PRINCIPLES OF COMMUNICATION PRACTICALS

Question 1 To generate AM, PM and FM signal using matlab/scilab software, Analyze the effect of increasing modulation index for each modulation technique.  
  
Answer Following files have been created using scilab software

* Amplitude Modulation

Am=5;

Ac=10;

m=Am/Ac;

fc=50;

fm=10;

t=0:0.001:.5;

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

subplot(2,2,1);

plot(t,Sm);

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

subplot(2,2,2);

plot(t,Sc);

Sam=(1+(m\*cos(2\*%pi\*fm\*t)));

Sam1=Ac\*Sam;

Sam2=cos(2\*%pi\*fc\*t).\*Sam1;

subplot(2,2,3);

plot(t,Sam2);

* Frequency Modulation

clear all;

t=0:0.001:1;

A=5;

fc=50;

fm=5;

m=0.5;

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

y=A\*sin(2\*%pi\*fc\*t);

Zfm=A\*cos(2\*%pi\*fc\*t+m\*sin(2\*%pi\*fm\*t));

subplot(3,1,1);

xlabel('Time');

ylabel('Amplitude');

title('Modulating Signal');

plot(t,x);

subplot(3,1,2);

xlabel('Time');

ylabel('Amplitude');

title('Carrier Signal');

plot(t,y);

subplot(3,1,3);

xlabel('Time');

ylabel('Amplitude');

title('Modulated Signal');

plot(t,Zfm);

* Phase Modulation

clf;

t=0:0.001:1;

A=5;

fc=50;

fm=5;

m=0.5;

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

subplot(3,1,1);

xlabel('Time');

ylabel('Amplitude');

title('Modulating Signal');

plot(t,x);

y=A\*sin(2\*%pi\*fc\*t);

subplot(3,1,2);

xlabel('Time');

ylabel('Amplitude');

title('Carrier Signal');

plot(t,y);

Zfm=A\*cos(2\*%pi\*fc\*t+m\*cos(2\*%pi\*fm\*t));

subplot(3,1,3);

xlabel('Time');

ylabel('Amplitude');

title('Modulated Signal');

plot(t,Zfm);

Question 2 To demodulate AM signal using RC low pass filter/Hilbert Transform

Fc = 20; % carrier frequency of 20 kHz

Fs = 160; % sampling rate of 160 samples per ms

Fm = 0.4; % modulating frequency of 0.4 kHz

t = 0:1/Fs:10; % t of 10 seconds

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

m = cos(2\*pi\*Fm\*t); % modulating signal

s = c + 0.25\*cos(2\*pi\*(Fc+Fm)\*t) + 0.25\*cos(2\*pi\*(Fc-Fm)\*t); % FCAM 50% mod

dsb\_sc = ammod(m,Fc,Fs); % DSB-SC signal

figure;

subplot(4,1,1);

plot(t, m);

title('Modulating Signal'); xlabel('time (s)'); ylabel('amplitude');

subplot(4,1,2);

plot(t, c);

title('Carrier Signal'); xlabel('time (s)'); ylabel('amplitude');

subplot(4,1,3);

plot(t, s);

title('FCAM Signal'); xlabel('time (s)'); ylabel('amplitude');

subplot(4,1,4);

plot(t, dsb\_sc);

title('DSB-SC Signal'); xlabel('time (s)'); ylabel('amplitude');

% Non-Coherent Detection Step 1: Envelope Detection

Vc(1) = 0; % initial capacitor voltage

for i = 2:length(s)

if s(i) > Vc(i-1) % diode on (charging)

Vc(i) = s(i);

else % diode off (discharging)

Vc(i) = Vc(i-1) - 0.023\*Vc(i-1);

end

end

Vd(1) = 0;

for i = 2:length(dsb\_sc)

if dsb\_sc(i) > Vd(i-1) % diode on (charging)

Vd(i) = dsb\_sc(i);

else % diode off (discharging)

Vd(i) = Vd(i-1) - 0.023\*Vd(i-1);

end

end

% Non-Coherent Detection Step 2: Low Pass RC Filter

h = fir1(100, 0.0125, 'low'); % 1 kHz cut-off frequency

foutputc = filter(h,1,Vc);

figure;

subplot(3,1,1);

plot(t, Vc);

title('Envelope detector output of FCAM signal'); xlabel('time (s)'); ylabel('amplitude');

subplot(3,1,2);

plot(t, Vd);

title('Envelope detector output of DSB-SC signal'); xlabel('time (s)'); ylabel('amplitude');

subplot(3,1,3);

plot(t, foutputc);

title('Non-coherent demodulated FCAM signal'); xlabel('time (s)'); ylabel('amplitude');

Question 3 To simulate BASK, BFSK, BPSK technique using matalb/scilab software.

Answer Following files have been created using matlab software

* Amplitude Shift Keying

clear all;

close all;

fc=100;

fp=10;

A=5;

t=0:0.001:1;

c=A.\*sin(2\*pi\*fc\*t);

subplot(3,1,1);

plot(t,c);

m=A/2.\*square(2\*pi\*fp\*t)+A/2;

subplot(3,1,2);

plot(t,m);

w=c.\*m;

subplot(3,1,3);

plot(t,w);

title('ASK');

* Frequency Shift Keying

clear all;

close all;

I = [0 1 1 0 1 0 0 1];

f = input('Enter frequency of carrier: ');

t=0:1/512:1;

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

Xfsk= [];

x1 = sin(2\*pi\*f\*t);

x2 = sin(2\*pi\*2\*f\*t);

for n=1:length(I)

if(I(n) == 1)

Xfsk=[Xfsk,x2];

elseif(I(n)~=1)

Xfsk=[Xfsk, x1];

end

end

subplot(3,1,1);

plot(t, x);

subplot(3,1,2);

plot(Xfsk);

* Phase Shift Keying

clear all;

close all;

A=5;

t=0:0.001:1;

f1=10;

f2=2;

x=A.\*sin(2\*pi\*f1\*t);

subplot(3,1,1);

plot(t,x);

title('carrier signal')

xlabel('time');

ylabel('amplitude');

u=square(2\*pi\*f2\*t);

subplot(3,1,2);

plot(t,u);

title('modulating signal')

xlabel('time');

ylabel('amplitude');

v=x.\*u;

subplot(3,1,3);

plot(t,v);

xlabel('time');

ylabel('amplitude');

title('psk');

Question 4 Create a sine wave and analyze the effect of adding noise to it.  
  
Answer The Following Program has been created in SciLab

clc;

clear all;

order=18;

t=0:0.01:1;/

x=sin(2\*%pi\*5\*t);

noise=rand(1,length(x));

x\_n=x+noise;

ref\_noise=noise\*rand(10);

w=zeros(order,1);

mu=0.01\*(sum(x.^2)/length(x));

N=length(x);

for k =1:1010

for i =1:N-order-1

buffer=ref\_noise(i:i + order -1);

desired(i)= x\_n(i)- buffer\*w ;

w = w+(buffer\*mu\*desired(i))';

end

end

subplot(4,1,1)

plot2d(t,x)

title('original signal')

subplot(4,1,2)

plot2d(t,noise,2)

title('Noise random signal')

subplot(4,1,3)

plot2d(t,x\_n,5)

title('Signal + Noise')

subplot(4,1,4)

plot(desired)

title('Noise remove signal')

Question 5 To generate dc, ac, combination of dc and ac, unit step, unit impulse, unit ramps in matlab.

* Write a program in matlab to calculate energy and power in a signal.
* Also plot energy and power spectrum for the same.

Answer

a) %\*\*\*impulse function \*\*\*

n = -10:10;

delta =(n==0);

subplot(4,1,1);

stem(n,delta,'.');

%\*\*\*unit step function\*\*\*

step =(n>=0);

subplot(4,1,2);

stem(n,step,'.');

%\*\*\*ramp function\*\*\*

ramp=n.\*(n>0);

subplot(4,1,3);

stem(n,ramp,'.');

b) %\*\*\*rectangular function\*\*\*

t = -10:0.01:10;

y = heaviside(t+5)- heaviside(t-5);

plot(t,y);

c) Energy and Power

clear all;

close all;

%x(n)=(1/2)^n\*u(n);

n=0:1:1000;

fs=1000;

x=(1/2).^n;

stem(n,x);

axis([0 25 0 1]);

disp('Calculate Energy of Signal');

E=sum(abs(x).^2);

energy=spectrum(x,n);

P=E/length(x);

Question 6 To generate PAM, PWM and PPM using matlab/scilab software.

Answer

* Pulse Ampitude Modulation

a=5;

f=1;

t=0:0.01:2;

x1=stem(t/3);

x2=sin(2\*pi\*f\*t);

y=x1.\*x2;

subplot(3,1,1);

stem(x1);

title('impulse signal');

xlabel( 'time index ');

ylabel('amplitude');

subplot(3,1,2);

plot(t,x2);

title( 'sin wave ');

subplot(3,1,3);

stem(t,y);

title('pam signal');

* Pulse Width Modulation

t=0:0.001:1;

s=sawtooth(2\*pi\*10\*t+pi);

m=0.75\*sin(2\*pi\*1\*t);

n=length(s);

for i=1:n

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

PWM(i)=1;

else if(m(i)<=s(i))

PWM(i)=0;

end

end

end

plot(t,PWM,'-g',t,m,'--r',t,s,'--b');

ylabel('amplitude');

xlabel('time index');

title('PWMWave');

axis([0 1 -1.5 1.5]);

* Pulse Phase Modulation

clear all;

close all;

fc=1000;

fs=10000;

f1=200;

t=0:1/fs:(((2/f1)-(1/fs)));

x1=0.4\*cos(2\*pi\*f1\*t)+0.5;

y1=modulate(x1,fc,fs,'ppm');

subplot(2,2,1);

plot(x1);

title('message signal');

subplot(2,2,2);

plot(y1);

axis([0 200 -0.2 1.2]);

title('ppm');

x\_rec=demod(y1,fc,fs,'ppm');

subplot(2,2,3);

plot(x\_rec);

Question 7 Matlab implementation of PCM with bit depth of 4.

clc;

close all;

clear all;

n=input('Enter n value for n-bit PCM system : ');

n1=input('Enter number of samples in a period : ');

L=2^n;

% % Signal Generation

% x=0:1/100:4\*pi;

% y=8\*sin(x); % Amplitude Of signal is 8v

% subplot(2,2,1);

% plot(x,y);grid on;

% Sampling Operation

x=0:2\*pi/n1:4\*pi; % n1 nuber of samples have tobe selected

s=8\*sin(x);

subplot(3,1,1);

plot(s);

title('Analog Signal');

ylabel('Amplitude--->');

xlabel('Time--->');

subplot(3,1,2);

stem(s);grid on; title('Sampled Sinal'); ylabel('Amplitude--->'); xlabel('Time--->');

% Quantization Process

vmax=8;

vmin=-vmax;

del=(vmax-vmin)/L;

part=vmin:del:vmax; % level are between vmin and vmax with difference of del

code=vmin-(del/2):del:vmax+(del/2); % Contaion Quantized valuses

[ind,q]=quantiz(s,part,code); % Quantization process

% ind contain index number and q contain quantized values

l1=length(ind);

l2=length(q);

for i=1:l1

if(ind(i)~=0) % To make index as binary decimal so started from 0 to N

ind(i)=ind(i)-1;

end

i=i+1;

end

for i=1:l2

if(q(i)==vmin-(del/2)) % To make quantize value inbetween the levels

q(i)=vmin+(del/2);

end

end

subplot(3,1,3);

stem(q);grid on; % Display the Quantize values

title('Quantized Signal');

ylabel('Amplitude--->');

xlabel('Time--->');

% Encoding Process

figure

code=de2bi(ind,'left-msb'); % Cnvert the decimal to binary

k=1;

for i=1:l1

for j=1:n

coded(k)=code(i,j); % convert code matrix to a coded row vector

j=j+1;

k=k+1;

end

i=i+1;

end

subplot(2,1,1); grid on;

stairs(coded); % Display the encoded signal

axis([0 100 -2 3]); title('Encoded Signal');

ylabel('/Amplitude--->');

xlabel('Time--->');

% Demodulation Of PCM signal

qunt=reshape(coded,n,length(coded)/n);

index=bi2de(qunt','left-msb'); % Getback the index in decimal form

q=del\*index+vmin+(del/2); % getback Quantized values

subplot(2,1,2); grid on;

plot(q); % Plot Demodulated signal

title('Demodulated Signal');

ylabel('Amplitude--->');

xlabel('Time--->');

Question 8 Plot SNR vs BER curve for BPSK technique.

clc;

clear all;

close all;

bit\_count=100000; % Number of random bits to be generated

SNR= 0:1:10; %Range S/N over which is simulate.

for k=1:1:length(SNR)

tote=0; %total error bits

totb=0; %total bits

while tote<100 %untill you get 100 error

rbits=round(rand(1,bit\_count)); %generate random bits

tx=-2\*(rbits-0.5); %bpsk modulation directly to bipolar NRZ

No=(1/10)^(SNR(k)/10); %Noise level

rx=tx+sqrt(No/2)\*(randn(1,length(tx))+ i\*randn(1,length(tx)));

rx2=rx<0 ; % BPSK demodulator logic at ry

diff=rbits-rx2 ; % Caclulate bit error

tote=tote+sum(abs(diff)); %total error

totb=totb+length(rbits); %total bits generated

end

ber(k)=tote/totb; %bit error rate

end

semilogy(SNR,ber,'\*r');

xlabel('S/N(db)');

ylabel('ber');

title('S/N vs ber plot for BPSK');

thber=0.5\*erfc(sqrt(10.^(SNR/10))); %theoretical ber

semilogy(SNR,thber);

legend('simulated curve','theoretical curve');