Sequential Logic Circuits:

Logic circuits are defined into two types, “combinational” and “sequential.” A combinational logic circuit is one whose outputs depend only on its current inputs. A sequential logic circuit is one whose outputs depend not only on its current inputs, but also on the past sequence of inputs, possibly arbitrarily far back in time. So it is impossible to describe the until the current time.

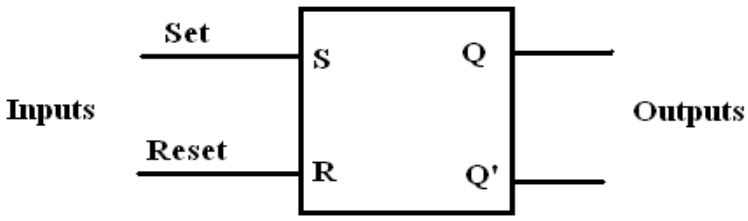
The basic memory elements of sequential cct. Are called Flip Flops, because of it's capability of storing information .

Flip Flop (F/F or F-F) :-

The flip flop is a logic cct that is capable of storing a single bit (0 or 1) of information and hold that bit as long as required if the power is continued to supply, until directed by an input signal to switch state.

1) The Set – Reset flip flop (RS F/F):

The simplest type of F/F , it's also known by the basic F/F.



Symbol of RS F/F

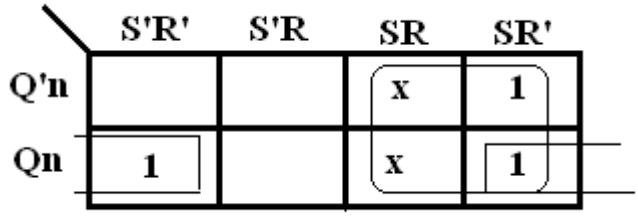
The Truth table of RS F/F is

S	R	Q	
0	0	N.C	No change
0	1	0	Reset
1	0	1	Set
1	1	x	Undefined

The truth table QR (characteristic table) is

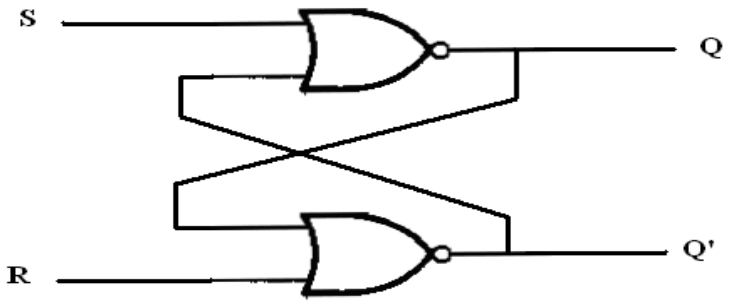
S	R	Qn	Qn+1	
0	0	0	0	No change
0	0	1	1	
0	1	0	0	Reset
0	1	1	0	
1	0	0	1	Set
1	0	1	1	
1	1	0	x	Undefined
1	1	1	x	

Where :
Qn : is the present state .
Qn+1 : is the next state.

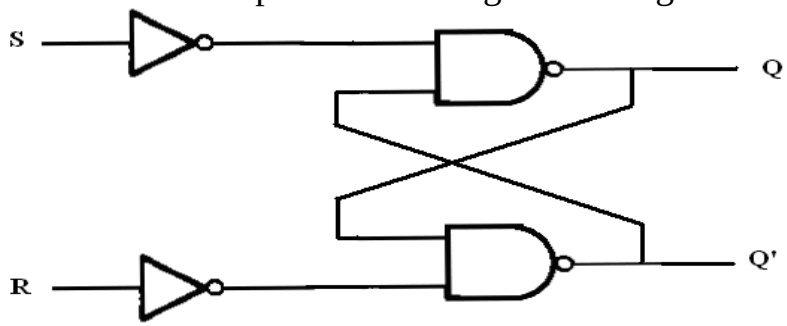


Qn+1 = S + R' Qn

The RS F/F can be implemented using 2 NOR gates as below

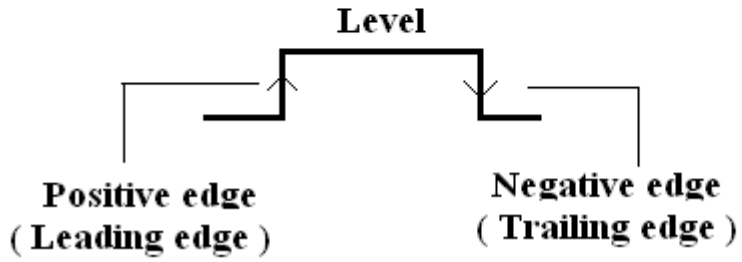


The RS F/F can be implemented using 2 NAND gates as below

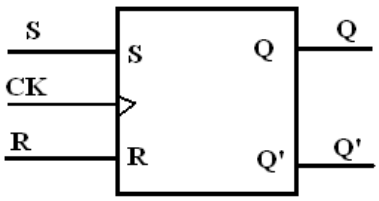


2) Clocked RS F/F :

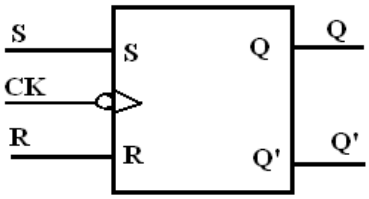
A clock pulse is a train of a sequare wave pulses .



The RS F/F depends on (Level) , but the clocked RS F/F depends on the (Positive edge) or (Negative edge).



Positive edge clocked RS F/F

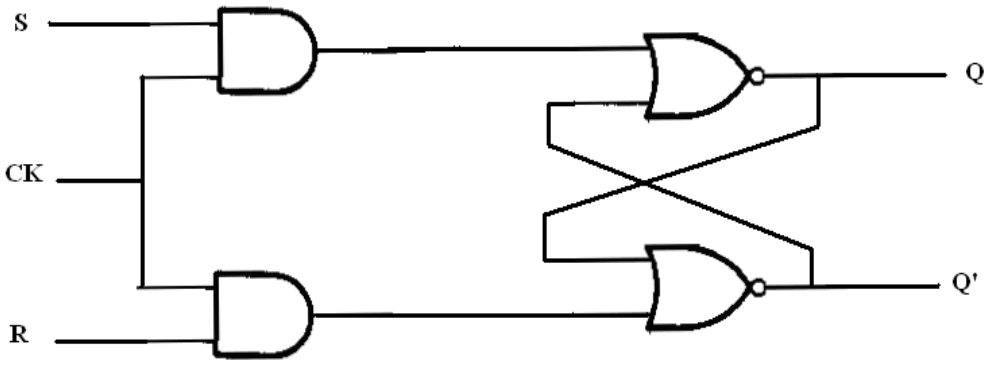


Negative edge clocked RS F/F

The truth table of clocked RS F/F is

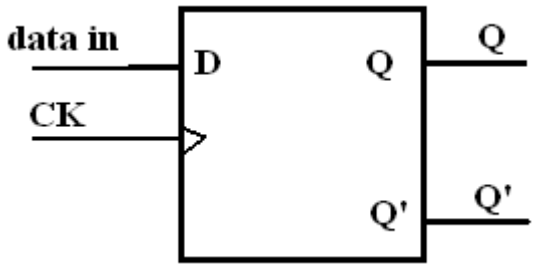
CK	S	R	Q
↑	0	0	N.C
↑	0	1	0
↑	1	0	1
↑	1	1	x

The clocked RS F/F can be implemented as below :-

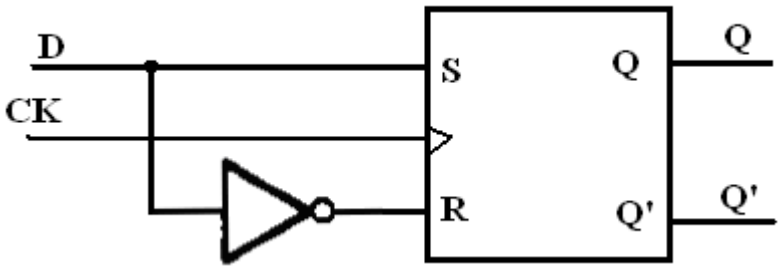


3) D Flip – Flops :-

Delay flip flop or data flip flop. The clocked D F/F some times called D- latch is a modification of the clocked RS F/F.



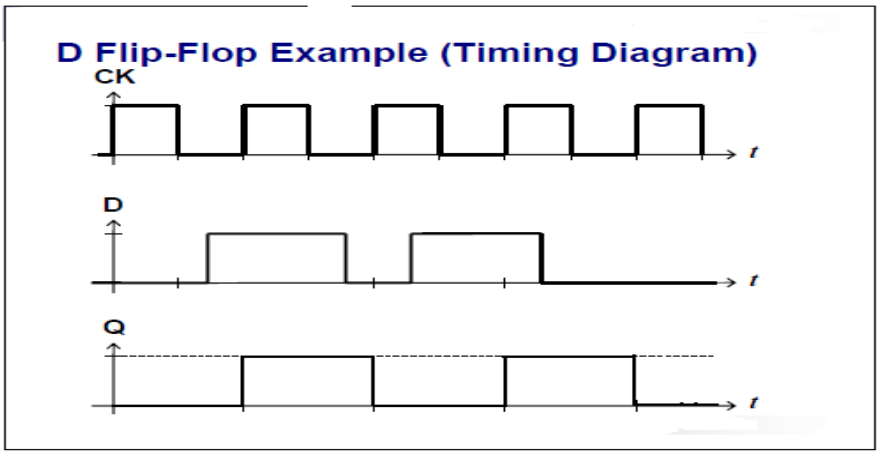
It is easily implemented using RS F/F by applying the input data (signal) to the S input and the complement of the input signal to the R input as shown in figure below.



Truth table of D F/F is :

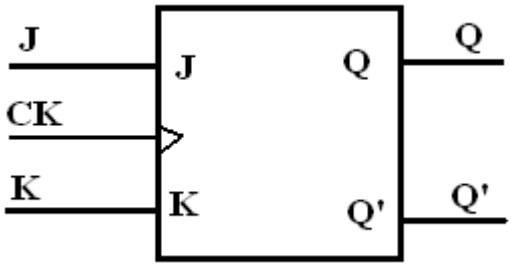
CK	D	Qn	Qn+1
↑	0	0	0
↑	0	1	0
↑	1	0	1
↑	1	1	1

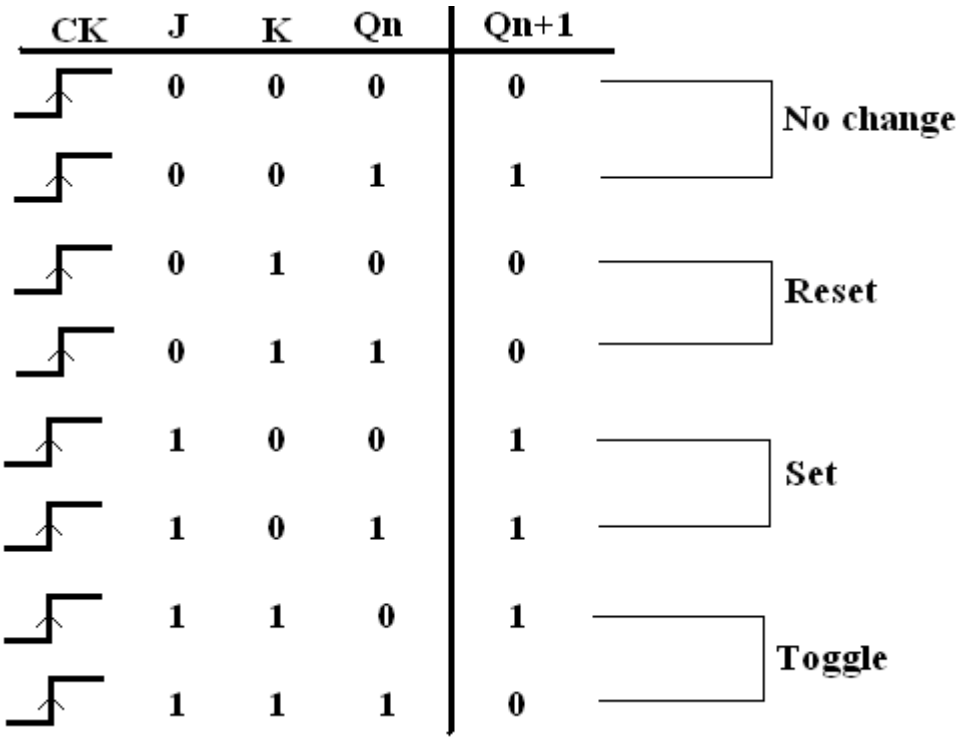
Qn+1 = D



4) JK flip flop :

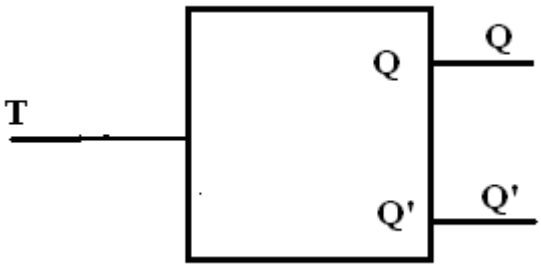
The clocked JK F/F is an RS F/F that has been modified by feeding the outputs back and ANDing with the inputs , the symbol of JK F/F is shown below :





5) Toggle Flip Flop (T F.F)

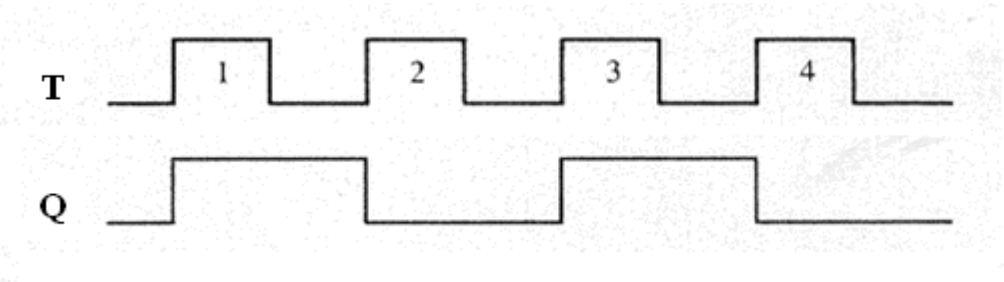
Toggle flip flop is a single input version of the JK F.F where the (J and K) are connected to (Logic 1)



Symbol of T F.F

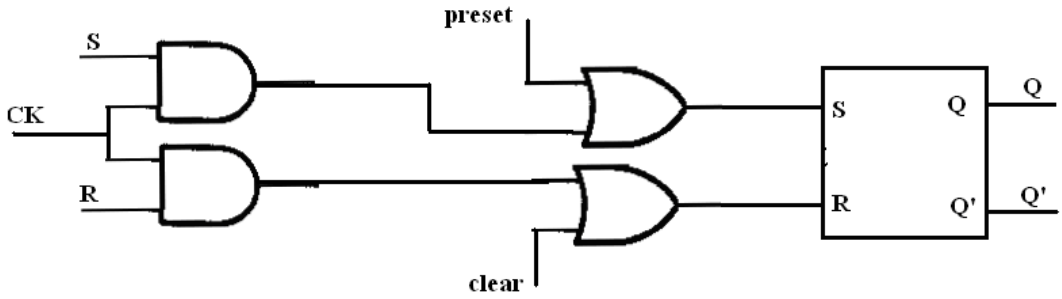
T	Qn	Qn+1	
0	0	0	No change
0	1	1	
1	0	1	Toggle
1	1	0	

Truth table of T F.F



Preset and Clear :

Preset and clear are a synchronous inputs because they activate the F.F independently of the clock pulse.



preset	clear	Q	
1	0	1	Set
0	1	0	Reset

independent of clock

Excitation Table (Oppsite of Truth Table)

In this table , we can know the value (state) of the inputs if the output present state (Qn) and the output next state (Qn+1) are known:

Qn	Qn+1	S	R
0	0	0	x
0	1	1	0
1	0	0	1
1	1	x	0

SR F.F excitation table

Qn	Qn+1	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

JK F.F excitation table

Qn	Qn+1	T
0	0	0
0	1	1
1	0	1
1	1	0

T F.F excitation table

Qn	Qn+1	D
0	0	0
0	1	1
1	0	0
1	1	1

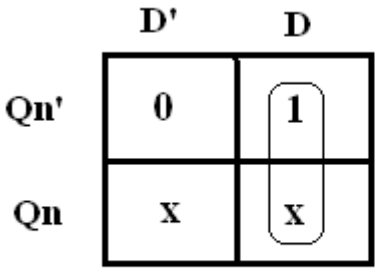
D F.F excitation table

Sequential circuit design Procedure

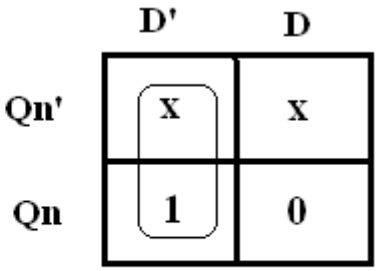
1- write the excitation table
2- draw the K- map and write the simplified expression
3- draw the logic diagram

Example : Using sequential cct design procedure convert JK F.F to D F.F

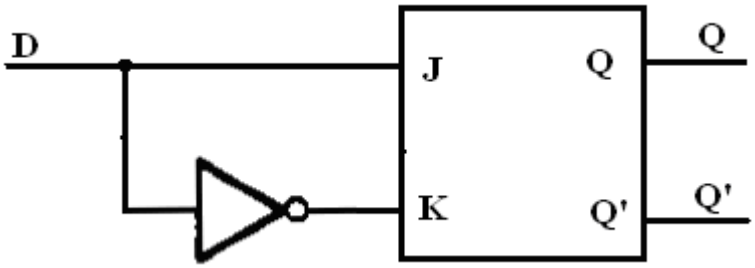
Qn	Qn+1	J	K	D
0	0	0	x	0
0	1	1	x	1
1	0	x	1	0
1	1	x	0	1



J = D



K = D'



Example : Using sequential cct design procedure convert RS F.F to T F.F

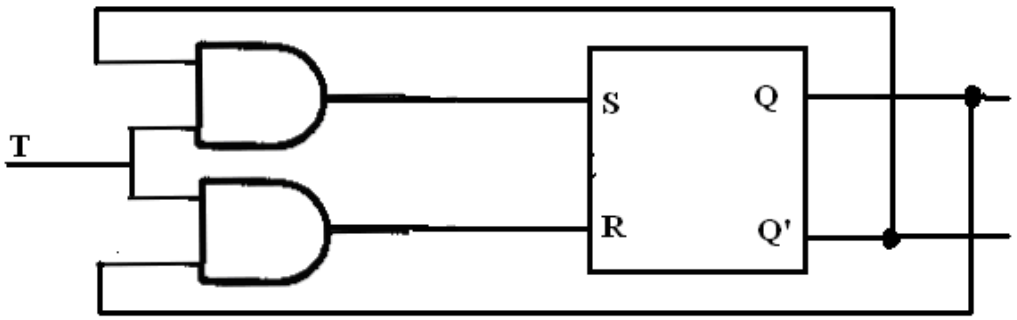
Qn	Qn+1	S	R	T
0	0	0	x	0
0	1	1	0	1
1	0	0	1	1
1	1	x	0	0

Qn' \ T' T	T'	T
Qn'	0	1
Qn	x	0

S = T Qn'

Qn \ T' T	T'	T
Qn'	x	0
Qn	0	1

R = T Qn



Counter

Such a group of flip- flops is a counter. The number of flip-flops used and the way in which they are connected determine the number of states and also the specific sequence of states that the counter goes through during each complete cycle. Counters are classified according to the way they are clocked: **asynchronous and synchronous**. Within each of these two categories, counters are classified primarily by the type of sequence, the number of states, or the number of flip-flops in the counter.

The modulus (MOD) is the number of unique states through which the counter will sequence. The maximum possible number of states of a counter is 2^n where n is the number of flip-flops. Counters can be designed to have a number of states in their sequence that is less than the maximum of 2^n. This type of sequence is called a truncated sequence. One common modulus for counters with truncated sequences is 10 (Modules10).

a)- Asynchronous Counter

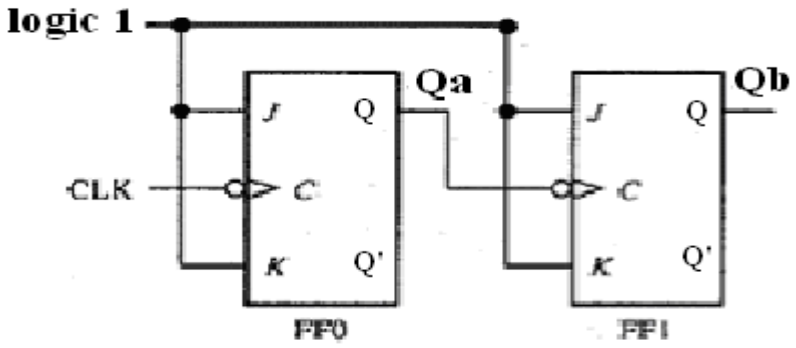
Asynchronous counters called ripple counters, the first flip-flop is clocked by the external clock pulse and then each successive flip-flop is clocked by the output of the preceding flipflop. The term asynchronous refers to events that do not have a fixed time relationship with each other . An asynchronous counter is one in which the flip-flops within the counter do not change states at exactly the same time because they do not have a common clock pulse

1) 2-Bits (MOD 4) Asynchronous Counter

Number of states (MOD) = 2^2 = 4

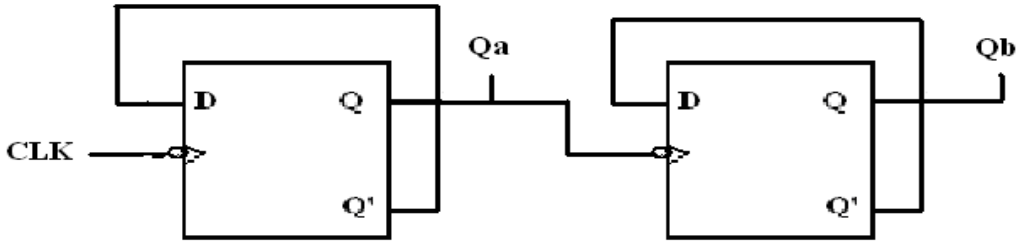
The counter can be count up or down , if the counter designed to operate with the negative edge of (CLK) ,it will count up by connect the output (Q) of the least stage to the (CLK) of the next stage , and it will count down when connect (Q') . the counter can be implemented using any type of F.F as below:

A)- 2 bits asynchronous UP counter using JK F.F



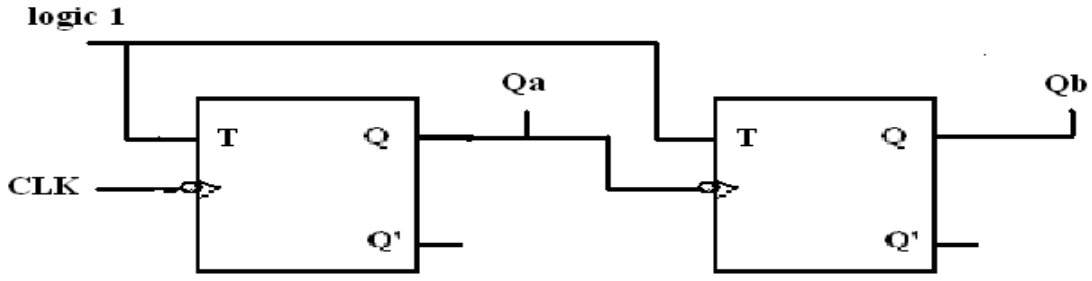
Logic diagram of 2 bits counter using JK F.F

B)- a 2 bits asynchronous UP counter using D F.F

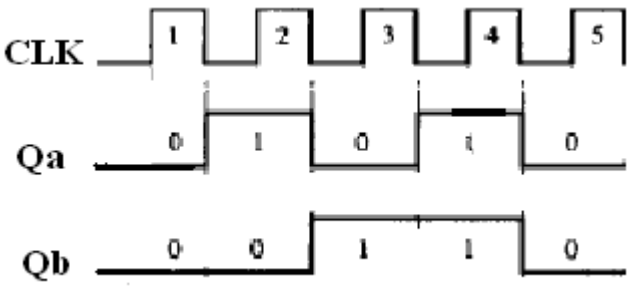


Logic diagram of 2bits up counter using D FF

C)- **2 bits asynchronous UP counter using T F.F**

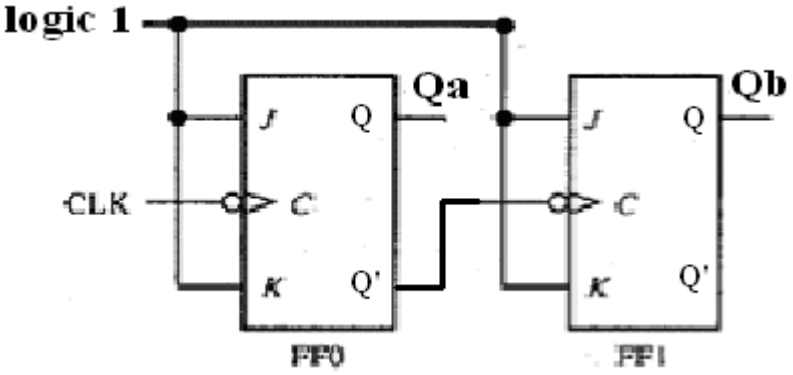


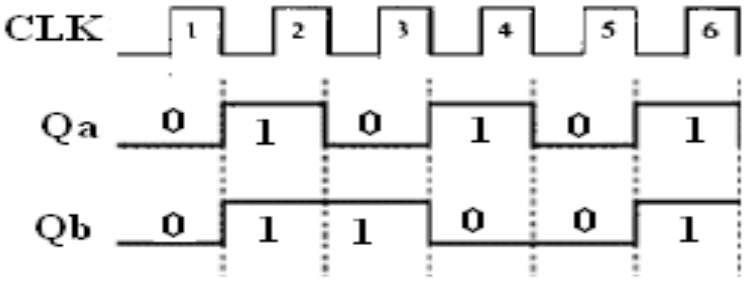
Logic diagram of 2bits up counter using T FF



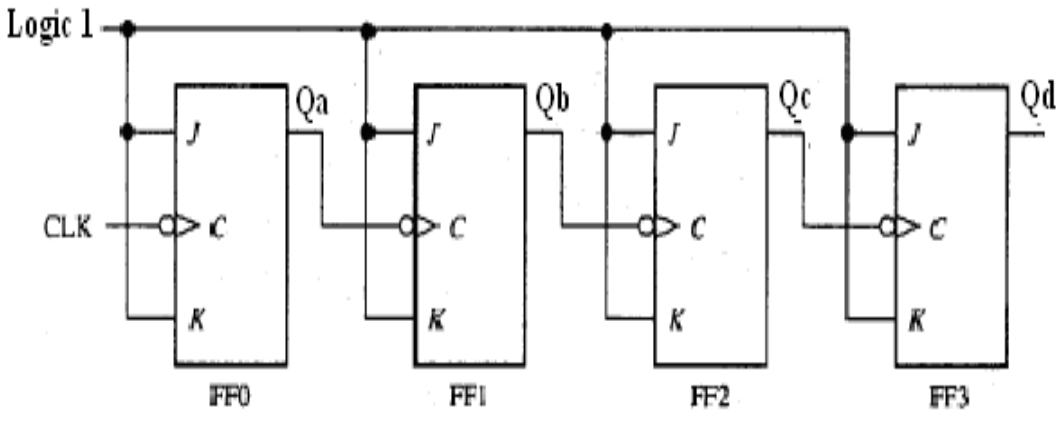
Timing diagram of 2bits up counter

D)- a 2 bits asynchronous DOWN counter using JK F.F

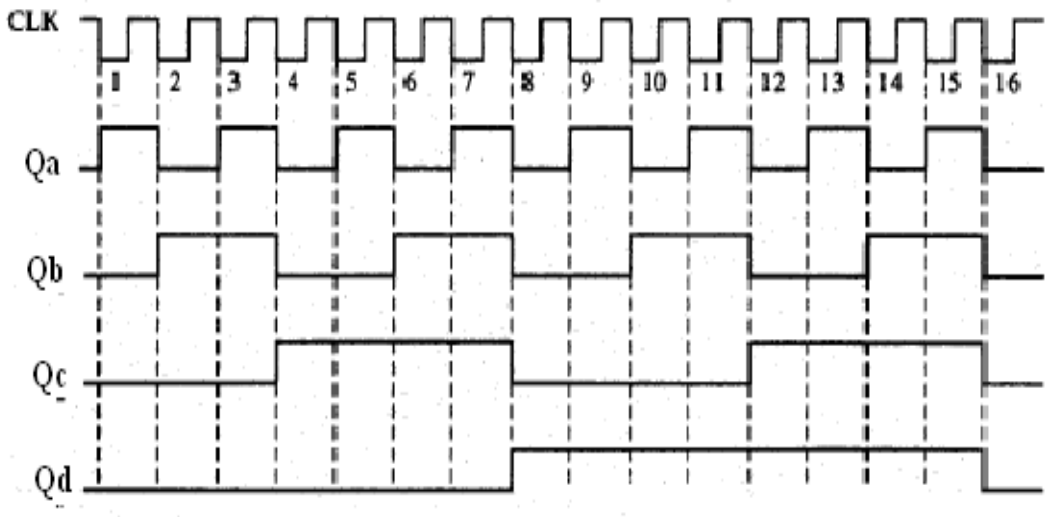




E)- a 4 bit (MOD 16) asynchronous UP counter



Logic diagram

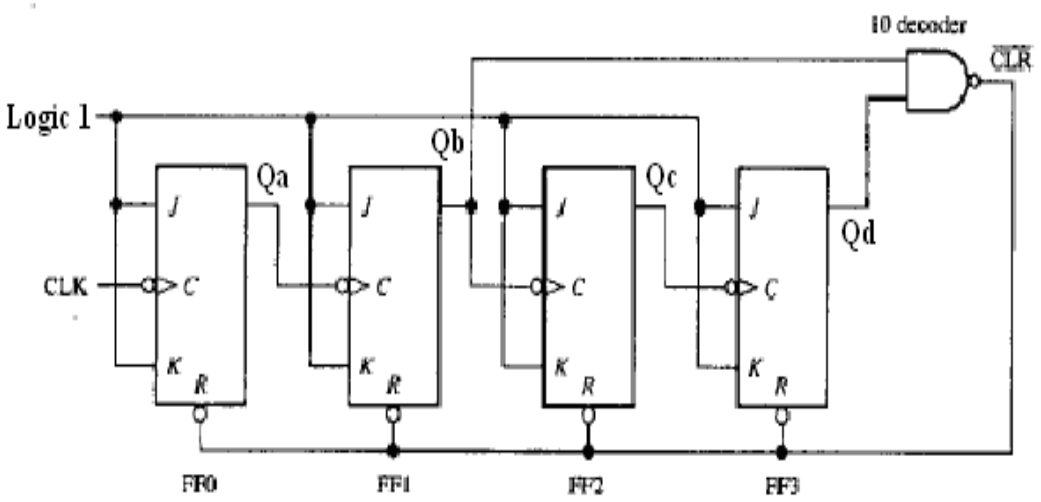


Timing diagram

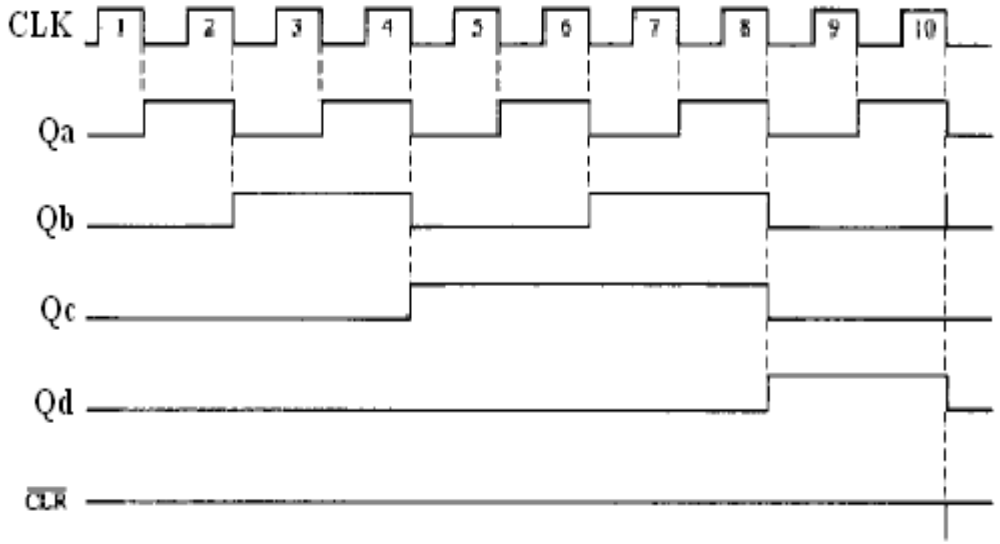
Asynchronous Counter Design Procedure

1) Select Counter Type
• Up or Down
• Modulus
2) Select Flip-Flop Type
3) Determine Number of Flip-Flops
• $2^{\text{Flip-Flops}} >=$ Modulus
4) Design Count Limit (modulus less than the maximum no. of states)
• Input to reset logic circuit is count limit plus one for up counters (minus one for down counters) the design is simple by taking the (logic 1) bits of the MOD number and connected to a NAND gate and the output of it to the (Clear) of all F.Fs. as shown in (MOD 10) below

Example : Design an asynchronous Decade counter (MOD 10)



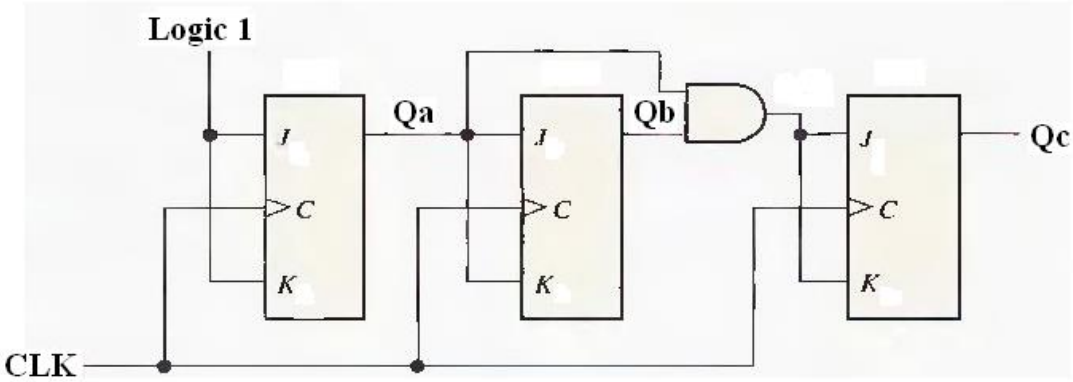
Logic diagram of decade counter



Timing diagram of decade counter

b)- Synchronous Counter

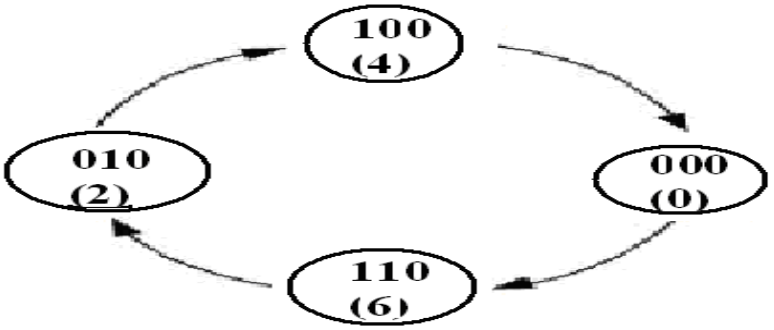
In synchronous counters, the clock input is connected to all of the flip-flops so that they are clocked simultaneously. The term synchronous refers to events that have a fixed time relationship with each other.as shown in figure below a 3 bits synchronous counter.



The synchronous counter can be designed using the sequential design procedures with the following steps:

- **draw the state flow diagram**
- **Write the state transition table**
- **from this table find the expression for each F.F input using K-map**
- **Draw the logic diagram and the timing diagram**

Example 1 : Design a counter which has 4, 0 ,6, 2 as a count sequence using T F.F.



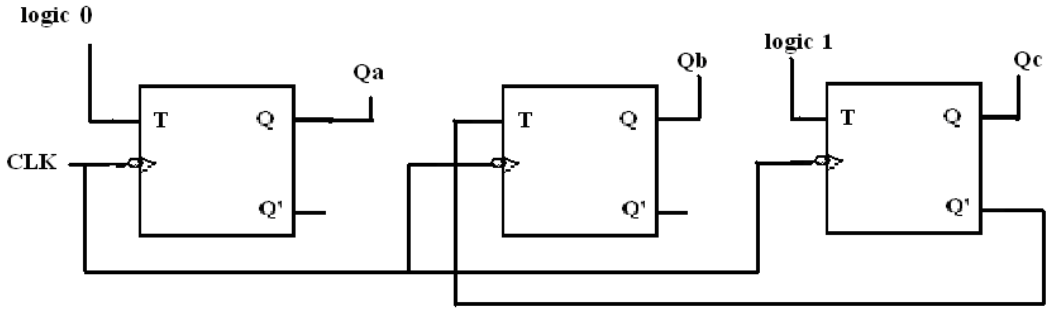
State flow diagram

Present state			Next state			F.F input		
Qc	Qb	Qa	Qc	Qb	Qa	Tc	Tb	Ta
1	0	0	0	0	0	1	0	0
0	0	0	1	1	0	1	1	0
1	1	0	0	1	0	1	0	0
0	1	0	1	0	0	1	1	0

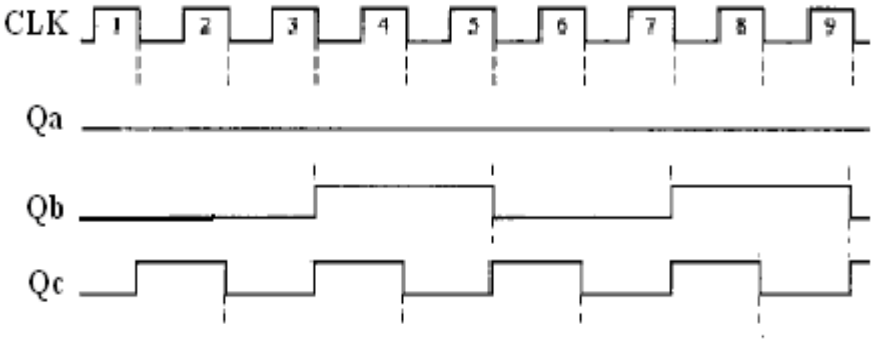
From the state transition table above
Ta = 0 ; Tc = 1

	$Q_b'Q_a'$	$Q_b'Q_a$	Q_bQ_a	Q_bQ_a'
Q_c'	1	X	X	1
Q_c	0			0

$Tb = Q_c'$



Logic diagram



Timing diagram

Example 2 : Design a counter which has 4, 0 ,6, 2 as a count sequence using JK F.F.

Present state			Next state			F.F Inputs		
Qc	Qb	Qa	Qc	Qb	Qa	JcKc	JbKb	JaKa
1	0	0	0	0	0	x 1	0 x	0 x
0	0	0	1	1	0	1 x	1 x	0 x
1	1	0	0	1	0	x 1	x 0	0 x
0	1	0	1	0	0	1 x	x 1	0 x

From the states transition table

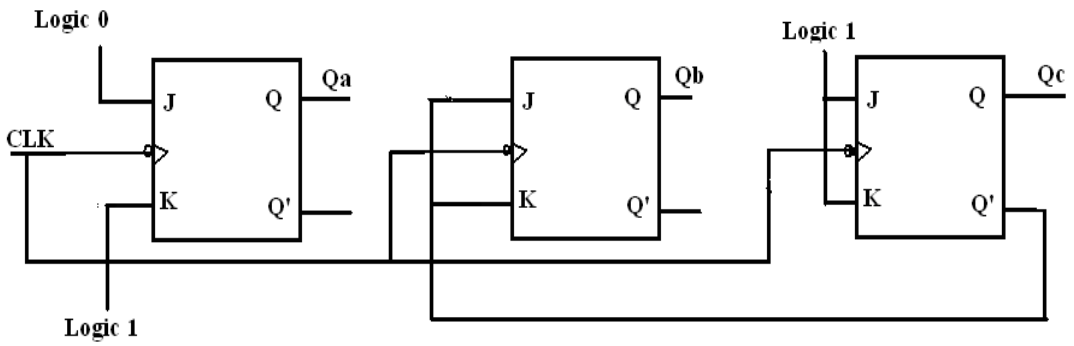
Jc = 1 ; Kc = 1 ; Ja = 0 ; Ka = 1

Q_c' \ $Q_b'Q_a'$	$Q_b'Q_a$	Q_bQ_a	Q_bQ_a'
1	x	x	x
0			x

Jb = Q_c'

Q_c' \ $Q_b'Q_a'$	$Q_b'Q_a$	Q_bQ_a	Q_bQ_a'
x	x	x	1
x			0

Kb = Q_c'



Logic diagram

Example 3 : Design a counter which has 4, 0 ,6, 2 as a count sequence using D F.F.

Present state			Next state			F.F input		
Qc	Qb	Qa	Qc	Qb	Qa	Dc	Db	Da
1	0	0	0	0	0	0	0	0
0	0	0	1	1	0	1	1	0
1	1	0	0	1	0	0	1	0
0	1	0	1	0	0	1	0	0

From the state transition table

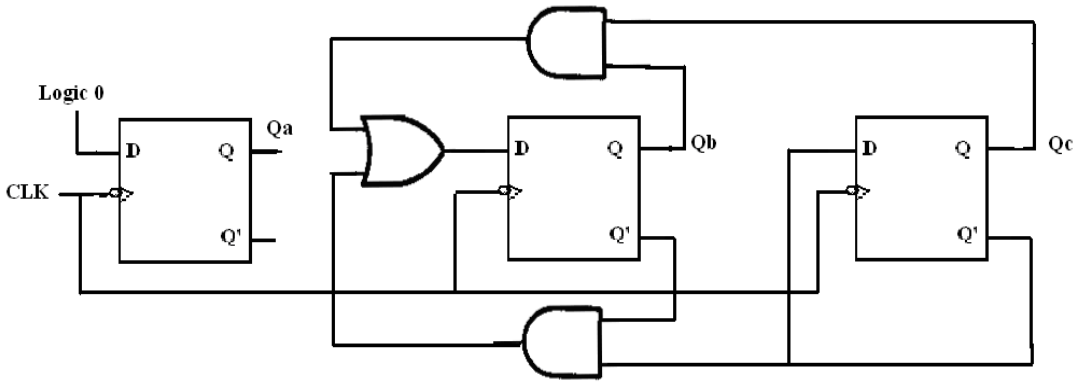
Da = 0

	$Q_b'Q_a'$	$Q_b'Q_a$	Q_bQ_a	Q_bQ_a'
Q_c'	1	X	X	1
Q_c	0			0

$D_c = Q_c'$

	$Q_b'Q_a'$	$Q_b'Q_a$	Q_bQ_a	Q_bQ_a'
Q_c'	1	X		0
Q_c	0		X	1

$D_b = Q_c'.Q_b' + Q_c.Q_b$



Registers

The Register is a collection of flip-flops, basic function is to store a binary word, other is modify the stored word by shifting it's bits LEFT or RIGHT.

The ***shift register*** is a register that moves information on the clock signal left or right.

a) Shift Right Register.

The circuit diagram shown below used to move the stored bits right where the output of the first flip-flop is connected to the next F/F and so on.

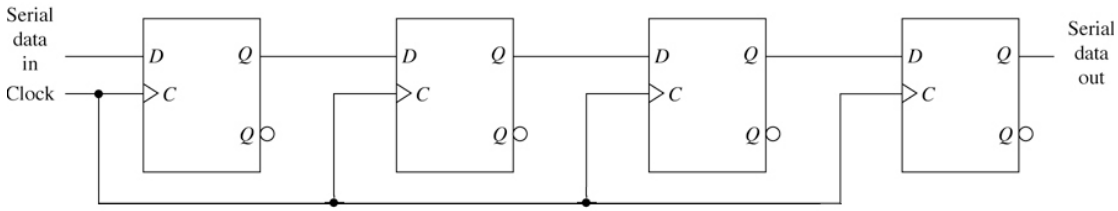


Figure: Shift Right Register

b) Shift Left Register.

The circuit diagram shown below used to move the stored bits right where the output of the last F/F is connected to the previous F/F and so on.

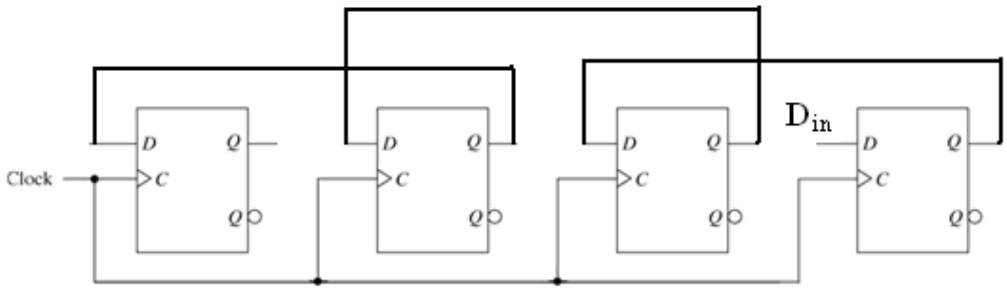
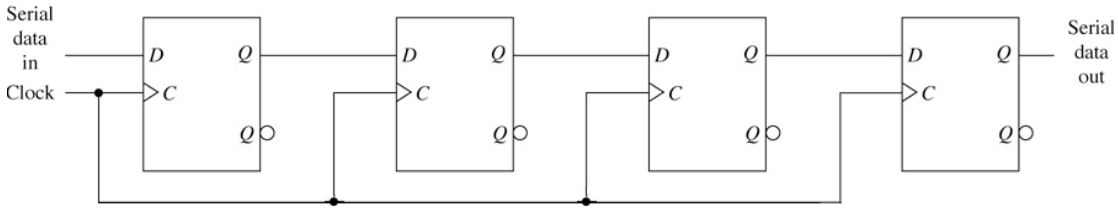


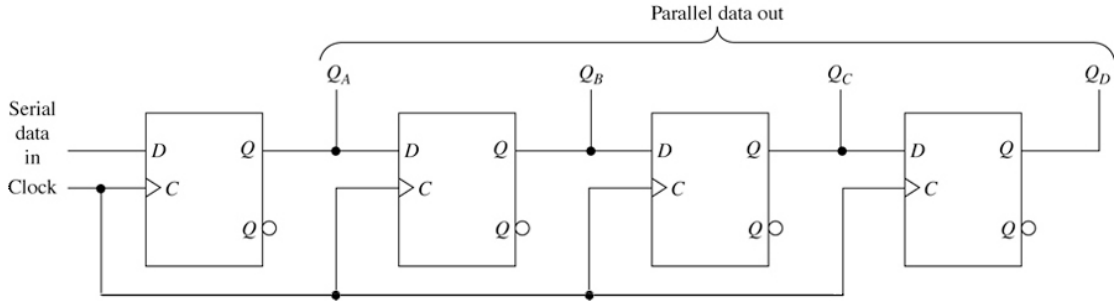
Figure: Shift Left Register

Shift Registers can be classified by the way data is entered for storage, and by the way data is outputted from the register as below.

- **Serial-in / Serial-out (SISO):** the input to the first stage and the output from the last stage. Which use to data delay.



- **Serial-in / parallel-out (SIPO)**: The input to the first stage and the output from the all stages. Which is used to convert the serial data to parallel



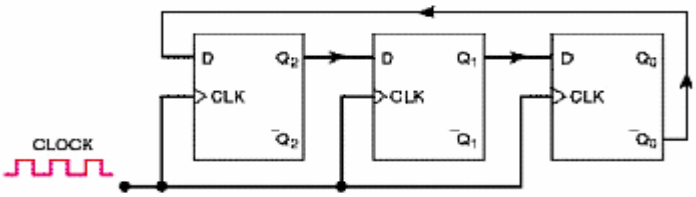
- **parallel-in / serial-out (PISO)**: the input to all stages and the output from the last stage. Which is used to convert the parallel data to serial.

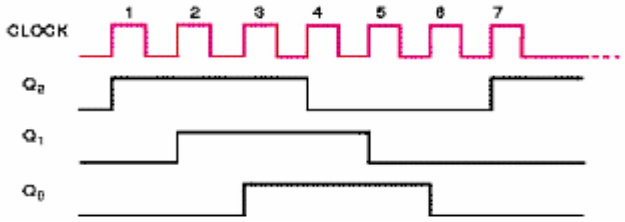
- **parallel-in / parallel-out (PIPO)**: the input to all stages and the output from all stages . which is used to store binary data.

Register–Applications:• temporally store bits• Shifting bits for calculation, serial comm.–Shift register, serial adder.

Johnson Counter

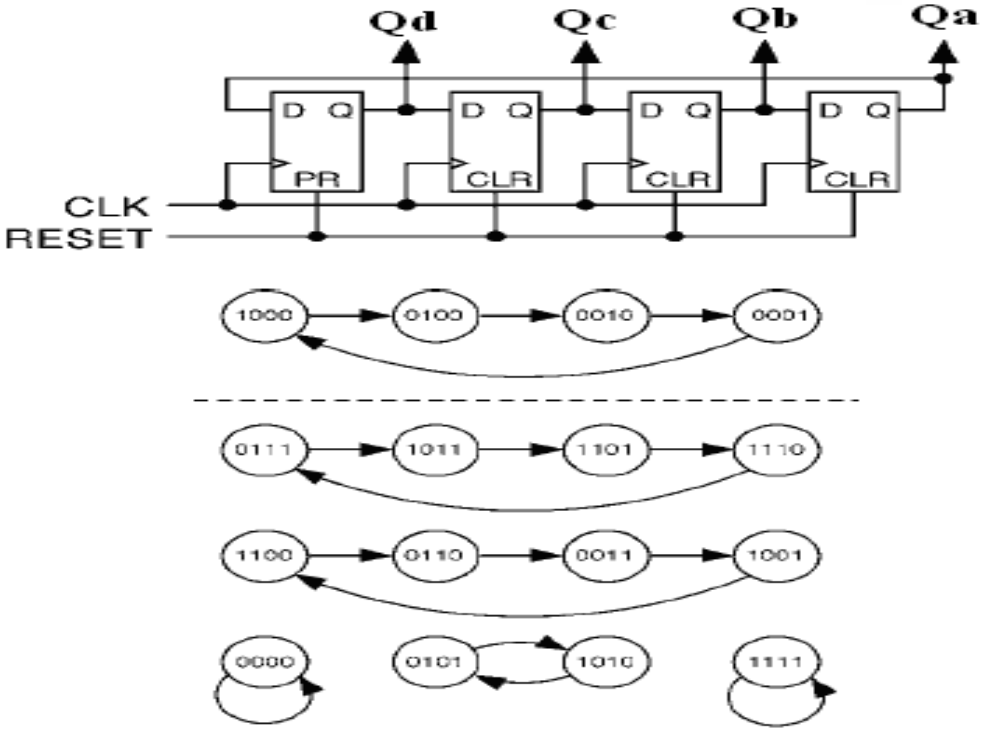
A Johnson counter can be considering of as a special type of shift register. In Johnson counter, the last bit of the shift register, inverted and fed back into the first bit, as shown in Figure below , Only a single bit changes during a clock transition, This structure and counting sequence allows an N-bit Johnson counter to count up to 2n possible states as opposed to the 2^n possible states allowed in a binary counter





Ring Counter

a circular shift register (with *k* flipflops) that at any time, only one flip-flop is set (having value 1) and all others are cleared (with value 0). It is used to generate *k* (periodic) timing signals.



Frequency division

One application of counter is use for frequency division, by which it can be get a low frequency from a high frequency.

- The output frequency of each FF = the clock frequency of input / 2.
- The output frequency of the counter = the clock frequency / MOD.

If we are connected a three counter in cascaded:

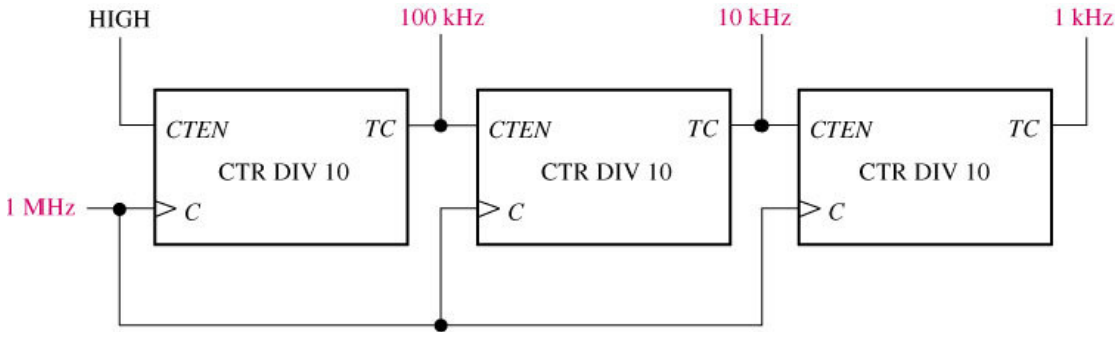
Total MOD = MOD1 * MOD2 * MOD3

Example : Using a MOD10 asynchronous counter to get an output frequency of (1 KHz) from the input frequency (1 MHz)

Sol

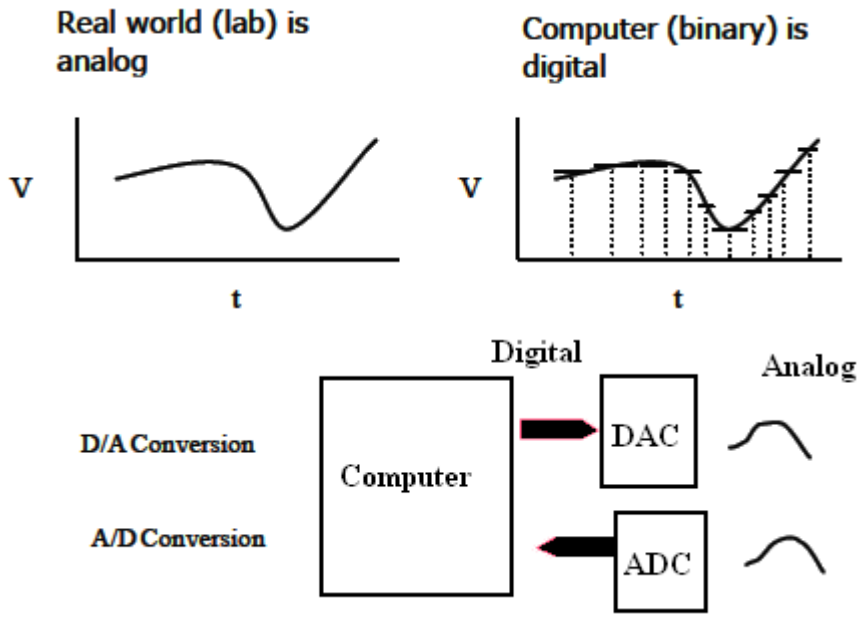
Total MOD = input frequency / output frequency
= $1 * 10^6 / 1*10^3$
= 1000

No. of counter used = 3



Interfacing between the Computer & Real World

It is clear that the always quantities in real world are an analog signal which it can take any value from zero to infinity with continuous time , and this represent the big challenge to any computer because they are operate with two voltage values only (digital). Therefore, that it must be a technique to convert between these two types of signals. This technique is a **digital to analog converter (DAC)** and an **analog to digital converter (ADC)**

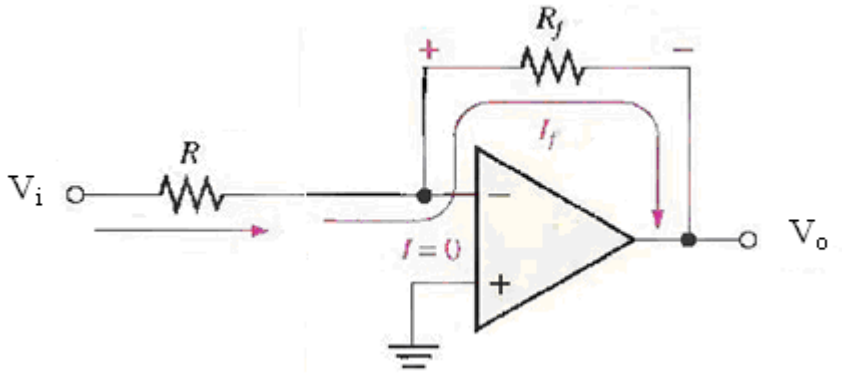


A) Digital to Analog Converter

The digital to analog converter known as the D/A converter (read D-to-A converter or DAC) is a major interface circuit that forms the bridge between the analog and digital worlds.

The DAC is an electronic component that converts digital logic levels into an analog voltage. The output of a DAC is just the sum of all the input bits weighted in a particular manner. The basic elements of a DAC are the resistors and the operational amplifier (OP- AMP) . an OP-AMP is a linear amplifier that has two inputs (inverting & non inverting) and one output. It has a very high voltage gain and a very high input impedance and a very low output impedance as shown in figure below

$V_o = - V_i * R_f / R$



Digital-to-Analog Conversion methods

The two basic types of a digital to analog converter are:

1-Binary-Weighted- Resistance Digital-to-Analog Converter

This method uses a resistor network with resistance values that represent the binary weights of the input bits of the digital code. Fig4-20 shows a 4-bit DAC. Each of the input resistors will either have current or have no current, depending on the input voltage level. If the input voltage is zero (binary 0), the current is also zero. If the input voltage is HIGH (binary 1), the amount of current depends on the input resistor value and is different for each input resistor, as indicated in the figure.

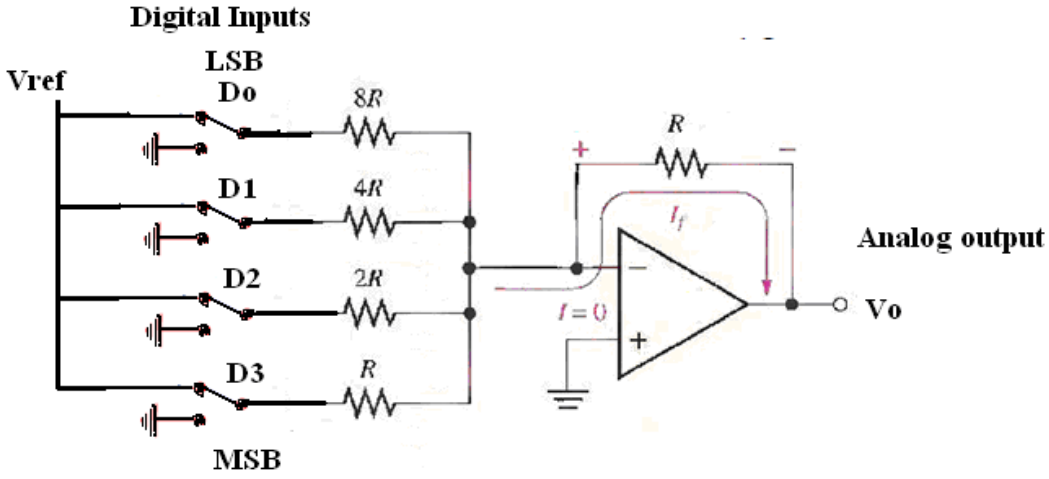


Figure : a four bit weighted resistance DAC

The digital input as a control, when the bit (D) is logic 1 it will connect a Vref and when it is logic 0 it will connect a ground (0 volt)

The output voltage equation for N bits digital input is :

$Vo = (1/2)* V_{ref} (D_{N-1} + D_{N-2} / 2 + D_{N-3} / 2^2 + \ldots\ldots\ldots\ldots + D_0 / 2^{N-1})$

Where: V_{ref} reference voltage.

Example : Calculate the analog output voltage of a weighted resistor DAC if the digital input is (1101) and the reference voltage is (10 v).

Sol:

With four bits N= 4

$Vo = (1/2)V_{ref} (D_3 + D_2 / 2 + D_1 / 4 + D_0 / 8)$

= 5 (1 + 1/2 + 0/4 + 1/8)
= 5 *13/8
= 8.125 v

2-The R/2R ladder Digital-to-Analog Converter

The R/2R ladder is overcomes one of the problems in the binary-weighted-input DAC in that it requires only two resistor values, as shown in figure below for 4 bits.

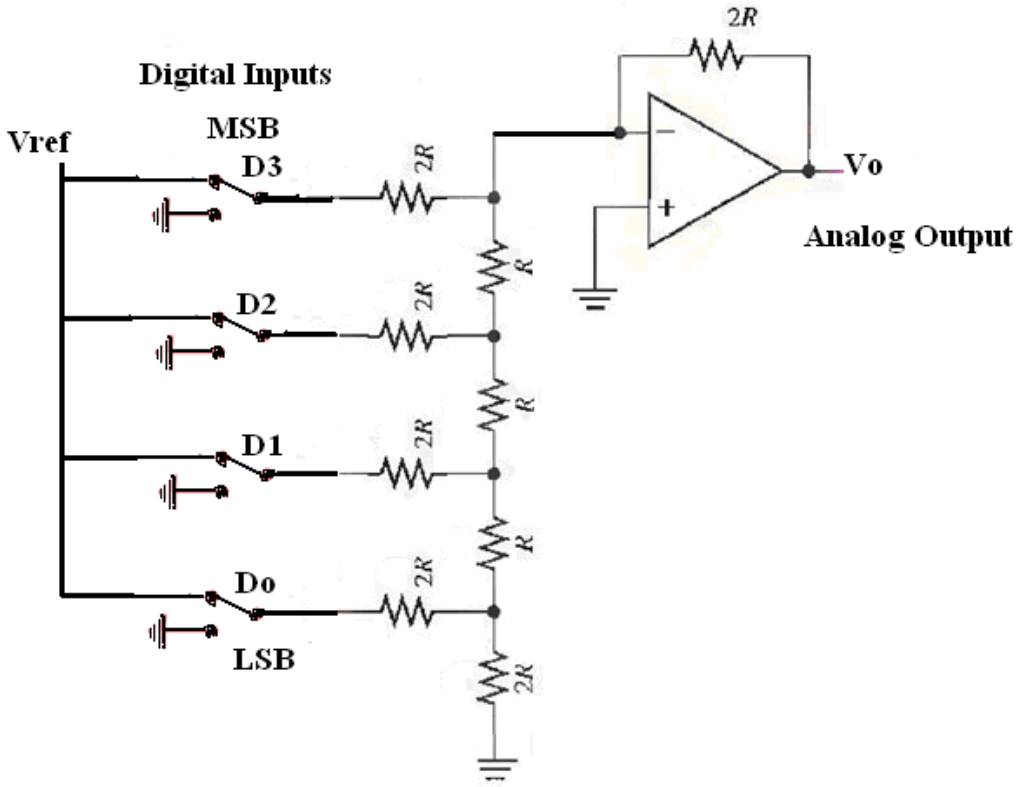


Figure: A Four bit R /2R Ladder DAC

The equation of the R/2R Ladder DAC is:

$$Vo = V_{ref} (D_0 * 2^0 + D_1 * 2^1 + D_2 * 2^2 + D_3 * 2^3 + ... D_{N-1} * 2^{N-1})/2^N$$

Where D is the logic level

Example : Find the analog output voltage from a six bits ladder which has a digital input (110010) assume that Vref =20 v.

Sol:

$Vo = 20* (0 * 2^0 + 1 * 2^1 + 0* 2^2 + 0 * 2^3 + 1 * 2^4 + 1 * 2^5)/2^6$

$Vo = 20* (0 + 2 + 0 + 0 + 16 + 32)/64$

$Vo = 1000 / 64$

$Vo = 15.625$ v

B) Analog to Digital Converter (ADC)

The purpose of an ADC is to produce a digital binary number that is proportional to an analog input signal . there are many types of ADC

1- Counter Type
2. Integrating or Dual Slope
3. Parallel or Flash Counter Type
4. Successive Approximation

1- Counter Type ADC

This type consist of a counter , analog comparator , and DAC When the output of a digital comparator is (Logic 1) the counter is count up until the output of the analog comparator is (Logic 0) the counter will stop and the binary output of this counter is the digital value corresponding to the analog input as shown in figure below:

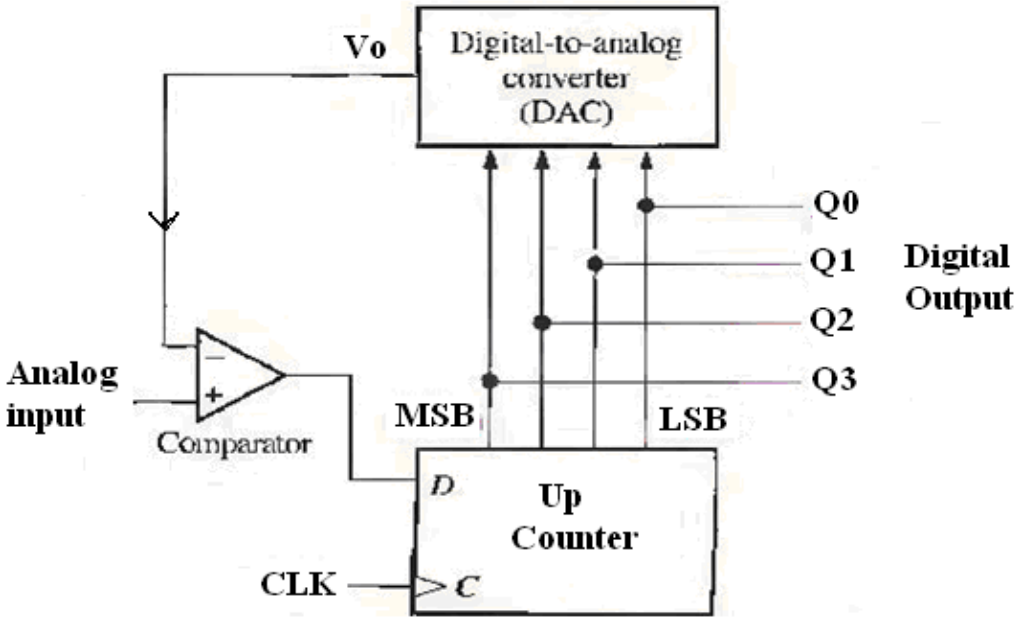


Figure: A 4 Bits Counter Type ADC

4-Successive-Approximation Analog-to-Digital Converter:

One of the most widely used methods of analog-to-digital conversion is successive- approximation. It has a much faster conversion time than the dual-slope conversion, but it is slower than the flash method. It also has a fixed conversion time that is the same for any value of the analog input. Figure below shows a basic block diagram of a 4-bit successive approximation ADC. It consists of a DAC, a successive-approximation register (SAR), and a comparator. The basic operation is as follows:

The input bits of the DAC are enabled (made equal to a 1) one at a time, starting with the most significant bit (MSB). As each bit is enabled, the comparator produces an output that indicates whether the input signal voltage is greater or less than the output of the DAC. If the DAC output is greater than the input signal, the comparator's output is LOW, causing the bit in the register to reset. If the output is less than the input signal, the 1 bit is retained in the register. The system does this with the MSB first, then the next most significant bit, then the next, and so on. After all the bits of the DAC have been tried, the conversion cycle is complete, example of a 4-bit conversion.

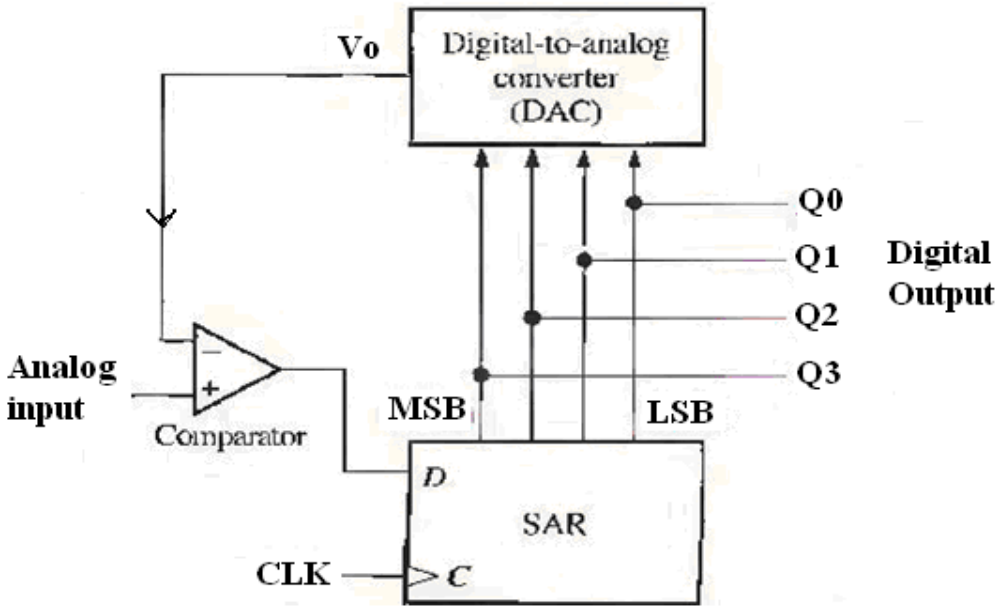


Figure: A 4 Bits Successive-approximation ADC

Multivibrators

Pulse generators are circuits that generate a pulse waveform directly and most of them use the relaxation principle (charging & discharging of capacitor). The most common type is the multivibrator, which consists of two stages, resistance coupled amplifier with the output of each stage coupled resistively to the other. In generation collector current of one stage is a maximum when Ic of the other stage is at cut-off at regular intervals. Multivibrators are classified into three types:

(A) Astable multivibrator

It is a multivibrator in which stages are switched from ON to OFF state in regular time intervals without any triggering. It is also called (free running) and has no stable state. The circuit diagram is shown in Figure below.

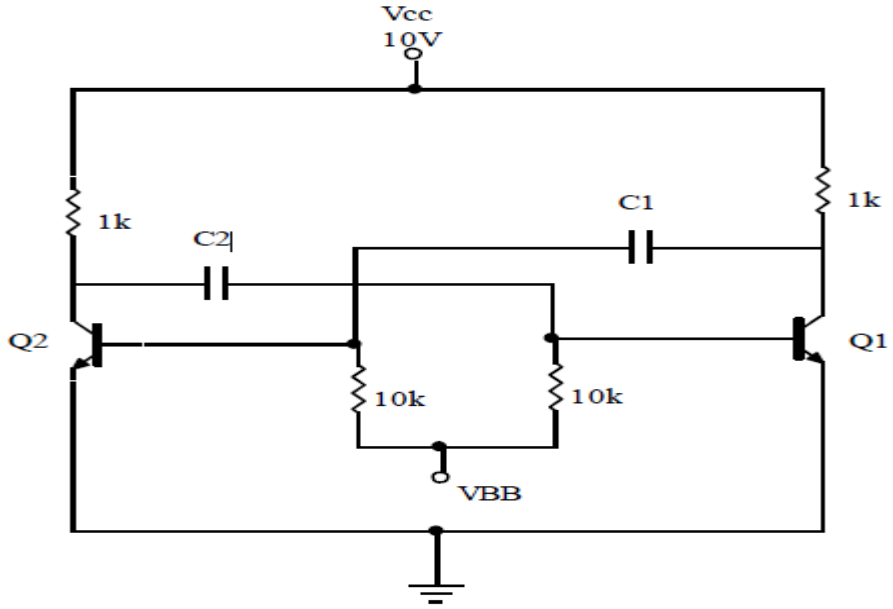


Figure : The Astable Multivibrator

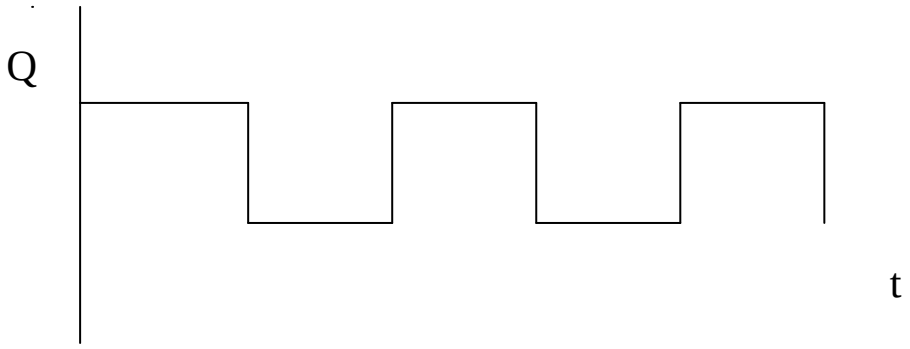


Figure: The Astable Timing Diagram

(B) Monostable Multivibrator

In this case, one stage is ON until it is triggered, in the same time the other stage is made to be ON for a predetermined length of time, and then switched back to its original state automatically. The circuit diagram is shown in Figure below.

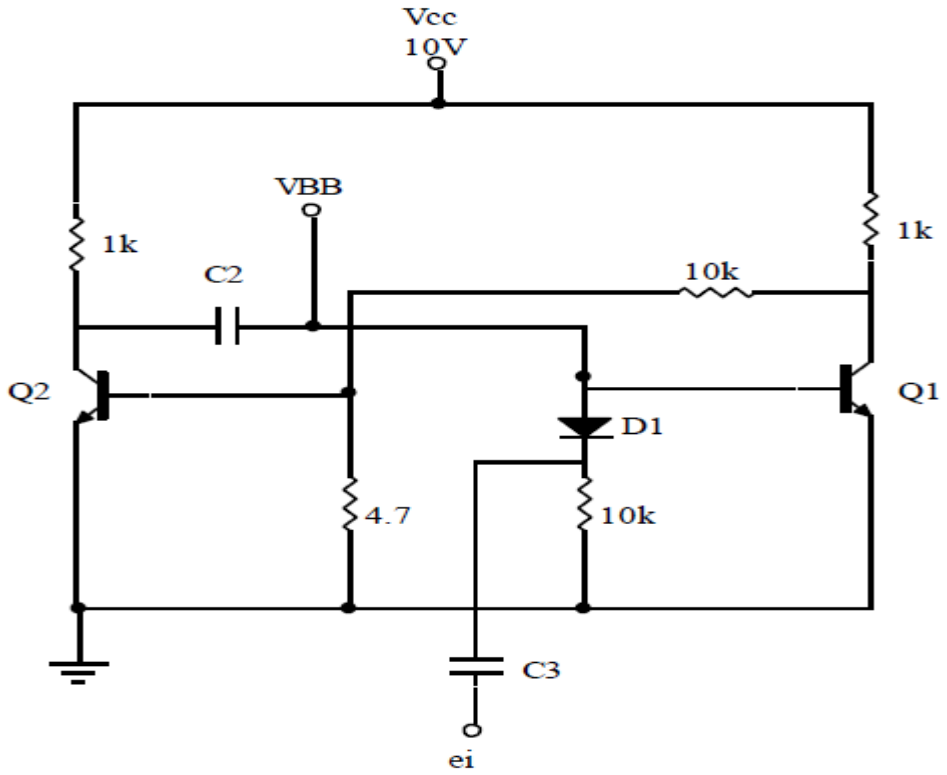


Figure :The Monostable Multivibrator

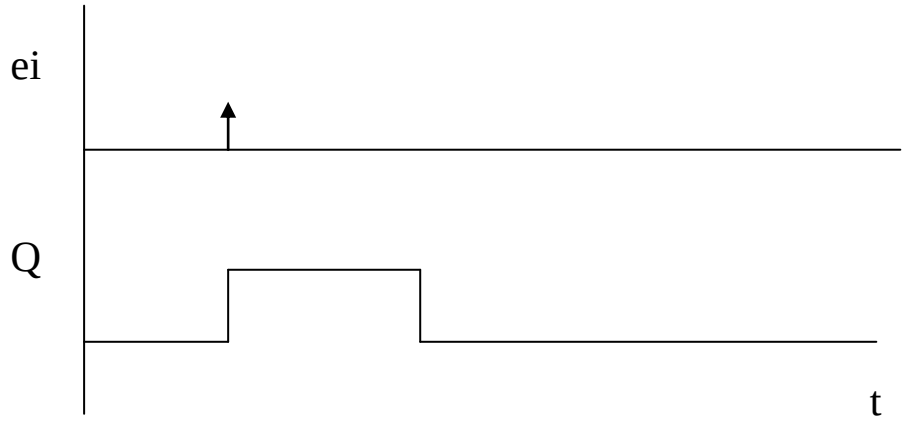


Figure . the Monostable Timing diagram

C) Bistable multivibrator

In this case, one stage remains stable in one state (ON. or Off) until a triggering pulse is applied to initiate the switching action to reverse the stability condition (it has two stable state), the circuit diagram is shown in Figure below.

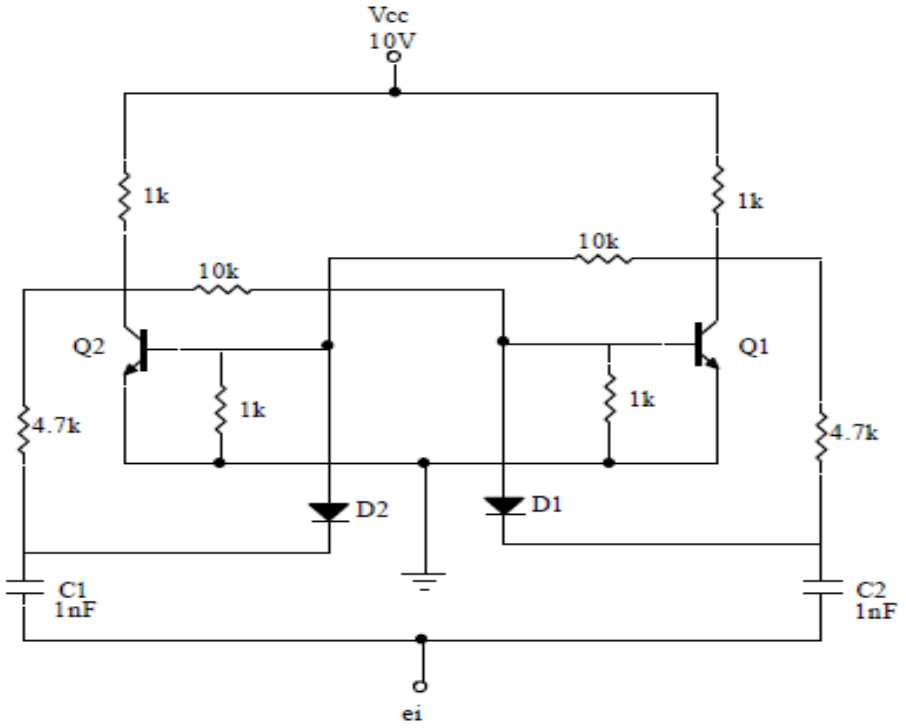


Figure: The Bistable Multivibrator

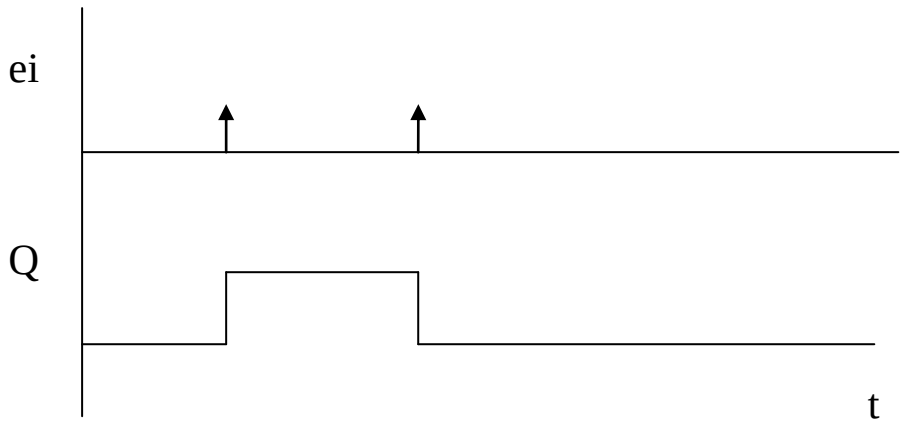


Figure . the Bistable Timing diagram