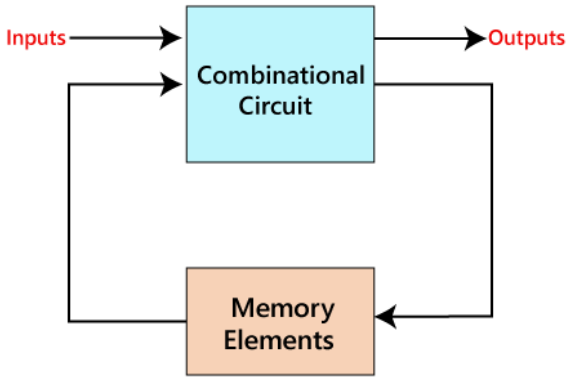
MODULE IV: Sequential Circuit & Design

Important topics

1. Comparison between synchronous and asynchronous sequential circuit.
2. Comparison between Sequential and combinational circuit.
3. Compare NOR latch and NAND latch
4. SR, JK, Master Slave J K flip flops with diagram and characteristic table.
5. Design of 3,4 bit counters with diagram
6. Diagrams and brief explanation of shift registers.

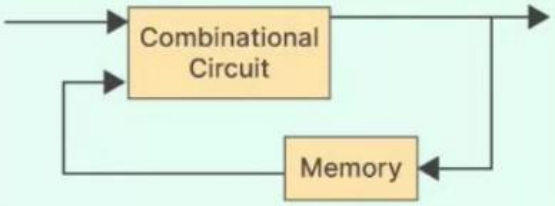
Sequential circuit

- A sequential circuit refers to a special type of circuit in which the outputs depend on a combination of both the present inputs as well as the previous outputs.
- That means sequential circuits include memory elements that are capable of storing binary information. That binary information defines the state of the sequential circuit at that time.
- There are two types of sequential circuits synchronous and asynchronous.
- Example Latch, Flipflop etc



Block Diagram

Comparison between Sequential and combinational circuit

Combinational Circuit	Sequential Circuit
Output only depends on the present input	Output depends on present input & past output
Memory element is absent	Memory element is present
No clock signal is applied	Clock signal is required
	
Example - Half Adder, Full Adder, Multiplexer	Example - Flipflop, Counters, Registers

Synchronous and Asynchronous sequential circuit

Asynchronous sequential circuit:

- These circuits **do not use a clock signal** but uses the pulses of the inputs.
- These circuits are **faster** than synchronous sequential circuits because there is clock pulse and change their state immediately when there is a change in the input signal.
- We use asynchronous sequential circuits when speed of operation is important and **independent** of internal clock pulse.
- Circuits are used in low power and high speed operations such as simple microprocessors, digital signal processing units.

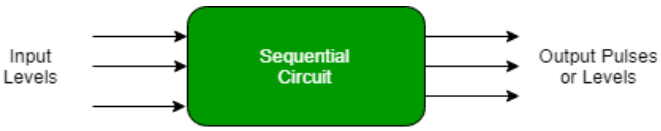
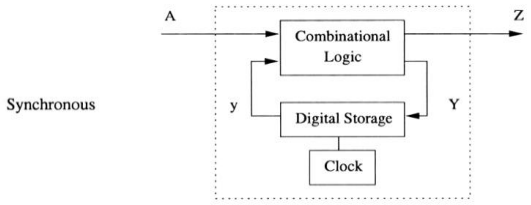


Figure: Asynchronous Sequential Circuit

Synchronous sequential circuit:

- These circuits **use clock signal**
- The output pulse is the same duration as the clock pulse for the clocked sequential circuits.
- Since they wait for the next clock pulse to arrive to perform the next operation, so these circuits are bit **slower** compared to asynchronous.
- Synchronous circuits are used in counters, shift registers, memory units.



Sr. No.	Synchronous sequential circuits	Asynchronous sequential circuits
1.	In synchronous circuits, memory elements are clocked flip-flops.	In asynchronous circuits, memory elements are either unclocked flip-flops or time delay elements.
2.	In synchronous circuits, the change in input signals can affect memory element upon activation of clock signal.	In asynchronous circuits change in input signals can affect memory element at any instant of time.
3.	The maximum operating speed of clock depends on time delays involved.	Because of absence of clock, asynchronous circuits can operate faster than synchronous circuits.
4.	Easier to design.	More difficult to design.

Latches

- Latches are digital circuits that store a single bit of information and hold its value until it is updated by new input signals.
- It is is a special type of logical circuit and have two stable states low and high (1 or 0)
- They are used in digital systems as temporary storage elements to store binary information.
- Latches can be implemented using various digital logic gates, such as AND, OR, NOT, NAND, and NOR gates.

Types of Latches

1. S-R (Set-Reset) Latches:

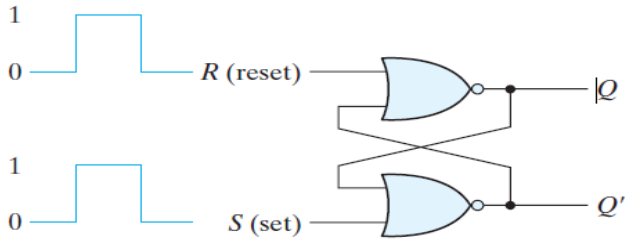
S-R latches are the simplest form of latches and are implemented using two inputs: S (Set) and R (Reset). The S input sets the output to 1, while the R input resets the output to 0. When both S and R are at 1, the latch is said to be in an "undefined" state.

- The *SR* latch is a circuit with two cross-coupled NOR gates or two cross-coupled NAND gates
- Two inputs labeled *S* for set and *R* for reset.
- The latch has two useful states.
- The outputs Q and Q' must be compliment to each other

- Output of Q of first inverter connected to the input of second inverter.
- Output of Q' of second inverter connected to the input of first inverter.
- When the power is switched on , the circuit switches to one of the stable states, and it is not possible to predict the state.

SR latch with NOR-gate

- When output $Q = 1$ and $Q' = 0$, the latch is said to be in the *set state* .
- When $Q = 0$ and $Q' = 1$, it is in the *reset state* .



(a) Logic diagram

S	R	Q	Q'	
1	0	1	0	
0	0	1	0	(after $S = 1, R = 0$)
0	1	0	1	
0	0	0	1	(after $S = 0, R = 1$)
1	1	0	0	(forbidden)

(b) Function table

FIGURE 5.3
SR latch with NOR gates

- Active HIGH SR Latch(NOR gate) Truth table

S	R	Q_N	Q_{N+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	-
1	1	1	-

- Case1: SET=0:RESET=0: This is the normal resting state of NOR latch and it has no effect on the output state, Q and Q' remains in the previous state

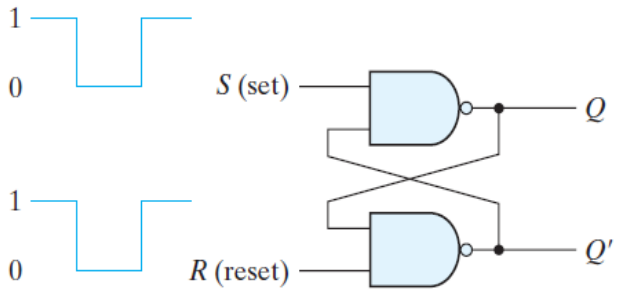
- Case2: SET=1,RESET=0 This will always set Q=1,where it will remain even after SET returns to 0

- Case 3: SET=0 RESET=1: This will always set Q=0,where it will remain even after RESET returns to 0

- Case4: SET=1 RESET=1 This condition tries to SET and RESET the latch at the same time, and it produce Q=Q'=0, so this Condition should not be used.

SR latch with NAND-gate

- When output $Q = 0$ and $Q' = 1$, the latch is said to be in the *set state* .
- When $Q = 1$ and Q '= 0, it is in the *reset state*
- When $Q = 1$ and Q '= 1, it is in not used latch is said to be in a forbidden state



(a) Logic diagram

S	R	Q	Q′	
1	0	0	1	
1	1	0	1	(after $S = 1, R = 0$)
0	1	1	0	
1	1	1	0	(after $S = 0, R = 1$)
0	0	1	1	(forbidden)

(b) Function table

FIGURE 5.4
SR latch with NAND gates

S	R	Q	Q_{+1}	STATE
0	0	0	X	INVALID
0	0	1	X	
0	1	0	1	SET
0	1	1	1	
1	0	0	0	RESET
1	0	1	0	
1	1	0	0	NO CHANGE
1	1	1	1	

Truth Table of SR Flip Flop

2. D (Data) Latches:

D latches are also known as transparent latches and are implemented using two inputs: D (Data) and a clock signal. The output of the latch follows the input at the D terminal as long as the clock signal is high. When the clock signal goes low, the output of the latch is stored and held until the next rising edge of the clock.

Flip Flop

- A flip-flop in electronics is a circuit with two stable states that can be used to store binary data. The stored data can be changed by applying varying inputs.
- A flip-flop is a digital memory circuit that stores one bit of data. They are the primary blocks of the most sequential circuits.

- Difference between flip-flop and latch is that the flip-flop is an edge-triggered type of memory circuit while the latch is a level-triggered type.
- It means that the output of a latch changes whenever the input changes.
- Flip-flops and latches are fundamental building blocks of digital electronics systems used in computers, communications, and many other types of systems. Both are used as data storage elements

Types of Flip–Flops

- S-R flip-flop
- J-K flip-flop
- D flip-flop
- T flip-flop

S-R Flip Flop(SET-RESET Flip Flop)

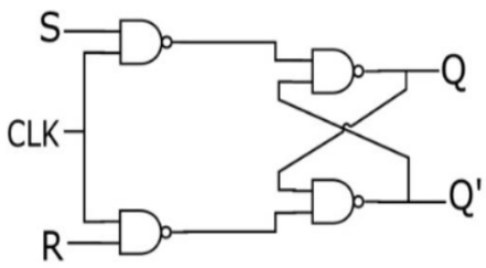
Flip flop is a synchronous sequential ckt. Since it contains a clock, **the circuit will be active only if the clock is enabled**

It is a Flip Flop with two inputs, one is S and other is R. S here stands for Set and R here stands for Reset. Set basically indicates set the flip flop which means output or Q=1 and reset indicates resetting the flip flop which means output Q=0

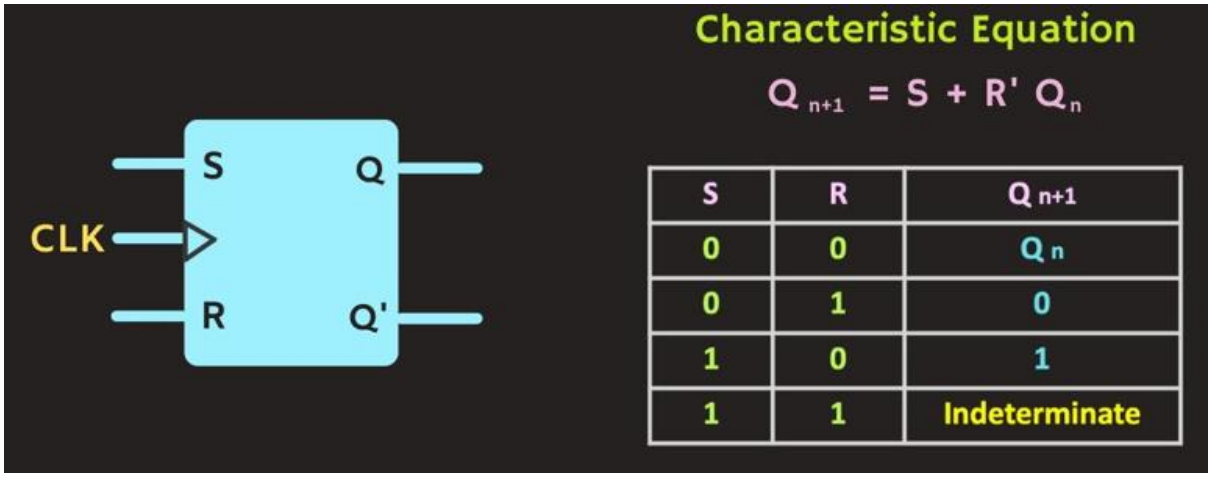
It has a set input (S) and a reset input (R).When in this circuit when S is set as active, the output Q would be high and the Q' will be low.

If R is set to active then the output Q is low and the Q' is high.

Once the outputs are established, the results of the circuit are maintained until S or R get changed, or the power is turned off.



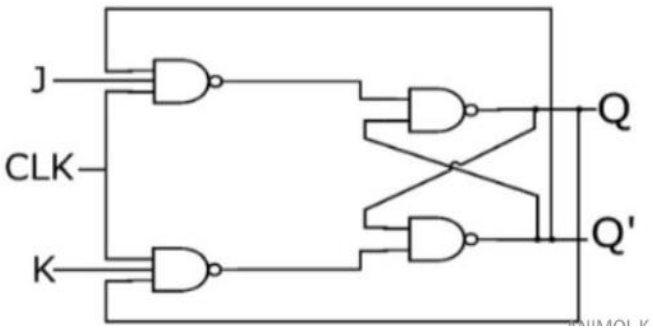
S	R	Q	State
0	0	0	No Change
0	1	0	Reset
1	0	1	Set
1	1	X	



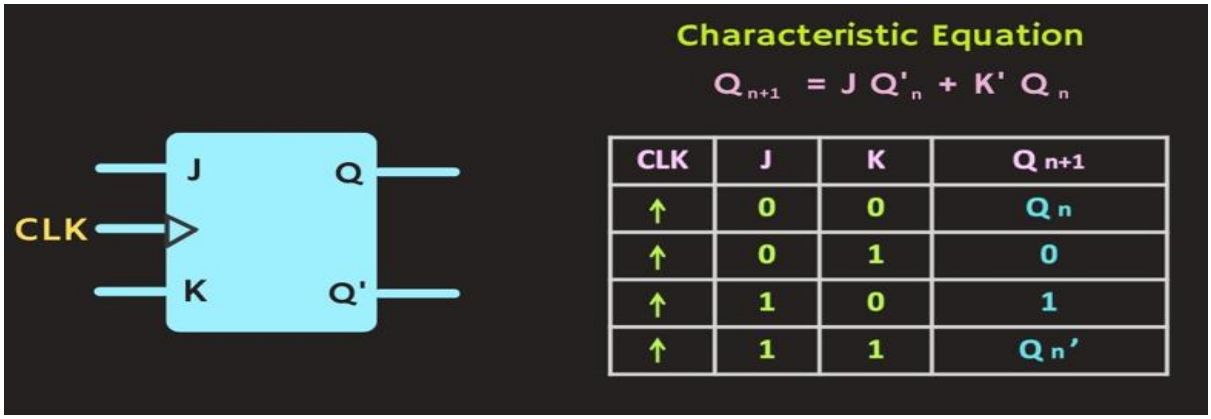
J-K Flip-flop- Jack Kilby flip flop

Because of **the invalid state corresponding to S=R=1 in the SR flip-flop, there is a need of another flip-flop.**

The JK flip-flop operates with only positive or negative clock transitions.
The operation of the JK flip-flop is similar to the SR flip-flop.When the input J and K are different then the output Q takes the value of J at the next clock edge.When J and K both are low then NO change occurs at the output.If both J and K are high, then at the clock edge, the output will toggle from one state to the other.

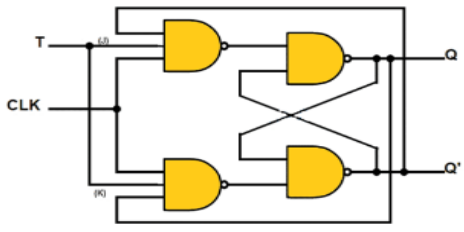


J	K	Q	State
0	0	0	No Change
0	1	0	Reset
1	0	1	Set
1	1	Toggles	Toggle



T Flip-flop- Toggle flip flop

- A T flip-flop (Toggle Flip-flop) is a simplified version of JK flip-flop.
- The T flop is obtained by connecting the J and K inputs together.
- The flip-flop has one input terminal and clock input.
- These flip-flops are said to be T flip-flops because of their ability to toggle the input state.
- Toggle flip-flops are mostly used in counters



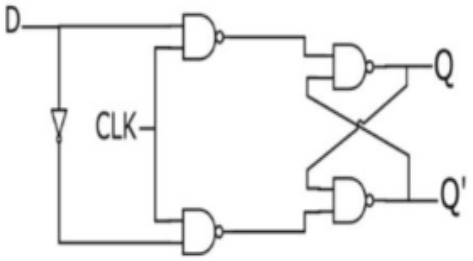
CLK	T	Q n+1
↑	0	Q n
↑	1	Q n'

D FLIP FLOP-Data flip flop

- In a D flip-flop, the output can only be changed at positive or negative clock transitions, and when the inputs changed at other times, the output will remain unaffected.

- The D flip-flops are generally used for shift-registers and counters.
- The change in output state of D flip-flop depends upon the active transition of clock.
- The output (Q) is same as input and changes only at active transition of clock

D	Q
0	0
1	1

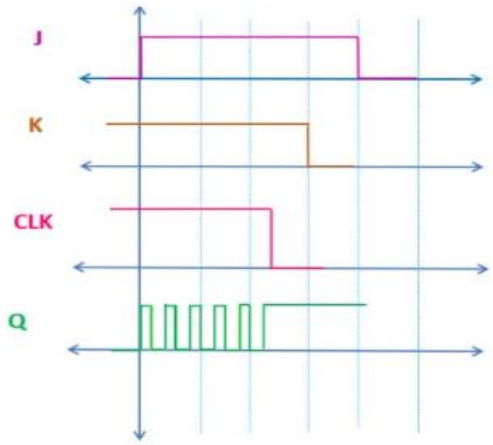


Inputs	Present Output	Next Output
D	Q_n	Q_{n+1}
0	0	0
0	1	0
1	0	1
1	1	1

Master slave J K- Flip Flop

Race Around Condition in JK Flip-Flop

In "JK Flip Flop", when both the inputs and CLK set to 1 for a long time, then Q output toggle until the CLK is 1. Thus, the uncertain or unreliable output produces. This problem is referred to as a race-around condition in JK flip-flop and avoided by ensuring that the CLK set to 1 only for a very short time. To overcome race around condition MSFF has been designed.

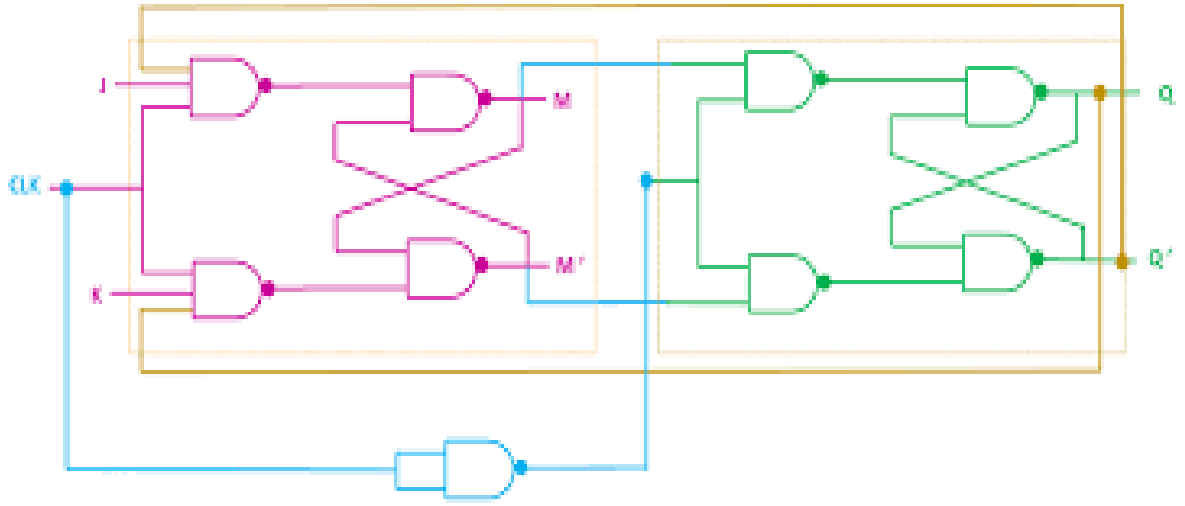


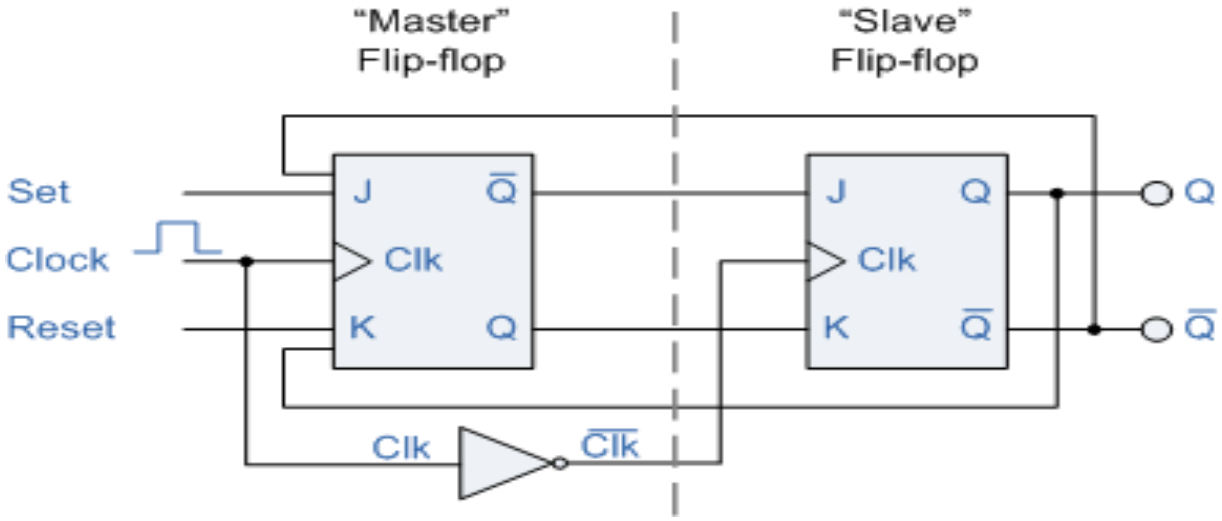
When J = 1 and K = 1, for this input combination, the master flip flop toggles on the positive transition of the clock pulse and the slave flip flop toggles on the negative transition of the clock pulse. Hence, the problem of the race around condition of the JK flip flop is solved.

- Master-slave flip flop is constructed by combining two JK flip flops.

- Master Slave JK Flip Flop has two JK Flip Flops connected together in series combination along with an inverter which gives inverted clock pulse to slave flip flop.

- In this combination of two JK flip flop, one acts as a master flip flop and the other acts as a slave flip flop. In this master-slave flip flop, the outputs of the master JK flip flop are connected to the inputs of the slave JK flip flop. The outputs of the slave flip flop are fed back to the inputs of the master JK flip flop.
- In "master-slave flip flop", apart from these two flip flops, an inverter or NOT gate is also used. For passing the inverted clock pulse to the "slave" flip flop, the inverter is connected to the clock's pulse.





- Master-slave flip flop is constructed by combining two JK flip flops.
- These flip flops are connected in a series configuration. In these two flip flops, the 1st flip flop work as "master", called the master flip flop, and the 2nd work as a "slave", called slave flip flop.
- The master-slave flip flop is designed in such a way that the output of the "master" flip flop is passed to both the inputs of the "slave" flip flop. The output of the "slave" flip flop is passed to inputs of the master flip flop.
- In "master-slave flip flop", an inverter or NOT gate is also used. For passing the inverted clock pulse to the "slave" flip flop, the inverter is connected to the clock's pulse.

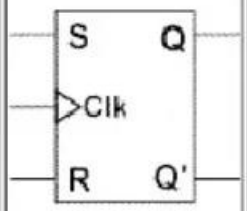
Working

- When the clock pulse goes to 1, the slave is isolated; J and K inputs may affect the state of the system. The slave flip-flop is isolated until the CP goes to 0. When the CP goes back to 0, information is passed from the master flip-flop to the slave and output is obtained.

- Firstly the master flip flop is positive level triggered and the slave flip flop is negative level triggered, so the master responds before the slave.
- If J=0 and K=1, the high Q' output of the master goes to the K input of the slave and the clock forces the slave to reset, thus the slave copies the master.
- If J=1 and K=0, the high Q output of the master goes to the J input of the slave and the Negative transition of the clock sets the slave, copying the master.
- If J=1 and K=1, it toggles on the positive transition of the clock and thus the slave toggles on the negative transition of the clock.
- If J=0 and K=0, the flip flop is disabled and Q remains unchanged.

SUMMARY

SR

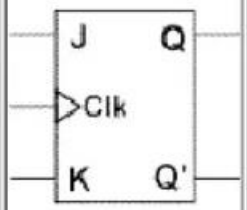


S	R	$Q_{(next)}$
0	0	Q
0	1	0
1	0	1
1	1	?

$Q_{(next)} = S + R'Q$

$SR = 0$

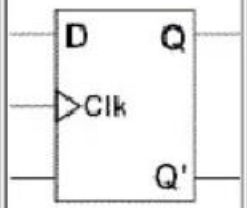
JK



J	K	$Q_{(next)}$
0	0	Q
0	1	0
1	0	1
1	1	Q'

$Q_{(next)} = JQ' + K'Q$

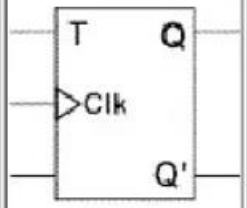
D



D	$Q_{(next)}$
0	0
1	1

$Q_{(next)} = D$

T



T	$Q_{(next)}$
0	Q
1	Q'

$Q_{(next)} = TQ' + T'Q$

Triggering in Flip Flop

- Clock Signal and Triggering
- The clock signal refers to a periodic signal where the ON and OFF times do not need to be the same.
- Thus, whenever the ON and OFF times of the clock signal happen to be the same, we use a square wave to represent the clock signal.

Types of Triggering

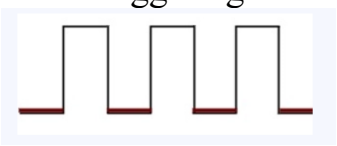
1. Level triggering
2. Edge Triggering

1. Level Triggering

There are two levels present in the clock signal – the logic Low and the logic High. The circuit is only activated in the case of a level triggering whenever the clock pulse happens to be at any particular level. Thus, there are the following types of level triggering in a sequential circuit:

- **Positive Level Triggering**
  - If the sequential circuit is operated with the clock signal when it is in logic high, then that type of triggering is known as a positive level triggering.

- **Negative Level Triggering**
  - If the sequential circuit is operated with the clock signal when it is in logic low, then that type of triggering is known as a negative level triggering.

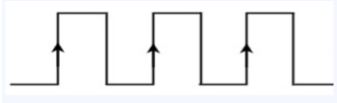


Edge Triggering

This transition either occurs from Logic High to Logic Low or from Logic Low to Logic High.

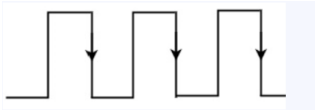
- **Positive edge triggering**

If the sequential circuit is operated with the clock signal transit from low to high. then that type of triggering is known as a positive edge triggering



- **Negative edge triggering**

If the sequential circuit is operated with the clock signal transit from low to high. then that type of triggering is known as a positive edge triggering



COUNTERS

A Counter is a device which stores the number of times a particular event or process has occurred, often in relationship to a clock signal. Counters are used in digital electronics for counting purpose, they can count specific event happening in the circuit. For example, in UP counter a counter increases count for every rising edge of clock. The main properties of a counter are timing , sequencing , and counting.
Counter works in two modes
Up counter
Down counter

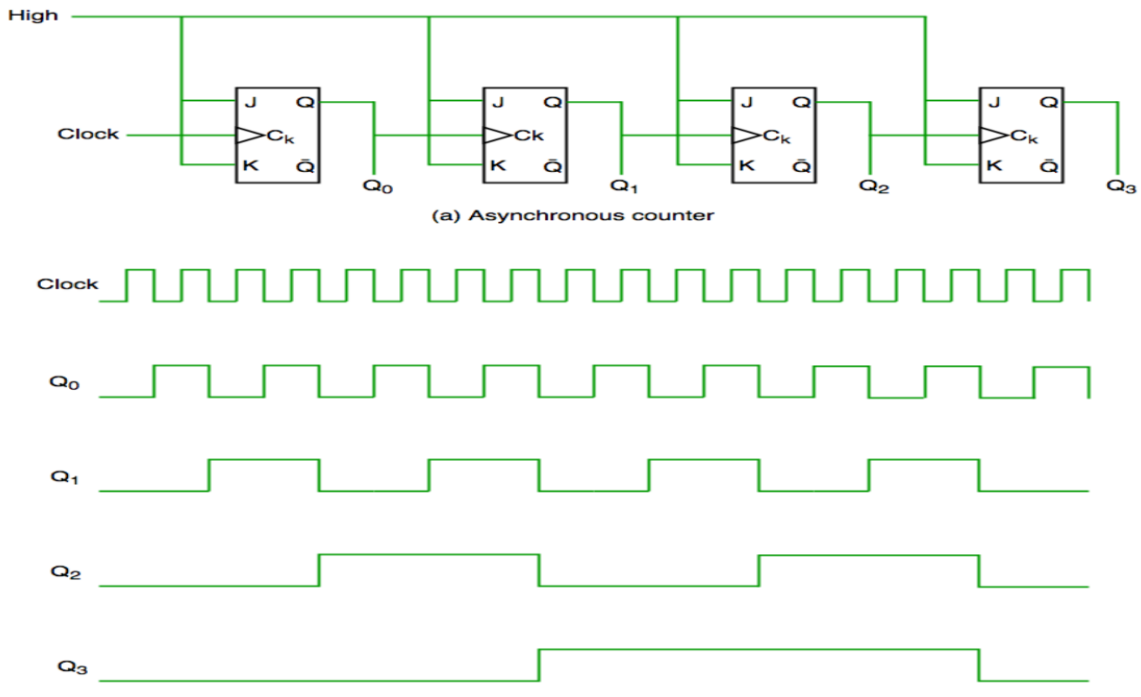
Asynchronous Counter --Ripple counter

In asynchronous counter don't use universal clock, only first flip flop is driven by main clock and the clock input of rest of the following flip flop is driven by output of previous flip flops.

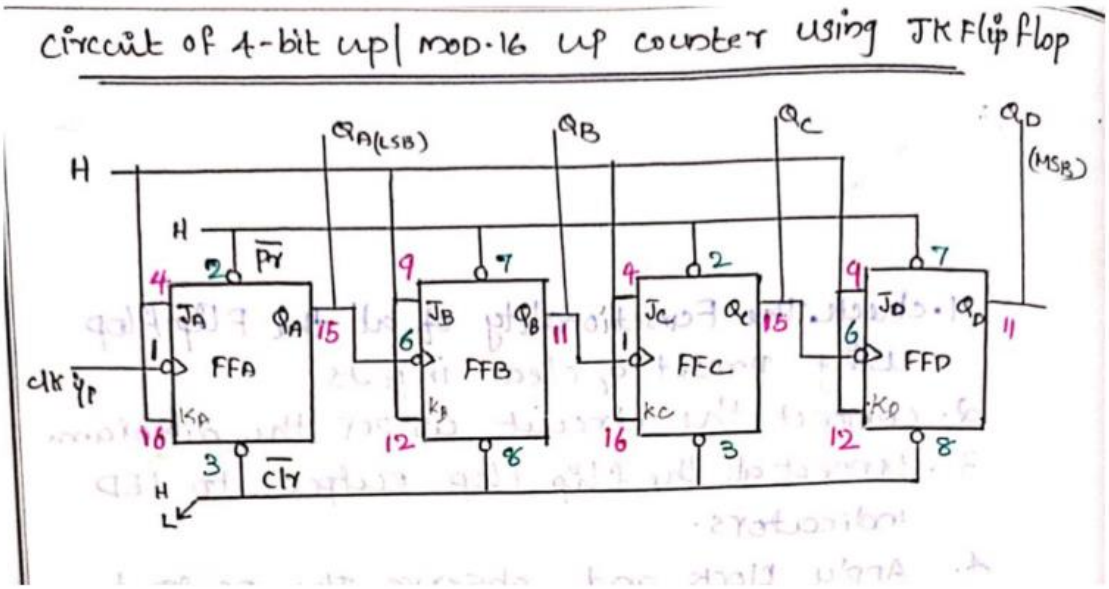
Q0 is changing as soon as the rising edge of clock pulse is encountered, Q1 is changing when rising edge of Q0 is encountered(because Q0 is like clock pulse for second flip flop) and so on.

In this way ripples are generated through Q0,Q1,Q2,Q3 hence it is also called RIPPLE counter.

A ripple counter is a cascaded arrangement of flip flops where the output of one flip flop drives the clock input of the following flip flop

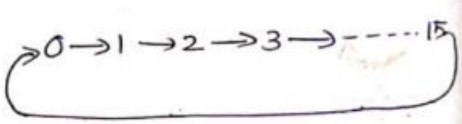


Asynchronous 4 bit UP Counter

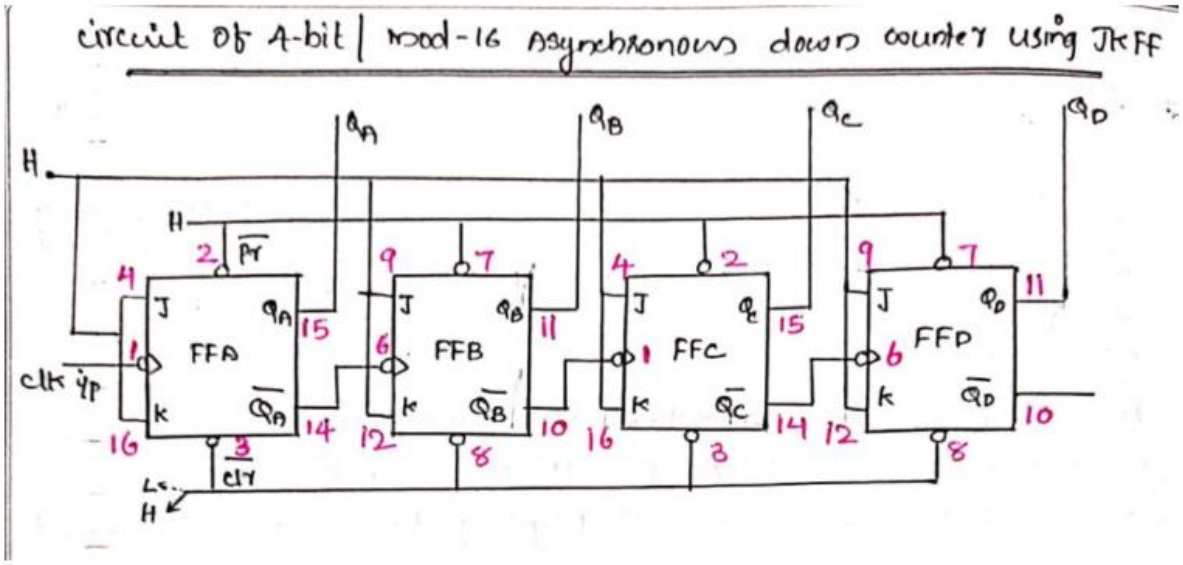


output				input		
Q_D	Q_C	Q_B	Q_A	clk	$\overline{Pr}$	$\overline{clr}$
0	0	0	0	0	H	0
0	0	0	1	1	H	H
0	0	1	0	2	H	H
0	0	1	1	3	H	H
0	1	0	0	4	H	H
0	1	0	1	5	H	H
0	1	1	0	6	H	H
0	1	1	1	7	H	H
1	0	0	0	8	H	H
1	0	0	1	9	H	H
1	0	1	0	10	H	H
1	0	1	1	11	H	H
1	1	0	0	12	H	H
1	1	0	1	13	H	H
1	1	1	0	14	H	H
1	1	1	1	15	H	H

State diagram

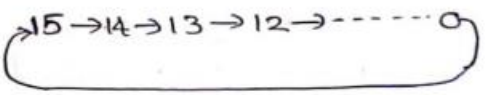


ASYNCHRONOUS 4 BIT BINARY DOWN COUNTER



$\overline{PR}$	$\overline{CLR}$	clk	Q_D	Q_C	Q_B	Q_A
0	H	x (0)	1	1	1	1
H	H	↓ (1)	1	1	1	0
H	H	↓ (2)	1	1	0	1
H	H	↓ (3)	1	1	0	0
H	H	↓ (4)	1	0	1	1
H	H	↓ (5)	1	0	1	0
H	H	↓ (6)	1	0	0	1
H	H	↓ (7)	1	0	0	0
H	H	↓ (8)	0	1	1	1
H	H	↓ (9)	0	1	1	0
H	H	↓ (10)	0	1	0	1
H	H	↓ (11)	0	1	0	0
H	H	↓ (12)	0	0	1	1
H	H	↓ (13)	0	0	1	0
H	H	↓ (14)	0	0	0	1
H	H	↓ (15)	0	0	0	0

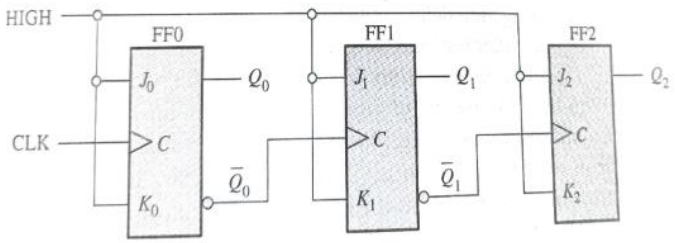
State diagram



3 bit Asynchronous Binary Counter

- 3 bit asynchronous counter has 8 states due to its 3 flipflops

- Counter progresses through a binary count of 0 through 7 then recycles to the zero state

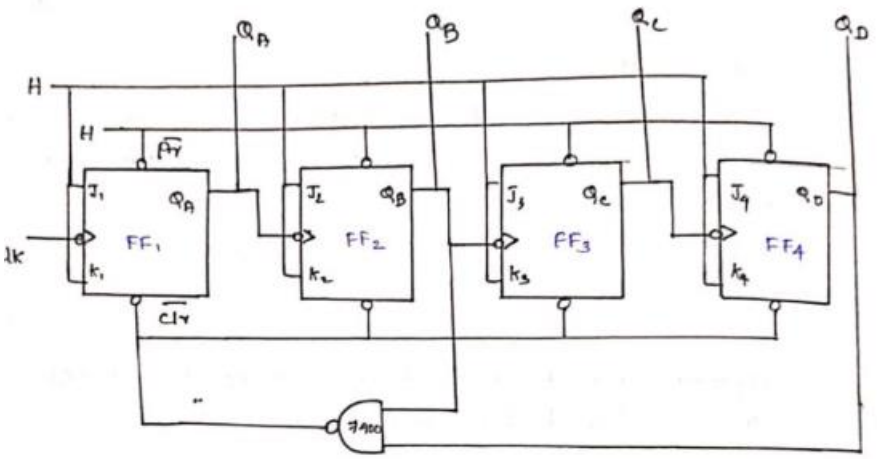


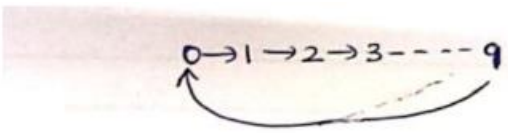
CLOCK PULSE	Q_2	Q_1	Q_0
Initially	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8 (recycles)	0	0	0

Asynchronous MOD 10 (BCD/ DECADE Counter)

- Counter with 10 states in their sequence are called Decade counter.
- It has a count sequence of 0 through 9 is a BCD decade counter because its 10 state sequence produces the BCD code.
- Decade counter requires 4 flipflops and NAND gate is connected to them

Decade counter mod-10

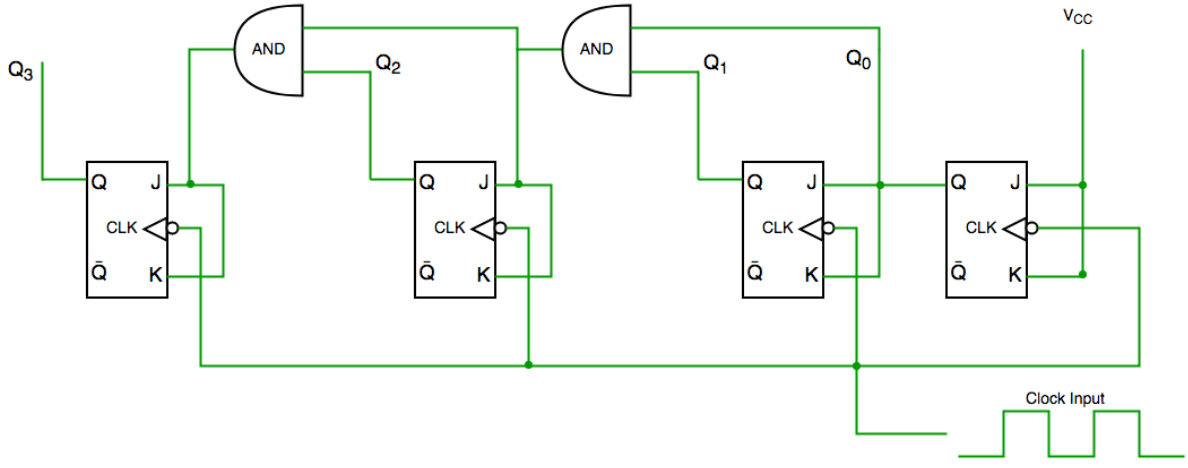


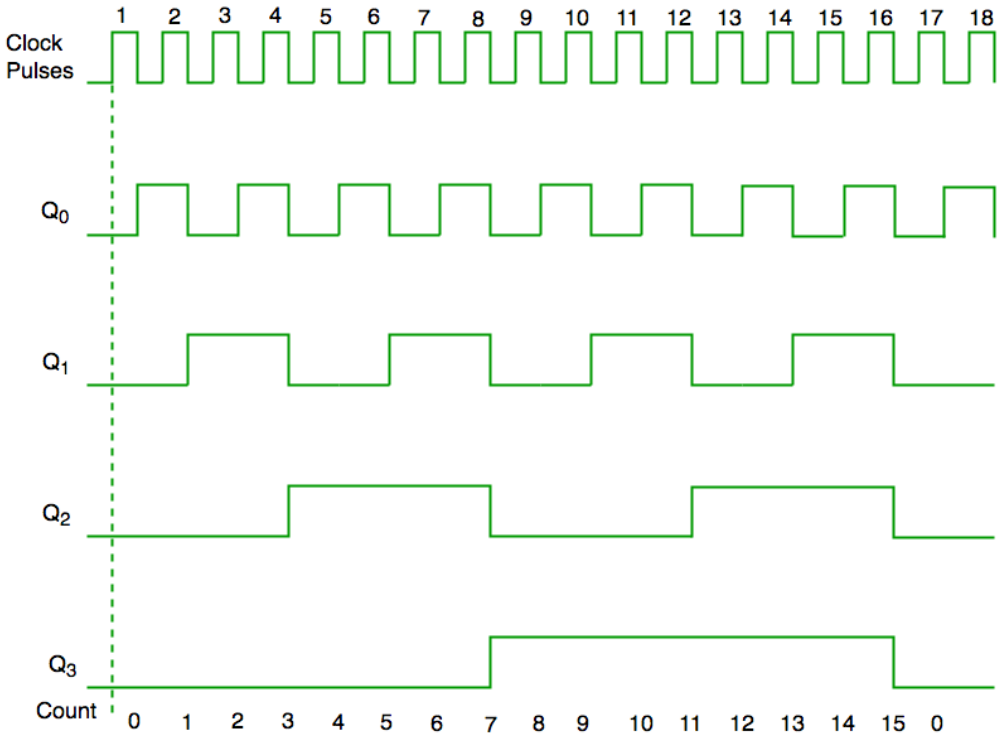


clk	Q_D	Q_C	Q_B	Q_A
1	0	0	0	0
2	0	0	0	1
3	0	0	1	0
4	0	0	1	1
5	0	1	0	0
6	0	1	0	1
7	0	1	1	0
8	0	1	1	1
9	1	0	0	0
10	1	0	0	1

Synchronous Counter

Synchronous counter has one global clock which drives each flip flop so output changes in parallel.The one advantage of synchronous counter over asynchronous counter is, it can operate on higher frequency than asynchronous counter as it does not have cumulative delay because of same clock is given to each flip flop.

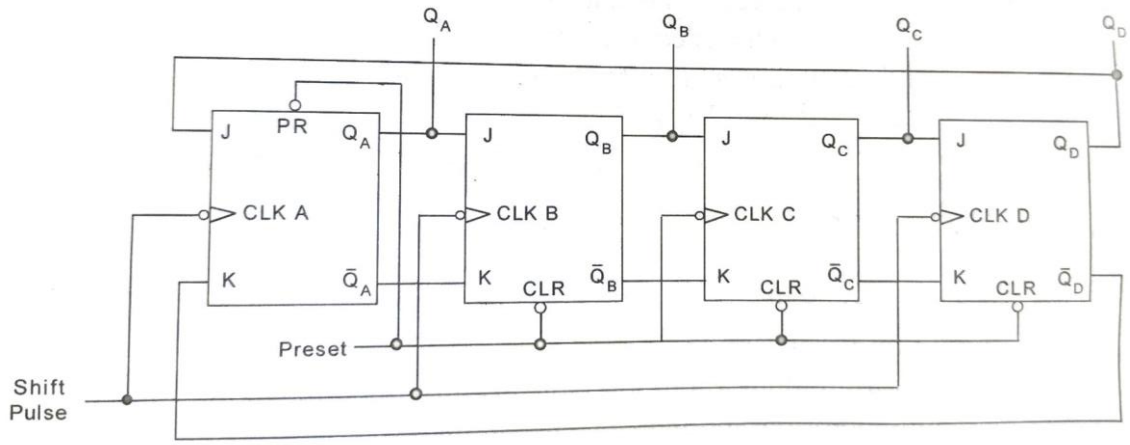




	Asynchronous Ripple Counters	Synchronous Counters
1.	The Delay time of all flip flops are added. Therefore, there is considerable propagation delay.	The clock pulses are applied to all flip flops simultaneously. Hence there is minimum propagation delay.
2.	Frequency of operation is lesser than in a synchronous counter.	Frequency of operation can be much higher than that in ripple counter.
3.	The maximum frequency depends on modulus.	The maximum frequency does not depend on modulus.
4.	Circuit is simple.	Circuit is complex.
5.	Minimum number of logic devices are needed.	The number of logic devices required is more than in ripple counter.
6.	Standard designs not available.	Standard designs available.
7.	Less costly than synchronous	Most costly than ripple counter. counter.

RING COUNTER

- A ring counter is a Shift Register (a cascade connection of flip-flops) with the output of the last flip flop connected to the input of the first.
- QD and QD' are connected to the J and K inputs of the flipflop A



Shift Register

A group of flip flops which is used to store multiple bits of data and the data is moved from one flip flop to another is known as Shift Register.

To form an n-bit shift register, we have to connect n number of flip flops.

The flip flops are connected in such a way that the first flip flop's output becomes the input of the other flip flop.

A Shift Register can shift the bits either to the left or to the right.

A Shift Register, which shifts the bit to the left, is known as "Shift left register", and it shifts the bit to the right, known as " Shift Right register"

The shift register is classified into the following types:

Serial In Serial Out

Serial In Parallel Out

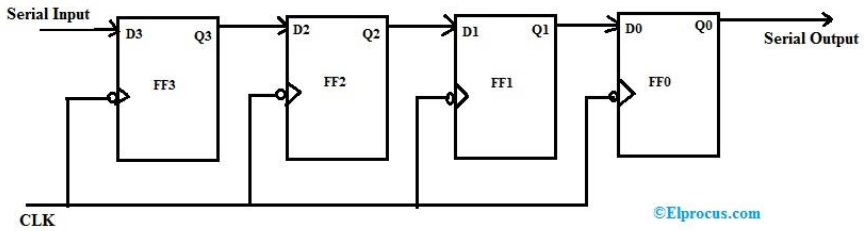
Parallel In Serial Out

Parallel In Parallel Out

1. Serial In Serial Out

In "Serial Input Serial Output", the data is shifted "IN" or "OUT" serially.

In SISO, a single bit is shifted at a time in either right or left direction under clock control.



Initially, all the flip-flops are set in "reset" condition i.e. Q3,Q2,Q1,Q0=0.
In this 4-bit shift register example like “1111”, the LSB bit is ‘1’ and the MSB bit is ‘1’.

First, the high signal (LSB bit) is used as an input to the first D3 flip flop, then D3=1. But primarily all the D FFs outputs will be 0. So, D2=D1=D0=0. When D3 input is high signal (1) then D3 will cause ‘Q3’ to be ‘1’. Therefore the overall o/p for 1st falling edge will become 1000.

Similarly, when the next data i/p bit in the above 4-bit like high signal (1) is given at flip flop D3, again this ‘D3’ will cause ‘Q3’ to be 1, however, ‘Q3’ is given as input to FF ‘D2’. So, this ‘D2’will cause ‘Q2’ to be 1 when all the remaining outputs will become 0.

As a result, we will obtain ‘11’ for a 2nd falling edge; so will obtain ‘11’ at the stored bit in the shift register, thus the overall o/p for the 2nd falling edge will get o/p as“1100”.

When the third input bit like high signal (1) is applied at the ‘D3’ FF then earlier ‘Q2’ o/p will cause the ‘D1’ i/p to be ‘0’. This will give the output Q3, Q2 & Q1 as ‘1’ whereas ‘Q0’ will be ‘0’. So the overall o/p for the 3rd falling edge will get o/p as “1110”.

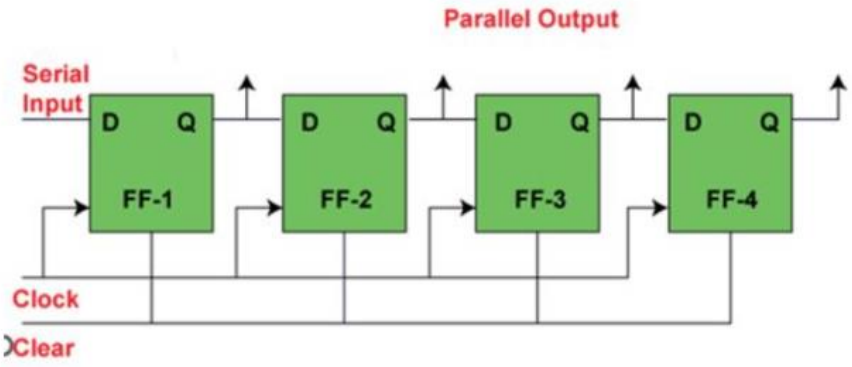
In addition, an MSB bit like high signal (1) is given as input, after that ‘1’ at ‘Q1’ will cause input ‘D0’ to be ‘1’, thus, this will make ‘Q0’ be ‘1’. Therefore, finally, SISO shift register store 1111 bit & shows in the o/p.

Clk	$D_n=Q_3$	$Q_3=D_2$	$Q_2=D_1$	$Q_1=D_0$	Q_0
Initially		0	0	0	0
(1) ↓	1 →	1	0	0	0
(2) ↓	1 →	1	1	0	0
(3) ↓	1 →	1	1	1	0
(4) ↓	1 →	1	1	1	1

→ Direction of data travel

Serial IN Parallel OUT

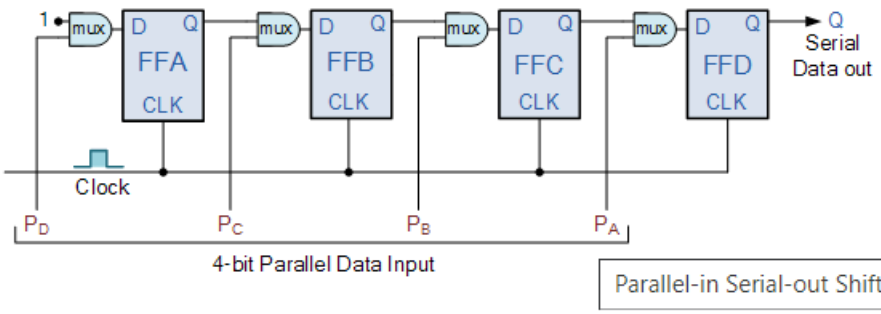
In the "Serial IN Parallel OUT" shift register, the data is passed serially to the flip flop, and outputs are fetched in a parallel way.
The data is passed bit by bit in the register, and the output remains disabled until the data is not passed to the data input.
When the data is passed to the register, the outputs are enabled, and the flip flops contain their return value.
Below is the block diagram of the 4-bit serial in the parallel-out shift register.
The circuit having four D flip-flops contains a clear and clock signal to reset these four flip flops.
In SIPO, the input of the second flip flop is the output of the first flip flop, and so On. The same clock signal is applied to each flip flop since the flip flops synchronize each other. The parallel outputs are used for communication.



Parallel-in to Serial-out (PISO)

In Parallel-in to Serial-out shift register data is loaded into the register in a parallel format in which all the data bits enter their inputs simultaneously, to the parallel input pins PA to PD of the register. The data is then read out sequentially in the normal shift-right mode from the register at Q representing the data present at PA to PD. This data is outputted one bit at a time on each clock cycle in a serial format.

4-bit Parallel-in to Serial-out



Parallel-in Serial-out Shift Register

Parallel IN Parallel OUT

In "Parallel IN Parallel OUT", the inputs and the outputs come in a parallel way in the register.The inputs D_A, D_B, D_C, D_D are directly passed to the data inputs. The clock pulse is required for loading all the bits. The loaded bits appear a t the output side.