**EXPERIMENT – 1**

**Aim:**

To study Sampling Theorem

**Software Used**: MATLAB R2013a

**Program**

clc;

close all;

clear all;

t=0:0.0005:0.02;

T=0.04;

fm=1/T;

n1=0:40;

size(n1)

x\_t=sin(2\*pi\*2\*fm\*t);

subplot(2,2,1)

plot(200\*t,x\_t);

xlabel('t');

ylabel('x(t)');

title('continuous time signal');

fs1=20\*fm;%>niq rate

fs2=4\*fm;% =niq rate

fs3=0.4\*fm;%<niq rate

n=0:20;

xn1=sin(2\*pi\*n\*fm/fs1);

subplot(2,2,2)

stem(n,xn1);

xlabel('time in seconds');

ylabel('x(n)');

title('sample signal with fs>2fm');

n=0:4;

xn1=sin(2\*pi\*n\*fm/fs2);

subplot(2,2,3)

stem(n,xn1);

xlabel('time in seconds');

ylabel('x(n)');

title('sample signal with fs=2fm');

n=0:10;

xn3=sin(2\*pi\*n\*fm/fs3);

subplot(2,2,4)

stem(n,xn3);

xlabel('time in seconds');

ylabel('x(n)');

title('sample signal with fs<2fm');

 OUTPUT WAVEFORM

**EXPERIMENT – 2**

**Aim:**

To calculate S/N ratio and Probability of error of Differential Pulse Code Modulation

**Software Used:**MATLAB R2013a

**Program:**

% S/N ratio

clc;

close all;

clear all;

p=input('enter the power');

v=input('enter the no of bits');

xm=input('enter the value of max level');

snr=3\*p\*(2^(2\*v))/(xm^2);

% Probability of error

ebno=10:0.1:20;

y=sqrt(ebno);

pe=0.5\*erfc(0.5\*y);

loglog(ebno,pe);

xlabel('E/No')

ylabel('POE')

title('Error probability as a function of x')

OUTPUT

**S/N ratio**

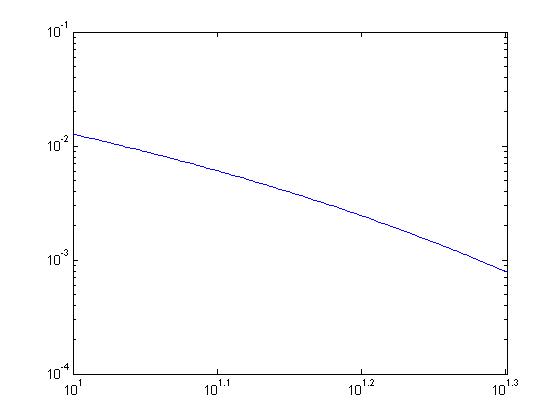
enter the power5

enter the no of bits6

enter the value of max level8

snr =

>>960

**Probability of Error**

EXPERIMENT - 3

Aim :

To Study all line encoding technique.

Software Used: MATLAB R2013a

Program:

clc;

close all;

clear all;

x=round(rand(1,10));

nx=length(x);

sign=1;

i=1;

while i<nx+1

t=i:0.001:i+1-0.001;

if x(i)==1

unipolar\_nrz=square(t\*2\*pi,100);

bipolar\_nrz=square(t\*2\*pi,100);

ami\_nrz=sign\*square(t\*2\*pi,100);

unipolar\_rz=(1+square(t\*2\*pi,50))/2;

polar\_rz=(1+square(t\*2\*pi,50))/2;

ami\_rz=sign\*(1+square(t\*2\*pi,50))/2;

sign=sign\*-1;

manchester\_code=-square(t\*2\*pi,50);

else

unipolar\_nrz=0;

bipolar\_nrz=-square(t\*2\*pi,100);

ami\_nrz=0;

unipolar\_rz=0;

polar\_rz=-(1+square(t\*2\*pi,50))/2;

ami\_rz=0;

manchester\_code=square(t\*2\*pi,50);

end

subplot(4,2,1);

plot(t,unipolar\_nrz);

ylabel('unipolar\_nrz');

hold on;

grid on;

axis([1 10 -2 2]);

subplot(4,2,3);

plot(t,bipolar\_nrz);

ylabel('bipolar\_nrz');

hold on;

grid on;

axis([1 10 -2 2]);

subplot(4,2,5);

plot(t,ami\_nrz);

ylabel('ami\_nrz');

hold on;

grid on;

axis ([1 10 -2 2]);

subplot(4,2,2);

plot(t,unipolar\_rz);

ylabel('unipolar\_rz');

hold on;

grid on;

axis ([1 10 -2 2]);

subplot(4,2,4);

plot(t,polar\_rz);

ylabel('polar\_rz');

hold on;

grid on;

axis ([1 10 -2 2]);

subplot(4,2,6);

plot(t,ami\_rz);

ylabel('ami\_rz');

hold on;

grid on;

axis ([1 10 -2 2]);

subplot(4,2,7);

plot(t,manchester\_code);

ylabel('manchester\_code');

hold on;

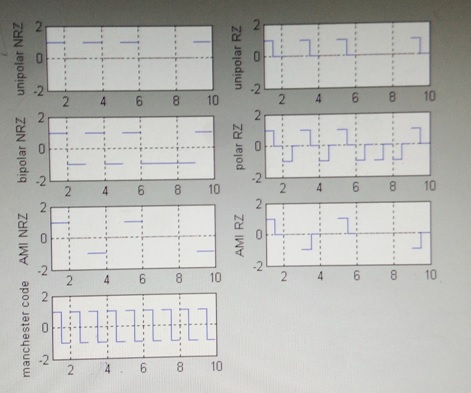
grid on;

axis ([1 10 -2 2]);

i=i+1;

end

OUTPUT WAVEFORM



**EXPERIMENT - 5**

**Aim:**

To generate quantized PCM signal using Matlab Code

**Software Used**: MATLAB R2013a

**Program**

clc;

close all;

clear all;

t=0:0.0001:20 %sample at nyquist rate

c=input('Enter n-bit value:');

part=-1:0.1:1;

codebook=-1:0.1:1.1;

msg=cos(t);

[~,quants]=quantiz(msg,part,codebook);

subplot(3,1,1);

plot(t,msg);

title('message signal');

subplot(3,1,2);

plot(t,quants);

title('sampled signal');

y=uencode(quants,c);

ybin=dec2bin(y,c)

subplot(3,1,3);

plot(t,y);

title('quantized PCM signal');

OUTPUT

Enter n-bit value:2



**EXPERIMENT - 4**

**Aim**:

To study S/N ratio of Delta modulation

**Software Used**: MATLAB R2013a

**Program**

clc;

clear all;

close all;

fM=input('Enter value of fM');

fm=input('Enter value of fm');

fs=input('Enter value of fs');

SNR=(3\*fs^3)/(8\*pi^2\*fm^2\*fM)

OUTPUT

Enter value of fM400

Enter value of fm250

Enter value of fs500

SNR =

0.1900

**EXPERIMENT - 6**

**AIM:** To implement and analyse Amplitude phase shift keying waveform

**PROGRAM:**

clc;

clear all;

close all;

%Generated Carrier Signal

Tb=1; fc=10;

t3=0:Tb/100:8;

t=0:Tb/100:1;

c1=sqrt(2/Tb)\*sin(2\*pi\*fc\*t3);

c=sqrt(2/Tb)\*sin(2\*pi\*fc\*t);

%Generated 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=zeros(1,length(t));

end

message(i,:)=m\_s;

%product of carrier and message

ask\_sig(i,:)= c.\*m\_s

t1=t1+(Tb);

t2=t2+(Tb);

%plot the message and ASK signal

subplot(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('ASK signal');

xlabel('t---->');

ylabel('s(t)');

grid on;

hold on;

end

hold off;

%plot the carrier signal and input binary data

subplot(5,1,3);

plot(t3,c1);

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]

%correlator

x=sum(c.\*ask\_sig(i,:));

%descision Device

if x>0

demod(i)=1;

else

demod(i)=0;

end

t1=t1+(Tb);

t2=t2+(Tb);

end

%plot demodulated binary data bits

subplot(5,1,5);

stem(demod);

title('Binary data bits');

xlabel('n---->');

ylabel('b(n)');

grid on

OUTPUT

