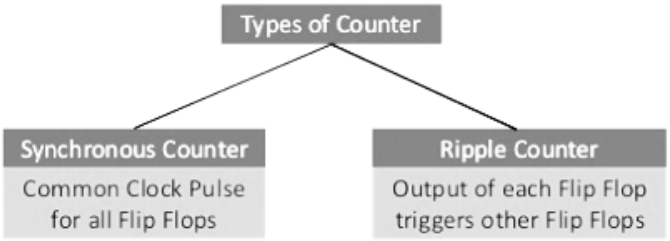
Unit 6: Counters, Registers and Memory

Counters:

Counters are used in digital electronics for counting purpose, they can count specific event happening in the circuit. Counter is a sequential circuit which is used for a counting pulses. Counter is the widest application of flip-flops.

There are two types of counters

1. Synchronous
2. Asynchronous.



Asynchronous Counters/ Ripple Counters

In asynchronous counter we don't use universal clock, only first flip flop is driven by main clock and the clock input of rest of the following counters is driven by output of previous flip flops. The required number of logic gates to design asynchronous counters is very less. So they are simple in design. Another name for Asynchronous counters is "Ripple counters". These counters are frequently used for measurement of Time, Measurement of Frequency, Measurement of Distance, Measurement of Speed, Waveform generation, Frequency Division, Digital Computers, Direct Counting etc….

Binary ripple Counter:

A binary ripple counter consist of a series of connection of complementing flip flips(T or JK type) with the output of each flip flop connected to the CP input of the next higher order flip flop. The flip flop holding the list significant bit receives the incoming count pulses.

A counter may be an up counter that counts upwards or can be a down counter that counts downwards or can do both i.e. count up as well as count downwards depending on the input control. The sequence of counting usually gets repeated after a limit. When counting up, for n-bit counter the count sequence goes from 000, 001, 010, … 110, 111, 000, 001, … etc. When counting down the count sequence goes in the opposite manner: 111, 110, … 010, 001, 000, 111, 110, … etc.

The diagram of 4-bit binary ripple counter is shown below. All J and K inputs are equal to 1. The small circle on the CP input indicates that the flip flop complements during a negative going transition or when the output to which it is connected goes from 1 to 0.

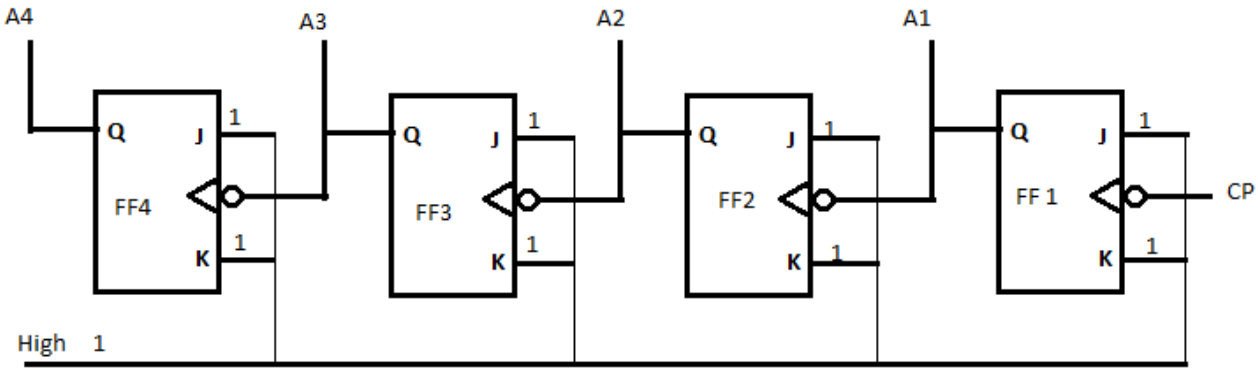


Fig 5. 1: 4-bit binary ripple counter using JK flip flop

To understand the operation of the binary counter refer to the count sequence given in the table.

Clock pulse	A4	A3	A2	A1	Decimal
Initially	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	2
3	0	0	1	1	3
4	0	1	0	0	4
5	0	1	0	1	5
6	0	1	1	0	6
7	0	1	1	1	7
8	1	0	0	0	8
9	1	0	0	1	9
10	1	0	1	0	10
11	1	0	1	1	11
12	1	1	0	0	12
13	1	1	0	1	13
14	1	1	1	0	14
15	1	1	1	1	15

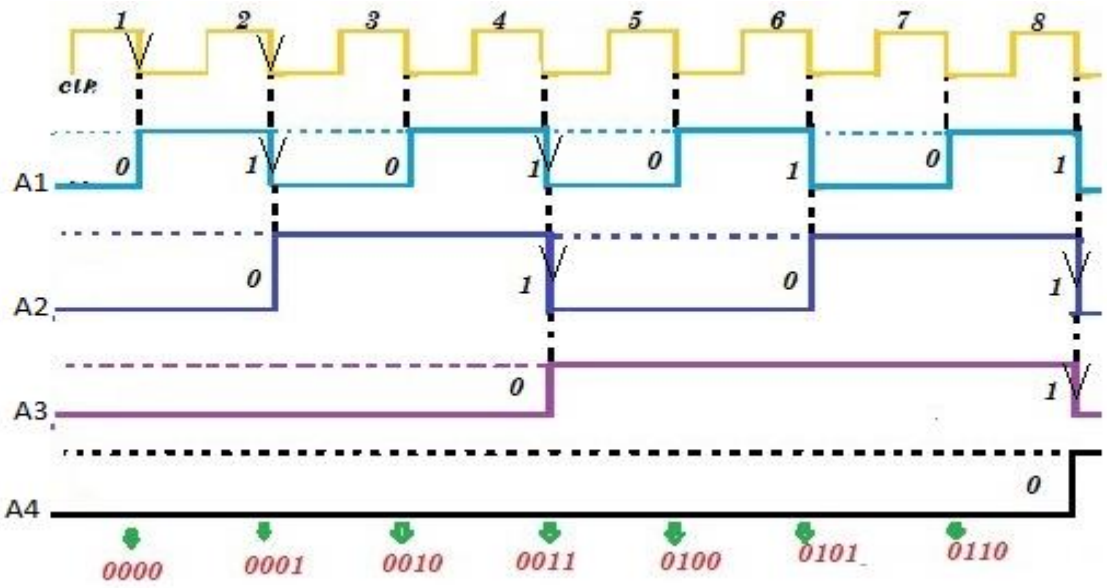
FF1 toggles in each and every clock pulse

FF2 toggles on every 2 clock pulse

FF3 toggles on every 4 clock pulse

FF4 toggles on every 8 clock pulse

Timing Diagram



- At first, assume all the output of flip flop is cleared to 0 0 0 0
- As clock pulse 1 is applied at the clock input of FF1 it toggles on negative edge and display 0 0 0 1
- Clock pulse 2 causes FF1 to toggle again, producing output Q of FF1 to 0 which causes FF2 to toggle and display 0 0 1 0
- FF1 is LSB, it must change state on every pulse, FF2 must toggle only half as often FF1 and again FF3 must toggle only half as FF2 and so on.

* A counter having n flip-flops can have a maximum of 2^n states. The number of states that a counter owns is known as its mod (modulo) number. Hence a 4-bit counter is **a mod-16 counter**.

Start

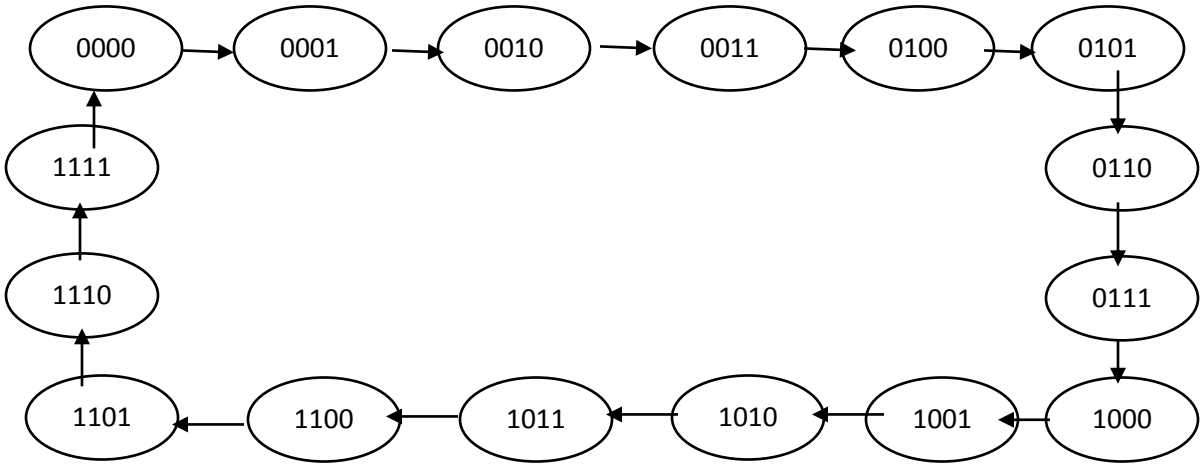


Fig: State diagram of 4 bit binary ripple counter

Synchronous Counters

If the "clock" pulses are applied to all the flip-flops in a counter simultaneously, then such a counter is called as synchronous counter. Synchronous counters are distinguished from ripple counters in that the clock pulse are applied to CP input of all flip flops. The common pulse triggers all the flip flops simultaneously, rather than one at a time. The decision whether a flip flop is to be complemented or not is determined from the values of the J and K inputs at the time of the pulse. If J=k=0, the flip flop remains unchanged. If J=K=1, the flip flop complements.

Depending on the way in which the counting progresses, the synchronous or asynchronous counters are classified as follows −

1. Up counters
2. Down counters
3. Up/Down counters

Synchronous Binary UP counter:

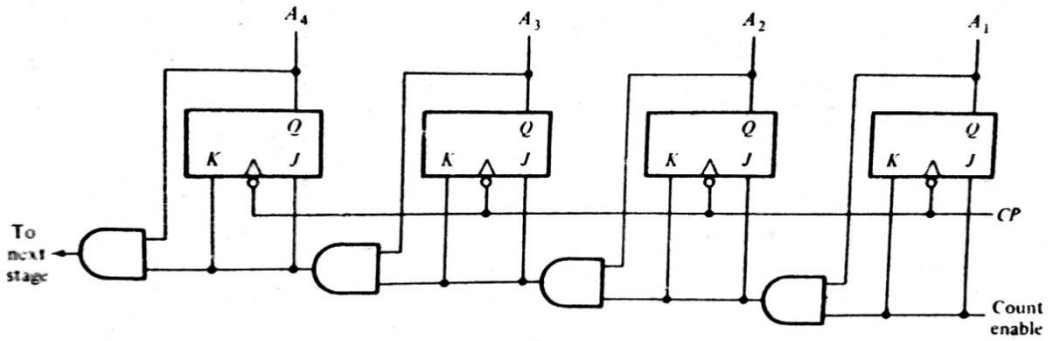


Figure 7-17 4-bit synchronous binary counter

The CP terminal of all the flip flops are connected to a common clock pulse source. The first stage A1 has its J and K equal to 1 if the counter is enabled. The other J and K inputs are equal to 1 if all previous low order bits are equal to 1 and the count is enabled. The chain of AND gates generates the required logic for the J and K inputs in each stage. The counter can be extended to any number of stages, which each stage having an additional flip flop and AND gates that gives an output of 1 if all previous flip flop outputs are 1's.

Truth table:

Clock pulse	A4	A3	A2	A1	Decimal
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	2
3	0	0	1	1	3
4	0	1	0	0	4
5	0	1	0	1	5
6	0	1	1	0	6
7	0	1	1	1	7
8	1	0	0	0	8
9	1	0	0	1	9
10	1	0	1	0	10
11	1	0	1	1	11
12	1	1	0	0	12
13	1	1	0	1	13
14	1	1	1	0	14
15	1	1	1	1	15
16	0	0	0	0	0

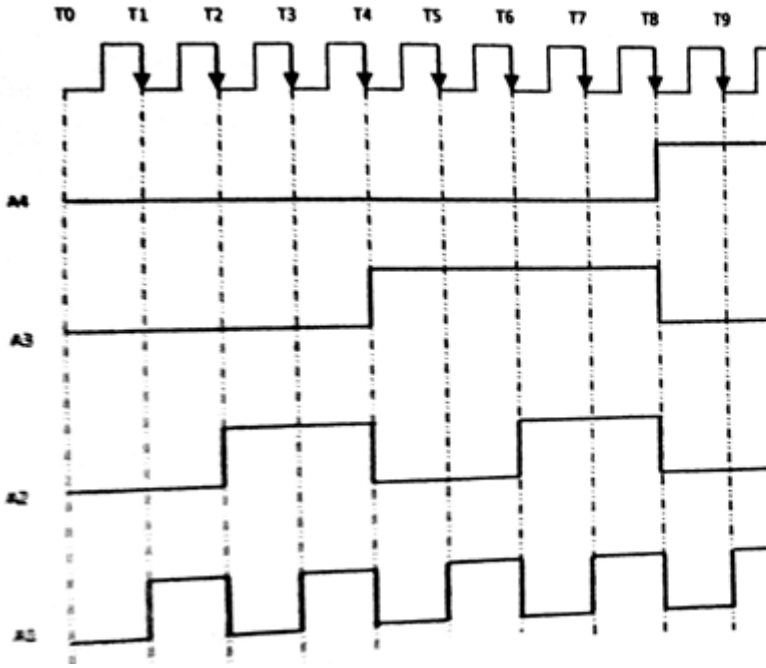


Fig: Wave diagram of 4 bit binary up counter

Down Counter:

A 4-bit down counter is a digital counter circuit, which provides a binary countdown from binary 1111 to 0000.

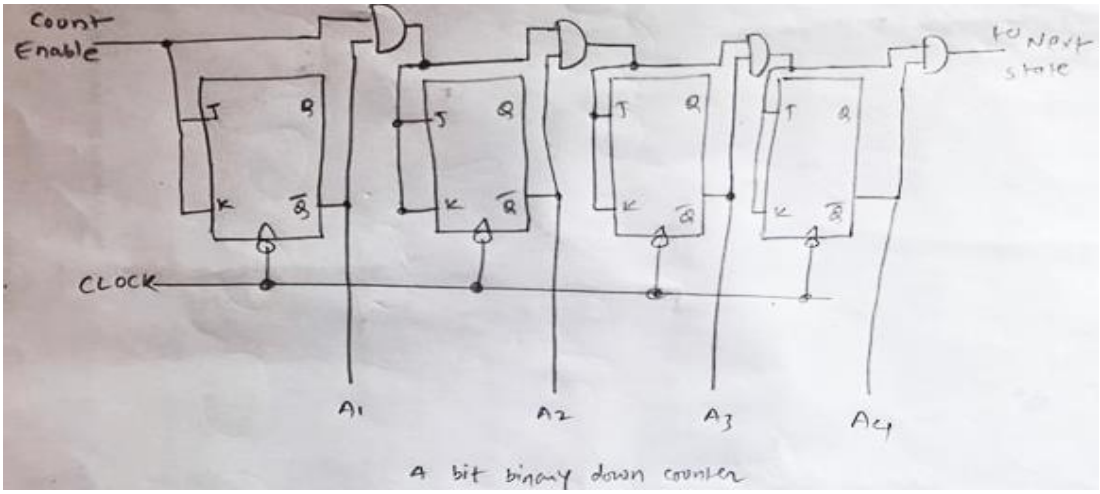


Figure: 4 bit binary down counter

The inputs to the AND gates must come from the complement outputs Q' and not from the normal output Q of the previous flip flops.

Truth table

Clock pulse	A4	A3	A2	A1
0	1	1	1	1
1	1	1	1	0
2	1	1	0	1
3	1	1	0	0
4	1	0	1	1
5	1	0	1	0
6	1	0	0	1
7	1	0	0	0
8	0	1	1	1
9	0	1	1	0
10	0	1	0	1
11	0	1	0	0
12	0	0	1	1
13	0	0	1	0
14	0	0	0	1
15	0	0	0	0

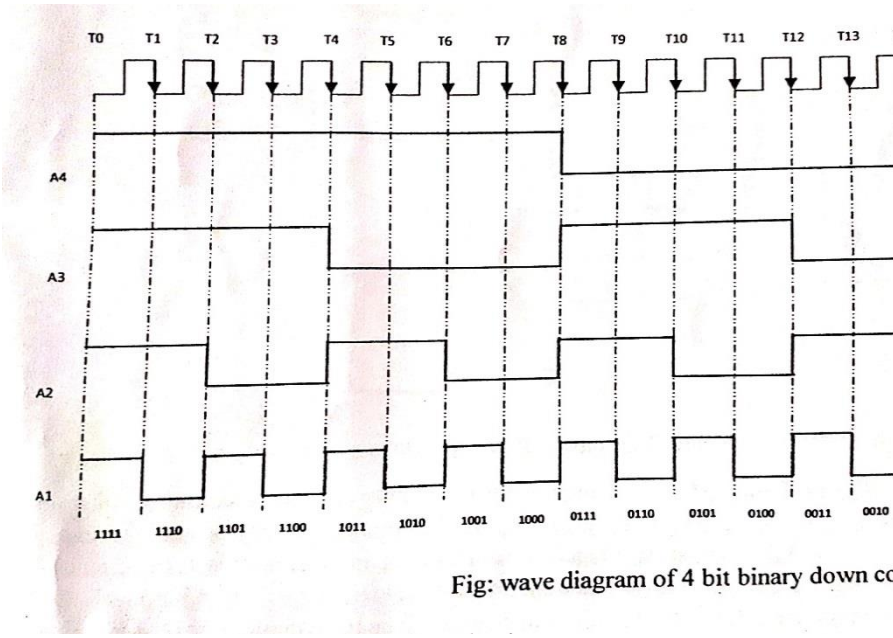


Fig: wave diagram of 4 bit binary down co

Binary UP/Down Counter (Bidirectional Counter)

Up-down counters can count both upwards as well as downwards. The two operations can be combined in one circuit. Bidirectional counters also known as Up/Down counters, are capable of counting in either direction through any given count sequence and they can be reversed at any point within their count sequence by using an additional control input.

A binary up/down counter can be constructed as given below. T flip flops employed in this circuit may be considered as JK flip flops with J and K terminals tied together.

- ⊕ When the UP input control is 1, the circuit counts UP, since the T inputs are determined from the normal outputs in Q.
- ⊕ When the down input control is 1, the circuit counts down, since the complement Q' determines the state of the T inputs.
- ⊕ When both the UP and DOWN signals are 0's, the register does not change state and remains in the same count.

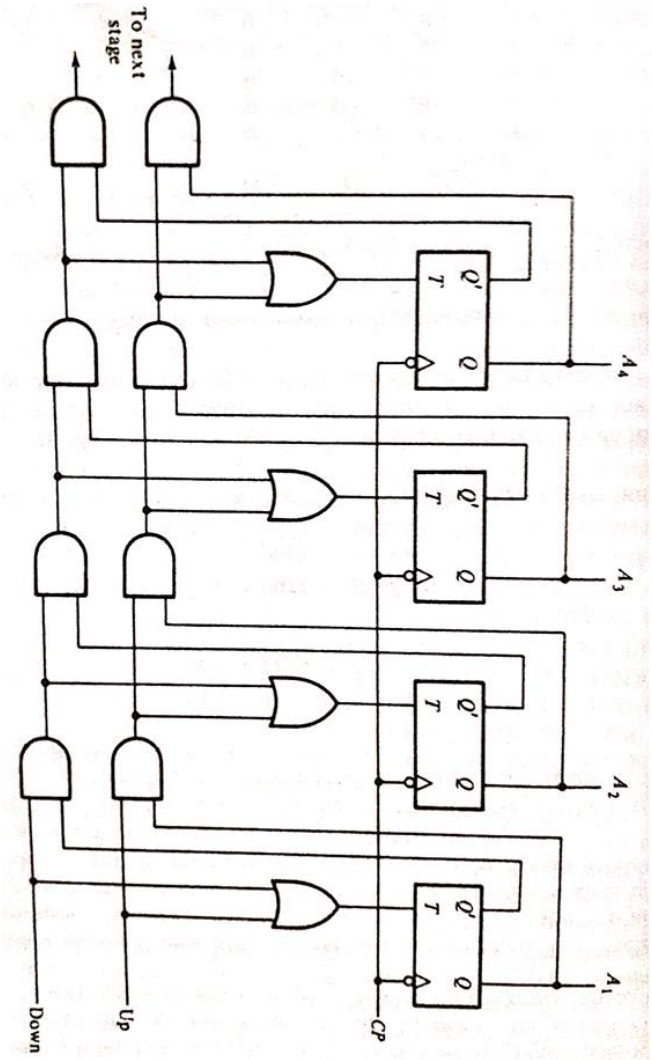


Fig: 4 bit up/down binary counter

Difference between Synchronous and asynchronous counters

SYNCHRONOUS COUNTER	ASYNCHRONOUS COUNTER
In synchronous counter, all flip flops are triggered with same clock simultaneously.	In asynchronous counter, different flip flops are triggered with different clock, not simultaneously.
Synchronous Counter is faster than asynchronous counter in operation.	Asynchronous Counter is slower than synchronous counter in operation.
Synchronous Counter does not produce any decoding errors.	Asynchronous Counter produces decoding error.
Synchronous Counter is also called **Parallel Counter.**	Asynchronous Counter is also called **Serial Counter**.
Synchronous Counter designing as well implementation are complex due to increasing the number of states.	Asynchronous Counter designing as well as implementation is very easy.
Synchronous Counter will operate in any desired count sequence.	Asynchronous Counter will operate only in fixed count sequence (UP/DOWN).
Synchronous Counter examples are: Ring counter, Johnson counter.	Asynchronous Counter examples are: Ripple UP counter, Ripple DOWN counter.
It is expensive to design	It is cheaper to design

Counter Applications

Some of the application of counters are:

- Digital clocks
- Analog to digital converters
- Frequency counters
- Frequency divider circuits
- Time measurement in devices like washing machines, microwave ovens

Practicable example of counter:

- In our kitchen appliances, we use microwave ovens. In that we set some temperature to heat the food item kept in it. Internally the counter calculates the increase or decrease in temperature and time. If it reaches the pre-set temperature, then it prevents from further heating and spoiling of that food item.
- Washing machines: We use counters in washing machines also. Similar to the counting operation in microwave oven, the counter in washing machine counts the time which we set it to operate.
- To calculate the number of people entering and leaving a stadium or auditorium we use, counters at entry gate or door.

Basic shift Register Operations

We know that one flip-flop can store one-bit of information. In order to store multiple bits of information, we require multiple flip-flops. The group of flip-flops, which are used to hold store the binary data is known as register.

If the register is capable of shifting bits either towards right hand side or towards left hand side is known as **shift register**. An 'N' bit shift register contains 'N' flip-flops.

Shift registers are basically a type of register which have the ability to transfer ("shift") data. Registers are generically storage devices which are created by connecting a specific number of flip flops together in series and the amount of data (number of bits) which can be stored by the register is always directly proportional to the number of flip flops, as each flip flop is capable of storing only one bit at a time. When the flip-flops in a register are connected in such a way that the output of one flip flop, becomes the input of the other, a shift register is created. There are different kinds of flip flops, but the most commonly used in the creation of shift registers are the D flip flops. It is because when there is change on the clock pulse (CP), the output of the D flip flop will be same until the next clock cycle. The truth table of D flip flop is shown below:

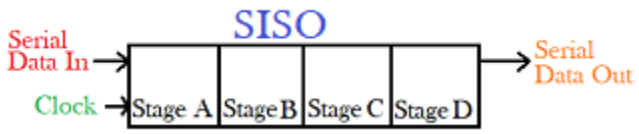
Clock	Input	Output	
	D	Q	Q'
0	X	-	-
1	0	0	1
1	1	1	0

Shift Register Types

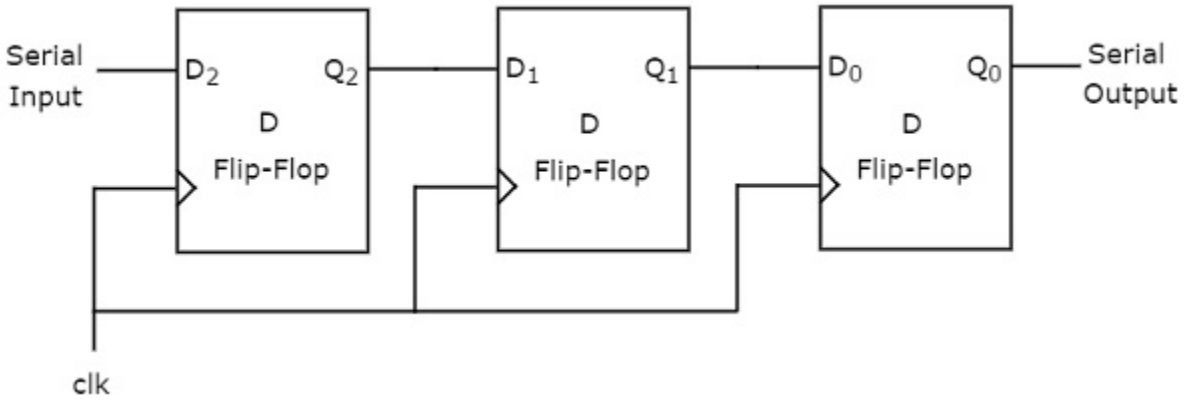
Following are the four types of shift registers based on applying inputs and accessing of outputs.

1. Serial In – Serial Out shift register (SISO)
2. Serial In – Parallel Out shift register (SIPO)
3. Parallel In – Serial Out shift register (PISO)
4. Parallel In – Parallel Out shift register (PIPO)
5. Bidirectional Shift Registers

1. Serial In – Serial Out (SISO) Shift Register



The shift register, which allows serial input and produces serial output is known as Serial In – Serial Out (SISO) shift register. The block diagram of 3-bit SISO shift register is shown in the following figure.



This block diagram consists of three D flip-flops, which are cascaded. That means, output of one D flip-flop is connected as the input of next D flip-flop. All these flip-flops are synchronous with each other since, the same clock signal is applied to each one.

In this shift register, we can send the bits serially from the input of left most D flip-flop. Hence, this input is also called as serial input. For every positive edge triggering of clock signal, the data shifts from one stage to the next. So, we can receive the bits serially from the output of right most D flip-flop. Hence, this output is also called as serial output.

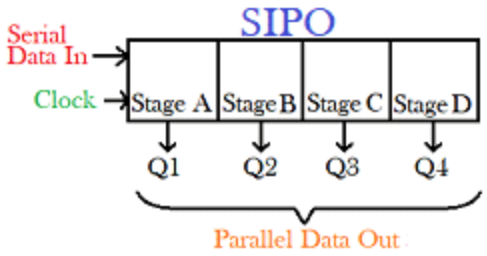
Let us see the working of 3-bit SISO shift register by sending the binary information **"011"** from LSB to MSB serially at the input. Assume, initial status of the D flip-flops from leftmost to rightmost is $Q_2Q_1Q_0$ = 0 0 0. We can understand the working of 3-bit SISO shift register from the following table.

No of CP	Serial Input	Q_2	Q_1	Q_0
0	-	0	0	0
1	1	1 LSB	0	0
2	1	1	1	0
3	0	0 MSB	1	1 LSB
4	-	-	0	1
5	-	-	-	0 MSB

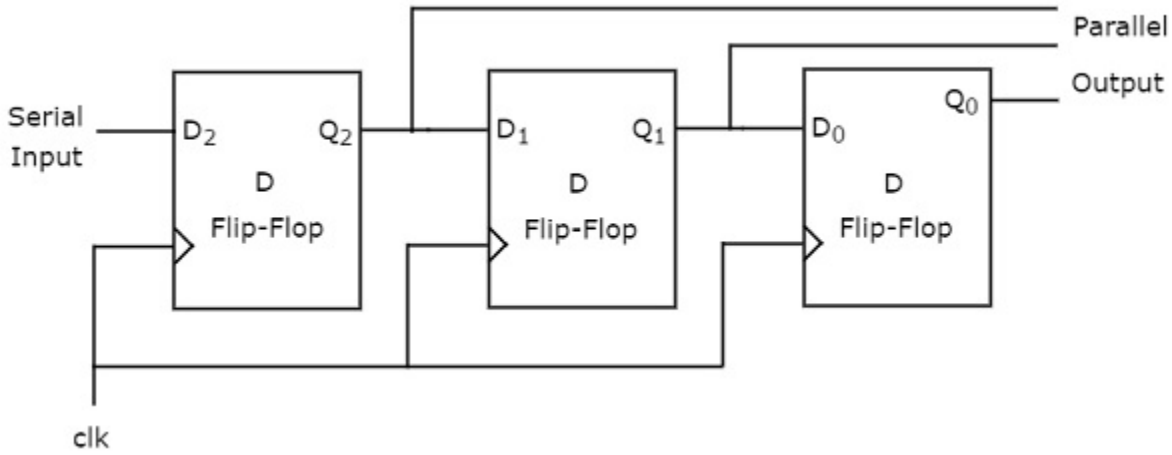
The initial status of the D flip-flops in the absence of clock signal is $Q_2Q_1Q_0$=0 0 0. Here, the serial output is coming from Q_0. So, the LSB 1 is received at 3rd positive edge of clock and the MSB 0 is received at 5th positive edge of clock.

Therefore, the 3-bit SISO shift register requires five clock pulses in order to produce the valid output. Similarly, the N-bit SISO shift register requires 2N-1 clock pulses in order to shift 'N' bit information.

2. Serial In – Parallel Out shift register (SIPO)



The shift register, which allows serial input and produces parallel output is known as Serial In – Parallel Out SIPO shift register. The block diagram of 3-bit SIPO shift register is shown in the following figure.



This circuit consists of three D flip-flops, which are cascaded. That means, output of one D flip-flop is connected as the input of next D flip-flop. All these flip-flops are synchronous with each other since, the same clock signal is applied to each one.

In this shift register, we can send the bits serially from the input of left most D flip-flop. Hence, this input is also called as serial input. For every positive edge triggering of clock signal, the data shifts from one stage to the next. In this case, we can access the outputs of each D flip-flop in parallel. So, we will get parallel outputs from this shift register.

Let us see the working of 3-bit SIPO shift register by sending the binary information **"011"** from LSB to MSB serially at the input.

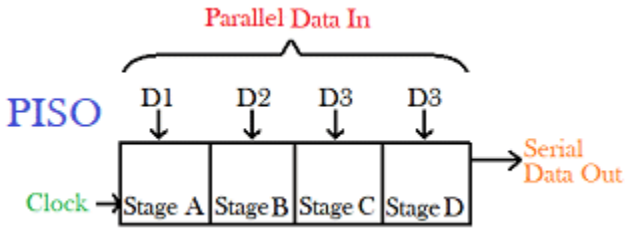
Assume, initial status of the D flip-flops from leftmost to rightmost is $Q_2Q_1Q_0$=0 0 0. Here, Q_2 & Q_0 are MSB & LSB respectively. We can understand the working of 3-bit SIPO shift register from the following table.

No of CP	Serial Input	Q_2 (MSB)	Q_1	Q_0 (LSB)
0	-	0	0	0
1	1	1 (LSB)	0	0
2	1	1	1	0
3	0	0 (MSB)	1	1

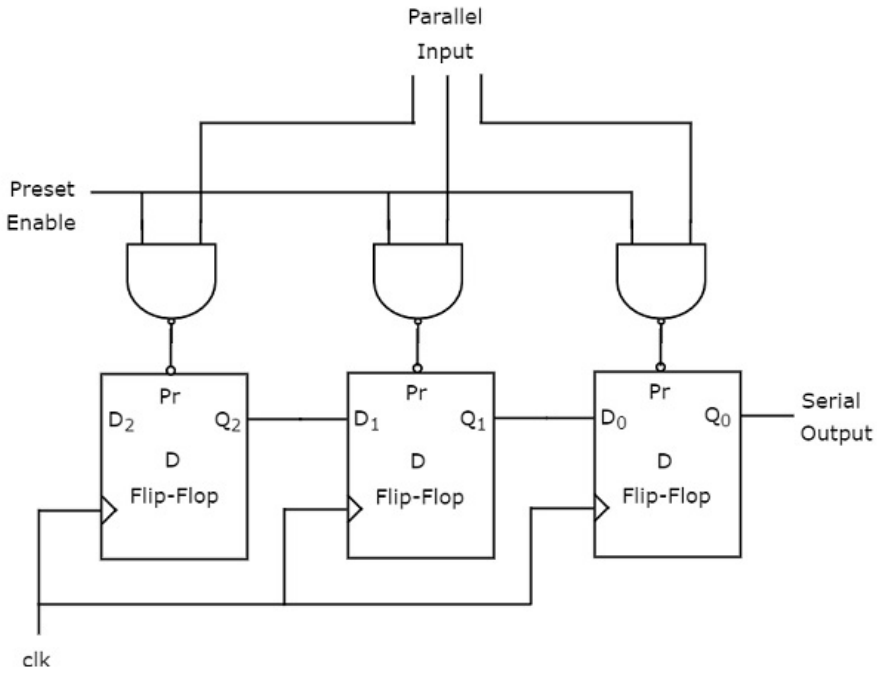
The initial status of the D flip-flops in the absence of clock signal is $Q_2Q_1Q_0$= 0 0 0. The binary information "011" is obtained in parallel at the outputs of D flip-flops for third positive edge of clock.

So, the 3-bit SIPO shift register requires three clock pulses in order to produce the valid output. Similarly, the N-bit SIPO shift register requires N clock pulses in order to shift 'N' bit information.

3. Parallel In – Serial Out shift register (PISO)



The input to this register is given in parallel i.e. data is given separately to each flip flop and the output is collected in serial at the output of the end flip flop. The clock pulse is directly connected to all the flip flops. The block diagram of 3-bit PISO shift register is shown in the following figure.



This circuit consists of three D flip-flops, which are cascaded. That means, output of one D flip-flop is connected as the input of next D flip-flop. All these flip-flops are synchronous with each other since, the same clock signal is applied to each one.

In this shift register, we can apply the parallel inputs to each D flip-flop by making Preset Enable to 1. For every positive edge triggering of clock signal, the data shifts from one stage to the next. So, we will get the serial output from the right most D flip-flop.

Let us see the working of 3-bit PISO shift register by applying the binary information "011" in parallel through preset inputs.

Since the preset inputs are applied before positive edge of Clock, the initial status of the D flip-flops from leftmost to rightmost will be $Q_2Q_1Q_0$=011. We can understand the working of 3-bit PISO shift register from the following table.

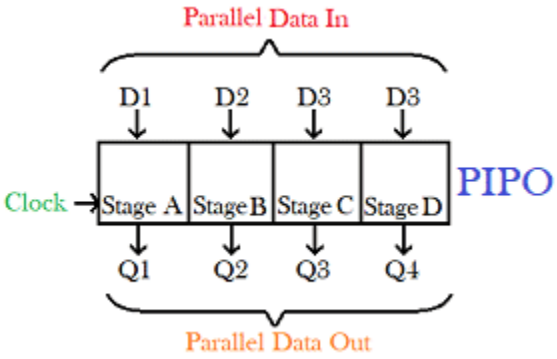
No of CP	Q_2	Q_1	Q_0
0	0	1	1 (LSB)
1	-	0	1
2	-	-	0 (MSB)

Here, the serial output is coming from Q_0. So, the LSB 1 is received before applying positive edge of clock and the MSB 0 is received at 2nd positive edge of clock.

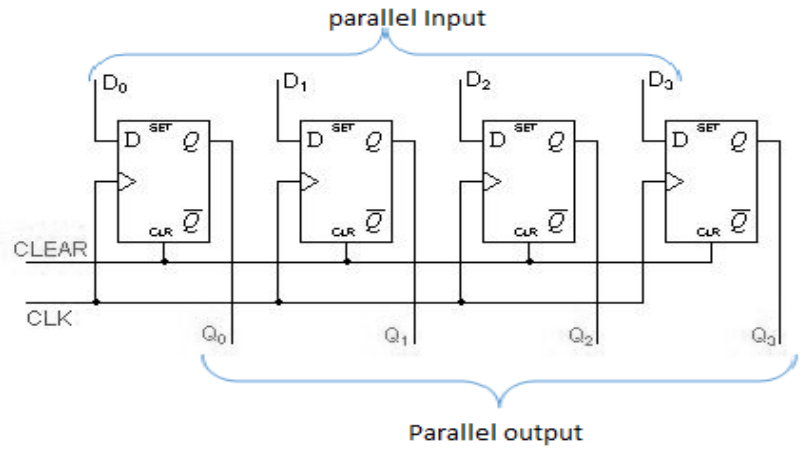
Therefore, the 3-bit PISO shift register requires two clock pulses in order to produce the valid output. Similarly, the N-bit PISO shift register requires N-1 clock pulses in order to shift 'N' bit information.

4. Parallel In – Parallel Out shift register (PIPO)

In this register, the input is given in parallel and the output also collected in parallel. The clear (CLR) signal are connected to all the flip flops. A PIPO shift registers are used as temporary storage devices and also as a delay elements.

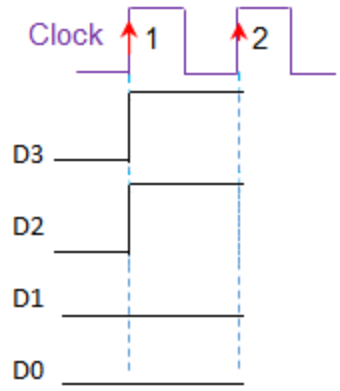


The block diagram of 4-bit PIPO shift register is shown in the following figure.



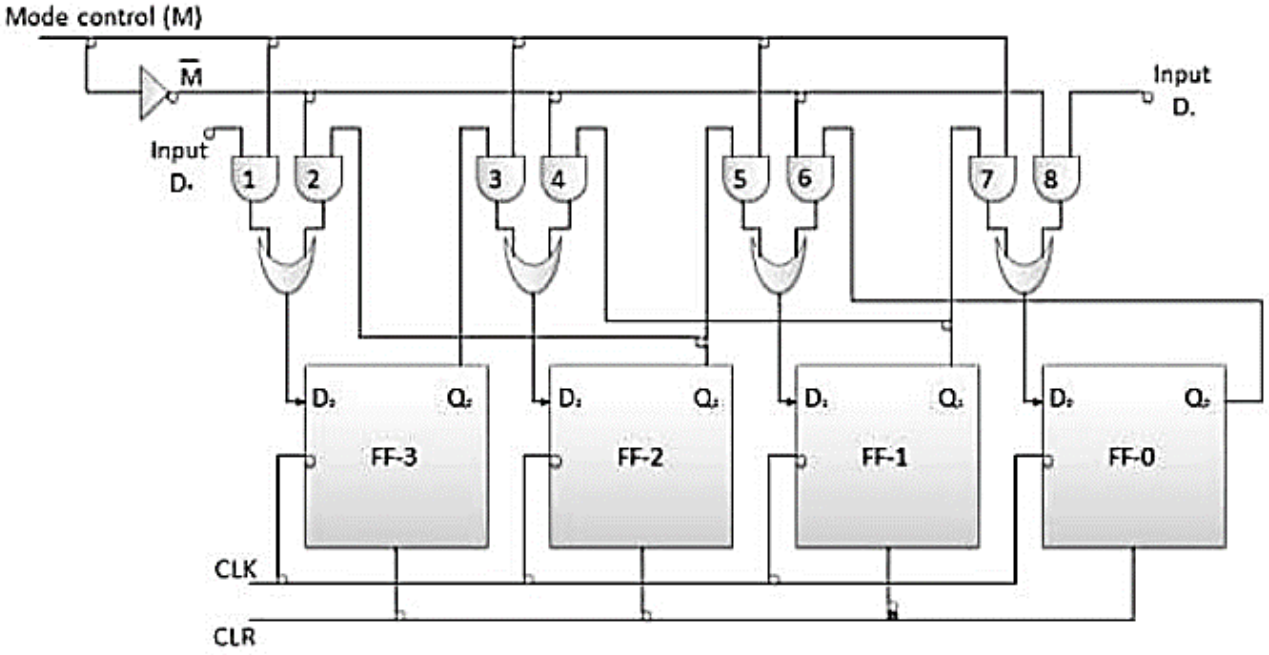
Let the data input be 1100

No of CP	D3	D2	D1	D0
0	0	0	0	0
1	1	1	0	0



5. Bidirectional Shift Registers

- ⊕ Bidirectional shift register allows shifting of data either to left or to the right side.
- ⊕ It can be implemented using logic gates circuitry that enables the transfer of data from one stage to the next stage to the right or to the left, depend on the level of control line.
- ⊕ The RIGHT/LEFT is the control input signal which allows data shifting either towards right or towards left.
- ⊕ A high on this line enables the shifting of data towards right and low enables it towards left.
- ⊕ When RIGHT/LEFT is high, gates G1, G2, G3 and G4 are enabled.
- ⊕ The state of Q output of each flip flop is passed through the D input of the following flip flop.
- ⊕ When the pulse arrives, the data are shifted one place to the right.
- ⊕ When the RIGHT/LEFT signal is low, gates G5, G6, G7 are enabled.
- ⊕ The Q output of each flip-flop is passed through the D input of the preceding flip-flop.



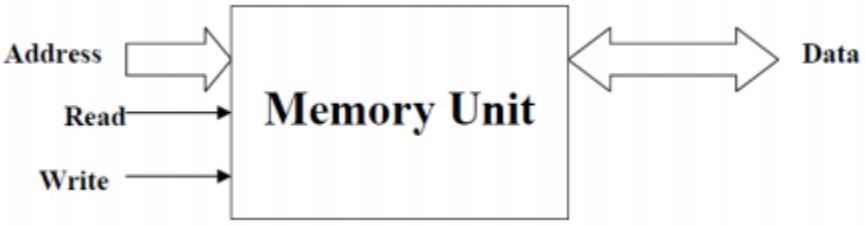
Basic Memory Operations and Memory types

A memory unit is a collection of storage registers together with the associated circuits needed to transfer information in and out of the registers. The storage registers in a memory units are called memory registers. The components that forms the binary cells o registers in a memory unit must have certain basic properties. Some of the basic properties are:

1. It must have a reliable two state property for binary representation
2. It must be small in size
3. The cost per bit of storage must be as low as possible.
4. The time of access to a memory register should be relatively fast.

Two Basic Memory Operations

The memory unit supports two fundamental operations: Read and Write. The read operation read a previously stored data and the write operation stores a value in memory, see the figure below.



The following steps have to be followed in a typical read cycle:

1. Place the address of the location to be read on the address bus.
2. Activate the memory read control signal on the control bus.
3. Wait for the memory to retrieve the data from the address memory location.
4. Read the data from the data bus.
5. Drop the memory read control signal to terminate the read cycle.

The following steps have to be followed in a typical read cycle:

1. Place the address of the location to be written on the address bus.
2. Place the data to be written on the data bus.
3. Activate the memory write control signal on the control bus.
4. Wait for the memory to store the data at the address location.
5. Drop the memory write control signal to terminate the write cycle

Types of Memory

Generally, memory/storage is classified into 2 categories:

1. **Volatile Memory:** This loses its data, when power is switched off.
2. **Non-Volatile Memory:** This is a permanent storage and does not lose any data when power is switched off.

The total memory capacity of a computer can be visualized by hierarchy of components. The memory hierarchy system consists of all storage devices contained in a computer system from the slow Auxiliary Memory to fast Main Memory and to smaller Cache memory.

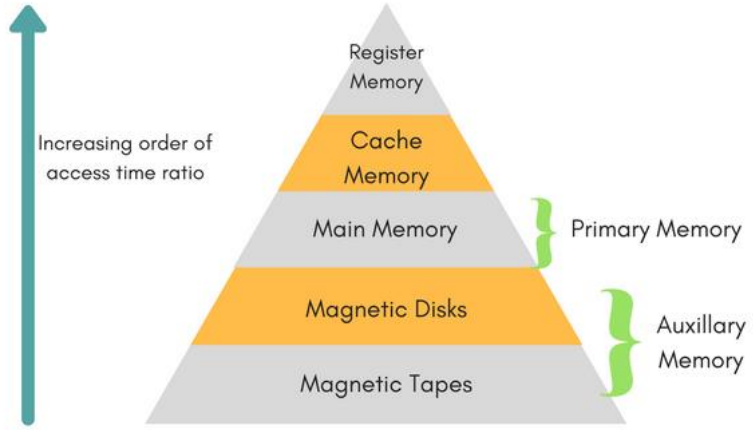
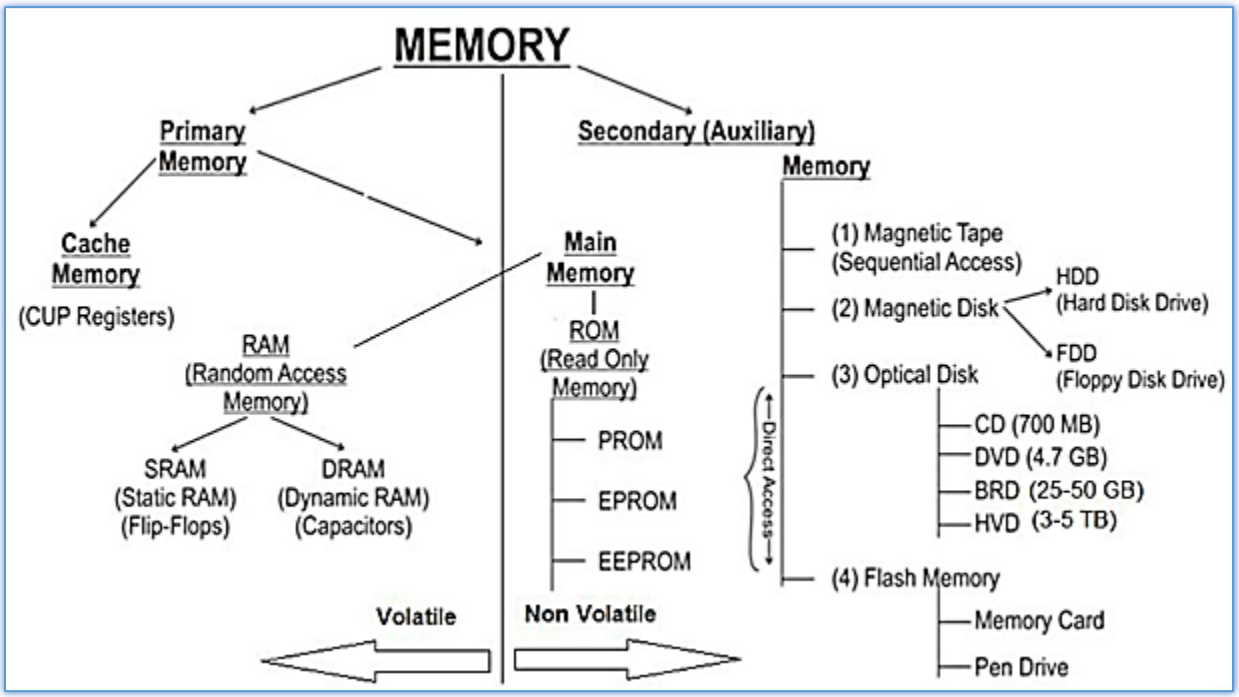


Figure1: Memory Hierarchy

Types of Computer Memory:



What is a Mod n counter?

Mod n or Modulus of n, is a way of referring to the maximum count of a counter. Every counter has a limit with regards to the number they can count up or down to. Mod n expresses that limit.

It is an important label for a counter because it gives us the maximum count of the counter, as well as the number of flip-flops present in the counter.

Example: Mod 8 counter
Mod 8 means n = 8.

$8 = 2^N$
Thus, N = 3.
Which means that this is a counter with three flip-flops, which means three bits, having eight stable states (000 to 111) and capable of counting eight events or up to the decimal number $2^N – 1 = 7$.

* Therefor a “MOD N” counter will require “N” number of flip flops connected together to count a single data bit and provide 2^n different output states.

- End of Unit 6 -