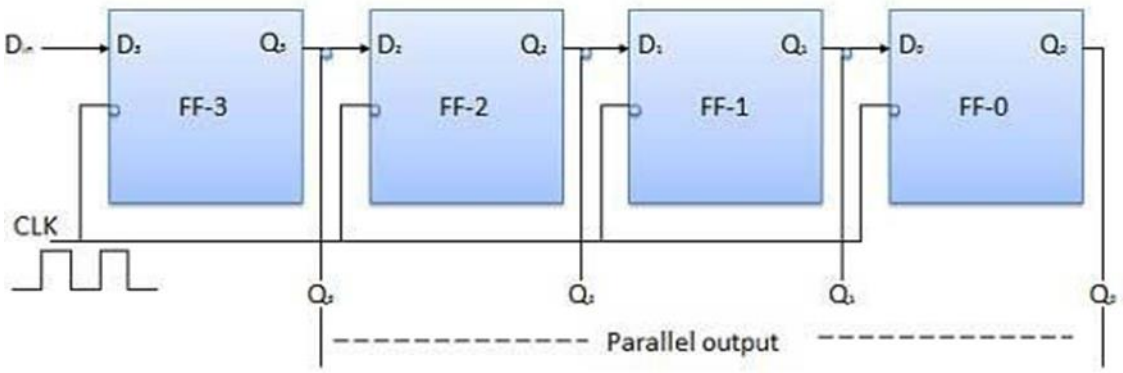




The above figure illustrates entry of the four bits 1010 into the register. After imputing of data's, Four bits (1010) being serially shifted out of the register and replaced by all zeros.

Serial-in to Parallel-out (SIPO)

- The register is loaded with serial data, one bit at a time, with the stored data being available at the output in parallel form.
- Clock cycles are required to load a four bit word.



4.4 Parallel-in to Serial-out (PISO)

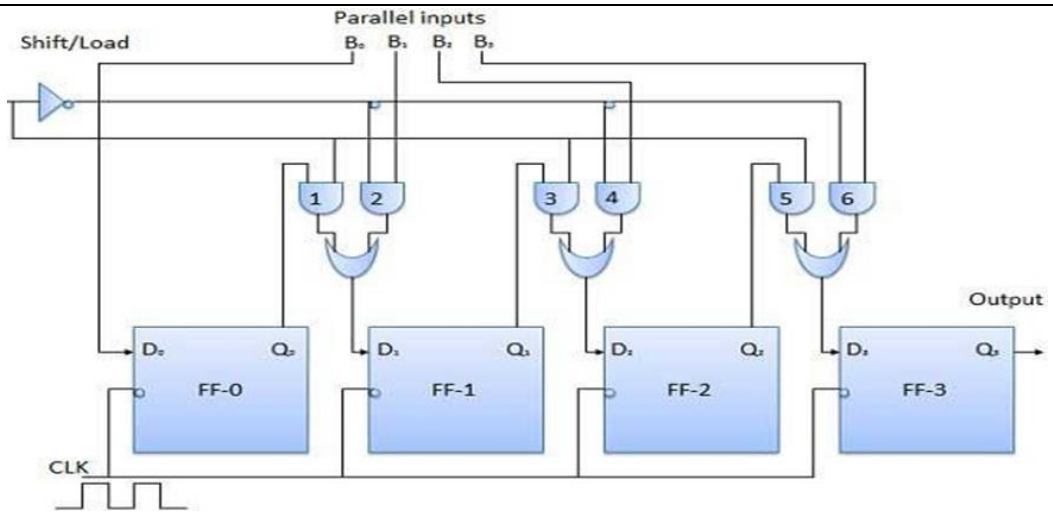
- The parallel data is loaded into the register simultaneously and is shifted out of the register serially one bit at a time under clock control.
- The circuit shown below is a four bit parallel input serial output register.
- Output of previous Flip Flop is connected to the input of the next one via 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, and B3bits 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.

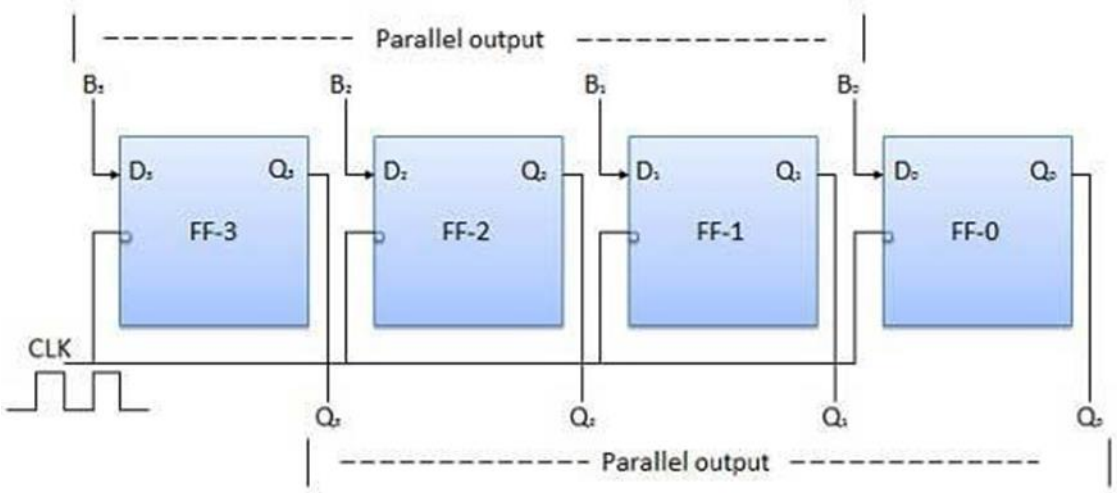
SHIFT MODE

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 take place



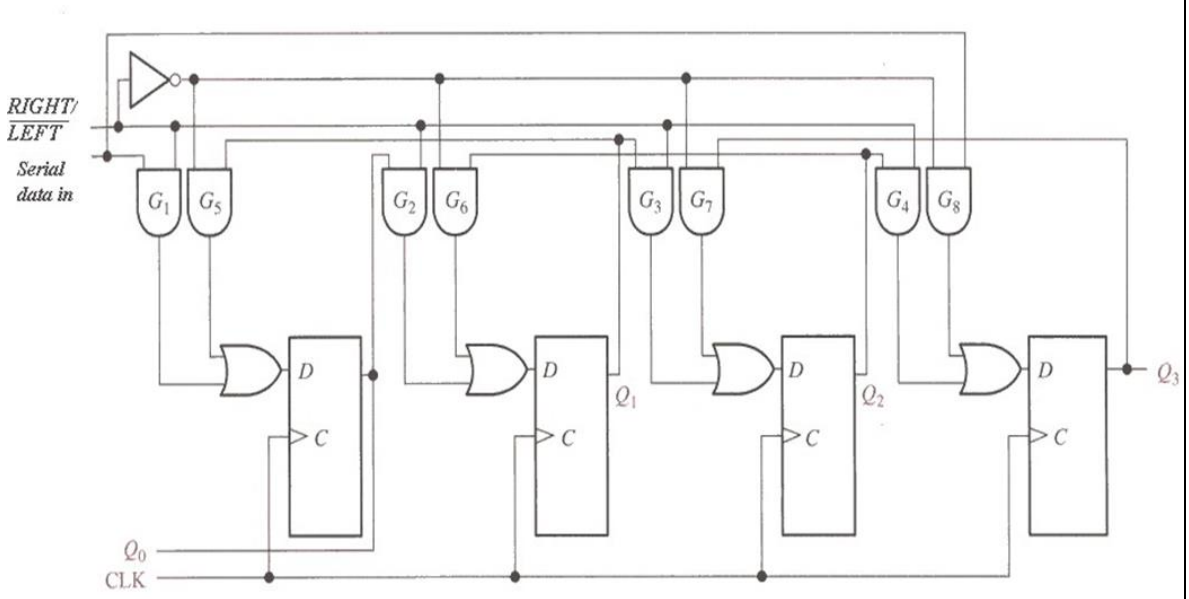
Parallel-in to parallel-out (PIPO)

- The parallel data is loaded simultaneously into the register, and transferred together to their respective outputs by the same clock pulse.
- In this mode, the 4 bit binary input B0, B1, B2, B3 is applied to the data inputs D0, D1, D2, and 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.



Bidirectional Shift Registers

A bidirectional, or reversible, shift register is one in which the data can be shift either left or right. A four- bit bidirectional shift register using D flip-flops is shown below.



Here a set of NAND gates are configured as OR gates to select data inputs from the right or left adjacent bistables, as selected by the LEFT/RIGHT control line.

With Right/Left = 1: Shift right operation

- Then the AND gates 1, 3, 5 and 7 are enable whereas the remaining AND gates 2, 4, 6 and 8 will be disabled.
- The data is shifted to right bit by bit on the application of clock pulses. Thus we get the serial right shift operation.

With Right/Left = 0: Shift left operation

- When the mode control M is connected to 0 then the AND gates 2, 4, 6 and 8 are enabled while 1, 3, 5 and 7 are disabled.

The data is shifted left bit by bit on the application of clock pulses. Thus we get the serial right shift operation.