

MASINDE MULIRO UNIVERSITY OF SCIENCE & TECHNOLOGY
DEPARTMENT OF ELECTRICAL & COMMUNICATION ENGINEERING

LAB 01: GENERATION, MODULATION & DEMODULATION OF ANALOG SIGNALS USING MATLAB

AMPLITUDE MODULATION (AM)

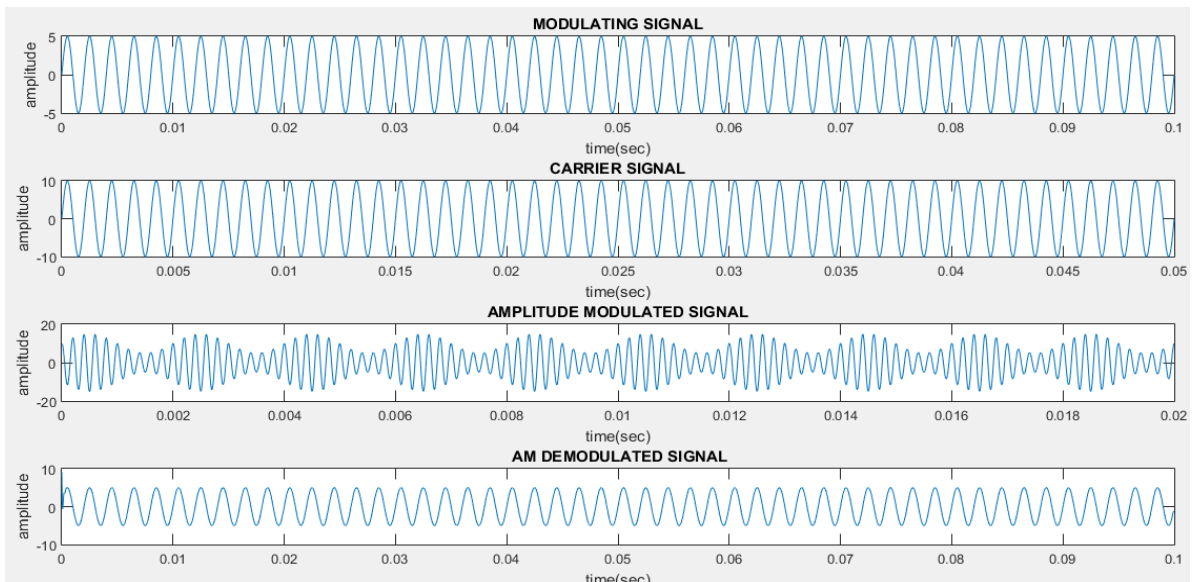
Purpose:

To Generate & to detect AM signals using MATLAB2010a or Later Version.

Matlab Code

```
t=linspace(0,0.1,50000); % time range of signals with 100,000 steps
fc=1000; %frequency of carrier signal
fm=500; %frequency of message signal
fs=100000; %sampling frequency---fs>=2(fc+BW)
Am=5; %amplitude of the message signal
Ac=10; %amplitude of the carrier signal
m=Am/Ac; %modulation index for the AM wave
wc=2*pi*fc*t; %carrier frequency in radians
wm=2*pi*fm*t; %message frequency in radians
ec=Ac*sin(wc); %carrier signal
em=Am*sin(wm); %message signal
y=ammod(em,fc,fs,0,Ac); % generate AM signal
z=amdemod(y,fc,fs,0,Ac); %demodulate AM signal
l=50000;
subplot(4,1,1),
plot(t(1:l),em(1:l))
xlabel('time(sec)');
ylabel('amplitude');
title('MODULATING SIGNAL');
subplot(4,1,2),
plot(t(1:l/2),ec(1:l/2))
xlabel('time(sec)');
ylabel('amplitude');
title('CARRIER SIGNAL');
subplot(4,1,3),
plot(t(1:l),y(1:l))
axis([0 0.02 -20 20]) %setting axis dimensions
xlabel('time(sec)');
ylabel('amplitude');
title('AMPLITUDE MODULATED SIGNAL');
subplot(4,1,4),
plot(t(1:l),z(1:l))
xlabel('time(sec)');
ylabel('amplitude');
title(' AM DEMODULATED SIGNAL');
```

Matlab Output



LAB 02: GENERATION & DETECTION OF ANALOG SIGNALS USING MATLAB

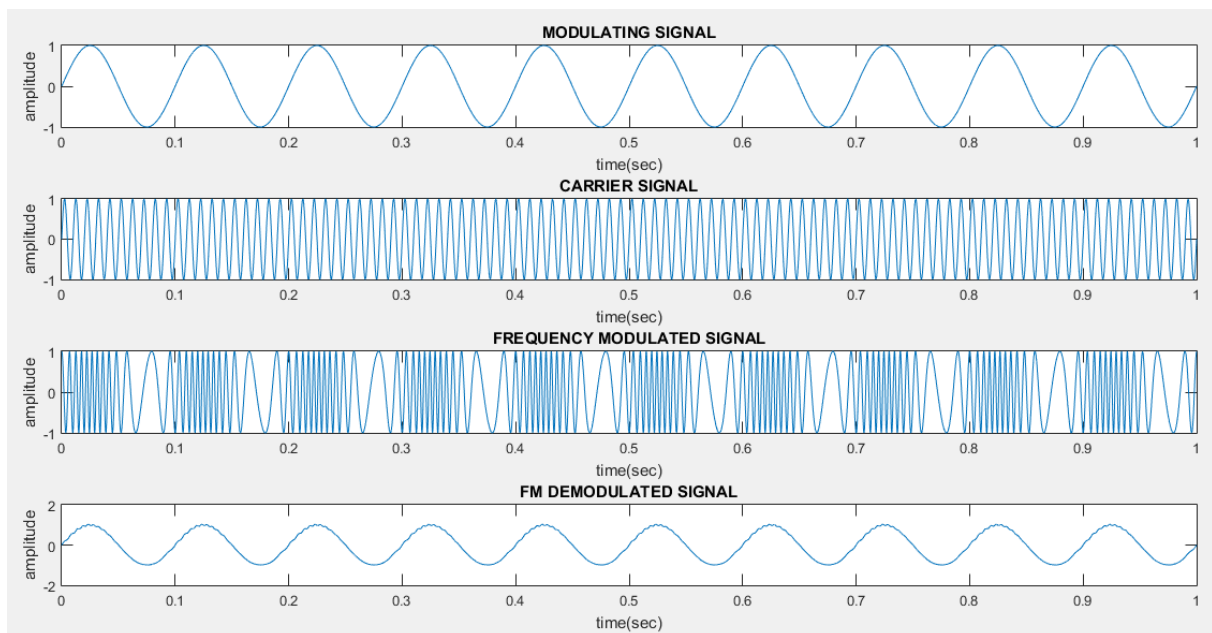
FREQUENCY MODULATION (FM)

Purpose:

To Generate & to detect FM signals using MATLAB2010a or Later Version.

Matlab Code

```
Fs = 8000; % Sampling rate of signal
Fc = 100; % Carrier frequency
fm=10; %frequency of message
t = linspace(0,1,10000); % Sampling times
x = sin(2*pi*fm*t); % Generate baseband signal
dev = 75; % Frequency deviation in modulated signal
y = fmmmod(x,Fc,Fs,dev); % Modulate both channels.
z = fmdemod(y,Fc,Fs,dev); % Demodulate both channels.
subplot(4,1,1),
plot(t,x)
xlabel('time(sec)');
ylabel('amplitude');
title('MODULATING SIGNAL');
subplot(4,1,2),
plot(t,sin(2*pi*Fc*t))
xlabel('time(sec)');
ylabel('amplitude');
title('CARRIER SIGNAL');
subplot(4,1,3),
plot(t,y)
xlabel('time(sec)');
ylabel('amplitude');
title('FREQUENCY MODULATED SIGNAL');
subplot(4,1,4),
plot(t,z)
xlabel('time(sec)');
ylabel('amplitude');
title('FM DEMODULATED SIGNAL');
```



LAB 03: GENERATION & DETECTION OF ANALOG SIGNALS USING MATLAB

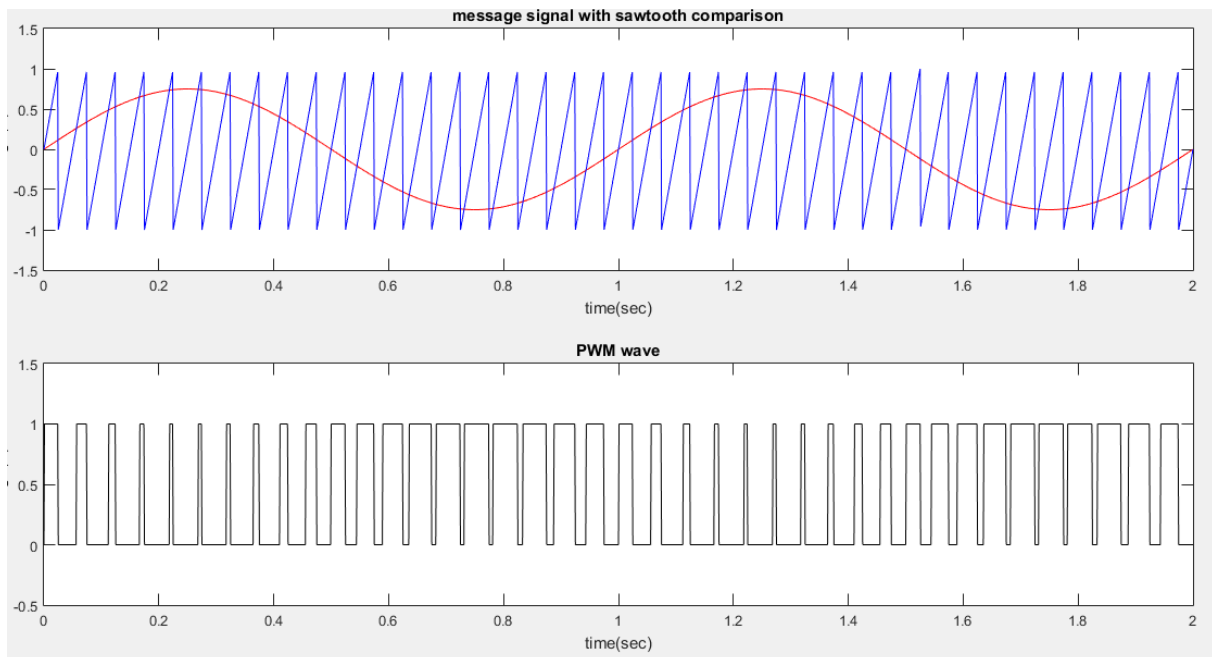
PULSE WIDTH MODULATION (PWM)

Purpose:

To Generate & to detect PWM signals using MATLAB2010a or Later Version.

Matlab Code

```
t=0:0.001:2;  
s=sawtooth(2*pi*20*t+pi);  
m=0.75*sin(2*pi*1*t);  
n=length(s);  
for i=1:n  
    if (m(i)>=s(i))  
pwm(i)=0;  
elseif (m(i)<=s(i))  
pwm(i)=1;  
end  
end  
subplot(2,1,1),  
plot(t,m,'-r',t,s,'-b');  
axis([0 2 -1.5 1.5]);  
title('message signal with sawtooth comparison')  
xlabel('time(sec)');  
ylabel('voltage(V)');  
subplot(2,1,2),  
plot(t,pwm,'-k')  
axis([0 2 -0.5 1.5]);  
title('PWM wave');  
xlabel('time(sec)');  
ylabel('voltage(V)');
```



LAB 03: GENERATION & DETECTION OF ANALOG SIGNALS USING MATLAB

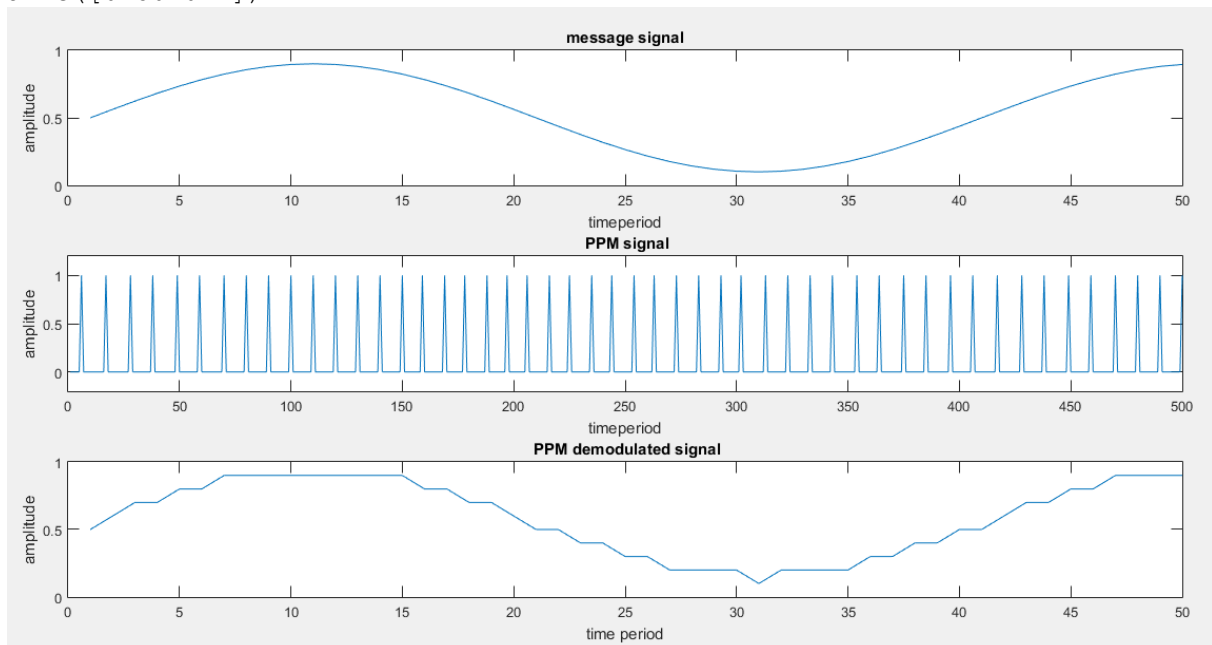
PULSE POSITION MODULATION (PPM)

Purpose:

To Generate & to detect PPM signals using MATLAB2010a or Later Version.

Matlab Code

```
fc=4000;  
fs=40000;  
fm=1000;  
t=0:1/fs:(2/fm-1/fs);  
mt=0.4*sin(2*pi*fm*t)+0.5;  
st=modulate(mt,fc,fs,'PPM');  
dt=demod(st,fc,fs,'PPM');  
figure  
subplot(3,1,1);  
plot(mt);  
title('message signal');  
xlabel('time period');  
ylabel('amplitude');  
axis([0 50 0 1])  
subplot(3,1,2);  
plot(st);  
title(' PPM signal');  
xlabel('time period');  
ylabel('amplitude');  
axis([0 500 -0.2 1.2])  
subplot(3,1,3);  
plot(dt);  
title(' PPM demodulated signal');  
xlabel('time period');  
ylabel('amplitude');  
axis([0 50 0 1])
```



LAB 04: GENERATION & DETECTION OF ANALOG SIGNALS USING MATLAB

PULSE AMPLITUDE MODULATION (PAM)

Purpose:

To Generate & to detect PAM signals using MATLAB2010a or Later Version.

Matlab Code

```
fc=100;
fm=fc/10;
fs=100*fc;
t=0:1/fs:4/fm;
mt=cos(2*pi*fm*t);
ct=0.5*square(2*pi*fc*t)+0.5;
st=mt.*ct;
    tt=[ ];           %single sided PAM
for i=1:length(st);
    if st(i)==0;
        tt=[tt,st(i)];
    else
        tt=[tt,st(i)+2];
    end
end
figure(1)
subplot(4,1,1);
    plot(t,mt);
    title('message signal');
    xlabel('time period');
    ylabel('amplitude');
subplot(4,1,2);
    plot(t,ct);
    title('carrier signal');
    xlabel('time period');
    ylabel('amplitude');
subplot(4,1,3);
    plot(t,st);
    title('BIPOLAR PAM');
    xlabel('timeperiod');
    ylabel('amplitude');
subplot(4,1,4);
    plot(t,tt);
    title('PAM');
    xlabel('timeperiod');
    ylabel('amplitude'); %demodulation
    dt=st.*ct;
dt_frequency=fftshift(abs(fft(dt)));
filter=fir1(200, fm/fs, 'low');
original_t_signal=conv(filter,dt);
original_f_signal=fftshift(abs(fft(original_t_signal)));
t1=0:1/(length(original_t_signal)-1):1;
    f=-fs/2:fs/(length(original_f_signal)-1):fs/2;
figure(2)
subplot(2,1,1);
    plot(t1,original_t_signal);
    title('time domain signal');
    xlabel('timeperiod');
    ylabel('amplitude');
subplot(2,1,2);
    plot(f,original_f_signal);
    title('frequency domain signal');
    xlabel('frequency');
    ylabel('amplitude');
axis([-50 50 0 2000])
```

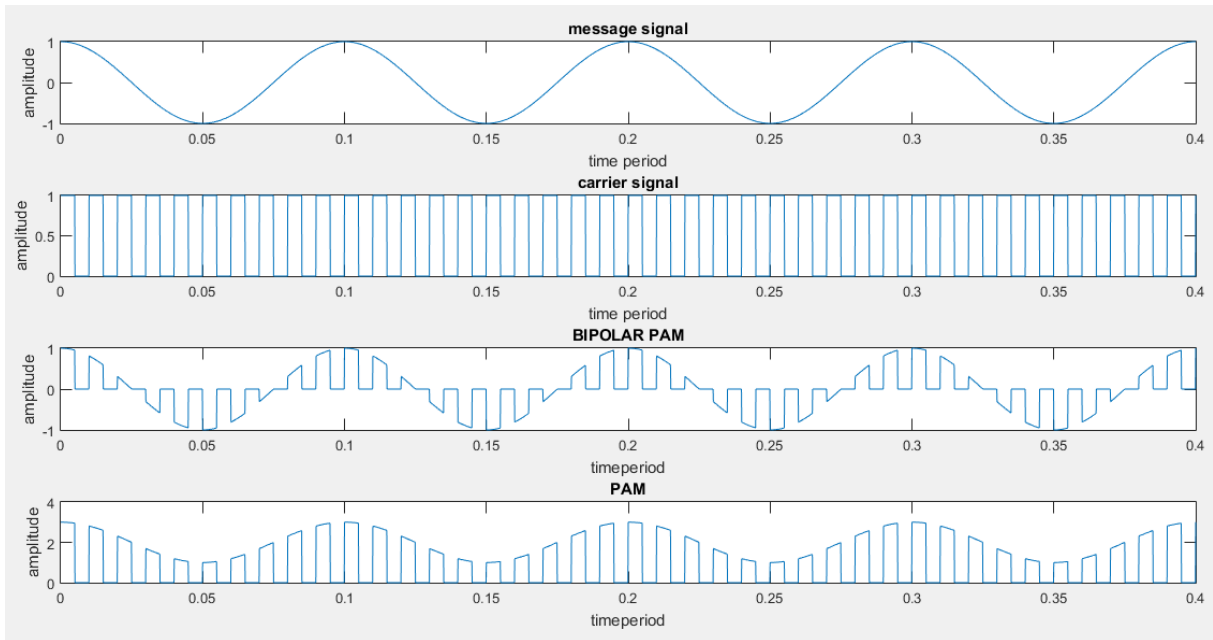


Figure 1: Message, carrier, bipolar PAM and PAM

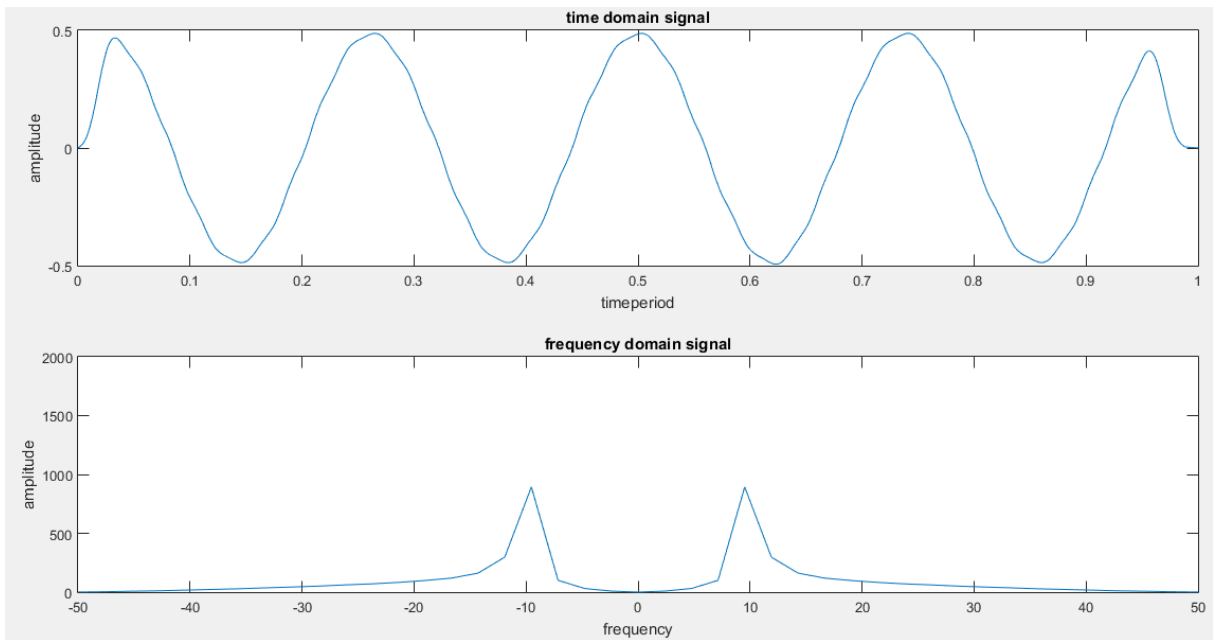


Figure 2: Time and Frequency domain signals