

## 7,10.a,8b "NRZ RZ Sin Raised"

```
clc;clear;close all;

x=[1 1 0 0 1 1
0];bitrate=1;sampling_rate=100;t=0:1/sampling_rate:length(x);

signal=zeros(1,length(t));signal1=signal;

for i=1:length(x)

if x(i)==1

signal((i-1)*sampling_rate+1:i*sampling_rate)=1;

signal1((i-1)*sampling_rate+1:(i-1/2)*sampling_rate)=1;

else

signal((i-1)*sampling_rate+1:i*sampling_rate)=-1;

signal1((i-1)*sampling_rate+1:(i-1/2)*sampling_rate)=-1;

end

end

subplot(3,1,1);

plot(t,signal),axis([0 length(x) -1.5 1.5]),grid on;

xlabel('Time');ylabel('Amplitude');title('Polar NRZ plot');

subplot(3,1,2);

plot(t,signal1),axis([0 length(x) -1.5 1.5]),grid on;

xlabel('Time');ylabel('Amplitude');title('Polar RZ plot');

t=0:1/sampling_rate:1;f=0.5;

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

x(x<0)=0;

subplot(3,1,3);

plot(t,x),axis([0 1 0 1.2]),grid on;

xlabel('Time');ylabel('Amplitude');title('Half sinusoidal pulse');

nSamp=4;span=10;rolloff=0.25;

txfilter=comm.RaisedCosineTransmitFilter('RolloffFactor',rolloff,'FilterSpanInSymbols',span,'OutputSamplesPerSymbol',nSamp);
```

```
fvtool(txfilter,impulse);

data=sign(randn(1,400));

T=64;

for i=1:length(data)

daTp((i-1)*64+1:i*64)=[data(i) zeros(1,63)];

end

yrz=conv(daTp,prz(T));

yrz=yrz(1:end-T+1);

ynrz=conv(daTp,pnrz(T));

ynrz=ynrz(1:end-T+1);

ysine=conv(daTp,psine(T));

ysine=ysine(1:end-T+1);Td=4;

yrcos=conv(daTp,prcos(0.5,Td,T));

yrcos=yrcos(2*Td*T:end-2*Td*T+1);

eye1=eyediagram(yrz,2*T,T,T/2);title('RZ Eye-Diagram');

eye2=eyediagram(ynrz,2*T,T,T/2);

title('NRZ Eye-Diagram');

eye3=eyediagram(ysine,2*T,T,T/2);

title('Half-Sine Eye-Diagram');

eye4=eyediagram(yrcos,2*T,T);

title('Raised-Cosine Eye-Diagram');

function pout=prz(T)

pout=[zeros(1,T/4) ones(1,T/2) zeros(1,T/4)];

end

function y=prcos(rollfac,length,T)

y=rcosfir(rollfac,length,T,1,'normal');end

function pout=psine(T)

pout=sin(pi*(0:T-1)/T);

end

function pout=pnrz(T)

pout=ones(1,T);

end
```

## 9. "PCM Encoding Decoding"

```
clc;clear;close all;

f=2;fs=20*f;t=0:1/fs:1;a=2;x=a*sin(2*pi*f*t);

figure;

subplot(4,1,1);plot(t,x,'r-');

xlabel('Time');ylabel('Amplitude');title('Input
Signal');grid on;

x1=x+a;

q_op=round(x1);

subplot(4,1,2);plot(t,q_op,'k+-');

xlabel('Time');ylabel('Amplitude');title('Quantized
Signal');grid on;

enco=de2bi(q_op,'left-msb');

[m,n]=size(enco);

y3=reshape(enco',1,m*n);

t=linspace(0,1.23,1230);

signal=zeros(1,length(y3)*10);

for i=1:length(y3)
    if y3(i)==1
        signal((i-1)*10+1:i*10)=1;
    else
        signal((i-1)*10+1:i*10)=-1;
    end
end

subplot(4,1,3);plot(t,signal);axis([0 1.23 -2 2]);grid
on;

xlabel('Time');ylabel('Amplitude');title('PCM
Signal');

y4=zeros(1,length(signal)/10);

for i=1:length(signal)/10
    if all(signal((i-1)*10+1:i*10)==1)
        y4(i)=1;
    else
        y4(i)=0;
    end
end
```

```
end

end

y5=reshape(y4,m,n);

deco=bi2de(y5,'left-msb');

xr=deco-a;

t=0:1/fs:1;

subplot(4,1,4);

plot(t,xr);

xlabel('Time');ylabel('Amplitude');title('PCM
Demodulated Signal');grid on;
```

## 10.b BIT ERROR

```
clc;clear;close all;

EbN0dB=0:10;

EbN0=10.^(EbN0dB/10);

pe_bpsk=0.5*erfc(sqrt(EbN0));

pe_bfsk=0.5*erfc(sqrt(EbN0/2));

pe_bask=0.5*erfc(sqrt(EbN0/4));

semilogy(EbN0dB,pe_bpsk,'r*-
',EbN0dB,pe_bfsk,'k+-',EbN0dB,pe_bask,'b>-');

legend('BPSK','BFSK','BASK');

xlabel('Eb/No(dB)');

ylabel('BER');

grid on;
```

## 11. DPSK

```
clc;clear;close all;

x=[1 0 1 0 1 0 1 0];

disp('Input bits');

disp(x);

x1(1)=bitxor(1,x(1));

for i=2:8

x1(i)=bitxor(x1(i-1),x(i));

end

x1=[1,x1];

t=linspace(0,0.9,900);

y=sin(2*pi*50*t);

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

y1=zeros(1,length(t));

for i=1:length(x1)

if x1(i)==1

signal(1,(i-1)*100+1:i*100)=1;

else

y1(1,(i-1)*100+1:i*100)=signal(1,(i-1)*100+1:i*100).*y(1,(i-1)*100+1:i*100);

signal(1,(i-1)*100+1:i*100)=-1;

y1(1,(i-1)*100+1:i*100)=signal(1,(i-1)*100+1:i*100).*y(1,(i-1)*100+1:i*100);

end

end

subplot(3,1,1);plot(t,signal);axis([0 0.9 -2 2]);grid on;

xlabel('Time in seconds');ylabel('Amplitude');title('Coded Input Signal');

subplot(3,1,2);plot(t,y);axis([0 0.9 -2 2]);grid on;

xlabel('Time in seconds');ylabel('Amplitude');title('Carrier');

subplot(3,1,3);plot(t(1:900),y1);axis([0 0.9 -2 2]);grid on;
```

```
xlabel('Time in seconds');ylabel('Amplitude');title('DPSK Modulated Wave');

zl=zeros(9,100);

for i=1:9

zl(i,:)=y1(100*(i-1)+1:100*i);

end

q=zeros(1,9);

q(1)=1;

for i=2:9

q(i)=sum(zl(i,:).*zl(1,:));

if q(i)>0

q(i-1)=1;

else

q(i-1)=0;

end

end

disp('Decoded Data');

disp(q);
```

## 12.QPSK

```
clc;clear;close all;
x=[0 0 0 1 0 1 1];
disp('Input bits');
disp(x);
t=linspace(0,0.8,800);
y1=sin(2*pi*50*t);
y2=sin(2*pi*50*t+pi/2);
y3=sin(2*pi*50*t+pi);
y4=sin(2*pi*50*t+3*pi/2);figure;
subplot(411);plot(t,y1);axis([0 0.8 -2 2]);grid on;
xlabel('Time in
seconds');ylabel('Amplitude');title('Carrier with
phase 0');
subplot(412);plot(t,y2);axis([0 0.8 -2 2]);grid on;
xlabel('Time in
seconds');ylabel('Amplitude');title('Carrier with
phase 90');
subplot(413);plot(t,y3);axis([0 0.8 -2 2]);grid on;
xlabel('Time in
seconds');ylabel('Amplitude');title('Carrier with
phase 180');
subplot(414);plot(t,y4);axis([0 0.8 -2 2]);grid on;
xlabel('Time in
seconds');ylabel('Amplitude');title('Carrier with
phase 270');j=1;
signal=zeros(1,length(t));
for i=1:2:length(x)
    if x(i)==0 && x(i+1)==0
        signal(1,(i-1)*100+1:i*100)=-1;
        signal(1,i*100+1:(i+1)*100)=-1;
        q1(1,j*100+1:(j+1)*100)=y1(1:100);
    elseif x(i)==0 && x(i+1)==1
        signal(1,(i-1)*100+1:i*100)=-1;
        signal(1,i*100+1:(i+1)*100)=1;
```

```
        q1(1,j*100+1:(j+1)*100)=y2(1:100);
    elseif x(i)==1 && x(i+1)==0
        signal(1,(i-1)*100+1:i*100)=1;
        signal(1,i*100+1:(i+1)*100)=-1;
        q1(1,j*100+1:(j+1)*100)=y3(1:100);
    elseif x(i)==1 && x(i+1)==1
        signal(1,(i-1)*100+1:i*100)=1;
        signal(1,i*100+1:(i+1)*100)=1;
        q1(1,j*100+1:(j+1)*100)=y4(1:100);end
    j=j+1;endfigure;
subplot(211);plot(t,signal);axis([0 0.8 -2 2]);grid on;
xlabel('Time in
seconds');ylabel('Amplitude');title('Input Signal');
subplot(212);plot(t(1:400),q1);axis([0 0.8 -2 2]);grid
on;
xlabel('Time in
seconds');ylabel('Amplitude');title('QPSK
Modulated Wave');i=1;
for j=1:4
    if corr(q1(j*100-99:j*100),y1(1:100))>0
        x1(i)=0;
        x1(i+1)=0;
    elseif corr(q1(j*100-99:j*100),y2(1:100))>0
        x1(i)=0;
        x1(i+1)=1;
    elseif corr(q1(j*100-99:j*100),y3(1:100))>0
        x1(i)=1;
        x1(i+1)=0;
    elseif corr(q1(j*100-99:j*100),y4(1:100))>0
        x1(i)=1;
        x1(i+1)=1;end
    i=i+2;end
disp('Received bits');
disp(x1);
```