**PRACTICAL FILE**

**BASIC INFORMATION THEORY AND COMMUNICATION (ESC X58)**

**BE(IT) 2nd Semester**



Submitted to Submitted by

**Ms Shewangi** **Pragya Sharma**

**UE228079**

**IT Section-2**

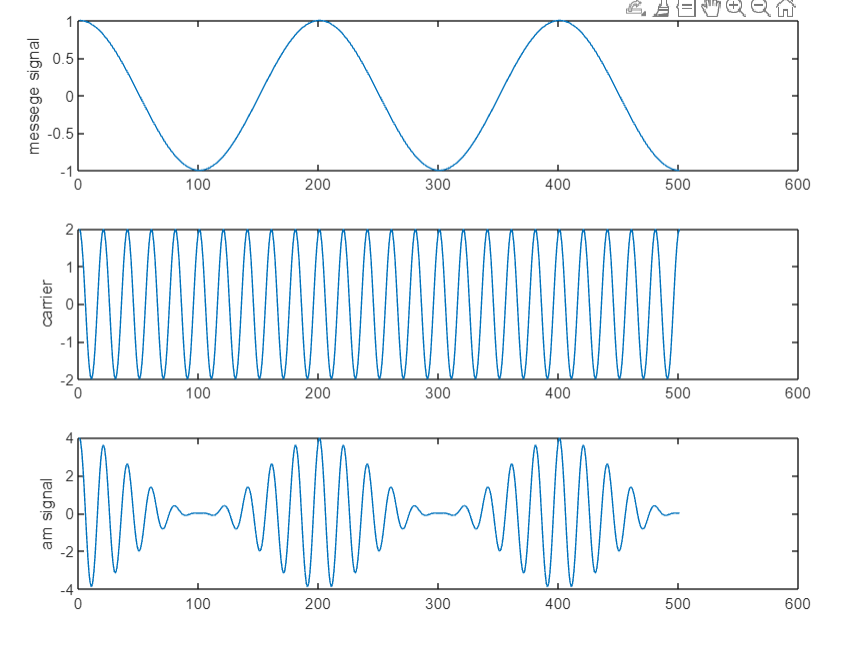
**DEPARTMENT OF INFORMATION TECHNOLOGY**

University Institute of Engineering and Technology

Panjab University, Chandigarh

# **INDEX**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S .No.** | **Experiment Name** | **Page No.** | **Date** | **Signature** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |



**PRAGYA UE228079**

**EXPERIMENT-1**

**AIM:** To generate Amplitude Modulated wave.

**CODE**:

clc;

clear all;

close all;

ac=2;

fc=0.5;

am= 1;

fm=0.05;

fs=150;

ka=1;

t=[0:0.1:50];

ct=ac\*cos(2\*pi\*fc\*t);

mt=am\*cos(2\*pi\*fm\*t);

am=ct.\*(1+ka\*mt);

subplot(3,1,1);

plot(mt);

ylabel('messege signal');

subplot(3,1,2);

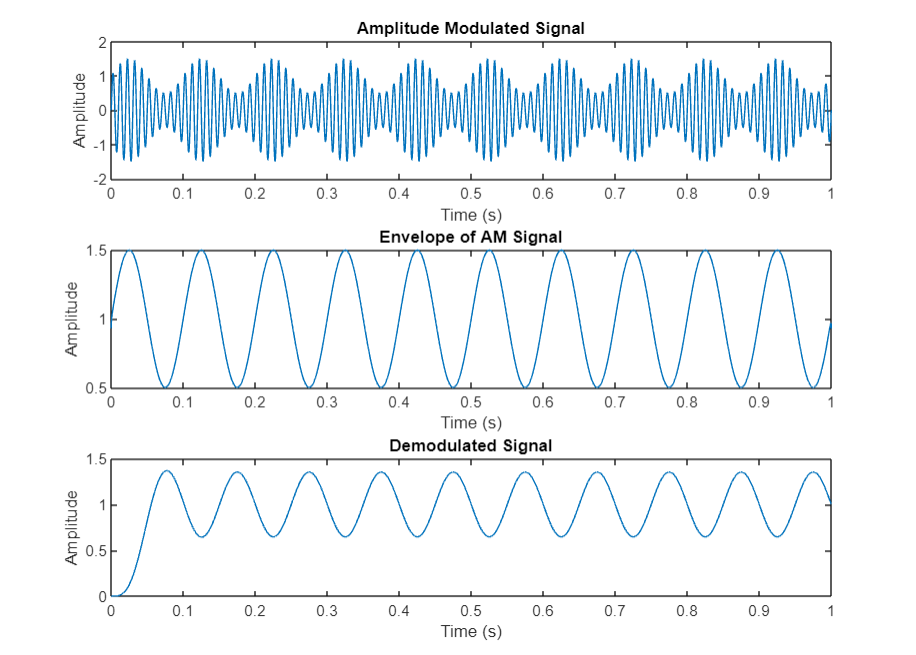
plot(ct);

ylabel('carrier');

subplot(3,1,3);

plot(am);

ylabel('am signal');



**PRAGYA UE228079**

**EXPERIMENT-2**

**AIM:** To detect Amplitude Modulated wave.

**CODE**:  
% Amplitude modulation parameters

fc = 100; % carrier frequency in Hz

fm = 10; % message frequency in Hz

Ac = 1; % carrier amplitude

Am = 0.5; % message amplitude

Fs = 10000; % sampling frequency in Hz

T = 1/Fs; % sampling period

% Generate modulated signal

t = 0:T:1;

carrier = Ac\*sin(2\*pi\*fc\*t); % carrier signal

message = Am\*sin(2\*pi\*fm\*t); % message signal

modulated = (1 + message).\*carrier; % AM signal

% Demodulate using envelope detection

envelope = abs(hilbert(modulated)); % take the absolute value of the Hilbert transform

[b, a] = butter(4, 2\*fm/Fs, 'low'); % design a low-pass filter

demodulated = filter(b, a, envelope); % apply the filter to the envelope

% Plot the results

subplot(3,1,1);

plot(t, modulated);

title('Amplitude Modulated Signal');

xlabel('Time (s)');

ylabel('Amplitude');

subplot(3,1,2);

plot(t, envelope);

title('Envelope of AM Signal');

xlabel('Time (s)');

ylabel('Amplitude');

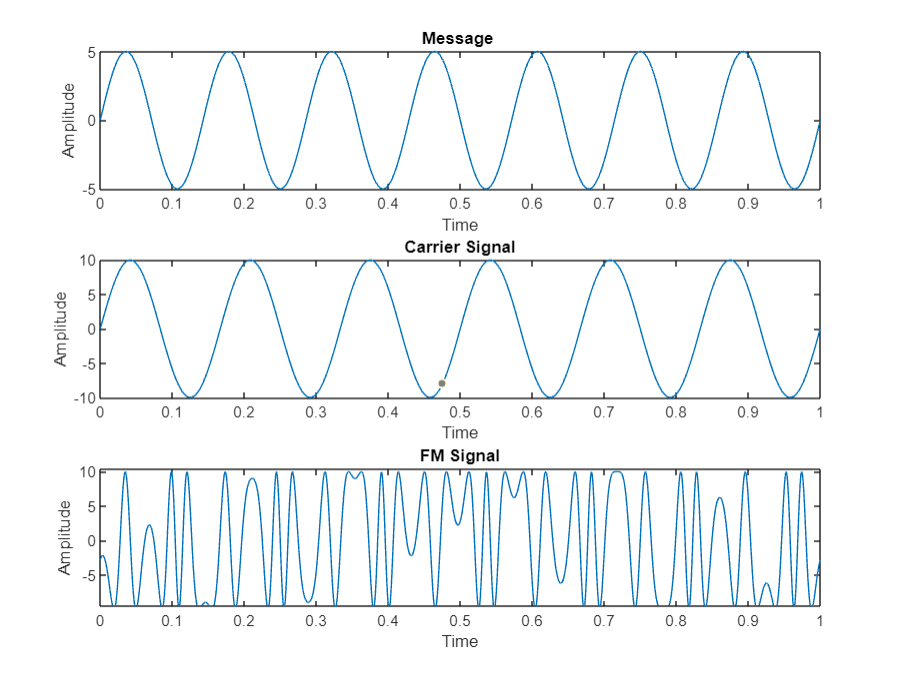
subplot(3,1,3);

plot(t, demodulated);

title('Demodulated Signal');

xlabel('Time (s)');

ylabel('Amplitude');

  
 **PRAGYA UE228079**

**EXPERIMENT-3**

**AIM:** To generate Frequency Modulated wave.

**CODE:**  
clc

clear all

close all

t = 0:0.001:1; %upto 1000 samples

vm = input('Enter Amplitude (Message) = ');

vc = input('Enter Amplitude (Carrier) = ');

fM = input('Enter Message frequency = ');

fc = input('Enter Carrier frequency = ');

m = input('Enter Modulation Index = ');

msg = vm\*sin(2\*pi\*fM\*t);

subplot(3,1,1); %plotting message signal

plot(t,msg);

xlabel('Time');

ylabel('Amplitude');

title('Message ');

carrier = vc\*sin(2\*pi\*fc\*t);

subplot(3,1,2); %plotting carrier signal

plot(t,carrier);

xlabel('Time');

ylabel('Amplitude');

title('Carrier Signal');

y = vc\*sin(2\*pi\*fc\*t+m.\*cos(2\*pi\*fM\*t));

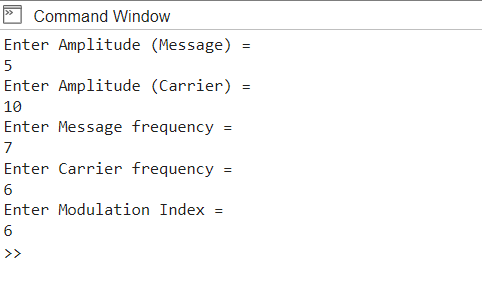
subplot(3,1,3);%plotting FM (Frequency Modulated) signal

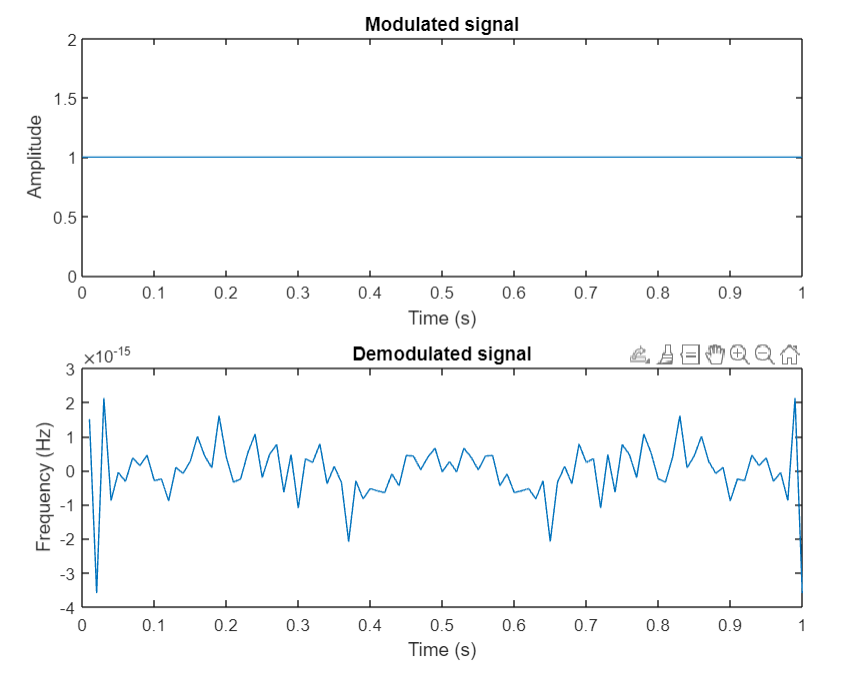
plot(t,y);

xlabel('Time');

ylabel('Amplitude');

title('FM Signal');





**PRAGYA UE228079**

**EXPERIMENT-4**

AIM: To detect Frequency Modulated wave.

CODE:

% Generate a signal with frequency modulation

fc = 100; % carrier frequency

fm = 500; % modulation frequency

fs = 100; % sampling frequency

t = 0:1/fs:1; % time vector

x = cos(2\*pi\*fc\*t + 2\*sin(2\*pi\*fm\*t)); % modulated signal

% Perform frequency demodulation

xr = real(hilbert(x)); % take the real part of the hilbert transform

phi = unwrap(angle(hilbert(x))); % calculate the instantaneous phase

fm\_hat = diff(phi)\*fs/(2\*pi); % differentiate the phase to get the modulating frequency

% Plot the results

subplot(2,1,1)

plot(t,x)

title('Modulated signal')

xlabel('Time (s)')

ylabel('Amplitude')

subplot(2,1,2)

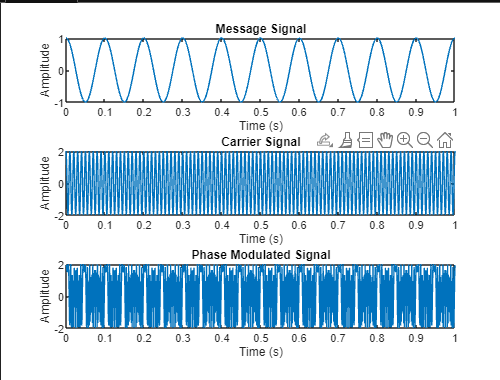
plot(t(2:end),fm\_hat)

title('Demodulated signal')

xlabel('Time (s)')

ylabel('Frequency (Hz)')

**OUTPUT:**



***Pragya Sharma***

**EXPERIMENT-5**

**AIM:** To generate Phase Modulated Wave

**CODE:**

% Phase Modulation

clear all;

clc;

% Parameters

fc = 100; % Carrier frequency

fm = 10; % Message frequency

fs = 1000; % Sampling frequency

kf = 50; % Frequency deviation constant

Am = 1; % Message amplitude

Ac = 2; % Carrier amplitude

% Message signal

t = 0:1/fs:1;

m = Am\*cos(2\*pi\*fm\*t);

% Carrier signal

c = Ac\*cos(2\*pi\*fc\*t);

% Phase modulation

pm = Ac\*cos(2\*pi\*fc\*t + kf\*m);

% Plotting

subplot(3,1,1);

plot(t,m);

xlabel('Time (s)');

ylabel('Amplitude');

title('Message Signal');

subplot(3,1,2);

plot(t,c);

xlabel('Time (s)');

ylabel('Amplitude');

title('Carrier Signal');

subplot(3,1,3);

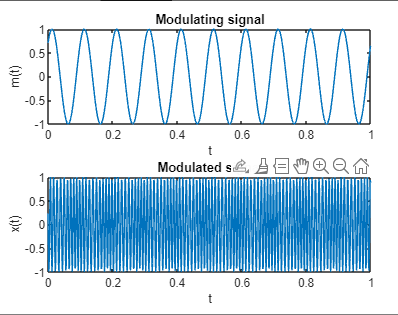
plot(t,pm);

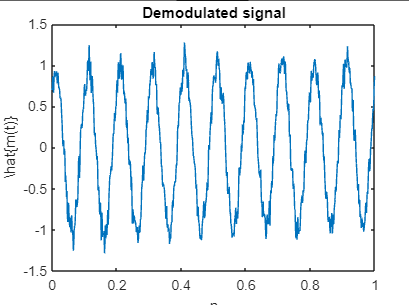
xlabel('Time (s)');

ylabel('Amplitude');

title('Phase Modulated Signal');

**OUTPUT:**



****

***Pragya Sharma***

**EXPERIMENT-6**

**AIM:** To detect Phase Modulated wave.

**CODE:**

clearvars; clc;

fc = 100; %carrier frequency

fm = 10; %frequency of modulating signal

alpha = 1; %amplitude of modulating signal

theta = pi/4; %phase offset of modulating signal

beta = pi/5; %constant carrier phase offset

receiverKnowsCarrier= 'False'; %If receiver knows the carrier frequency & phase offset

fs = 8\*fc; %sampling frequency

duration = 0.5; %duration of the signal

t = 0:1/fs:1-1/fs; %time base

m\_t = alpha\*sin(2\*pi\*fm\*t + theta); %modulating signal

x = cos(2\*pi\*fc\*t + beta + m\_t ); %modulated signal

figure(); subplot(2,1,1)

plot(t,m\_t) %plot modulating signal

title('Modulating signal'); xlabel('t'); ylabel('m(t)')

subplot(2,1,2)

plot(t,x) %plot modulated signal

title('Modulated signal'); xlabel('t');ylabel('x(t)')

nMean = 0; %noise mean

nSigma = 0.1; %noise sigma

n = nMean + nSigma\*randn(size(t)); %awgn noise

r = x + n; %noisy received signal

%Demodulation of the noisy Phase Modulated signal

z= hilbert(r); %form the analytical signal from the received vector

inst\_phase = unwrap(angle(z)); %instaneous phase

if strcmpi(receiverKnowsCarrier,'True')

offsetTerm = 2\*pi\*fc\*t+beta; %if carrier frequency & phase offset is known

else

p = polyfit(t,inst\_phase,1); %linearly fit the instaneous phase

estimated = polyval(p,t); %re-evaluate the offset term using the fitted values

offsetTerm = estimated;

end

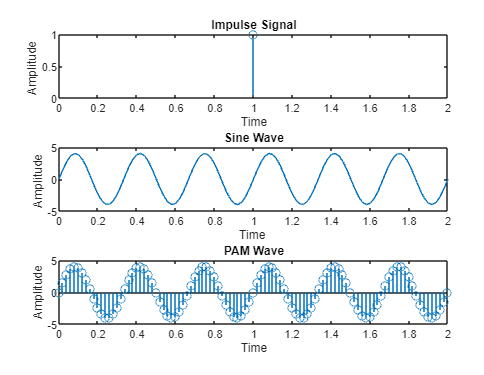
demodulated = inst\_phase - offsetTerm;

figure()

plot(t,demodulated); %demodulated signal

title('Demodulated signal'); xlabel('n'); ylabel('\hat{m(t)}');

**OUTPUT:**

****

***Pragya Sharma***

**EXPERIMENT-7**

**Aim:** To generate Pulse Amplitude Modulated Wave.

**CODE:**

clc;

close all;

clear all;

a = input('Enter the amplitude = ');

f = input('Enter the frequency = ');

t = 0:0.02:2; % for a total of 16 samples

x1 = 1; %generation of an impulse signal

x2 = a\*sin(2\*pi\*f\*t); %generation of sine wave

y = x1.\*x2; %modulation step

subplot(3,1,1); %for impulse signal plot

stem(x1);

title('Impulse Signal');

xlabel('Time');

ylabel('Amplitude ');

subplot(3,1,2) %for sine wave plot

plot(t,x2);

title('Sine Wave');

xlabel('Time ');

ylabel('Amplitude ');

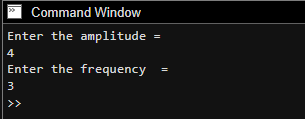
subplot(3,1,3) %for PAM wave plot

stem(t,y);

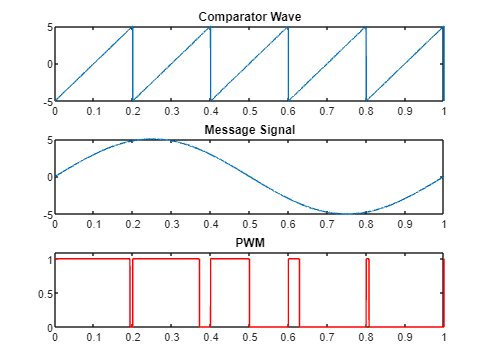
title('PAM Wave');

xlabel('Time');

ylabel('Amplitude');



**OUTPUT:**

****

***Pragya Sharma***

**EXPERIMENT-8**

**AIM:** To generate Pulse Width Modulated Wave.

**Code:**

fs=input('Comparator Sawtooth frequency:');

fm=input('Message frequency(Assuming it to be a sine wave):');

a=input('Enter Amplitude of Message:');

t=0:0.0001:1; %sampling rate of 10kHz

stooth=1.01\*a.\*sawtooth(2\*pi\*fs\*t); %generating a sawtooth wave

%to make the two non zero lobes of pwm not to overlap the amplitude of

%sawtooth wave must be atleast more than a bit to the message amplitude

subplot(3,1,1);

plot(t,stooth); % plotting the sawtooth wave

title('Comparator Wave');

msg=a.\*sin(2\*pi\*fm\*t); %generating message wave

subplot(3,1,2);

plot(t,msg); %plotting the sine message wave

title('Message Signal');

for i=1:length(stooth)

if (msg(i)>=stooth(i))

pwm(i)=1; %is message signal amplitude at i th sample is greater than

%sawtooth wave amplitude at i th sample

else

pwm(i)=0;

end

end

subplot(3,1,3);

plot(t,pwm,'r');

title('PWM');

axis([0 1 0 1.1]); %to keep the pwm visible during plotting.

