srivenkateshwaraa
College of Engineering & Technology
(Approved by AICTE, New Delhi & Affiliated to Pondicherry University, Puducherry)
13-A, Pondy - Villupuram Main Road, Ariyur, Puducherry - 605 102.
ASPIRE TO EXCEL

ISO 21001

DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING

Subject Name: **PULSE AND DIGITAL CIRCUITS**

Subject Code: **EE T45**

UNIT – I LINEAR WAVE SHAPPING CIRCUITS

Linear wave shaping circuits: RC, RL and RLC circuits – Pulse transformer – Steady state switching characteristics of devices– Clipping and clamping circuits–Switching circuits.

1. DISCUSS ABOUT WORKING OF CLIPPER CIRCUIT IN DETAIL. [NOV-2016]

Definition:

1. The circuit with which the waveform is shaped by removing (or clipping) a portion of the input signal without distorting the remaining part of the alternating waveform is called a ***clipper***.

2. Clipping circuits are also referred to as voltage (or current) limiters, amplitude selector or slicers.

3. The four general categories of clippers are

    1. Positive clipper

        1. Series Positive clipper

        2. Shunt Positive clipper

    2. Negative clipper

        1. Series negative clipper

        2. Shunt negative clipper

    3. Biased clipper

    4. Combinational clipper

Positive clipper:

Series Positive clipper:

4. In the series positive clipper when the input voltage is positive the diode does not conduct and acts as an open circuit and hence the positive half cycle does not appear at the output. i.e the positive half cycle is clipped off.

5. When the input signal is negative, the diode conducts and acts as a closed switch (short circuit), the negative half appears at the output as shown in figure below

Shunt Positive clipper:

6. When the input voltage is positive the diode conducts and acts as a short circuit and hence there is zero signal at the output i.e the positive half cycle is clipped off.

7. When the input signal is negative the diode does not conduct and acts as on open switch, the negative half cycle appears at the output as shown below

Negative clipper:

8. In the negative clipping circuit, the diode is connected in a direction opposite to that of a positive clipper.

Series negative clipper

9. In the series negative clipper, during the positive half cycle of the input signal, the diode conducts and acts as a short circuit and hence the positive half cycle of the input signal will appear at the output as shown in figure.

10. During the negative half cycle of the input signal the diode does not conduct and acts as on open circuit.

11. The negative half cycle will not appear at the output i.e the negative half cycle is clipped off.

Shunt negative clipper

12. When the input signal is positive the diode does not conduct and acts as on open switch, the positive half cycle appears at the output as shown below

13. When the input voltage is negative the diode conducts and acts as a short circuit and hence there is zero signal at the output i.e the positive half cycle is clipped off.

Biased clipper:

Biased positive series clipper

14. Figure shows the circuits of shunt and series type positive clipping along with input and output waveforms.

15. In the biased series positive clipper the diode does not conduct as long as the input voltage is greater than +V_R and hence the output remains at +V_R .

16. When the input voltage becomes less than +V_R the diode conducts and acts as a short circuit.

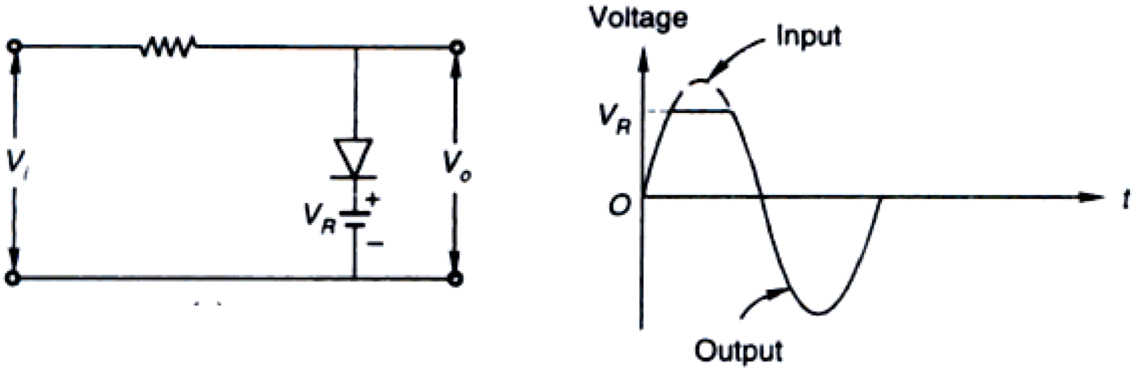
17. Hence all the input signals having less than +V_R as well as negative half cycle of the input wave will appear at the output

Biased positive shunt clipper

18. In the biased shunt positive clipper as shown in figure the diode conducts as long as the input voltage is greater than +V_R and hence the output remains at +V_R .

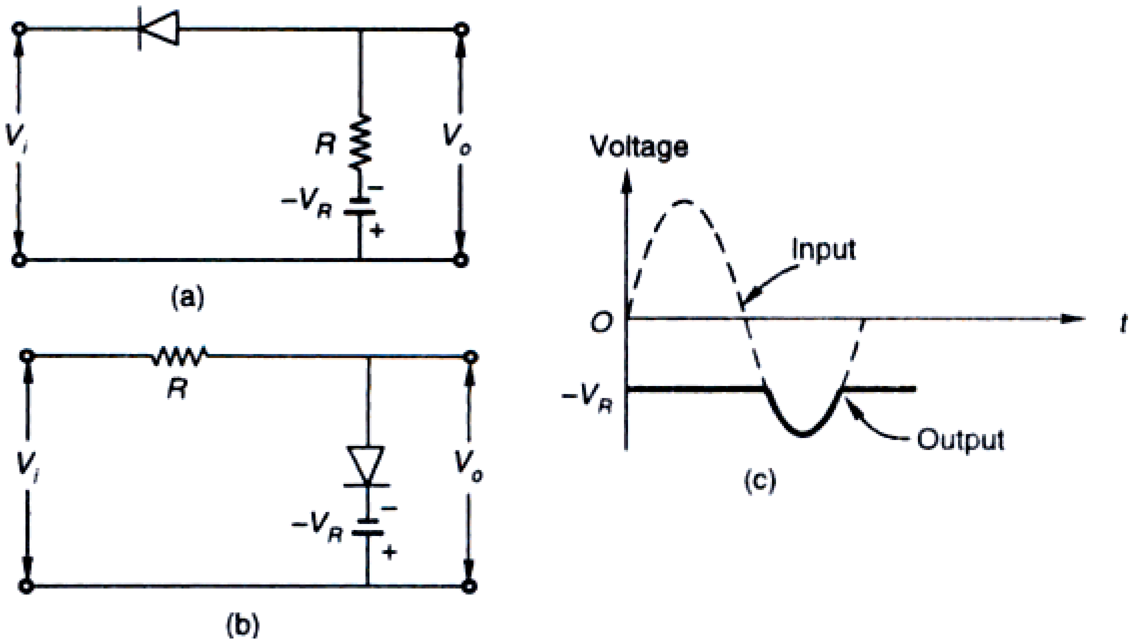
19. When the input voltage becomes less than +V_R the diode does not conducts and acts as a open circuit.

20. Hence all the input signals having less than +V_R as well as negative half cycle of the input wave will appear at the output



21. The clipping levels can be shifted up or down by varying the bias voltage +V_R

Biased positive clipper with reverse polarity of the battery V_R



Biased positive clipper with reverse polarity of V_R. (a) Series, (b) Shunt, and (c) Input and output signal waveforms

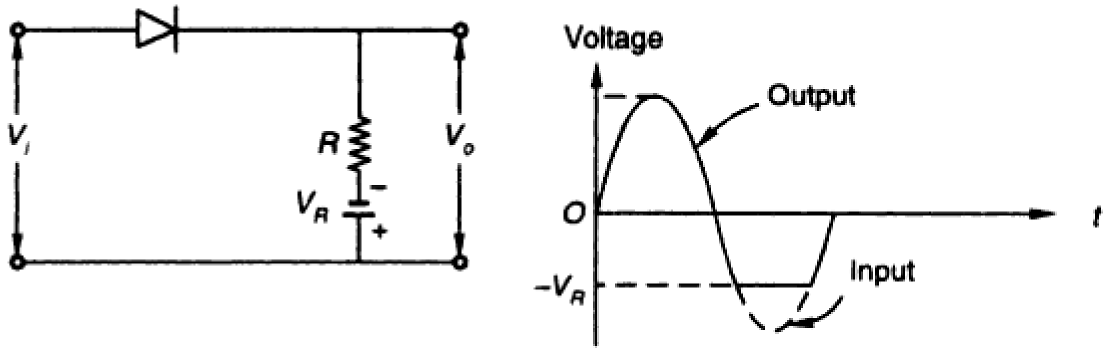
22. Above figure shows the biased series and shunt clippers with reverse polarity of $+V_R$ along with the input and output voltage waveforms.

23. Here the entire signal above $-+V_R$ is clipped off

Biased clipper:

Biased negative series clipper

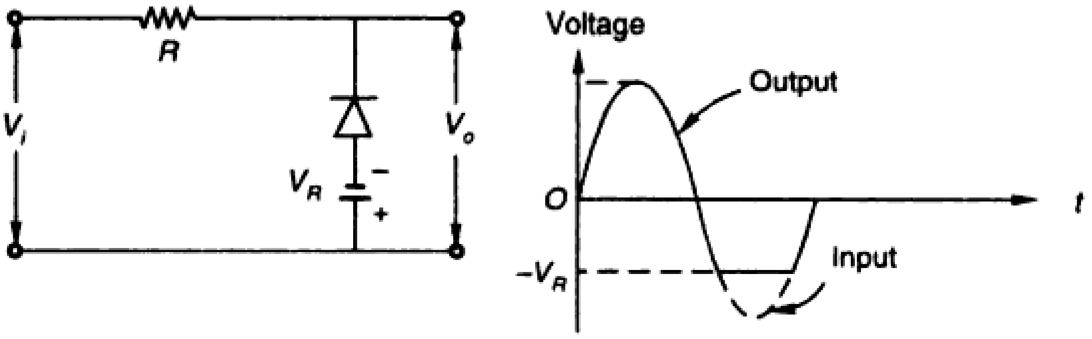
24. In the biased series negative clipper when the input voltage $V_i \leq -V_R$ the diode does not conduct and clipping takes place.



Biased negative shunt clipper

25. In the biased shunt negative clipper when the input voltage $V_i \leq -V_R$ the diode conducts and clipping takes place.

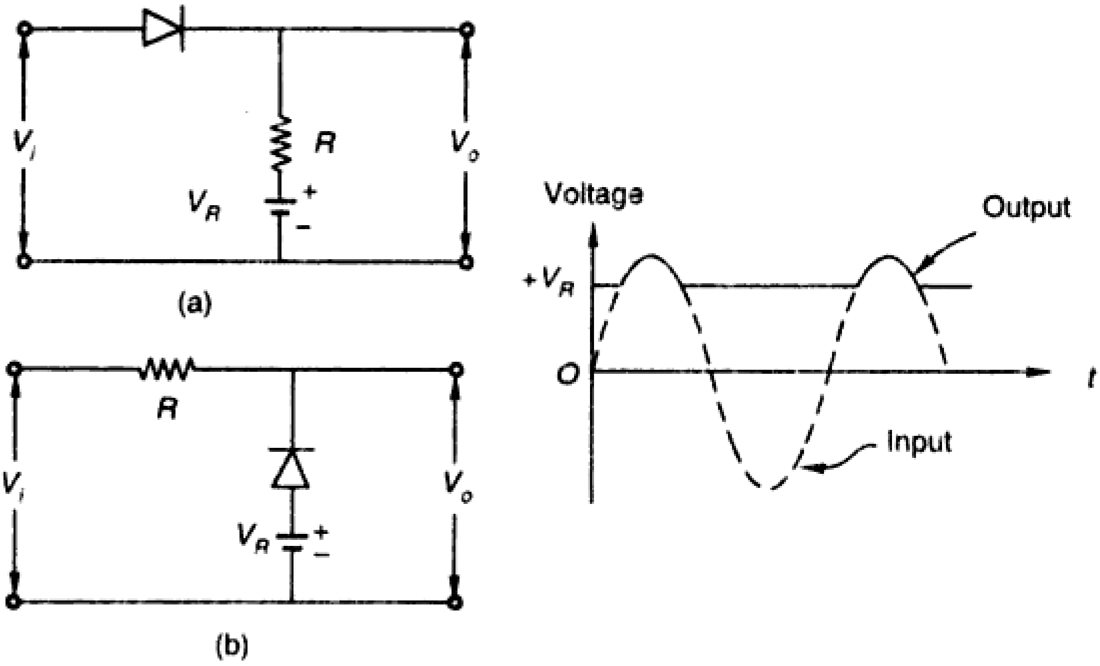
26. The clipping level can be shifted up and down by varying the bias voltage ($-V_R$)



Biased negative clipper with reverse polarity of the battery V_R

27. Above figure shows the biased series and shunt clippers with reverse polarity of $+V_R$ along with the input and output voltage waveforms.

28. Here the entire signal above $-+V_R$ is clipped off



Biased negative shunt clipper

29. This is the combination of a biased positive clipper and a biased negative clipper.

30. When the input signal voltage $V_i \geq -V_{R1}$ diode D1conducts and acts as a closed switch, while diode D2 is reverse biased and D2 acts as open switch.

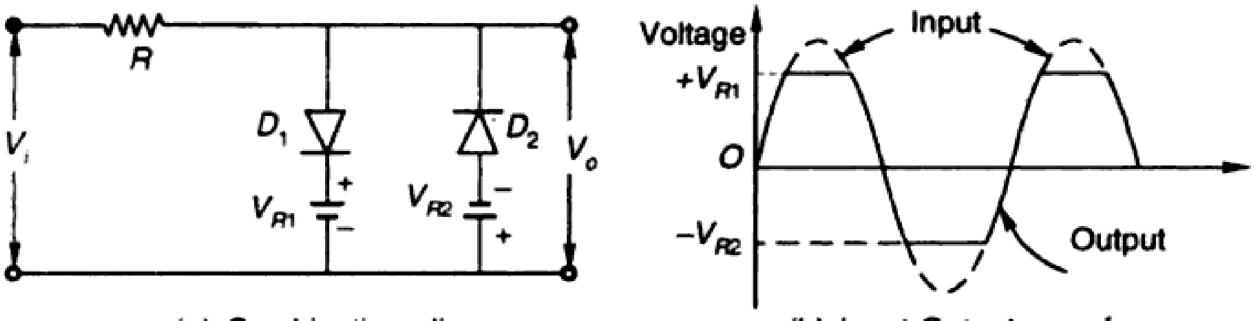
31. Hence the output voltage cannot exceed the voltage level of $+V_{R1}$ during the positive half cycle.

32. Similarly when the input signal voltage $V_i \leq -V_{R2}$ the diode D2 conducts and acts as closed switch while diode D1 is reverse biased and diode D1 acts as an open switch.

33. Hence the output voltage VO cannot go below the voltage level of –V_{R2} during the negative half cycle.

34. It is evident that the clipping levels may be changed by varying the values of V_{R1} and V_{R2}.

35. If V_{R1} = V_{R2} the circuit will clip both the positive and negative half cycles at the same viltage level and hence such a combination clipper is called ***symmetric clipper.***



(a) Combination clipper (b) Input-Output waveform

Applications of clippers:

1. Radars

2. Digital computers

3. Radio and television receivers etc

4. DISCUSS ABOUT WORKING OF CLAMPERS CIRCUIT IN DETAIL.

Definition:

5. Clamping network shifts (clamps) a signal to a different DC level i.e it introduces a DC level to an AC signal.

6. Hence the clamping network is also known as ***DC restorer.***

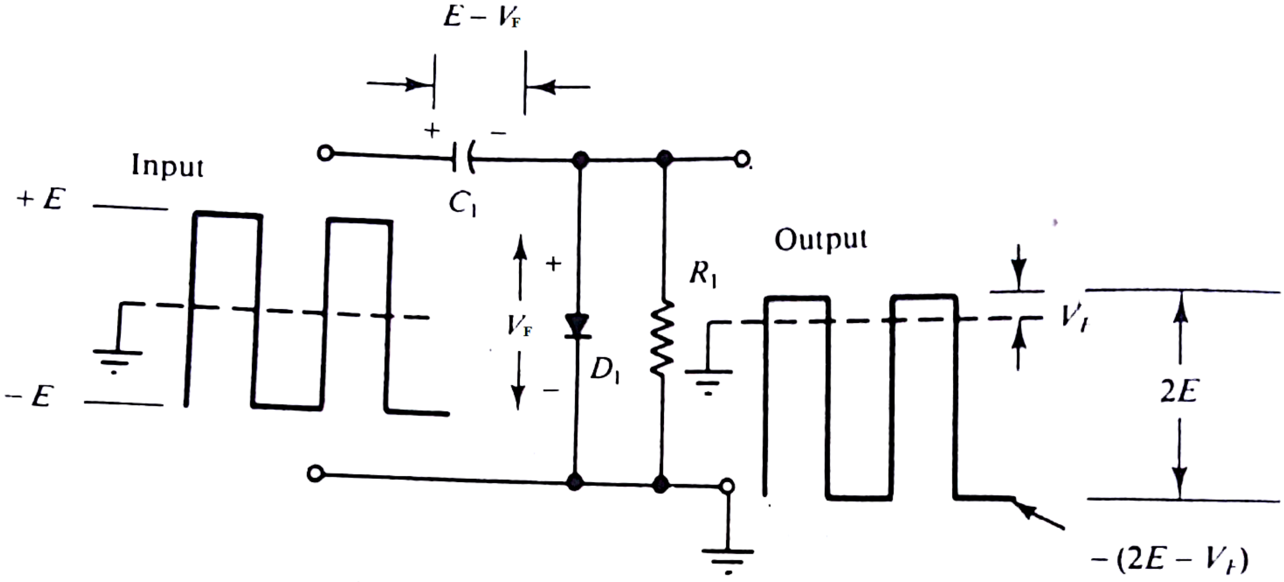
7. It is of two types

   1. Negative voltage clamper

   2. Positive voltage clamper

Negative voltage clamper

8. When the input is positive diode D1 is forward biased and capacitor C1 charges with the polarity as shown in figure below.



(a) Negative voltage clamper

9. During the positive input peak the output cannot exceed the diode forward bias voltage V_F.

10. At this time therefore the voltage on the right side of the capacitor is V_F, while on the left side of the capacitor the voltage is +E.

11. Thus the capacitor is charge to E-V_F, positive on the left and negative on the right as shown in above figure.

12. When the input switches to negative the diode is reverse biased and it has no further effect on the capacitor voltage.

13. R_1 has a high resistance value and cannot discharge the capacitor significantly during the negative portion of the input waveform.

14. While the input is negative the output voltage is the sum of the input voltage and capacitor voltage.

15. Since the polarity of the capacitor voltage is the same as the (negative) input, the result is a negative output larger than the input voltage. Thus

Negative output = $-E-(E-V_F) = -(2E - V_F)$

16. The peak to peak output is the difference between the negative and positive peak voltages

peak to peak output = (positive peak) – (negative peak)

$= V_F - (2E - V_F)$

$= 2E$

17. It is seen that the amplitude of the output waveform from the negative voltage clamper is exactly the same as that of the input.

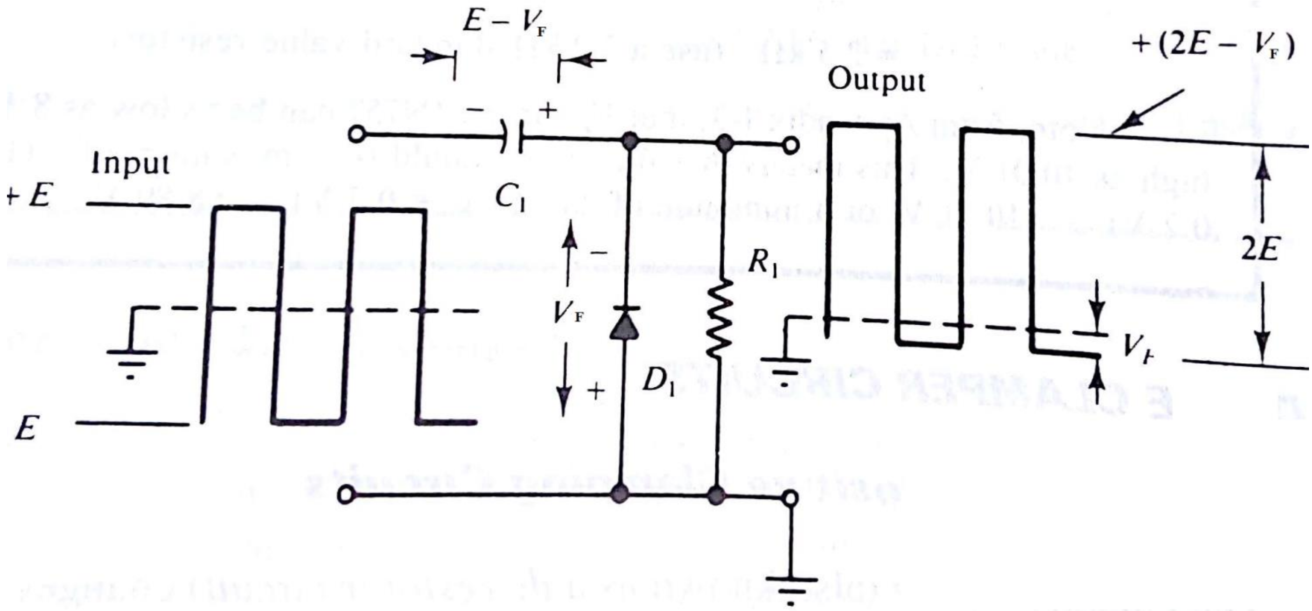
18. The function of R_1 is to discharge C1 over several cycles of the input waveform.

Negative voltage clamper

19. It functions exactly the same way as the negative voltage clamper. The diode connected as shown clamps the negative output peak at $–V_F$.

20. Capacitor C1 charges to $E-V_F$ positive on the right and negative on the left.

21. The positive output then becomes $2E - V_F$



Applications:

22. These circuits find application in television receivers to restore the DC reference signal to the video signal.

23. EXPLAIN IN DETAIL ABOUT STEADY STATE SWITCHING CHARACTERISITCS OF PN DIODE.

1. Diodes are often used in a switching mode.
2. When the applied bias voltage to the PN diode is suddenly reversed in the opposite direction, the diode response reaches a steady state after an interval of time called ***recovery time.***
3. The ***forward recovery time*** **(T_{fr})** is defined as the time required for forward voltage or current to reach a value after switching diode from its reverse to forward biased state.
4. The ***reverse recovery time*** **(T_{rr})** is defined as the time required for reverse voltage or current to reach a value after switching diode from its forward to reverse biased state.
5. Most diodes switch very quickly into the forward biased condition; however there is a longer turnoff time owing to the junction diffusion in reverse biased condition.
6. When the PN junction diode is forward biased, the minority electron concentration in the P region is approximately linear.

7. If the junction is suddenly reverse biased at t_1 , then because of stored electronic charge, the reverse current I_R is initially of the same magnitude as the forward current I_F.

8. When the pulse switches from positive to negative, the diode conducts in reverse instead of switching off sharply.

9. The reverse current I_R initially equals the forward current I_F , then it gradually decreases towards zero.

10. The high level of reverse of reverse current occurs because at the instant of reverse bias there are charge carriers crossing the junction depletion region, and these must be removed.

11. The typical values of **reverse recovery time T_{rr}** for switching diode ranges from **4ns to 50ns.**

12. After the instant t = t_2 the diode gradually recovers and ultimately reaches the steady state.

13. During the interval from t_1 to t_2 , the injected minority carrier have remained stored & hence this time interval is called ***storage time t_s***

14. The time interval between t_2 & t_3 when the diode has recovered nominally is called ***transition time t_t.***

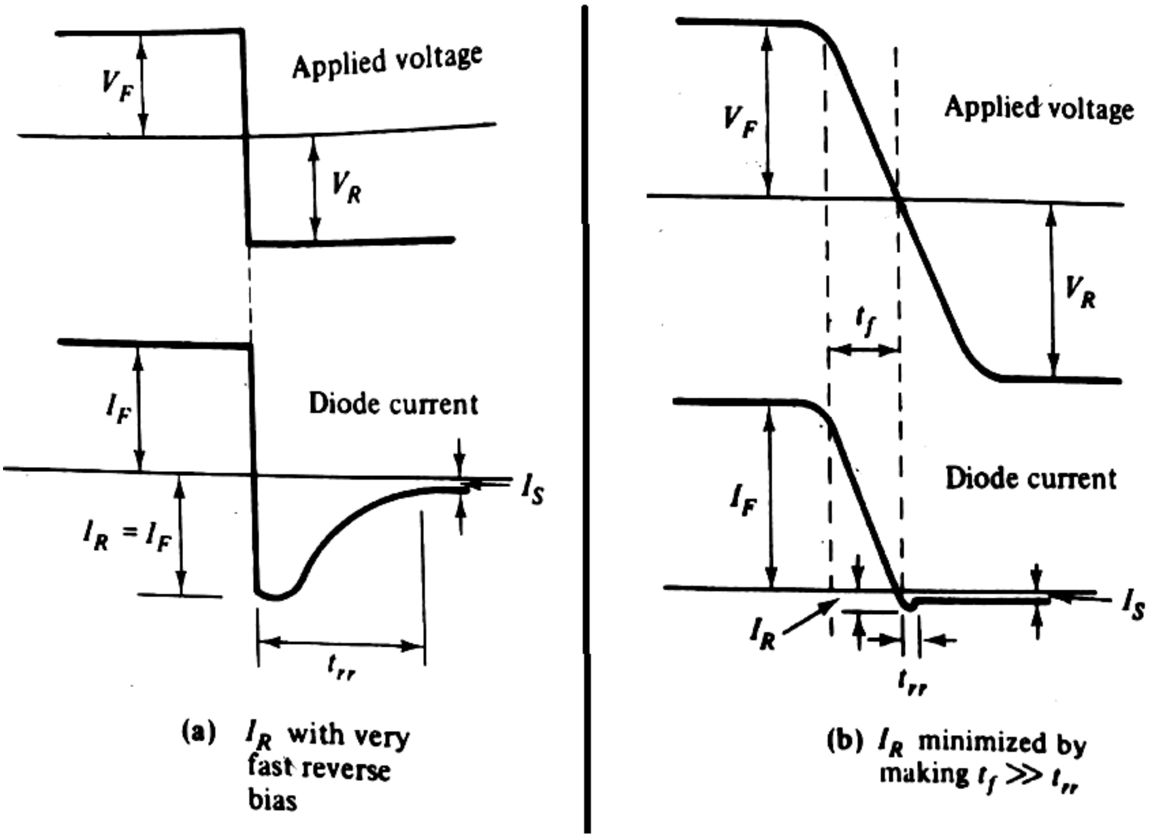
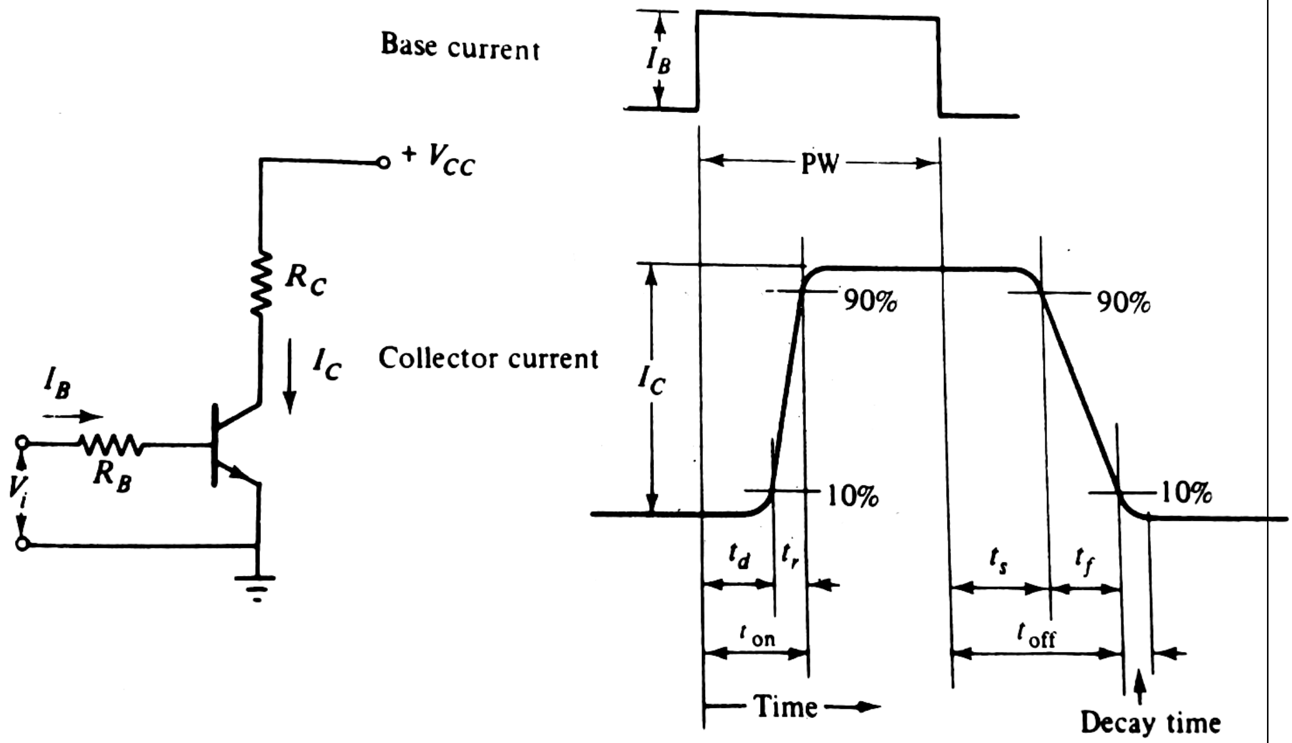


Figure: Switching characteristics of PN junction diode

15. **EXPLAIN IN DETAIL ABOUT STEADY STATE SWITCHING CHARACTERISITCS OF TRANSISTOR.**

1. Transistor can be used as a witch. One important characteristics of a switching transistor is the speed with which it can be switched ON and OFF.

2. Consider the figure shown below where the time relationship between collector current and base current is shown.



Time relationship between base current and collector current in a transistor switch

1. When the input current I_B is applied, the transistor does not switch ON immediately.

2. The time between application of base current and commencement of collector current is termed ***delay time*** $\boldsymbol{t_d}$

3. The ***delay time*** $\boldsymbol{t_d}$ is defined as the time required for I_C to reach 10% of its final level after I_B has commenced.

4. Even when the transistor begins to switch on a finite time elapses before I_C reaches its maximum level.

5. The ***rise time*** $\boldsymbol{t_r}$ is defined as time it takes for I_C to go from 10% to 90% of its maximum level.

6. The ***turn on time*** $\boldsymbol{t_{on}}$ for the transistor is the sum of t_d and t_r.

7. Similarly a transistor cannot be switched off instantaneously.

8. The ***turn off time*** $\boldsymbol{t_{off}}$ is composed of a ***storage time*** $\boldsymbol{t_s}$ and a ***fall time*** $\boldsymbol{t_f}$. The storage time results from the fact that the collector base junction is forward biased when the transistor is in saturation.

9. Charge carriers crossing a forward junction are trapped in the depletion region when the junction is reversed.

10. These charge carrier must be withdrawn or made to recombine with charge carriers of an opposite type before the collector current begins to fall.

11. The ***storage time t_s*** is defined as the time between I_B switch off and I_C falling to 90% of its maximum level.

12. The fall time tf is the time required for I_C to fall from 90% to 10% of its maximum.

13. Decay time is included in turn off time and it is the time required for I_C to go from its 10% level to I_{CO}.

14. From data sheet for general purpose transistor 2N3904 the turn on and turn off times are

Turn on time $t_{on} = t_d + t_r$ = 35ns +35ns = 70ns

Turn off time toff = $t_s + t_f$ = 200ns + 50 ns = 250ns

15. DRAW AND EXPLAIN THE RESPONSE OF LOW PASS RC CIRCUIT FOR THE FOLLOWING INPUT

1. **SINUSOIDAL INPUT**
2. **STEP VOLTAGE INPUT**
3. **PULSE INPUT**
4. **SQUARE WAVE INPUT**
5. **RAMP INPUT**
6. **EXPONENTIAL INPUT**

LOW PASS RC CIRCUIT

1. A low pass RC circuit is a circuit which transmits only low frequency signals and stops high frequency signals.
2. At f = 0 → XC = ∞ i.e open circuit
3. Thus the entire input appears at the output so the output is same as the input and the gain is unity.

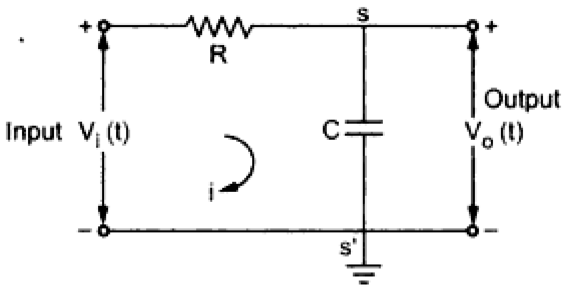


FIGURE A

$V_o = V_i - I R$

Since I = 0

$V_o = V_i$

$A = \frac{V_o}{V_i} = 1$

1. As frequency increases, X_C decreases so the circuit acts as short circuit and output falls to zero (i.e f increases then X_C decreases , so that circuit behaves like a short circuit.)

1. *LOW PASS RC CIRCUIT FOR A SINUSOIDAL INPUT*

2. The gain versus frequency curve is excited by a sinusoidal input. The curve is obtained by keeping input signal constant and varying the frequency.
3. At low frequency the output is equal to the input hence the gain is unity. $|A| = 1$
4. As the frequency increases the output decreases and gain decreases.
5. The frequency at which gain is $\frac{1}{\sqrt{2}} = 0.707$of its maximum value is called the cut off frequency.
6. For low pass filter there is no lower cut off frequencies.
7. The upper cutoff frequency is the frequency at which the gain is $\frac{1}{\sqrt{2}} = 0.707$ i.e 70% of its maximum value.
8. The bandwidth of the low pass circuit is equal to the upper cut off frequency f_2 itself.

Apply KVL

$Vi(t) = i(t)\,R + \frac{1}{C}\int i(t)\; dt$ -------------------(1)

$V_0(t) = \frac{1}{C}\int i(t)\; dt$ -------------------(2)

Taking laplace transform for (1) & (2)

$RI(S) + \frac{1}{C}\frac{I(S)}{S} = V_i(S)$

$\frac{1}{C}\frac{I(S)}{S} = V_0(S)$

For the network shown the magnitude of the steady state gain A is given by

$$A = \frac{V_0\,(S)}{V_i\,(S)}$$

$$A = \frac{\frac{1}{C}\frac{I(S)}{S}}{\mathrm{RI(S)} + \frac{1}{C}\frac{I(S)}{S}} = \frac{\frac{1}{C}\frac{\cancel{I(S)}}{S}}{\cancel{\mathrm{I(S)}}\left(\mathrm{R} + \frac{1}{SC}\right)}$$

$$A = \frac{\frac{1}{SC}}{(\mathrm{R} + \frac{1}{SC})} = \frac{\frac{1}{\cancel{SC}}}{(\frac{SRC+1}{\cancel{SC}})}$$

$$A = \frac{1}{(SRC+1)}$$

Put S= jω = j2πf

$$A = \frac{1}{(1+\mathrm{j}2\pi\mathrm{f}RC)}$$

$$\boxed{|A| = \frac{1}{\sqrt{1+(\mathrm{j}2\pi\mathrm{f}RC)^2}}} \longrightarrow (3)$$

At upper cut off frequency f_2

$$|A| = \frac{1}{\sqrt{2}}$$

$$|A| = \frac{1}{\sqrt{1+(\mathrm{j}2\pi\mathrm{f}RC)^2}}$$

Squaring both sides and equating the denominators

$2 = 1+(\mathrm{j}2\pi\mathrm{f}RC)^2$

The upper cut off frequency f2 = $\frac{1}{2\pi RC}$

Substitute f2 in equation (3)

$$|A| = \frac{1}{\sqrt{1+\left(\frac{f}{f_2}\right)^2}}$$

The angle ɵ by which the output leads the input is given by ɵ = $\tan^{-1}\left(\frac{f}{f_2}\right)$

9. LOW PASS RC CIRCUIT FOR A STEP VOLTAGE INPUT

10. A step signal is the one which maintains the value zero for all times t < 0, and maintains the value V for all times t > 0.

11. The transition between the two voltage levels takes place at t = 0

12. In the low pass RC circuit as shown in figure A, if the capacitor is initially uncharged, when a step input is applied, since the voltage across the capacitor cannot change simultaneously the output will be zero at t=0.

13. As the capacitor charges the output voltage rises exponentially towards the steady state value V with a time constant RC as shown in above figure.

Let V^l be the initial voltage across the capacitor

Apply KVL

$V_i(t) = i(t)\,R\ + \frac{1}{C}\int i(t)\ dt$ --------------------(1)

Differentiating the above equation

$\frac{dV_i(t)}{dt} = \frac{di(t)}{dt}R\ + \frac{1}{C}\ i(t)$

$V_i(t) = V$ $\qquad\qquad\qquad \frac{dV_i(t)}{dt} = 0$

$0 = \frac{di(t)}{dt}R\ + \frac{1}{C}\ i(t)$

Taking laplace transform on both sides

$0 = [SI(S) - \ I(0)\ ]R\ + \frac{I(S)}{C}$

$RI(0) = [RSI(S)\ + \frac{I(S)}{C}\ ]$

$RI(0) = I(S)\ [RS\ + \frac{1}{C}\ ]$

$I(0) = I(S)\ [S\ + \frac{1}{RC}]$

The initial current I(0) is given by $I(0) = \frac{V-V^l}{R}$

Substituting we get

$\frac{V-V^l}{R} = I(S)\ [S\ + \frac{1}{RC}]$

$I(S) = \frac{V-V^l}{R[S + \frac{1}{RC}]}$

$V_o(S) = V_i(S) - I(S)R$

$V_o(S) = \frac{V}{S} - \frac{V-V^l}{R[S + \frac{1}{RC}]}R \quad = \frac{V}{S} - \frac{V-V^l}{[S + \frac{1}{RC}]}$

Taking inverse laplace transform on both sides

$V_o(t) = V - [V - V^l]e^{-t/RC}$

If the capacitor is initially uncharged then $V_o(t) = V\ [1 - e^{-t/RC}]$

The rise time $t_r = 2.2RC$

This indicates that the rise time t_r is proportional to the time constant RC of the circuit

14. LOW PASS RC CIRCUIT FOR A PULSE INPUT

15. LOW PASS RC CIRCUIT FOR A SQUARE WAVE INPUT

16. LOW PASS RC CIRCUIT FOR A RAMP INPUT

17. ***LOW PASS RC CIRCUIT FOR A EXPONENTIAL WAVE INPUT***

18.

Taking the inverse Laplace transform on both sides and letting $RC/\tau = n$,

$$v_o(t) = V\left[1 - \frac{e^{-t/\tau}}{1-n} + \frac{e^{-t/RC}}{\frac{1}{n} - 1}\right]$$

If $t/\tau = x$, then $v_o(t) = V\left[1 - \frac{e^{-x}}{1-n} + \frac{n}{1-n}\, e^{-x/n}\right]$, if $n \neq 1$

and $\qquad v_o(t) = 1 - (1 + x)e^{-x}, \qquad$ if $n = 1$

These are the expressions for the voltage across the capacitor of a low-pass RC circuit excited by an exponential input of rise time $t_{r1} = 2.2\tau$.

If an exponential of rise time t_{r1} is passed through a low-pass circuit with rise time t_{r2}, the rise time of the output waveform t_r will be given by an empirical relation, $t_r = 1.05\sqrt{t_{r1}^2 + t_{r2}^2}$. This is same as the rise time obtained when a step is applied to a cascade of two circuits of rise times t_{r1} and t_{r2} assuming that the second circuit does not load the first.

19. DRAW AND EXPLAIN THE RESPONSE OF HIGH PASS RC CIRCUIT FOR THE FOLLOWING INPUT

1. **SINUSOIDAL INPUT**
2. **STEP VOLTAGE INPUT**
3. **PULSE INPUT**
4. **SQUARE WAVE INPUT**
5. **RAMP INPUT**
6. **EXPONENTIAL INPUT**

HIGH PASS RC CIRCUIT

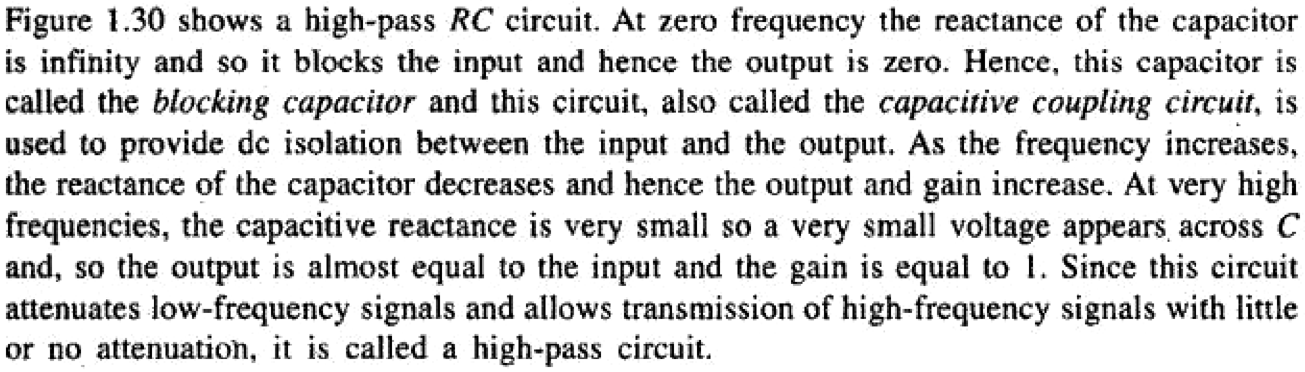


Figure 1.30 shows a high-pass *RC* circuit. At zero frequency the reactance of the capacitor is infinity and so it blocks the input and hence the output is zero. Hence, this capacitor is called the *blocking capacitor* and this circuit, also called the *capacitive coupling circuit*, is used to provide dc isolation between the input and the output. As the frequency increases, the reactance of the capacitor decreases and hence the output and gain increase. At very high frequencies, the capacitive reactance is very small so a very small voltage appears across *C* and, so the output is almost equal to the input and the gain is equal to 1. Since this circuit attenuates low-frequency signals and allows transmission of high-frequency signals with little or no attenuation, it is called a high-pass circuit.

1. HIGH PASS RC CIRCUIT FOR A SINUSOIDAL INPUT

Figure 1.31(a) shows the Laplace transformed high-pass RC circuit. The gain versus frequency curve of a high-pass circuit excited by a sinusoidal input is shown in Figure 1.31(b). For a sinusoidal input v_i, the output signal v_o increases in amplitude with increasing frequency. The frequency at which the gain is $1/\sqrt{2}$ of its maximum value is called the lower cut-off or lower 3-dB frequency. For a high-pass circuit, there is no upper cut-off frequency because all high frequency signals are transmitted with zero attenuation. Therefore, $f_2 = \infty$. Hence bandwidth $= f_2 - f_1 = \infty$.

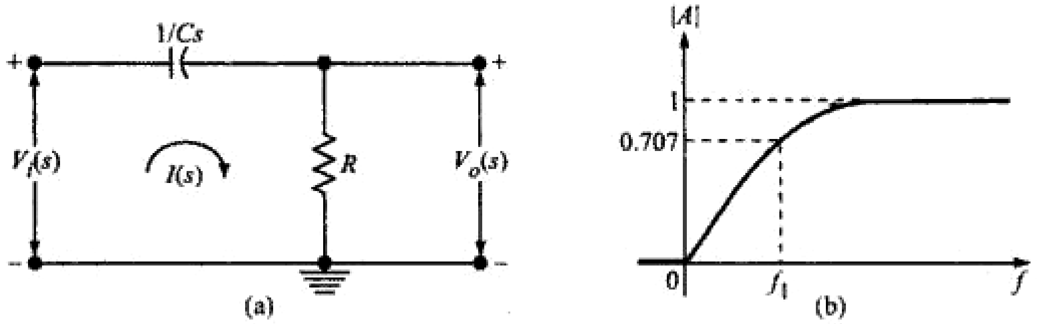


Figure 1.31 (a) Laplace transformed high-pass circuit and (b) gain versus frequency plot.

Expression for the lower cut-off frequency

For the high-pass RC circuit shown in Figure 1.31(a), the magnitude of the steady-state gain A, and the angle θ by which the output leads the input are given by

$$A = \frac{V_o(s)}{V_i(s)} = \frac{R}{R+\frac{1}{Cs}} = \frac{1}{1+\frac{1}{RCs}}$$

Putting $$s = j\omega,\ A = \frac{1}{1 - j\frac{1}{\omega RC}} = \frac{1}{1 - j\frac{1}{2\pi fRC}}$$

$$\therefore \qquad |A| = \frac{1}{\sqrt{1+\left(\frac{1}{2\pi fRC}\right)^2}} \quad \text{and} \quad \theta = -\tan^{-1}\frac{1}{2\pi fRC}$$

At the lower cut-off frequency f_1, $|A| = 1/\sqrt{2}$

$$\therefore \qquad \frac{1}{\sqrt{1+\left(\frac{1}{2\pi f_1 RC}\right)^2}} = \frac{1}{\sqrt{2}}$$

Squaring and equating the denominators,

$$\frac{1}{2\pi f_1 RC} = 1 \quad \text{i.e.} \quad f_1 = \frac{1}{2\pi RC}$$

This is the expression for the lower cut-off frequency of a high-pass circuit.

2. HIGH PASS RC CIRCUIT FOR A STEP INPUT

When a step signal of amplitude V volts shown in Figure 1.32(a) is applied to the high-pass RC circuit of Figure 1.30, since the voltage across the capacitor cannot change instantaneously the output will be just equal to the input at $t = 0$ (for $t < 0$, $v_i = 0$ and $v_o = 0$). Later when the capacitor charges exponentially, the output reduces exponentially with the same time constant RC. The expression for the output voltage for $t > 0$ is given by

$$v_o(t) = V_f - (V_f - V_{in})e^{-t/RC} = 0 - (0 - V)e^{-t/RC} = Ve^{-t/RC}$$

Figure 1.32(b) shows the response of the circuit for large, small, and very small time constants.

For $t > 5\tau$, the output will reach more than 99% of its final value. Hence although the steady state is approached asymptotically, for most applications we may assume that the final value has been reached after 5τ. If the initial slope of the exponential is maintained, the output falls to zero in a time $t = \tau$.

The voltage across a capacitor can change instantaneously only when an infinite current passes through it, because for any finite current $i(t)$ through the capacitor, the instantaneous change in voltage across the capacitor is given by $\frac{1}{C}\int_0^0 i(t)\,dt = 0$.

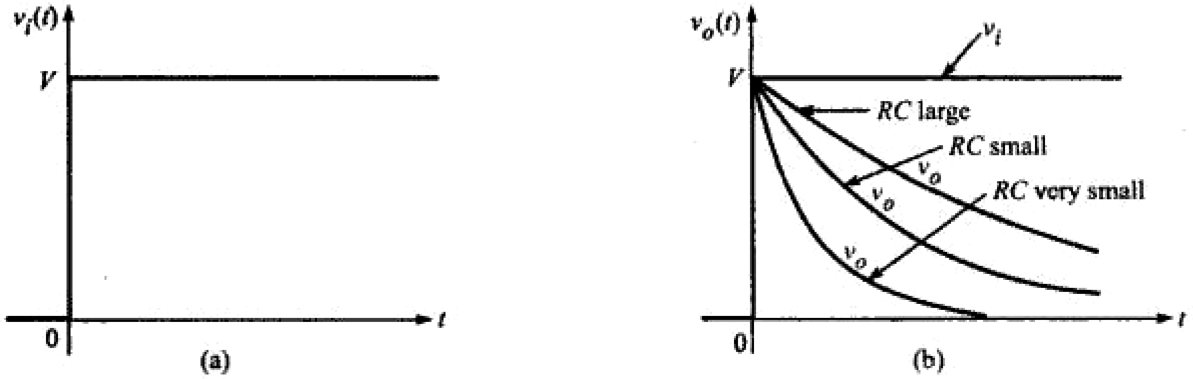


Figure 1.32 (a) Step input and (b) step response for different time constants.

3. HIGH PASS RC CIRCUIT FOR A PULSE INPUT

A pulse of amplitude V and duration t_p shown in Figure 1.4(a) is nothing but the sum of a positive step of amplitude V starting at $t = 0$ and a negative step of amplitude V starting at t_p as shown in Figure 1.4(b). So, the response of the circuit for $0 < t < t_p$ for the pulse input is the same as that for a step input and is given by $v_o(t) = Ve^{-t/RC}$. At $t = t_p$, $v_o(t) = V_p = Ve^{-t_p/RC}$. At $t = t_p$, since the input falls by V volts suddenly and since the voltage across the capacitor cannot change instantaneously, the output also falls suddenly by V volts to $V_p - V$. Hence at $t = t_{p+}, v_o(t) = Ve^{-t_p/RC} - V$. Since $V_p < V$, $V_p - V$ is negative. So there is an undershoot at $t = t_p$ and hence for $t > t_p$, the output is negative. For $t > t_p$, the output rises exponentially towards zero with a time constant RC according to the expression $(Ve^{-t_p/RC} - V)e^{-(t-t_p)/RC}$.

The output waveforms for $RC >> t_p$, RC comparable to t_p and $RC << t_p$ are shown in Figures 1.33(a), (b), and (c) respectively. There is distortion in the outputs and the distortion is the least when the time constant is very large. Observe that there is positive area and negative area in the output waveforms. The negative area will always be equal to the positive area. So if the time constant is very large the tilt (the almost linear decrease in the output voltage) will be small and hence the undershoot will be very small, and for $t > t_p$, the output rises towards the zero level very very slowly. If the time constant is very small compared to the pulse width (i.e. $RC/t_p << 1$), the output consists of a positive spike or pip of amplitude V volts at the beginning of the pulse and a negative spike of the same amplitude at the end of the pulse. Hence a high-pass circuit with a very small time constant is called a *peaking circuit* and this process of converting pulses into pips by means of a circuit of short time constant is called peaking.

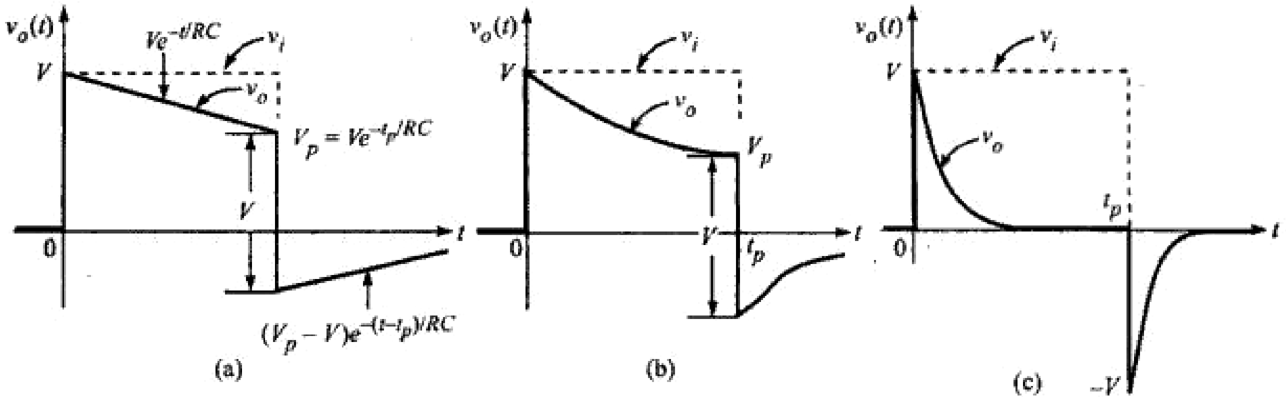
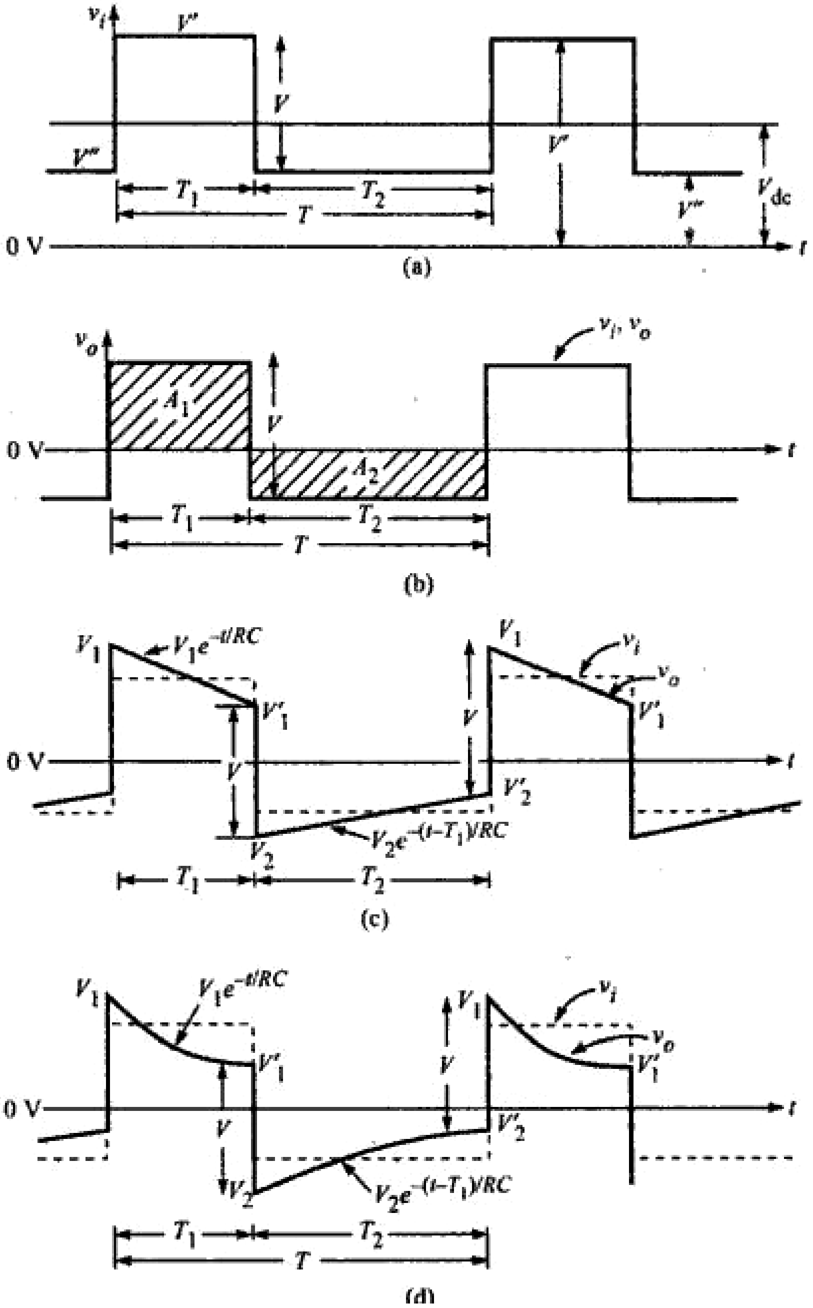


Figure 1.33 Pulse response for (a) $RC >> t_p$, (b) RC comparable to t_p and (c) $RC << t_p$.

4. HIGH PASS RC CIRCUIT FOR A SQUARE WAVE INPUT

A square wave shown in Figure 1.34(a) is a periodic waveform, which maintains itself at one constant level V' with respect to ground for a time T_1 and then changes abruptly to another level V'' and remains constant at that level for a time T_2, and then repeats itself at regular intervals of $T = T_1 + T_2$. A square wave may be treated as a series of positive and negative steps. The shape of the output depends on the time constant of the circuit. Figures 1.34(b), 1.34(c), 1.34(d), and 1.34(e) show the output waveforms of the high-pass RC circuit under steady-state conditions for the cases (a) $RC >> T$, (b) $RC > T$, (c) $RC = T$, and (d) $RC << T$ respectively.

When the time constant is arbitrarily large (i.e. RC/T_1 and RC/T_2 are very very large in comparison to unity) the output is same as the input but with zero dc level. When $RC > T$, the output is in the form of a tilt. When RC is comparable to T, the output rises and falls exponentially. When $RC << T$ (i.e. RC/T_1 and RC/T_2 are very very small in comparison to unity), the output consists of alternate positive and negative spikes. In this case the peak-to-peak amplitude of the output is twice the peak-to-peak value of the input.



(a)

(b)

(c)

(d)

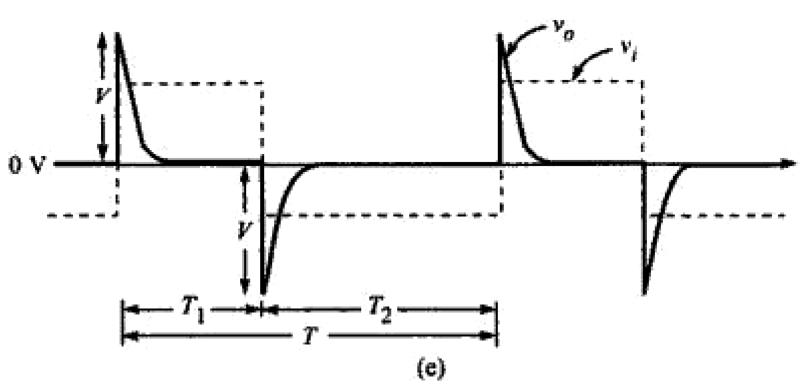


Figure 1.34 (a) A square wave input, (b) output when *RC* is arbitrarily large, (c) output when *RC* > *T*, (d) output when *RC* is comparable to *T*, and (e) output when *RC* << *T*.

In fact, for any periodic input waveform under steady-state conditions, the average level of the output waveform from the high-pass circuit of Figure 1.30 is always zero independently of the dc level of the input. The proof is as follows:

Writing KVL around the loop of Figure 1.30,

$$v_i(t) = \frac{1}{C}\int i(t)\,dt + v_o(t)$$

$$= \frac{1}{RC}\int v_o(t)\,dt + v_o(t) \qquad \left(\because\ i(t) = \frac{v_o(t)}{R}\right)$$

Differentiating,

$$\frac{dv_i(t)}{dt} = \frac{v_o(t)}{RC} + \frac{dv_o(t)}{dt}$$

Multiplying by *dt* and integrating this equation over one period *T*,

$$\int_{t=0}^{t=T} dv_i(t) = \int_{t=0}^{t=T} \frac{v_o(t)\,dt}{RC} + \int_{t=0}^{t=T} dv_o(t)$$

i.e.

$$v_i(T) - v_i(0) = \frac{1}{RC}\int_0^T v_o(t)dt + v_o(T) - v_o(0)$$

Under steady-state conditions, the output waveform (as well as the input signal) is repetitive with a period *T* so that $v_o(T) = v_o(0)$ and $v_i(T) = v_i(0)$.

Hence $\int_0^T v_o(t)dt = 0$. Since this integral represents the area under the output waveform over one cycle, we can say that the average level of the steady-state output signal is always zero.

This can also be proved based on frequency domain analysis as follows.

The periodic input signal may be resolved into a Fourier series consisting of a constant term and an infinite number of sinusoidal components whose frequencies are multiples of $f = 1/T$. Since the blocking capacitor presents infinite impedance to the dc input voltage, none of these dc components reach the output under steady-state conditions. Hence the output signal is a sum of sinusoids whose frequencies are multiples of *f*. This waveform is therefore periodic with a fundamental period *T* but without a dc component.

With respect to the high-pass circuit of Figure 1.30, we can say that:

1. The average level of the output signal is always zero, independently of the average level of the input. The output must consequently extend in both negative and positive directions with respect to the zero voltage axis and the area of the part of the waveform above the zero axis must equal the area which is below the zero axis.
2. When the input changes abruptly by an amount V, the output also changes abruptly by an equal amount and in the same direction.
3. During any finite time interval when the input maintains a constant level, the output decays exponentially towards zero voltage.

Under steady-state conditions, the capacitor charges and discharges to the same voltage levels in each cycle. So the shape of the output waveform is fixed.

For $0 < t < T_1$, the output is given by $v_{o1} = V_1 e^{-t/RC}$

At $t = T_1$, $v_{o1} = V_1' = V_1\, e^{-T_1/RC}$

For $T_1 < t < T_1 + T_2$, the output is $v_{o2} = V_2 e^{-(t-T_1)/RC}$

At $t = T_1 + T_2$, $v_{o2} = V_2' = V_2 e^{-T_2/RC}$

Also $V_1' - V_2 = V$ and $V_1 - V_2' = V$

From these relations V_1, V_1', V_2 and V_2' can be computed.

5. *HIGH PASS RC CIRCUIT FOR A RAMP INPUT*

A waveform which is zero for $t < 0$ and which increases linearly with time for $t > 0$ is called a ramp or sweep voltage.

When the high-pass RC circuit of Figure 1.30 is excited by a ramp input $v_i(t) = \alpha t$, where α is the slope of the ramp, then

$$V_i(s) = \frac{\alpha}{s^2}$$

From the Laplace transformed circuit of Figure 1.31(a),

$$V_o(s) = V_i(s)\frac{R}{R+\dfrac{1}{Cs}} = \frac{\alpha}{s^2}\,\frac{RCs}{1+RCs}$$

$$= \frac{\alpha}{s\left(s+\dfrac{1}{RC}\right)} = \alpha RC\left(\frac{1}{s}-\frac{1}{s+\dfrac{1}{RC}}\right)$$

Taking the inverse Laplace transform on both sides,
$v_o(t) = \alpha RC(1 - e^{-t/RC})$

For times t which are very small in comparison with RC, we have

$$v_o(t) = \alpha RC\left[1-\left\{1+\left(\frac{-t}{RC}\right)+\left(\frac{-t}{RC}\right)^2\frac{1}{2!}+\left(\frac{-t}{RC}\right)^3\frac{1}{3!}+\dots\right\}\right]$$

$$= \alpha RC\left[\frac{t}{RC} - \frac{t^2}{2(RC)^2} + \dots\right]$$

$$= \alpha t - \frac{\alpha t^2}{2RC} = \alpha t\left(1 - \frac{t}{2RC}\right)$$

Figure 1.36 shows the response of the high-pass circuit for a ramp input when (a) $RC >> T$, and (b) $RC << T$, where T is the duration of the ramp. For small values of T, the output signal falls away slightly from the input as shown in the Figure 1.36(a).

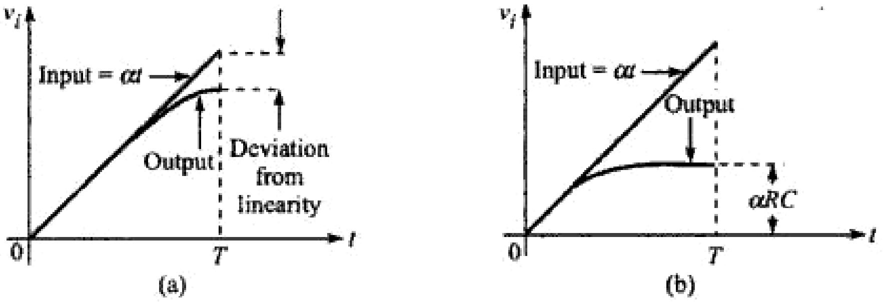


Figure 1.36 Response of the high-pass circuit for a ramp input when (a) $RC >> T$ and (b) $RC << T$.

When a ramp signal is transmitted through a linear network, the output departs from the input. A measure of the departure from linearity expressed as the transmission error e_t is defined as the difference between the input and the output divided by the input. The transmission error at a time $t = T$ is then

$$e_t = \left.\frac{v_i - v_o}{v_i}\right|_{t=T} \approx \left.\frac{\alpha t - \alpha t\left(1 - \frac{t}{2RC}\right)}{\alpha t}\right|_{t=T} \approx \frac{T}{2RC} = \pi f_1 T$$

where $f_1 = \dfrac{1}{2\pi RC}$ is the lower 3-dB frequency of the high-pass circuit.

For large values of t in comparison with RC, the output approaches the constant value αRC as indicated in Figure 1.36(b).

6. *HIGH PASS RC CIRCUIT FOR A EXPONENTIAL INPUT*

When the high-pass RC circuit of Figure 1.30 is excited by an exponential input $v_i(t) = V(1 - e^{-t/\tau})$ shown in Figure 1.8, where τ is the time constant of the input, the output taken across the resistor is given by

$$v_R(t) = v_i(t) - v_C(t)$$

For an exponential input $v_i(t) = V(1 - e^{-t/\tau})$, the expression for the voltage across the capacitor (derived earlier while dealing with *RC* low-pass circuit) is

$$v_o(t) = V\left(1 - \frac{e^{-x}}{1-n} + \frac{n}{1-n}e^{-x/n}\right)$$

$$\therefore \qquad v_R(t) = V(1 - e^{-t/\tau}) - V\left(1 - \frac{e^{-x}}{1-n} + \frac{n}{1-n}e^{-x/n}\right)$$

$$= V(1 - e^{-x}) - V\left(1 - \frac{e^{-x}}{1-n} + \frac{n}{1-n}e^{-x/n}\right)$$

$$= V\left(\frac{e^{-x}}{1-n} - e^{-x}\right) - V\left(\frac{ne^{-x/n}}{1-n}\right)$$

$$= V\left[e^{-x}\left(\frac{1}{1-n} - 1\right) - \frac{n}{1-n}e^{-x/n}\right]$$

$$= V\frac{n}{n-1}[e^{-x/n} - e^{-x}], \quad \text{if } n \neq 1$$

$$\text{and} \qquad v_R(t) = Vxe^{-x}, \quad \text{if } n = 1$$

If the time constant of the circuit is very high, n is high and the second term of the equation for $n \neq 1$ is negligible compared to the first term except for small values of time.

$$\therefore \qquad v_o \approx V\frac{n}{n-1}e^{-x/n} \approx Ve^{-t/RC}$$

This equation agrees with the way the circuit should behave for an ideal step voltage. The response of the high-pass circuit for different values of n is shown in Figure 1.37.

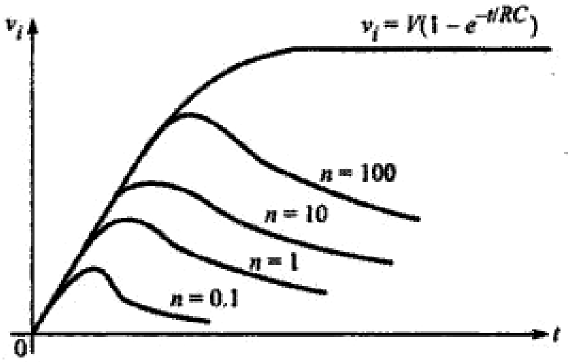


Figure 1.37 Response of high-pass circuit for exponential input.

Near the origin of time the output follows the input. Also, the smaller the circuit time constant, the smaller will be the output peak and the narrower will be the pulse. The larger the circuit time constant, the larger will be the peak output and also the wider will be the pulse.

The maximum output occurs when

$$\frac{dv_o}{dt}=0$$

i.e.
$$\frac{d}{dt}\left[V\frac{n}{n-1}(e^{-x/n}-e^{-x})\right]=0$$

i.e.
$$V\frac{n}{n-1}\left[\left(\frac{-1}{n}\right)\left(\frac{1}{\tau}\right)e^{-x/n}-e^{-x}\left(-\frac{1}{\tau}\right)\right]=0$$

i.e.
$$V\frac{n}{n-1}\left[\frac{e^{-x}}{\tau}-\frac{e^{-x/n}}{n\tau}\right]=0$$

∴
$$e^{-x}=\frac{e^{-x/n}}{n}$$

i.e.
$$n=e^{x(1-(1/n))}=e^{x((n-1)/n)}$$

i.e.
$$\ln n=\frac{x(n-1)}{n}\quad\text{or}\quad x=\frac{n}{n-1}\ln n$$

Since $x=t/\tau$, the time to rise to peak t_p is given by

$$t_p=\tau\frac{n}{n-1}\ln n$$

Also,
$$-x=\frac{-n}{n-1}\ln n=\ln[n^{n/(1-n)}]$$

To obtain the maximum value of output, substitute this value of $-x$ in the expression for $v_o(t)$

∴
$$V_o(\max)=\frac{Vn}{n-1}\exp\left[\frac{1}{n}\ln(n^{n/(1-n)})-\ln(n^{n/(1-n)})\right]$$

$$=V\frac{n}{n-1}\exp\left[\ln(n^{1/(1-n)})-\ln(n^{n/(1-n)})\right]$$

$$=V\frac{n}{n-1}\left[n^{(1/(1-n)}-n^{(n/(1-n))}\right]$$

$$=\frac{V}{n-1}\left[n^{1+(1/(1-n))}-n^{1+(n/(1-n))}\right]$$

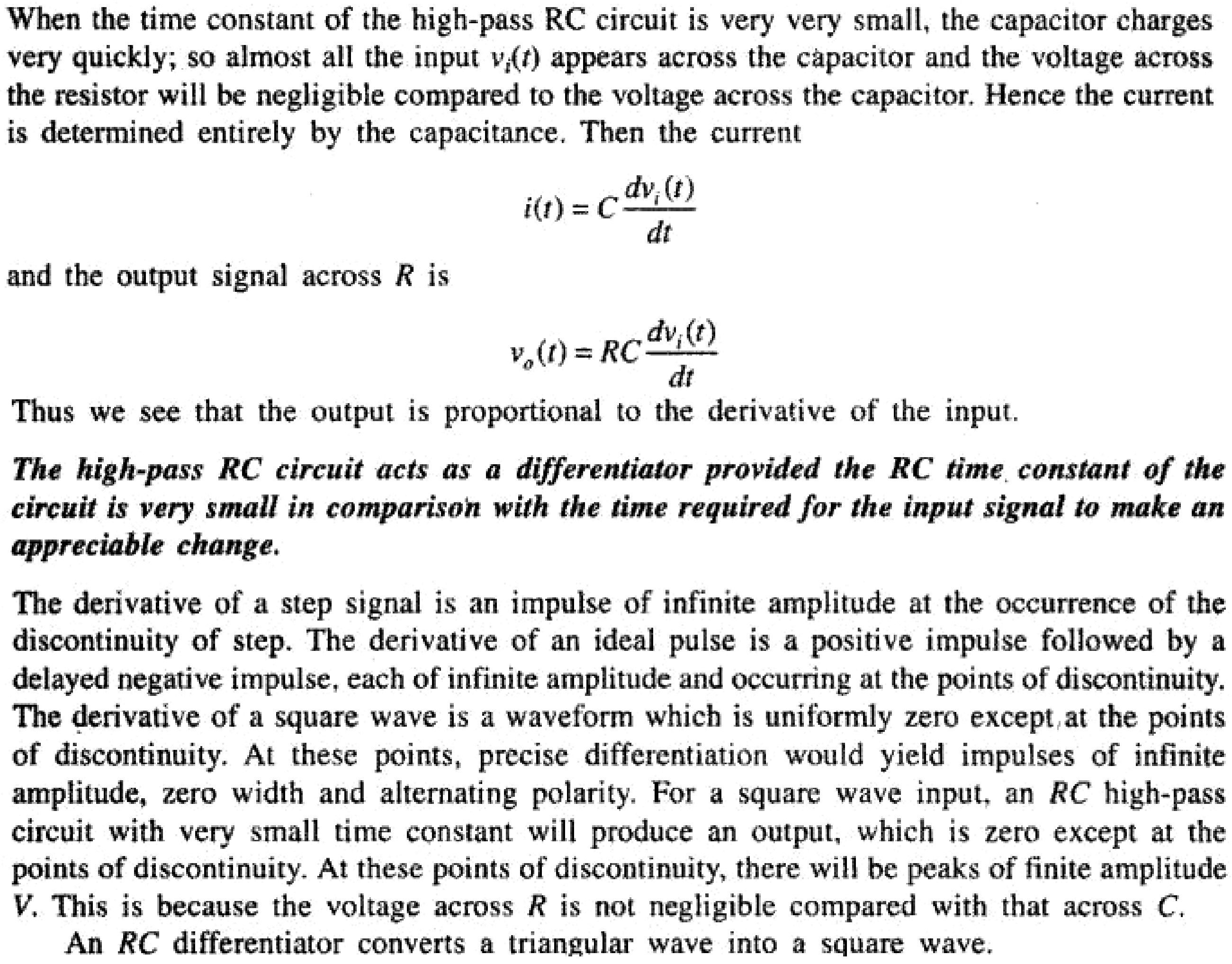
$$=\frac{V}{n-1}\left[n^{n/(1-n)}-n^{1/(1-n)}\right]$$

$$=\frac{V}{n-1}\left[n^{1/(1-n)}(n-1)\right]$$

$$=Vn^{1/(1-n)},\quad\text{for } n\neq 1$$

$$\frac{V_o(\max)}{V}=n^{1/(1-n)},\quad\text{for } n\neq 1$$

7. **EXPLAIN DIFFERENTIATING CIRCUIT AND DERIVE THE CONDITION FOR THE ACHIEVEMENT OF GOOD DIFFERENTIATION.[APRIL 2016]**



When the time constant of the high-pass RC circuit is very very small, the capacitor charges very quickly; so almost all the input $v_i(t)$ appears across the capacitor and the voltage across the resistor will be negligible compared to the voltage across the capacitor. Hence the current is determined entirely by the capacitance. Then the current

$$i(t) = C\frac{dv_i(t)}{dt}$$

and the output signal across R is

$$v_o(t) = RC\frac{dv_i(t)}{dt}$$

Thus we see that the output is proportional to the derivative of the input.

The high-pass RC circuit acts as a differentiator provided the RC time constant of the circuit is very small in comparison with the time required for the input signal to make an appreciable change.

The derivative of a step signal is an impulse of infinite amplitude at the occurrence of the discontinuity of step. The derivative of an ideal pulse is a positive impulse followed by a delayed negative impulse, each of infinite amplitude and occurring at the points of discontinuity. The derivative of a square wave is a waveform which is uniformly zero except at the points of discontinuity. At these points, precise differentiation would yield impulses of infinite amplitude, zero width and alternating polarity. For a square wave input, an RC high-pass circuit with very small time constant will produce an output, which is zero except at the points of discontinuity. At these points of discontinuity, there will be peaks of finite amplitude V. This is because the voltage across R is not negligible compared with that across C.

An RC differentiator converts a triangular wave into a square wave.

For the ramp $v_i = \alpha t$, the value of $RC(dv_i/dt) = \alpha RC$. This is true except near the origin. The output approaches the proper derivative value only after a lapse of time corresponding to several time constants. The error near $t = 0$ is again due to the fact that in this region the voltage across R is not negligible compared with that across C.

If we assume that the leading edge of a pulse can be approximated by a ramp, then we can measure the rate of rise of the pulse by using a differentiator. The peak output is measured on an oscilloscope, and from the equation $v_o = \alpha RC$, we see that this voltage divided by the product RC gives the slope α.

A criteria for good differentiation in terms of steady-state sinusoidal analysis is, that if a sine wave is applied to the high-pass RC circuit, the output will be a sine wave shifted by a leading angle θ such that:

$$\tan\theta = \frac{X_C}{R} = \frac{1}{\omega RC}$$

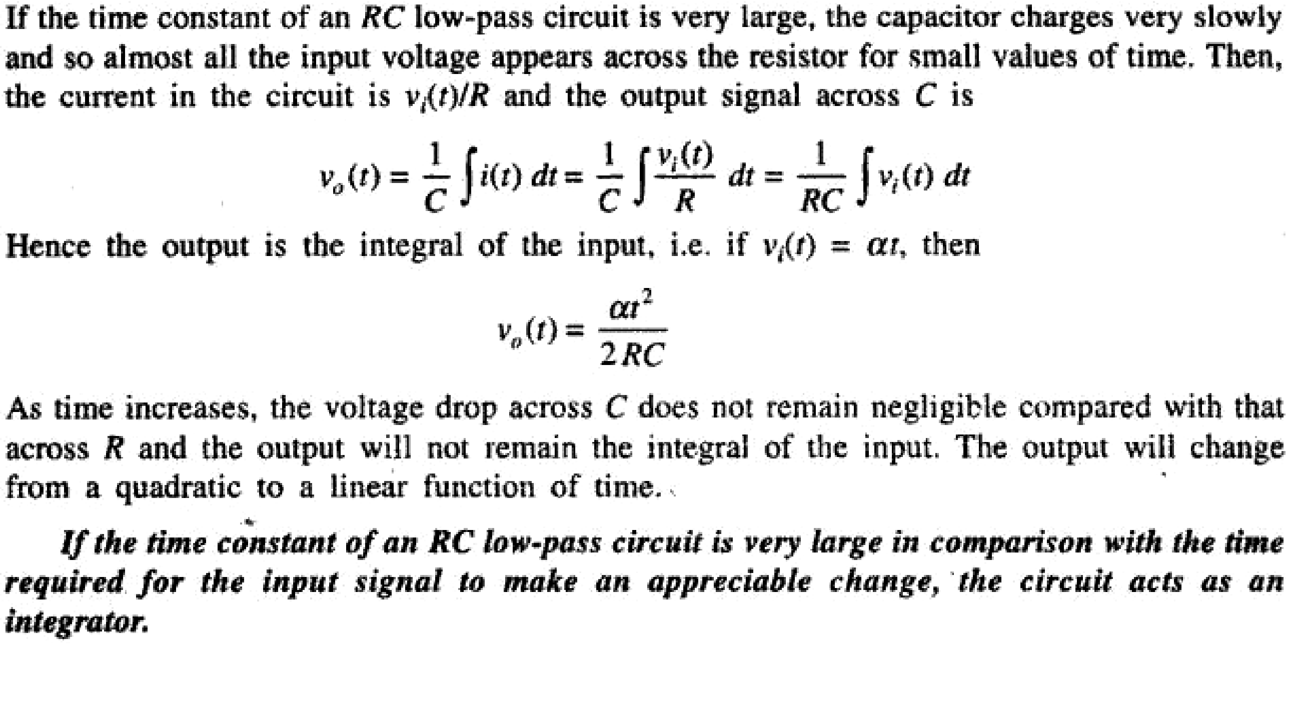
with the output being proportional to $\sin(\omega t + \theta)$. In order to have true differentiation, we must obtain $\cos \omega t$. In other words, θ must equal 90°. This result can be obtained only if $R = 0$ or $C = 0$. However, if $\omega RC = 0.01$, then $1/\omega RC = 100$ and $\theta = 89.4°$, which is sufficiently close to 90° for most purposes. If $\omega RC = 0.1$, then $\theta = 84.3°$ and for some applications this may be close enough to 90°.

If the peak value of input is V_m, the output is

$$v_o = \frac{V_m R}{\sqrt{R^2 + \dfrac{1}{\omega^2 C^2}}} \sin(\omega t + \theta)$$

and if $\omega RC << 1$, then the output is approximately $V_m \omega RC \cos \omega t$. This result agrees with the expected value $RC(dv_i/dt)$. If $\omega RC = 0.01$, then the output amplitude is 0.01 times the input amplitude.

8. **EXPLAIN INTEGRATOR CIRCUIT AND DERIVE THE CONDITION FOR THE ACHIEVEMENT OF GOOD INTEGRATION.[APRIL 2016]**



If the time constant of an *RC* low-pass circuit is very large, the capacitor charges very slowly and so almost all the input voltage appears across the resistor for small values of time. Then, the current in the circuit is $v_i(t)/R$ and the output signal across *C* is

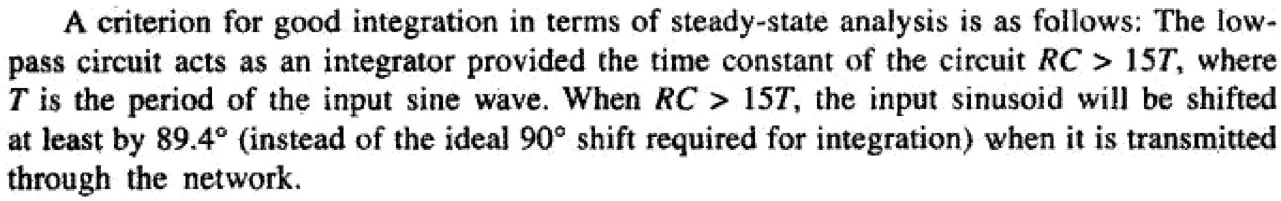
$$v_o(t) = \frac{1}{C}\int i(t)\,dt = \frac{1}{C}\int \frac{v_i(t)}{R}\,dt = \frac{1}{RC}\int v_i(t)\,dt$$

Hence the output is the integral of the input, i.e. if $v_i(t) = \alpha t$, then

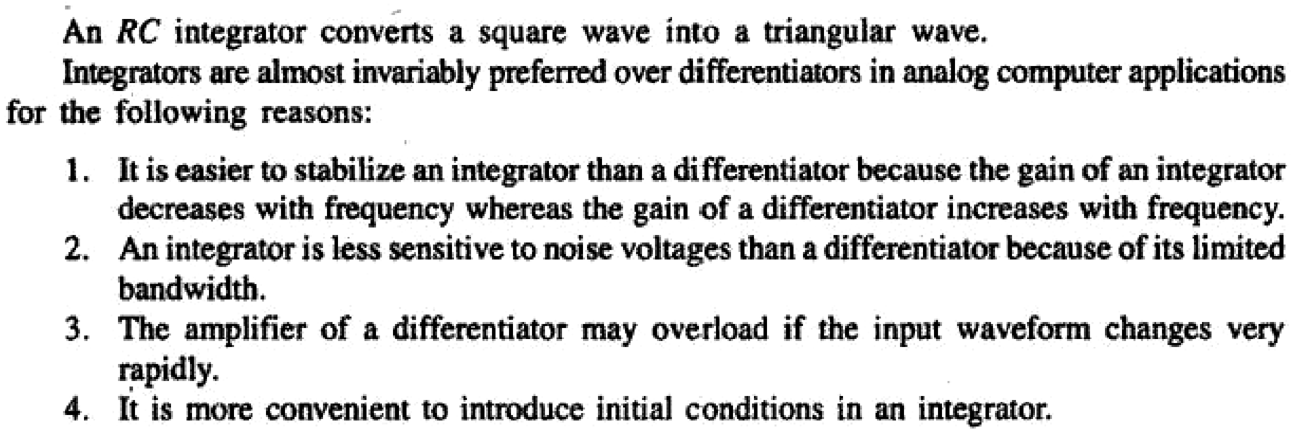
$$v_o(t) = \frac{\alpha t^2}{2RC}$$

As time increases, the voltage drop across *C* does not remain negligible compared with that across *R* and the output will not remain the integral of the input. The output will change from a quadratic to a linear function of time.

If the time constant of an RC low-pass circuit is very large in comparison with the time required for the input signal to make an appreciable change, the circuit acts as an integrator.



A criterion for good integration in terms of steady-state analysis is as follows: The low-pass circuit acts as an integrator provided the time constant of the circuit $RC > 15T$, where T is the period of the input sine wave. When $RC > 15T$, the input sinusoid will be shifted at least by 89.4° (instead of the ideal 90° shift required for integration) when it is transmitted through the network.



An RC integrator converts a square wave into a triangular wave.

Integrators are almost invariably preferred over differentiators in analog computer applications for the following reasons:

1. It is easier to stabilize an integrator than a differentiator because the gain of an integrator decreases with frequency whereas the gain of a differentiator increases with frequency.
2. An integrator is less sensitive to noise voltages than a differentiator because of its limited bandwidth.
3. The amplifier of a differentiator may overload if the input waveform changes very rapidly.
4. It is more convenient to introduce initial conditions in an integrator.

UNIT – II MULTIVIBRATORS AND TIME BASE CIRCUITS

. Bistable, monostable and astable multi-vibrators using BJT– Schmitt trigger circuit using BJT – Voltage and current sawtooth sweeps – Fixed amplitude sweep – Constant current sweep – UJT– Sawtooth Miller and boot strap time base–Multivibrators using negative resistance devices (UJT and Tunnel diodes).

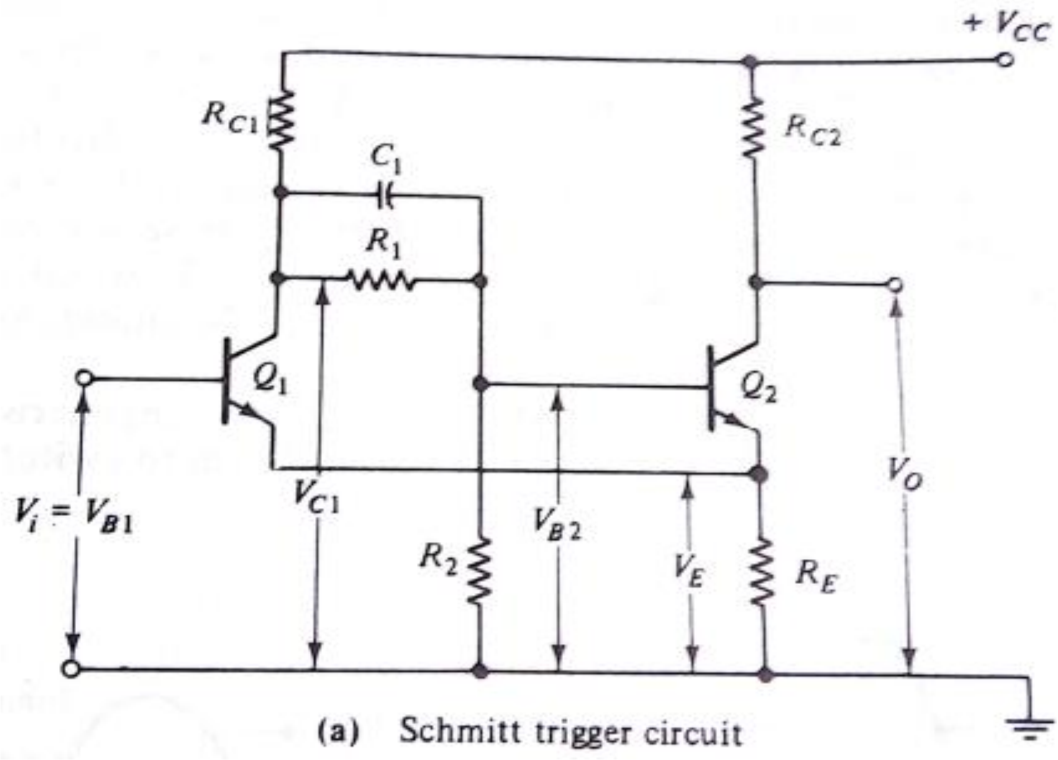
11 MARKS

1. DISCUSS ABOUT WORKING OF SCHMITT TRIGGER WITH NEAT DIAGRAM. [NOV-2016]

Definition:

- It is a fast operating voltage level detector.
- When the input voltage arrives at the upper or lower trigger levels, the output voltage rapidly changes.
- It operates with almost any input waveform and always gives a pulse type output.

Circuit Description:



(a) Schmitt trigger circuit

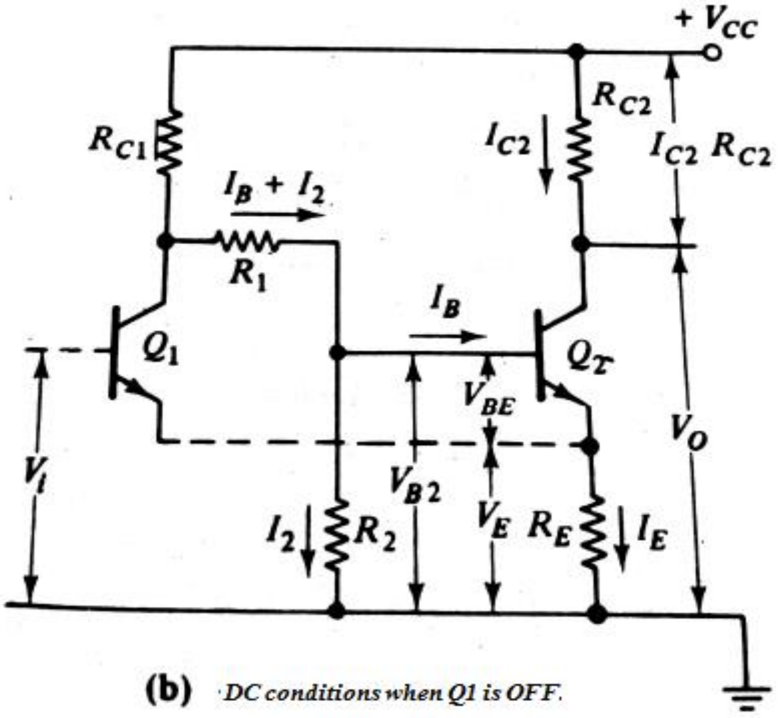
- Transistor Q1 & Q2 have a common emitter resistor R_E.
- The Q2 base voltage (V_{B2}) is derived via potential divider (R1 & R2) from the collector of Q1.
- Transistor Q1 & Q2 have collector resistance R_{C1} & R_{C2} respectively.

- ➢ The arrangement is such that when transistor Q1 is ON, Transistor Q2 is OFF; then transistor Q1 is OFF, Transistor Q2 is ON.
- ➢ Speed up capacitor C_1 is provided to improve the circuit switching speed.

Circuit Operation :

Consider the DC conditions when Q1 is OFF.

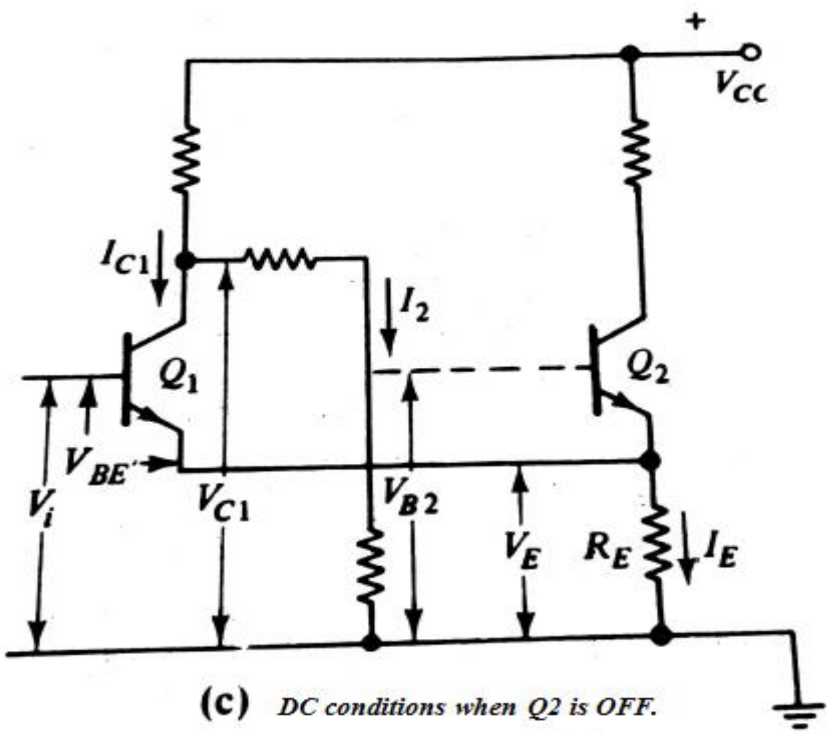
- ➢ When Q1 is off it can be regarded as an open circuit; therefore it can be left out of the circuit as shown below.
- ➢ The Q2 base voltage is now derived from V_{CC} by a potential divider consisting of R1,R2 & R_{C1}.
- ➢ Thus Q2 is ON and a collector current IC2 flows producing a voltage drop across R_{C2}
- ➢ The output voltage is $(V_{CC} – I_{C2}R_{C2})$



(b) *DC conditions when Q1 is OFF.*

Consider the DC conditions when Q2 is OFF.

- ➢ When Q2 is off it can be regarded as an open circuit; therefore it can be left out of the circuit as shown below.



(c) *DC conditions when Q2 is OFF.*

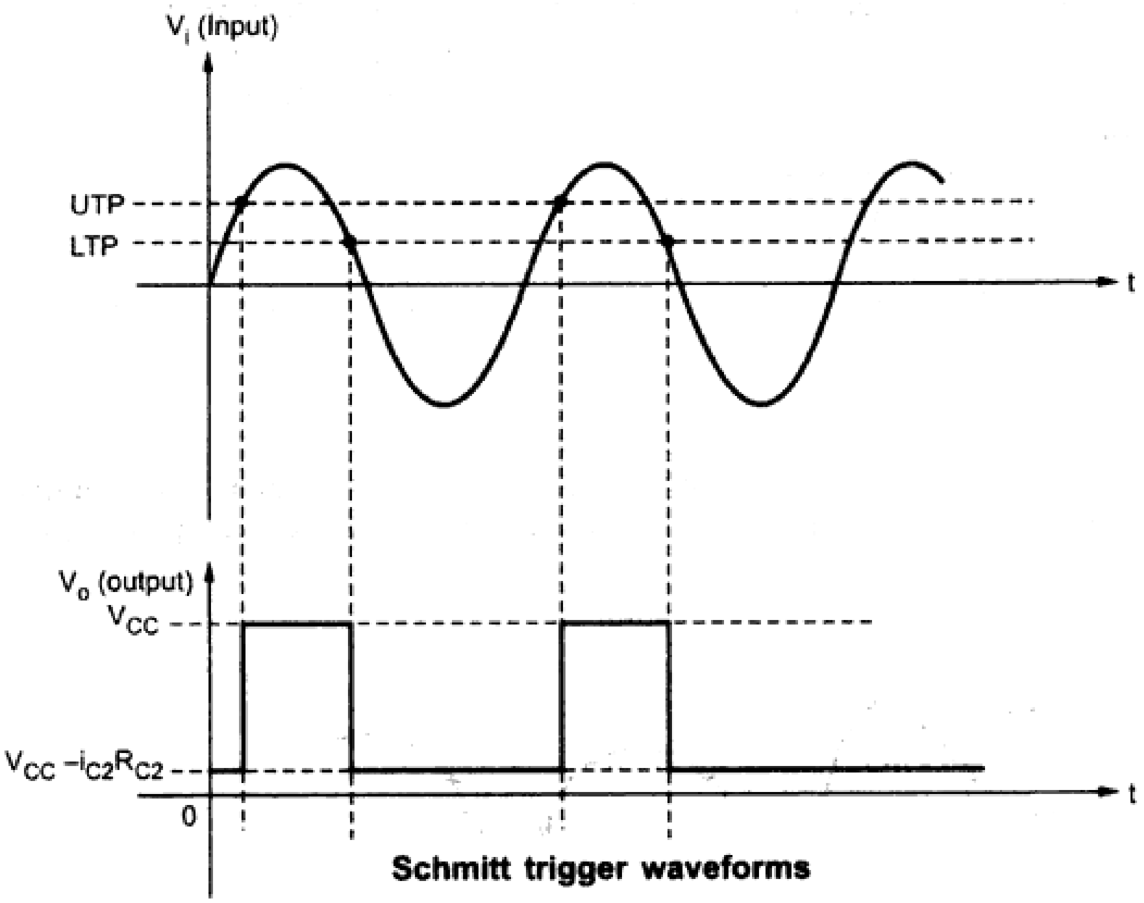
- Q1 is now triggered ON, the emitter voltage becomes $V_E = V_i – V_{BE}$.
- Also the collector current I_{C1} causes a voltage drop across R_{C1} causing V_{B2} to fall below the level of V_E.
- Thus when the Q1 is ON, Q2 is biased OFF and I_{C2} becomes zero.
- At this point there is no longer any significant voltage drop across R_{C2} and the output voltage is approximately V_{CC}.

Upper Trigger Point (UTP)

- From fig (b) it is seen that with Q2 ON, $V_E = V_{B2} – V_{BE}$. This is the voltage at the emitter terminal of both transistors, since they are connected together.
- Transistor Q1 will not switch ON until its base voltage becomes greater than VE.
- Q1 switches ON approximately at $V_i = V_E + V_{BE}$, which equals V_{B2}.
- If V_i is suddenly made greater than this level Q1 would switch ON rapidly.
- The lowest level of V_i that causes Q1 to switch ON is known as the ***Upper trigger point (UTP)***

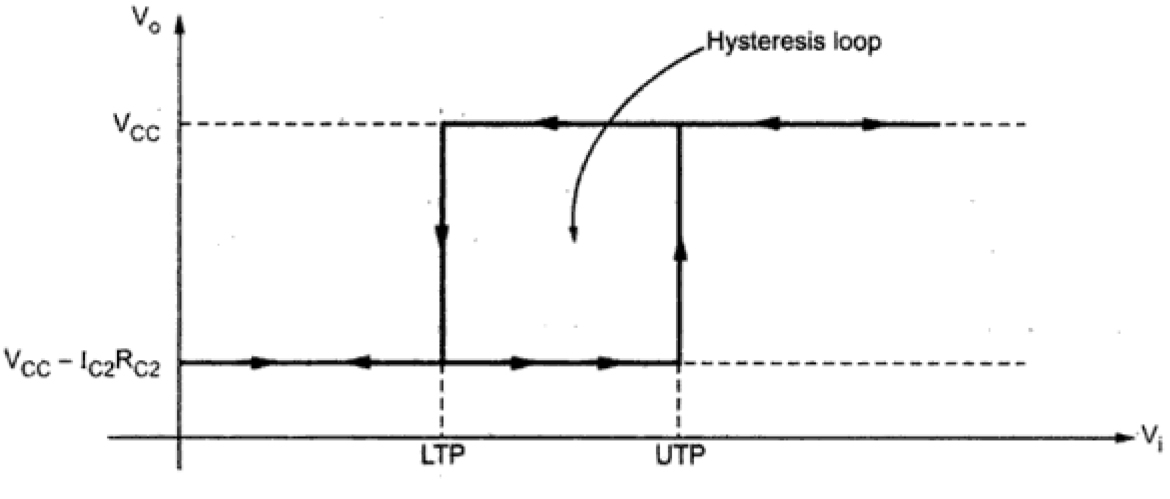
Lower Trigger Point (LTP)

- From fig (c) with Q1 ON , $I_E = (V_i – V_{BE})/R_E$ .Thus a reduction in Vi also reduces IE and since IC~IE, IC1 also becomes small.
- The voltage drop across RC1 is approximately IC1RC1 and the collector voltage of Q1 is VC1=(Vcc –Ic1Rc1).
- Therefore when Vi is reduced IC1 becomes smaller causing Vc1 to rise.
- If Vi is reduced by a very small amount the resultant small increase in VB2 may leave Q2 base still below the level of its emitter voltage.
- Q2 switches ON again only when VB2 and Vi becomes equal.
- The input voltage at which this occurs is known as ***Lower Trigger Point (LTP)***



Schmitt trigger waveforms

- From the above waveforms it is seen than when the input arrives at the upper trigger point the output switches to VCC.
- When he input falls to the lower trigger point the output drops to (Vcc – Ic2Rc2)



HYSTERESIS

- The graph of output voltage against input voltage is called transfer characteristics of the Schmitt trigger.

- It can be seen than once output changes its state, it remains there indefinitely until the input voltage crosses any of the threshold levels.
- So when output changes its state from low to high at UTP, it remains there till input crosses ltp and vice versa.
- This characteristics of Schmitt trigger is called hysteresis.
- It is also called dead band or dead zone as there is no change in this zone, though input changes.
- The loop of the characteristics is called hysteresis loop.
- The difference between UTP and LTP is called the width of hysteresis
- The amount of hysteresis can be changed by changing the values of RC1 and RC2, while the UTP can be increased by increasing the value of RE.

Applications

1. It is used as amplitude comparator.
2. It is used as squaring circuit.
3. It is used for wave shaping circuits.
4. The hysteresis in Schmitt trigger is valuable when conditioning noisy signals for using digital circuits.

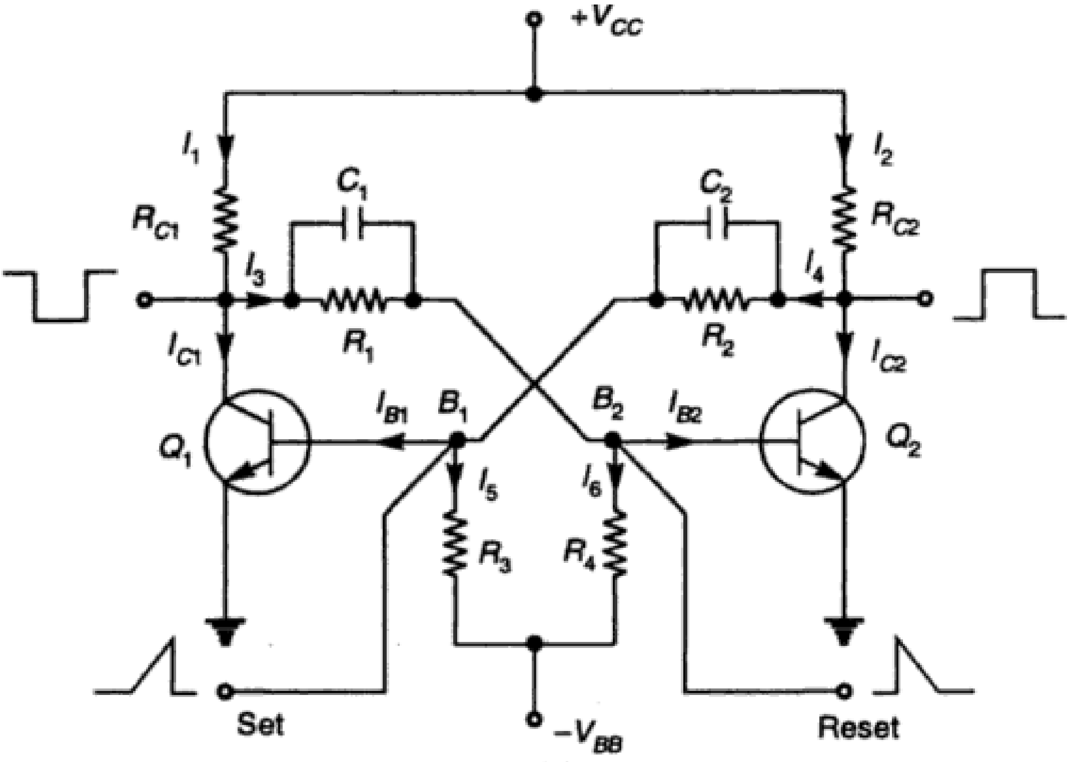
2. DISCUSS ABOUT WORKING OF BISTABLE MULTIVIBRATOR WITH NEAT DIAGRAM. [ ]

Definition:

- The bistable multivibrator is also referred to as a flip flop, Eccles Jordan circuit.
- It has two stable states.
- A trigger pulse is applied to the circuit will cause it to switch from one state to the other. Another trigger pulse is then required to switch the circuit back to its original state.

Circuit Description:

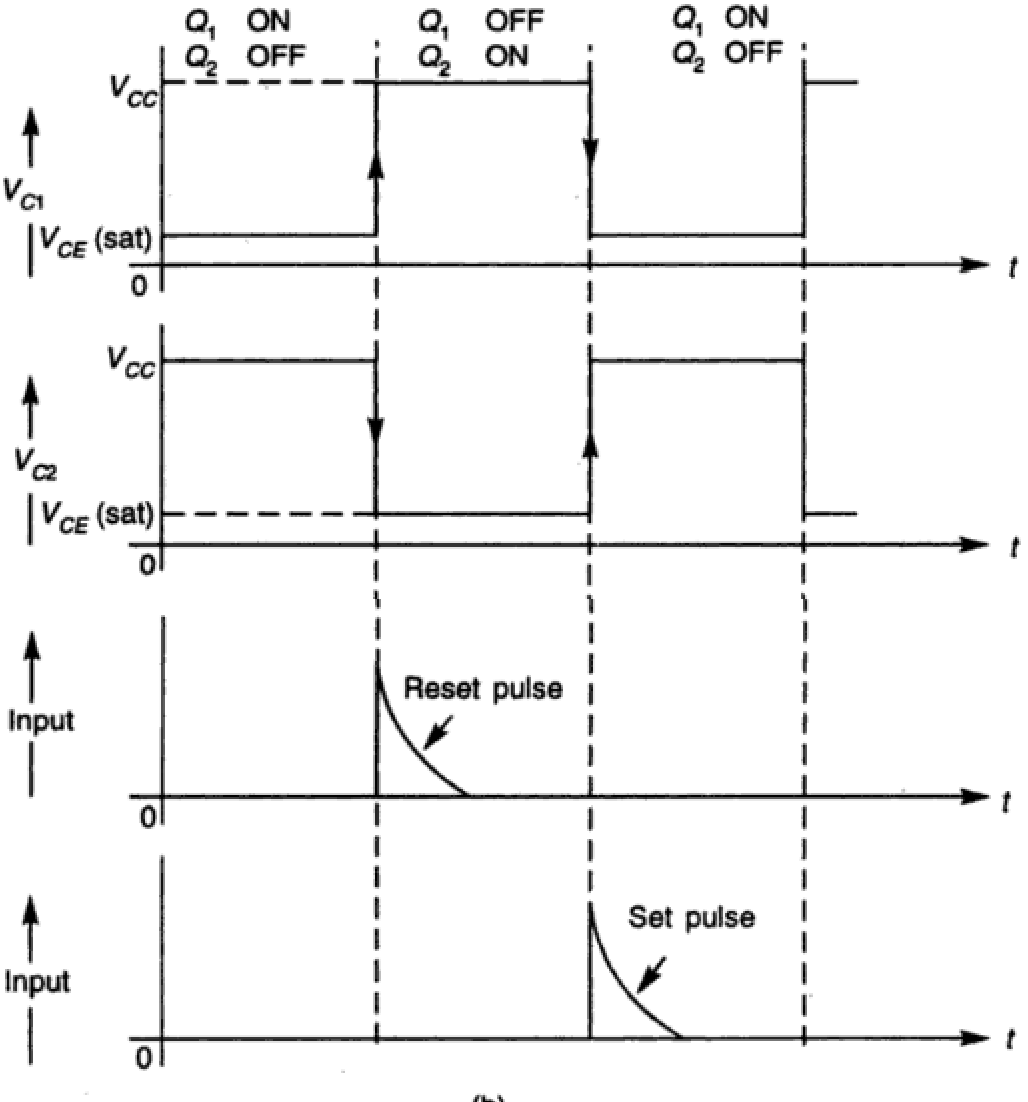
- Figure below shows the circuit of bistable multivibrator using two NPN transistors. In this circuit the output of a transistor Q2 is coupled to the base of transistor Q1 through a resistor R2.
- Similarly the output of Q1 is coupled to the base of Q2 through resistor R1.
- The main purpose of capacitors C1 and C2 is to improve the switching characteristics of the circuit by passing the high frequency components of these square wave pulses.
- This allows fast rise and fall times so that these square waves will not be distorted.
- C1 and C2 are thus called commutating capacitors.



(a)

Circuit Operation :

- When the circuit is first switched on one of the transistor will start conducting more than the other. This transistor is driven into saturation (i.e ON).
- Then because of regenerative feed back action the other transistor is taken into cutoff (i.e OFF).
- Assume that Q1 is ON and Q2 is OFF.
- It is a stable state of the circuit and will remain in this state till a trigger pulse is applied from outside
- A positive triggering pulse applied to the reset input 9base of Q2 increases its forward bias, thereby turning transistor Q2 ON and an increase in collector current and a decrease in collector voltage.
- The fall in collector voltage is coupled to the base of Q1 where it reverse biases the base emitter circuit and Q1 is thus turned OFF..
- The circuit is then in its second stable state and remains so till a positive trigger pulse is applied to set input (base of Q1)
- A similar action can be achieved by applying a negative pulse at the set input for a transition from the first stable state to the second stable state and by applying a negative pulse at the reset input, reverse transition can be obtained



(b)

- Figure above shows the waveforms at the collector of transistor Q1 and Q2 of the bistable multivibrator in response to the trigger pulses applied to the set and rest input.
- It is evident from these waveforms that the output waveforms are the complement of each other.

Applications

1. It is used as memory element in shift registers, counters and so on
2. It is used as frequency divider.
3. It is used to generate square waves of symmetrical shape by sending regular triggering pulse to the input.
4. It is used in processing of pulse type waveforms

3. DISCUSS ABOUT WORKING OF ASTABLE MULTIVIBRATOR WITH NEAT DIAGRAM. [ ]

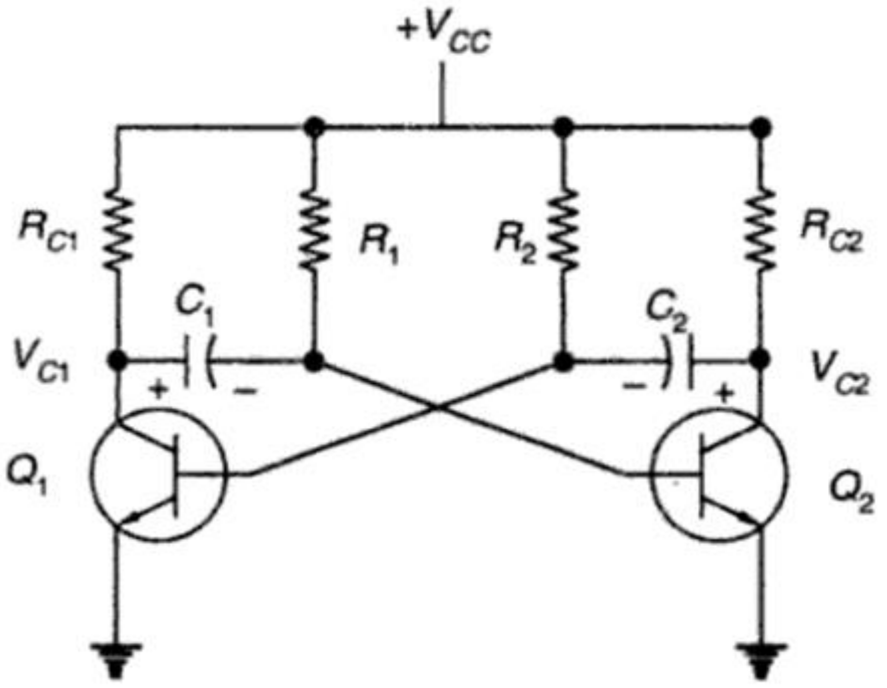
Definition:

- It is a free running multivibrator which generates square wave without any external triggering pulse.
- It has no stable states, i.e it has two quasi stable states.

- It switches back and forth from one state to the other, remaining in each state for a time depending upon the discharging of a capacitive circuit.

Circuit Description:

- Figure below shows the basic symmetric transistor astable multivibrator in which the components in one half of a cycle of the circuit are identical to their counter part in the other half.



(a) Astable multivibrator—circuit diagram

- The square wave output can be taken from collector point of Q1 or Q2.
- The two collector resistance are equal to Rc.$\mathrm{R_{C1}} = \mathrm{R_{C2}} = \frac{\mathrm{V_{CC} - V_{CE(SAT)}}}{\mathrm{I_{C(MAX)}}}$
- The collector Q1is coupled to the base of Q2 through capacitor C2 while the collector of Q2 is coupled to the base of Q1 through capacitor C1.
- The capacitive coupling is used between the stages due to which neither transistor can remain permanently cut off

Circuit Operation :

- When the supply voltage +Vcc is applies one transistor will conduct more than the other due to circuit imbalance.
- Initially let us assume that transistor Q1 is conducting and Q2 is cut off.
- Then Vc1 the output of Q1 is equal to Vce(sat) i.e approximately zero volt and Vc2 =+Vcc.
- At this instant C charges exponentially with a time constant R1C1 towards the supply voltage through R1 and correspondingly VB2 also increases exponentially towards Vcc
- When Vb2 crosses the cut in voltage Q2 starts conducting and Vc2 falls to Vce(sat).
- Also Vb1 falls due to cpacitive coupling between collector of Q2 and base of Q1 therby driving Q1 into OFF state.
- Now the rise in voltage Vc1 is coupled through C1 to the base of Q2 causing a small overshoot in voltage Vb2.Thus Q1 is OFF and Q2 is ON.
- At this instant the voltage levels are

Vb1 is negative

Vc1 = Vcc

Vb2 = Vbe(sat)

Vc2 = Vce(sat)

- When Q1 is OFF and Q2 is ON the voltage Vb1 increases exponentially with a time constant R2C2 towards Vcc.
- Therefore Q1 is driven into saturation and Q2 is cut off .
- Now the voltage levels are

Vb2 is negative

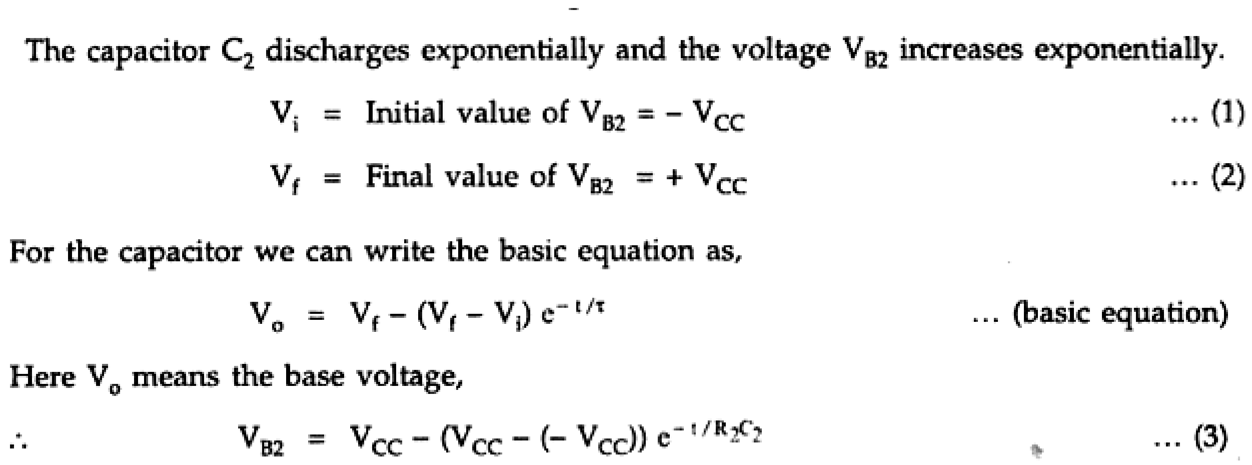
Vc2 = Vcc

Vb1 = Vbe(sat)

Vc1 = Vce(sat)

- It is clear that Q2 is ON, the falling voltage Vc2 permits the discharging of the capacitor C2 which drivesQ1 into cut off.
- The rising voltage of Vc1 feeds back to the base of Q2 tending to turn it ON.
- This process is said to be regenerative

Expression for time period T:



The capacitor C_2 discharges exponentially and the voltage V_{B2} increases exponentially.

$$V_i = \text{Initial value of } V_{B2} = -V_{CC} \quad \ldots (1)$$

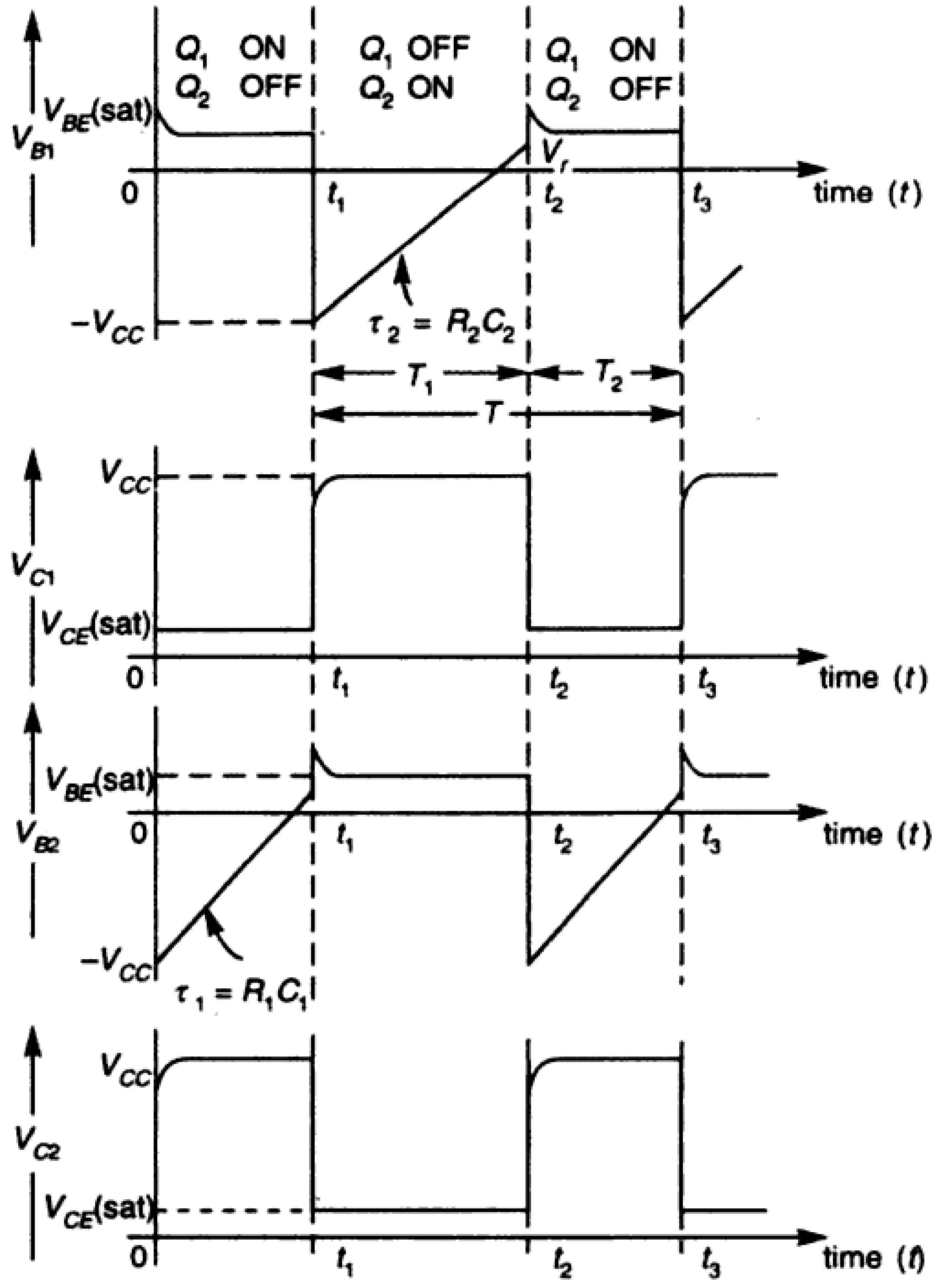
$$V_f = \text{Final value of } V_{B2} = +V_{CC} \quad \ldots (2)$$

For the capacitor we can write the basic equation as,

$$V_o = V_f - (V_f - V_i)\, e^{-t/\tau} \quad \ldots \text{(basic equation)}$$

Here V_o means the base voltage,

$$\therefore \quad V_{B2} = V_{CC} - (V_{CC} - (-V_{CC}))\, e^{-t/R_2C_2} \quad \ldots (3)$$



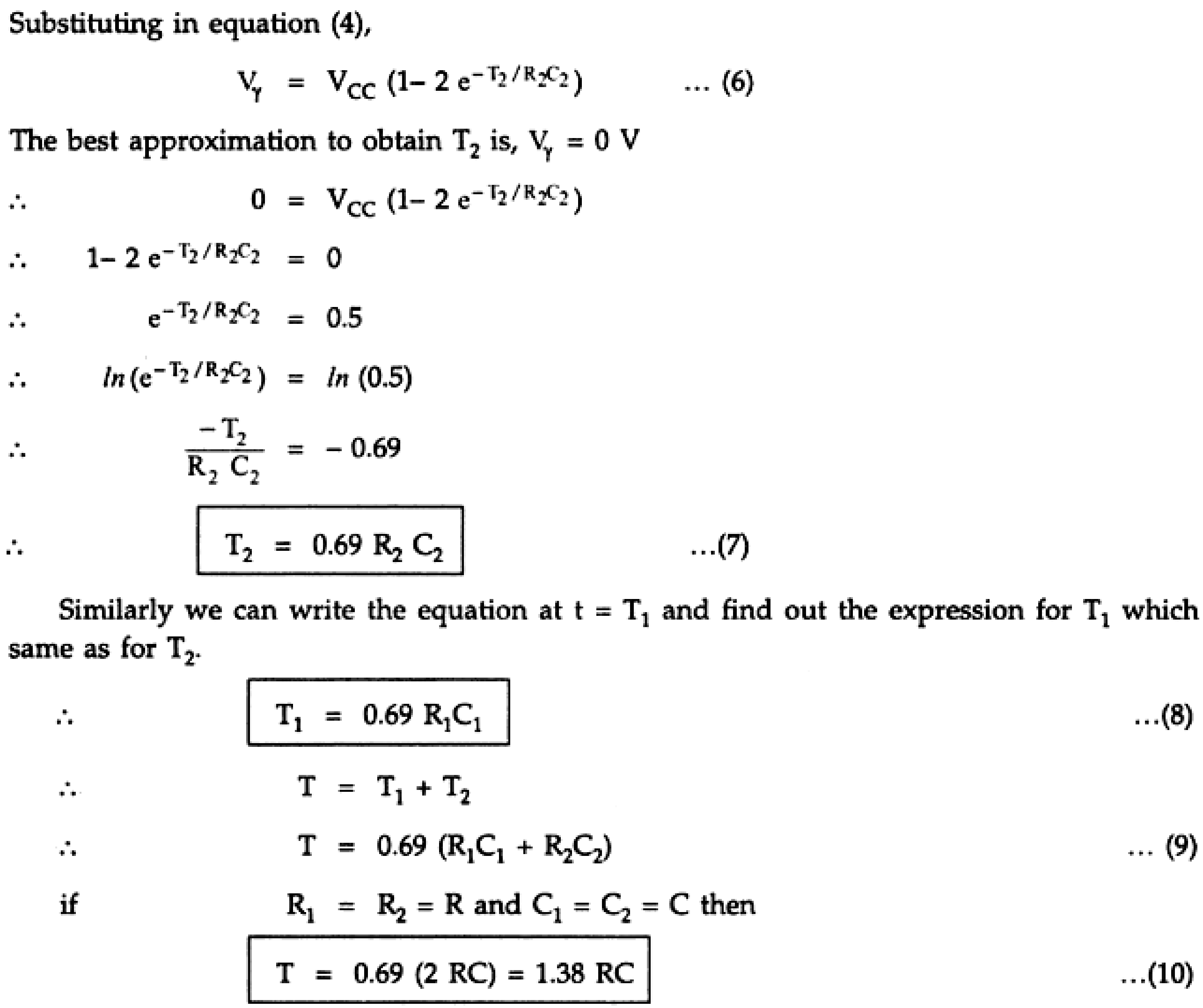
(b) Waveforms at base and collector of Q_1 and Q_2

$$\therefore \quad V_{B2} = V_{CC} - 2\,V_{CC}\,e^{-t/R_2C_2}$$

$$\therefore \quad V_{B2} = V_{CC}\,(1 - 2\,e^{-t/R_2C_2}) \qquad \ldots(4)$$

We know that at switching time,

$$\boxed{t = T_2 \text{ and } V_{B2} = V_\gamma} \qquad \ldots(5)$$



Substituting in equation (4),

$$V_Y = V_{CC}\,(1 - 2\,e^{-T_2/R_2C_2}) \qquad \text{... (6)}$$

The best approximation to obtain T_2 is, $V_Y = 0$ V

$$\therefore \quad 0 = V_{CC}\,(1 - 2\,e^{-T_2/R_2C_2})$$

$$\therefore \quad 1 - 2\,e^{-T_2/R_2C_2} = 0$$

$$\therefore \quad e^{-T_2/R_2C_2} = 0.5$$

$$\therefore \quad ln\,(e^{-T_2/R_2C_2}) = ln\,(0.5)$$

$$\therefore \quad \frac{-T_2}{R_2\,C_2} = -\,0.69$$

$$\therefore \quad \boxed{T_2 = 0.69\,R_2\,C_2} \qquad \text{...(7)}$$

Similarly we can write the equation at $t = T_1$ and find out the expression for T_1 which same as for T_2.

$$\therefore \quad \boxed{T_1 = 0.69\,R_1C_1} \qquad \text{...(8)}$$

$$\therefore \quad T = T_1 + T_2$$

$$\therefore \quad T = 0.69\,(R_1C_1 + R_2C_2) \qquad \text{... (9)}$$

if $R_1 = R_2 = R$ and $C_1 = C_2 = C$ then

$$\boxed{T = 0.69\,(2\,RC) = 1.38\,RC} \qquad \text{...(10)}$$

For asymmetrical configuration

$T=0.69(R_1C_1 + R_2C_2)$

$T_{on}=0.69\ R_1C_1$

$T_{off}=0.69\ R_2C_2$

Applications

1. It is used as square wave generator , voltage to frequency converter and in pulse synchronization as clock for binary logic signals
2. It is used in construction of digital voltmeter and SMPS.
3. It can be used as oscillators over a wide range of audio and radio frequencies.

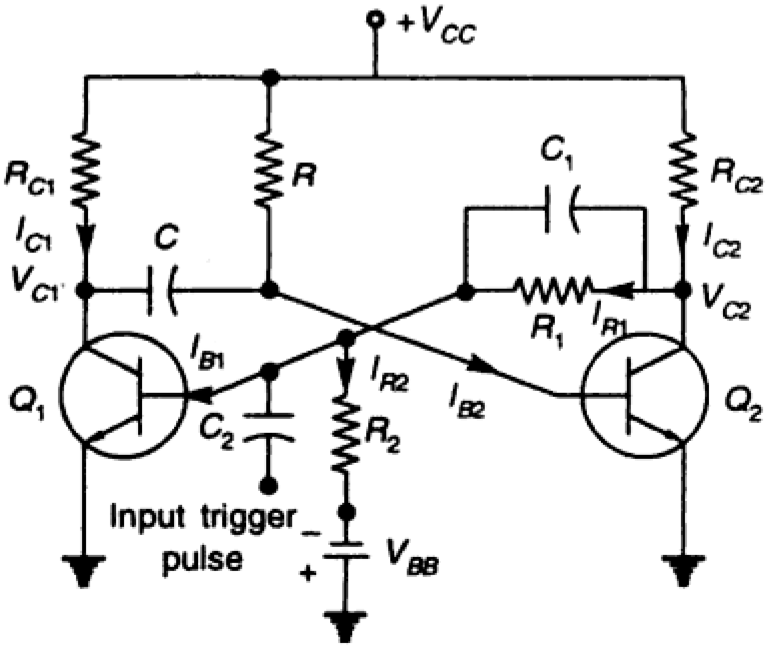
4. DISCUSS ABOUT WORKING OF MONOSTABLE MULTIVIBRATOR WITH NEAT DIAGRAM. [ ]

Definition

- Monostable multivibrator has one stable state and one quasi stable state.
- It is also known as one shot multivibrator. It remains in its stable state until an input pulse triggers it into its quasi stable state for a time duration determined by discharging an RC circuit and the circuit returns to its original stable state.
- It remains there until the next trigger pulse is applied.
- Thus a monostable multivibrator cannot generate square waves of its own like an astable multivibrator.
- Only external trigger pulses will cause it to generate the rectangular waves.

Circuit Description :

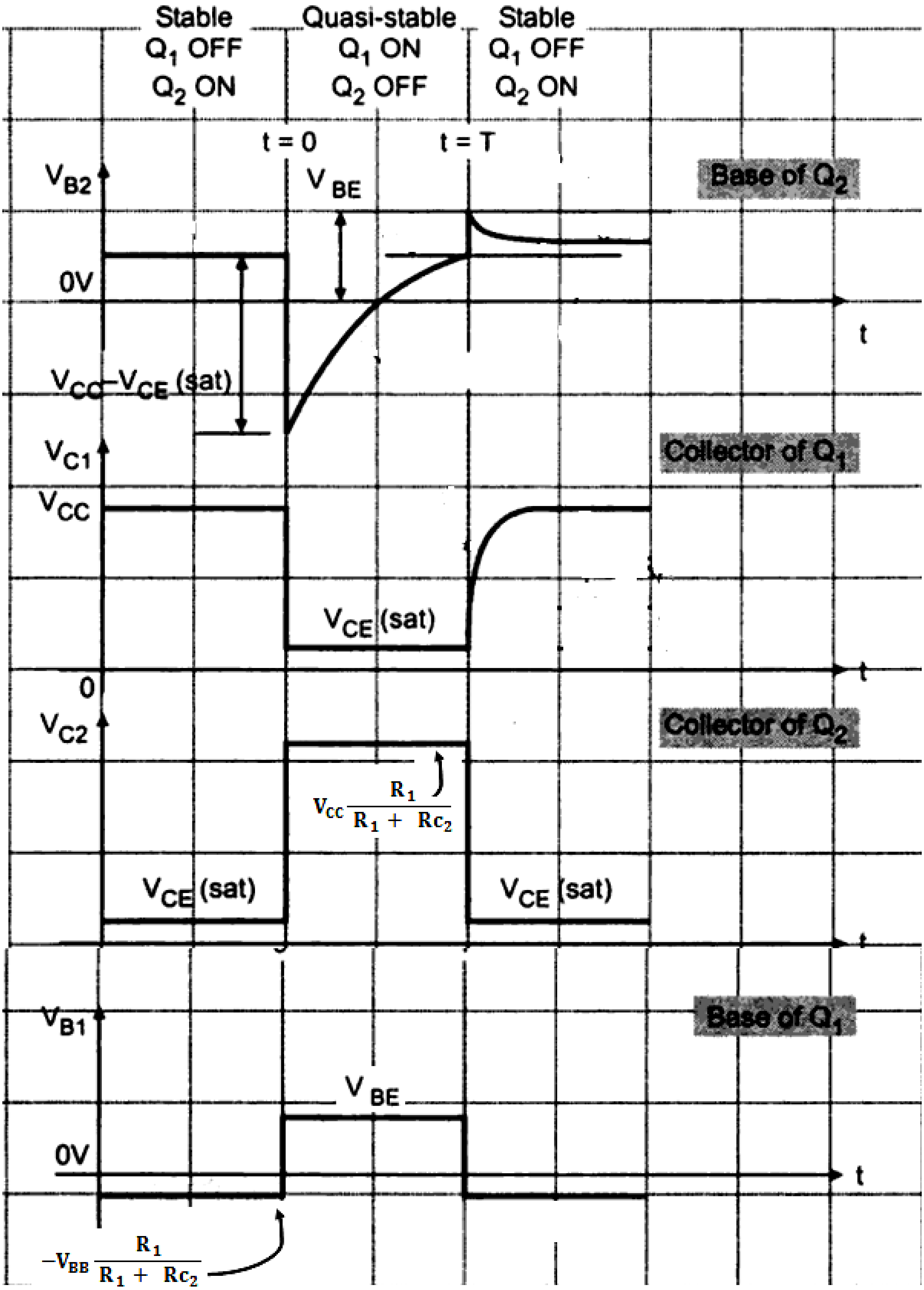
- It consists of two identical transistors Q1and Q2 with equal collector resistance RC1 and RC2.
- The output of Q2 is coupled to the input of base Q1 through a resistive attenuator in which C1 is a small speed ip capacitor to speed up the transition.
- The value of R2 and –Vbb are chosen so as to reverse bias Q1 and keep it in the OFF state.
- The collector supply VCC and R will forward bias Q2 and keep it in the ON state. Actually this is the stable state for the circuit.



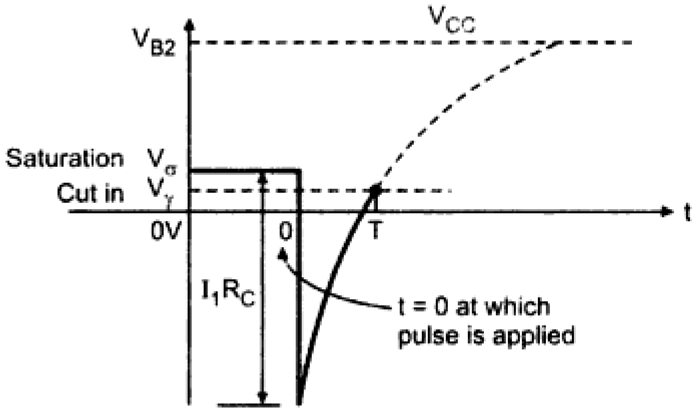
Circuit Operation :

- When a positive trigger pulse of short duration and sufficient magnitude is applied to the base of Q1 through C2, transistor Q1 starts conducting and thereby decreasing the voltage at its collector VC1 which is coupled to the base of Q2 through capacitor C.
- This decreases the forward bias on Q2 and its collector current decreases.
- The increasing positive potential on the collector of Q2 is applied to the base of Q1 through R1.
- This further increases the base potential of Q1 and Q1 is quickly driven to saturation and Q2 is cutoff.
- The capacitor C is charged to approximately to +VCC through the path Vcc, R and Q1.

- As the capacitor C discharges the base of Q2 is forward biased and collector current starts to flow into Q2.
- Thus Q2 is quickly driven to saturation and Q1 is cutoff.
- This is the stable state for the circuit and remains in this condition until another trigger pulse causes the circuit to switch over the states.



Derivation of pulse width:



To write equation for exponential charging of capacitor we can write,

$t = 0^+,\qquad V_i = V_\sigma - I_1 R_C$

$t = \infty,\qquad V_f = V_{CC}$

$$\therefore \quad \boxed{V_C = V_f - (V_i - V_f)\, e^{-t/\tau}} \quad \text{... (basic equation)}$$

where τ = time constant

$$\therefore \quad V_{B2} = V_{CC} - (V_{CC} - V_\sigma + I_1R_C)\ e^{-t/\tau} \qquad \text{... (1)}$$

$$\text{at } t = T, \quad V_{B2} = V_\gamma \qquad \text{... (2)}$$

Substituting (2) into (1) and solving for T we get,

$$\therefore \quad \boxed{T = \tau\, ln\left(\frac{V_{CC} + I_1 R_C - V_\sigma}{V_{CC} - V_\gamma}\right)} \qquad \text{... (3)}$$

Where V_σ = 0.3 V for germanium

= 0.8 V for silicon

When Q_1 is in saturation under quasi-stable state we can write,

$$V_{C1} = V_{CE}(sat) \qquad \text{... (4)}$$

$$\therefore \quad I_1 R_C = V_{CC} - V_{CE}(sat) \qquad \text{... (5)}$$

Substituting in (3),

$$\therefore \quad T = \tau\, ln\left(\frac{V_{CC} + V_{CC} - V_{CE}(sat) - V_{BE}(sat)}{V_{CC} - V_\gamma}\right) \qquad \text{... (6)}$$

This is because, $V_\sigma = V_{BE}(sat)$

This is because, $V_\sigma = V_{BE}(sat)$

$$\therefore \quad T = \tau\, ln\left(\frac{2V_{CC} - V_{CE}\,(sat) - V_{BE}\,(sat)}{V_{CC} - V_\gamma}\right) \qquad \text{... (7)}$$

$$\therefore \qquad T = \tau \ln \left\{ \frac{2\left[V_{CC} - \left(\frac{V_{CE}\,(\text{sat}) + V_{BE}\,(\text{sat})}{2} \right) \right]}{V_{CC} - V_{\gamma}} \right\} \qquad \ldots(8)$$

$$T = 0.69\ RC$$

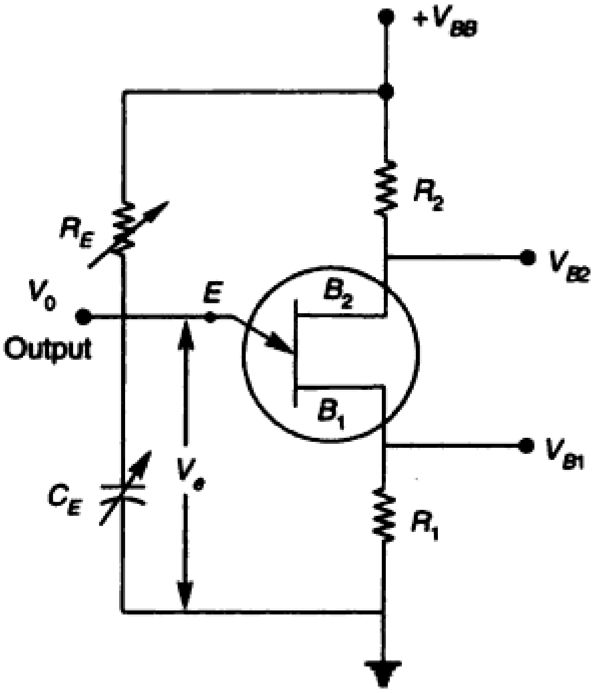
Applications

- It is used to function as an adjustable pulse width generator.
- It is used to generate sharp pulses from the distorted pulses.
- It is used to generate uniform width pulses from a variable width input pulse train.
- It is used as a time delay unit since it produces a transition at a fixed time after the trigger signal.

5. DISCUSS ABOUT WORKING OF UJT RELAXATIONA OSCILLATOR WITH NEAT DIAGRAM. [ ]

Circuit Description & Operation:

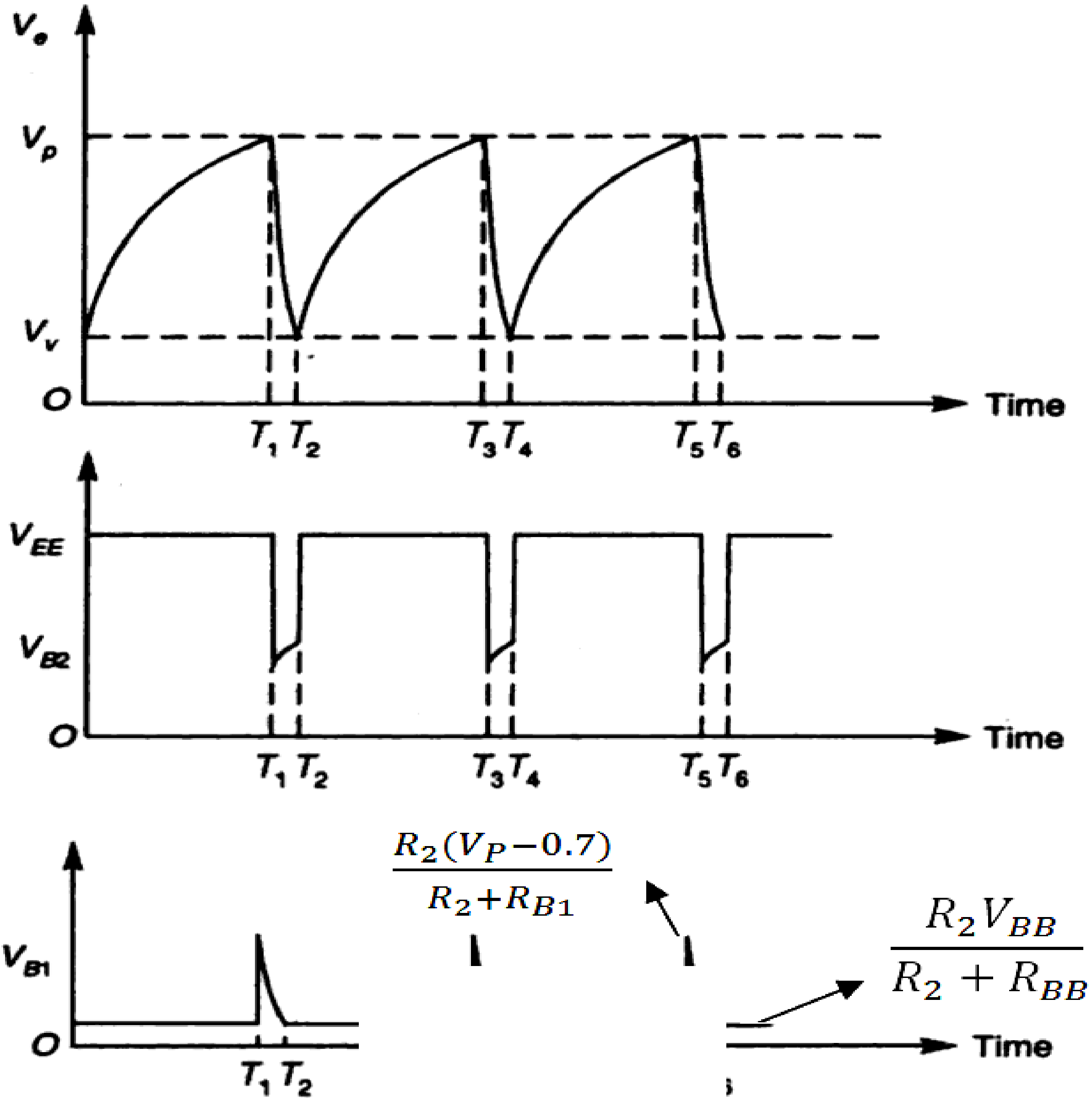
- A unijunction transistor can be used in conjunction with a capacitor and a charging resistor to construct an oscillator with an approximate ramp type output.
- Figure shows the UJT relaxation oscillator in which capacitor C_E charges through R_E as the supply voltage V_{BB} is switched ON.



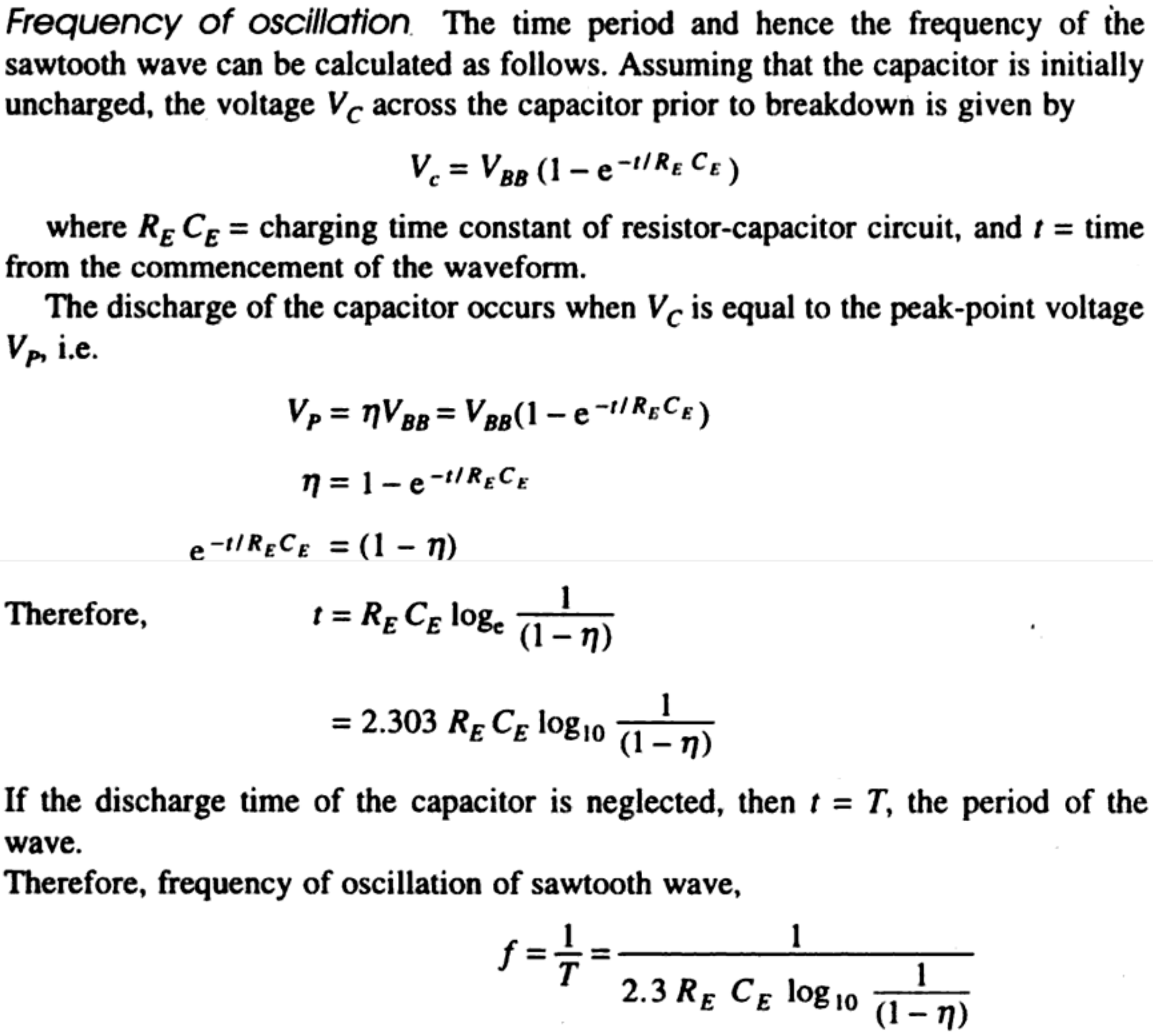
- The voltage across the capacitor increases exponentially and when the capacitor voltage reaches the peak point voltage VP, the UJT starts conducting and the capacitor voltage is discharged rapidly through VB1 and R1.
- After the peak point voltage of UJT is reached, it provides negative resistance to the discharge path which is useful in the working of the relaxation oscillator.
- As the capacitor voltage reaches zero, the devices then cuts off and capacitor C_E starts to charge again.
- This cycle is repeated continuously generating a saw tooth waveform across C_E.

- The inclusion of external resistors R_2 and R_1 in series with B_2 and B_1 provides spike waveforms.
- When the UJT fires, the sudden surge of current through B1 causes drop across R1, which provides going spikes.
- Also at the time of firing, fall of V_{EB1} causes I_2 to increase rapidly which generates negative going spike across R_2
- In the design of UJT relaxation oscillator the charging resistor R_E must be selected between certain upper and lower limits.
- Resistance R_E must not be so large that the emitter current is less than the peak point current I_P when V_V is at the firing voltage otherwise the device may not switch ON.
- If R_E is very small then when V_{EB1} is at the emitter saturation level a current greater than the valley point current I_V might flow into emitter terminal. In this case the UJT may not switch off.
- Thus for correct UJT operation R_E must be selected between two limits that allow the emitter current to be a minimum of I_P and a maximum of I_V.

$$\frac{V_{BB} - V_V}{I_V} < R_E < \frac{V_{BB} - V_P}{I_P}$$



Frequency of oscillation:



Frequency of oscillation. **The time period and hence the frequency of the sawtooth wave can be calculated as follows. Assuming that the capacitor is initially uncharged, the voltage V_C across the capacitor prior to breakdown is given by**

$$V_c = V_{BB}\,(1 - e^{-t/R_E C_E})$$

where $R_E C_E$ = charging time constant of resistor-capacitor circuit, and t = time from the commencement of the waveform.

The discharge of the capacitor occurs when V_C is equal to the peak-point voltage V_P, i.e.

$$V_P = \eta V_{BB} = V_{BB}(1 - e^{-t/R_E C_E})$$

$$\eta = 1 - e^{-t/R_E C_E}$$

$$e^{-t/R_E C_E} = (1 - \eta)$$

Therefore,

$$t = R_E\,C_E \log_e \frac{1}{(1-\eta)}$$

$$= 2.303\ R_E\,C_E \log_{10} \frac{1}{(1-\eta)}$$

If the discharge time of the capacitor is neglected, then $t = T$, the period of the wave.

Therefore, frequency of oscillation of sawtooth wave,

$$f = \frac{1}{T} = \frac{1}{2.3\ R_E\ C_E\ \log_{10} \dfrac{1}{(1-\eta)}}$$

Applications

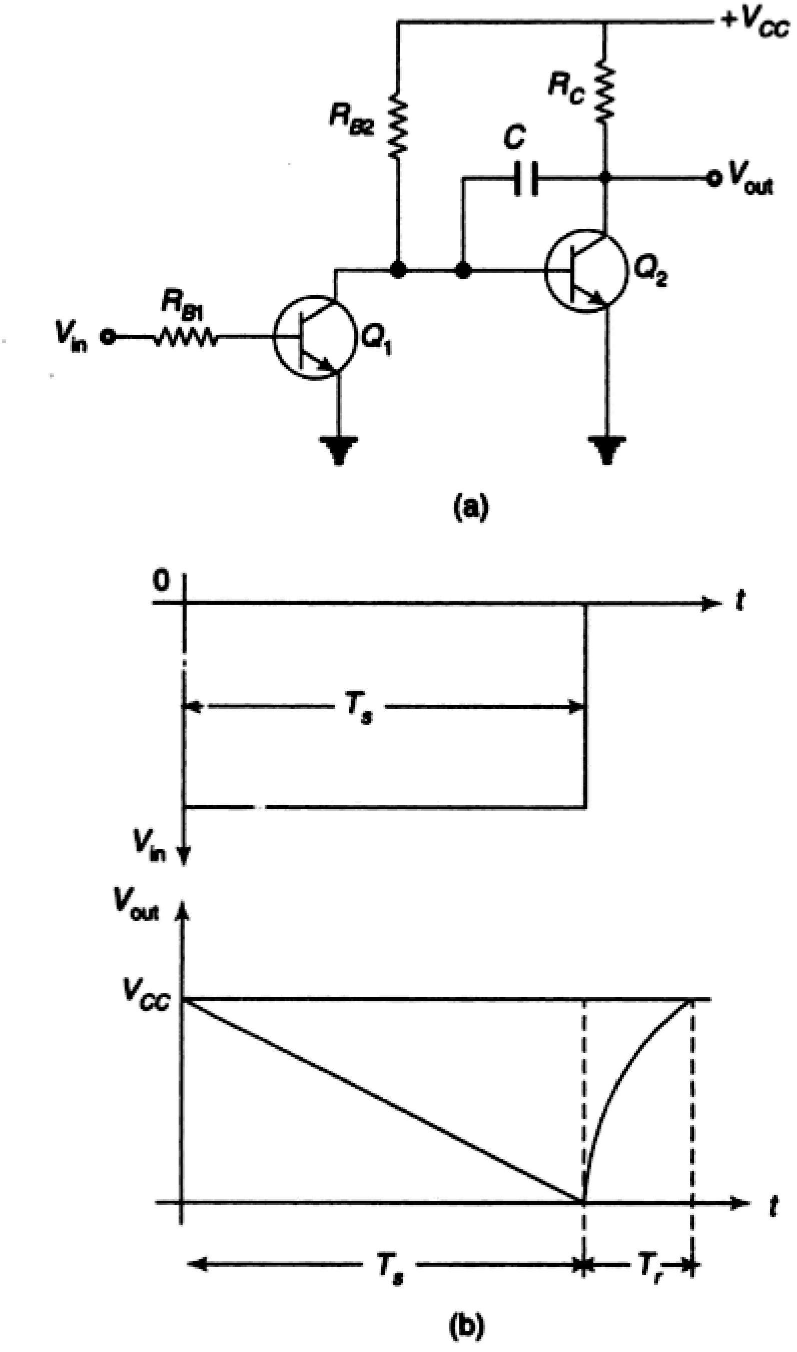
1. It is used for triggering the device for SCR and TRIAC.
2. It is also used in timing circuits, saw tooth generator, simple oscillators and phase control.

6. DISCUSS ABOUT WORKING OF MILLER INTEGRATOR CIRCUIT []

Introduction

In the miller circuit, an operational integrator is used to convert a step into ramp waveform.

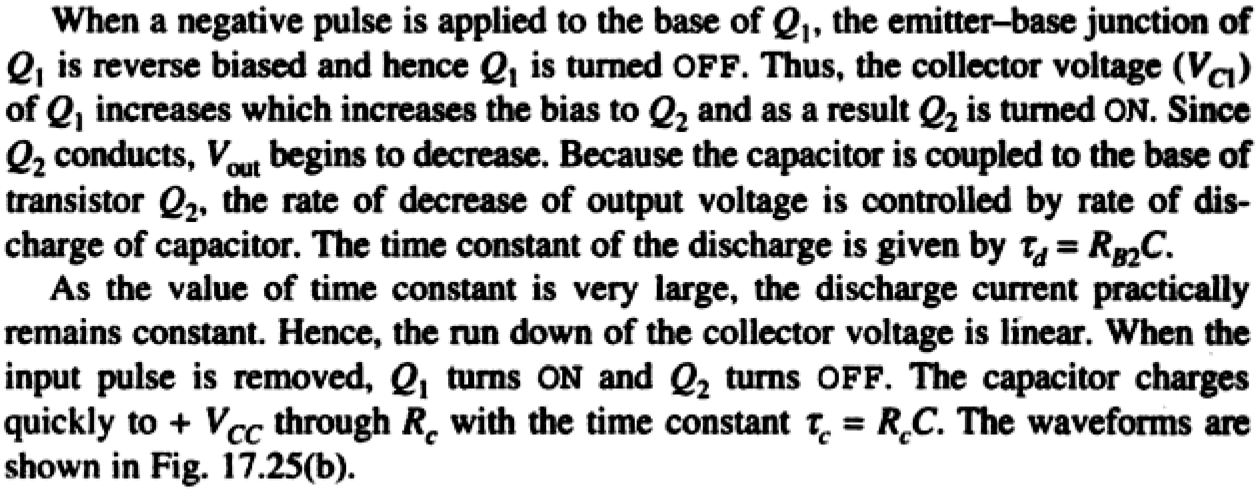
Miller integrator Figure 17.25(a) shows the circuit of a Miller integrator or a sweep circuit. Transistor Q_1 acts as a switch and transistor Q_2 is a common-emitter amplifier, i.e. a high gain amplifier. Consider the case when Q_1 is ON and Q_2 is OFF. At this condition, the voltage across the capacitor C and the output voltage V_o is equal to V_{CC}.



(a)

(b)

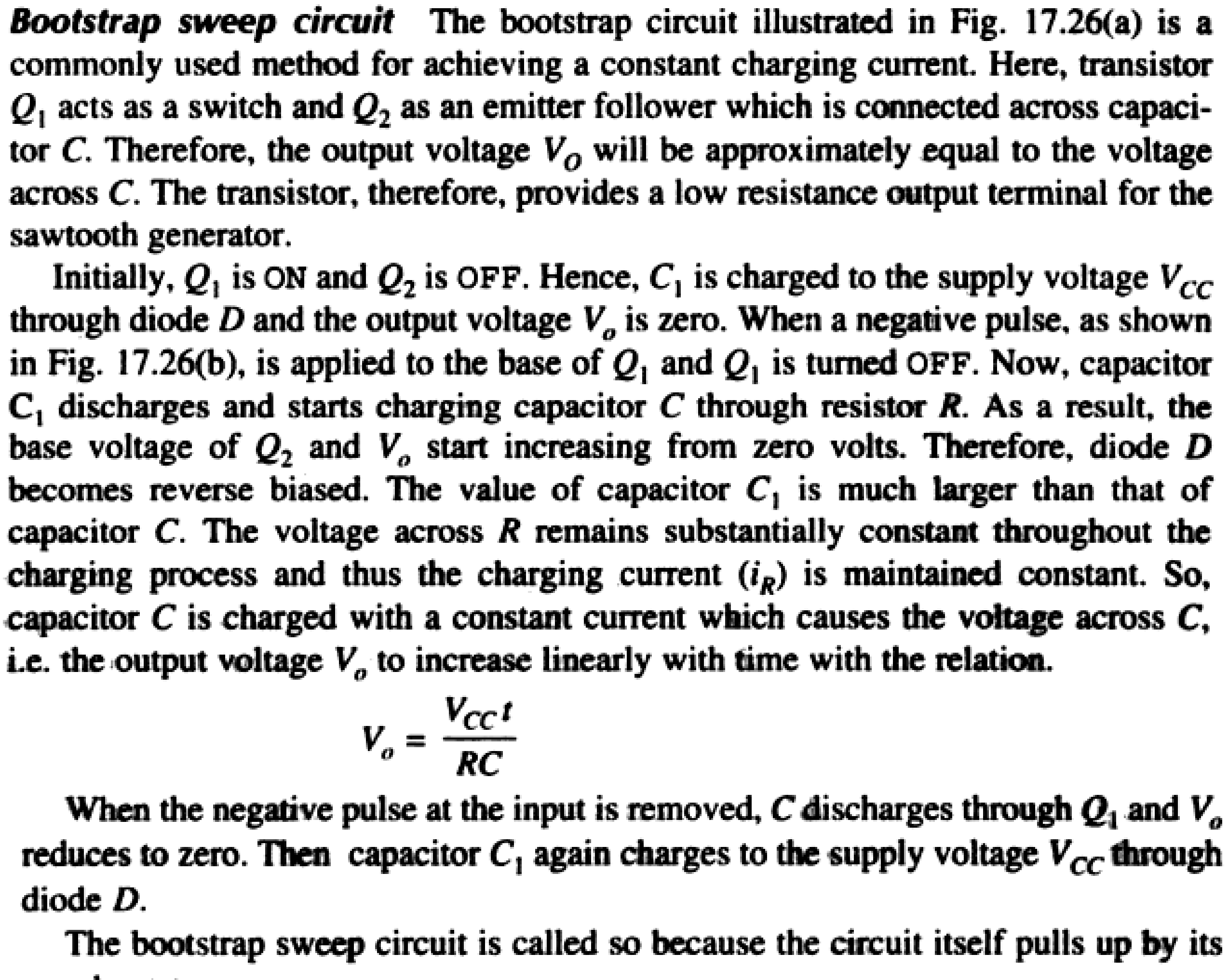
Fig. 17.25 Miller integrator: (a) Circuit diagram, and (b) Input and output waveforms



When a negative pulse is applied to the base of Q_1, the emitter–base junction of Q_1 is reverse biased and hence Q_1 is turned OFF. Thus, the collector voltage (V_{C1}) of Q_1 increases which increases the bias to Q_2 and as a result Q_2 is turned ON. Since Q_2 conducts, V_{out} begins to decrease. Because the capacitor is coupled to the base of transistor Q_2, the rate of decrease of output voltage is controlled by rate of discharge of capacitor. The time constant of the discharge is given by $\tau_d = R_{B2}C$.

As the value of time constant is very large, the discharge current practically remains constant. Hence, the run down of the collector voltage is linear. When the input pulse is removed, Q_1 turns ON and Q_2 turns OFF. The capacitor charges quickly to + V_{CC} through R_c with the time constant $\tau_c = R_cC$. The waveforms are shown in Fig. 17.25(b).

7. DISCUSS ABOUT WORKING OF BOOTSTRAP SWEEP CIRCUIT []



Bootstrap sweep circuit The bootstrap circuit illustrated in Fig. 17.26(a) is a commonly used method for achieving a constant charging current. Here, transistor Q_1 acts as a switch and Q_2 as an emitter follower which is connected across capacitor C. Therefore, the output voltage V_O will be approximately equal to the voltage across C. The transistor, therefore, provides a low resistance output terminal for the sawtooth generator.

Initially, Q_1 is ON and Q_2 is OFF. Hence, C_1 is charged to the supply voltage V_{CC} through diode D and the output voltage V_o is zero. When a negative pulse, as shown in Fig. 17.26(b), is applied to the base of Q_1 and Q_1 is turned OFF. Now, capacitor C$_1$ discharges and starts charging capacitor C through resistor R. As a result, the base voltage of Q_2 and V_o start increasing from zero volts. Therefore, diode D becomes reverse biased. The value of capacitor C_1 is much larger than that of capacitor C. The voltage across R remains substantially constant throughout the charging process and thus the charging current (i_R) is maintained constant. So, capacitor C is charged with a constant current which causes the voltage across C, i.e. the output voltage V_o to increase linearly with time with the relation.

$$V_o = \frac{V_{CC}\,t}{RC}$$

When the negative pulse at the input is removed, C discharges through Q_1 and V_o reduces to zero. Then capacitor C_1 again charges to the supply voltage V_{CC} through diode D.

The bootstrap sweep circuit is called so because the circuit itself pulls up by its own bootstrap.

8. EXPLAIN IN DETAIL ABOUT NEGATIVE RESISTANCE OSCILLATORS IN DETAIL.

All oscillators do not require positive feedback for their operation. If the positive resistance of the LC tank circuit is cancelled by introducing the right amount of negative resistance across the tank circuit, then the steady oscillation can be maintained. There are several devices such as dynatron, transitron, UJT and tunnel diode that exhibit a region of negative resistance within the *V–I* characteristics as shown in Fig. 15.22. Such devices operated in the negative resistance region are placed across a high *Q* parallel LC circuit as the frequency determining section. For oscillation to occur the negative resistance should be numerically less than the dynamic resistance of the tuned circuit.

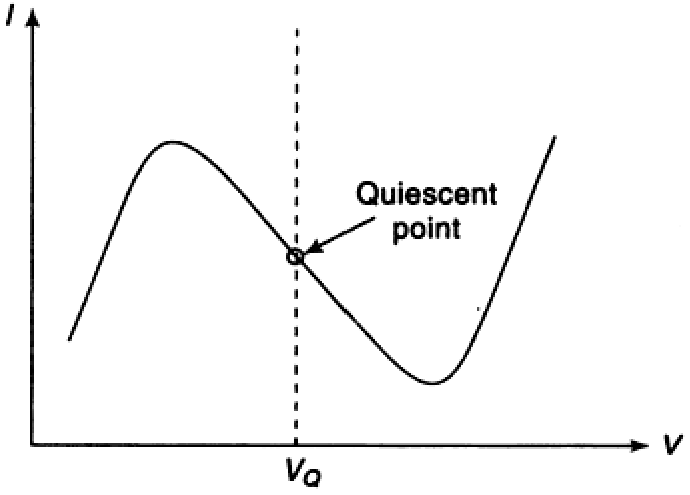


Fig. 15.22 *V–I* characteristics of negative resistance oscillators

Tunnel diode oscillator Figure 15.23 shows the tunnel diode oscillator. If the parallel tank circuit with a resistance *R*, a capacitance *C* and an inductance *L* is connected across the tunnel diode whose negative resistance is – R_n, the net resistance R_{eq} represents *R* and $-R_n$ in parallel and is given by

$$R_{eq} = \frac{-R_n \cdot R}{R - R_n}$$

If $R > R_n$, then R_{eq} is negative and oscillations can build-up. The oscillation amplitude then grows until it occupies a voltage range greater than the extent of the negative resistance region of the characteristics. When the operating point enters the region of positive resistance, the amplitude of oscillation is limited. To obtain the maximum output, the quiescent point must be accurately located at the center of the negative resistance region. The frequency of oscillation is given by

$$f_o = \frac{1}{2\pi\sqrt{LC}}$$

A tunnel diode has a characteristic with a negative resistance region between voltages of approximately 0.1 and 0.3 V and can be used as an oscillator at frequencies up to 100 GHz.

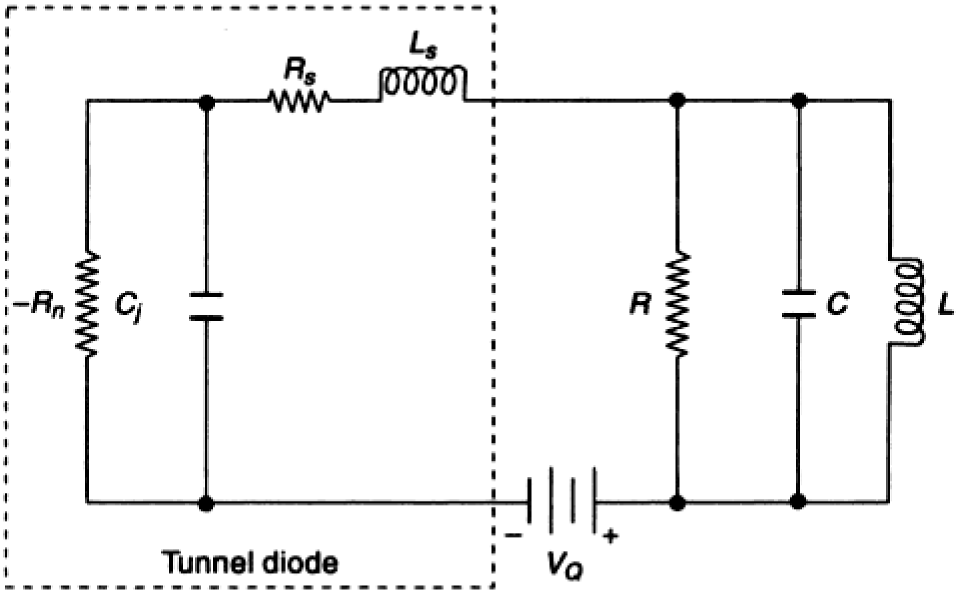


Fig. 15.23 Tunnel diode oscillator

9. EXPLAIN IN DETAIL ABOUT FIXED AMPLITUDE SWEEP.

An excellent method[4] of obtaining a fixed-amplitude time base is to replace the monostable gate with a bistable circuit and to add a feedback loop from the output back to the input of the binary via the "hold-off circuit,"

The gate generator is an emitter-coupled binary (a Schmitt circuit) with a large hysteresis voltage V_H (Sec. 10-11). Initially transistor $Q5$ is cut off and diode $D2$ is conducting, so that the stability-control P may be used to

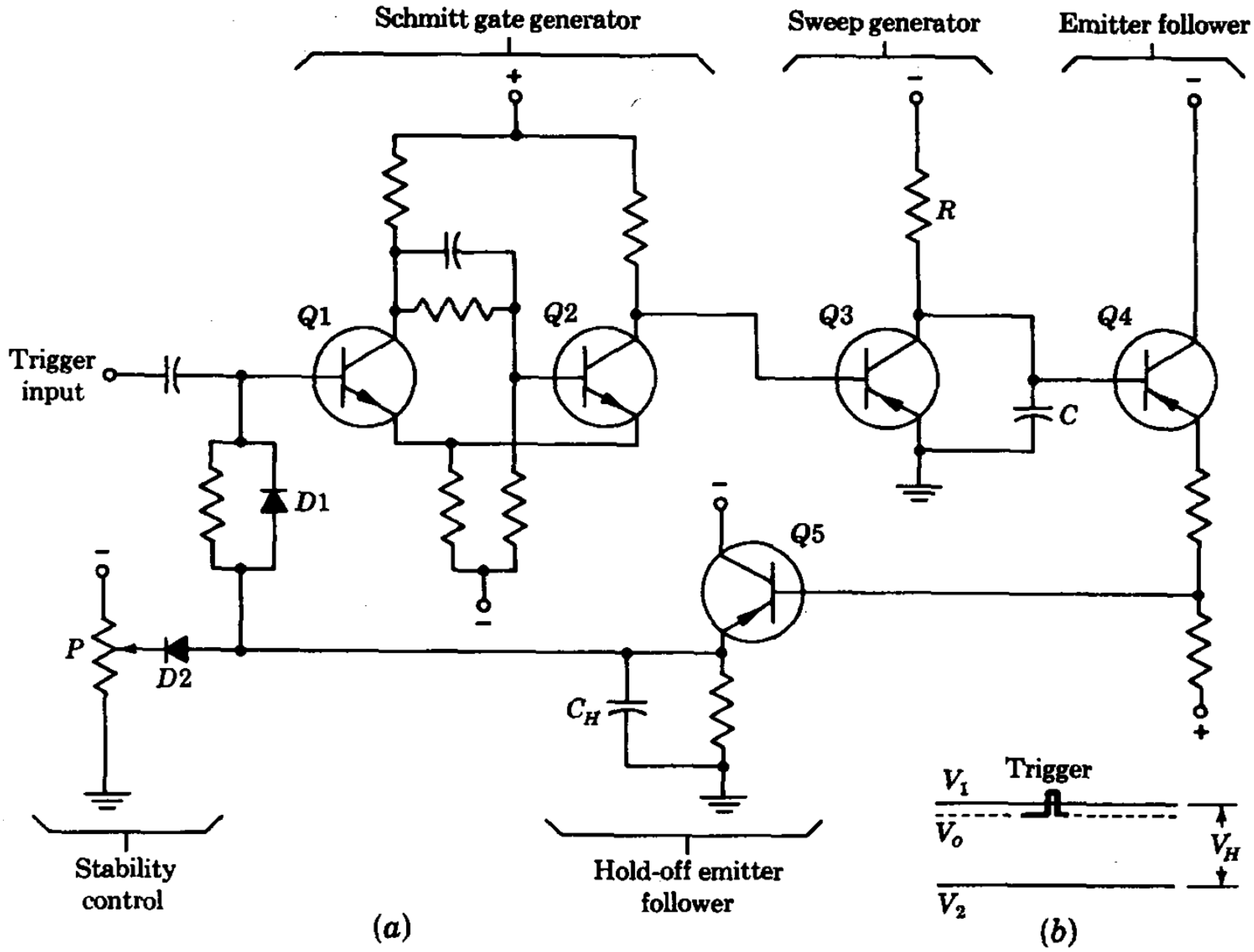
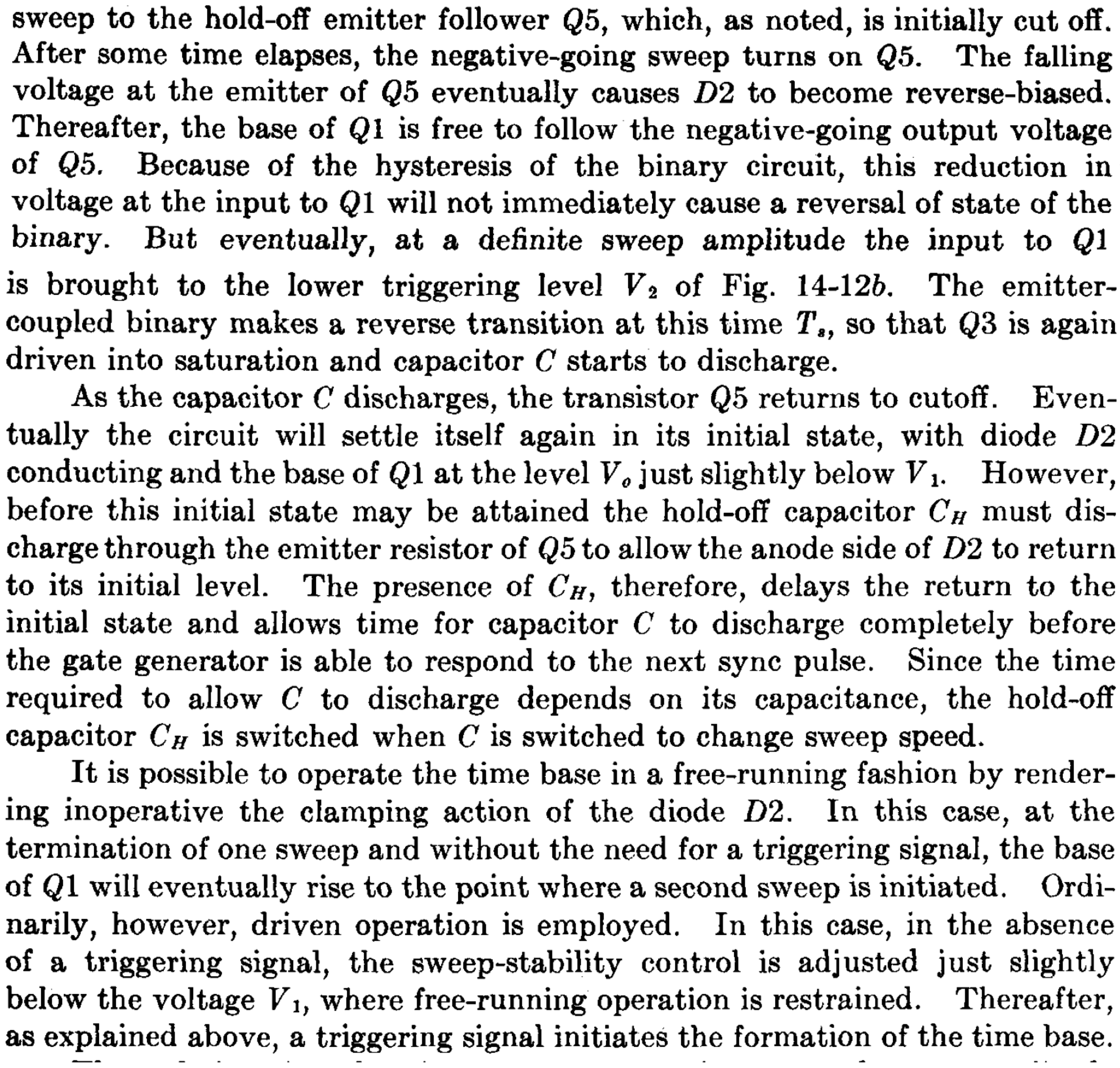


Fig. 14-12 (a) A technique for generating a time-base waveform of fixed amplitude. (b) The Schmitt critical levels are V_1 and V_2. The base of $Q1$ is initially at V_o. An input trigger causes a transition, and then the voltage fed back from the sweep drives the base of $Q1$ toward V_2, where the reverse transition takes place.

adjust the bias of $Q1$ to be V_o just below the upper triggering which is level V_1, as indicated in Fig. 14-12*b*. Hence, $Q1$ is OFF, $Q2$ is ON, and $Q3$ is in saturation, so that the voltage across the sweep capacitor C is clamped to V_{CE}(sat). The trigger input waveform consists of positive and negative sync pulses (obtained by shaping the input signal). A negative pulse is bypassed by $D1$ but a positive signal larger than $V_1 - V_o$ causes a transition in the binary. The resulting positive step at $Q2$ reverse-biases $Q3$ and initiates a negative-going sweep voltage across C. The emitter follower $Q4$ is used for the sake of its high input impedance. This emitter follower transmits the



sweep to the hold-off emitter follower $Q5$, which, as noted, is initially cut off. After some time elapses, the negative-going sweep turns on $Q5$. The falling voltage at the emitter of $Q5$ eventually causes $D2$ to become reverse-biased. Thereafter, the base of $Q1$ is free to follow the negative-going output voltage of $Q5$. Because of the hysteresis of the binary circuit, this reduction in voltage at the input to $Q1$ will not immediately cause a reversal of state of the binary. But eventually, at a definite sweep amplitude the input to $Q1$ is brought to the lower triggering level V_2 of Fig. 14-12*b*. The emitter-coupled binary makes a reverse transition at this time T_s, so that $Q3$ is again driven into saturation and capacitor C starts to discharge.

As the capacitor C discharges, the transistor $Q5$ returns to cutoff. Eventually the circuit will settle itself again in its initial state, with diode $D2$ conducting and the base of $Q1$ at the level V_o just slightly below V_1. However, before this initial state may be attained the hold-off capacitor C_H must discharge through the emitter resistor of $Q5$ to allow the anode side of $D2$ to return to its initial level. The presence of C_H, therefore, delays the return to the initial state and allows time for capacitor C to discharge completely before the gate generator is able to respond to the next sync pulse. Since the time required to allow C to discharge depends on its capacitance, the hold-off capacitor C_H is switched when C is switched to change sweep speed.

It is possible to operate the time base in a free-running fashion by rendering inoperative the clamping action of the diode $D2$. In this case, at the termination of one sweep and without the need for a triggering signal, the base of $Q1$ will eventually rise to the point where a second sweep is initiated. Ordinarily, however, driven operation is employed. In this case, in the absence of a triggering signal, the sweep-stability control is adjusted just slightly below the voltage V_1, where free-running operation is restrained. Thereafter, as explained above, a triggering signal initiates the formation of the time base.

10.EXPLAIN IN DETAIL ABOUT CONSTANT CURRENT SWEEP.

Except for very small values of collector-to-base voltage, the collector current of a transistor *in the common-base configuration* is very nearly constant when the emitter current is held fixed (Fig. 6-12). We may use this characteristic to generate a quite linear sweep by causing a constant current to flow into a capacitor. In the circuit of Fig. 14-13*a*, if V_{EB} is the emitter-to-base voltage, the emitter current is

$$I_E = \frac{V_{EE} - V_{EB}}{R_e} \tag{14-17}$$

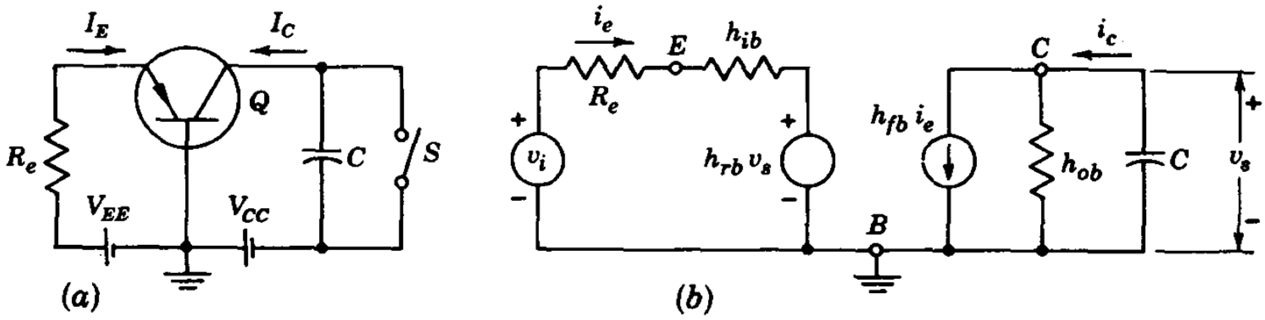


Fig. 14-13 (a) The constant collector current of a CB transistor is used to charge a capacitor linearly; (b) the small-signal model from which to calculate the slope error ($v_i = V_{EE} - V_\gamma \equiv V_i$).

Assuming, for the moment, that V_{EB} remains constant with time after the switch S is opened, then the collector current will be a constant whose nominal value is $I_C = h_{FB} I_E \approx -\alpha I_E$, and the capacitor will charge linearly with time.

To inquire into the sources of nonlinearity we shall replace the transistor by an equivalent circuit using CB hybrid parameters (Sec. 1-3). This model is valid only for changes from the quiescent values. Hence, let us define the initial condition as one where the transistor is just brought to the point of conduction by an emitter threshold bias of V_γ (Sec. 6-17) and a collector voltage V_{CC}. The equivalent circuit from which to determine the sweep voltage v_s is indicated in Fig. 14-13*b*, where the effective input signal v_i (the change from the quiescent value) is $v_i = V_{EE} - V_\gamma \equiv V_i$. Since only variations from the initial condition are under consideration, then the supply V_{CC} is replaced by its internal impedance (assumed to be negligible).

Kirchhoff's voltage law applied to the input mesh and KCL applied to the output node of Fig. 14-13*b* yield, respectively,

$$v_i = i_e(R_e + h_{ib}) + h_{rb}v_s = V_i \tag{14-18}$$

$$i_c = i_e h_{fb} + h_{ob}v_s = -C\frac{dv_s}{dt} \tag{14-19}$$

Subject to the initial condition that $v_s = 0$ at $t = 0$, the solution to these equations is

$$v_s = \frac{\alpha\tau V_i}{C(R_e + h_{ib})}(1 - \epsilon^{-t/\tau}) \tag{14-20}$$

where $\alpha = -h_{fb}$, $V_i = V_{EE} - V_\gamma$, and

$$\frac{1}{\tau} = \frac{1}{C}\left(h_{ob} + \frac{\alpha h_{rb}}{R_e + h_{ib}}\right) \tag{14-21}$$

Expanding the exponential into a power series in t/τ and retaining only the first term,

$$v_s = \frac{\alpha V_i t}{C(R_e + h_{ib})} \tag{14-22}$$

This result is consistent with Fig. 14-13*b*, which gives an emitter current of $i_e = V_i/(R_e + h_{ib})$ at $t = 0+$ and a short-circuit collector current α times as large. If the capacitor current were to remain constant at this value, then $v_s = \alpha i_e t/C$, in agreement with Eq. (14-22).

The sweep amplitude V_s is obtained from Eq. (14-22) with $t = T_s$. The slope error, given by Eq. (14-9), is

$$e_s = \frac{T_s}{\tau} = \frac{V_s}{V_i}\left[h_{rb} + \frac{h_{ob}}{\alpha}(R_e + h_{ib})\right] \tag{14-23}$$

where use was made of Eqs. (14-21) and (14-22). The generator $h_{rb}v_s$, representing the reaction of the collector voltage on the input circuit, causes the emitter current to change as the sweep forms. The first term in Eq. (14-23) results from this change in emitter current. The fact that the collector current is not precisely constant (even for constant emitter current) because of the shunting effect of h_{ob} gives rise to a slope error whose value is given by the second term in Eq. (14-23).

srivenkateshwaraa
College of Engineering & Technology
(Approved by AICTE, New Delhi & Affiliated to Pondicherry University, Puducherry)
13-A, Pondy - Villupuram Main Road, Ariyur, Puducherry - 605 102.
ASPIRE TO EXCEL

ISO 21001

UNIT – III COMBINATIONAL CIRCUITS

Binary arithmetic –BCD addition and subtraction–Code converters-Parity generator–Binary to BCD and BCD to binary conversions–Design of combination circuits using NAND and NOR gates–Design of encoders, decoders, multiplexers, demultiplexer – Serial adders–Binary multiplier – Simplification of k-map, Flip-Flops: RS, D, JKand T types.

11 MARKS

1. DISCUSS ABOUT VARIOUS NUMBER SYSTEMS WITH EXAMPLES AND PERFORM CONVERSTIONS. [ ]

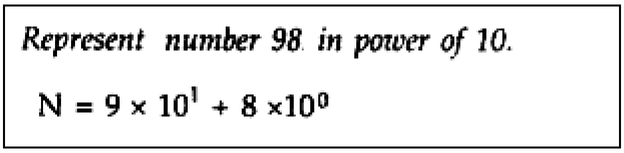
Definition:

- Number system is a basic for counting various items. It is of following types
  - ✓ Decimal number system
  - ✓ Binary number system
  - ✓ Octal system
  - ✓ Hexa decimal system

Decimal number system

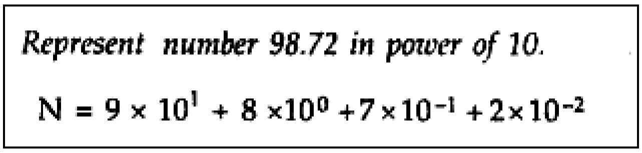
- The decimal number system has ten digits, 0 to 9.
- Since there are ten distinct digits used in decimal system therefore the decimal number system has a base (or radix) of 10.
- The weights for whole numbers are positive power of ten that increases from right to left beginning with 10^0 =1
  ………10^4,10^3,10^2,10^1,10^0
- For fractional numbers the weights are negative power of 10 that decrease from left to right beginning with 10^{-1}
  ………10^4,10^3,10^2,10^1,10^0**.**10^{-1},10^{-2},10^{-3}………

Decimal point



Represent number 98 in power of 10.

$N = 9 \times 10^1 + 8 \times 10^0$



Represent number 98.72 in power of 10.

$N = 9 \times 10^1 + 8 \times 10^0 + 7 \times 10^{-1} + 2 \times 10^{-2}$

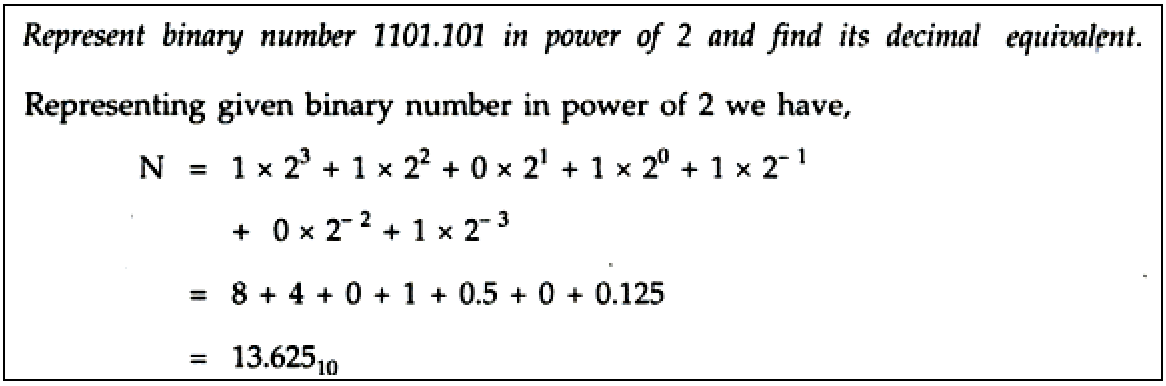
Binary number system

- The binary number system is less complicated than decimal. The two binary digits are 0 and 1.
- Here the base is 2. The weightage is given as

  ………$2^4,2^3,2^2,2^1,2^0\bullet 2^{-1},2^{-2},2^{-3}$………

  Binary point

- Right most bit is called Least significant bit [LSB] and left most bit is called most significant bit [MSB]



Represent binary number 1101.101 in power of 2 and find its decimal equivalent.

Representing given binary number in power of 2 we have,

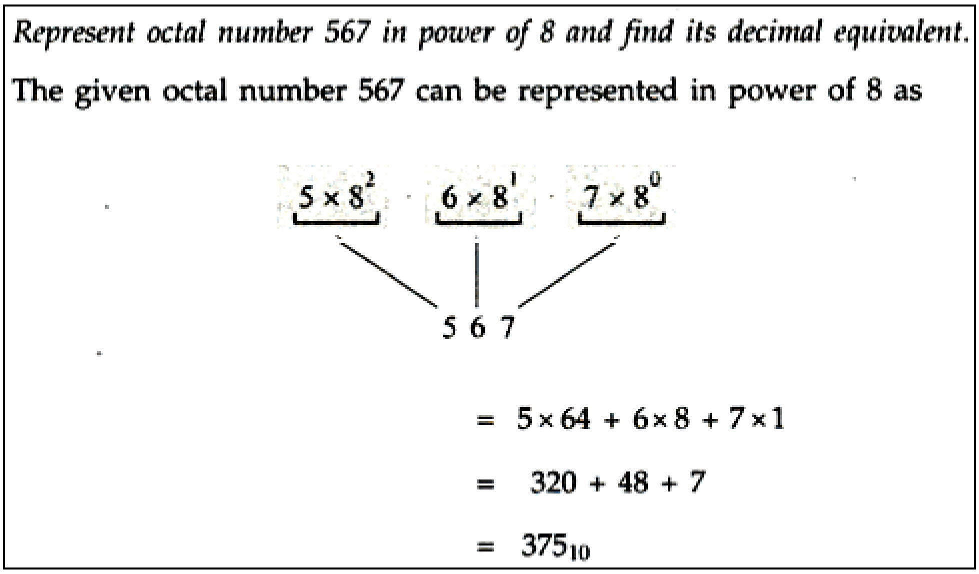
$$\begin{aligned} N &= 1\times2^3 + 1\times2^2 + 0\times2^1 + 1\times2^0 + 1\times2^{-1} \\ &\quad + 0\times2^{-2} + 1\times2^{-3} \\ &= 8 + 4 + 0 + 1 + 0.5 + 0 + 0.125 \\ &= 13.625_{10} \end{aligned}$$

Octal number system

- The octal number system is composed of eight digits which are 0,1,2,3,4,5,6,7
- It has base of 8.The weightage is given as

………$8^4,8^3,8^8,8^1,8^0\bullet 8^{-1},8^{-8},8^{-3}$………

Binary point



Represent octal number 567 in power of 8 and find its decimal equivalent.

The given octal number 567 can be represented in power of 8 as

$$\underbrace{5\times8^2}\quad\underbrace{6\times8^1}\quad\underbrace{7\times8^0}$$

5 6 7

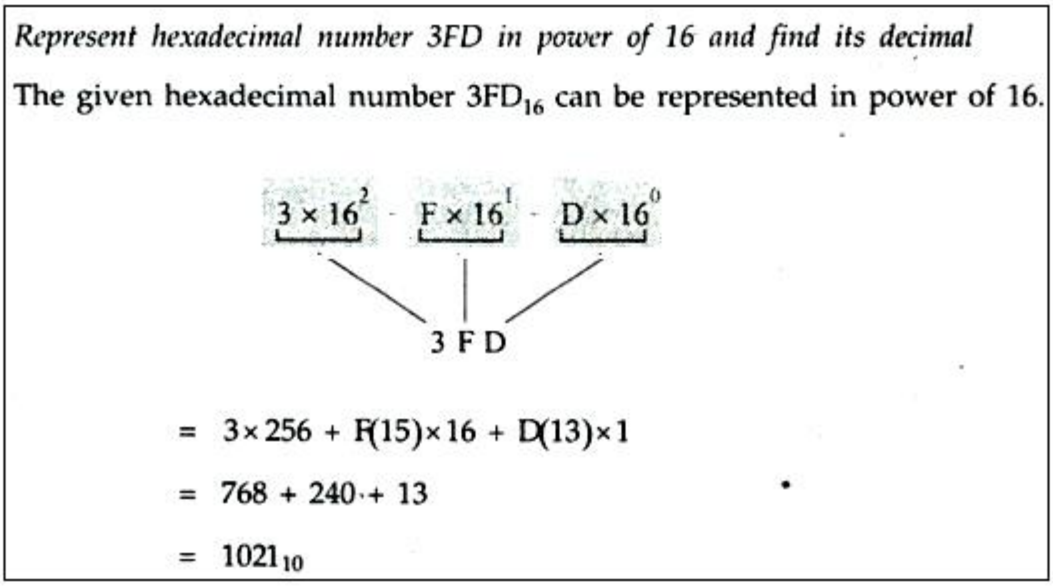
$$\begin{aligned} &= 5\times64 + 6\times8 + 7\times1 \\ &= 320 + 48 + 7 \\ &= 375_{10} \end{aligned}$$

Hexa decimal number system

- The octal number system is composed of 16 digits which are 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F
- It has base of 8.The weightage is given as

……….$16^4,16^3,16^{16},16^1,16^0.16^{-1},16^{-2},16^{-3}$………

Binary point



Represent hexadecimal number 3FD in power of 16 and find its decimal

The given hexadecimal number $3FD_{16}$ can be represented in power of 16.

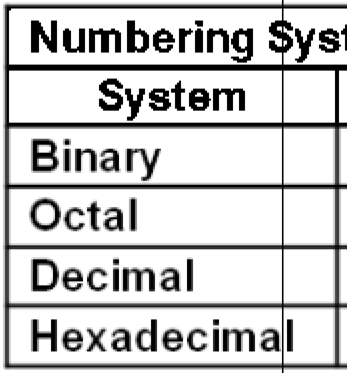
$$\underbrace{3\times16^2}\cdot\underbrace{F\times16^1}\cdot\underbrace{D\times16^0}$$

3 F D

$= 3\times256 + F(15)\times16 + D(13)\times1$

$= 768 + 240 + 13$

$= 1021_{10}$



Numbering Syst
System
Binary
Octal
Decimal
Hexadecimal

DECIMAL	BINARY	OCTAL	HEXA
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

2. WRITE A NOTE ON NUMBER RPRESENTAION IN BINARY.

- Different formats used for binary representation of both positive and negative decimal numbers include the
  - Sign-bit magnitude method,
  - The 1's complement method and
  - The 2's complement method.

Single bit magnitude

- In the sign-bit magnitude representation of positive and negative decimal numbers, the MSB represents the 'sign', with a '0' denoting a plus sign and a '1' denoting a minus sign.
- The remaining bits represent the magnitude. In eight-bit representation, while MSB represents the sign, the remaining seven bits represent the magnitude.
- For example, the eight-bit representation of +9 would be 00001001, and that for −9 would be 10001001.
- An n−bit binary representation can be used to represent decimal numbers in the range of $-(2^{n-1}-1)$ to $+(2^{n-1}-1)$.
- That is, eight-bit representation can be used to represent decimal numbers in the range from −127 to +127 using the sign-bit magnitude format.

1's complement

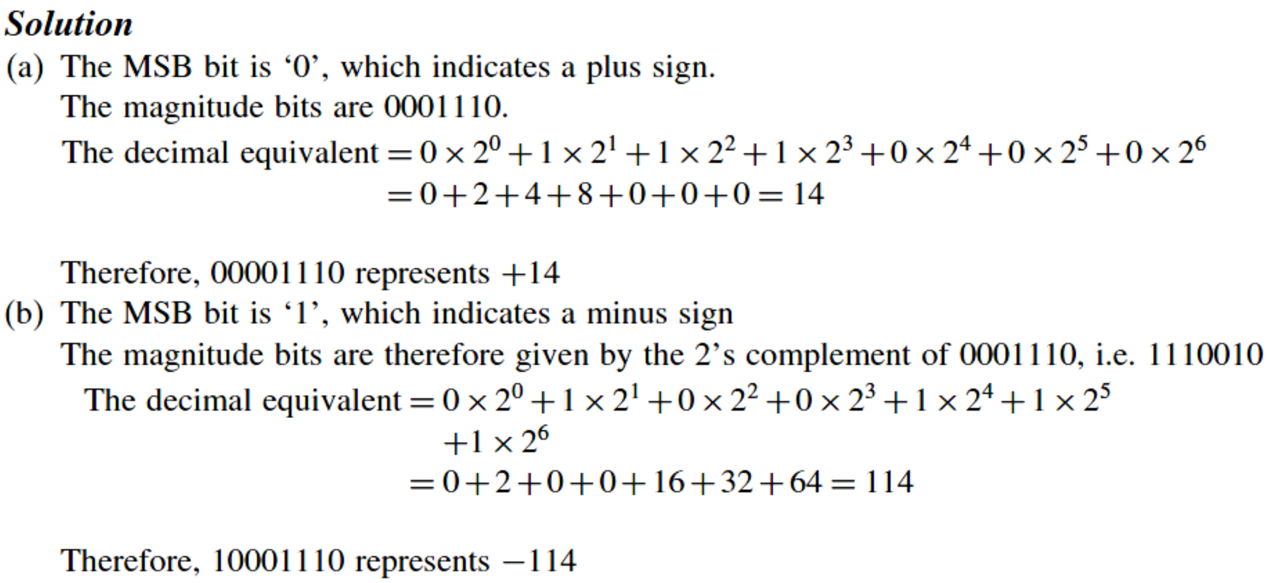
- In the 1's complement format, the positive numbers remain unchanged.
- The negative numbers are obtained by taking the 1's complement of the positive counterparts.
- For example, +9 will be represented as 00001001 in eight-bit notation, and −9 will be represented as 11110110, which is the 1's complement of 00001001.
- Again, n-bit notation can be used to represent numbers in the range from $-(2^{n-1}-1)$ to $+(2^{n-1}-1)$ using the 1's complement format.
- The eight-bit representation of the 1's complement format can be used to represent decimal numbers in the range from −127 to +127.

2's complement

- In the 2's complement representation of binary numbers, the MSB represents the sign, with a '0' used for a plus sign and a '1' used for a minus sign.
- The remaining bits are used for representing magnitude. Positive magnitudes are represented in the same way as in the case of sign-bit or 1's complement representation.
- Negative magnitudes are represented by the 2's complement of their positive counterparts.
- For example, +9 would be represented as 00001001, and −9 would be written as 11110111.
- Please note that, if the 2's complement of the magnitude of +9 gives a magnitude of −9, then the reverse process will also be true, i.e. the 2's complement of the magnitude of −9 will give a magnitude of +9.
- The n-bit notation of the 2's complement format can be used to represent all decimal numbers in the range from $+(2^{n-1}-1)$ to $-(2^{n-1})$.
- The 2's complement format is very popular as it is very easy to generate the 2's complement of a binary number and also because arithmetic operations are relatively easier to perform when the numbers are represented in the 2's complement format.

3. FIND THE DECIMAL EQUIVALENT OF THE FOLLOWING BINARY NUMBERS EXPRESSED IN THE 2'S COMPLEMENT FORMAT:

(a) 00001110
(b) 10001110



Solution

(a) The MSB bit is '0', which indicates a plus sign.

The magnitude bits are 0001110.

$$\text{The decimal equivalent} = 0 \times 2^0 + 1 \times 2^1 + 1 \times 2^2 + 1 \times 2^3 + 0 \times 2^4 + 0 \times 2^5 + 0 \times 2^6$$
$$= 0 + 2 + 4 + 8 + 0 + 0 + 0 = 14$$

Therefore, 00001110 represents $+14$

(b) The MSB bit is '1', which indicates a minus sign

The magnitude bits are therefore given by the 2's complement of 0001110, i.e. 1110010

$$\text{The decimal equivalent} = 0 \times 2^0 + 1 \times 2^1 + 0 \times 2^2 + 0 \times 2^3 + 1 \times 2^4 + 1 \times 2^5 + 1 \times 2^6$$
$$= 0 + 2 + 0 + 0 + 16 + 32 + 64 = 114$$

Therefore, 10001110 represents -114

4. PERFORM VARIOUS NUMBER SYSTEM CONVERSTION USING SUITABLE EXAMPLES

Finding the Decimal Equivalent

- The decimal equivalent of a given number in another number system is given by the sum of all the digits multiplied by their respective place values.
- The integer and fractional parts of the given number should be treated separately.
- Binary-to-decimal, octal-to-decimal and hexadecimal-to-decimal conversions are illustrated below with the help of examples.

Binary-to-Decimal Conversion

The decimal equivalent of the binary number $(1001.0101)_2$ is determined as follows:

- The integer part = 1001
- The decimal equivalent = $1 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 = 1 + 0 + 0 + 8 = 9$
- The fractional part = .0101
- Therefore, the decimal equivalent = $0 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} + 1 \times 2^{-4}$
  $= 0 + 0.25 + 0 + 0.0625 = 0.3125$
- Therefore, the decimal equivalent of $(1001.0101)_2 = 9.3125$

Octal-to-Decimal Conversion

The decimal equivalent of the octal number $(137.21)_8$ is determined as follows:

- The integer part = 137
- The decimal equivalent = $7 \times 8^0 + 3 \times 8^1 + 1 \times 8^2 = 7 + 24 + 64 = 95$
- The fractional part = .21
- The decimal equivalent = $2 \times 8^{-1} + 1 \times 8^{-2} = 0.265$
- Therefore, the decimal equivalent of $(137.21)_8 = (95.265)_{10}$

Hexadecimal-to-Decimal Conversion

The decimal equivalent of the hexadecimal number $(1E0.2A)_{16}$ is determined as follows:

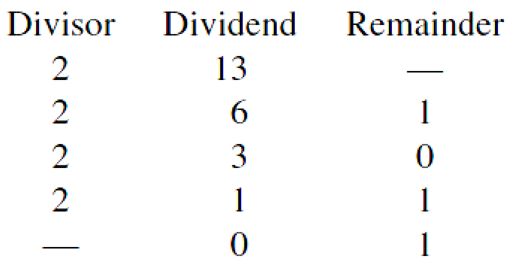
- The integer part = 1E0
- The decimal equivalent = $0 \times 16^0 + 14 \times 16^1 + 1 \times 16^2 = 0 + 224 + 256 = 480$
- The fractional part = 2A
- The decimal equivalent = $2 \times 16^{-1} + 10 \times 16^{-2} = 0.164$
- Therefore, the decimal equivalent of $(1E0.2A)_{16} = (480.164)_{10}$

Decimal -to- Binary Conversion

- The integer and fractional parts are worked on separately.
- For the integer part, the binary equivalent can be found by successively dividing the integer part of the number by 2 and recording the remainders until the quotient becomes ‘0’.
- The remainders written in reverse order constitute the binary equivalent.
- For the fractional part, it is found by successively multiplying the fractional part of the decimal number by 2 and recording the carry until the result of multiplication is ‘0’.
- The carry sequence written in forward order constitutes the binary equivalent of the fractional part of the decimal number.
- If the result of multiplication does not seem to be heading towards zero in the case of the fractional part, the process may be continued only until the requisite number of equivalent bits has been obtained. This method of decimal–binary conversion is popularly known as the double-dabble method.

The binary equivalent of the decimal number $(13.375)_{10}$ is determined as follows:

- The integer part = 13



Divisor	Dividend	Remainder
2	13	—
2	6	1
2	3	0
2	1	1
—	0	1

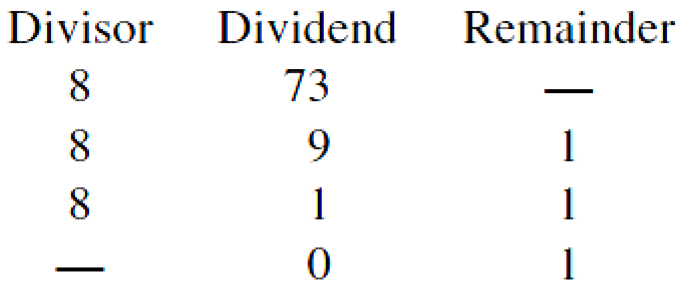
- The binary equivalent of $(13)_{10}$ is therefore $(1101)_2$
- The fractional part = .375
- 0.375 × 2 = 0.75 with a carry of 0
- 0.75 × 2 = 0.5 with a carry of 1
- 0.5 × 2 = 0 with a carry of 1
- The binary equivalent of $(0.375)_{10}= (.011)_2$
- Therefore, the binary equivalent of (13.375)10 = (1101.011)2

Decimal -to- octal Conversion

- The process of decimal-to-octal conversion is similar to that of decimal-to-binary conversion. The progressive division in the case of the integer part and the progressive multiplication while working on the fractional part here are by ‘8’ which is the radix of the octal number system.
- Again, the integer and fractional parts of the decimal number are treated separately.

- The octal equivalent of the decimal number $(73.75)_{10}$ is determined as follows:

- The integer part = 73

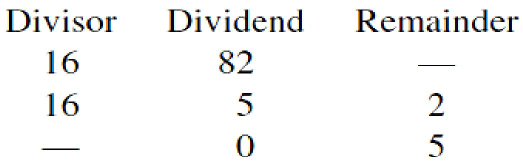


Divisor	Dividend	Remainder
8	73	—
8	9	1
8	1	1
—	0	1

- The octal equivalent of $(73)_{10}$= $(111)_8$
- The fractional part = 0.75
- 0.75 × 8 = 0 with a carry of 6
- The octal equivalent of $(0.75)_{10}$= $(.6)_8$
- Therefore, the octal equivalent of $(73.75)_{10}$= $(111.6)_8$

Decimal -to- Hexa decimal Conversion

- The process of decimal-to-hexadecimal conversion is also similar.
- Since the hexadecimal number system has a base of 16, the progressive division and multiplication factor in this case is 16.
- The hexa decimal equivalent of the decimal number $(82.25)_{10}$ is determined as follows:
- The integer part = 82



Divisor	Dividend	Remainder
16	82	—
16	5	2
—	0	5

- The hexadecimal equivalent of $(82)_{10}$= $(52)_{16}$
- The fractional part = 0.25
- 0.25 × 16 = 0 with a carry of 4
- Therefore, the hexadecimal equivalent of $(82.25)_{10}$ = $(52.4)_{16}$

Binary–Octal and Octal–Binary Conversions

- An octal number can be converted into its binary equivalent by replacing each octal digit with its three-bit binary equivalent.
- We take the three-bit equivalent because the base of the octal number system is 8 and it is the third power of the base of the binary number system, i.e. 2.
- All we have then to remember is the three-bit binary equivalents of the basic digits of the octal number system.
- A binary number can be converted into an equivalent octal number by splitting the integer and fractional parts into groups of three bits, starting from the binary point on both sides.
- The 0s can be added to complete the outside groups if needed.
- The binary equivalent of the octal number $(374.26)_8$ and the octal equivalent of $(1110100.0100111)_2$ is determined as follows:

- The given octal number = $(374.26)_8$
- The binary equivalent = $(011\ 111\ 100.010\ 110)_2$= $(011111100.010110)_2$
- Any 0s on the extreme left of the integer part and extreme right of the fractional part of the equivalent binary number should be omitted.
- Therefore, $(011111100.010110)_2$= $(11111100.01011)_2$
- The given binary number = $(1110100.0100111)_2$
- $(1110100.0100111)_2$= $(1\ 110\ 100.010\ 011\ 1)_2$= $(001\ 110\ 100.010\ 011\ 100)_2$= $(164.234)_8$

Hex to binary and binary to hex Conversions

- A hexadecimal number can be converted into its binary equivalent by replacing each hex digit with its four-bit binary equivalent.
- We take the four-bit equivalent because the base of the hexadecimal number system is 16 and it is the fourth power of the base of the binary number system.
- All we have then to remember is the four-bit binary equivalents of the basic digits of the hexadecimal number system.
- A given binary number can be converted into an equivalent hexadecimal number by splitting the integer and fractional parts into groups of four bits, starting from the binary point on both sides.
- The 0s can be added to complete the outside groups if needed.

- The binary equivalent of the hex number $(17E.F6)_{16}$ and the hex equivalent of *(1011001110.011011101)*$_2$is determined as follows:

- The given hex number = $(17E.F6)_{16}$
- The binary equivalent = $(0001\ 0111\ 1110.1111\ 0110)_2$
  = $(000101111110.11110110)_2$
  = (101111110.1111011)2

- The 0s on the extreme left of the integer part and on the extreme right of the fractional part have been omitted.
- The given binary number = $(1011001110.011011101)_2$
  = $(10\ 1100\ 1110.0110\ 1110\ 1)_2$
- The hex equivalent = $(0010\ 1100\ 1110.0110\ 1110\ 1000)_2$= $(2CE.6E8)_{16}$

Hex–Octal and Octal–Hex Conversions

- For hexadecimal–octal conversion, the given hex number is firstly converted into its binary equivalent which is further converted into its octal equivalent.
- An alternative approach is firstly to convert the given hexadecimal number into its decimal equivalent and then convert the decimal number into an equivalent octal number.
- The former method is definitely more convenient and straightforward.
- For octal–hexadecimal conversion, the octal number may first be converted into an equivalent binary number and then the binary number transformed into its hex equivalent.
- The other option is firstly to convert the given octal number into its decimal equivalent and then convert the decimal number into its hex equivalent.
- The former approach is definitely the preferred one.

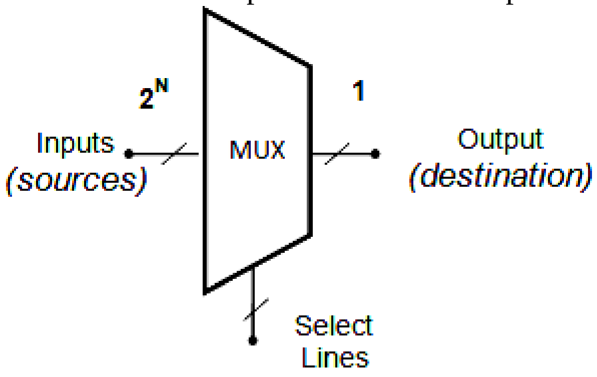
- The octal equivalent of the hex number $(2F.C4)_{16}$ and the hex equivalent of octal number *(762.013)*$_8$is determined as follows:

- The given hex number = $(2F.C4)_{16.}$
- The binary equivalent = $(0010\ 1111.1100\ 0100)_2$
  = $(00101111.11000100)_2$
  = $(101111.110001)_2$
  = $(101\ 111.110\ 001)_2$
  = $(57.61)_{8.}$
- The given octal number = $(762.013)_{8.}$
- The octal number = $(762.013)_8$ = $(111\ 110\ 010.000\ 001\ 011)_2$
  = $(111110010.000001011)_2$
  = $(0001\ 1111\ 0010.0000\ 0101\ 1000)_2$
  = $(1F2.058)_{16.}$

5. WHAT IS MULTIPLEXER? EXPLAIN IN DETAIL WITH RELEVANT LOGIC CIRCUITS AND TRUTH TABLE.

Definition

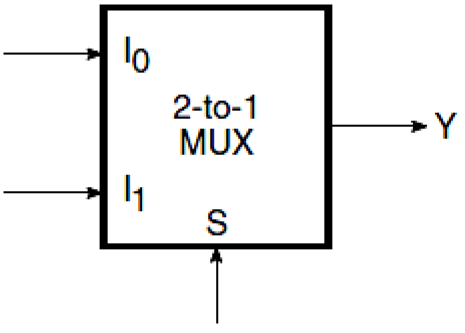
- A *multiplexer* or *MUX*, also called a ***data selector***, is a combinational circuit with more than one input line, one output line and more than one selection line.
- A multiplexer selects binary information present on any one of the input lines, depending upon the logic status of the selection inputs, and routes it to the output line.
- If there are n selection lines, then the number of maximum possible input lines is 2n and the multiplexer is referred to as a 2^n-to-1 multiplexer or 2^n×1 multiplexer.



MUX Types

- 2-to-1 (1 select line)
- 4-to-1 (2 select lines)
- 8-to-1 (3 select lines)
- 16-to-1 (4 select lines)

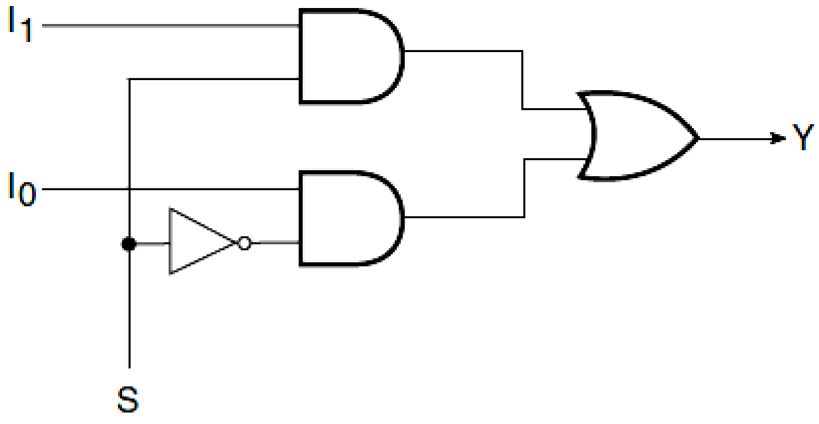
2 to 1 multiplexer



(a)

S	Y
0	I_0
1	I_1

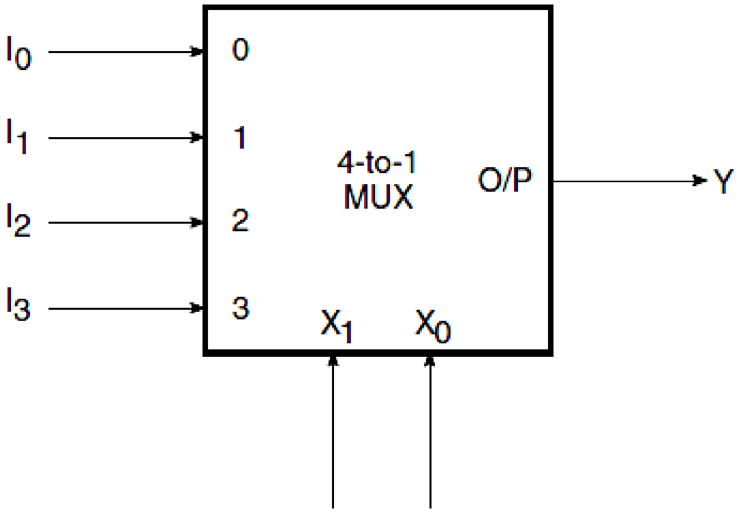
(b)



The circuit functions as follows:

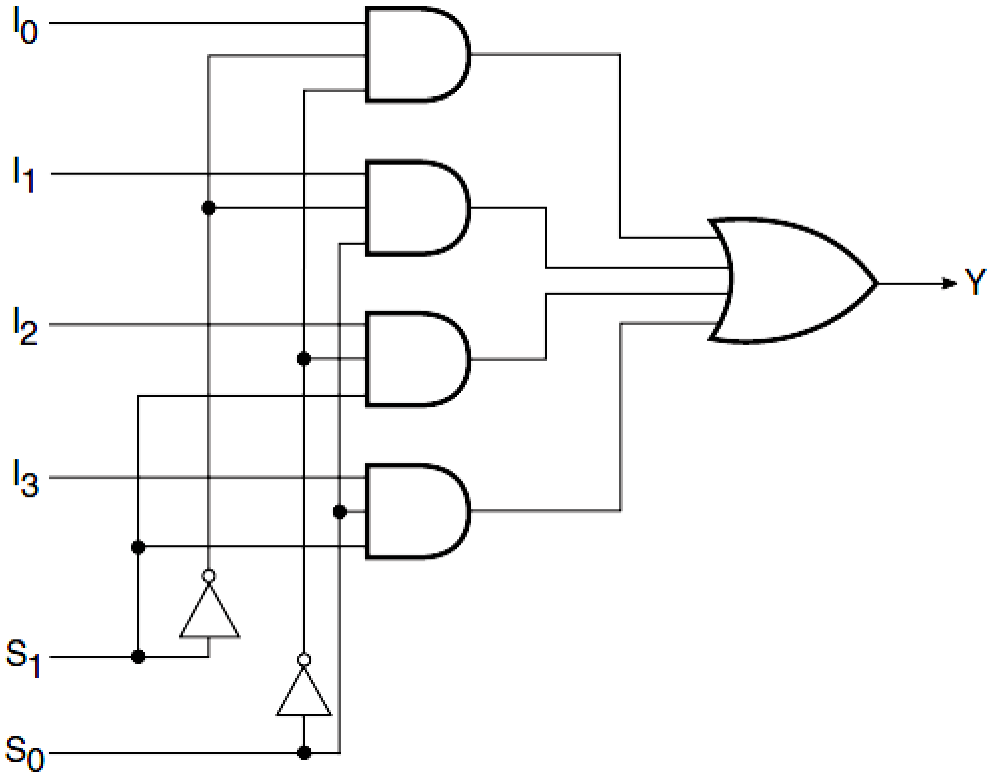
- For S = 0, the Boolean expression for the output becomes $Y = I_0$.
- For S = 1, the Boolean expression for the output becomes $Y = I_1$.

4 to 1 multiplexer



(a)

S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

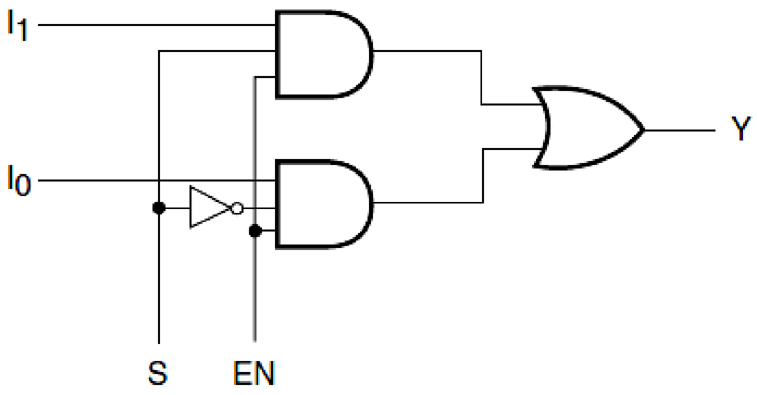


- ➢ The input combinations 00, 01, 10 and 11 on the select lines respectively switch I_0, I_1, I_2 and I_3 to the output.
- ➢ The operation of the circuit is governed by the Boolean function

$$Y = I_0\overline{S_1}\,\overline{S_0} + I_1\overline{S_1}\,S_0 + I_2S_1\overline{S_0} + I_3S_1S_0$$

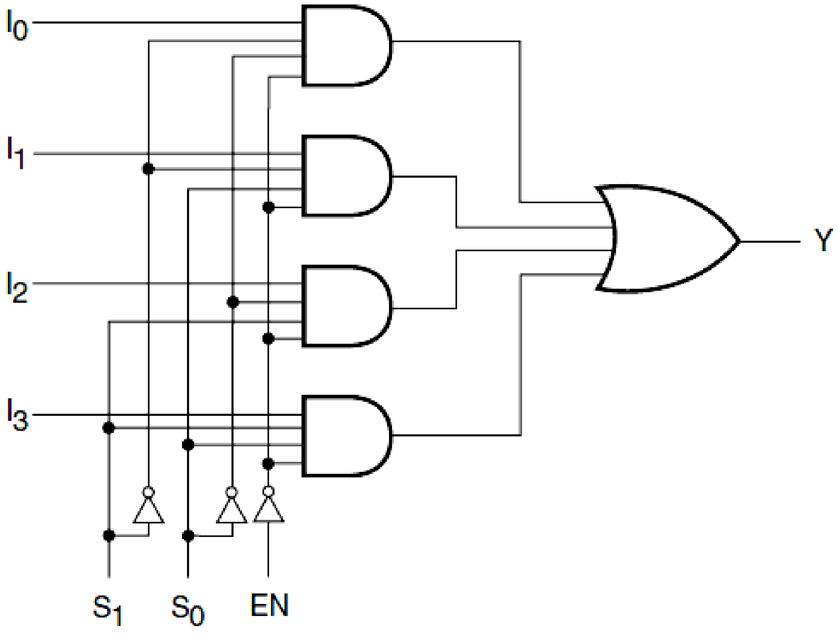
2 to 1 multiplexer and 4 to 1 multiplexer with ENABLE input

- Multiplexers usually have an ENABLE input that can be used to control the multiplexing function.
- When this input is enabled, that is, when it is in logic ‘1’ or logic ‘0’ state, depending upon whether the ENABLE input is active HIGH or active LOW respectively, the output is enabled.
- The multiplexer functions normally.
- When the ENABLE input is inactive, the output is disabled and permanently goes to either logic ‘0’ or logic ‘1’ state, depending upon whether the output is uncomplemented or complemented.
- Figure below shows how the 2-to-1 multiplexer can be modified to include an ENABLE input. The functional table of this modified multiplexer is also shown in Figure.



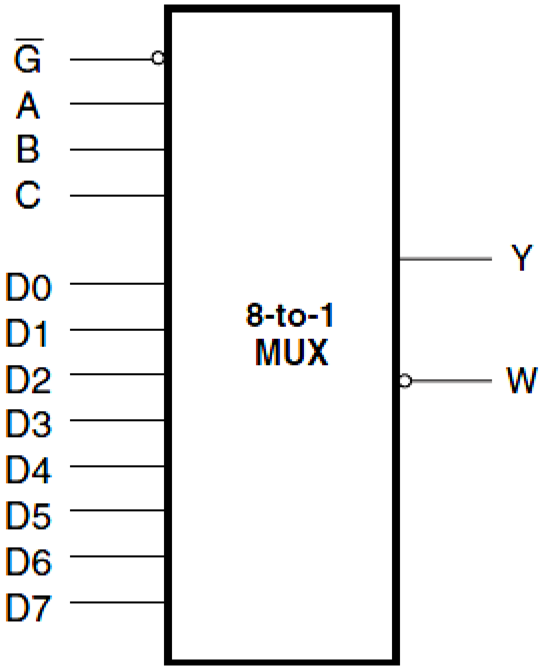
S	EN	Y
X	0	0
0	1	I_0
1	1	I_1

- The ENABLE input here is active when HIGH. Some IC packages have more than one multiplexer. In that case, the ENABLE input and selection inputs are common to all multiplexers
- within the same IC package.
- Figure below shows a 4-to-1 multiplexer with an active LOW ENABLE input.



S_1	S_0	EN	Y
X	X	1	0
0	0	0	I_0
0	1	0	I_1
1	0	0	I_2
1	1	0	I_3

8 to 1 multiplexer

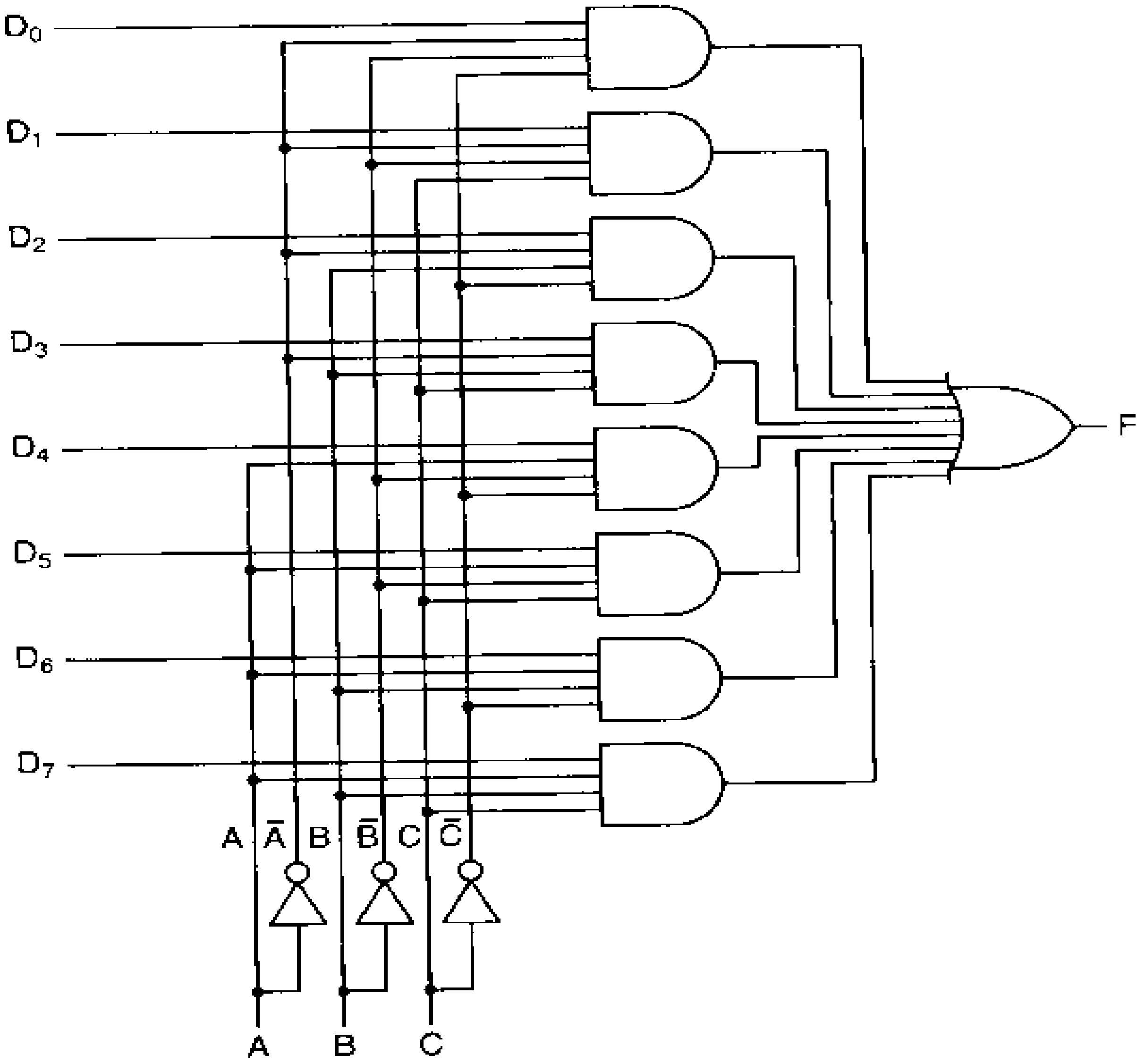


(a)

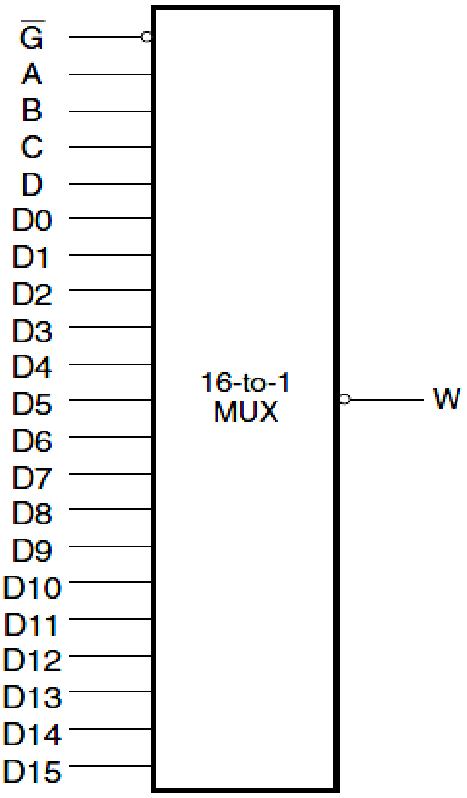
Inputs				Output	
Select			Enable		
C	B	A	$\overline{G}$	Y	W
X	X	X	H	L	H
L	L	L	L	D0	$\overline{D0}$
L	L	H	L	D1	$\overline{D1}$
L	H	L	L	D2	$\overline{D2}$
L	H	H	L	D3	$\overline{D3}$
H	L	L	L	D4	$\overline{D4}$
H	L	H	L	D5	$\overline{D5}$
H	H	L	L	D6	$\overline{D6}$
H	H	H	L	D7	$\overline{D7}$

$\overline{G}$: ENABLE input
A, B, C : Select inputs
D0-D7 : Data inputs
Y,W : outputs

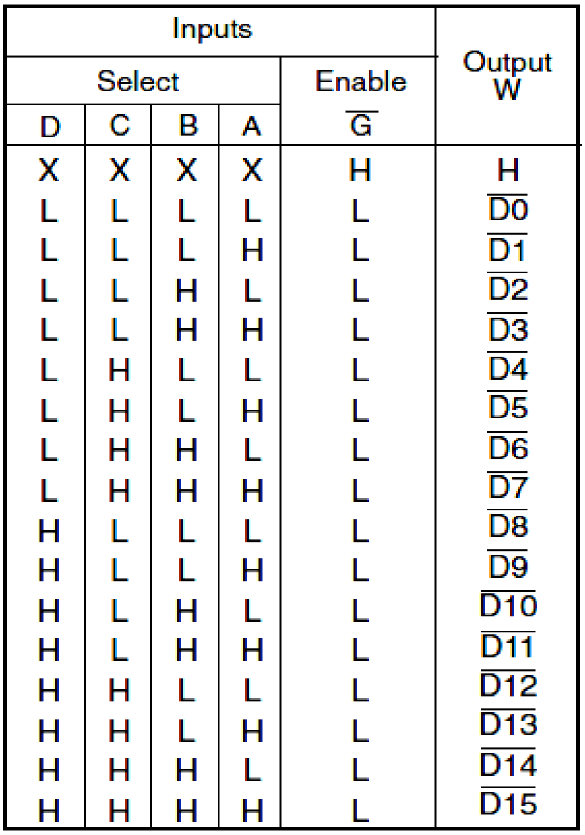
(b)



16 to 1 multiplexer

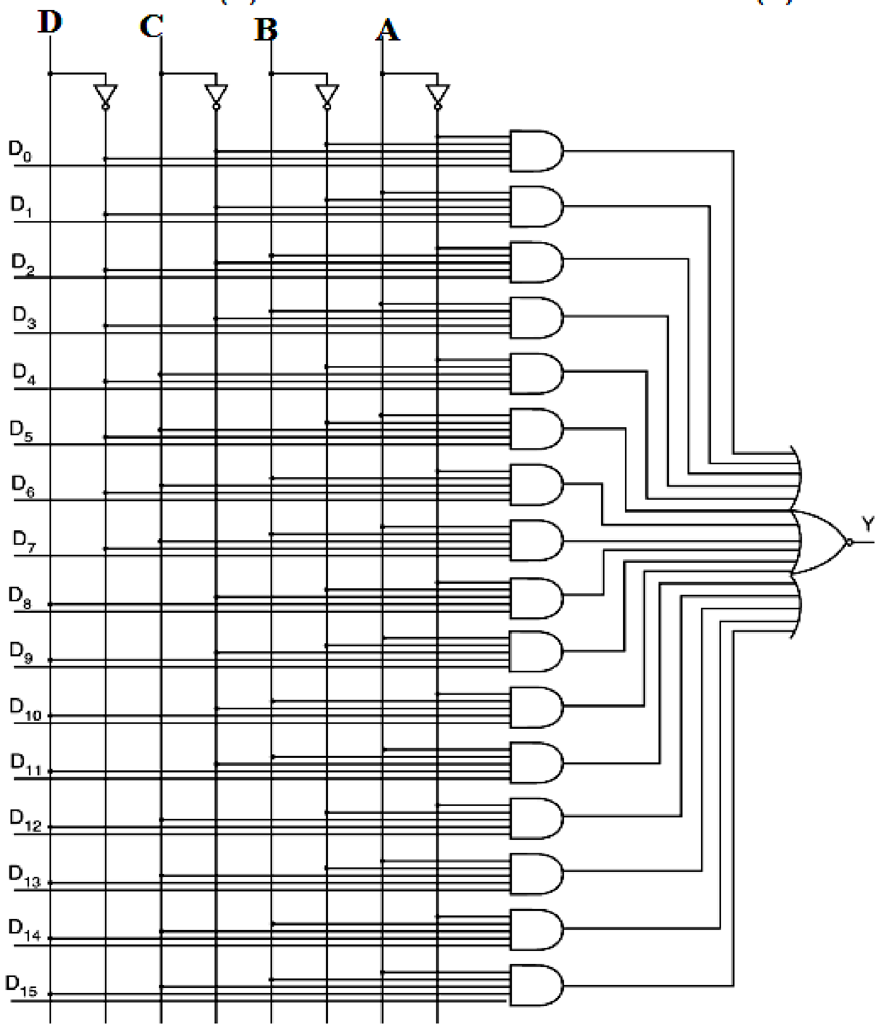


(a)



Inputs					Output W
Select				Enable	
D	C	B	A	$\overline{G}$	
X	X	X	X	H	H
L	L	L	L	L	$\overline{D0}$
L	L	L	H	L	$\overline{D1}$
L	L	H	L	L	$\overline{D2}$
L	L	H	H	L	$\overline{D3}$
L	H	L	L	L	$\overline{D4}$
L	H	L	H	L	$\overline{D5}$
L	H	H	L	L	$\overline{D6}$
L	H	H	H	L	$\overline{D7}$
H	L	L	L	L	$\overline{D8}$
H	L	L	H	L	$\overline{D9}$
H	L	H	L	L	$\overline{D10}$
H	L	H	H	L	$\overline{D11}$
H	H	L	L	L	$\overline{D12}$
H	H	L	H	L	$\overline{D13}$
H	H	H	L	L	$\overline{D14}$
H	H	H	H	L	$\overline{D15}$

(b)



(b) Logic diagram of 16-to-1 multiplexer

6. EXPLAIN THE PROCEDURE IN IMPLEMENTING BOOLEAN FUNCTIONS WITH MULTIPLEXERS.

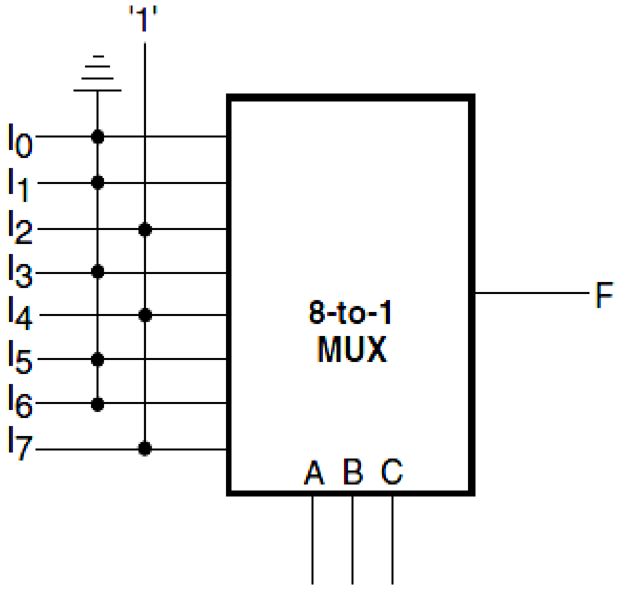
- One of the most common applications of a multiplexer is its use for implementation of combinational
- logic Boolean functions.
- The simplest technique for doing so is to employ a 2n-to-1 MUX to implement an n-variable Boolean function.
- The input lines corresponding to each of the minterms present in the Boolean function are made equal to logic ‘1’ state.
- The remaining minterms that are absent in the Boolean function are disabled by making their corresponding input lines equal to logic ‘0’.

EXAMPLE

For implementing the Boolean function given by the equation $\boldsymbol{f(A,B,C) = \sum 2,4,7}$
In terms of variables A, B and C, equation

$$f(A,B,C) = \overline{A}.B.\overline{C} + A.\overline{B}.\overline{C} + A.B.C$$

As shown in Figure the input lines corresponding to the three minterms present in the given Boolean Function are tied to logic ‘1’.
The remaining five possible minterms absent in the Boolean function are tied to logic ‘0’.



In this, a 2^n-to-1 MUX can be used to implement a Boolean function with n + 1 variables.

The procedure is as follows.

- Out of n + 1 variables, n are connected to the n selection lines of the 2^n-to-1 multiplexer.
- The left-over variable is used with the input lines.
- Various input lines are tied to one of the following: ‘0’, ‘1’, the left-over variable and the complement of the left-over variable.

Which line is given what logic status can be easily determined with the help of a simple procedure.

Steps

- It is a three-variable Boolean function. Conventionally, we will need to use an 8-to-1 multiplexer to implement this function.
- We will now see how this can be implemented with a 4-to-1 multiplexer.
- The chosen multiplexer has two selection lines.
- The first step here is to determine the truth table of the given Boolean function, which is shown in Table 8.1.

Table 8.1 Truth table.

Minterm	*A*	*B*	*C*	*f(A,B,C)*
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

- In the next step, two of the three variables are connected to the two selection lines, with the higher order variable connected to the higher-order selection line.
- For instance, in the present case, variables B and C are the chosen variables for the selection lines and are respectively connected to selection lines S1 and S0.
- In the third step, a table of the type shown in Table 8.2 is constructed.
- Under the inputs to the multiplexer, minterms are listed in two rows, as shown.
- The first row lists those terms where remaining variable A is complemented, and second row lists those terms where A is uncomplemented.
- This is easily done with the help of the truth table.
- The required minterms are identified or marked in some manner in this table.
- In the given table, these entries have been highlighted.
- Each column is inspected individually.
- If neither minterm of a certain column is highlighted, a '0' is written below that.
- If both are highlighted, a '1' is written.
- If only one is highlighted, the corresponding variable (complemented or uncomplemented) is written.
- The input lines are then given appropriate logic status. In the present case, I_0, I_1, I_2 and I_3 would be connected to A, 0, A and A respectively.
- Figure below shows the logic implementation.

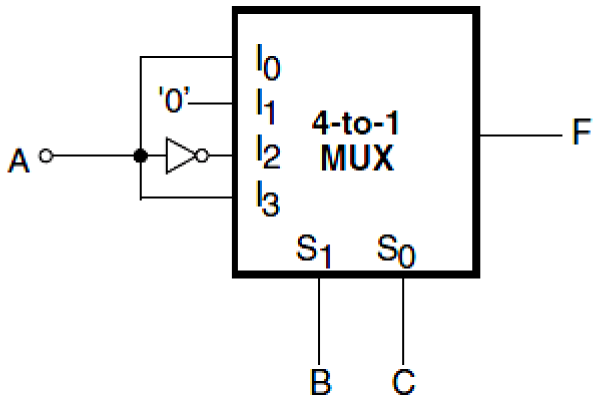


Table 8.2 Implementation table for multiplexers.

	I_0	I_1	I_2	I_3
$\bar{A}$	0	1	**2**	3
A	**4**	5	6	**7**
	A	0	$\bar{A}$	A

- It is not necessary to choose only the leftmost variable in the sequence to be used as input to the multiplexer. Any of the variables can be used provided the implementation table is constructed accordingly.
- Consider C as the left-out variable, the implementation table will be as shown in Table 8.3.
- Figure below shows the hardware implementation.

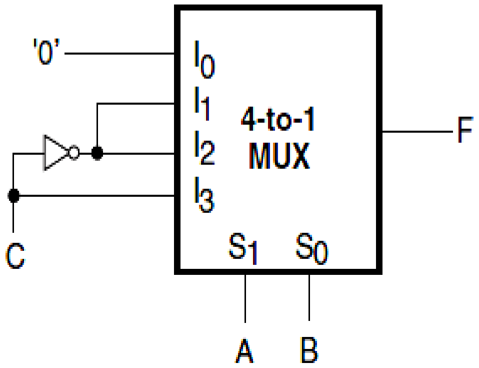
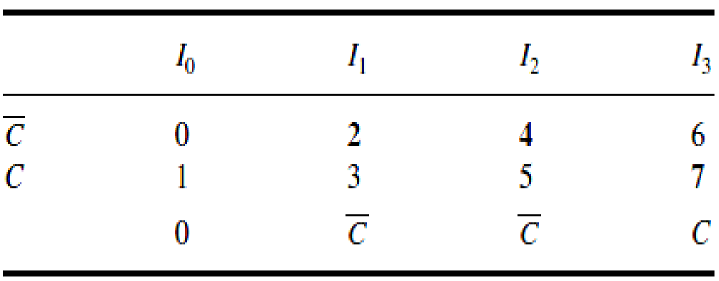


Table 8.3 Implementation table for multiplexers.



	I_0	I_1	I_2	I_3
$\overline{C}$	0	2	4	6
C	1	3	5	7
	0	$\overline{C}$	$\overline{C}$	C

For the case of B being the left-out variable, the implementation table is shown in Table 8.4 and the hardware implementation is shown in Figure below

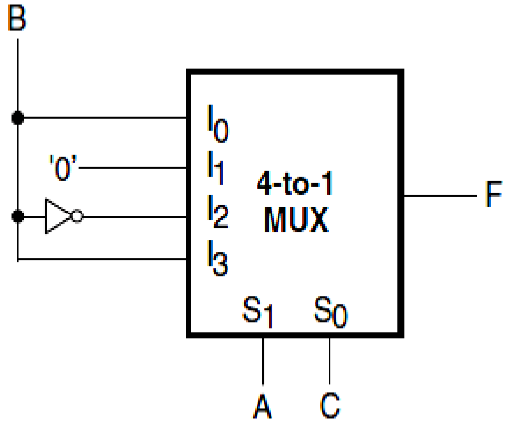
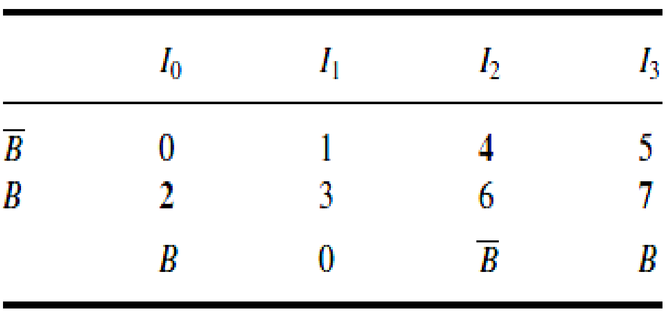


Table 8.4 Implementation table for multiplexers.



	I_0	I_1	I_2	I_3
$\overline{B}$	0	1	4	5
B	2	3	6	7
	B	0	$\overline{B}$	B

7. IMPLEMENT THE PRODUCT-OF-SUMS BOOLEAN FUNCTION EXPRESSED BY Π1,2,5 BY A SUITABLE MULTIPLEXER.

Solution

Let the Boolean function be f(A, B, C) = Π 1,2, 5.
The equivalent sum-of-products expression can be written as f(A, B, C) =∑ 0,3,4,6,7

- The truth table for the given Boolean function is given in Table 8.5. The given function can be implemented with a 4-to-1 multiplexer with two selection lines.
- Variables A and B are chosen for the selection lines.
- The implementation table as drawn with the help of the truth table is given in Table 8.6.Figure 8.12 shows the hardware implementation.

Table 8.5 Truth table.

C	B	A	$f(A,B,C)$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Table 8.6 Implementation table.

	I_0	I_1	I_2	I_3
$\overline{C}$	**0**	1	2	**3**
C	**4**	5	**6**	**7**
	1	0	C	1

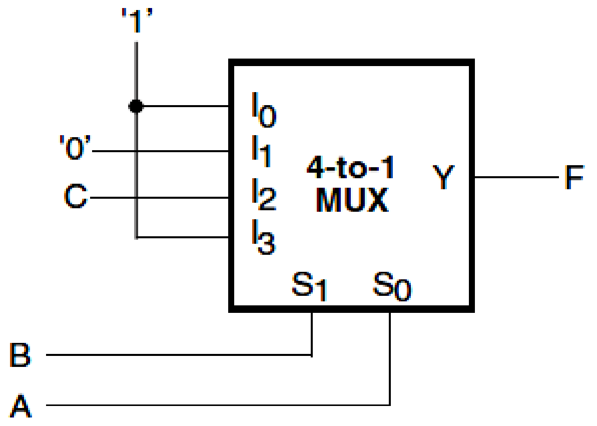
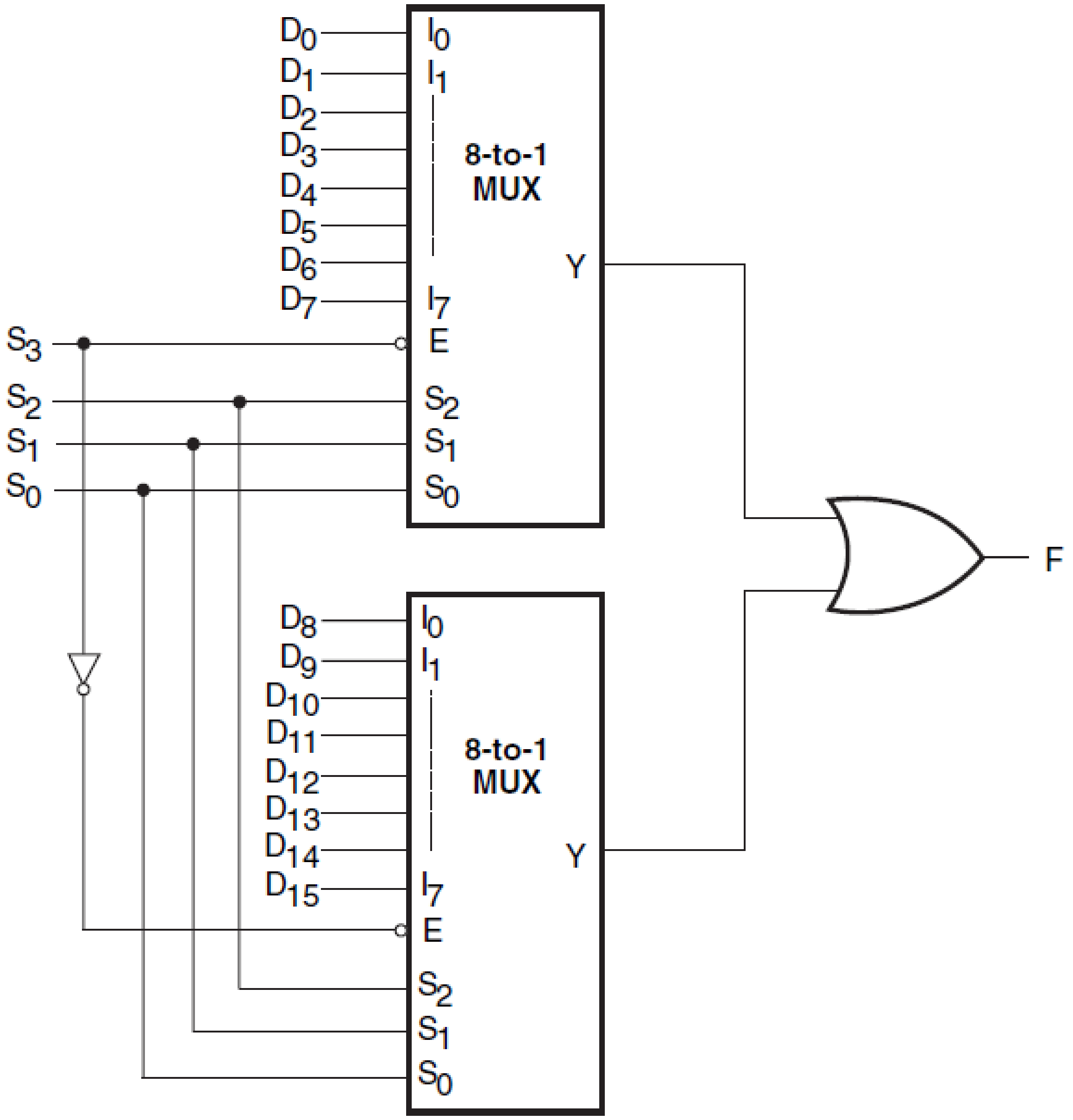


Figure 8.12 Example 8.1.

8. DESIGN A 16-TO-1 MULTIPLEXER USING TWO 8-TO-1 MULTIPLEXERS HAVING AN ACTIVE LOW ENABLE INPUT

- A 16-to-1 multiplexer can be constructed from two 8-to-1 multiplexers having an ENABLE input. The ENABLE input is taken as the fourth selection variable occupying the MSB position.
- Figure below shows the complete logic circuit diagram. IC 74151 can be used to implement an 8-to-1 multiplexer.
- The circuit functions as follows. When S3 is in logic '0' state, the upper multiplexer is enabled and the lower multiplexer is disabled.
- If we recall the truth table of a four-variable Boolean function, S3 would be '0' for the first eight entries and '1' for the remaining eight entries.
- Therefore, when S3 = 0 the final output will be any of the inputs from D0 to D7, depending upon the logic status of S2, S1 and S0.

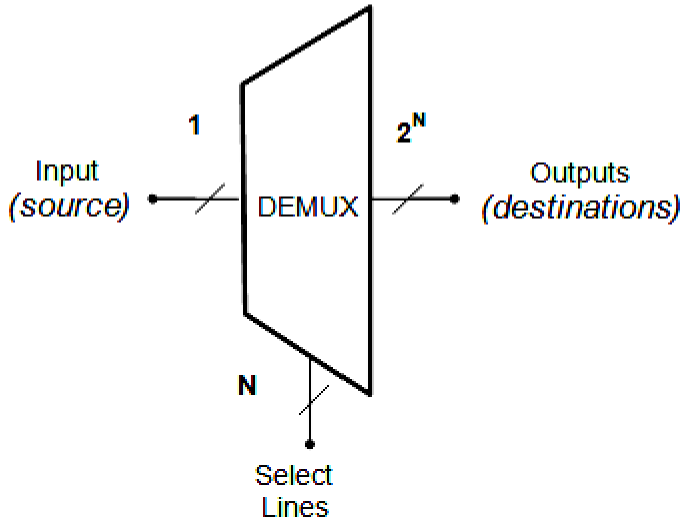
- Similarly, when S3 = 1 the final output will be any of the inputs from D8 to D15, again depending upon the logic status of S2, S1 and S0.
- The circuit therefore implements the truth table of a 16-to-1 multiplexer.



9. WHAT IS DEMULTIPLEXER? EXPLAIN IN DETAIL WITH RELEVANT LOGIC CIRCUITS AND TRUTH TABLE.

Definition

- A *demultiplexer* is a combinational logic circuit with an input line, ***2^noutput lines*** and ***n select lines***.
- It routes the information present on the input line to any of the output lines.
- The output line that gets the information present on the input line is decided by the bit status of the selection lines.
- A *decoder* is a special case of a demultiplexer without the input line.

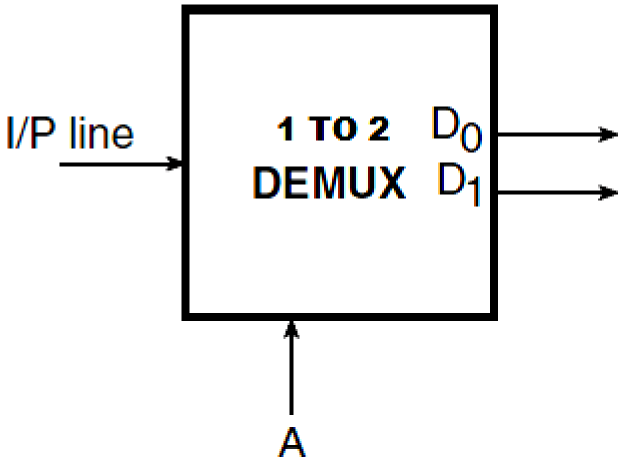


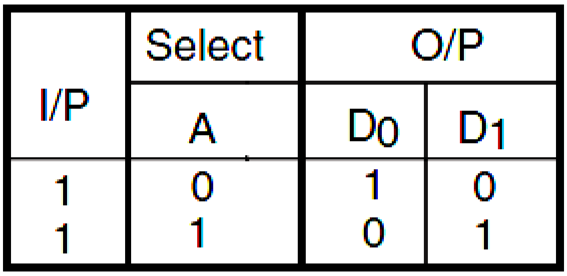
DEMUX Types

- 1-to-2 (1 select line)
- 1-to-4 (2 select lines)
- 1-to-8 (3 select lines)
- 1-to-16 (4 select lines)

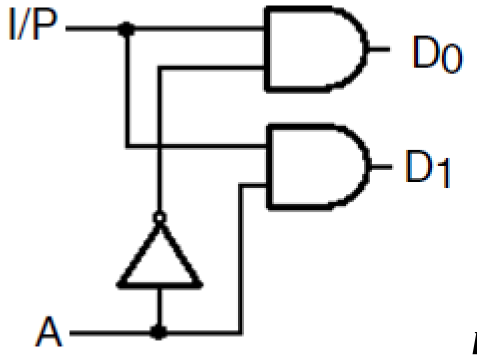
1-to-2 DEMUX(1 select line)

- Figure (a) shows the circuit representation of a 1-to-2 demultiplexer. Figure (b) shows the truth table of the demultiplexer when the input line is held HIGH.
- The input data is transmitted to one of the outputs D_n by means of select signals A
- The data is available at output D0 for A=0 and for A=1 the data is available at D1.





I/P	Select	O/P	
	A	D_0	D_1
1	0	1	0
1	1	0	1

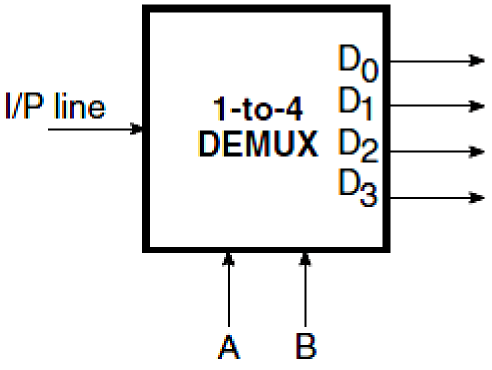


logic circuit for 1 to 2 DEMUX

1-to-4 DEMUX(2 select line)

Figure (a) shows the circuit representation of a 1-to-4 demultiplexer. Figure (b) shows the truth table of the demultiplexer when the input line is held HIGH.

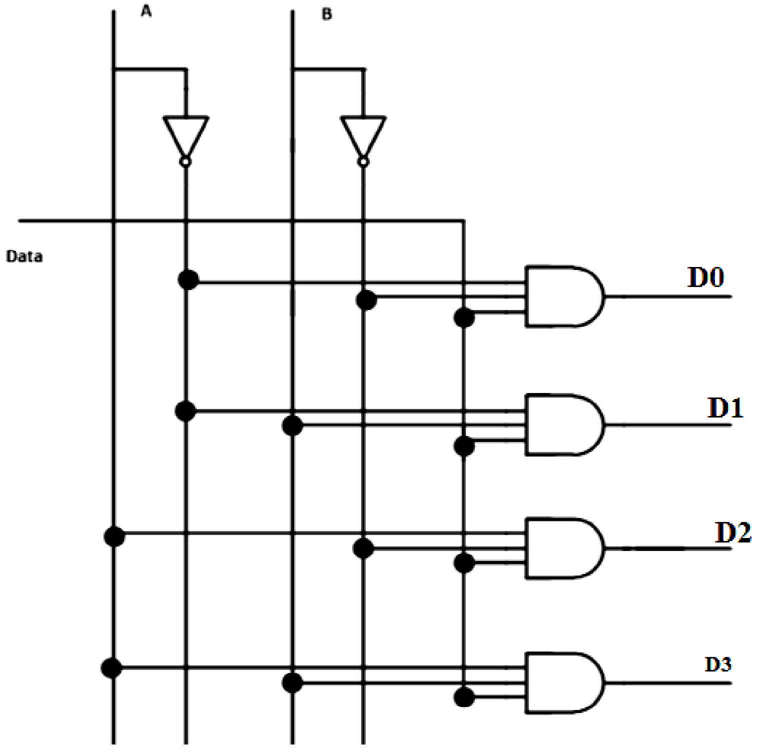
- The input data is transmitted to one of the outputs D_n by means of select signals A,B.
- The data is available at output D0 for A=0,B=0 and for A=1,B=1 the data is available at D3.
- For other combination of select lines, data is available on the respective output line



(a)

I/P	Select		O/P			
	A	B	D_0	D_1	D_2	D_3
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

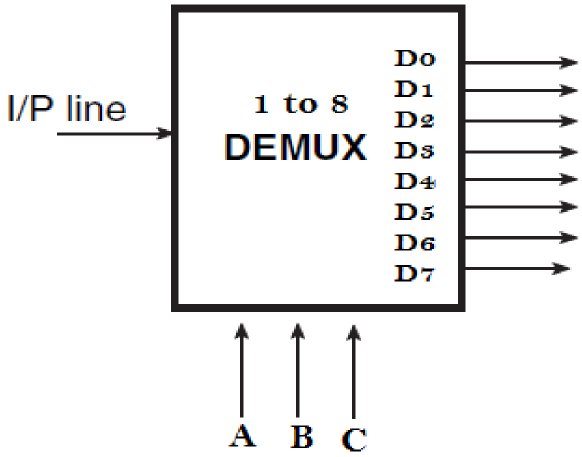
(b)



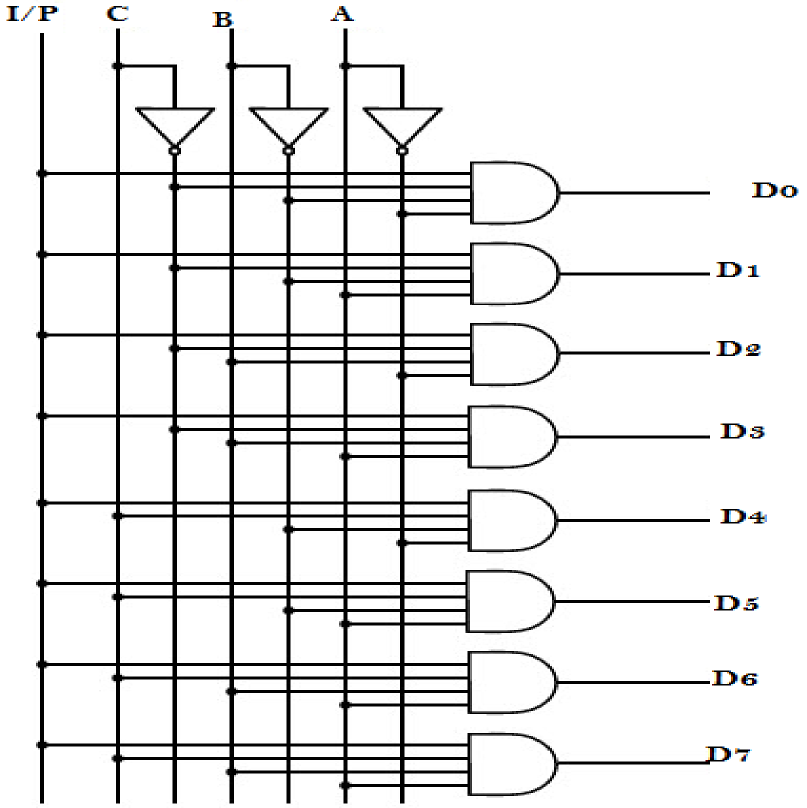
logic circuit for 1 to 4 DEMUX

1-to-8 DEMUX(3 select line)

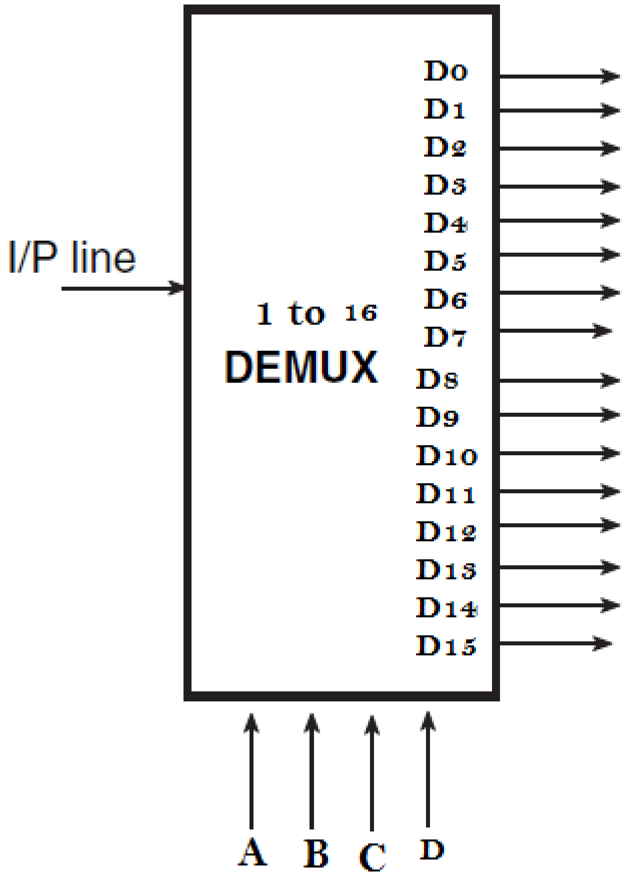
- The input data is transmitted to one of the outputs D_n by means of select signals A,B,C
- The data is available at output D0 for A=0,B=0,C=0 and for A=1,B=1,C=1 the data is available at D7.
- For other combination of select lines, data is available on the respective output line



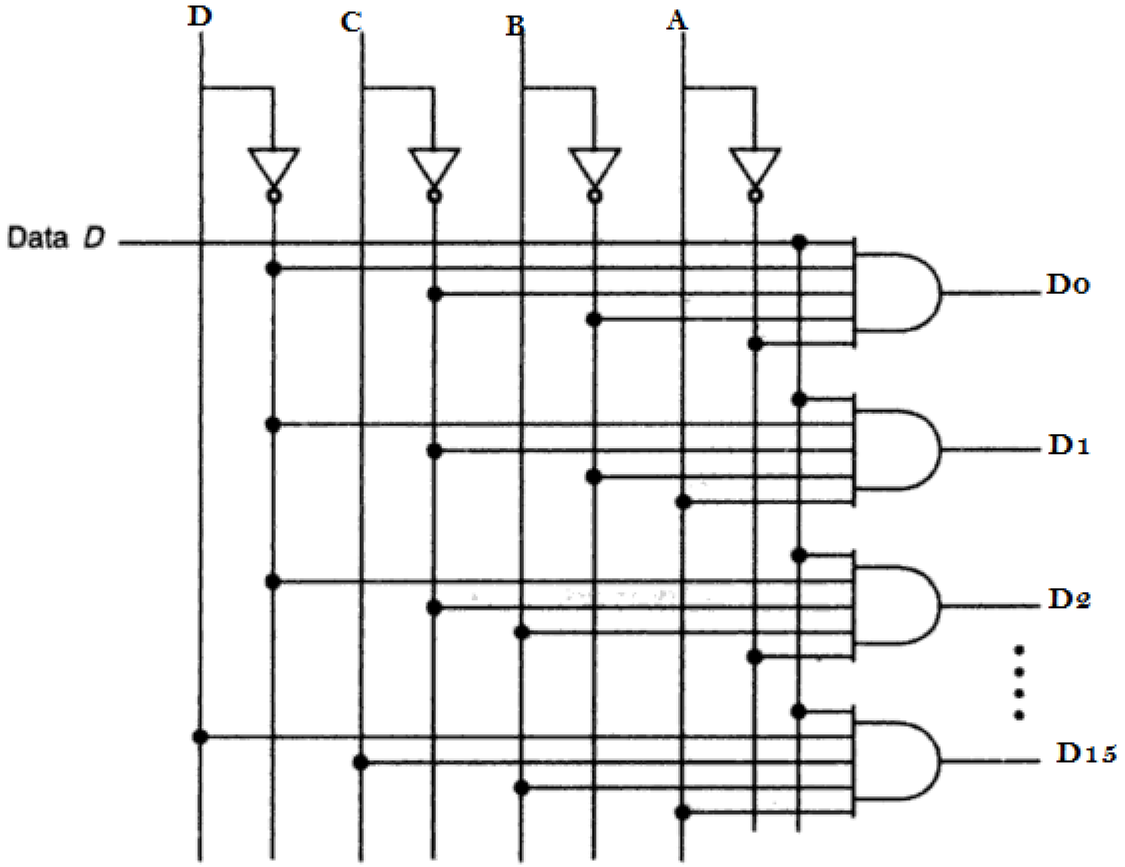
I/P	SELECT			OUTPUT							
	A	B	C	D0	D1	D2	D3	D4	D5	D6	D7
1	0	0	0	**1**	0	0	0	0	0	0	0
1	0	0	1	0	**1**	0	0	0	0	0	0
1	0	1	0	0	0	**1**	0	0	0	0	0
1	0	1	1	0	0	0	**1**	0	0	0	0
1	1	0	0	0	0	0	0	**1**	0	0	0
1	1	0	1	0	0	0	0	0	**1**	0	0
1	1	1	0	0	0	0	0	0	0	**1**	0
1	1	1	1	0	0	0	0	0	0	0	**1**



1-to-16 DEMUX(4 select line)



I/P	SELECT				OUTPUT															
	A	B	C	D	D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15
1	0	0	0	0	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	1	0	0	0	**1**	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	**1**	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	1	0	0	0	0	0	**1**	0	0	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0	**1**	0	0	0	0	0	0	0	0	0
1	0	1	1	1	0	0	0	0	0	0	0	**1**	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0	0	0	0	**1**	0	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0	0	0	0
1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0	0
1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**

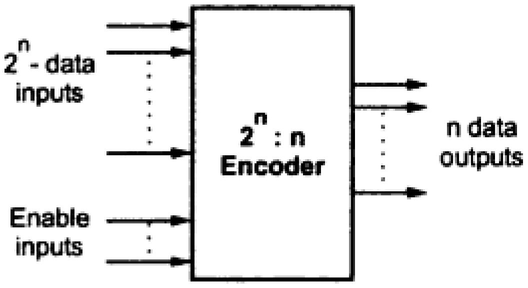


- The input data is transmitted to one of the outputs D_n by means of select signals A,B,C,D.
- The data is available at output D0 for A=0,B=0,C=0 &D=0 and for A=1,B=1,C=1,D=1 the data is available at D15.
- For other combination of select lines, data is available on the respective output line.

10. WHAT IS ENCODER? EXPLAIN IN DETAIL WITH RELEVANT LOGIC CIRCUITS AND TRUTH TABLE.

Definition

- An *encoder* is a multiplexer without its single output line. It is a combinational logic function that has 2^n (or fewer) input lines and n output lines, which correspond to n selection lines in a multiplexer.
- The n output lines generate the binary code for the possible 2^n input lines.



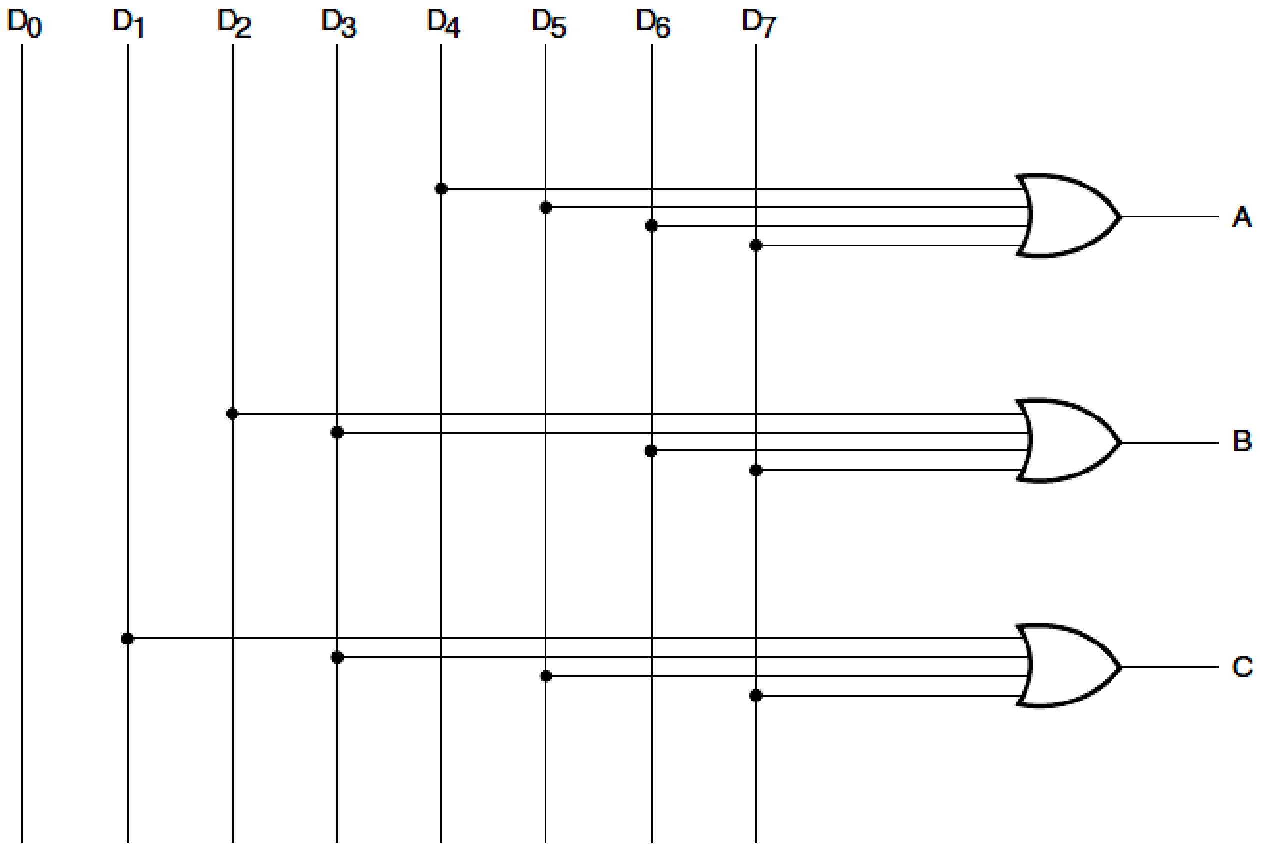
Octal-to-binary encoder.

- Such an encoder would have eight input lines, each representing an octal digit, and three output lines representing the three-bit binary equivalent.
- The truth table of such an encoder is given in Table below. In the truth table, D0 to D7 represent octal digits 0 to 7. A, B and C represent the binary digits.

Table . Truth table of an encoder.

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	A	B	C
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

- The eight input lines would have 28 = 256 possible combinations.
- However, in the case of an octal-to-binary encoder, only eight of these 256 combinations would have any meaning.
- The remaining combinations of input variables are ‘don’t care’ input combinations. Also, only one of the input lines at a time is in logic ‘1’ state.
- Figure below shows the hardware implementation of the octal-to-binary encoder described by the truth table in table.
- This circuit has the shortcoming that it produces an all 0s output sequence when all input lines are in logic ‘0’ state.
- This can be overcome by having an additional line to indicate an all 0s input sequence.



Priority Encoder

- A *priority encoder* is a practical form of an encoder.
- In this type of encoder, a priority is assigned to each input so that, when more than one input is simultaneously active, the input with the highest priority is encoded.
- Let us assume that the octal to- binary encoder described in the previous paragraph has an input priority for higher-order digits.
- Let us also assume that input lines D2, D4 and D7 are all simultaneously in logic ‘1’ state.
- In that case, only D7 will be encoded and the output will be 111. The truth table of such a priority encoder will then be modified to what is shown in Table below

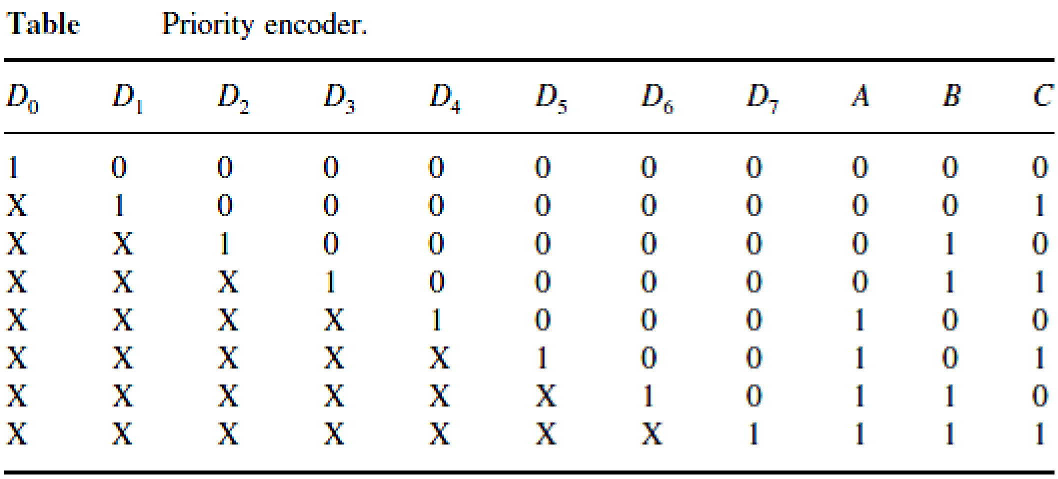
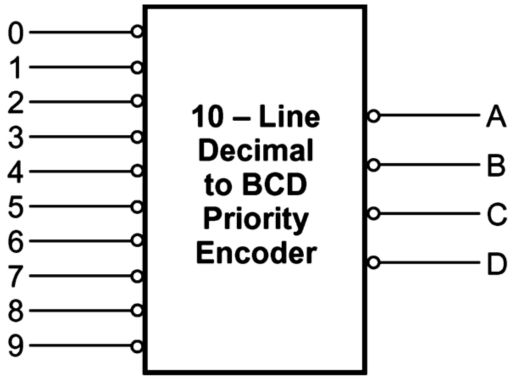


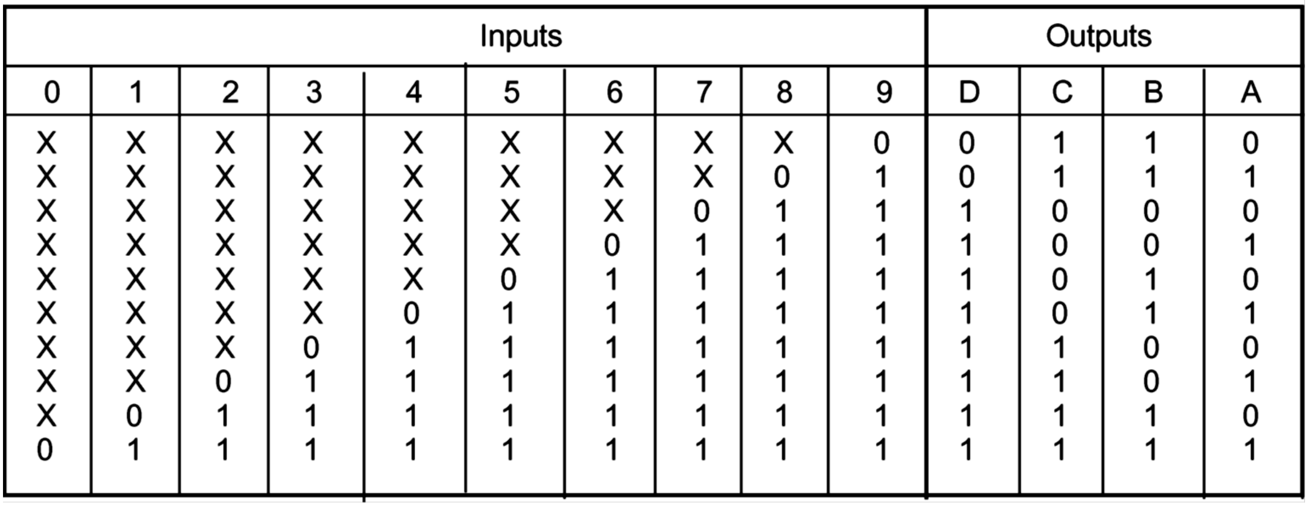
Table Priority encoder.

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	A	B	C
1	0	0	0	0	0	0	0	0	0	0
X	1	0	0	0	0	0	0	0	0	1
X	X	1	0	0	0	0	0	0	1	0
X	X	X	1	0	0	0	0	0	1	1
X	X	X	X	1	0	0	0	1	0	0
X	X	X	X	X	1	0	0	1	0	1
X	X	X	X	X	X	1	0	1	1	0
X	X	X	X	X	X	X	1	1	1	1

- Looking at the last row of the table, it implies that, if D7 = 1, then, irrespective of the logic status of other inputs, the output is 111 as D7 will only be encoded.

- As another example, Figure below shows the logic symbol and truth table of ***a 10-line decimal to four-line BCD encoder*** providing priority encoding for higher-order digits, with digit 9 having the highest priority.
- In the functional table shown, the input line with highest priority having a LOW on it is encoded irrespective of the logic status of the other input lines.





Inputs										Outputs			
0	1	2	3	4	5	6	7	8	9	D	C	B	A
X	X	X	X	X	X	X	X	X	0	0	1	1	0
X	X	X	X	X	X	X	X	0	1	0	1	1	1
X	X	X	X	X	X	X	0	1	1	1	0	0	0
X	X	X	X	X	X	0	1	1	1	1	0	0	1
X	X	X	X	X	0	1	1	1	1	1	0	1	0
X	X	X	X	0	1	1	1	1	1	1	0	1	1
X	X	X	0	1	1	1	1	1	1	1	1	0	0
X	X	0	1	1	1	1	1	1	1	1	1	0	1
X	0	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1

- Some of the encoders available in IC form provide additional inputs and outputs to allow expansion.
- IC 74148, which is an eight-line to three -line priority encoder, is an example. For instance, two 74148s can be cascaded to build a 16-line to four-line priority encoder.

11.DESIGN A FOUR-LINE TO TWO-LINE PRIORITY ENCODER WITH ACTIVE HIGH INPUTS AND OUTPUTS, WITH PRIORITY ASSIGNED TO THE HIGHER-ORDER DATA INPUT LINE.

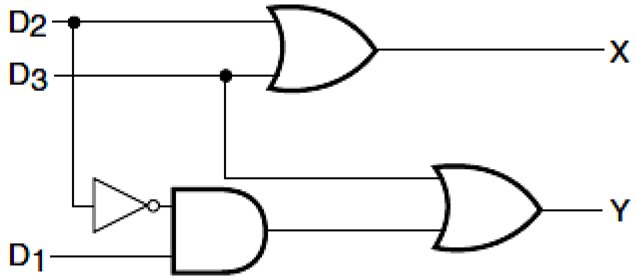
Solution

The Boolean expressions for the two output lines X and Y are given by the equations

$$X = D_2.\overline{D}_3 + D_3 = D_2 + D_3$$

$$Y = D_1.\overline{D}_2.\overline{D}_3 + D_3 = D_1.\overline{D}_2 + D_3$$

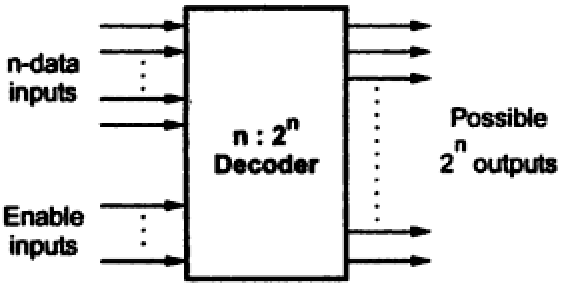
D_0	D_1	D_2	D_3	X	Y
1	0	0	0	0	0
X	1	0	0	0	1
X	X	1	0	1	0
X	X	X	1	1	1



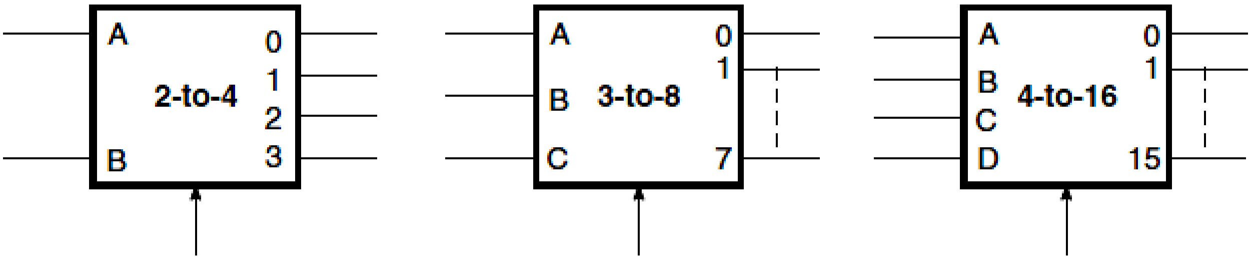
12. WHAT IS DECODER? EXPLAIN IN DETAIL WITH RELEVANT LOGIC CIRCUITS AND TRUTH TABLE.

Definition

- A decoder is a combinational circuit that decodes the information on n input lines to a maximum of 2^n unique output lines.
- It is a multiple input, multiple output logic circuit which converts coded inputs into coded outputs, where the input and output codes are generally different.

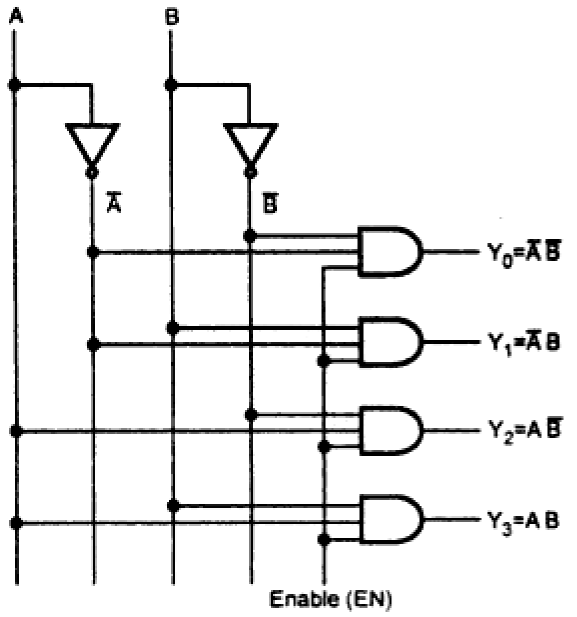


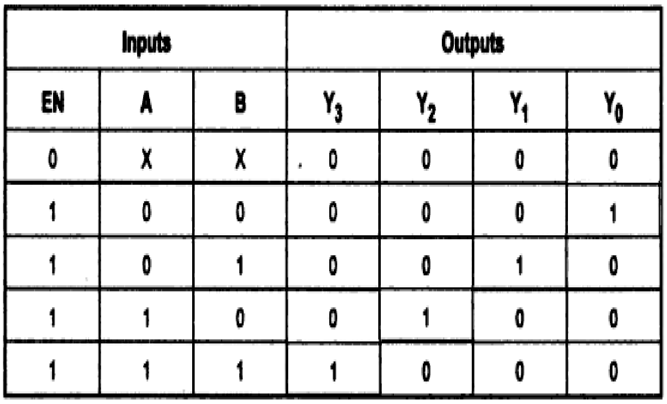
- Figure below shows the circuit representation of 2-to-4, 3-to-8 and 4-to-16 line decoders.



- If there are some unused or 'don't care' combinations in the n-bit code, then there will be fewer than 2^n output lines.
- In general, if n and m are respectively the numbers of input and output lines, then $m \leq 2^n$.
- A decoder can generate a maximum of 2^n possible minterms with an n-bit binary code.

2 to 4 decoders



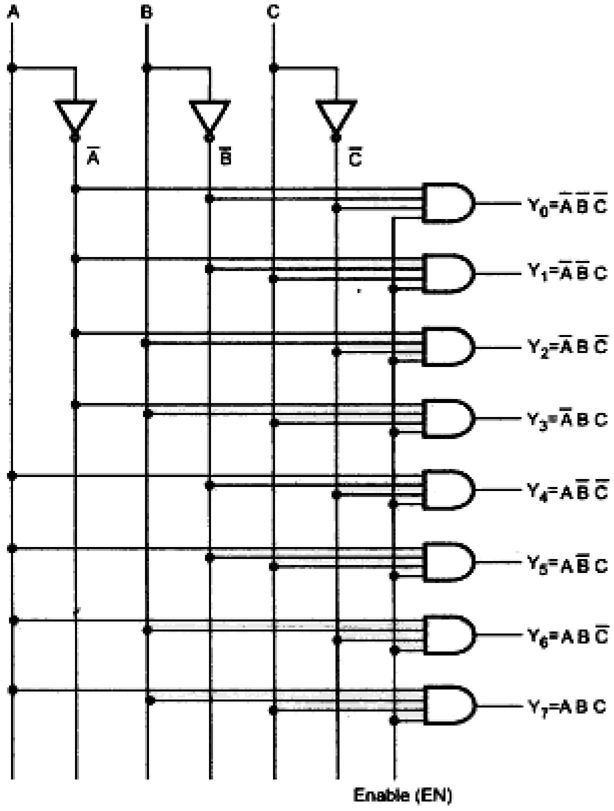


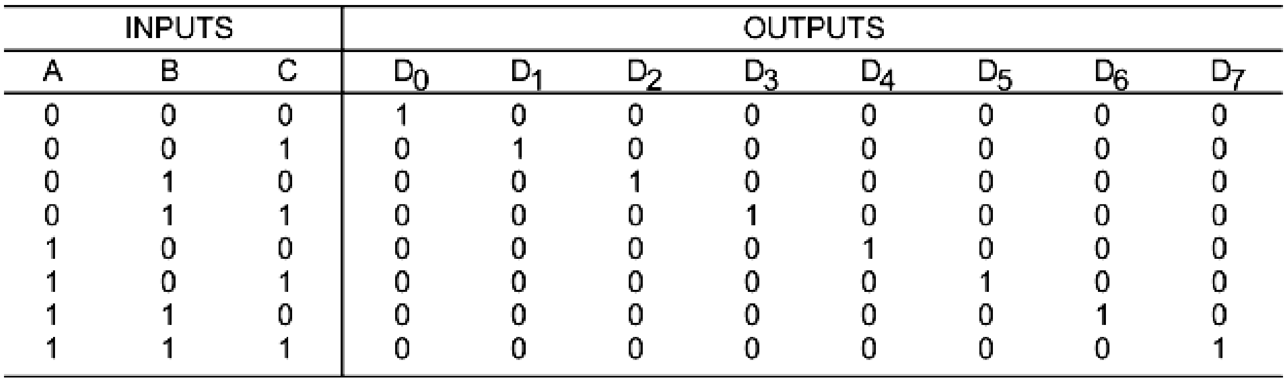
Inputs			Outputs			
EN	A	B	Y_3	Y_2	Y_1	Y_0
0	X	X	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

- Here two inputs are decoded into four outputs each output representing one of the minterms of the 2 input variables.
- The two inverters provide the complement of the inputs and each one of four AND gate generates one of the minterms.
- From the truth table if enable input is 1(EN=1) and only one of the outputs Y0 to Y3 is active for a given input.

3to 8 decoders

- Here three inputs are decoded into eight outputs each output representing one of the minterms of the three input variables.
- The three inverters provide the complement of the inputs and each one of eight AND gate generates one of the minterms.
- From the truth table if enable input is 1(EN=1) and only one of the outputs Y0 to Y7 is active for a given input.

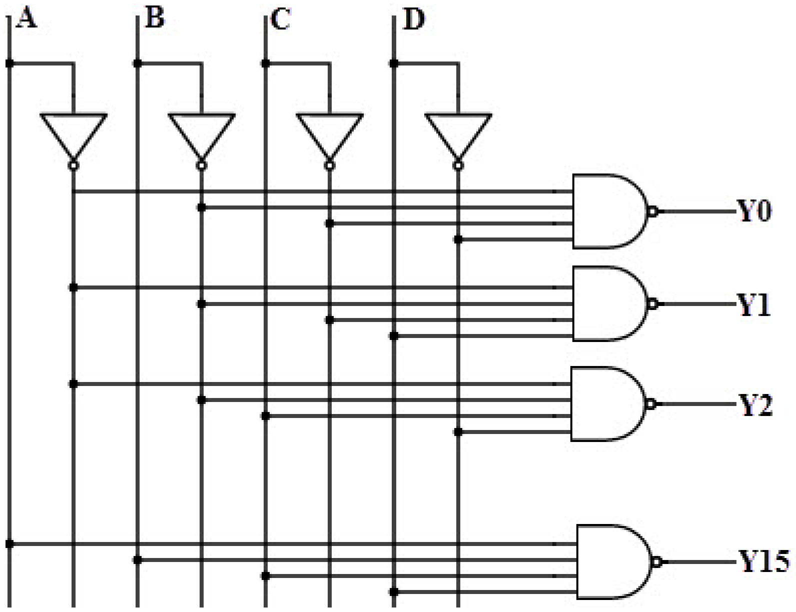

A
B
C
$\overline{A}$
$\overline{B}$
$\overline{C}$
$Y_0=\overline{A}\,\overline{B}\,\overline{C}$
$Y_1=\overline{A}\,\overline{B}\,C$
$Y_2=\overline{A}\,B\,\overline{C}$
$Y_3=\overline{A}\,B\,C$
$Y_4=A\,\overline{B}\,\overline{C}$
$Y_5=A\,\overline{B}\,C$
$Y_6=A\,B\,\overline{C}$
$Y_7=A\,B\,C$
Enable (EN)

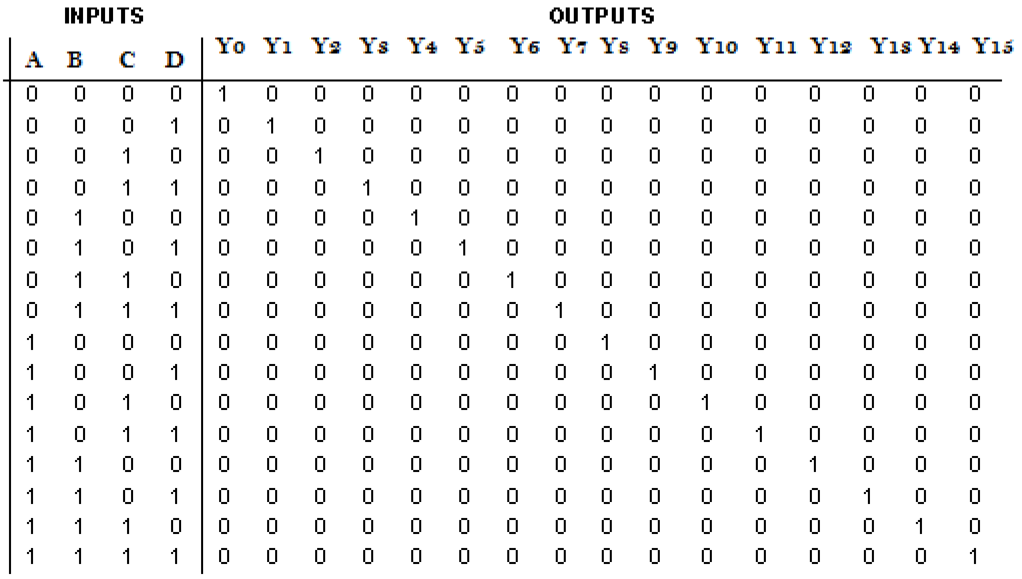


INPUTS			OUTPUTS							
A	B	C	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

4to 16 decoders

- Here four inputs are decoded into sixteen outputs each output representing one of the minterms of the four input variables.
- The four inverters provide the complement of the inputs and each one of sixteen AND gate generates one of the minterms.
- From the truth table if enable input is 1(EN=1) and only one of the outputs Y0 to Y15 is active for a given input.



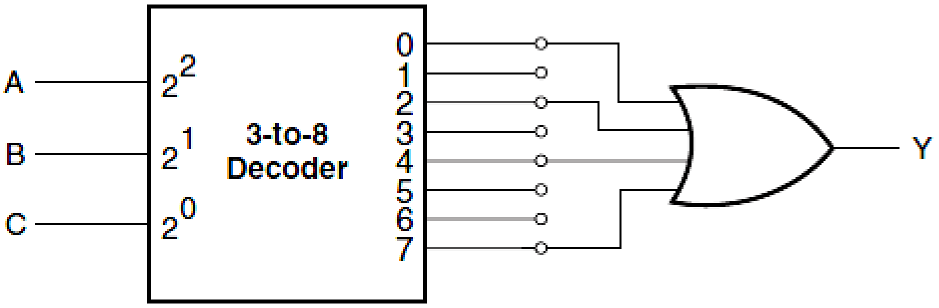


INPUTS				OUTPUTS															
A	B	C	D	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	Y_9	Y_{10}	Y_{11}	Y_{12}	Y_{13}	Y_{14}	Y_{15}
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Implementing Boolean Functions with Decoders

- A decoder can be conveniently used to implement a given Boolean function.
- The decoder generates the required minterms and an external OR gate is used to produce the sum of minterms.
- Figure below shows the logic diagram where a 3-to-8 line decoder is used to generate the Boolean function given by the equation

$$Y = A.\overline{B}.\overline{C} + \overline{A}.B.\overline{C} + A.B.C + \overline{A}.\overline{B}.\overline{C}$$



Implementing Boolean functions with decoders.

13.DISCUSS ABOUT VARIOUS TYPES OF FLIP FLOPS[ ]

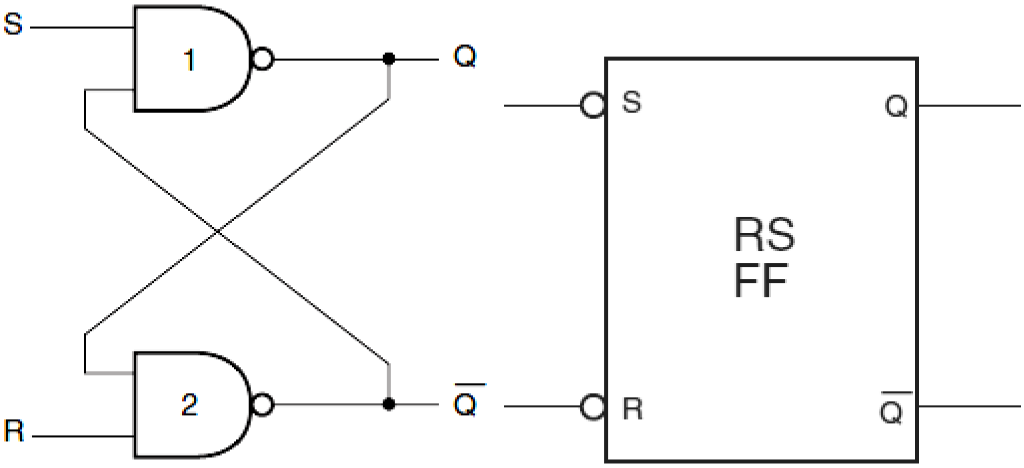
Definition

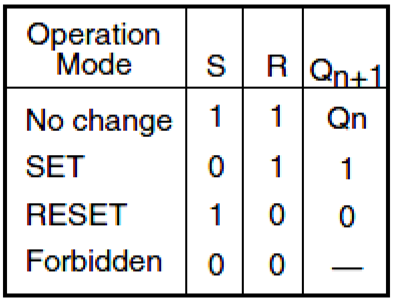
- A device that exhibits two stable states is extremely useful as a memory element in a binary system.
- Any electrical circuit that has this characteristics falls into the category of the device known as Flip Flop.
- While a logic gate is the most basic building block of combinational logic, its counterpart in sequential logic is the flip-flop.

RS flip flop

- The R-S flip-flop is the most basic of all flip-flops. The letters ‘R’ and ‘S’ here stand for RESET and SET.
- When the flip-flop is SET, its Q output goes to a ‘1’ state, and when it is RESET it goes to a ‘0’ state.
- The $\overline{Q}$ output is the complement of the Q output at all times.

R-S Flip-Flop with Active LOW Inputs





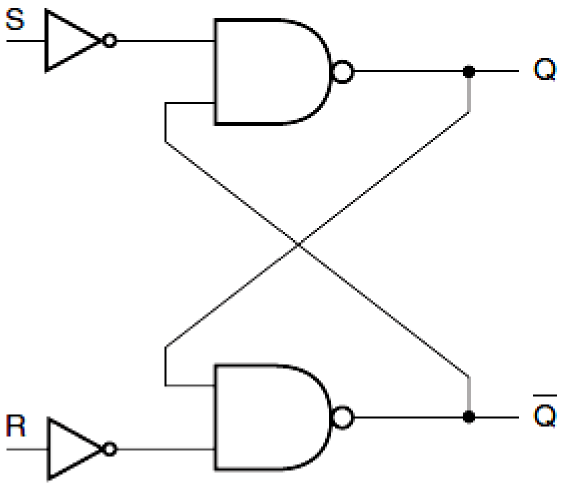
Operation Mode	S	R	Q_{n+1}
No change	1	1	Q_n
SET	0	1	1
RESET	1	0	0
Forbidden	0	0	—

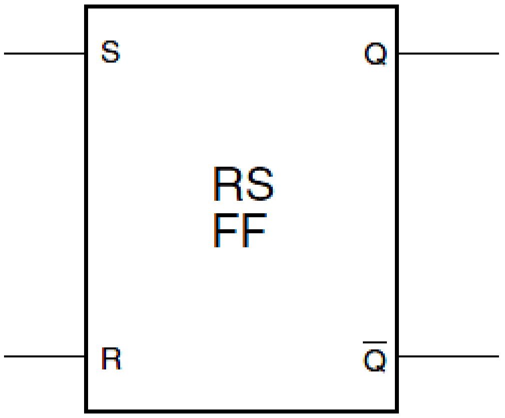
- ➢ Figure above shows a NAND gate implementation of an R-S flip-flop with active LOW inputs.
- ➢ The two NAND gates are cross-coupled. That is, the output of NAND 1 is fed back to one of the inputs of NAND 2, and the output of NAND 2 is fed back to one of the inputs of NAND 1.
- ➢ The remaining inputs of NAND 1 and NAND 2 are the S and R inputs. The outputs of NAND 1 and NAND 2 are respectively Q and Q outputs.

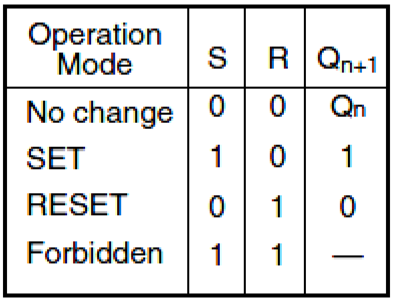
Operation of the R-S flip-flop

1. SET=RESET= 1 is the normal resting condition of the flip-flop. It has no effect on the output state of the flip-flop. Both Q and Q outputs remain in the logic state they were in prior to this input condition.
2. SET = 0 and RESET = 1 sets the flip-flop. Q and Q respectively go to the ‘1’ and ‘0’ state.
3. SET =1 and RESET =0 resets or clears the flip-flop. Q and Q respectively go to the ‘0’ and ‘1’ state.
4. SET = RESET = 0 is forbidden as such a condition tries to set (that is, Q = 1) and reset (that is, Q = 1) the flip-flop at the same time. To be more precise, SET and RESET inputs in the R-S flip-flop cannot be active at the same time.

R-S Flip-Flop with Active HIGH Inputs





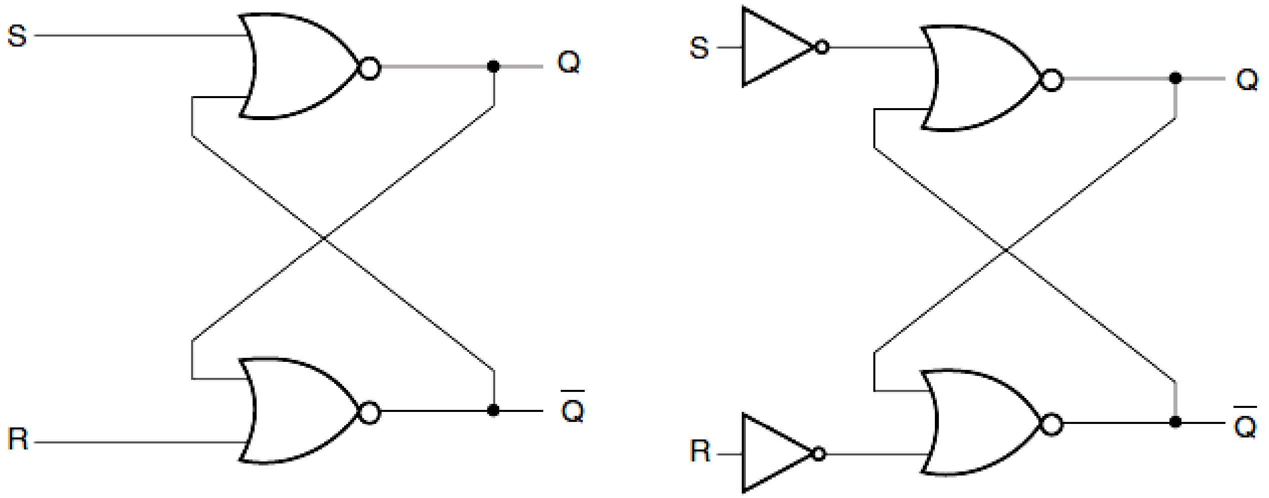


Operation Mode	S	R	Q_{n+1}
No change	0	0	Q_n
SET	1	0	1
RESET	0	1	0
Forbidden	1	1	—

- Figure above shows another NAND gate implementation of the R-S flip-flop. and its circuit symbol and function table. Such a circuit would have active HIGH inputs.
- The input combination R = S = 1 would be forbidden as SET and RESET inputs in an R-S flip-flop cannot be active at the same time.

NOR implementation of R-S Flip-Flop

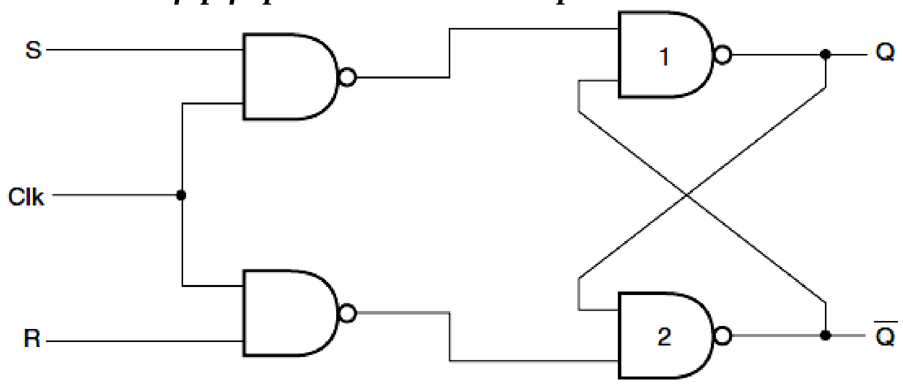
The R-S flip-flops (or latches may also be implemented with NOR gates as shown below



Clocked R-S *Flip-Flop*

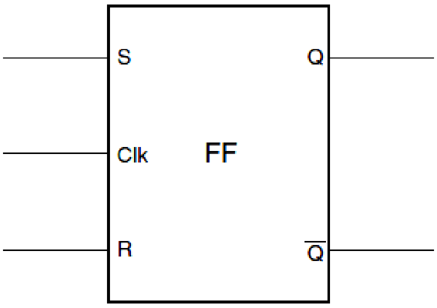
- In the case of a clocked R-S flip-flop, or for that matter any clocked flip-flop, the outputs change states as per the inputs only on the occurrence of a clock pulse.
- The clocked flip-flop could be a level-triggered one or an edge-triggered one.

Clocked R-S flip-flop with active HIGH inputs.

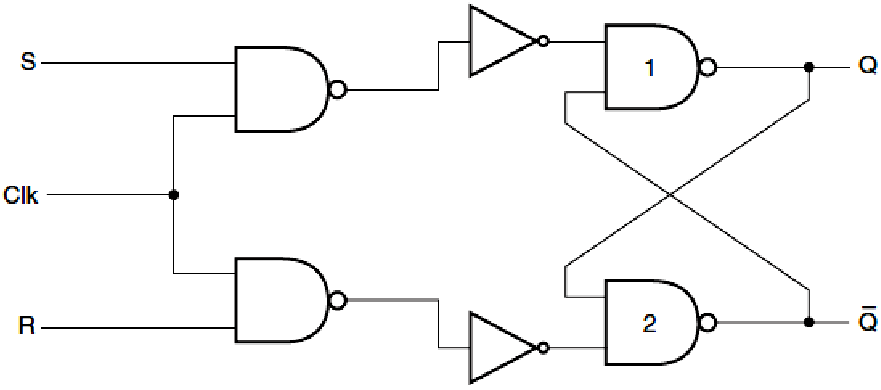


S	R	Clk	Q_{n+1}
0	0	0	Q_n
0	0	1	Q_n
0	1	0	Q_n
0	1	1	0
1	0	0	Q_n
1	0	1	1
1	1	0	Q_n
1	1	1	Invalid

- The two NAND gates at the input have been used to couple the R and S inputs to the flip-flop inputs under the control of the clock signal.
- When the clock signal is HIGH, the two NAND gates are enabled and the S and R inputs are passed on to flip-flop inputs with their status complemented.
- The outputs can now change states as per the status of R and S at the flip-flop inputs.
- For instance, when S = 1 and R = 0 it will be passed on as 0 and 1 respectively when the clock is HIGH.
- When the clock is LOW, the two NAND gates produce a ‘1’ at their outputs, irrespective of the S and R status.
- This produces a logic ‘1’ at both inputs of the flip-flop, with the result that there is no effect on the output states.



Clocked R-S flip-flop with active LOW inputs.



S	R	Clk	Q_{n+1}
0	0	0	Q_n
0	0	1	Invalid
0	1	0	Q_n
0	1	1	1
1	0	0	Q_n
1	0	1	0
1	1	0	Q_n
1	1	1	Q_n

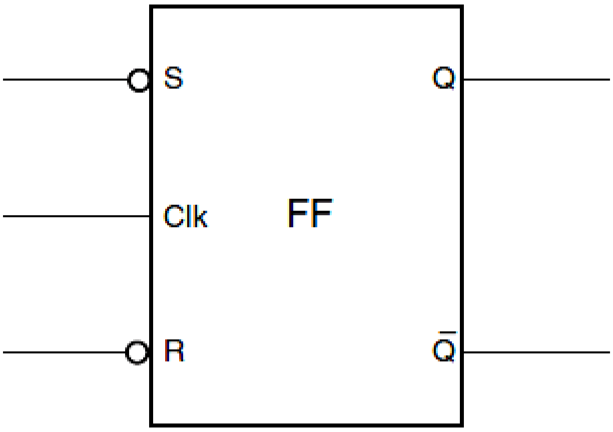
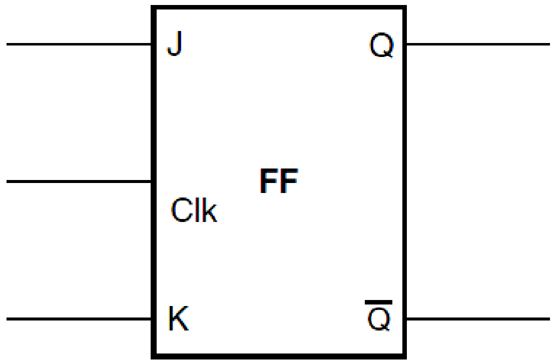


Figure above shows the clocked R-S flip-flop with active LOW R and S inputs

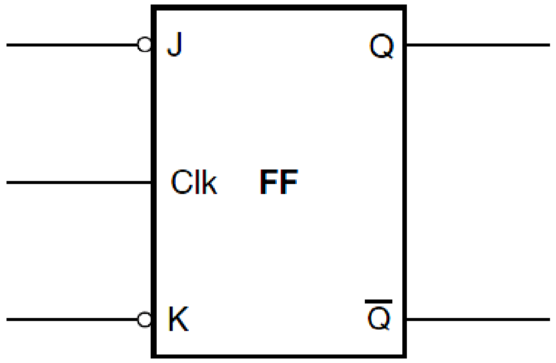
JK flip flop

- A J-K flip-flop behaves in the same fashion as an R-S flip-flop except for one of the entries in the function table.
- In the case of an R-S flip-flop, the input combination S = R = 1 (in the case of a flip-flop with active HIGH inputs) and the input combination S = R = 0 (in the case of a flip-flop with active LOW inputs) are prohibited.
- In the case of a J-K flip-flop with active HIGH inputs, the output of the flip-flop toggles, that is, it goes to the other state, for J = K = 1 .
- The output toggles for J = K = 0 in the case of the flip-flop having active LOW inputs.
- Thus, a J-K flip-flop overcomes the problem of a forbidden input combination of the R-S flip-flop.



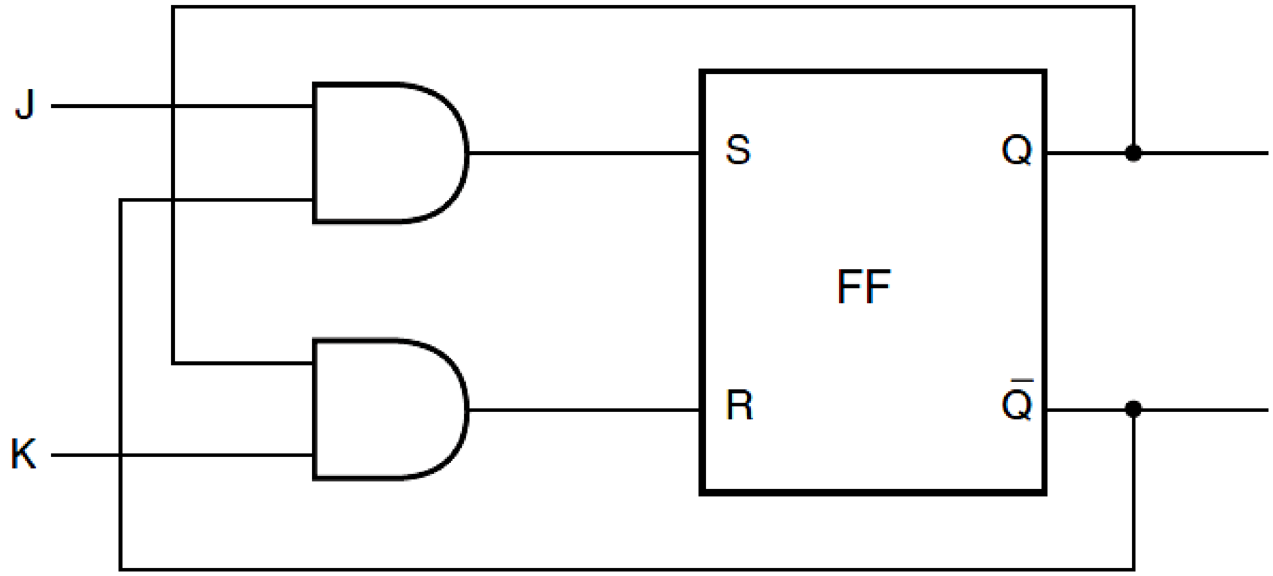
Operation Mode	J	K	Clk	Q_{n+1}
SET	1	0	1	1
RESET	0	1	1	0
NO CHANGE	0	0	1	Q_n
TOGGLE	1	1	1	$\overline{Q}_n$

(a)



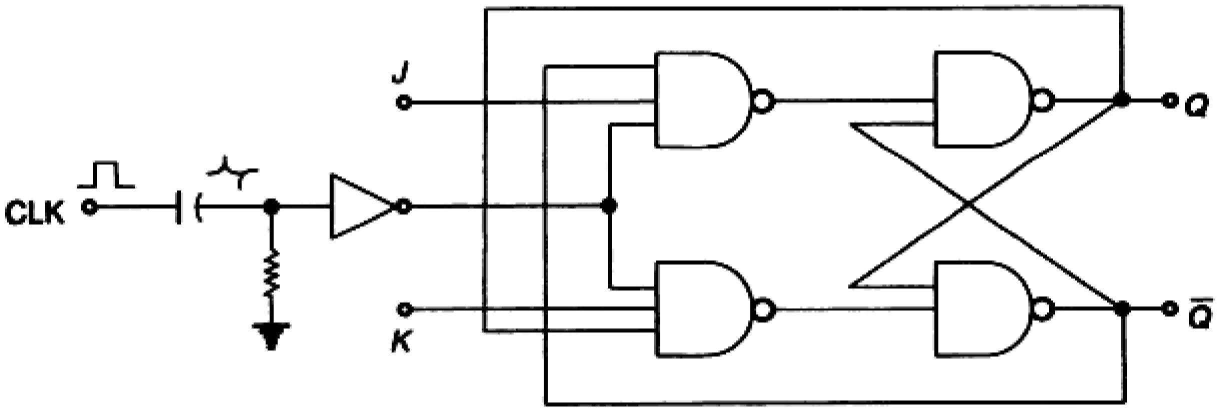
Operation Mode	J	K	Clk	Q_{n+1}
SET	0	1	1	1
RESET	1	0	1	0
NO CHANGE	1	1	1	Q_n
TOGGLE	0	0	1	$\overline{Q}_n$

- Figures above shows the circuit symbol of level-triggered J-K flip-flops with active HIGH and active LOW inputs, along with their function tables.
- Figure below shows the realization of a J-K flip-flop with an R-S flip-flop.



Realization of a J-K flip-flop with an R-S flip-flop.

Figure shows the negative edge triggered JK flip-flop. The flip-flop is inactive when the clock is low, high, or on its positive going edge.

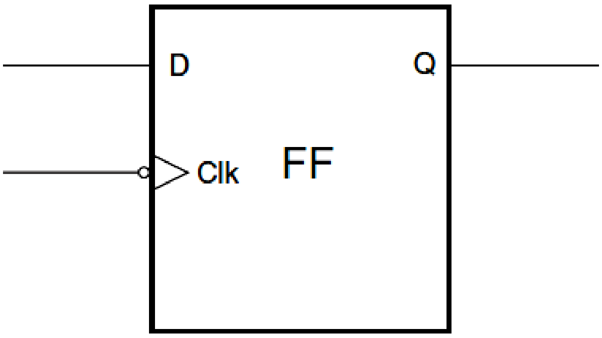


Negative edge triggered JK flip-flop

When $J = 0$ and $K = 0$, the circuit is inactive. When $J = 0$ and $K = 1$, the negative going edge of the clock pulse puts the outputs at $Q = 0$ and $Q' = 1$. When $J = 1$ and $K = 0$, the negative going edge of the clock pulse puts the Q outputs at $Q = 1$ and $Q' = 0$. When $J = 1$ and $K = 1$, the outputs Q and Q' toggles or alternate with each negative going clock edge.

D flip flop

- A D flip-flop, also called a *delay flip-flop*, can be used to provide temporary storage of one bit of information.
- Figure (a) shows the circuit symbol and function table of a negative edge-triggered D flip-flop.
- When the clock is active, the data bit (0 or 1) present at the D input is transferred to the output.
- In the D flip-flop of Fig. 10.39, the data transfer from D input to Q output occurs on the negative-going (HIGH-to-LOW) transition of the clock input.
- The D input can acquire new status when the clock is inactive, which is the time period between successive HIGH-to-LOW transitions.
- The D flip-flop can provide a maximum delay of one clock period.



(a)

D	Clk	Q
0	↓	0
1	↓	1

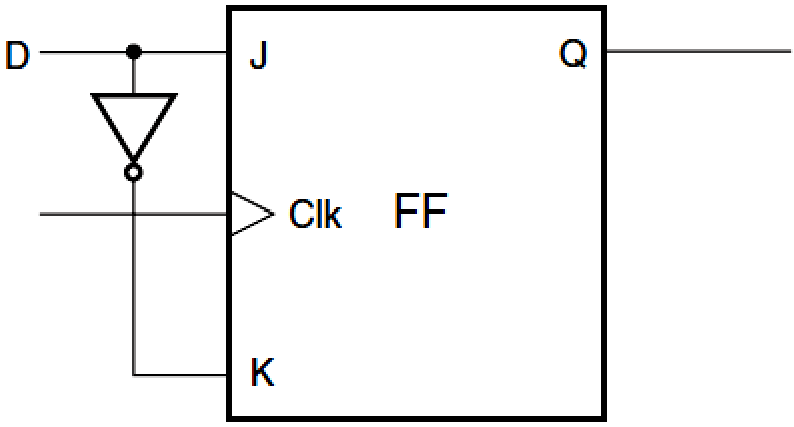
(b)

Q_n	D	Q_{n+1}
0	0	0
0	1	1
1	0	0
1	1	1

(c)

J-K *Flip-Flop as D Flip-Flop*

- Figure 10.40 shows how a J-K flip-flop can be used as a D flip-flop.
- When the D input is a logic ‘1’, the J and K inputs are a logic ‘1’ and ‘0’ respectively.
- According to the function table of the J-K flip-flop, under these input conditions, the Q output will go to the logic ‘1’ state when clocked.
- Also, when the D input is a logic ‘0’, the J and K inputs are a logic ‘0’ and ‘1’ respectively.
- Again, according to the function table of the J-K flip-flop, under these input conditions, the Q output will go to the logic ‘0’ state when clocked.
- Thus, in both cases, the D input is passed on to the output when the flip-flop is clocked.



Toggle Flip-Flop (T Flip-Flop)

The output of a *toggle flip-flop*, also called a T flip-flop, changes state every time it is triggered at its T input, called the toggle input. That is, the output becomes ‘1’ if it was ‘0’ and ‘0’ if it was ‘1’.

T flip-flop This changes state with each clock pulse and hence, it acts as a toggle switch. If $J = K = 1$, then output is the complement of the previous state, so that the *JK* flip-flop is converted into a T flip-flop. The realisation of a T flip-flop from a *JK* flip-flop is shown in Fig. 20.27.

The truth-table for a positive edge-triggered *T* flip-flop is shown in Table 20.20.

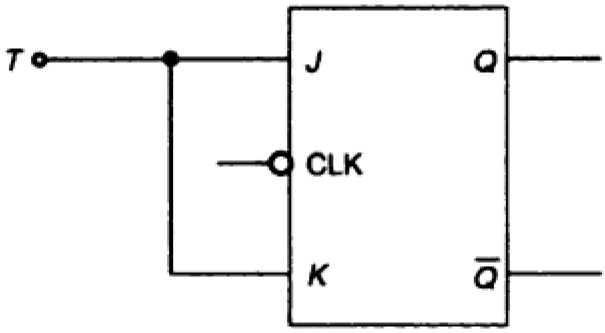


Fig. 20.27 Realisation of a T flip-flop from a JK flip-flop

Table 20.20 Truth table for positive edge triggered T flip-flop

Clock	*T*	Q_N
x	0	No change
x	1	No change
↑	0	No change
↑	1	$\overline{Q}_{N-1}$

srivenkateshwaraa
College of Engineering & Technology
(Approved by AICTE, New Delhi & Affiliated to Pondicherry University, Puducherry)
13-A, Pondy - Villupuram Main Road, Ariyur, Puducherry - 605 102.
ASPIRE TO EXCEL

ISO 21001

UNIT IV: SEQUENTIAL CIRCUITS:

Design of counters using Flip-flops– Synchronous, asynchronous, Up/Down counters, decade counter, ring counter, Johnson counter, BCD counter–Shift registers and bi-directional shift registers. Parallel/serial converters. Memory types and terminology – ROM – RAMs – Non-volatile RAMS – Sequential memories.

PART A - 2 Marks

TWO MARK QUESTIONS AND ANSWERS

1. **Define registers.(Jan'11)**

   A register is a group of flip-flops flip-flop can store one bit information. So an n-bit register has a group of n flip-flops and is capable of storing any binary information/number containing n-bits.

2. **Define shift registers.**

   The binary information in a register can be moved from one stage to other stage within the register or into or out of the register upon application of clock pules. This type of registers called shift registers.

3.What do you mean by present state?

The information stored in the memory elements at any given time defines the present state of the sequential circuit.

4. What do you mean by next state?

The present state and the external inputs determine the outputs and the next state of the sequential circuit.

5. Define synchronous sequential circuit

In synchronous sequential circuits, signals can affect the memory elements only at discrete instant of time.

6. Define Asynchronous sequential circuit?

In asynchronous sequential circuits change in input signals can affect memory element at any instant of time

7. Give the comparison between synchronous & Asynchronous sequential circuits?

Synchronous sequential circuits	Asynchronous sequential circuits
Memory elements are clocked flip-flops	Memory elements are either unlocked flip-flops or time delay elements.
Easier to design	More difficult to design
Speed of operation is limited by its clock	No limitation

8.What is fundamental mode sequential circuit?

- Input variables changes if the circuit is stable
- Input are levels, not pulses
- Only one input can change at a given time

9. What is meant by clocked sequential circuits? (April ' 13)

Sequential circuits use current input variables and previous input variables by storeing the information and putting back into the circuit on the next clock (activation) cycle.

10. What are pulse mode circuit?

-inputs are pulses

-width of pulses are long for circuit to respond to the input

-pulse width must not be long that it is still present after the new state is reached

11. **What are the significance of state assignment?**

In synchronous circuit-state assignments are made with the objective of circuit reduction

Asynchronous circuits-its objective is to avoid critical races

12. When do race conditions occur?

-two or more binary state variables change their value in response to the change in i/p variable

13. What is non critical race?

-final stable state does not depend on the order in which the state variable changes

-race condition is not harmful

14. What is critical race?

-final stable state depends on the order in which the state variable changes

-race condition is harmful

15. When does a cycle occur?

-asynchronous circuit makes a transition through a series of unstable state

16. What are the different techniques used in state assignment?

-shared row state assignment

-one hot state assignment

17. What are the steps for the design of asynchronous sequential circuit?

- ✓ Construction of primitive flow table
- ✓ Reduction of flow table
- ✓ State assignment is made
- ✓ Realization of primitive flow table

18. What is hazard?

-unwanted switiching transients

19. What is static 1 hazard?

-output goes momentarily 0 when it should remain at 1

20. What is static 0 hazards?

-output goes momentarily 1 when is should remain at 0

21. What is dynamic hazard?

-output changes 3 or more times when it changes from 1 to 0 or 0 to 1

22. What is the cause for essential hazards?

-unequal delays along 2 or more path from same input

23. What is flow table?

-state table of an synchronous sequential network

24. Define merger graph.

The merger graph is defined as follows. It contains the same number of vertices as the state table contains states. A line drawn between the two state vertices indicate each compatible state pair. It two states are incompatible no connecting line is drawn.

25. Define incompatibility

The states are said to be incompatible if no line is drawn in between them. If implied states are incompatible, they are crossed & the corresponding line is ignored.

26. Define state table.

For the design of sequential counters we have to relate present states and next states. The table, which represents the relationship between present states and next states, is called state table.

27. Define total state

The combination of level signals that appear at the inputs and the outputs of the delays define what is called the total state of the circuit.

28. Define primitive flow table: (Jan'11)

It is defined as a flow table which has exactly one stable state for each row in the table. The design process begins with the construction of primitive flow table.

29. What are the types of asynchronous circuits?

1. Fundamental mode circuits

2. Pulse mode circuits

30. What are races?

When 2 or more binary state variables change their value in response to a change in an input variable, race condition occurs in an asynchronous sequential circuit. In case of unequal delays, a race condition may cause the state variables to change in an unpredictable manner.

31. Define non critical race.

If the final stable state depends on the order in which the state variable changes, the race condition is not harmful and it is called a non critical race.

32. Define critical race?

If the final stable state depends on the order in whichthe state variable changes, the race condition is harmful and it is called a critical race.

33. What is a cycle?

A cycle occurs when an asynchronous circuit makes a transition through a series of unstable states. If a cycle does not certain a stable state, the circuit will go from one unstable to stable to another, until the inputs are changed.

34. Write a short note on fundamental mode asynchronous circuit.

Fundamental mode circuit assumes that. The input variables change only when the circuit is stable. Only one input variable can change at a given time and input are levels and not pulses.

35. Write a short note on pulse mode circuit.

Pulse mode circuit assumes that the input variables are pulses instead of level. the width of the pulses is long enough for the circuit to respond to the input and pulse width must not be so long that it is still present after the new state is reached.

36. Define the term counter. (April ’13)

A counter is used to count pulse and give the output in binary form.A register that goes through a prescribed sequence of states upon the application of input pulses is called a counter.

37. State ripple counter(Jan ’11)

A ripple counter is nothing but an asynchronous counter, in which the output of the flip -flop change state like a ripple in water

11 MARKS

1. Describe in detail about synchronous Counter &its types.

- The ripple or asynchronous counter is the simplest to build, but its highest operating frequency is limited because of ripple action.
- Each flip-flop has a delay time. In ripple counters these delay times are additive and the total “settling” time for the counter is approximately the product of the delay time of a single flip-flop and the total number of flip-flops.
- Again, there is the possibility of glitches occurring at the output of decoding gates used with a ripple counter.
- Both of these problems can be overcome, if all the flip-flops are clocked synchronously. The resulting circuit is known as a *synchronous counter*.

Synchronous counters can be designed for any count sequence (need not be straight binary). These can be designed following a systematic approach.

DESIGN PROCEDURE OF A SYNCHRONOUS COUNTER

Following certain general steps, synchronous counters of any given count sequence and modulus can be designed. The steps are listed below:

Step 1. From the given word description of the problem, draw a state diagram that describes the operation of the counter.

Step 2. From the state table, write the count sequences in the form of a table

Step 3. Find the number of flip-flops required.

Step 4. Decide the type of flip-flop to be used for the design of the counter. Then determine the flip-flop inputs that must be present for the desired next state from the present state using the excitation table of the flip-flops.

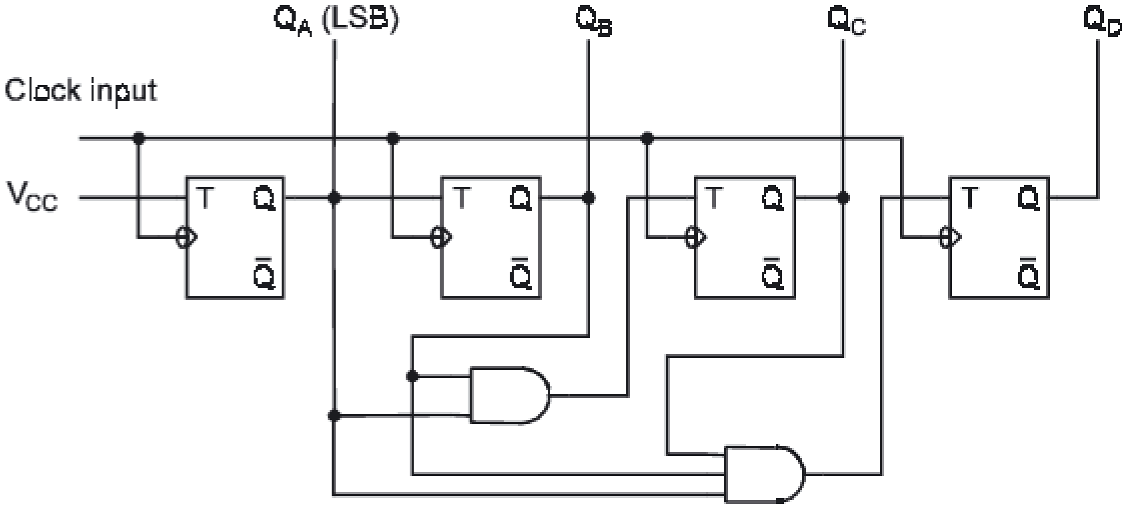
Step 5. Prepare K-maps for each flip-flop input in terms of flip-flop outputs as the input variables. Simplify the K-maps and obtain the minimized expressions.

Step 6. Connect the circuit using flip-flops and other gates corresponding to the minimized expressions.

SYNCHRONOUS UP COUNTER

- In this circuit the clock inputs of all the flip-flops are tied together so that the input clock signal may be applied simultaneously to each flip-flop.
- From the circuit, we can see that flip-flop A changes its state with the negative transition of each clock pulse.
- Flip-flop B changes its state only when the value of QA is 1 and a negative transition of the clock pulse takes place.
- Similarly, flip-flop C changes its state only when both QA and QB are 1 and a negative edge transition of the clock pulse takes place.

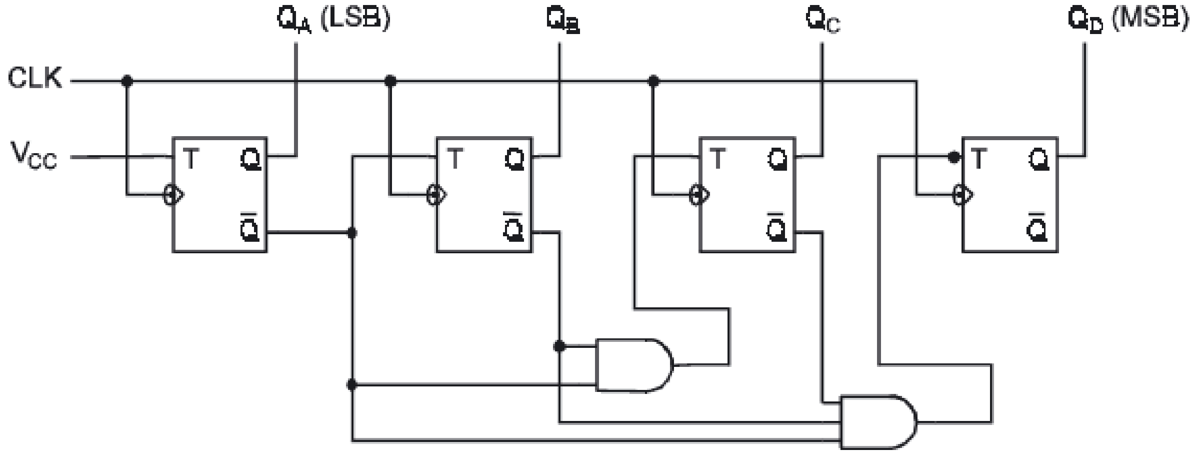
- In the same manner, the flip-flop D changes its state when QA = QB = QC = 1 and when there is a negative transition at clock input. The count sequence of the counter is given in Table



State	Q_D	Q_C	Q_B	Q_A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
0	0	0	0	0

SYNCHRONOUS DOWN-COUNTER

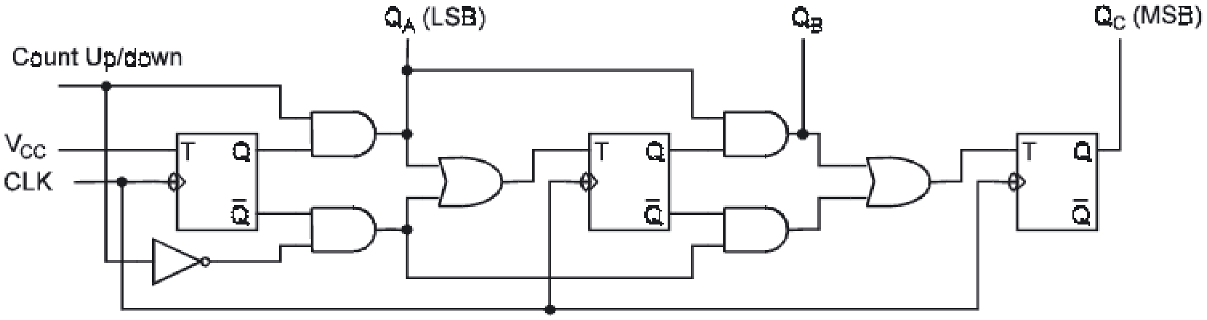
- A parallel down-counter can be made to count down by using the inverted outputs of flip-flops to feed the various logic gates. Even the same circuit may be retained and the outputs may be taken from the complement outputs of each flip-flop.
- The parallel counter can be converted to a down-counter by connecting the Q′A, Q′B, and Q′C outputs to the AND gates in place of QA, QB, and QC respectively as shown in Figure.



$State$	Q_D	Q_C	Q_B	Q_A
15	1	1	1	1
14	1	1	1	0
13	1	1	0	1
12	1	1	0	0
11	1	0	1	1
10	1	0	1	0
9	1	0	0	1
8	1	0	0	0
7	0	1	1	1
6	0	1	1	0
5	0	1	0	1
4	0	1	0	0
3	0	0	1	1
2	0	0	1	0
1	0	0	0	1
0	0	0	0	0
15	1	1	1	1

SYNCHRONOUS UP-DOWN COUNTER

- Combining both the functions of up- and down-counting in a single counter, we can make a synchronous up-down counter.
- Here the control input (countup/ down) is used to allow either the normal output or the inverted output of one flip-flop to the T input of the following flip-flop.
- Two separate control lines (count-Up and count- down) could have been used but in such case we have to be careful that both of the lines cannot be simultaneously in the high state.
- When the count-up/down line is high, then the upper AND gates will be active and the lower AND gates will remain inactive and hence the normal output of each flip-flop is carried forward to the following flip-flop.
- In such case, the counter will count from 000 to 111. On the other hand, if the control line is low, then the upper AND gates remain inactive, while the lower AND gates will become active. So the inverted output comes into operation and the counter counts from 111 to 000.



2.Explain in detail about ring counter.

Ring Counters

- Ring counters are generated from shift registers.
- The counter is constructed by (1) connecting the last output of the shift register to the input of the first flip-flop, and (2) initializing exactly one bit in the register to 1 (the first bit).

- With these conditions, on the applications of clock pulses, the 1 bit is shifted to adjacent flipflops in a circular fashion.

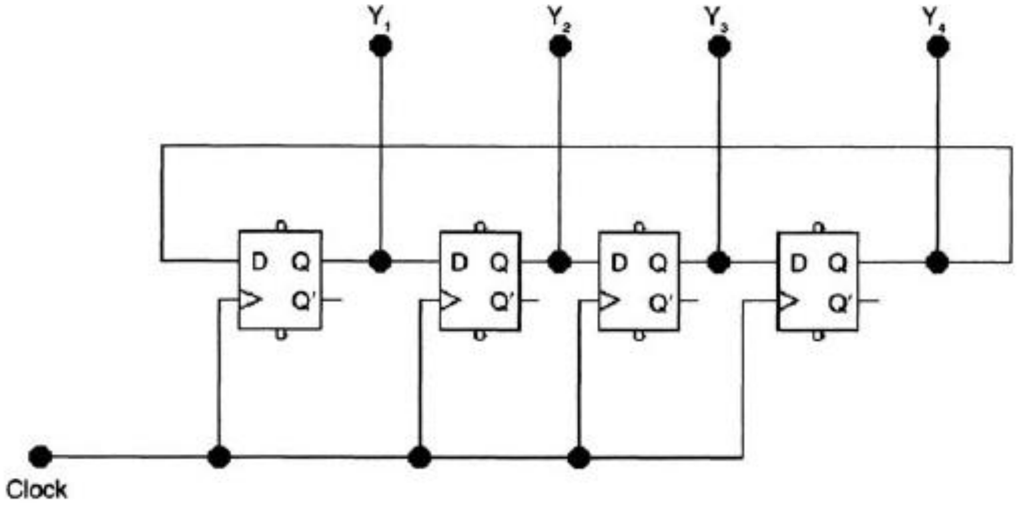
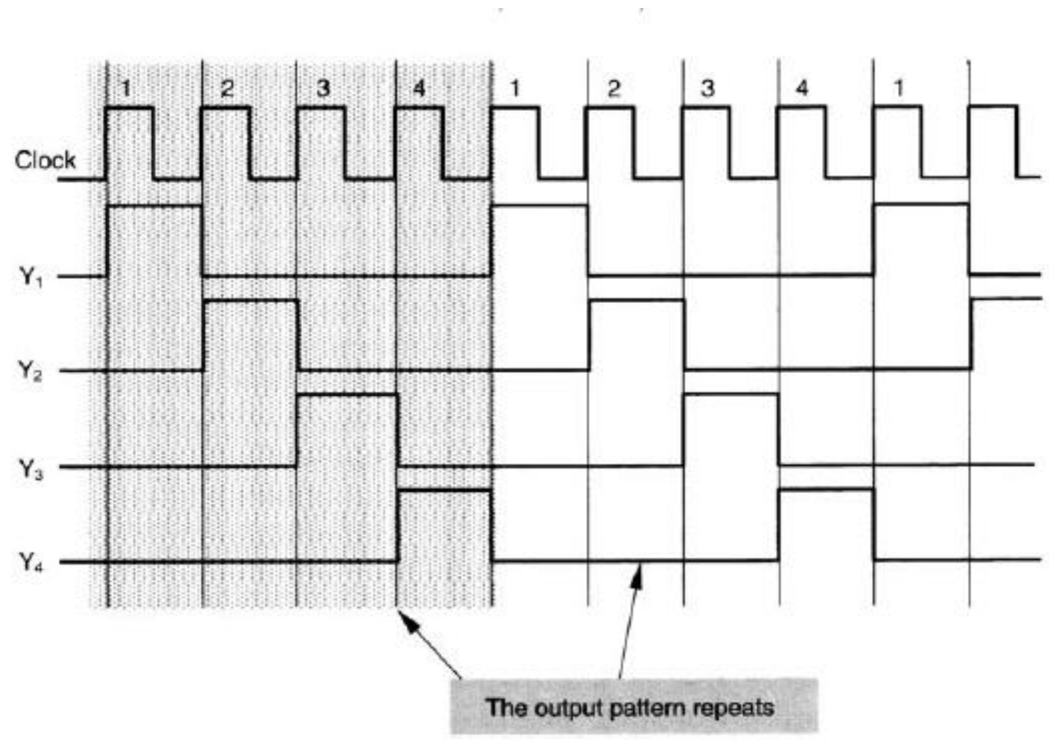


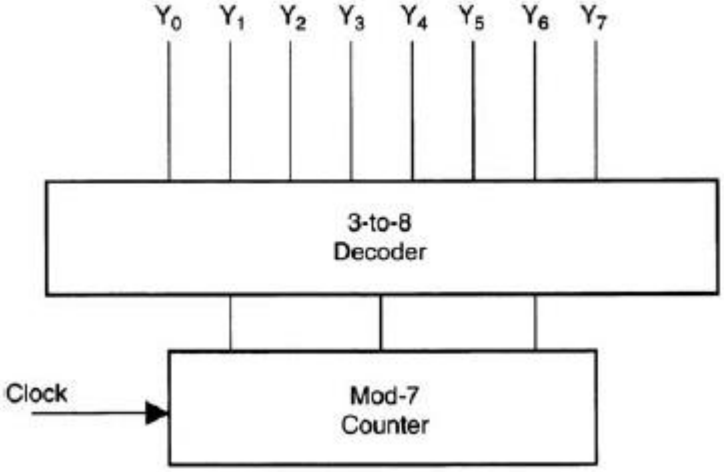
Figure 4-bit ring counter.

- The initial memory element outputs are *Y*1*Y*2*Y*3*Y*4=1000. On each rising edge of the clock, the output follows the sequence 1000, 0100, 0010, 0001. This sequence is repeated with 1000 following 0001.
- The outputs of the circuit satisfy the following two conditions. First, over one clock cycle, exactly one output assumes a value of 1.
- Second, the output is ordered where during the first clock pulse *Y*1 assumes a value of 1.
- During the second clock pulse, *Y*2 assumes a value of 1. Similar logic applies to the remaining outputs.



The timing diagram

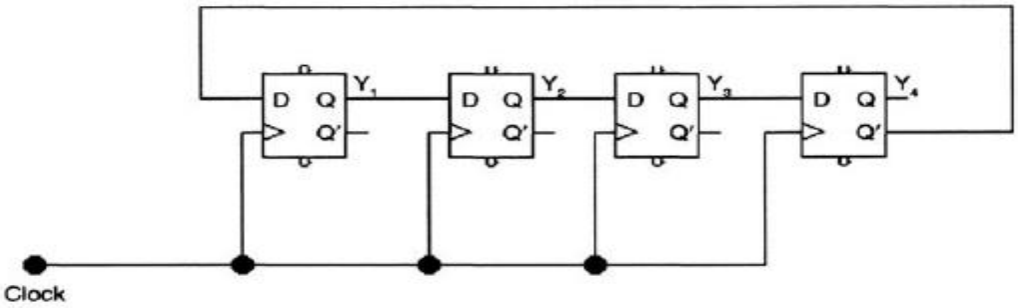
- An alternative method to generate the above sequence can be accomplished by using a 2-bit binary counter and a 2-to-4 decoder.



- In the figure, the counter output is 000 initially. As a result, the output of the 3-to-8 decoder is 10000000. On the application of the clock pulses, the counter output follows the sequence 000, 001, 010, 111, 000. As a result, the decoder output follows the sequence 10000000, 01000000, 00100000, 000100000, 000000000.

Johnson Counters

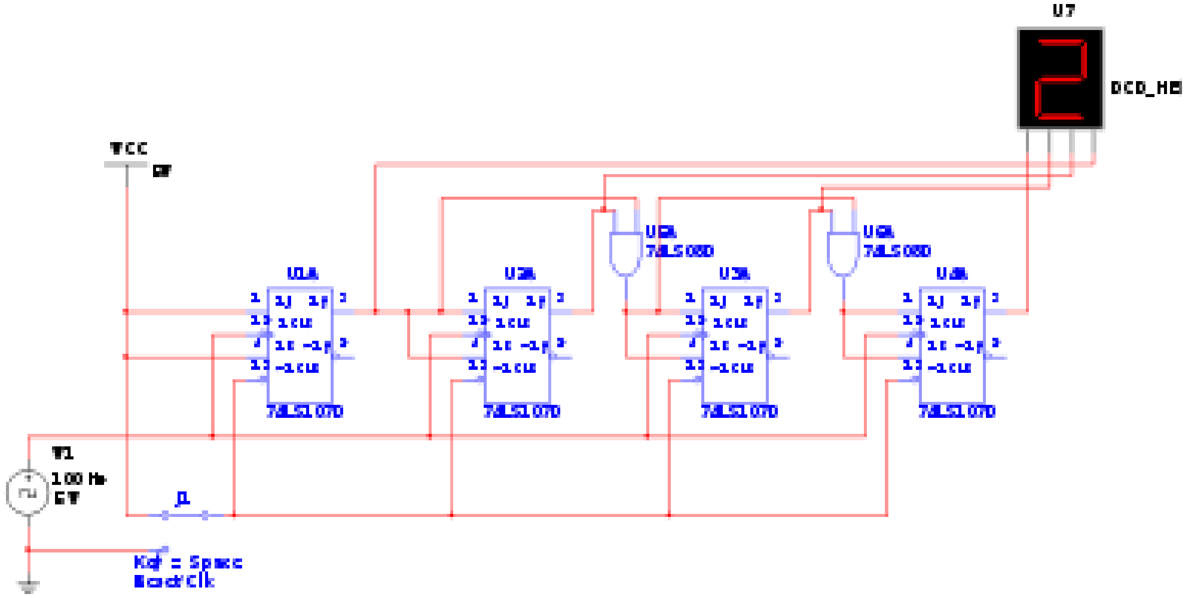
- The circuit is similar to a ring counter with the feedback connection changed. The complemented output of the last flipflop is now connected to the input of the first flip-flop.
- Assuming the initial outputs of all flip-flops are 0, on the application of clock cycles we obtain the repeated sequence 0000, 1000, 1100, 1110, 1111, 0111, 0011, 0001, 0000, etc.
- During the first clock pulse, the complement of the last flip-flop (right-most flip-flop) is shifted into the first flip-flop (left-most). Zeros are shifted in the remaining flip-flops.
- As a result, following the 0000 output, the next output obtained is 1000. On the next clock pulse, the last flipflop output is still 0. As a result, 1 is shifted into the first flip-flop. The result of the shift is to obtain the next output in the sequence, 1100. This process continues until the output becomes 1111. The same logic is applied to obtain the remaining outputs in the sequence.



- As can be seen from the sequence given above, with k-bit Johnson counter we generate *2k* different sequences. In this sequence, one can inspect the binary code for each state and derive a unique expression. The expression should evaluate to 1 only when a particular sequence is currently at the output. The expression generated should evaluate to 0 otherwise.
- This is done so as to distinguish between the patterns in the sequence.
- Generating the minterms corresponding to each element of the sequence is sufficient. By inspection of outputs, however, one can generate a simpler set of product terms.

3. Explain the synchronous,decade and presettable counters?

Synchronous counter



A 4-bit synchronous counter using JK flip-flops

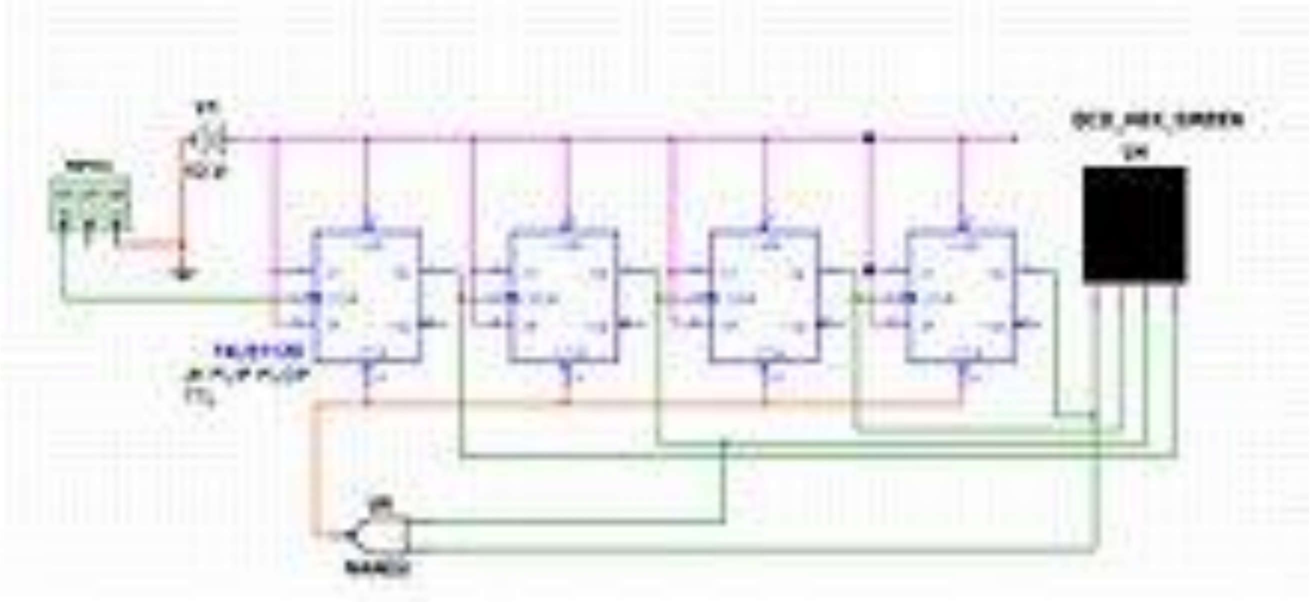
In synchronous counters, the clock inputs of all the flip-flops are connected together and are triggered by the input pulses. Thus, all the flip-flops change state simultaneously (in parallel). The circuit below is a 4-bit synchronous counter. The J and K inputs of FF0 are connected to HIGH. FF1 has its J and K inputs connected to the output of FF0, and the J and K inputs of FF2 are connected to the output of an AND gate that is fed by the outputs of FF0 and FF1.

A simple way of implementing the logic for each bit of an ascending counter (which is what is depicted in the image to the right) is for each bit to toggle when all of the less significant bits are at a logic high state. For example, bit 1 toggles when bit 0 is logic high; bit 2 toggles when both bit 1 and bit 0 are logic high; bit 3 toggles when bit 2, bit 1 and bit 0 are all high; and so on.

Synchronous counters can also be implemented with hardware finite state machines, which are more complex but allow for smoother, more stable transitions.

Hardware-based counters are of this type.A simple way of implementing the logic for each bit of an ascending counter (which is what is depicted in the image to the right) is for each bit to toggle when all of the less significant bits are at a logic high state

Decade counter



A circuit diagram of decade counter using JK Flip-flops (74LS112D)

A decade counter is one that counts in decimal digits, rather than binary. A decade counter may have each digit binary encoded (that is, it may count in binary-coded decimal, as the 7490 integrated circuit did) or other binary encodings. "A decade counter is a binary counter that is designed to count to 1010b (decimal 10). An ordinary four-stage counter can be easily modified to a decade counter by adding a NAND gate as in the schematic to the right. Notice that FF2 and FF4 provide the inputs to the NAND gate.

The NAND gate outputs are connected to the CLR input of each of the FFs." A decade counter is one that counts in decimal digits, rather than binary. It counts from 0 to 9 and then resets to zero. The counter output can be set to zero by pulsing the reset line low. The count then increments on each clock pulse until it reaches 1001 (decimal 9). When it increments to 1010 (decimal 10) both inputs of the NAND gate go high. The result is that the NAND output goes low, and resets the counter to zero. D going low can be a CARRY OUT signal, indicating that there has been a count of ten.

Presettable Counters

Presettable counters are those that can be preset to any starting count either asynchronously (independently of the clock signal) or synchronously (with the active transition of the clock signal). The presetting operation is achieved with the help of PRESET and CLEAR (or MASTER RESET) inputs available on the flip-flops. The presetting operation is also known as the 'preloading' or simply the 'loading' operation.

Presettable counters can be UP counters, DOWN counters or UP/DOWN counters. Additional inputs/outputs available on a presettable UP/DOWN counter usually include PRESET inputs, from where any desired count can be loaded, parallel load (PL) inputs, which when active allow the PRESET inputs to be loaded onto the counter outputs, and terminal count (TC) outputs, which become active when the counter reaches the terminal count.

4.What is shift register and its types and application? (May ' 12)

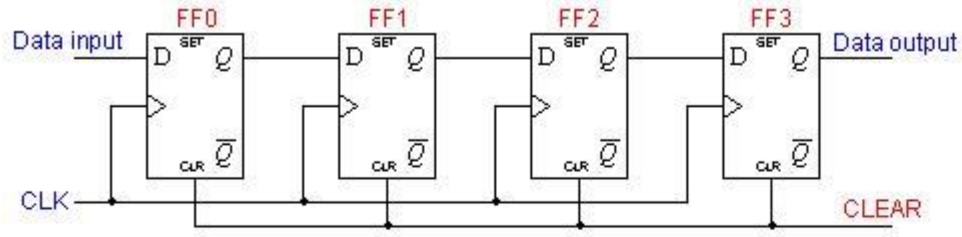
<u>Introduction</u>

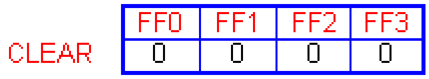
Shift registers are a type of sequential logic circuit, mainly for storage of digital data. They are a group of flip-flops connected in a chain so that the output from one flip-flop becomes the input of the next flip-flop. Most of the registers possess no characteristic internal sequence of states. All the flip-flops are driven by a common clock, and all are set or reset simultaneously.

In this chapter, the basic types of shift registers are studied, such as Serial In - Serial Out, Serial In - Parallel Out, Parallel In - Serial Out, Parallel In - Parallel Out, and bidirectional shift registers. A special form of counter - the shift register counter, is also introduced.

Serial In - Serial Out Shift Registers

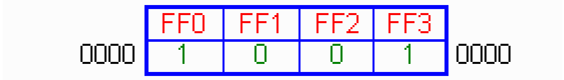
A basic four-bit shift register can be constructed using four D flip-flops, as shown below. The operation of the circuit is as follows. The register is first cleared, forcing all four outputs to zero. The input data is then applied sequentially to the D input of the first flip-flop on the left (FF0). During each clock pulse, one bit is transmitted from left to right. Assume a data word to be 1001. The least significant bit of the data has to be shifted through the register from FF0 to FF3.





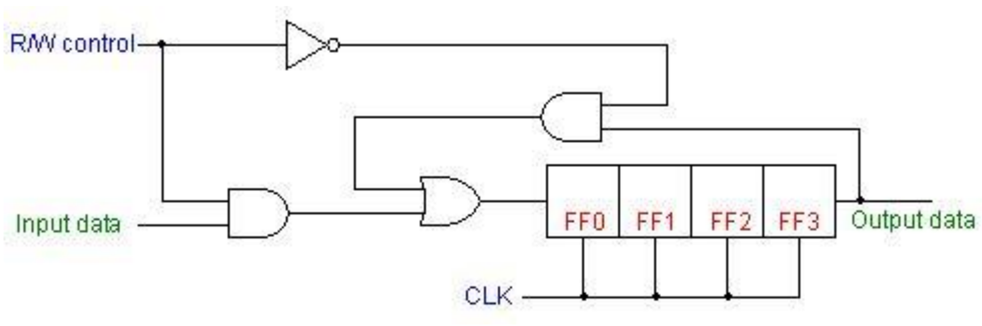
	FF0	FF1	FF2	FF3
CLEAR	0	0	0	0

In order to get the data out of the register, they must be shifted out serially. This can be done destructively or non-destructively. For destructive readout, the original data is lost and at the end of the read cycle, all flip-flops are reset to zero.

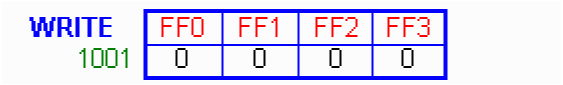


	FF0	FF1	FF2	FF3	
0000	1	0	0	1	0000

To avoid the loss of data, an arrangement for a non-destructive reading can be done by adding two AND gates, an OR gate and an inverter to the system. The construction of this circuit is shown below.



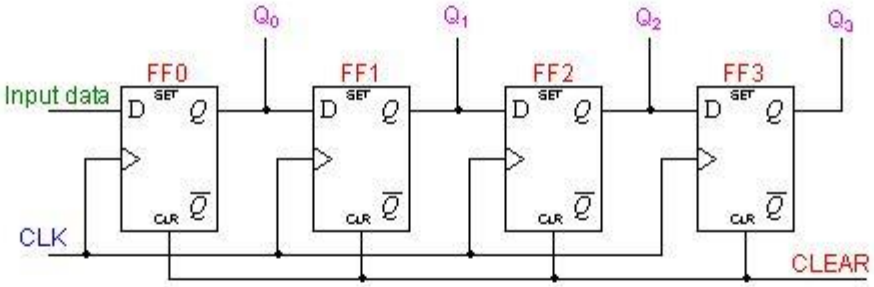
The data is loaded to the register when the control line is HIGH (ie WRITE). The data can be shifted out of the register when the control line is LOW (ie READ). This is shown in the animation below.



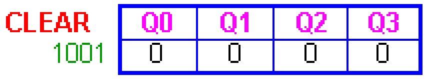
WRITE	FF0	FF1	FF2	FF3
1001	0	0	0	0

Serial In - Parallel Out Shift Registers

For this kind of register, data bits are entered serially in the same manner as discussed in the last section. The difference is the way in which the data bits are taken out of the register. Once the data are stored, each bit appears on its respective output line, and all bits are available simultaneously. A construction of a four-bit serial in - parallel out register is shown below.



In the animation below, we can see how the four-bit binary number 1001 is shifted to the Q outputs of the register.

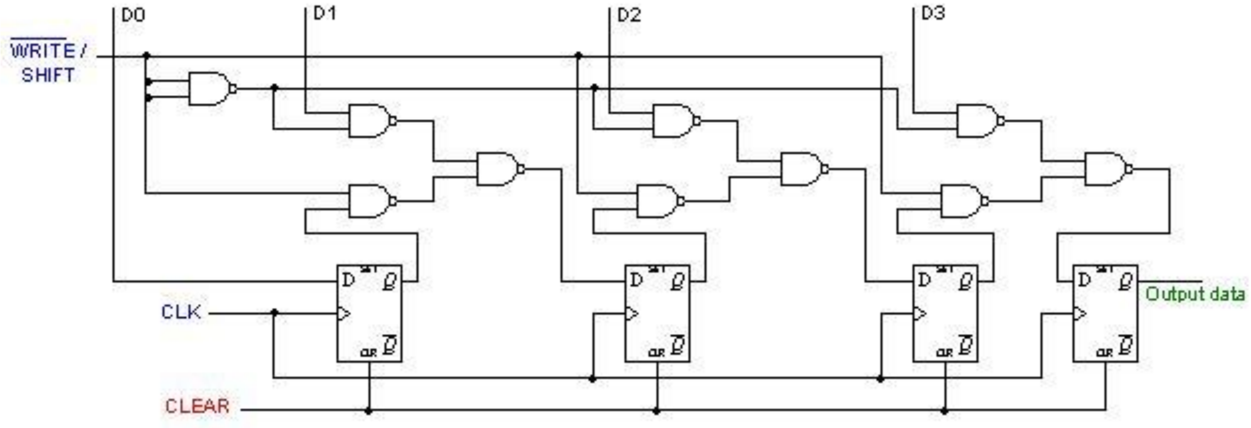


CLEAR
1001

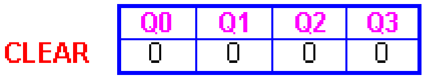
Q0	Q1	Q2	Q3
0	0	0	0

Parallel In - Serial Out Shift Registers

A four-bit parallel in - serial out shift register is shown below. The circuit uses D flip-flops and NAND gates for entering data (ie writing) to the register.



D0, D1, D2 and D3 are the parallel inputs, where D0 is the most significant bit and D3 is the least significant bit. To write data in, the mode control line is taken to LOW and the data is clocked in. The data can be shifted when the mode control line is HIGH as SHIFT is active high. The register performs right shift operation on the application of a clock pulse, as shown in the animation below.

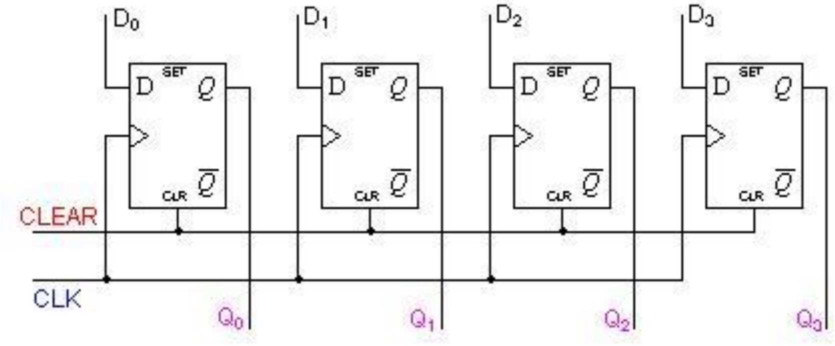


CLEAR

Q0	Q1	Q2	Q3
0	0	0	0

Parallel In - Parallel Out Shift Registers

For parallel in - parallel out shift registers, all data bits appear on the parallel outputs immediately following the simultaneous entry of the data bits. The following circuit is a four-bit parallel in - parallel out shift register constructed by D flip-flops.

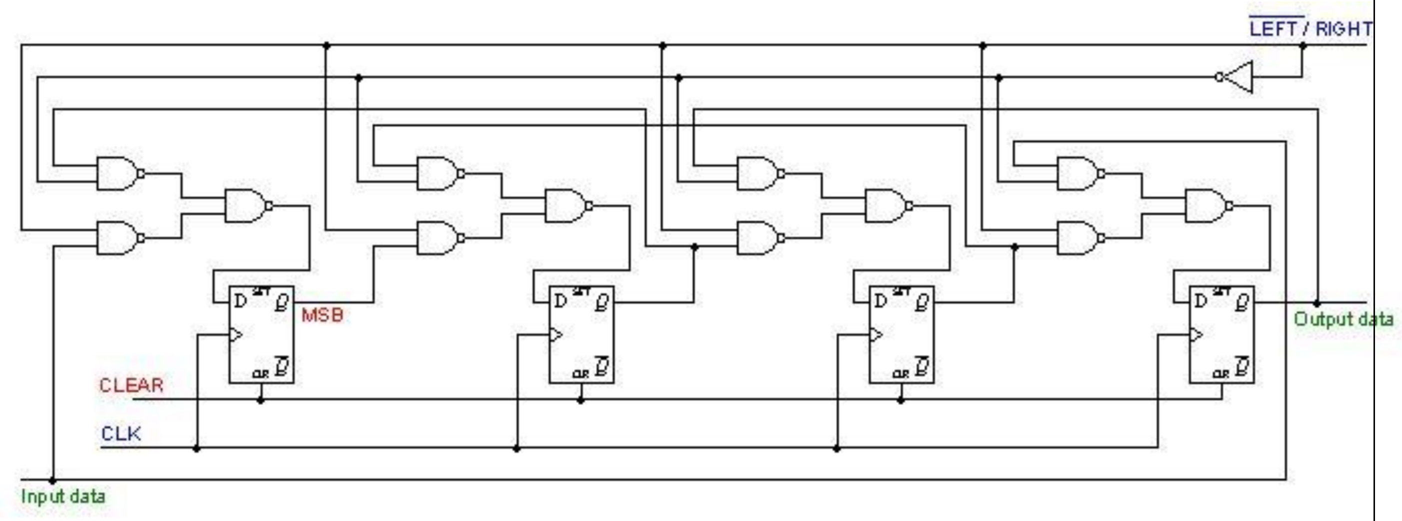


The D's are the parallel inputs and the Q's are the parallel outputs. Once the register is clocked, all the data at the D inputs appear at the corresponding Q outputs simultaneously.

Bidirectional Shift Registers

The registers discussed so far involved only right shift operations. Each right shift operation has the effect of successively dividing the binary number by two. If the operation is reversed (left shift), this has the effect of multiplying the number by two. With suitable gating arrangement a serial shift register can perform both operations.

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. The animation below performs right shift four times, then left shift four times. Notice the order of the four output bits are not the same as the order of the original four input bits. They are actually reversed!

Shift Register Counters

Two of the most common types of shift register counters are introduced here: the Ring counter and the Johnson counter. They are basically shift registers with the serial outputs connected back to the serial inputs in order to produce particular sequences. These registers are classified as counters because they exhibit a specified sequence of states.

Applications

Shift registers can be found in many applications. Here is a list of a few.

To produce time delay

The serial in -serial out shift register can be used as a time delay device. The amount of delay can be controlled by:

1. the number of stages in the register
2. the clock frequency

To simplify combinational logic

The ring counter technique can be effectively utilized to implement synchronous sequential circuits. A major problem in the realization of sequential circuits is the assignment of binary codes to the internal states of the circuit in order to reduce the complexity of circuits required. By assigning one flip-flop to one internal state, it is possible to simplify the combinational logic required to realize the complete sequential circuit. When the circuit is in a particular state, the flip-flop corresponding to that state is set to HIGH and all other flip-flops remain LOW.

To convert serial data to parallel data

A computer or microprocessor-based system commonly requires incoming data to be in parallel format. But frequently, these systems must communicate with external devices that send or receive serial data. So, serial-to-parallel conversion is required. As shown in the previous sections, a serial in - parallel out register can achieve this.

PONDICHERRY UNIVERSITY QUESIONS

2 MARKS

1. Define registers.**(Jan'11) (Ref. Qn. No.: 18, Page No.: 2)**
2. What is meant by clocked sequential circuits?
   (April ' 13) (Ref. Qn. No.: 26, Page No.: 2)
3. Define primitive flow table: **(Jan'11) (Ref. Qn. No.: 44, Page No.: 2)**
4. Define the term counter. **(April '13) (Ref. Qn. No.: 52, Page No.: 2)**
5. State ripple counter**(Jan '11) (Ref. Qn. No.: 54, Page No.: 2)**

11 MARKS

1. What is shift register and its types and application? **(May ' 12) (Ref. Qn. No.: 2, Page No.: 32)**

srivenkateshwaraa
College of Engineering & Technology
(Approved by AICTE, New Delhi & Affiliated to Pondicherry University, Puducherry)
13-A, Pondy - Villupuram Main Road, Ariyur, Puducherry - 605 102.
ASPIRE TO EXCEL
ISO 21001

UNIT V: DESIGNOFSEQUENTIAL CIRCUITS:

Design of Synchronous sequential circuits: Model Selection – State transition diagram – state

synthesistable–Design equations and circuitdiagram–Statereduction technique.Asynchronous sequential circuits–Analysis–Problemswith asynchronoussequential circuits– Designof asynchronous sequential circuits State transition diagram, Primitive table, State reduction, state

assignment anddesign equations.

2 marks

1. Whataresecondary variables?

-presentstate variablesin asynchronous sequentialcircuits

2. Whatareexcitationvariables?

-nextstate variablesin asynchronous sequentialcircuits

3. Whatisfundamentalmode sequentialcircuit?

- -inputvariableschangesifthe circuitisstable
- -inputs are levels, notpulses
- -onlyone inputcan change ata given time

4. Whatarepulsemode circuit?

- -inputs are pulses
- -width ofpulsesarelongforcircuittorespondto the input
- -pulsewidthmustnotbe solongthatitis stillpresentafterthe newstateisreached

5. Whatarethe significanceofstate assignment?

In synchronouscircuits-state assignments are madewith the objective ofcircuit reduction Asynchronouscircuits-its objective is to avoid criticalraces

6. When doesacycle occur?

-asynchronouscircuitmakesa transition through a seriesofunstable state

7. Whatarethe different techniquesused instate assignment?

-shared rowstate assignment

-onehotstateassignment

8. Whatarethe stepsfor the design ofasynchronous sequentialcircuit?

- -constructionofprimitive flowtable
- -reductionofflowtable
- -state assignment is made
- -realization ofprimitive flowtable

9. Whatisflow table?

-state tableofan synchronous sequentialnetwork

10. Whatis primitiveflowchart?

-one stablestateperrow

11. Definemergergraph.

Themergergraphisdefinedasfollows.Itcontainsthe samenumberofverticesasthestatetablecontains states.Alinedrawn betweenthetwostateverticesindicateseachcompatiblestatepair.Ittwostatesare incompatibleno connectingline is drawn.

12. Define state table.

Forthedesignofsequentialcounterswehavetorelatepresentstatesandnextstates.Thetable,which representstherelationshipbetween presentstatesandnextstates, is called statetable.

13. Whatarethestepsforthe designofasynchronous sequentialcircuit?

- Constructionofaprimitiveflowtable fromthe problemstatement.
- Primitive flow table is reduced byeliminatingredundantstatesusingthe state reduction
- State assignment is made
- Theprimitive flowtableisrealized usingappropriatelogic elements.

14. Define primitiveflowtable:

Itisdefinedasaflowtablewhichhasexactlyonestablestateforeachrowinthetable.Thedesign processbegins withthe construction ofprimitive flowtable.

15. Writea shortnote onpulsemodecircuit.

Pulsemodecircuitassumesthattheinputvariablesarepulsesinsteadoflevel.Thewidthofthepulsesis

longenoughforthecircuittorespondtotheinputandthepulsewidthmustnotbesolongthatitisstill presentafterthe newstateis reached.

11 marks

1. Designa finitestatemachinefor thegiven sequencedetectorandimplement with flip flop.

A**Finite StateMachine (FSM)**is acircuitthatcan existin afinite numberofstates, usuallya rather smallnumber. Finite StateMachines with more than 32 states are rare.The FSMhasa memorythat storesits state. If the FSMhasNstates,thenits memorycan beimplemented withP flip–flops where

$$\mathbf{2^{P-1}<N\leq 2^{P}}$$

Typicalvalues: 3 states2 flip–flops

4 states2 flip–flops

5 states3 flip–flops

8 states3 flip–flops

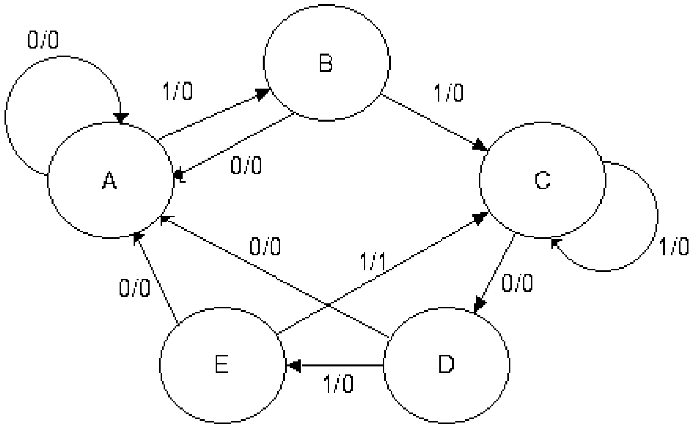
Toolsto describe finitestatesmachines include

1) Thestate diagram

2) Thestate table

3) Thetransitiontable

StateDiagramfora SequenceDetector



NOTE:Wehave five states, labeled “A”, “B”,“C”, “D”, and“E”.

Wehave labelededgesconnectingthestates. Each is labeled Input/Output.

Thisis adirected graph with labeled edgesand loops.

1. The main function of the state diagram for the FSM is to indicate what the nextstate willbe given thepresentstateandinput.
2. Here the input is labeled X. Were the input two bits at a time, theinputwould belabeledasX_1X_0,with$X_1$the moresignificantbit.

3. The labeling of the arcs between the states indicates that there is output associated with each transition. Not all Finite State Machines have outputassociated withthetransition.Thisone does.

4. This and all typical FSM represents a synchronous machine.

Transitions between states and production of output (if any) takes place at a fixed phaseofthe clock, dependingon the flip–flopsusedtoimplement the circuit.

5. Were we pressed to be more specific, we would associate the transitions

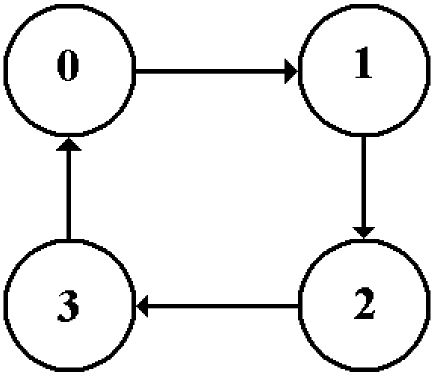
withtherisingedge ofthe clock. Thisisusuallyan unnecessarydetail.

2. Designa finitestatemachinefor theModulo–4 Counterandimplement withDflipflop.

Hereis the state diagramfora modulo–4counter.

Thereis no inputbuttheclock. It justcountsclockpulses. Note

the direction ofthe arrows;thisisan up–counter.



StateTables

Thestate tableisa tabularformof thestate diagram. Itis easierto workwith.

Hereis the statetable forthe sequencedetector.

PresentState	NextState /Output	
	X= 0	X= 1
A	A/0	B/0

B	A/0	C/0
C	D/0	C/0
D	A/0	E / 0
E	A/0	C/1

Hereis the statetable forthe modulo–4 counter.

PresentState	NextState
0	1
1	2
2	3
3	0

Transition Tables

Transitiontablesarejuststatetablesin whichthelabels have beenreplaced bybinarynumbers. Often the labelsareretainedtofacilitatetranslationto binary.

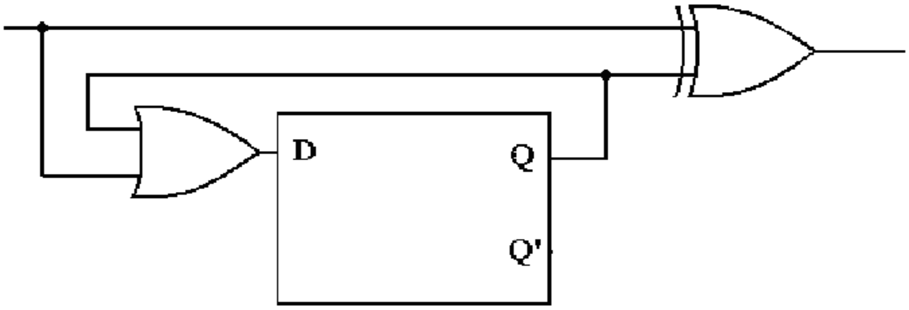
Hereis thetransition table for the sequencedetector.

PresentState	NextState /Output	
	X= 0	X= 1
A= 000	000 /0	001 /0
B= 001	000 /0	010 /0
C= 010	011 /0	010 /0
D= 011	000 /0	100 /0
E = 100	000 /0	010 /1

Hereis thetransition table for the modulo–4 counter.Thereis no outputtable.

PresentState	NextState
0 = 0 0	0 1
1 = 0 1	1 0
2 = 1 0	1 1
3 = 1 1	0 0

Sample Circuitfor Analysis


D
Q
Q'

Theanalysis ofsuch acircuitfollowsa fixedsetofsteps.

1) Determinetheinputsand outputs of the circuit. Assignvariablestorepresentthese.

2) Characterizetheinputs andoutputs ofthe flip-flops. ShowasBooleanexpressions.

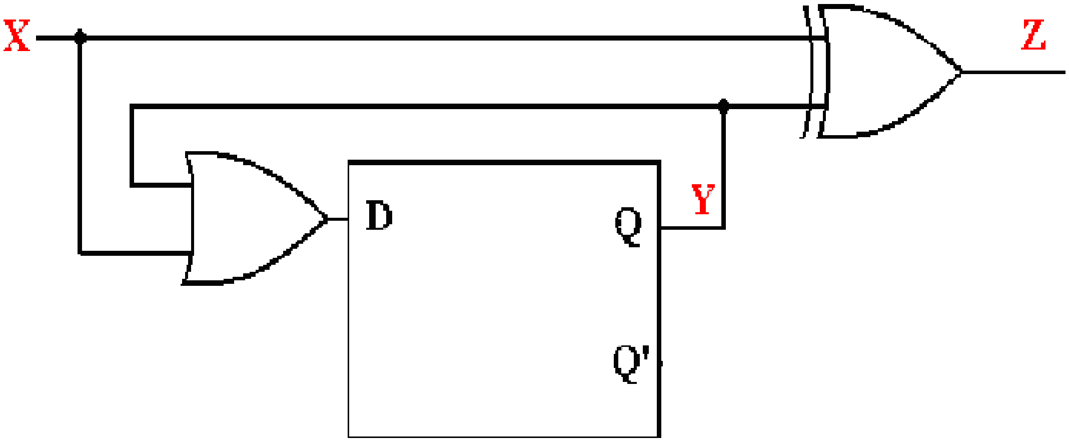
3) Construct the NextStateand OutputTables.

4) Construct the State Diagram.

5) Ifpossible, identifythe circuit.There are no goodrules for thisstep.

Step 1:Determinetheinputsand outputs of thecircuit.

Thecircuithasoneinputand one output, with one internalvariableof interest.



TheinputislabeledasX.

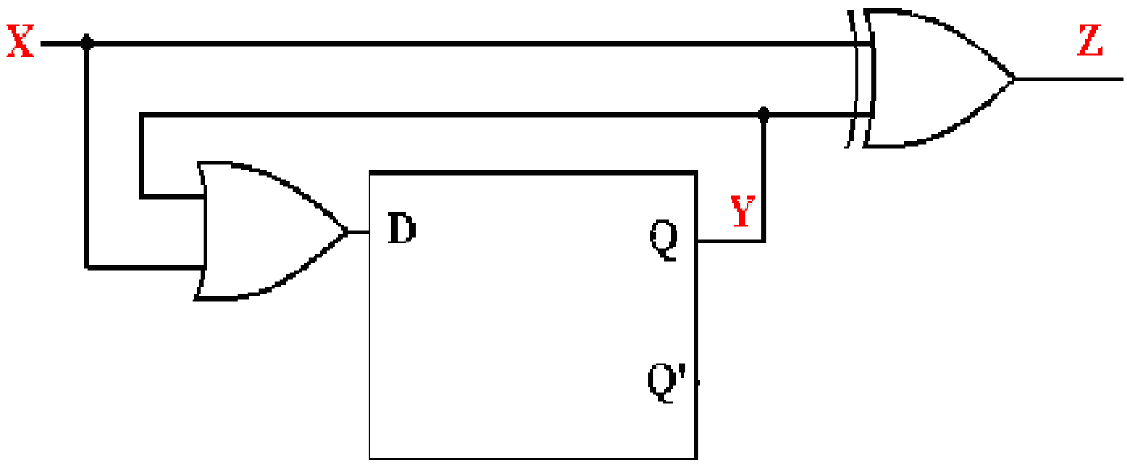
Theoutputislabeled asZ.

Theinternallinethatis fedbackintothe flip–flop is labeledasY.

NOTE:Thereis outputassociatedwiththeinputbecause we see the gate producing

Zbased on the inputX.

Step2:Show theinputs and outputs asBooleanexpressions.



Input:	X
Output:	Z= X⊕Y
Input to Flip–Flop:	D=X +Y

Step 3:Constructthe NextState andOutputTables

Hereis the nextstatetable.

X	Q(**t**)= Y	D= X+Y	Q(**t**+1)
0	0	0	0
0	1	1	1
1	0	1	1
1	1	1	1

Weknowthe presentstateof the flip–flop;callit Y.

Given Y and X, the input,we can compute D.This determines the nextstate.

Hereis the outputtable. Itdepends ontheinputandpresentstate.

X	Y=Q(t)	Z
0	0	0
0	1	1
1	0	1
1	1	0

Step 3A:Constructthe NextState/OutputTable

Justcombine the twotablesintoonetable.

X	Q(**t**)= Y	D= X+Y	Q(**t**+1)/Z
0	0	0	0 /0
0	1	1	1 /1
1	0	1	1 /1
1	1	1	1 /0

Wethen putthetableintoastandard formthatwilllead tothe state diagram.

PresentState	NextState/Output	
	X= 0	X= 1
0	0 /0	1 /1
1	1 /1	1 /0

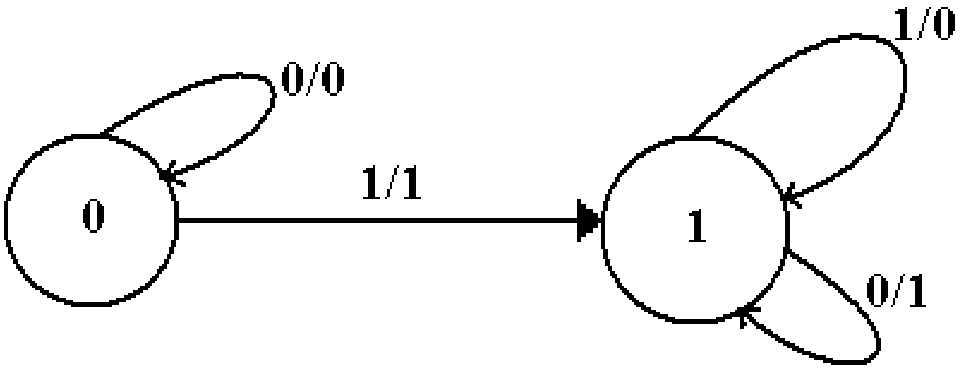
Weuse thisto build a statediagram. Thetwo statesare Q = 0 and Q= 1.
Theoutputs are associatedwiththetransitions.

Step 4:Constructthe State Diagram.

Here againisthestate tablewith output.

PresentState	NextState/Output	
	X= 0	X= 1
0	0 /0	1 /1
1	1 /1	1 /0

Hereis the state diagram.



Thisistherequired answer.

Step 5:Identify the CircuitifPossible

Thisis often hard to do.

The keyhere isthatthecircuitstaysin state 0 untilthefirst1 is input.
Whenthefirst1 is input itgoes to state 1 and stays there forallinput.

Wenowcharacterize theoutputasafunction ofthe input foreach ofthe twostates.

Input	**Q(T)**	**Output**	
0	0	0	For Q(**t**)=0, the output isX
1	0	1	
0	1	1	For Q(**t**)=1, the output is**X**.
1	1	0	

Itcan be shownthatthisisa serialgeneratorforatwo’s–complement.

Thebinaryintegeris read fromLeastSignificantBitto MostSignificantBit. Up to andincludingthefirst(leastsignificant)1,theinputis copied.

After thatitiscomplemented.

0001 1100 becomes1110 0100
0010 1101 becomes1101 0011. Trythis,it works.

3. Design ofa Sequential Circuit-Procedure

Webegin withtherulesfora simple procedure todo the design.

1) Derive the state diagramand statetableforthecircuit.

2) Countthe numberofstatesin the state diagram(callit N)and calculatethe numberofflip-flops needed (call itP) bysolvingthe equation

   $2^{P-1} < N \le 2^{P}$.Thisisbestsolved by guessingthe valueofP.

3) Assign a unique P-bitbinarynumber (state vector)to each state.

Often,the firststate =0, the nextstate = 1,etc.

4) Derive the state transition tableandthe outputtable.

5) Separatethestate transitiontableinto Ptables, one foreachflip-flop.

WARNING:Things can getmessyhere;neatness counts.

6) Decideon thetypesofflip-flops touse.When indoubt, useallJK’s.

7) Derive the inputtablefor each flip-flop usingthe excitationtablesforthetype.

8) Derive the inputequations foreach flip-flop based as functionsoftheinput and currentstate ofallflip-flops.

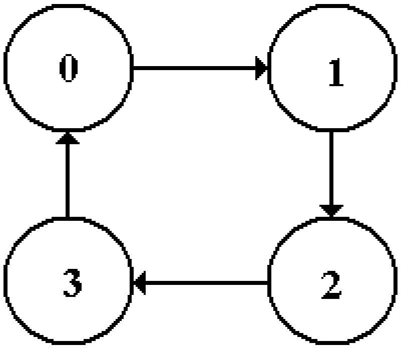
9) Summarize the equations bywritingthemin one place.

10) Draw thecircuitdiagram. Mosthomeworkassignments willnotgo thisfar, as the circuitdiagrams arehardto draw neatly.

4. Design a Modulo–4 Counter

Step 1:Derivethe state diagramand state tableforthe circuit.

Hereis the state diagram. Note that it isquitesimpleand involvesnoinput.



Hereis the statetable forthe modulo–4 counter

PresentState	NextState
0	1
1	2
2	3
3	0

Step 2:Countthe Number ofStates

Obviously, there are only fourstates,numbered 0through 3.

Determine the numberofflip–flops needed.

Solve $2^{P-1} < N \leq 2^P$.IfN= 4, we have P = 2and $2^1 < 4 \leq 2^2$. Weneed twoflip–flopsforthis design. Number them1 and 0.

Theirstateswillbe Q_1andQ_0or$Y_1$and Y_0,dependingon thecontext.

Remember:2^1= 2, 2^2= 4,2^3= 8, 2^4= 16, 2^5= 32, 2^6= 64, 2^7= 128, etc.

Step 3 Assign a uniqueP-bitbinary number(statevector)

to each state.

Here P =2, so weassign aunique2–bitnumber to each state.

For anumberofreasonsthe firststate, state 0, mustbeassigned Y_1= 0 andY_0= 0. For acounter,thereis onlyone assignment thatis notcomplete nonsense.

State	2-bit Vector
0	0 0
1	0 1
2	1 0
3	1 1

The2–bitvectorsarejust the unsigned binaryequivalentof the decimalstate numbers.

Step 4 Derivethestatetransitiontable.

PresentState		NextState
0	00	01
1	01	10
2	10	11
3	11	00

Strictlyspeaking, we should have dropped the decimallabelsinthis step.
However, this representation is often usefulforgivingthe binarynumbers.

Thestate transitiontabletells us whattherequired**nextstate**willbe
foreach **presentstate**.

Step 5 Separate the statetransition table intoP tables,
onefor eachflip-flop.

Here P =2, so weneedtwotables.

Flip-Flop 1			Flip-Flop 0	
PresentState	NextState		PresentState	NextState
Y_1Y_0	$Y_1($ **t**$+1)$		Y_1Y_0	$Y_0($ **t**$+1)$
0 0	0		0 0	1
0 1	1		0 1	0
1 0	1		1 0	1
1 1	0		1 1	0

Each flip–flopisrepresented withthecompletepresentstate and itsown nextstate.

Step 6 Decideon thetypes offlip-flopsto use.

When in doubt, useallJK's.

Ourdesign will useJKflip–flops.

For design work, itisimportantthatwe remember the**excitationtable**.
Hereitis.

Q(t)	**Q(t+1)**	**J**	**K**
0	0	0	d
0	1	1	d
1	0	d	1
1	1	d	0

Step 7 Derivetheinputtableforeachflip-flop usingthe excitation tablesfor thetype.

Hereis thetable forflip–flop 1.

PS	NS	Input	
Y_1Y_0	Y_1	J_1	K_1
0 0	0	0	d
0 1	1	1	d
1 0	1	d	0
1 1	0	d	1

Hereis thetable forflip–flop 0.

PS	NS	Input	
Y_1Y_0	Y_0	J_0	K_0
0 **0**	1	1	d
0 **1**	0	d	1
1 **0**	1	1	d
1 **1**	0	d	1

Step 8 Derivetheinputequationsforeach flip-flop

Iusea setof intuitive rulesbasedon observation and notonformalmethods.

1) Ifa column doesnothave a0 init, matchittotheconstantvalue 1.

Ifa column doesnothave a1 init, matchittotheconstantvalue 0.

2) If the column hasboth 0’s and 1’sinit,tryto matchitto a single variable, which mustbepartof the presentstate. Onlythe 0’s and 1’sina column mustmatch the suggested function.

3) Ifevery0 and 1 inthe column is amismatch, match tothe complement ofafunction ora variable in the presentstate.

4) Ifall theabove fails,tryforsimple combinations ofthepresentstate.
NOTE:Theuseofthecomplementofa statein step3 isduetothefact that

eachflip–flopoutputsbothitsstateandthecomplementofits state.

Step 8 Derivetheinputequationsforeachflip-flop

Hereis theinput tableforFlip–Flop1

PS	NS	Input	
Y_1Y_0	Y_1	J_1	K_1
0 0	0	0	d
0 1	1	1	d
1 0	1	d	0
1 1	0	d	1

J_1=Y_0 **K_1=Y_0**

Hereis theinput tableforFlip–Flop0

PS	NS	Input	
Y_1Y_0	Y_0	J_0	K_0
0 **0**	1	1	d
0 **1**	0	d	1
1 **0**	1	1	d
1 **1**	0	d	1

Heretheyare.

1 **K_0=1**

J

Step 9 Summarize the equationsbywritingthem in one place.

0

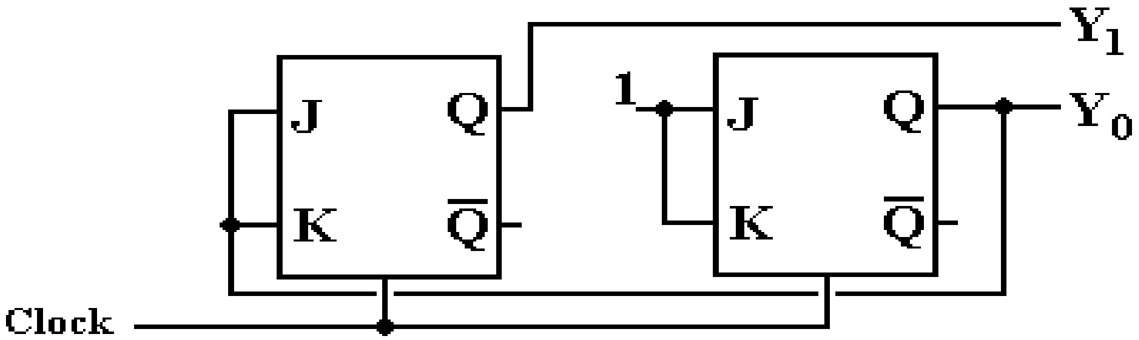
=

J_1= Y_0 K_1= Y_0

J_0= 1 K_0= 1

For homeworkand tests, this is requiredsothatIcan easilyfindtheanswers.

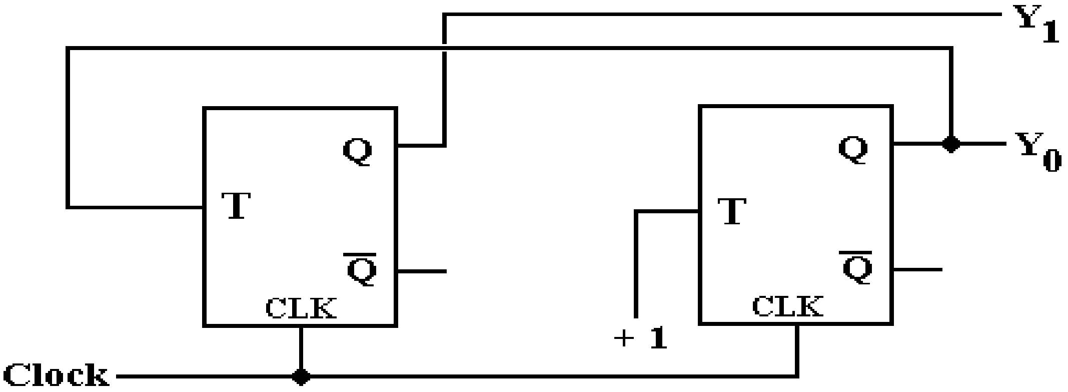
Step 10 Draw thecircuitdiagram.



Butnote thateach flip–flophasinputJ=K.Thissuggestsa simplification.

Step 10A   Draw thecircuitdiagramagain.

Hereis asimplerversion.


Y_1
Q
T
Q̄
CLK
Q
T
Q̄
CLK
Y_0
+ 1
Clock