

AMAN RAJ (1804425) (ETC 6)

School of electronics engineering,  KIIT Deemed to be University

**Laboratory record Communication engineering lab (ec 2094) AUTUMN 2020**

**Index Page**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Experiment No. | Aim of the Experiment | Date of Experiment | Date of Submission | Faculty Remarks |
| 01 | Generation and detection of Amplitude Modulation and Demodulation, DSBSC and SSB SC modulation. | 03/08/20 | 17/08/20 |  |
| 02 | Study of Frequency modulation and Demodulation Techniques. | 10/08/20 | 17/08/20 |  |
| 03 | Generation and detection of PAM, PWM and PPM techniques | 24/08/20 | 31/08/20 |  |
| 04 | Study of Pulse Code Modulation (PCM) and demodulation. Multiplexing of signal using Time Division Multiplexing (TDM) technique. |  |  |  |
| 05 | Generation and detection of Delta modulation Technique |  |  |  |
| 06 | (i) Study of different Data formatting techniques.  (ii)Generation and Detection of Amplitude Shift Keying (ASK) |  |  |  |
| 07 | (i) Generation and Detection of Frequency Shift Keying (FSK).  (ii)Generation and Detection of Binary Phase Shift Keying (BPSK) |  |  |  |
| 08 | Generation and Detection of Quadrature Phase Shift Keying (QPSK). |  |  |  |
| 09 | Open Ended Experiment-1 |  |  |  |
| 10 | Open Ended Experiment-2 |  |  |  |

|  |  |
| --- | --- |
| **Experiment Number** | 01 |
| **Date of Experiment** | 03/08/2020 |
| **Date of Submission** | 17/08/2020 |
| **Name of the student** | AMAN RAJ |
| **Roll Number** | 1804425 |
| **Section** | ETC 6 |

**Aim of The Experiment :-**

Generation and detection of Amplitude Modulation and Demodulation, DSBSC and SSB SC modulation.

**Equipment / Software Required:-**

The Equipment and Software required are as follows:

( GNU OCTAVE GUI )

**Theory**

**Amplitude modulation:** AM is a modulating technique which is generally used to transmit Radio signals over a carrier signal, where the amplitude of the modulated changes in response of the signal.

**Modulating Index(μ)**, also known as modulation depth, of a modulation scheme describes by how much the modulated variable of the carrier signal varies around its unmodulated level. It is defined differently in each modulation scheme.

**DSBSC:** The transmission of a signal, which contains a carrier along with two sidebands can be termed as Double sideband Full Carrier system or simply DSBSC.

**SSBSC:** The process of suppressing one of the sidebands along with the carrier and transmitting a single sideband is called as Single sideband Suppressed Carrier system or simply SSBSC.

**Code:-**

Detail Code

**<<< This code will amplitude modulate and then demodulate a sine wave>>>**

**%generating user defined AM Signals**

clc;

clear all;

close all;

Load pkg signal

t = linspace(0,1,1000) %Time of 1 secs divided by 1000 times

%Carrier wave

fc = input('Enter fc = ')

ac = input('Enter ac = ')

xc= cos(2\*pi\*fc\*t)

%Message signal

fm = input('Enter fm = ');

am = input('enter am = ');

xm = cos(2\*pi\*fm\*t);

**%Amplitude modulation**

y = [ac + am\*xm].\*xc;

%plot AM

subplot(4,1,1)

plot(t,xc);

xlabel("Time -->")

ylabel("Amplitude -->")

title("Carrier Wave")

subplot(4,1,2)

plot(t,xm)

xlabel("Time -->")

ylabel("Amplitude -->")

title("Message Wave")

subplot(4,1,3)

plot(t,y)

xlabel("Time -->")

ylabel("Amplitude -->")

title("Modulated Wave")

%if else statement

mu = am/ac;

if mu==1

disp('Critical modulation');

elseif mu>1

disp('Over modulated signal');

elseif mu<1

disp('under modulated signal');

end

**%Demodutating The wave**

dm = y.^2;

[b,a] = butter(10,0.1);

xd = filter(b,a,dm);

subplot(4,1,4)

plot(t,xd);

xlabel("Time -->")

ylabel("amplitude")

title("Demodulated Wave")

**<<<This code will generate a DSBCS Modulated signal>>>**

**%DSBCS Generation**

clc;

clear all;

close all;

t = linspace(0,4,1000);

fc = input('Enter the carrier frequency: ')

ac = input('Enter the carrier amplitude: ')

fm = input('Enter the message frequency: ')

am = input('Enter the message amplitude: ')

y = am\*cos(2\*pi\*fm\*t) **%message signal**

z = ac\*cos(2\*pi\*fc\*t) **%carrier signal**

w = ((am\*ac)/2).\*(cos(2\*pi\*(fc+fm)\*t)+cos(2\*pi\*(fc-fm)\*t)) %DSBSC Modulation

subplot(3,1,1)

plot(t,z)

xlabel("time -->")

ylabel("magnitude -->")

title("Carrier Signal")

subplot(3,1,2)

plot(t,y)

xlabel("time -->")

ylabel("magnitude -->")

title("Message Signal")

subplot(3,1,3)

plot(t,w)

xlabel("time -->")

ylabel("magnitude -->")

title("DSBSC Signal")

**<<< Generates an upper sideband, a lower side band SSBSC>>>**

**% Generation of SSB-SC Signal**

clear all

close all

clc

fc = input('Enter the frequency of Carrier: ')

ac = input('Enter the amplitude of Carrier: ')

fm = input('Enter the frequency of Message: ')

am = input('Enter the amplitude of Message: ')

t = linspace(0,1,1000)

m = am\*cos(2\*pi\*fm\*t) **%Message signal**

c = ac\*cos(2\*pi\*fc\*t) **%carrier Signal**

Susb = ((am\*ac)/2).\*cos(2\*pi\*(fc+fm)\*t) %upper Sideband

Slsb = ((am\*ac)/2).\*cos(2\*pi\*(fc-fm)\*t) %lower Sideband

subplot(4,1,1)

plot(t,c,'r')

xlabel("Time -->")

ylabel("Amplitude-->")

title("Carrier Wave")

subplot(4,1,2)

plot(t,m,'g')

xlabel("Time -->")

ylabel("Amplitude-->")

title("Message Wave")

subplot(4,1,3)

plot(t,Susb,'b')

xlabel("Time -->")

ylabel("Amplitude-->")

title("Upper Sideband SSB-SC Signal")

subplot(4,1,4)

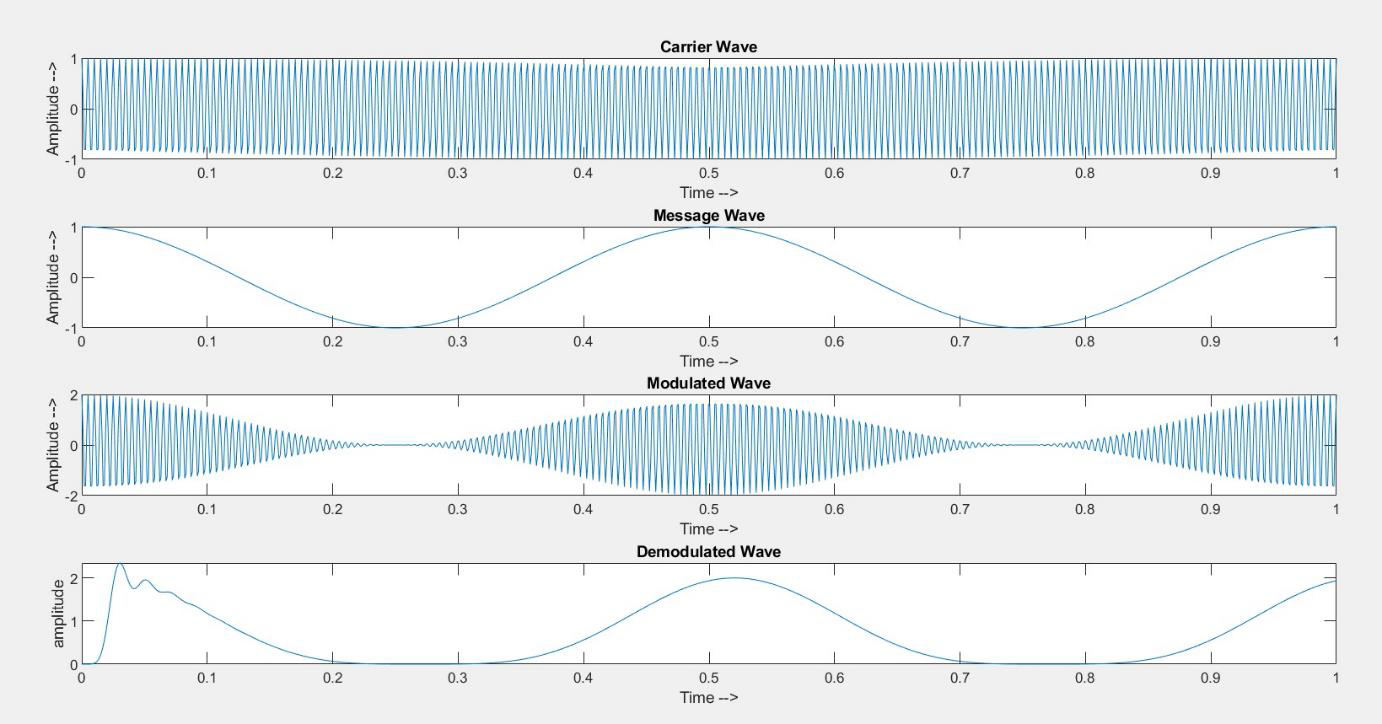
plot(t,Slsb,'k')

xlabel("Time -->")

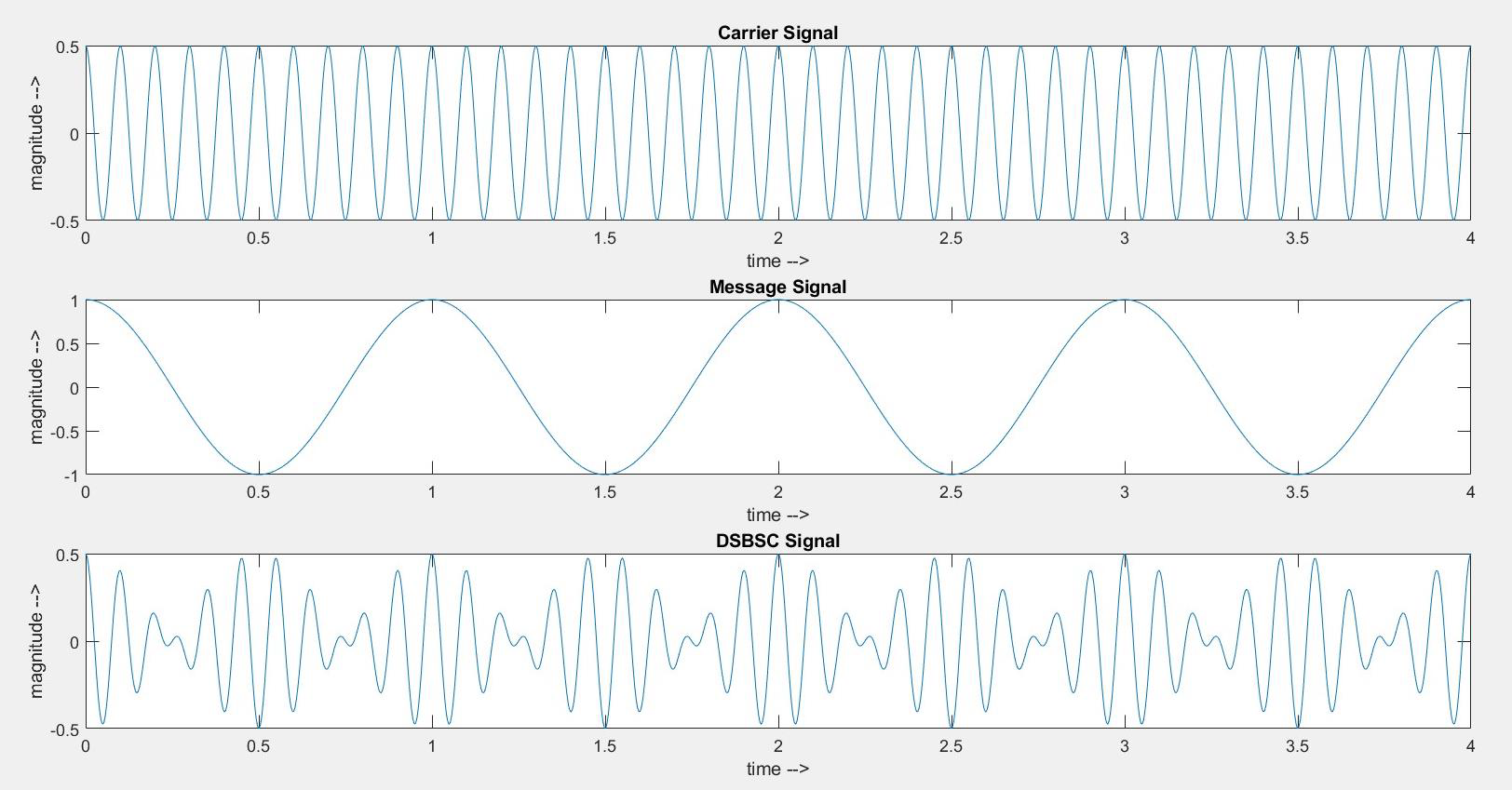
ylabel("Amplitude-->")

title("Lower Sideband SSB-SC Signal")

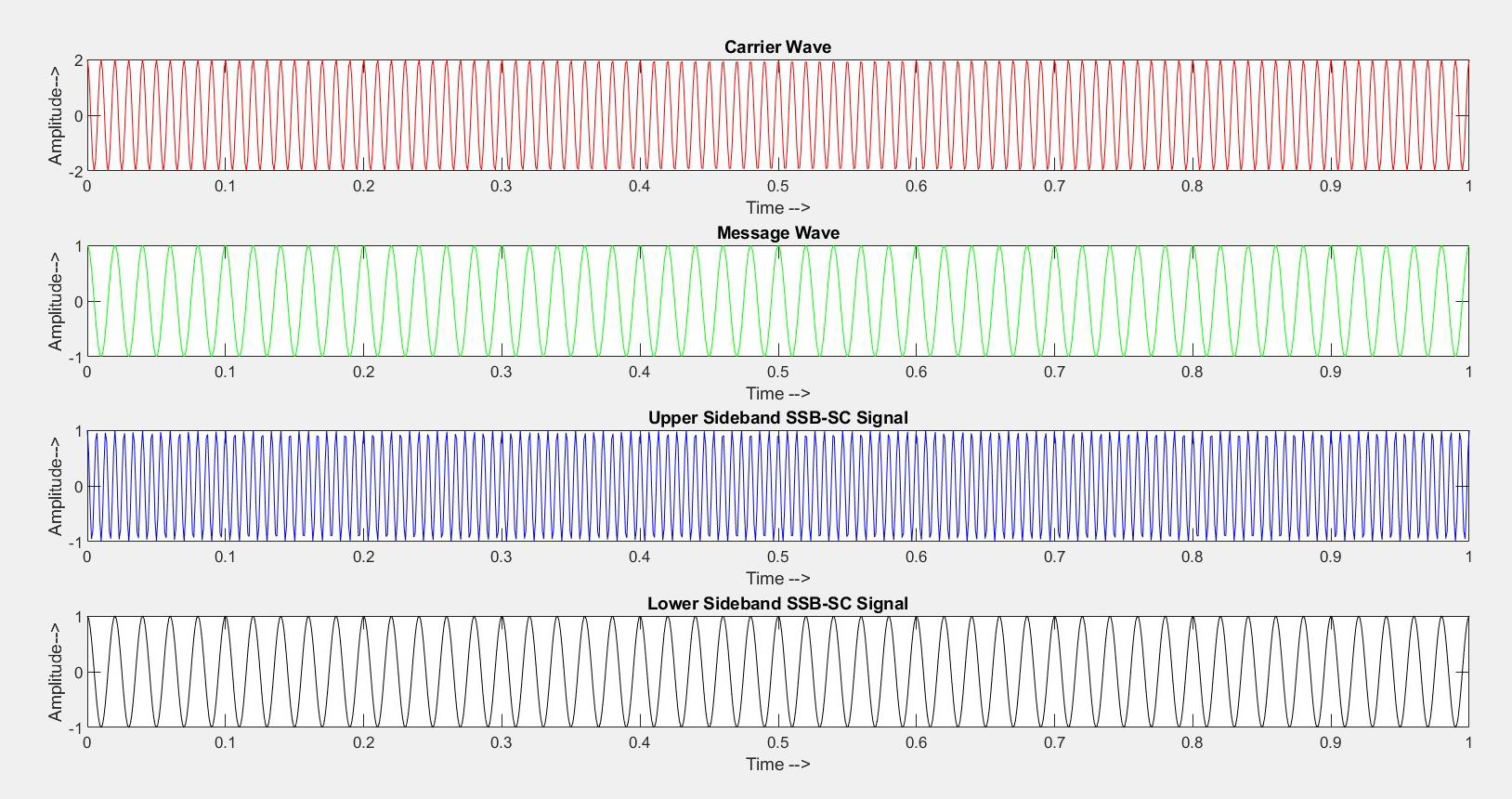
**Output/Graph:-**



**Fig 1 : Amplitude modulating and Demodulating a Sine wave of 2Hz**



**Fig 2: Generation of DSBSC Signal**



**Fig 3: Generation of SSBSC Signal**

**Discussion or Inference of the experiment**

**Discussion or Inference of the experiment**

Through this experiment we have learn about different kind of amplitude modulation and demodulation techniques that is practiced in analog radio communication technology. It helped me to visualize the difference between the different techniques of transmission.

**Conclusion:-**

The simulation of experiment is done successfully using GNU OCTAVE Software.

|  |  |
| --- | --- |
| **Experiment Number** | 02 |
| **Date of Experiment** | 10/08/2020 |
| **Date of Submission** | 17/08/2020 |
| **Name of the student** | AMAN RAJ |
| **Roll Number** | 1804425 |
| **Section** | ETC 6 |

**Aim of The Experiment :-**

Study of Frequency modulation and Demodulation Techniques.

**Equipment / Software Required:-**

GNU OCTAVE GUI

**Theory**

Frequency Modulation (FM) is a form of modulation in which changes in the carrier wave frequency correspond directly to changes in the baseband signal.

**Code:-**

**<<< This code generates a FM signal and Demodulates it>>**

**%Generating a FM Signal and Demodulating it**

clc;

clear all;

close all;

fc=input('Enter the carrier signal: ');

fm=input('Enter the message signal: ');

mu=input('Modulation index ');

t=linspace(0,1,1000);

c=cos(2\*pi\*fc\*t);**%carrier signal**

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

subplot(4,1,1);

plot(t,c,'r'); **%plotting the carrier signal**

ylabel('amplitude');

xlabel('time');

title('Carrier signal');

subplot(4,1,2);

plot(t,m,'g'); **%plotting the message signal**

ylabel('amplitude');

xlabel('time');

title('Message signal');

y=cos(2\*pi\*fc\*t-(mu\*cos(2\*pi\*fm\*t))); **% FM Generation**

subplot(4,1,3);

plot(t,y,'b'); **% Plotting the FM Generation**

ylabel('amplitude');

xlabel('time');

title('Frequency Modulated signal');

**%FM Demodulation**

dem = diff(y);

dem = [0,dem];

rect\_dem = abs(dem)

[b,a]=butter(10,0.06);

rec = filter(b,a,rect\_dem);

subplot(4,1,4)

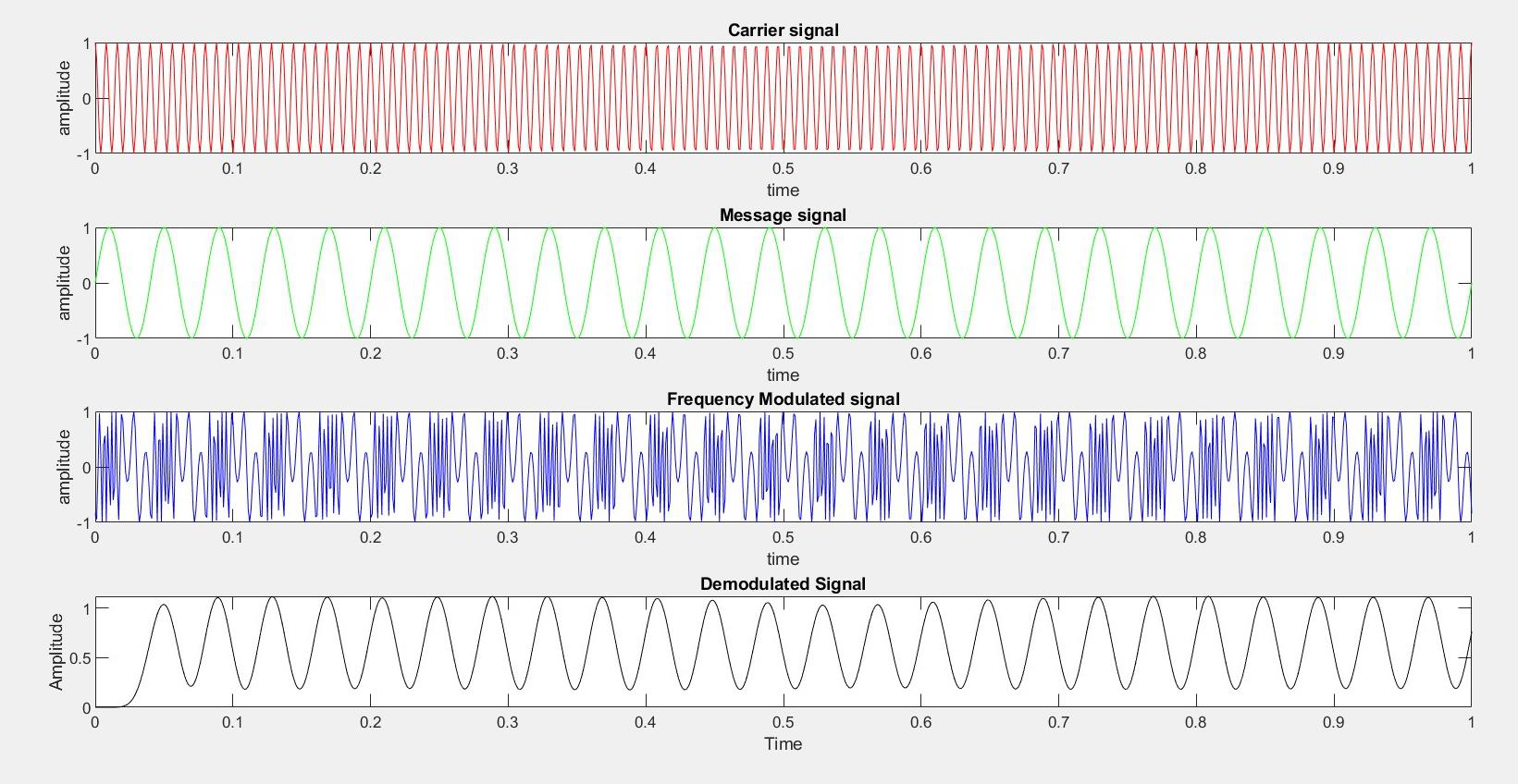
plot(t,rec,'k')

xlabel('Time')

ylabel('Amplitude')

title('Demodulated Signal')

**Output/Graph:-**



**Fig 1: Generation of FM Signal and demodulating it.**

**Discussion or Inference of the experiment**

This experiment we have learn techniques of radio transmission that are currently being practiced. It also let me know the reasons that FM is better than AM in radio signal transmissions. I also learn how to demodulate a FM signal.

**Conclusion:-**

Simulation of experiment is done successfully using GNU OCTAVE Software.

|  |  |
| --- | --- |
| **Experiment Number** | 03 |
| **Date of Experiment** | 24/08/2020 |
| **Date of Submission** | 31/08/2020 |
| **Name of the student** | AMAN RAJ |
| **Roll Number** | 1804425 |
| **Section** | ETC 6 |

**Aim of The Experiment :-**

Generation and detection of PAM, PWM and PPM techniques

**Equipment / Software Required:-**

GNU OCTAVE GUI

**Theory :-**

**PAM signal**: It is a form of signal modulation where the message information is encoded in

the amplitude of a series of signal pulses. It is an analog pulse modulation scheme in which

the amplitudes of a train of carrier pulses are varied according to the sample value of the

message signal. Demodulation is performed by detecting the amplitude level of the carrier at

every single period.

Pulse-amplitude modulation is widely used in modulating signal transmission of digital data,

with non-baseband applications having been largely replaced by pulse-code modulation, and,

more recently, by pulse-position modulation.

In particular, all telephone modems faster than 300 bit/s use quadrature amplitude modulation

(QAM). (QAM uses a two-dimensional constellation).

**PWM signa**l: A Pulse Width Modulation (PWM) Signal is a method for generating an analog signal using a digital source. A PWM signal consists of two main components that define its behaviour a duty cycle and a frequency. The duty cycle describes the amount of time the signal is in a high (on) state as a percentage of the total time of it takes to complete one cycle.

The frequency determines how fast the PWM completes a cycle (i.e. 1000 Hz would be 1000

cycles per second), and therefore how fast it switches between high and low states. By

cycling a digital signal off and on at a fast enough rate, and with a certain duty cycle, the

output will appear to behave like a constant voltage analog signal when providing power to

devices.

**PPM signal :-** Pulse-position modulation is a form of signal modulation in which M message bits are encoded by transmitting a single pulse in one of possible required time shifts.

This is repeated every T seconds, such that the transmitted bit rate is bits per second.

CODE:-

PAM MODULATION

**%PULSE AMPLITUDE MODULATION PAM**

clc;

close all;

clear all;

pkg load signal

t=0:0.0001:2;

f=5;

x=sin(2\*pi\*f\*t);

Ts=0.02;

**%PULSE GENERATION**

for k=1:length(t)

if mod(t(1,k),Ts)==0

sa(1,k)=1; %PULSE

else sa(1,k)=0;

end

end

ypam=x.\*sa;

subplot(5,1,1);

plot(t,x);

title('message')

subplot(5,1,2);

plot(t,sa);

title('switching signal');

subplot(5,1,3);

plot(t,ypam);

title('NaturalPAM');

**%FLAT TOP PAM**

ypamft=zeros(1,length(t));

k=1;

while k <length(t)

if ypam(1,k)~=0

ypamft(1,k:k+49)=ypam(1,k)\*ones(1,50); % pulse duration is 50\*0.001

k=k+49;

else ypamft(1,k)=0;

k=k+1;

end

end

subplot(5,1,4);

plot(t,ypamft);

title('Flat top FAM')

**%Demodulation**

[b,a]=butter(7,0.005);

demod=filter(b,a,ypamft);

subplot(5,1,5);

plot(t,demod);

title('Demodulation of PAM');

**PWM MODULATION :**

clc;

clear all;

close all;

pkg load signal

t=0:0.001:1;

s=1\*sawtooth(2\*pi\*50\*t);

m=1\*sin(2\*pi\*1\*t);

subplot(4,1,1);

plot(t,s);

title('sawtooth signal');

subplot(4,1,2);

plot(t,m);

title('sine wave');

n=length(s);

for i=1:n

if (m(i)>=s(i))

pwm(i)=1;

elseif (m(i)<=s(i))

pwm(i)=0;

end

end

subplot(4,1,3);

plot(t,pwm);

ylabel('Amplitude');

xlabel('Time index');

title('PWM Wave');

axis([0 1 0 2]);

%Demodulation

[b,a]=butter(10,0.05);

demod=filter(b,a,pwm);

subplot(4,1,4);

plot(t,demod);

title('Demodulation of PWM');

**PPM MODULATION AND DEMODULATION, GENERATION THROUGH PWM WAVE**

clear all

pkg load signal;

fs=400;

t=0:1/fs:1;

x1=20\*sawtooth(2\*pi\*20\*t);

figure;subplot(6,1,1);plot(t,x1);title('sawtooth sihnal');

m=10\*sin(2\*pi\*2\*t);subplot(6,1,2);plot(t,m,t,x1,'r');title('message signal');

pwm=(m>x1);

subplot(6,1,3);plot(t,pwm,'r');title('PWM Signal');

ipwm=~pwm;

dipwm=diff(ipwm);n=0:1/fs:0.999;

subplot(6,1,4),plot(n,dipwm);title('Signal after inverter and differentiator');

l=length(dipwm);ppm=zeros(1,l);

for i =1:(l-1)

if dipwm(1,i)>0

ppm(1,i)=1;

end

end

subplot(6,1,5);plot(n,ppm);title('PPM Signal')

[b,a]=butter(5,0.02);

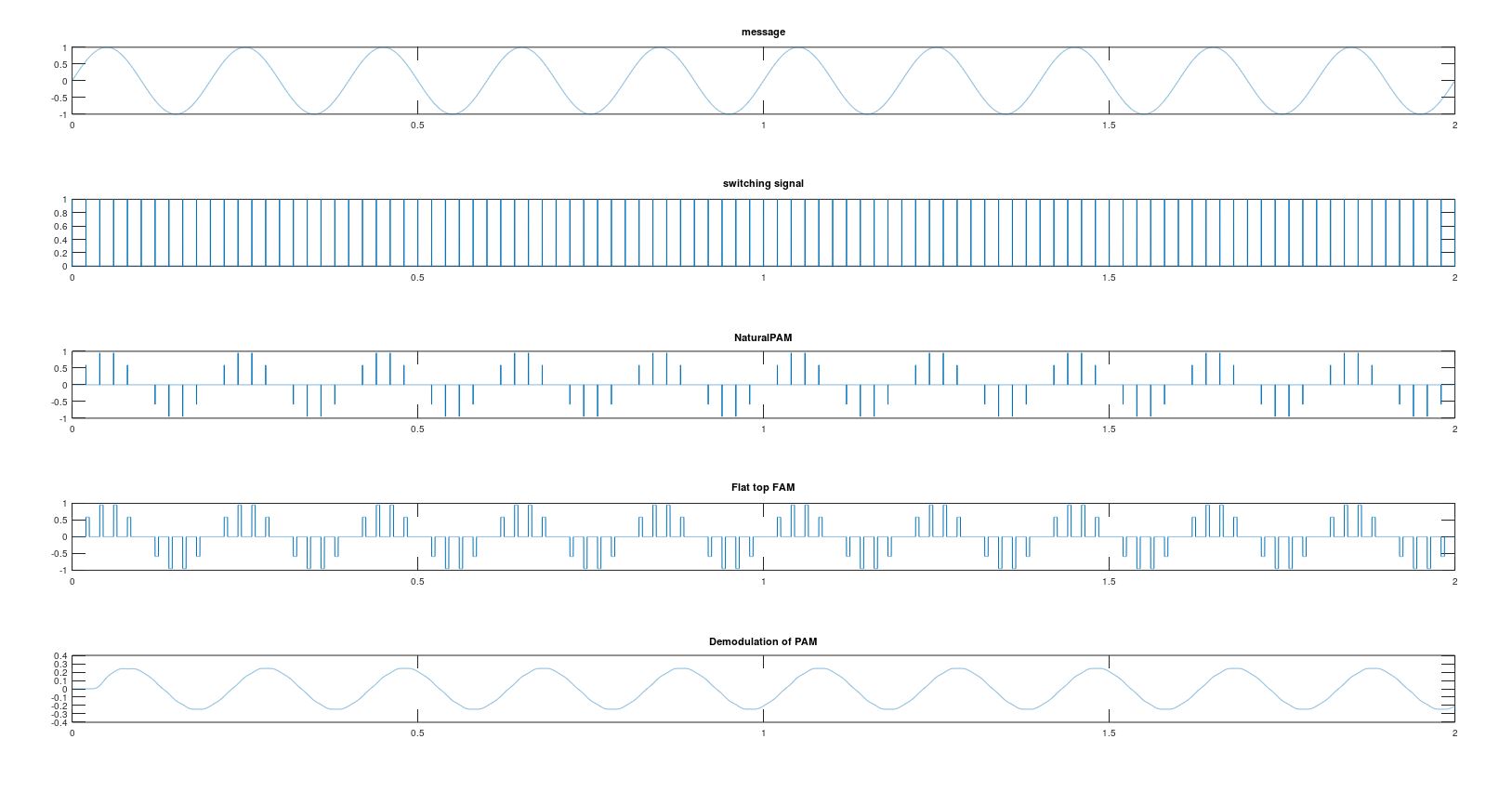
s1=filter(b,a,pwm);

subplot(6,1,6);

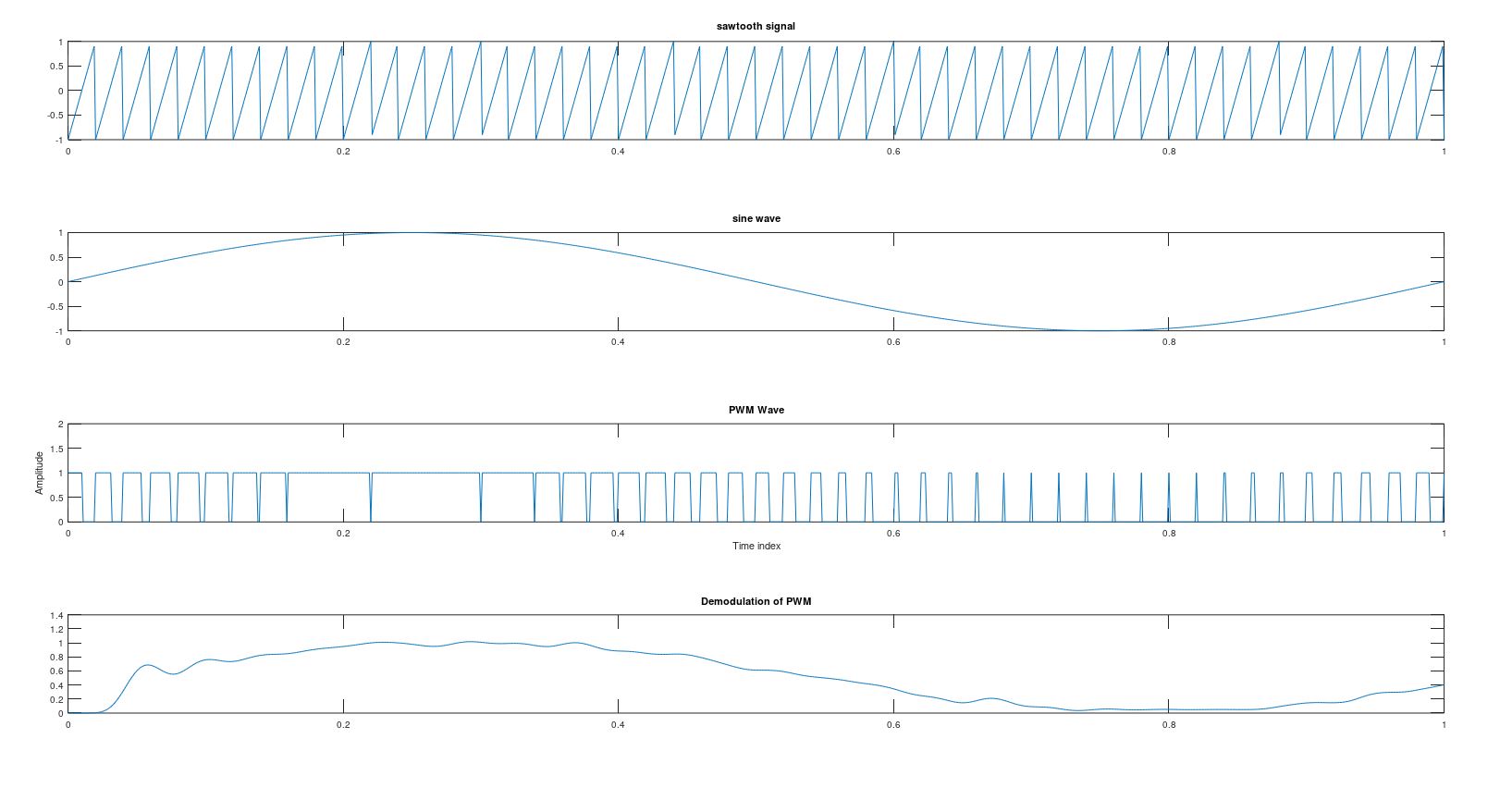
plot(t,s1);

title('Demodulated message signal');

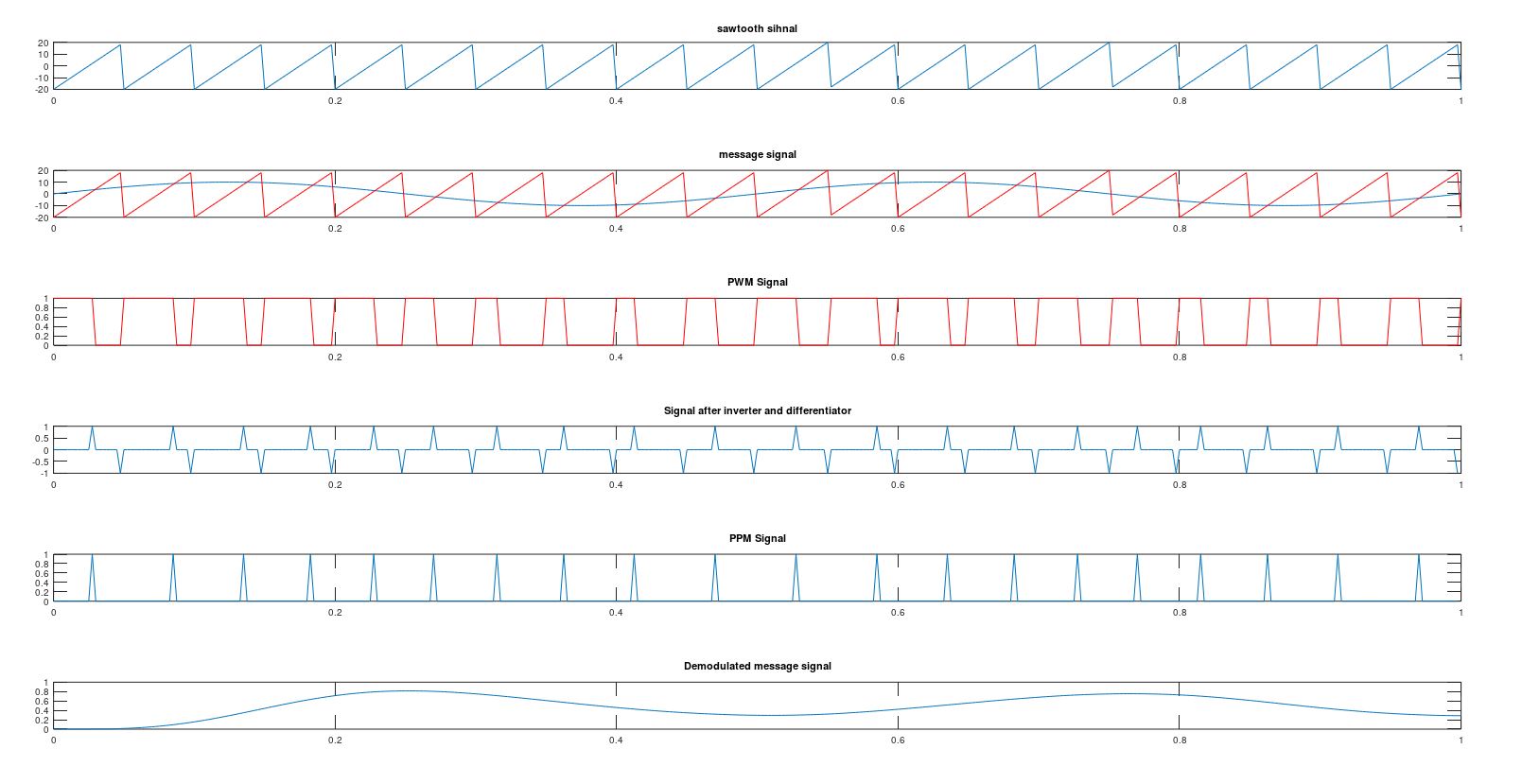
**OUTPUT/GRAPH:-**

****

***Fig: PAM modulation and demodulation***

****

***Fig: PWM modulation and demodulation***

****

***Fig: PPM modulation***

**Discussion of the experiment:-**

Through this experiment we got to learn about different pulse modulation techniques such as PAM, PWM, PPM. I also observed advantages of Pulse Position Modulation (PPM), over PWM and PAM.

By understanding, Pulse position modulation has low noise interference when compared

to PAM because amplitude and width of the pulses are made constant during modulation.

Noise removal and separation is also very easy in PPM.

I learned about differentiator and inverter circuit, and how it is used for generation of PPM

signal, through PWM signal.

**Conclusion:-**

The simulation of PAM, PWM and PAM was done successfully using OCTAVE GNU Software.