Date _____

Expt. No. 01 Page No. _____

Generation of Signals

Aim:- Write a MATLAB program to generate the continuous time signal and discreate time signal of unit step, unit impulse, Ramp, periodic Sinusoidal sinc function, plot all the signals.

Software Required:- Matlab Software.

Theory:-

If the amplitude of the signal is defined at every instant of time then it is called continuous time signal. If the amplitude of the signal is defined at only at some instants of time then it is called discreate time signal.

If the signal repeats itself at regular intervals then it is called periodic signal otherwise they are called aperiodic signals.

Examples: Aperiodic signals - ramp, Impulse, unit, step, sinc
periodic signals - square, sawtooth, triangular

unit step signal

$$u(t) = \begin{cases} 1, & t \geq 0 \\ 0, & t < 0 \end{cases} \qquad u[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

unit impulse signal

$$\delta(t) = \begin{cases} 1, & t = 0 \\ 0, & t \neq 0 \end{cases} \qquad \delta[n] = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases}$$

Teacher's Signature _____

Date _____

Expt. No. _____ Page No. _____

Ramp signal

$$r(t) = \begin{cases} t, & t \geq 0 \\ 0, & t < 0 \end{cases} \qquad r[n] = \begin{cases} n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

→ Equivalently $r(t) = t\,u(t)$ $\qquad r[n] = n\,u[n]$

Periodic sinusoidals

sin and cos → $A\cos(\omega t + \phi)$ & $A\sin(\omega t + \phi)$

Sinc function

$$\text{sinc}(x) = \frac{\sin x}{x} \quad \text{if } x \neq 0$$

Procedure:

- open MATLAB
- open new m-file
- Type the program
- Save in current directory
- compile and Run the program
- For the output see Command window / figure window

Program:

```
% Generation of signals
clc;
clear all;
close all;
```

Teacher's Signature _____

Date _____

Expt. No. _____ Page No. _____

```
% generation of unit step signal
t = linspace(-20,20,1000);
y = t>=0;
figure(1);
subplot(2,1,1);
plot(t,y)
grid
ylabel('u(t)');
xlabel('Time');
title('continuous time unit step');
axis([-20 20 -0.5 1.5]);
n = -10:1:10;
y = n>=0;
subplot(2,1,2);
stem(n,y)
grid
ylabel('u(n)');
xlabel('Time');
title('Discrete time unit step');
axis([-10 10 -0.5 1.5])

% generation of unit impulse

t = (-5:0.01:5);
impulse = t==0;
figure(2);
subplot(2,1,1);
plot(t, impulse);
ylabel('delta(t)');
```

Teacher's Signature _____

Date _____

Expt. No. _____ Page No. _____

```
xlabel('Time');
title('continuous time unit impulse signal');
n=-5:1:5;
y=(n==0);
subplot(2,1,2);
stem(n,y);
ylabel('\delta(n)');
xlabel('Time');
title('Discrete time unit impulse signal');

% generation of Ramp signal

t=linspace(-20,20,1000);
y=t.*(t>=0);
figure(3);
subplot(2,1,1);
plot(t,y);
grid on;
ylabel('r(t)');
xlabel('Time');
title('continuous time Ramp signal');
n=-10:1:10;
y=n.*(n>=0);
subplot(2,1,2);
stem(n,y);
grid on;
ylabel('r(n)');
title('Discrete time Ramp Signal');
```

Teacher's Signature _____

Expt. No. _____ Page No. _____

```
% generation of periodic sinusoidal
x=0:pi/100:2*pi;
y1=sin(x);
y2=cos(x);
figure(4);
plot(x,y1,'--',x,y2,'-');
grid on;
xlabel('\leq x \leq 2\pi');
ylabel('Amplitude');
legend('sin(x)','cos(x)');
title('continuous time periodic sinusidal signals');
xticks(0:pi/2:2*pi);
xticklabels({'0','1pi/2','1pi','3\pi/2','2\pi'})
axis([0 2*pi -2 2]);
n=0:pi/10:2*pi;
y1=sin(n);
y2=cos(n);
figure(5);
subplot(2,1,1);
stem(n,y1,'g');
xlabel('\leq n \leq 2\pi');
ylabel('sin(n)');
title('Discrete time sine signal
xticks(0:pi/2:2*pi);
xticklabels({'0','1pi/2','1pi','3\pi/2','2\pi'})
axis([0 2*pi -2 2]);
subplot(2,1,2);
stem(n,y2,'m');
```

Teacher's Signature _____

Date _____

Expt. No. _____ Page No. _____

```
ylabel ('cos(n)');
title ('Discreate Time cosine signal');
xticks (0:pi/2:2*pi);
xticklabels ({'0', '1pi/2', '1pi', '3\pi/2', '2\pi'})
axis ([0 2*pi -2 2]);

% sinc function generation
x = -2*pi:pi/1000:2*pi;
y = sinc(x);
figure(6)
subplot(2,1,1);
plot(x,y);
grid;
xticks(-2*pi:pi/2:2*pi);
xticklabels({'-2\pi', '-3\pi/2', '-1pi', '-1pi/2', '0', '1pi/2', '3\pi/2', '2\pi'})
ylabel('sinc(x)');
xlabel('0\leq x \leq 2\pi');
title('continuous sinc signal');
axis([-2*pi 2*pi -0.5 1]);
n = (-2*pi:pi/10:2*pi);
y = sinc(n);
subplot(2,1,2);
stem(n,y,'r');
grid;
xticks(-2*pi:pi/2:2*pi);
ylabel('sinc(x)');
title('Discreate sinc signal');
axis([-2*pi 2*pi -0.5 1]);
```

Teacher's Signature _____

Date _____

Expt. No. _____ Page No. _____

Result: generated the continuous time signals and discrete time signals - namely unit step, unit impulse, Ramp, periodic sinusoidal & sinc function using MATLAB.

Teacher's Signature _____

Date

Expt. No. 02 Page No.

Fourier series and Fourier transform

Aim: Find the fourier transform of a square pulse using MATLAB. Plot its amplitude and phase spectrum.

Software Required: MatLab Software

Theory:

Aperiodic signal can be expressed as a continuous sum (integral) of everlasting exponentials.

$$x(t) \Longleftrightarrow X(\omega)$$

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt$$

We can plot the spectrum $X(\omega)$ as a function of ω since $X(\omega)$ is complex. We have both amplitude & angle or phase spectra.

$$X(\omega) = |X(\omega)| e^{j \angle X(\omega)}$$

→ where $|X(\omega)|$ is the amplitude of $X(\omega)$.

$\angle X(\omega)$ is the angle or phase of $X(\omega)$

Procedure:

- Open MATLAB
- Open new M-file
- Type the program
- Save in current directory.

Teacher's Signature

Date _____

Expt. No. _____ Page No. _____

- compile and Run the program
- for output see figure window

Program:

```
clc;
x = 0:0.01:10;
s = square(2*pi*x);
Y = abs(fft(x));
subplot(3,1,1);
plot(x,x)
axis([-2 12 -2 2]);
title('Square Pulse');
grid on;
subplot(3,1,2);
plot(fftshift(Y));
title('Amplitude Spectrum');
grid on;
subplot(3,1,3);
z = angle(fftshift(x));
stem(z);
title('Phase spectrum');
grid on;
```

Result:

Fourier transform of the square pulse was generated and observed.

Teacher's Signature _____

Date _____

Expt. No. _____ Page No. _____

(b) **Aim:** Write a MATLAB program to find the trigonometric f. series coefficients of a rectangular periodic signal. Reconstruct the signal by combining the f. series coefficients with appropriate weightings.

Software Required: MATLab Software.

Theory:

→ Fourier developed a mathematical model to transform signals b/w time domain to frequency domain which is called fourier transform.

→ To represent any periodic signal $x(t)$, Fourier developed an expression called f. series.

→ this is in terms of an infinite sum of sines and cosines or exponentials & F.S uses orthogonality conditions.

$$x(t) = \sum_{n=-\infty}^{\infty} a_n e^{jn\omega_0 t}$$

Procedure:

- Switch on the computer & click on the MATLAB icon
- write code
- Run the code
- for o/p see figure window
- observe the plotted graph

Teacher's Signature _____

Expt. No. _____ Page No. _____

Program:

```
clear all;
close all;
function f = exab(n,t,T,K)
wo = (2*pi/T);
for i=1:n
    a(i) = 2* sinc(2*pi* i * t/T )/(pi*i);
end
disp('Trignometric coffs:');
ao = (2*t/T );
for j=1:length(K)
    f(j)=ao;
    for i=1:n
        f(j) = f(j)+a(i) * cos(i*wo *K(j));
    end
end
plot(K,f);
```

Result:-

The trigonometric fourier series coefficients of a rectangular periodic signal was calculated and observed.

Teacher's Signature _____

Date _____

Expt. No. _____ Page No. _____

© **Aim:** Write a MATLAB program to find the trigonometric and exponential fourier series coefficients of a periodic signal of rectangular. plot the discrete spectrum of the signal

Software Required: MATLAB Software

Theory:

To represent any periodic signal $x(t)$, Fourier developed an expression called Fourier series. This is in terms of an infinite sum of sines and cosines or exponentials.

$$x(t) = \sum_{n=-\infty}^{\infty} a_n e^{jn\omega_0 t}$$

where $$a_n = \frac{1}{T_0} \int_{T_0}^{T} x(t) e^{-jn\omega_0 t} dt$$

Trigonometric F.S

$$f(t) = a_0 + \sum_{n=1}^{\infty} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t)$$

where $$a_n = \frac{2}{T} \int_{t_0}^{t_0+T} f(t) \cos n\omega_0 t \, dt$$

$$a_0 = \frac{1}{T} \int_{t_0}^{t_0+T} f(t) dt$$

$$b_n = \frac{2}{T} \int_{t_0}^{t_0+T} f(t) \cos n\omega_0 t \, dt$$

Teacher's Signature _____

Expt. No. _____ Page No. _____

Exponential F.S

$$f(t) = \sum_{n=-\infty}^{\infty} F_n e^{jn\omega_0 t}$$

where $F_n = \frac{1}{T}\int_{t_0}^{t_0+T} f(t) e^{-jn\omega_0 t}\, dt$

Procedure:

- Open MATLAB
- Open new m-file
- Type the program
- Compile and Run program
- Observe o/p graphs

Program:

```
clear all;
close all;
function f=ex2c(n,t,T,k,~)
w0=(2*pi/T);
for i=1:n
    f(i)=(2*t*sinc(2*i*t/T)/T);
end
f0=(2*t/T);
j=sqrt(-1)
for l=1:length(k)
    y(l)=f0;
    for m=1:n
```

Teacher's Signature _____

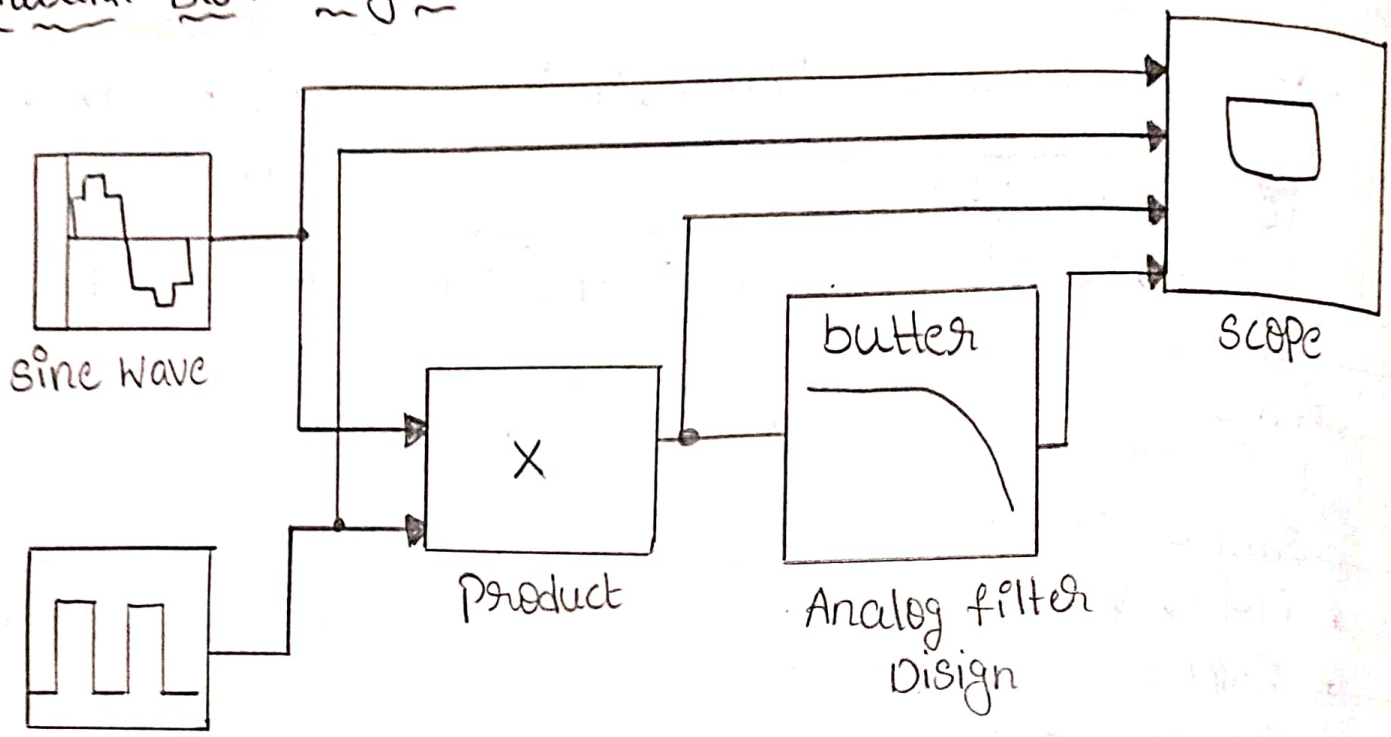
```
        Y(l) = Y(l) + f(m) * exp(-j*m*w0*k(l));
    end
    for m = n+1 : 2*n
        Y(l) = Y(l) + f1(m-n) * exp(j*(m-n)*w0*k(l));
    end
end
F = fft(Y);
subplot(211);
plot(k,Y);
title('Reconstructed signal')
subplot(212);
stem(abs(F));
title('Discrete signal')
```

Result:

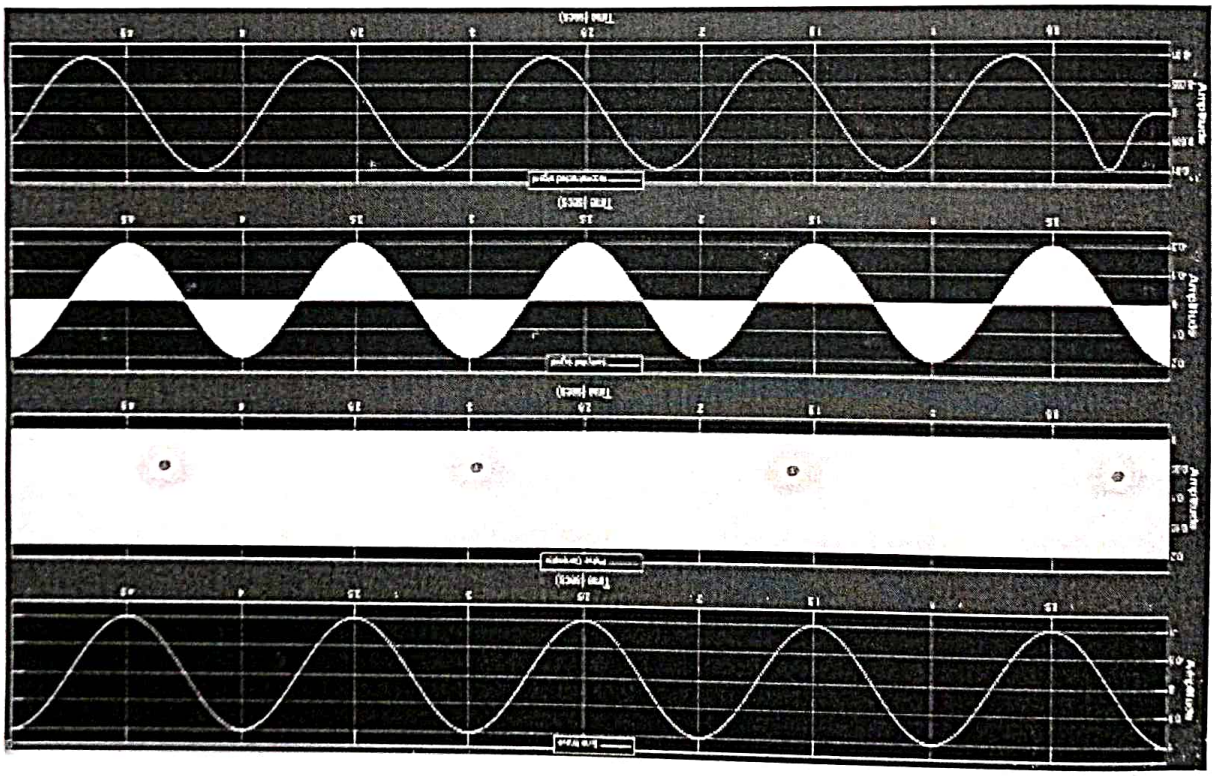
The trigonometric and exponential Fourier series coefficients of a periodic signal of rectangular was calculated and observed.

Teacher's Signature _____

Simulink Blok diagram



Case-1 when $f_s > 2f_a$



Expt. No. 03 Date

Page No.

Sampling and reconstruction of signal

Aim: To verify sampling theorem for different sampling frequencies using Simulink.

Apparatus Required: (a) Hardware Tools: Computer system
(b) Software Tool: MATLAB7.0 upgraded version

Theory:

Sampling theory is the bridge b/w the continuous time and discrete-time worlds.

As a first step to convert analog signals into digital form, the samples of the analog signals are taken at regular intervals. The levels of these signals are then encoded and send to the receiver. At the reciever these samples are recovered and form that the original signal is reconstructed.

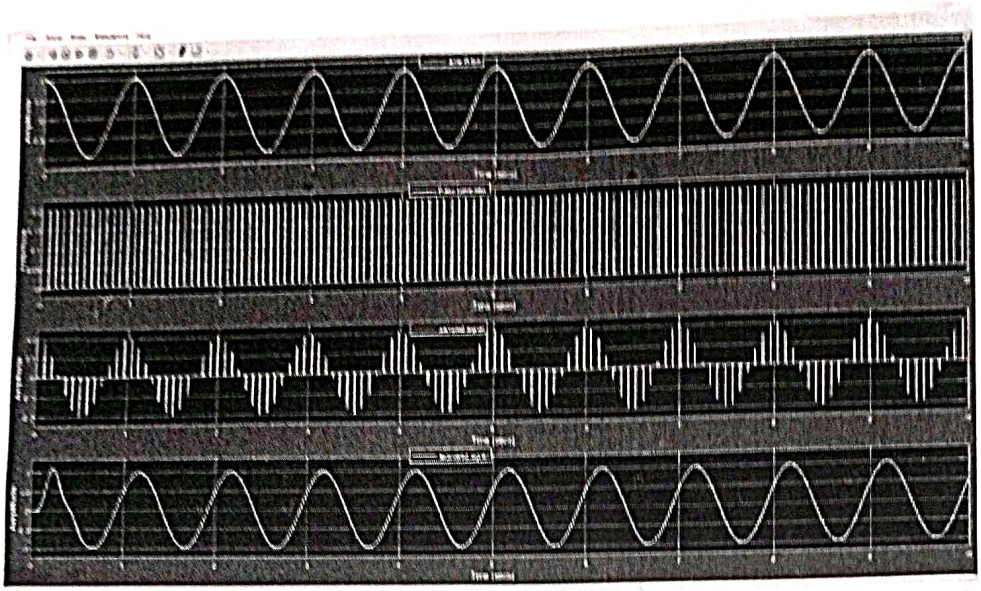
Sampling theorem states that original signal can be faithfully reconstructed only if the sampling frequency is at least double that of the highest frequency component in sampled signal.

Procedure:

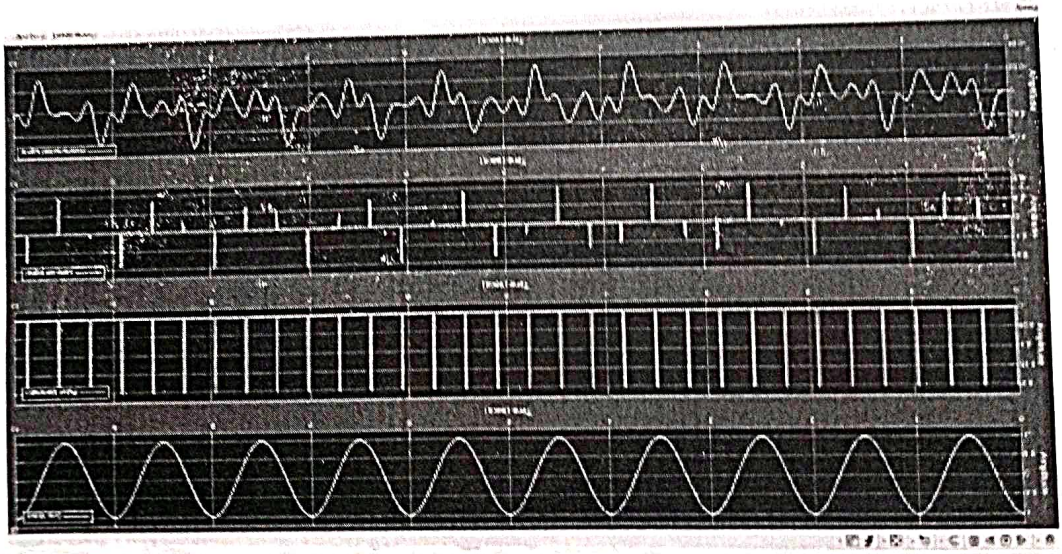
1. Switch on the computer and click on the MATLAB icon
2. Go to Start at the bottom of the command window then select "Simulink" then go to library browser.

Teacher's Signature

case-2 when $f_s = 2f_a$



case-3 when $f_s < 2f_a$



Date

Expt. No. Page No.

drag it into creating file & once you open the MATLAB then click on the Simulink icon

3. Go to file and select new and then select model you will get a new window.

4. Arrange the functional blocks as shown in Simulink model

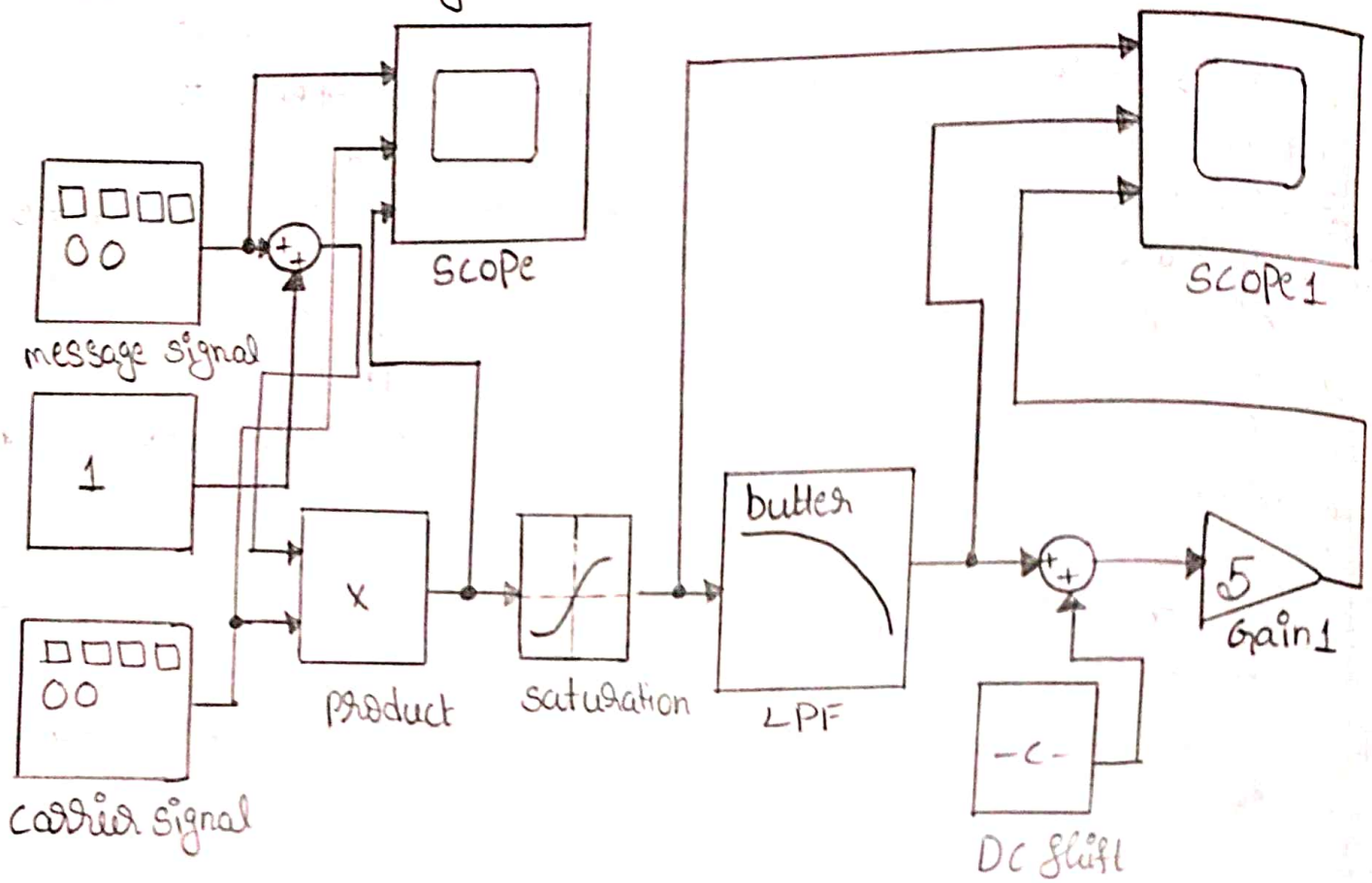
5. Assign required parameters to each functional block

6. observe the outputs on scope.

Result: Sampling theorem is verified.

Teacher's Signature

Simulink Block diagram.



Observation table

m	Vmax	Vmin	$\frac{V_{max}-V_{min}}{V_{max}+V_{min}}$
0.5	2.7	0.9	$\frac{2.7-0.9}{2.7+0.9} = 0.5$
1	4	0	$4/4 = 1$
1.5	5	-1	$5-(-1)/5+(-1) = 1.5$

Date

Expt. No. 04 Page No.

Amplitude modulation & Demodulation

Aim:- To study the function of Amplitude modulation and demodulation (under modulation) using simulink.

Apparatus Required:-

(a) Hardware tools: Computer system

(b) Software tool: MATLAB 7.0 and above version

Theory:-

Amplitude modulation is defined as a process in which the amplitude of carrier wave c(t) is varied linearly with the instantaneous value of m(t). let m(t) is the source message signal & F.T is M(f) let c(t) is carrier signal

→ TO move the freq response of m(t) to a new frequency band centered at f_c Hz, we use frequency shifting property of F.T

→ multiply m(t) by a sinusoid of frequency f_c such that

$$s(t) = m(t) \cos 2\pi f_c t$$

$$S(f) = \frac{1}{2}[M(f-f_c) + M(f+f_c)]$$

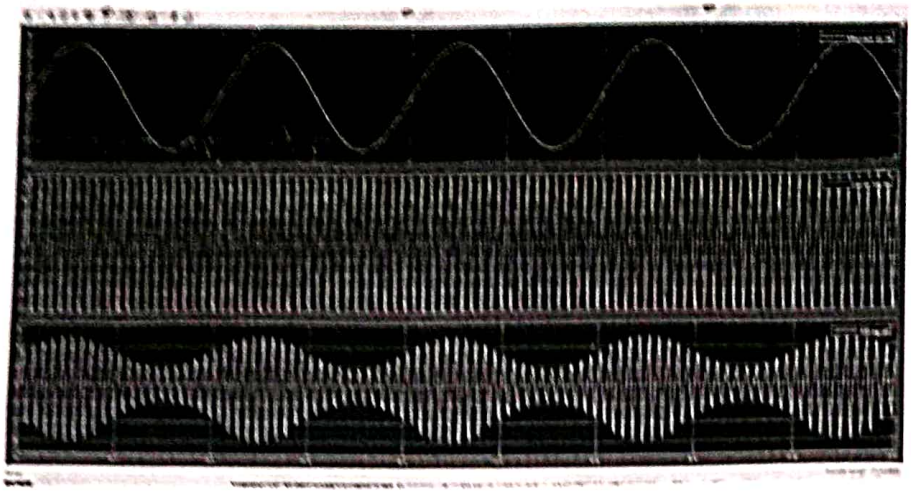
Mathematical Representation of AM [Time domain Representation]

Let the message signal be sinusoidal and be represented as $e_m = E_m \cos \omega_m t$

Similarly $e_c = E_c \cos \omega_c t$

Teacher's Signature

50% Matulation



50% Demodulation

Date

Expt. No. Page No.

where e_m → instantaneous amplitude of modulating signal
E_m → peak amplitude
ω_m → $2\pi f_m$ and f_m → freq of modulating signal
E_c → peak amplitude of carrier signal
f_c → carrier freq.

→ AM wave is expressed by the following eqn

$$e_{AM} = A \cos \omega_c t$$

where $A \rightarrow E_c + E_m \cos \omega_m t$

$$e_{AM} = \{E_c + E_m \cos \omega_m t\} \cos \omega_c t$$

$$e_{AM} = E_c \left\{1 + \frac{E_m}{E_c} \cos \omega_m t\right\} \cos \omega_c t$$

Let $\frac{E_m}{E_c} = m$ be the modulation index.

$$e_{AM} = E_c \{1 + m \cos \omega_m t\} \cos \omega_c t$$

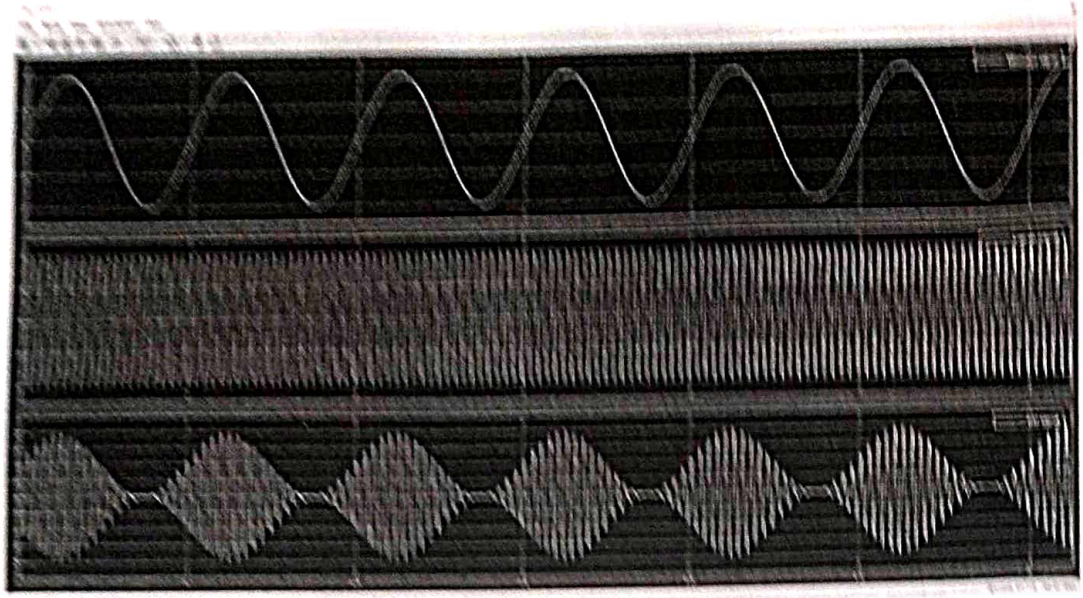
when $E_m \leq E_c$ → no distortion
$E_m > E_c$ → distortion (over modulation)

$$e_{AM} = \underbrace{E_c \cos \omega_c t}_{\text{carrier}} + \underbrace{\frac{mE_c}{2} \cos(\omega_c + \omega_m)t}_{\text{USB}} + \underbrace{\frac{mE_c}{2} \cos(\omega_c - \omega_m)t}_{\text{LSB}}$$

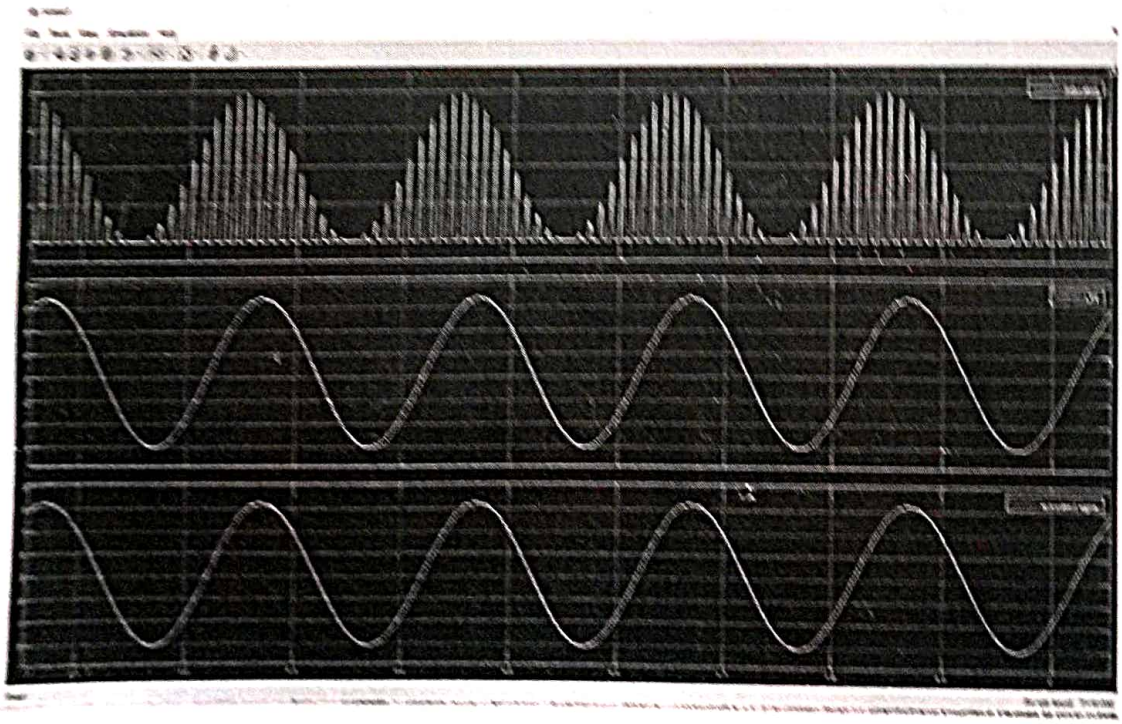
$B.W = 2f_m$

Teacher's Signature

100% modulation



100% Demodulation



Date

Expt. No. Page No.

Carrier Power (P_c) $P_c = \frac{E_c^2}{R}$

$P_{USB} = P_{LSB} = \frac{m^2}{4} P_c$

Total Power $P_t = P_c \left[1 + \frac{m^2}{2}\right]$

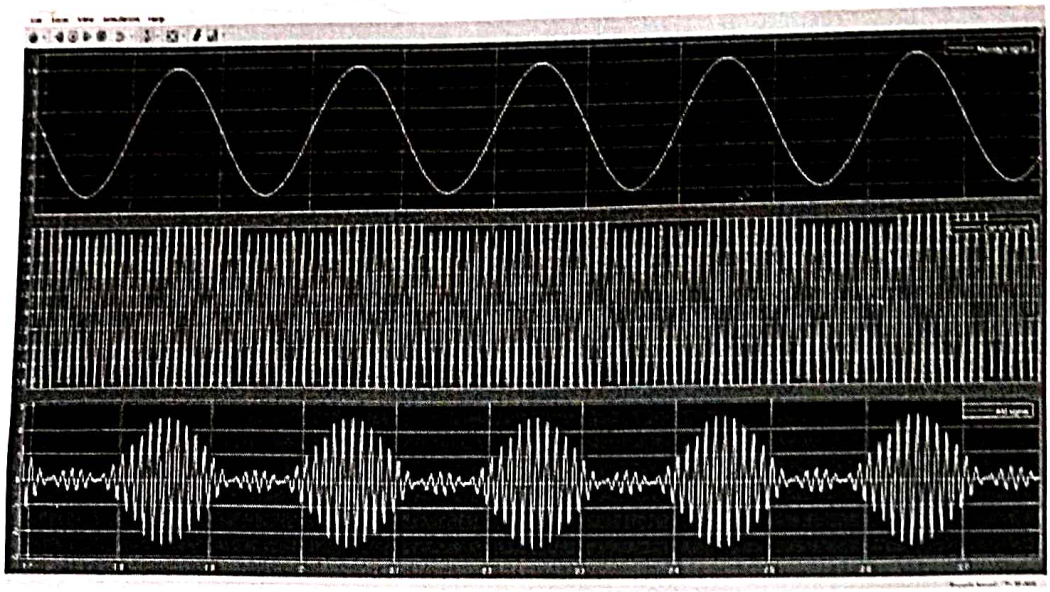
Procedure:-

1. Switch on the computer and click on the MATLAB icon.
2. Go to start at the bottom of the command window then select 'simulink' then go to library browser and drag it in to creating file
3. Arrange the functional blocks as shown in simulink model.
4. Assign required parameters to each functional block.
5. Observe the output on scope

Result:- observed the waveforms of Amplitude modulation for 3 conditions - 50% modulation (under modulation)
100% modulation (exact modulation)
150% modulation (over modulation), and observed the reconstructed message signal

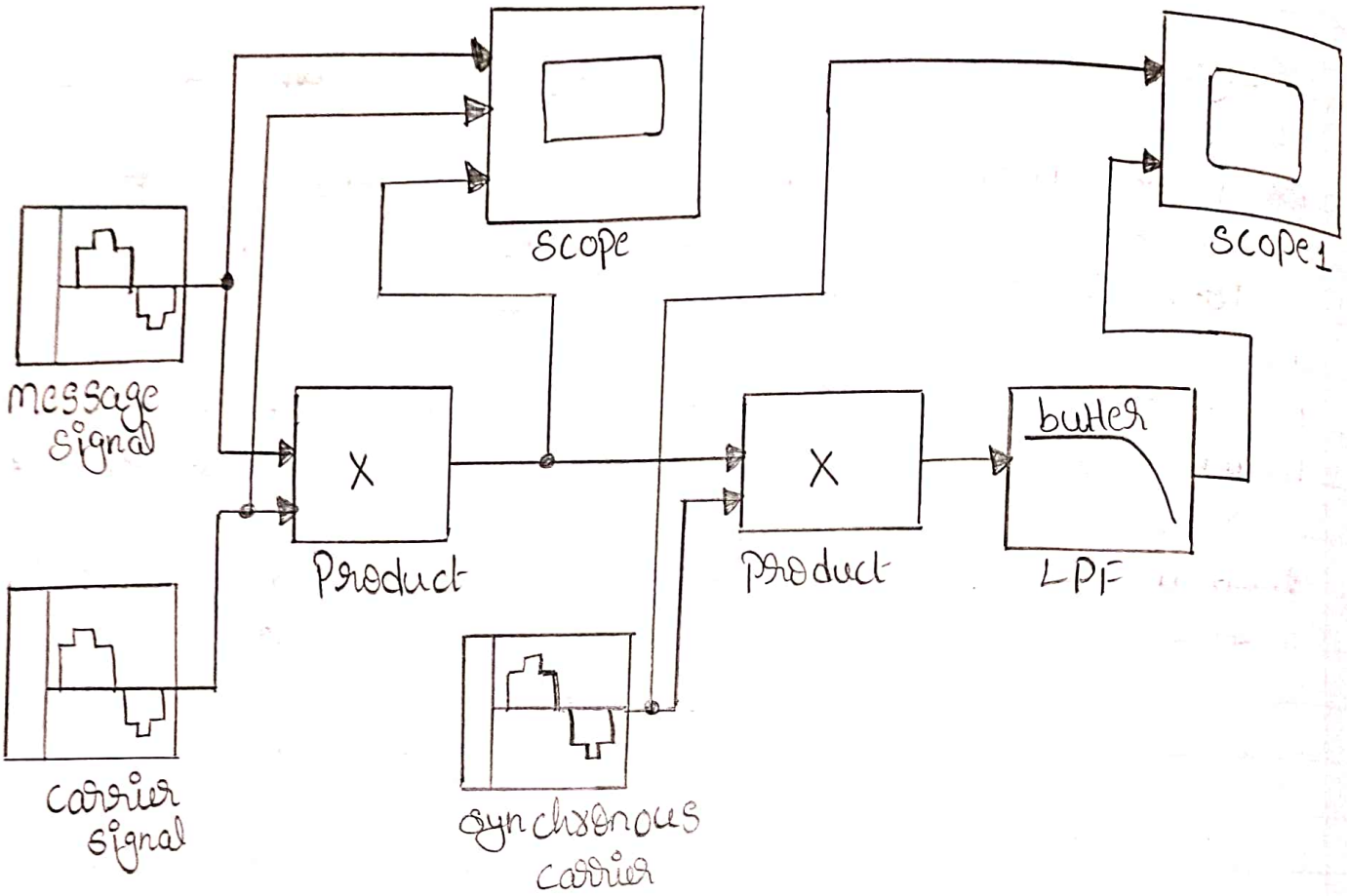
Teacher's Signature

150% modulation

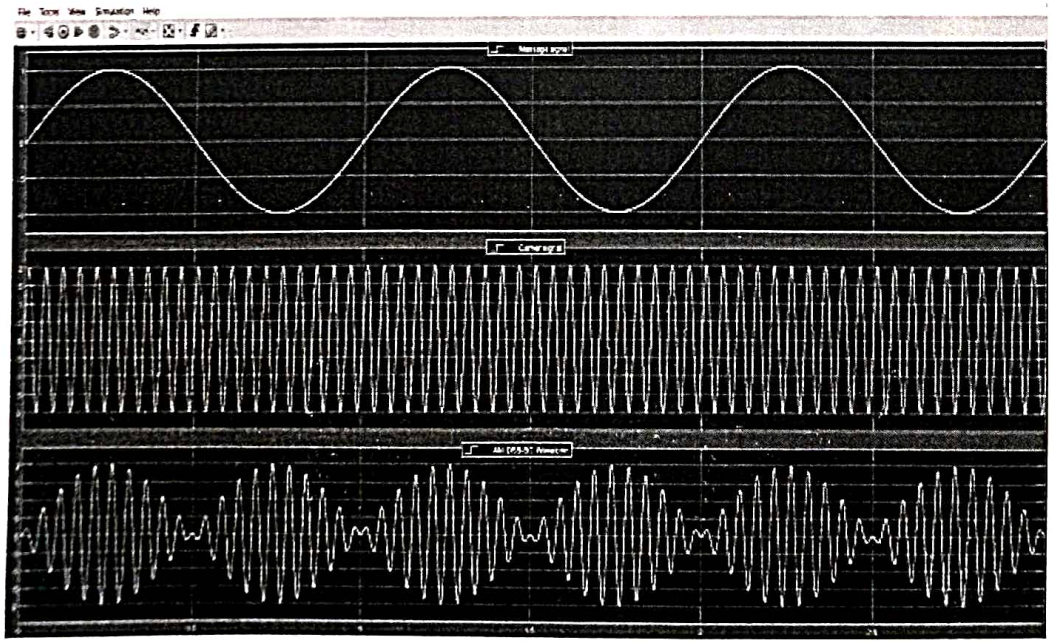


150% Demodulation

Simulink block diagram



DSB-SC Modulation



Date _____

Expt. No. 05 Page No. _____

DSB-SC Modulation & Demodulation

Aim:- TO Perform the DSB-SC signal Generation and Detection using Simulink.

Apparatus Required:

ⓐ Hardware TOOLS: computer system
ⓑ Software TOOL: MATLAB 7.0 and above version

Theory:

Amplitude modulations characterized by an information bearing carrier amplitude A(t) that is a linear function of the baseband m(t).

At the same time, the carrier frequency f_c and phase remain constant

If the carrier amplitude is made directly proportional to the modulating signal m(t), then modulated signal is

$$m(t)\cos 2\pi f_c t$$

$$m(t)\cos 2\pi f_c t \Longleftrightarrow \frac{1}{2}[M(f-f_c) + M(f+f_c)]$$

→ the modulation process does not introduce a sinusoid at f_c for this reason it is called double-sideband suppressed carrier (DSB-SC) modulation.

→ To recover the original signal m(t) from the modulated signal it is necessary to retranslate the spectrum to its original position.

Teacher's Signature _____

DSB-SC Demodulation

Date _____

Expt. No. _____ Page No. _____

The process of recovering the signal from the modulated signal (retranslating the spectrum to its original position) is referred to as demodulation.

$$f(t) = \frac{1}{2} m(t)$$

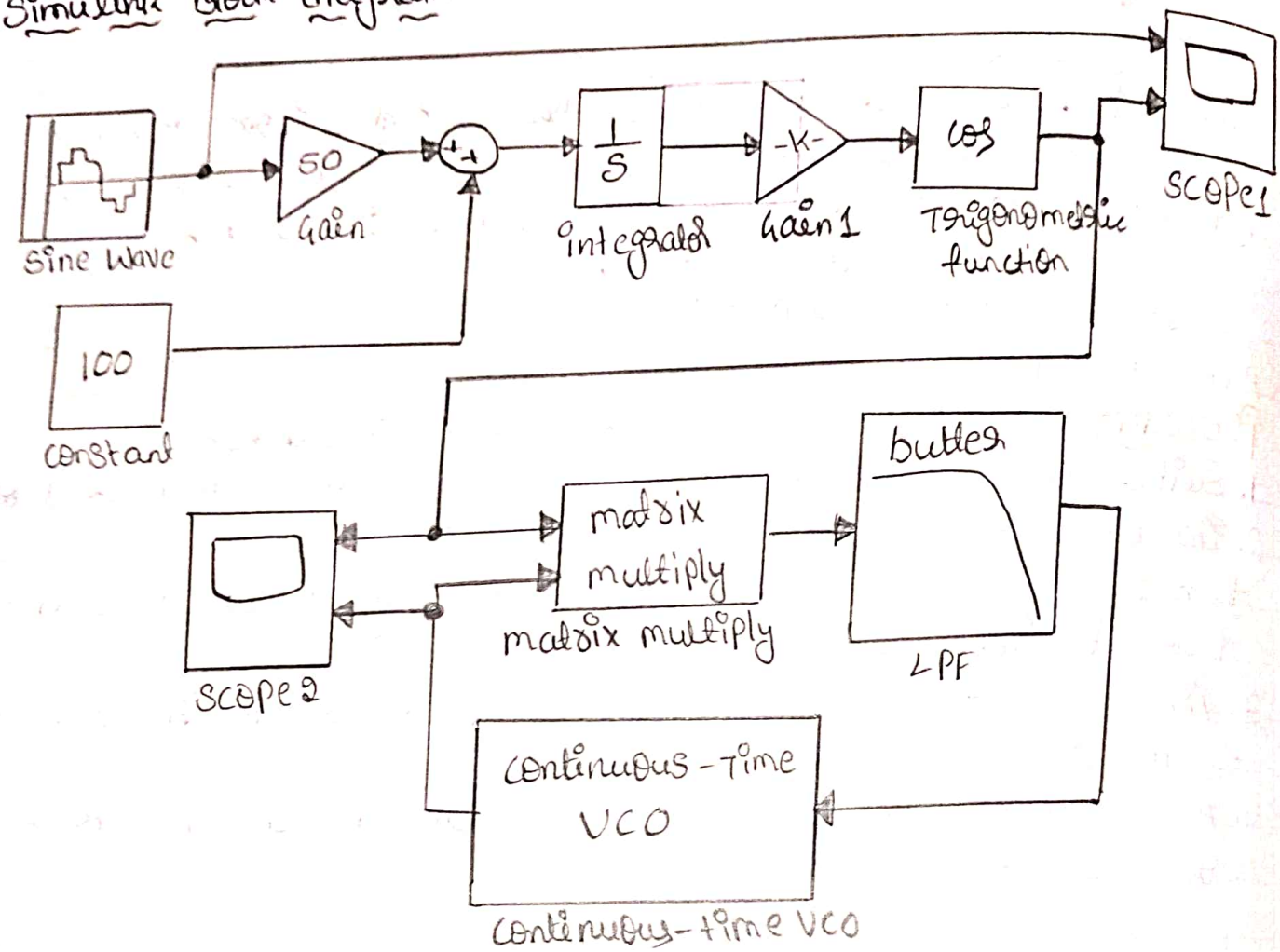
Procedure:-

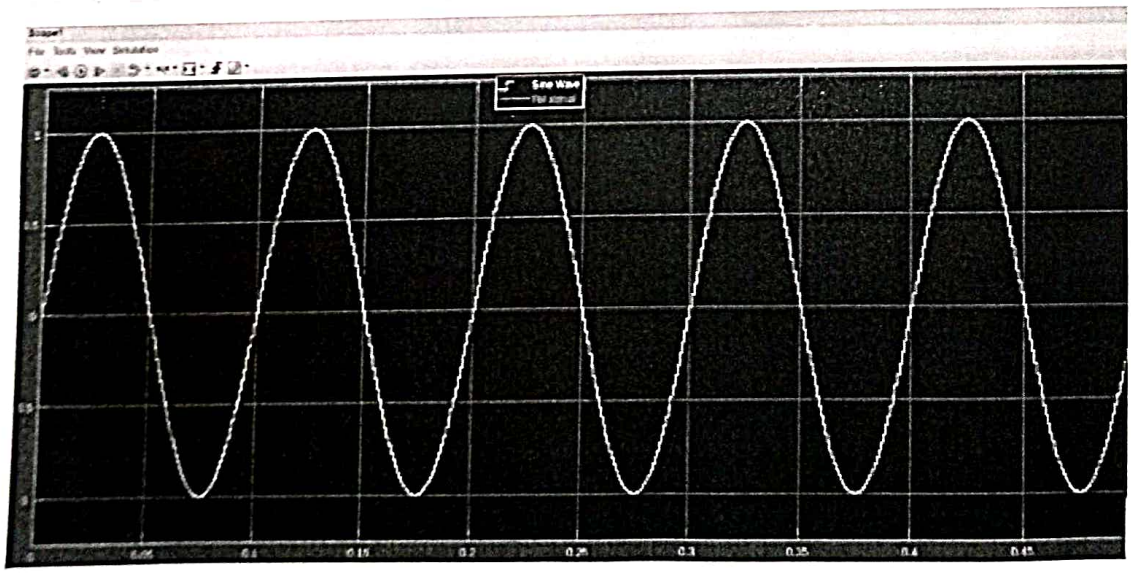
1. Switch on the computer and click on the MATLAB icon.
2. Go to Start at the bottom of the command window. then select "Simulink" then go to library browser and drag it into creating file.
3. Arrange the functional blocks as shown in simulink model
4. Assign required parameters to each functional block
5. Observe the outputs on scope.

Result:-

Teacher's Signature _____

Simulink Block diagram





Date

Expt. No. 06

Page No.

Frequency modulation and Demodulation

Aim:- To perform the frequency modulation signal Generation and Detection using simulink.

Apparatus Required:-

(a) Hardware Tools: computer system

(b) Software Tool: MATLAB 7.0 or above version

Theory:-

Frequency modulation is the process in which information (message signal) is transmitted over a carrier wave by varying its instantaneous frequency. The difference between instantaneous frequency and central frequency of carrier will be directly proportional to the instantaneous value of the amplitude of message signal 555 timer wired in astable made can be used for generating FM waves

FM equation → $V = A \sin\{\omega_c t + (\frac{\Delta f}{f_m}) \sin \omega_m t\}$

$$V = A \sin\{\omega_c t + m_f \sin \omega_m t\}$$

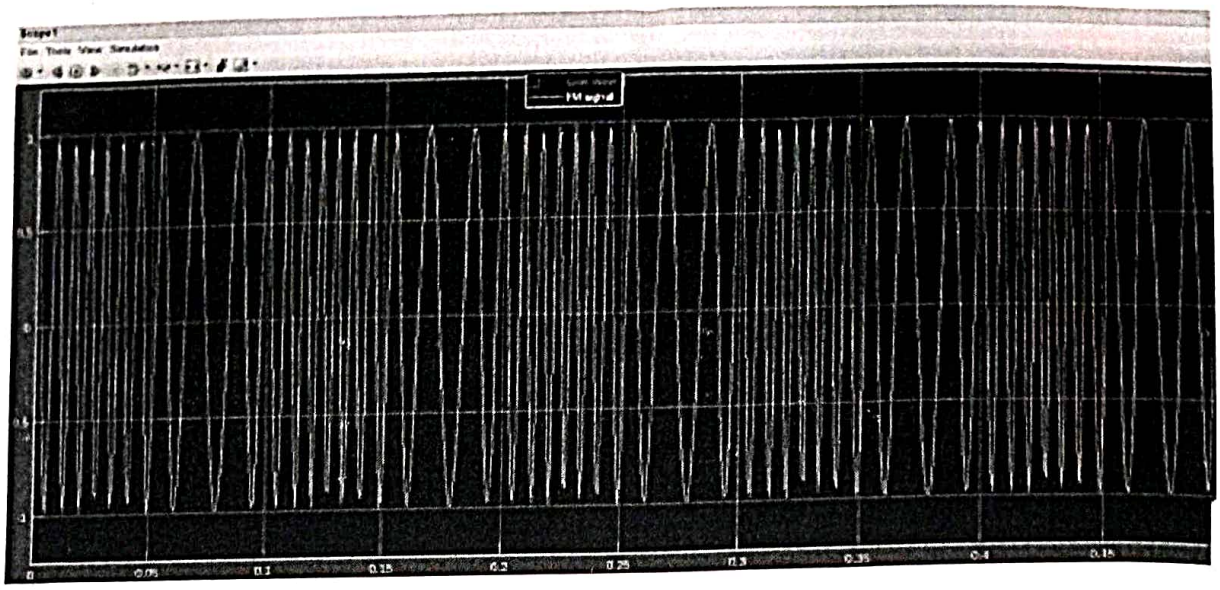
where → A = Amplitude of the FM signal

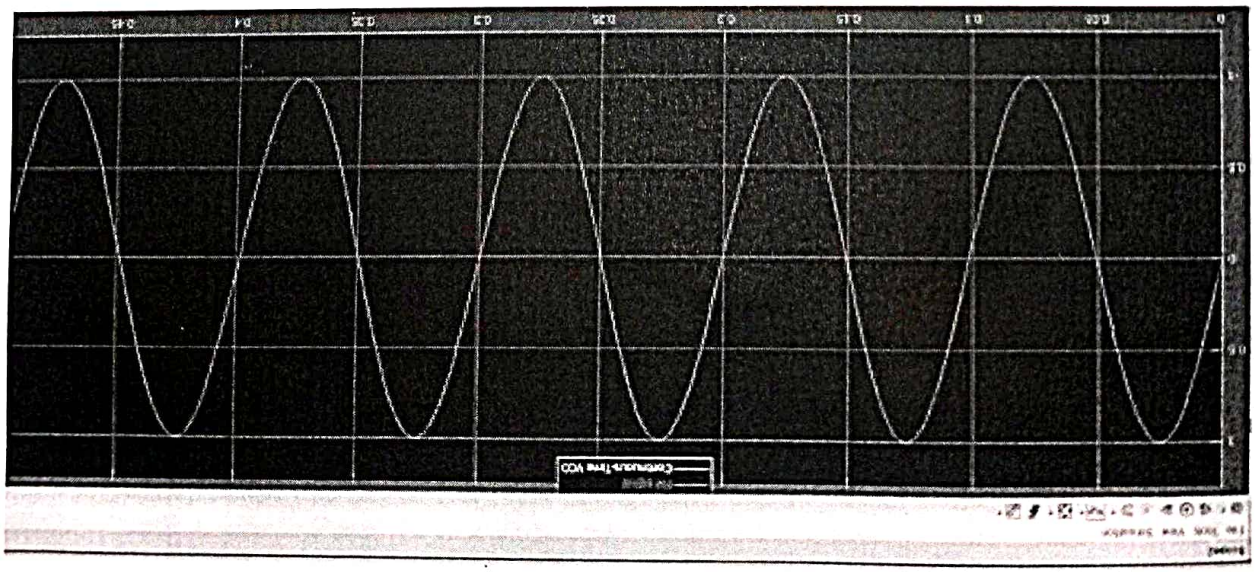
Δf = frequency deviation

m_f = modulation index of FM

$m_f = \frac{\Delta f}{f_m}$

Teacher's Signature


Scope1
File Tools View Simulation
FM signal
1
0.5
0
-0.5
-1
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45


Sinewave-Time VCO

Date _____

Expt. No. _____ Page No. _____

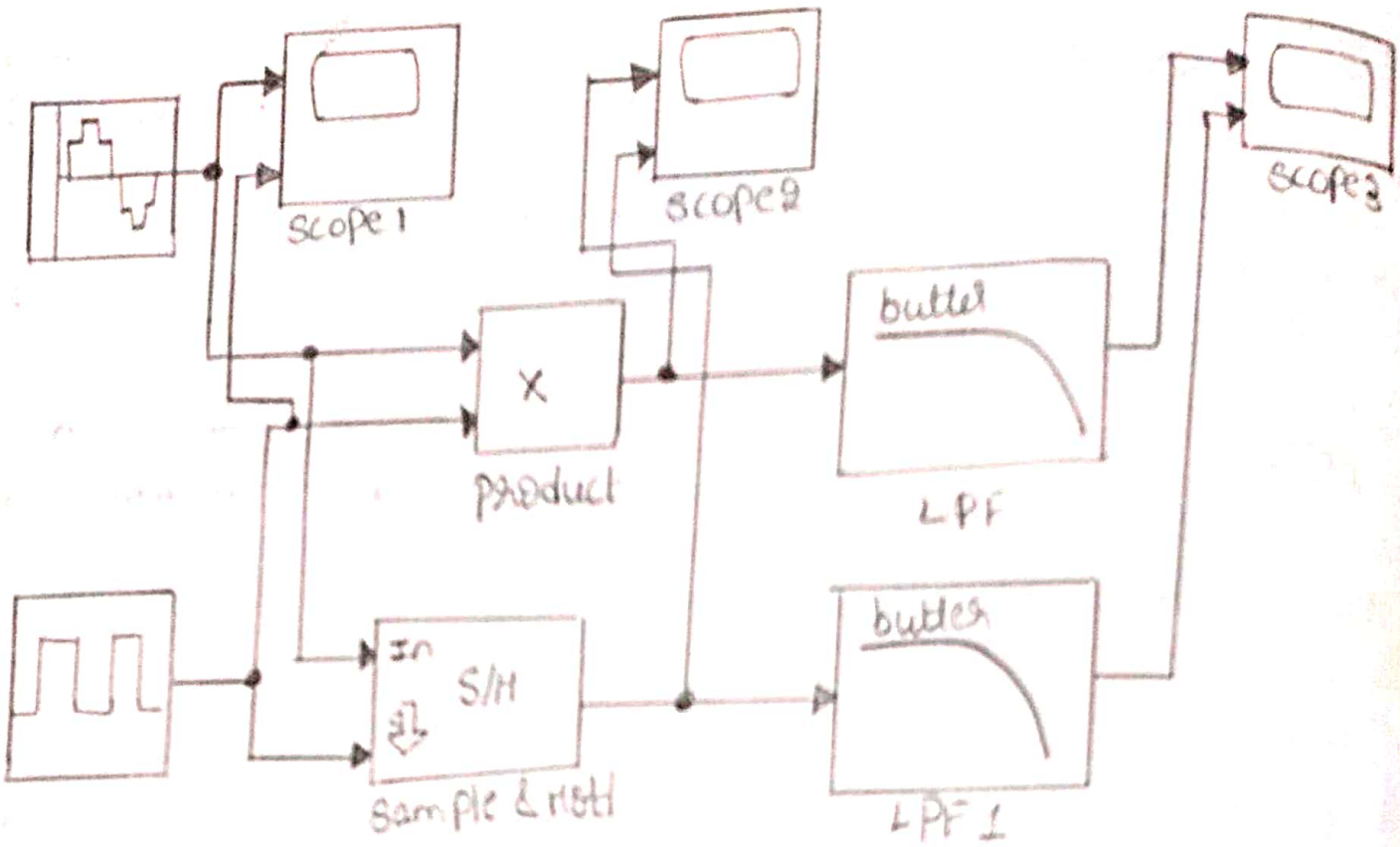
Procedure:-

1. Switch on the computer and click on the MATLAB icon
2. Go to start at the bottom of the command window. then go to library browser and drag it into existing file
3. Arrange the functional block as in simulink model.
4. Assign required parameters to each functional block.
5. observe the outputs on scope.

Result:-

Teacher's Signature _____

Simulink Block diagram



Date _____

Expt. No. 07 Page No. _____

PAM, PWM, PPM and PCM

Aim: To perform PAM, PWM, PPM and PCM using Simulink

Apparatus Required:

(a) Hardware Tools: Computer System

(b) Software Tool: MATLAB 7.0 & above version

(1) Pulse Amplitude modulation

Theory: PAM is the simplest pulse modulation scheme. In pulse amplitude modulation system, the amplitude of a carrier pulse train is varied in accordance with the instantaneous level of the modulating signal.

⇒ pulses can be of a rectangular form & some other appropriate shape

→ In natural PAM, a signal sampled at the Nyquist rate is reconstructed by passing it through an efficient LPF with exact cutoff frequency.

→ though the PAM signal is passed through an LPF it cannot recover the signal without distortion.

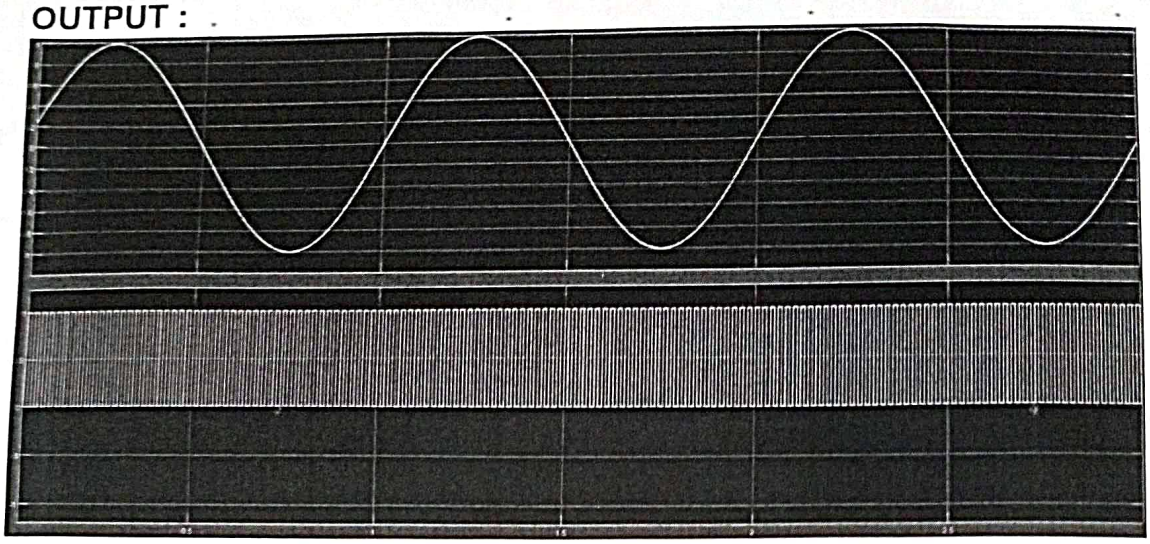
⇒ Hence to avoid this noise, flat-top sampling is done

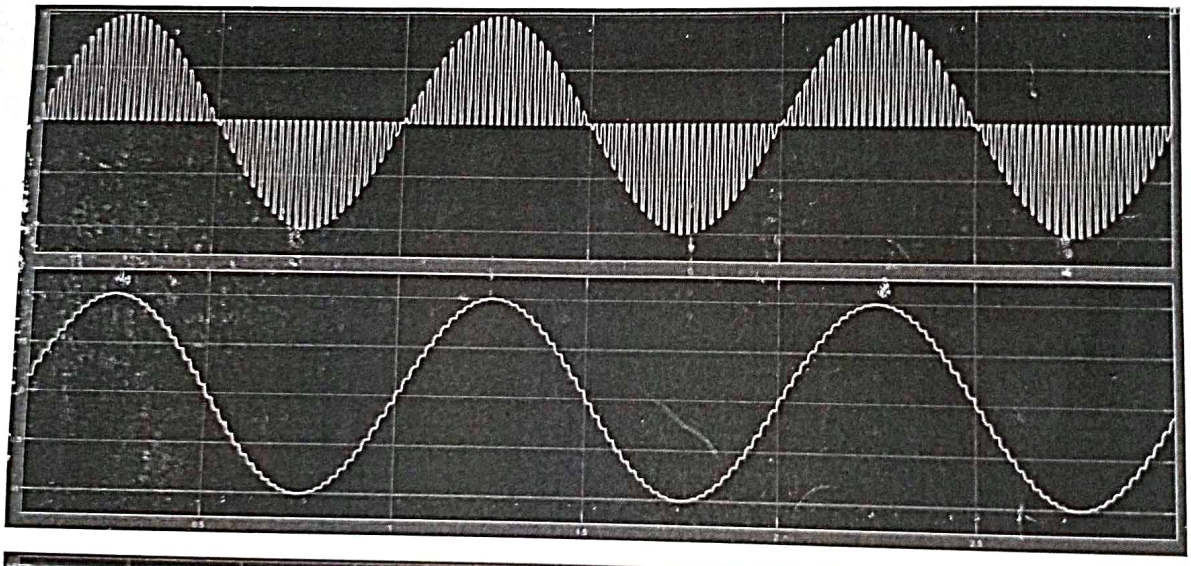
→ Flat-top sampling is the process in which sampled signal can be represented in pulses in which the amplitude of the signal can't be changed w.r.to the analog signal, to be sampled

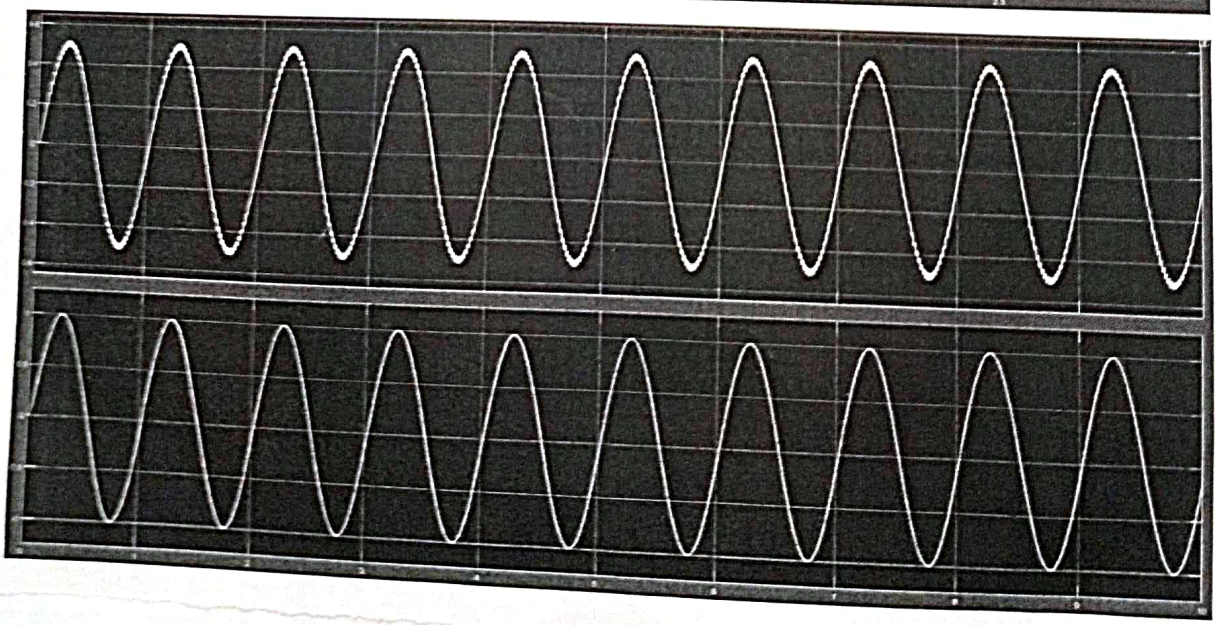
→ the tops of amplitude remain flat. this process simplifies the ckt design.

Teacher's Signature _____

OUTPUT :







Expt. No. _____ Date _____

Page No. _____

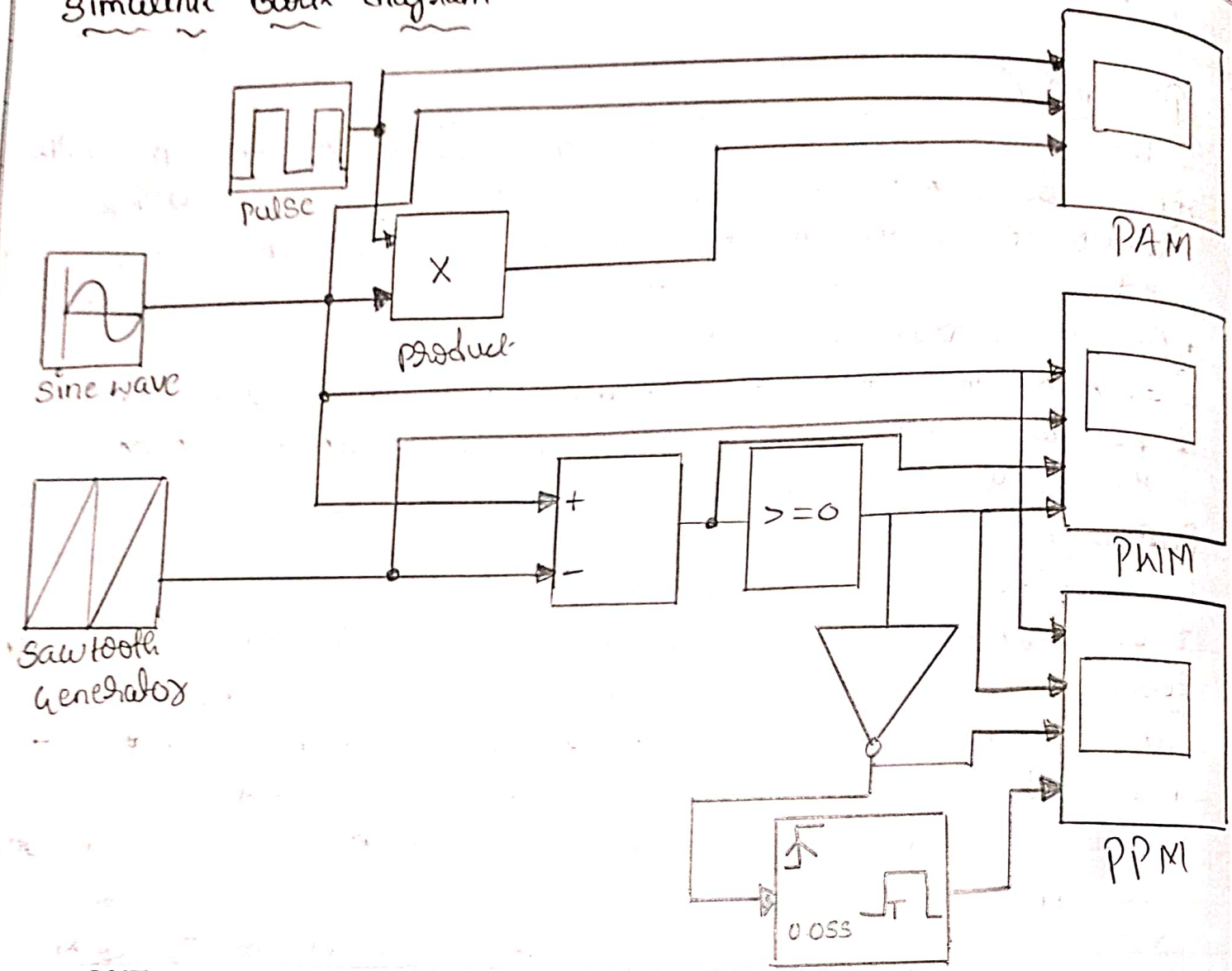
Procedure:

1. Switch on the computer and click on the MATLAB icon
2. GO to Start at the bottom of the command window then select "simulink" then go to library browser and drag it into creating file.
3. Arrange the functional blocks as shown in simulink model
4. Assign required parametexs to each functional block.
5. observe the output on scope

Result:

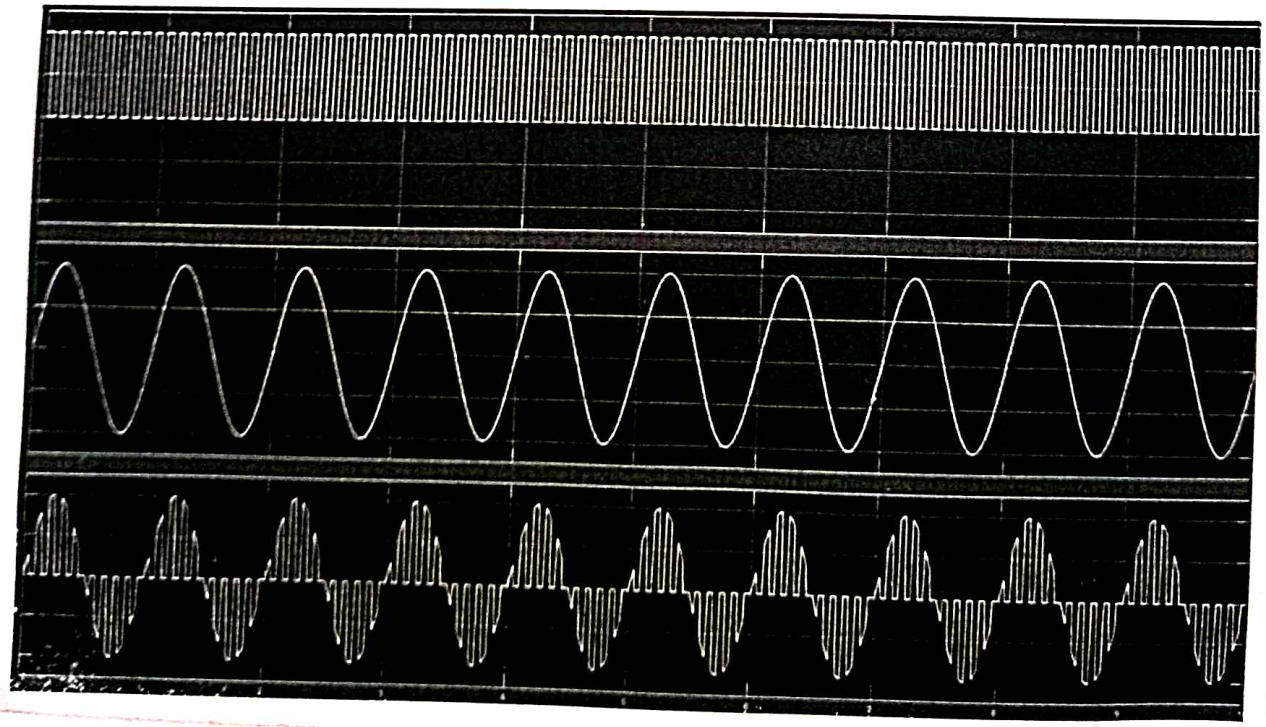
Teacher's Signature _____

Simulink Block diagram



OUTPUT :

PAM DIAGRAM :



Date

Expt. No. Page No.

② Pulse width and pulse position modulation

→ In pulse-duration modulation (PDM), the samples of the message signal are used to vary the duration of the individual pulses.

→ This form of modulation is also referred to as pulse-width modulation & pulse-length modulation

→ the modulating signal may vary the time of occurrence of the leading edge, the trailing edge & both edges of the pulse

→ PDM is wasteful of power in that long pulses expend considerable power during the pulse while leaving the pulse no additional information.

→ If this unused power is subtracted from PDM, so that only time transitions are essentially preserved we obtain a more efficient type of pulse modulation known as pulse-position modulation.

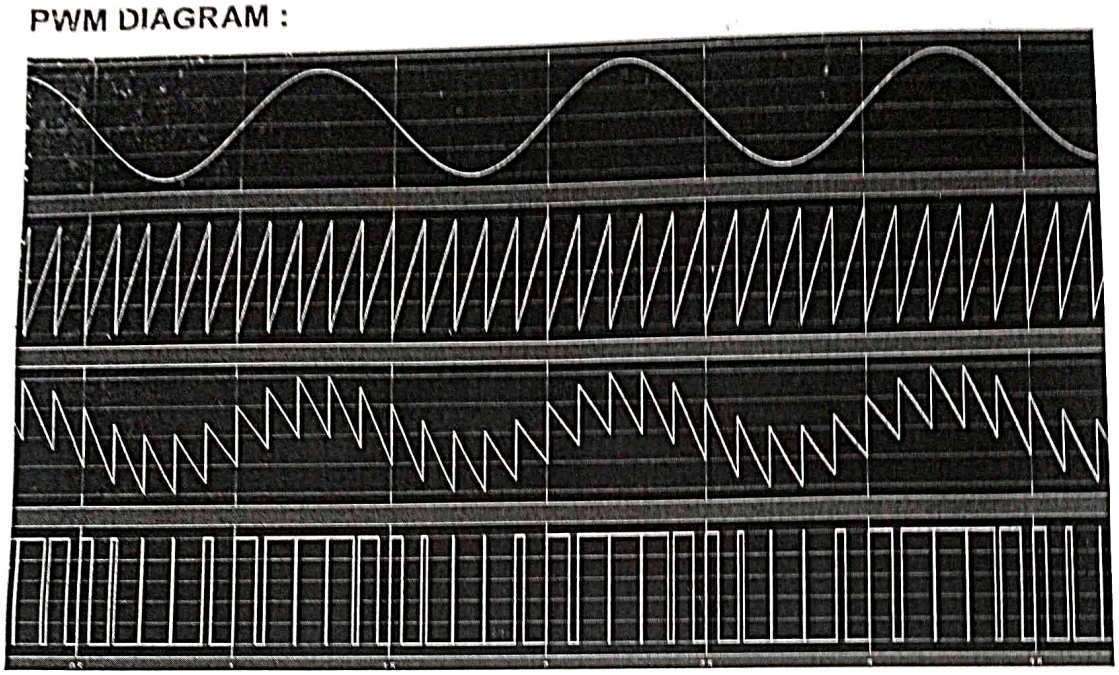
→ In PPM, the position of a pulse relative to its unmodulated time of occurrence is varied in accordance with the message signal.

Procedure:

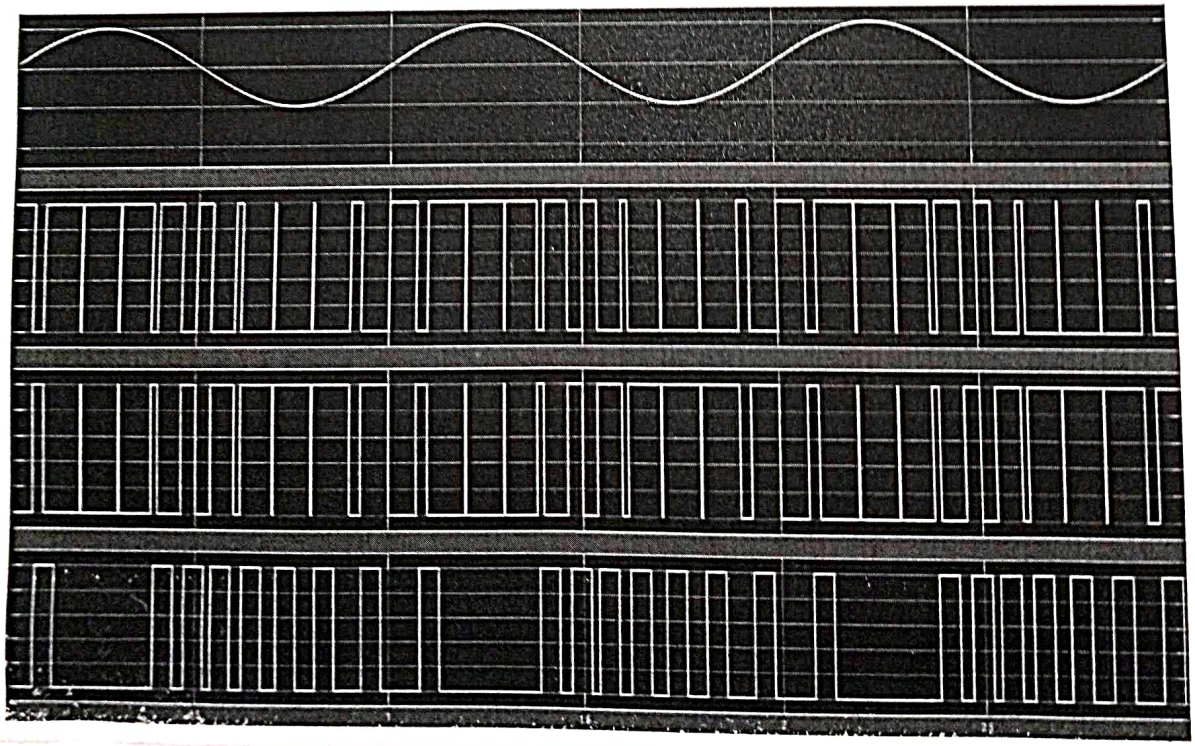
1. Switch on the computer and click on the MATLAB icon.
2. Go to start at the bottom of the command window then select "simulink" then go to library browser and drag it into creating file
3. Arrange the functional blocks as shown in simulink model.

Teacher's Signature

PWM DIAGRAM :



PPM DIAGRAM :



Date _____

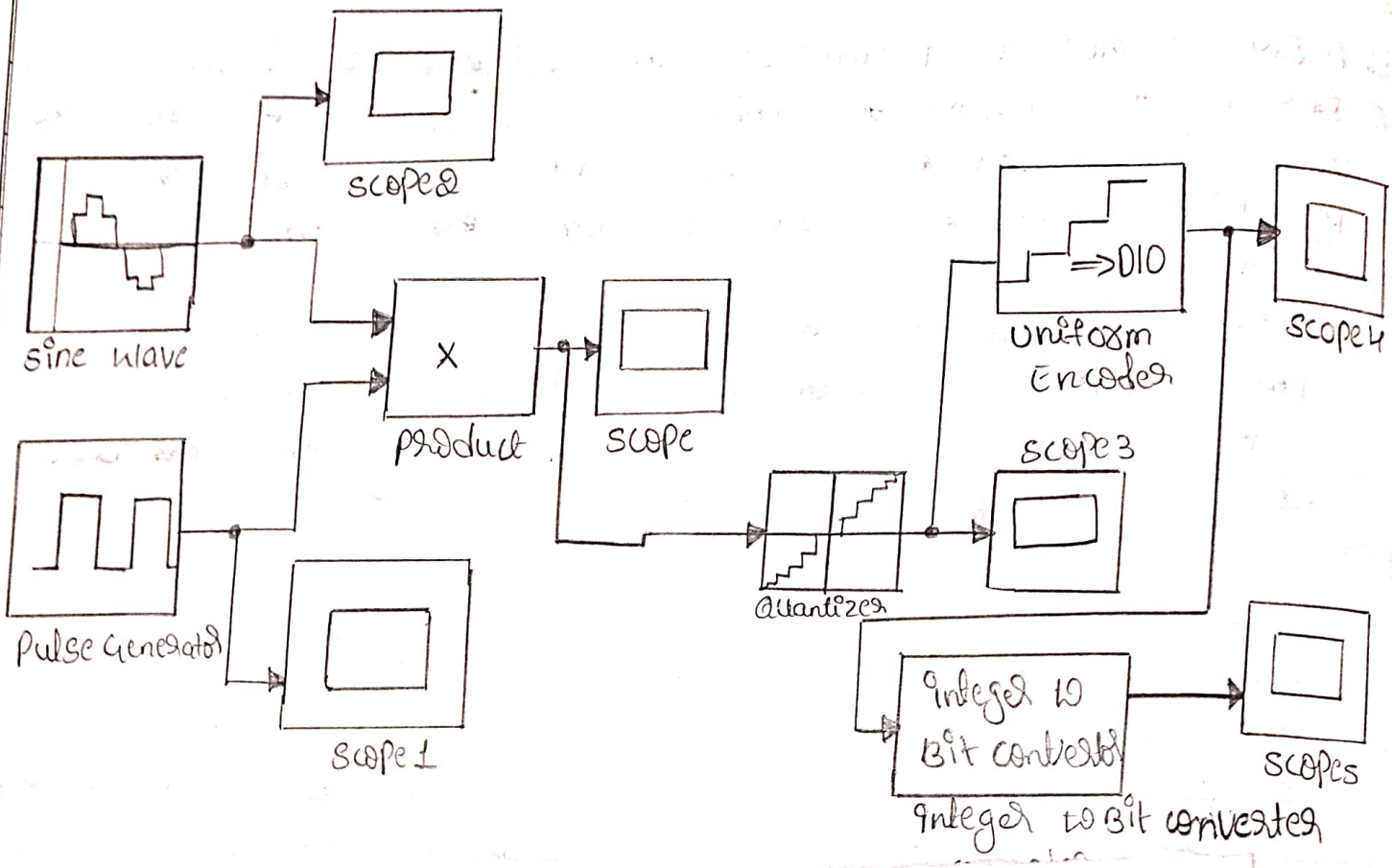
Expt. No. _____ Page No. _____

4. Assign required parameters to each functional block
5. Observe the outputs on scope

Result:

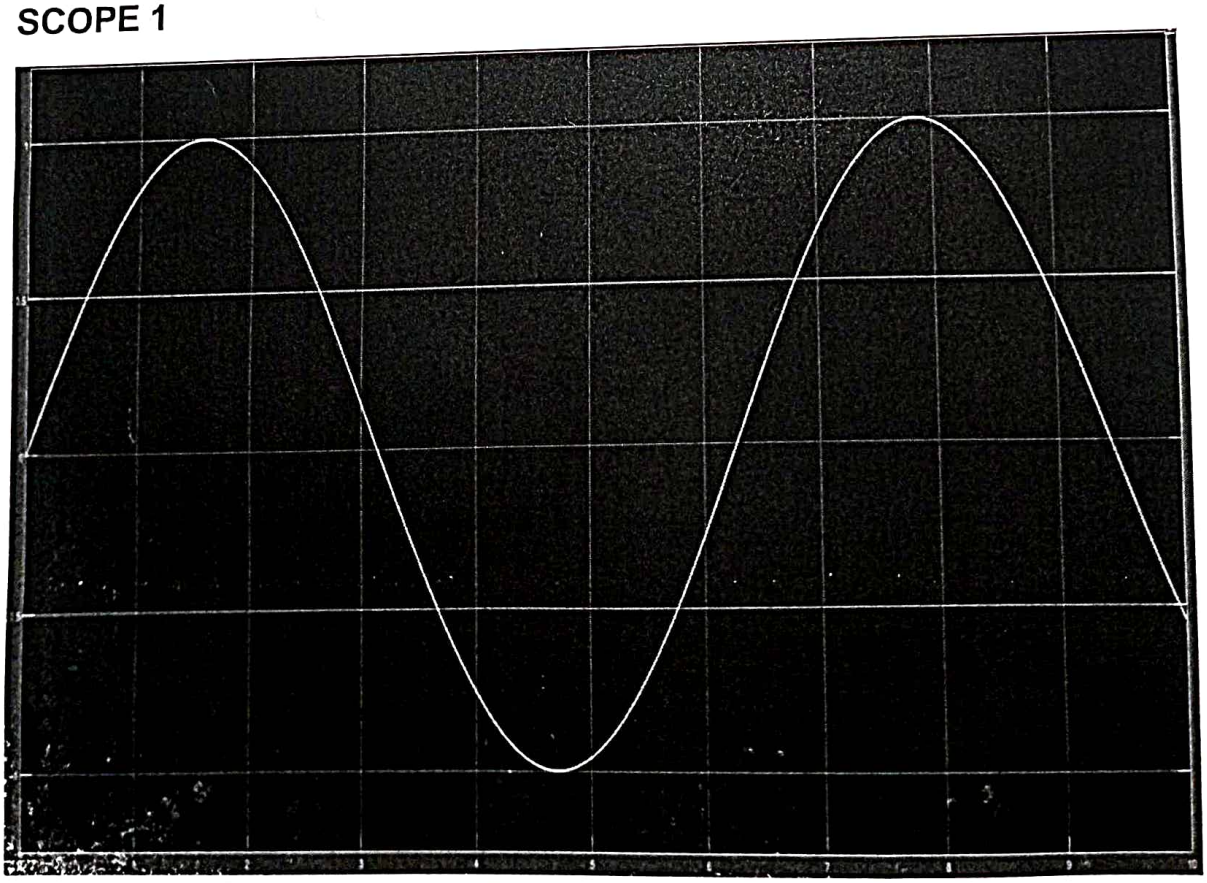
Teacher's Signature _____

Simulink block diagram



OUTPUT :

SCOPE 1



Date _____

Expt. No. _____ Page No. _____

(3) Pulse code modulation:

PCM basically is a tool for converting an analog signal into a digital signal (A/D conversion)

→ An analog signal is characterized by an amplitude that can take on any value over a continuous range. This means that it can take on an infinite no. of values

→ Digital signal amplitude can take on only a finite no. of values

→ An analog signal can be converted into a digital signal by means of sampling and quantizing, that is rounding off its value to one of the closet permissible numbers of quantized levels

→ the amplitude of the analog signal m(t) lie in the range (-mp, mp), which is portioned into L subintervals, each of magnitude $\Delta v = 2mp/L$

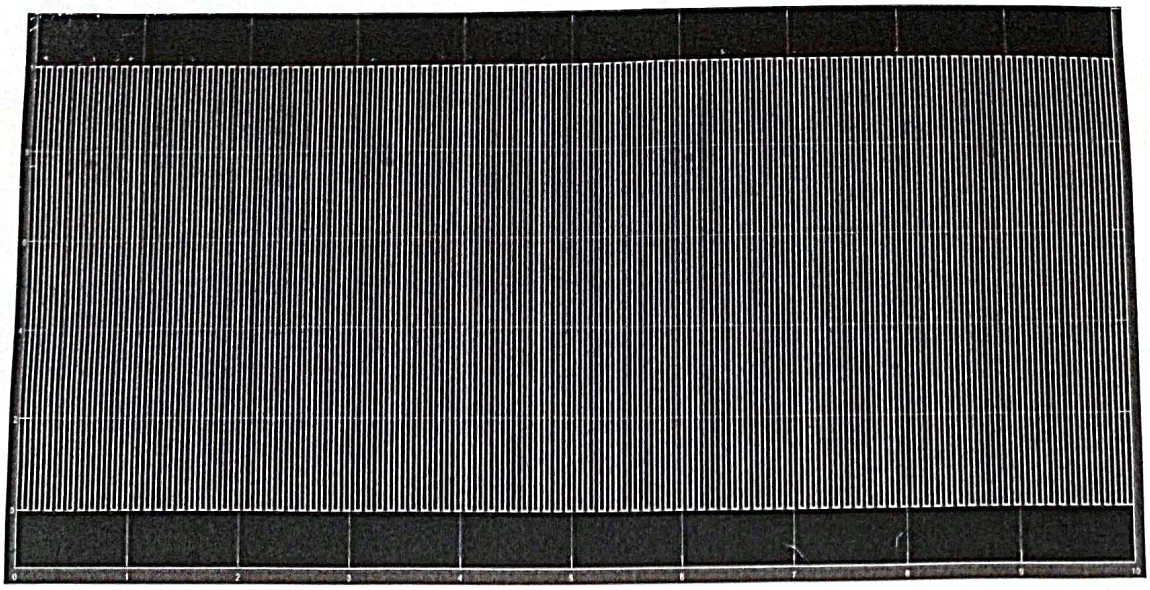
→ Next, each sample amplitude is approximated by the midpoint value of the subinterval in which the sample falls. Each sample is now approximated to one of the L numbers

Procedure:

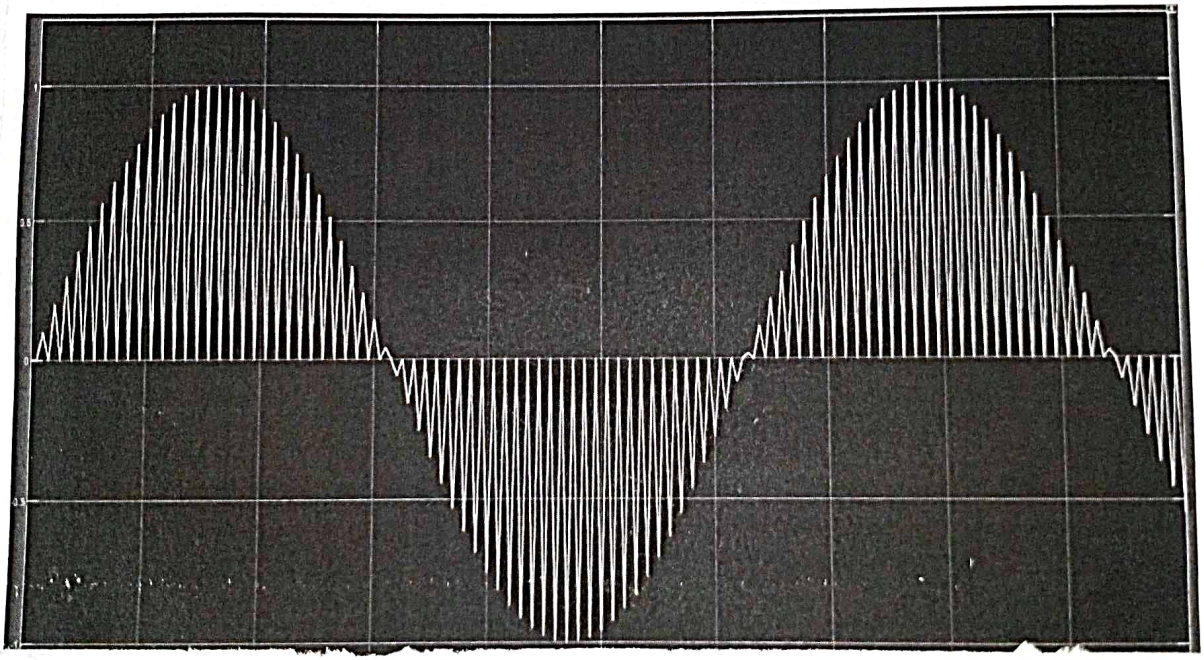
1. Switch on the computer and click on the MATLAB icon
2. Go to start at the bottom of the command window then select 'simulink' then go to library browser and drag it into creating file
3. Arrange the functional blocks as shown in simulink model
4. Assign required parameters to each functional block
5. Observe the outputs on scope

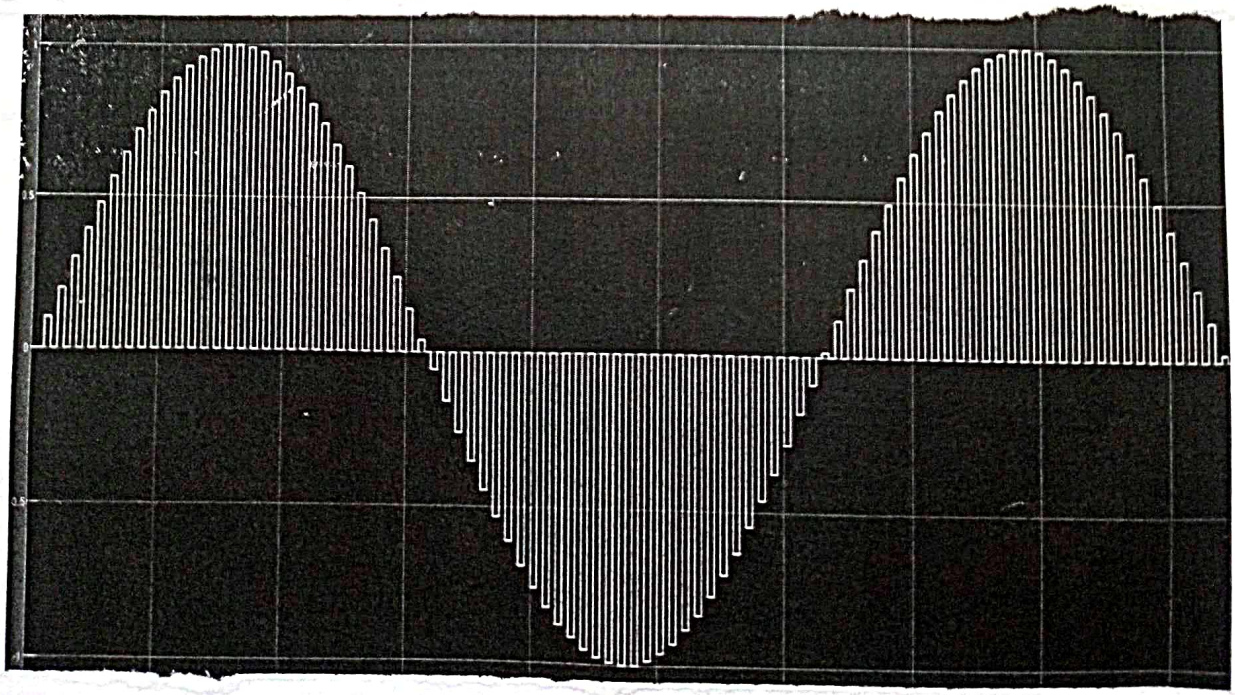
Teacher's Signature _____

SCOPE 2



SCOPE 3



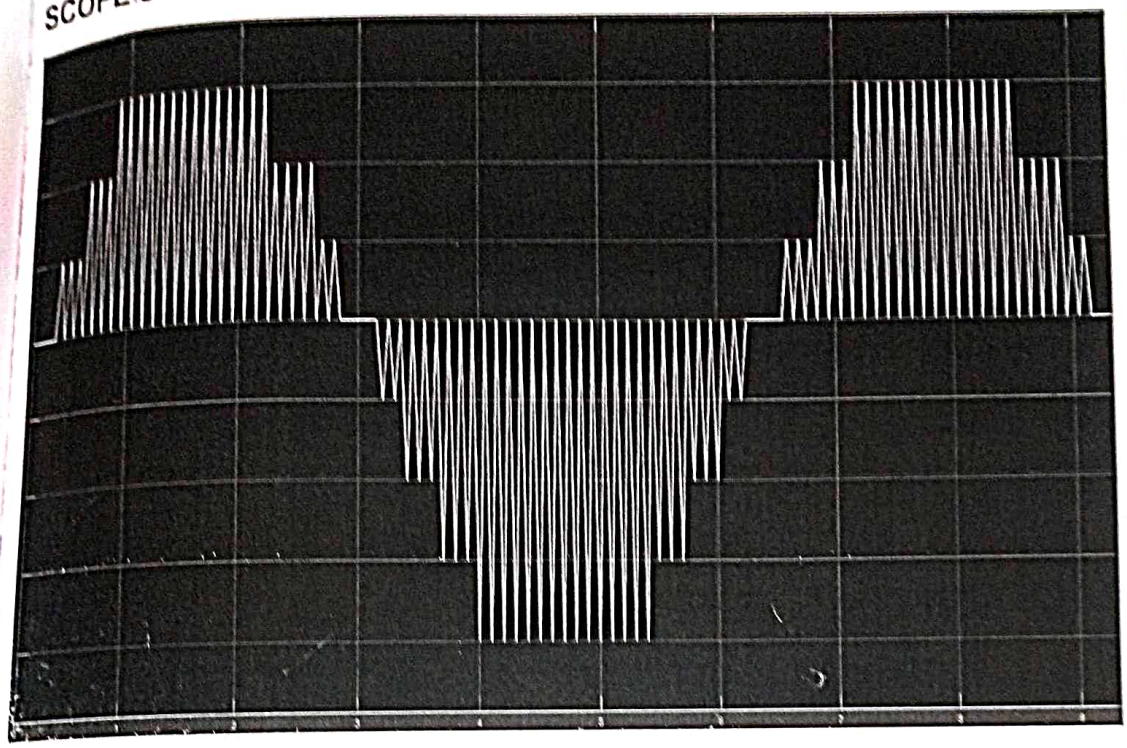


Date _____

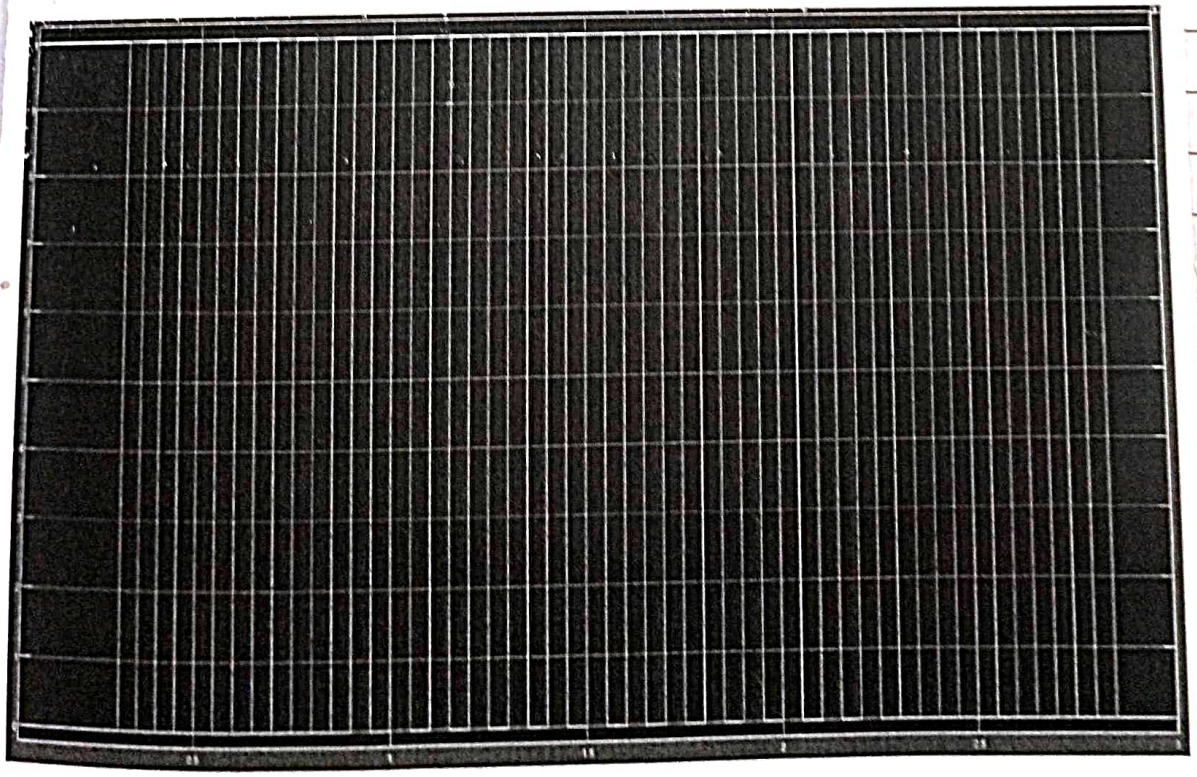
Expt. No. _____ Page No. _____

Result :-

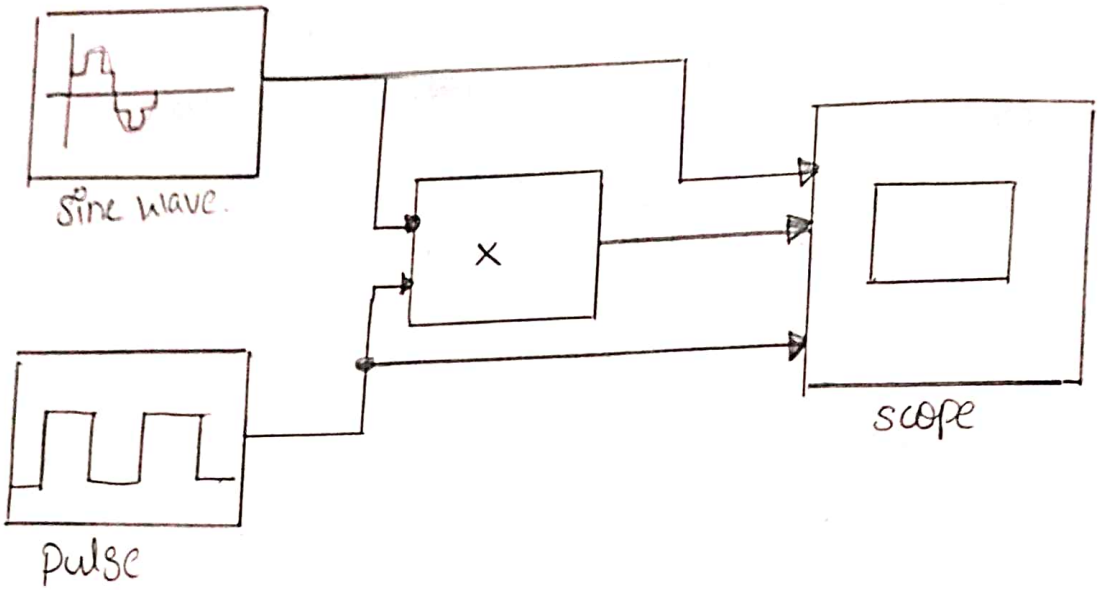
SCOPE 5



SCOPE 6



Block Diagram
Amplitude shift keying modulation



Amplitude Shift Keying Demodulation using MATLAB Code

```
clc;
clear all;
close all;
Tb=1; fc=10; % Carrier signal.
t=0:Tb/100:1;
C=sqrt(2/Tb)*sin(2*pi*fc*t);
% generation message signal.
N=8;
m=rand(1,N);
t1=0; t2=Tb
for i=N:N
  t=[t1:.01:t2]
if m(i)=1;
m_s=ones(1,length(t));
else
m(i)=0;
m_s=ones(1,length(t));
end
```

Date _____

Expt. No. 08 Page No. _____

(a) Amplitude Shift Keying modulation

Aim: To Study Amplitude Shift Keying (modulation & Demodulation)

Apparatus Required:

(a) Hardware Tools: Computer System

(b) Software Tool: MATLAB 7.0 and above version.

Theory:

Amplitude Shift Keying (ASK) is a digital modulation scheme where the binary data is transmitted using a carrier signal with two different amplitude levels.

For binary 0 and 1, the carrier switches b/w these two levels. In its simplest form, a carrier is sent during one i/p and no carrier is sent during the other. This kind of modulation scheme is called on-off keying.

BASK

$$b(t) = \begin{cases} \sqrt{E_b}, & \text{for binary symbol 1} \\ 0, & \text{for binary symbol 0} \end{cases}$$

$$s(t) = \begin{cases} \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t), & \text{for symbol 1} \\ 0 & \text{for symbol 0} \end{cases}$$

Procedure:

→ Open MATLAB

→ Open new file in Simulink.

→ connect blocks for modulation.

→ write code for Demodulation.

→ Observe wave forms.

Teacher's Signature _____

```
message(i,:)=m_s;
%.product of carrier and message
ask_sig(i,:)=c.*m_(s);
t1=t1+(Tb+.01);
t2=t2+(Tb+01);
%.plot the message and ASK signal.
sub(5,1,2);axis([0 N -2 2]);plot(t,message(i,:):r');
title('message signal');xlabel('t--->');ylabel('m(t)');grid on
hold on
subplot(5,1,4);plot(t,ask_sig(i,:));
title('message signal');xlabel('t--->');ylabel('s(t)');grid on
hold on
and
hold off
subplot(5,1,3);plot(t,c);
title('carrier signal');xlabel('t--->');ylabel('c(t)');grid on.
subplot(5,1,1);stem(m);
title('binary data bits');xlabel('n--->');ylabel('b(n)');grid on
% ASK Demodulation
t1=0; t2=Tb
for i=1:N
t=[t1:Tb/100:t2]
x=sum(c.*ask_sig(i,:));% correlator.
if x>0  % decision device
demod(i)=1;
else
demod(i)=0;
end
t1=t1+(Tb+.01);
t2=t2+(Tb+01);
end
subplot(5,1,5);stem(demod);
title('ASK demodulated signal');xlabel('n--->');ylabel('b(n)');grid on
```

Date

Expt. No. Page No.

(6) Amplitude Shift Keying Demodulation

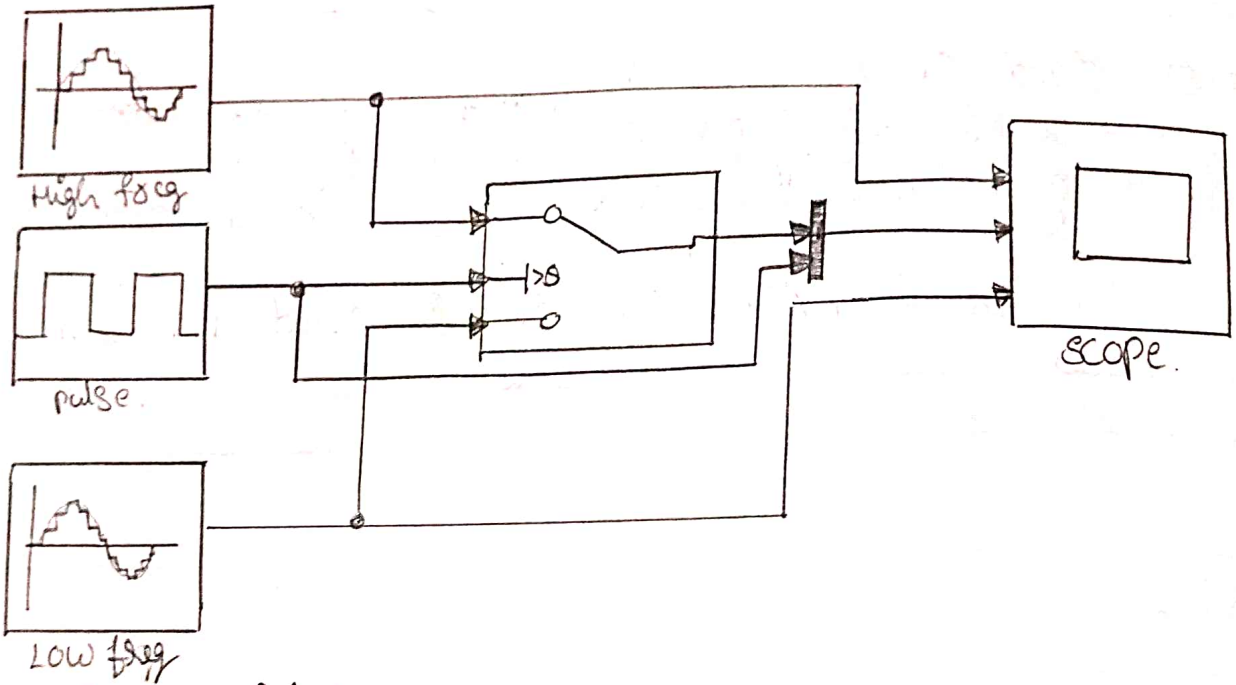
→ BASK signal is generated by using a product modulator with two inputs. One input, the ON/OFF signal i.e is the modulating signal the sinusoidal carrier wave c(t).

→ Detection of the BASK wave is concerned, the simplest is way to use an envelope detector. exploiting the nonconstant envelope property of BASK signal.

Result:-

Teacher's Signature

Block diagram:-
modulation



code for Demodulation:-

```
clc;
clear all;
close all;
% Generate carrier signal
Tb=1; fc1=2; fc2=5;
t=0:(Tb/100):Tb;
c1 = sqrt(2/Tb)*sin(2*pi*fc1*t);
c2 =sqrt(2/Tb) * sin(2*pi*fc2*t);
% generate message signal
N=8
m =rand(1,N);
t1=0; t2=Tb
for i=1:N
   t =[t1:(Tb/100):t2]
if m(i)>0.5
m(i)=1;
```

Date

Expt. No. 09

Page No.

Frequency Shift Keying

Aim: To study frequency shift keying modulation and demodulation.

Apparatus required:

(a) Hardware TOOLS: computer system

(b) Software TOOLS: MATLAB 7.0 and above version

Theory:

Binary frequency-shift keying (BFSK): In which the carrier amplitude and carrier phase are both maintained constant while the carrier frequency is keyed btw the two possible values used to represent symbol 0 and 1.

$$s_i(t) = \begin{cases} \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_1 t), & \text{for symbol 1 corresponding to } i=1 \\ \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_2 t), & \text{for symbol 0 corresponding } i=2 \end{cases}$$

→ where E_b is the transmitted signal energy per bit when the frequencies f_1 and f_2 are chosen in such a way that they differ from each other by an amount equal to the reciprocal of the bit duration the BFSK signal is referred to as Sunde's BFSK after its originator.

→ this modulated signal is a continuous phase signal in the sense that phase continuity is always maintained including the inter-bit switching times.

Teacher's Signature

```
m_s = ones(1, length(t));
inv_m_s = ones(1, length(t));
end
message(i,:) = m_s;
% multiplier
fsk_sig1(i,:) = c1.*m_s;
fsk_sig2(i,:) = c2.*inv_m_s;
fsk = fsk_sig1 + fsk_sig2;
% plotting the message signal and the modulated signal
subplot(3,2,2); axis([0 N -2 2]); plot(t, message(i,:),'r');
title('message signal'); xlabel('t--->'); ylabel('m(t)'); grid on;
subplot(3,2,5); plot(t, fsk(i,:));
title('FSK signal'); xlabel('t--->'); ylabel('s(t)'); grid on;
t1 = t1 + (Tb+0.01); t2 = t2 + (Tb+0.01);
end
hold off
% plotting binary data bits and carrier signal
subplot(3,2,1); stem(m);
title('binary data'); xlabel('n--->'); ylabel('b1(n)'); grid on;
subplot(3,2,3); plot(t, c1);
title('carrier signal-1'); xlabel('t--->'); ylabel('c1(t)'); grid on;
subplot(3,2,4); plot(t, c2);
title('carrier signal-2'); xlabel('t--->'); ylabel('c2(t)');
grid on;

% FSK Demodulation
t1 = 0; t2 = Tb
for i = 1:N
t = [t1:(Tb/100):t2)
%. correlator
x1 = sum((c1.*fsk_sig1(i,:)));
x2 = sum((c2.*fsk_sig2(i,:)));
```

Expt. No. _____ Page No. _____

Procedure:

open MATLAB
open new m file in MATLAB
write code for Demodulation
open Simulink software
connect BLOCKS for modulation circuit
observe wave forms

Result:

Teacher's Signature _____

```
X = X1 - X2;
%. decision device
if X>0
demod(i)=1;
 else
demod(i)=0;
 end
t1=t1+(Tb+.01);
tQ = tQ+(Tb+01);
 end
% plotting the demodulated data bits
Subplot (3,2,6); Stem (demod);
title('demodulated data'); xlabel('n--->'); ylabel('bn)')
 grid on;
```

Date _____

Expt. No. 10 _____ Page No. _____

Phase Shift Keying

Aim: To generate and demodulate phase shift keyed (PSK) signal using MATLAB

Apparatus Required:

(a) Hardware Tools: Computer System

(b) Software Tool: MATLAB 7.0 and above version

Theory:

(a) Binary Phase-Shift Keying (BPSK)

BPSK, the pair of signals is used to represent symbols 1 and 0, respectively, are defined by

$$S_i(t) = \begin{cases} \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t), & \text{for symbol 1 corresponding to } i=1 \\ \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \pi), & \text{for symbol 0 corresponding to } i=2 \end{cases}$$

$$= -\sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t)$$

→ T_b denoting the bit duration and

→ E_b denoting the transmitted signal energy per bit

Teacher's Signature _____

Date

Expt. No. Page No.

(b) Quadriphase - Shift Keying (QPSK)

->An important goal of digital communication is the efficient utilization of channel bandwidth

->This goal is attained by a bandwidth-conserving modulation scheme known as QPSK

$$S_i(t) = \begin{cases} \sqrt{\frac{2E}{T}} \cos\left[2\pi f_c t + (2i-1)\frac{\pi}{4}\right], & 0 \le t \le T \\ 0, & \text{elsewhere} \end{cases}$$

Procedure:
open MATLAB
open new file
Type Program
Save in current directory
compile and Run the program
for the output see command window / figure window

code for BPSK

```
% BPSK modulation
clc;
clear all;
close all;
%generate carrier signal
Tb=1;
```

Teacher's Signature

Date

Expt. No. Page No.

```
t=0:Tb/100:Tb;
fc=8
c=sqrt(2/Tb)*sin(2*pi*fc*t);
% generate message signal
N=8
m=rand(1,N);
t1=0; t2=Tb
for i=1:N
t=[t1:.01:t2]
if m(i)>0.5
m(i)=1;
m_s=ones(1,length(t));
else
m(i)=0
m_s=-1*ones(1,length(t));
end
message(i,:)=m_s;
% Product of carrier and message signal
bpsk_sig(i,:)=c.*m_s;
% plot the message and BPSK modulated signal
subplot(5,1,2); axis([0 N -2 2]); plot(t,message(i,:),'r');
title('message signal (POLAR form)'); xlabel('t--->');
ylabel('m(t)');
grid on; hold on;
subplot(5,1,4); plot(t,bpsk_sig(i,:));
title('BPSK signal'); xlabel('t--->'); ylabel('s(t)');
grid on; hold on;
t1=t1+1.01; t2=t2+1.01;
end
```

Teacher's Signature

Date

Expt. No. Page No.

```
hold off
% plot the i/p binary data and carrier signal
subplot(5,1,1); stem(m);
title('binary data bits'); xlabel('n---->'); ylabel('b(n)');
grid on;
subplot(5,1,3); plot(t,c);
title('carrier signal') xlabel('t--->'); ylabel('c(t)');
grid on;

%.BPSK Demodulation
t1=0; t2=Tb
for i=1:N
t=[t1:.01:t2]
% correlator
x=sum(c.*bpsk_sig(i,:));
% decision device
if x>0
demod(i)=1;
else
demod(i)=0;
end
t1=t1+1.01;
t2=t2+1.01;
end
% plot the demodulated data bits
subplot(5,1,5) : stem(demod);
title('demodulated.') : xlabel('n--->'); ylabel('b(n)');
grid on;
```

Teacher's Signature

Date

Expt. No.

Page No.

code for QPSK

```
% QPSK modulation
clc;
clear all;
close all;
% Generate QPSK carrier signal
Tb=1; t=0:(Tb/100):Tb; fc=1;
c1=sqrt(2/Tb)*cos(2*pi*fc*t);
c2=sqrt(2/Tb)*sin(2*pi*fc*t);
% generate message signal
N=8; m=rand(1,N);
t1=0:2*Tb;
for i=1:2:(N-1)
t=[t1:(Tb/100):2*Tb]
if m(i)>0.5
m(i)=1;
m_s=ones(1,length(t));
else
m(i)=0;
m_s=-1*ones(1,length(t));
end
% Add bits modulated signal
add_sig(i,:)=c1.*m_s;
if m(i+1)>0.5 2
m(i+1)=1;
m_s=ones(1,length(t));
else
m(i+1)=0;
```

Teacher's Signature

Expt. No. _____ Date _____

Page No. _____

```
% decision device
if (x1>0 && x2>0)
demod(i) = 1;
demod(i+1) = 1;
elseif (x1>0 && x2<0)
demod(i) = 1;
demod(i+1) = 0
elseif (x1<0 && x2<0)
demod(i) = 0;
demod(i+1) = 1;
end
t1 = t1 + (Tb + 0.01); t2 = t2 + (Tb + 0.1);
end
subplot(3,2,5); stem(demod);
title('QPSK demodulated bits'); xlabel('n --->');
ylabel('b(n)'); grid on;
```

Result:

Teacher's Signature _____