APLITUDE MODULATION AND DEMODULATION

```
close all;
clear all;
clc;
fs=8000;
fm=20;
fc=500;
Am=1;
Ac=1;
t=[0:0.1*fs]/fs;
m=Am*cos(2*pi*fm*t);
c=Ac*cos(2*pi*fc*t);
ka=0.5;
u=ka*Am;
s1=Ac*(1+u*cos(2*pi*fm*t)).*cos(2*pi*fc*t);
subplot(4,3,1:3);
plot(t,m);
title('Modulating or Message signal(fm=20Hz)');
subplot(4,3,4:6);
plot(t,c);
title('Carrier signal(fc=500Hz)');
subplot(4,3,7);
plot(t,s1);
title('Under Modulated signal(ka.Am=0.5)');
Am=2;
ka=0.5;
u=ka*Am;
s2=Ac*(1+u*cos(2*pi*fm*t)).*cos(2*pi*fc*t);
subplot(4,3,8);
plot(t,s2);
title('Exact Modulated signal(ka.Am=1)');
Am=5;
ka=0.5;
u=ka*Am;
s3=Ac*(1+u*cos(2*pi*fm*t)).*cos(2*pi*fc*t);
subplot(4,3,9);
plot(t,s3);
title('Over Modulated signal(ka.Am=2.5)');
 r1= s1.*c;
[b a] = butter(1,0.01);
mr1= filter(b,a,r1);
subplot(4,3,10);
plot(t,mr1);
title(' deModulated signal for(ka.Am=0.5)');
r2= s2.*c;
[b a] = butter(1,0.01);
mr2= filter(b,a,r2);
subplot(4,3,11);
plot(t,mr2);
title(' deModulated signal for(ka.Am=1)');
r3= s3.*c;
[b a] = butter(1,0.01);
mr3= filter(b,a,r3);
subplot(4,3,12);
plot(t,mr3);
title(' deModulated signal for(ka.Am=2.5)');
```

FREQUENCY MODULATION

```
close all;
clear all;
clc;
fs = 10000;  % Sampling frequency
Ac = 1;      % Carrier amplitude
Am = 1;      % Modulating signal
fm = 35;     % Modulating frequency
fc = 500;    % Carrier frequency
B = 10;      % Modulation index
t = (0:(1/fs):0.1);
wm = 2 * pi * fm;
m_t = Am * cos(wm * t);
subplot(4, 1, 1);
plot(t, m_t);
title('Modulating or Message signal
(fm=35Hz)');
xlabel('Time (s)');
ylabel('Amplitude');
grid on;
 wc = 2 * pi * fc;
c_t = Ac * cos(wc * t);
subplot(4, 1, 2);
plot(t, c_t);
title('Carrier signal (fc=500Hz)');
xlabel('Time (s)');
ylabel('Amplitude');
grid on;
s_t = Ac * cos(wc * t + B * sin(wm * t));
subplot(4, 1, 3);
plot(t, s_t);
title('FM Modulated signal');
xlabel('Time (s)');
ylabel('Amplitude');
grid on;
 frequencyDeviation = B * fm;
d = fmdemod(s_t, fc, fs,
frequencyDeviation);
subplot(4, 1, 4);
plot(t, d);
title('Demodulated signal');
xlabel('Time (s)');
ylabel('Amplitude');
grid on;
```

DSB-SC MODULATOR & DETECTOR

```
close all
clear all
clc
t =0:0.000001:.001;
Vm= 1;
Vc= 1;
fm = 2000;
fc= 50000;
m_t = Vm*sin(2*pi*fm*t);
subplot(4,1,1);
plot(t,m_t);
c_t = Vc*sin(2*pi*fc*t);
subplot(4,1,2);
plot(t,c_t);
subplot(4,1,3);
s_t = m_t.*c_t;
hold on;
plot(t,s_t);
plot(t,m_t,'r:');
plot(t,-m_t,'r:');
hold off;
r = s_t.*c_t;
[b a] = butter(1,0.01);
mr= filter(b,a,r);
subplot(4,1,4);
plot(t,mr);
```

SIMULATION OF DSSS GENERATION SCHEMES

```
clc;
close all;
clear all;
b = [1 0 1 0 1 0 1 0];
ln = length(b);
for i = 1:ln
   if b(i) == 0
      b(i) = -1;
   end
end
k = 1;
for i = 1:ln
   for j = 1:8
      bb(k) = b(i);
      k = k + 1;
   end
end
len = length(bb);
pr_sig = round(rand(1, len));
for i = 1:len
   if pr_sig(i) == 0
      pr_sig(i) = -1; % Convert 0 to -1
   end
end
bbs = bb .* pr_sig;
dsss = [];
t = 0:0.1/10:2*pi;
c1 = cos(t);   % Carrier for bit -1
c2 = cos(t + pi); % Carrier for bit 1
for k = 1:len
   if bbs(k) == -1
      dsss = [dsss c1]; % Use carrier c1 for -1
   else
      dsss = [dsss c2]; % Use carrier c2 for 1
   end
end
figure;
subplot(3, 1, 1);
stairs(bb, 'linewidth', 2);
axis([0 len -2 3]);
title('Original Bit Sequence b(t)');
xlabel('Time');
ylabel('Amplitude');
grid on;
subplot(3, 1, 2);
stairs(pr_sig, 'linewidth', 2);
axis([0 len -2 3]);
title('Pseudo-random Bit Sequence
pr\_sig(t)');
xlabel('Time');
ylabel('Amplitude');
grid on;
subplot(3, 1, 3);
plot(dsss, 'linewidth', 1);
title('Spread Spectrum Signal (DSSS)');
xlabel('Time');
ylabel('Amplitude');
grid on;
```

EYE DIAGRAM

```
N=100;
X=2*round(rand(N,1))-1;
 X1=X';
 a=-5.5:1/100:1.5;
 b=-3.5:1/100:3.5;
 c=-1.5:1/100:5.5;
 for i=1:1:(N+2-11);
 d=X1(1,i)*sinc(b);
 e=X1(1,i+1)*sinc(b);
 f=X1(1,i+2)*sinc(b);
 hold on, plot(a,f);
 hold on, plot(b,e);
 hold on, plot(c,d);
 end
 title('Eye Diagram - Before Equalization');
 xlabel('Time(s)');
 ylabel('Amplitude'); axis([-1 1 -2 2])
```

SIMULATION OF ASK GENERATION & DETECTION SCHEMES

```
clc;
clear all;
close all;
fm=input("enter the message frequency:");
fc=input("enter the carrier frequency:");
A=1;
t=0:0.001:1;
m=(A.*square(2*pi*fm*t)+A)/2;
c=sin(2*pi*fc*t);
s=m.*c;
subplot(5,1,1);
plot(t,m);
title('MESSAGE SIGNAL');
xlabel('t -- >');
ylabel('m(t)');
grid on;
subplot(5,1,2);
plot(t,c);
title('CARRIER SIGNAL');
xlabel('t -- >');
ylabel('c(t)');
grid on;
subplot(5,1,3);
plot(t,s);
title('ASKSIGNAL');
xlabel('t --- >');
ylabel('s(t)');
grid on;
y_dem =c.*s;
t4=0:0.001:1;
z=square(y_dem); % intregation
%A_dem=round((2*z/0.001));
for i=0:1000
   if(y_dem(i+1)>0)
      X(i+1)=1;
   else
X(i+1)=0;
   end
end
subplot(5,1,4);
plot(t,X)
subplot(5,1,5);
plot(t,X)
title('ASK Demodulated signal');
xlabel('t -- >');
ylabel('b(n)');
```

SIMULATION OF FSK GENERATION & DETECTION SCHEMES

```
clc;
clear all;
close all;
f1 = input("Enter the frequency of Carrier 1 (Hz): ");
f2 = input("Enter the frequency of Carrier 2 (Hz): ");
f3 = input("Enter the frequency of Message signal
(Hz): ");
t = 0:0.001:1; % Time vector
m = (square(2*pi*f3*t) + 1) / 2; % Binary message
signal (0 and 1)
c1 = sin(2*pi*f1*t); % Carrier signal 1
c2 = sin(2*pi*f2*t); % Carrier signal 2
s = zeros(1, length(t)); % Initialize FSK signal
for i = 1:length(t)
   if m(i) == 1
      s(i) = c1(i); % Use Carrier 1 for binary 1
   else
      s(i) = c2(i); % Use Carrier 2 for binary 0
   end
end
```

```
subplot(3, 2, 1);
plot(t, m, 'LineWidth', 1.5);
xlabel('Time (s)');
ylabel('Amplitude');
title('Message Signal');
grid on;
subplot(3, 2, 2);
plot(t, c1, 'LineWidth', 1.5);
xlabel('Time (s)');
ylabel('Amplitude');
title('Carrier 1 Signal');
grid on;
subplot(3, 2, 3);
plot(t, c2, 'LineWidth', 1.5);
xlabel('Time (s)');
ylabel('Amplitude');
title('Carrier 2 Signal');
grid on;
subplot(3, 2, 4);
plot(t, s, 'LineWidth', 1.5);
xlabel('Time (s)');
ylabel('Amplitude');
title('FSK Modulated Signal');
grid on;
y2 = zeros(1, length(t)); % Initialize
demodulated signal
for j = 1:length(t)
   if s(j) == c1(j)
      y2(j) = 1; % Detect binary 1
   else
      y2(j) = 0; % Detect binary 0
   end
end
subplot(3, 2, 5);
plot(t, y2, 'LineWidth', 1.5);
xlabel('Time (s)');
ylabel('Amplitude');
title('Demodulated FSK Signal');
grid on;
```