EXPERIMENT 2

Name : Jacob V Sanoj
SRN : PES1UG20EC083

Date : 27/01/2021

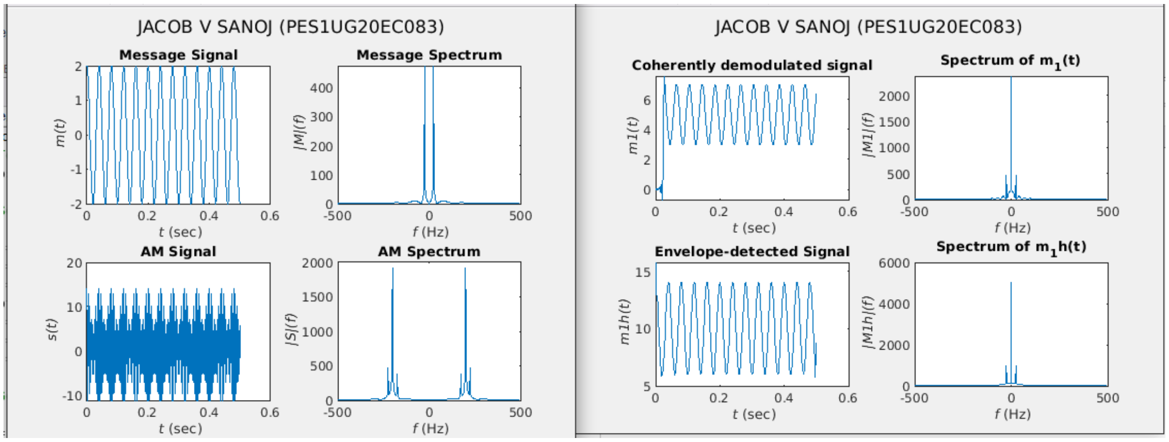
UNDER MODULATION:

Code:

```
% To demonstrate standard AM modulation
close all
% Set the parameters for the simulation
Ts = 0.001; % Time resolution
Ac = 10;    % Carrier amplitude
fc = 200;   % Carrier frequency in Hz
Nlpf = 50;  % Length of the FIR LPF at the receiver, for coherent demodulation
Bm = 150;   % Bandwidth of the FIR LPF
ka = 0.2;   % Amplitude sensitivity
fm = 25;    % Modulating frequency
% Generating m(t)
t = [0:Ts:0.5];             % The time range for displaying the signals
mt = 2*cos(2*pi*fm*t);
% Multiplying m(t) with the carrier to generate s(t)
st = (1 + ka * mt) .* (Ac * cos(2*pi*fc*t));
% To compute and plot the spectra of m(t) and s(t). We will use the fft command
to compute the spectrum
Nfft = length(t);                 % Find the length of m(t)
Nfft = 2^(ceil(log2(Nfft)));      % Set the FFT length as the next higher power
of 2
f = ((-Nfft/2):(Nfft/2)-1)/(Nfft*Ts); % Set the frequency scale, to display the
FFT output in terms of analog frequency (in Hz)
Mf = fft(mt,Nfft);                % Spectrum of m(t)
Mf = fftshift(Mf);                % Circularly shift the FFT output to bring the
dc component to the center, so that the spectrum plot will be from -pi to pi
Sf = fft(st,Nfft);                % Spectrum of s(t)
Sf = fftshift(Sf);
% Demodulation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Coherent demodulation:
% Multiply s(t) with the local carrier to generate v(t)
vt = st .* cos(2*pi*fc*t);
% We need to apply an LPF on v(t) to obtain m1(t), the recovered version of
% m(t). We will use an FIR LPF for this purpose.
% Design the LPF using fir1. The cutoff frequency is to be specified as a
% fraction of the sampling frequency (fs = 1/Ts)
h = fir1(Nlpf, 2*Bm*Ts);
% Filter v(t) with the LPF
```

```
m1t = filter(h,1,vt);
% Its spectrum:
M1f = fft(m1t,Nfft);
M1f = fftshift(M1f);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Envelope detection:
% Here, we use "hilbert", which returns the pre-envelope of the signal. Its
% absolute value is the natural envelope of the signal
m1h = hilbert(st);
m1ht = abs(m1h);
% Spectrum:
M1hf = fft(m1ht,Nfft);
M1hf = fftshift(M1hf);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Plot the results
figure;
subplot(221)
plot(t,mt);
title('Message Signal');
xlabel('{\it t} (sec)');
ylabel('{\it m(t)}');
subplot(223)
plot(t,st);
title('AM Signal');
xlabel('{\it t} (sec)');
ylabel('{\it s(t)}');
subplot(222)
plot(f,abs(Mf));
title('Message Spectrum');
xlabel('{\it f} (Hz)');
ylabel('{\it |M|(f)}');
subplot(224)
plot(f,abs(Sf));
title('AM Spectrum');
xlabel('{\it f} (Hz)');
ylabel('{\it |S|(f)}');
sgtitle('JACOB V SANOJ (PES1UG20EC083)');
figure;
subplot(221)
plot(t,m1t);
title('Coherently demodulated signal');
xlabel('{\it t} (sec)');
ylabel('{\it m1(t)}');
subplot(223)
plot(t,m1ht);
title('Envelope-detected Signal');
xlabel('{\it t} (sec)');
ylabel('{\it m1h(t)}');
subplot(222)
```

```
plot(f,abs(M1f));
title('Spectrum of m_1(t)');
xlabel('{\it f} (Hz)');
ylabel('{\it |M1|(f)}');
subplot(224)
plot(f,abs(M1hf));
title('Spectrum of m_1h(t)');
xlabel('{\it f} (Hz)');
ylabel('{\it |M1h|(f)}');
sgtitle('JACOB V SANOJ (PES1UG20EC083)');
```



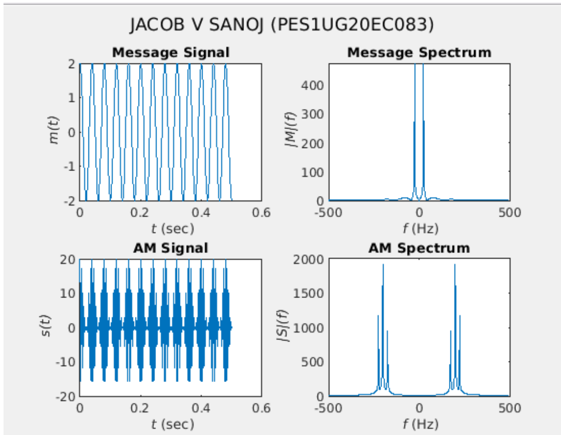
RIGHT MODULATION

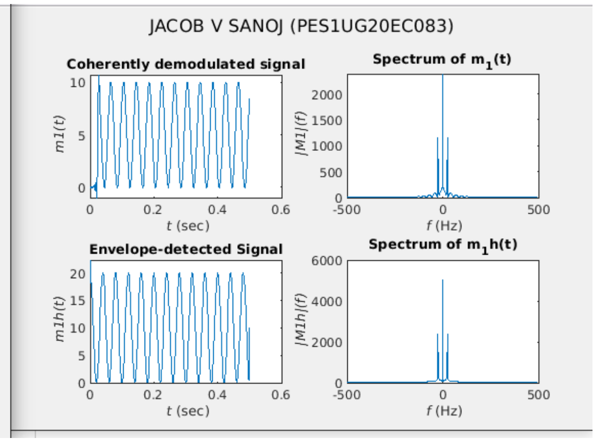
Code:

```
% To demonstrate standard AM modulation
close all
% Set the parameters for the simulation
Ts = 0.001; % Time resolution
Ac = 10;    % Carrier amplitude
fc = 200;   % Carrier frequency in Hz
Nlpf = 50;  % Length of the FIR LPF at the receiver, for coherent demodulation
Bm = 150;   % Bandwidth of the FIR LPF
ka = 0.5;   % Amplitude sensitivity
fm = 25;    % Modulating frequency
% Generating m(t)
t = [0:Ts:0.5];          % The time range for displaying the signals
mt = 2*cos(2*pi*fm*t);
% Multiplying m(t) with the carrier to generate s(t)
st = (1 + ka * mt) .* (Ac * cos(2*pi*fc*t));
% To compute and plot the spectra of m(t) and s(t). We will use the fft command
to compute the spectrum
Nfft = length(t);                 % Find the length of m(t)
Nfft = 2^(ceil(log2(Nfft)));      % Set the FFT length as the next higher power
of 2
f = ((-Nfft/2):(Nfft/2)-1)/(Nfft*Ts); % Set the frequency scale, to display the
FFT output in terms of analog frequency (in Hz)
Mf = fft(mt,Nfft);                % Spectrum of m(t)
Mf = fftshift(Mf);                % Circularly shift the FFT output to bring the
dc component to the center, so that the spectrum plot will be from -pi to pi
Sf = fft(st,Nfft);                % Spectrum of s(t)
Sf = fftshift(Sf);
% Demodulation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Coherent demodulation:
% Multiply s(t) with the local carrier to generate v(t)
vt = st .* cos(2*pi*fc*t);
% We need to apply an LPF on v(t) to obtain m1(t), the recovered version of
% m(t). We will use an FIR LPF for this purpose.
% Design the LPF using fir1. The cutoff frequency is to be specified as a
% fraction of the sampling frequency (fs = 1/Ts)
h = fir1(Nlpf, 2*Bm*Ts);
% Filter v(t) with the LPF
m1t = filter(h,1,vt);
% Its spectrum:
M1f = fft(m1t,Nfft);
M1f = fftshift(M1f);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Envelope detection:
% Here, we use "hilbert", which returns the pre-envelope of the signal. Its
```

```
% absolute value is the natural envelope of the signal
m1h = hilbert(st);
m1ht = abs(m1h);
% Spectrum:
M1hf = fft(m1ht,Nfft);
M1hf = fftshift(M1hf);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Plot the results
figure;
subplot(221)
plot(t,mt);
title('Message Signal');
xlabel('{\it t} (sec)');
ylabel('{\it m(t)}');
subplot(223)
plot(t,st);
title('AM Signal');
xlabel('{\it t} (sec)');
ylabel('{\it s(t)}');
subplot(222)
plot(f,abs(Mf));
title('Message Spectrum');
xlabel('{\it f} (Hz)');
ylabel('{\it |M|(f)}');
subplot(224)
plot(f,abs(Sf));
title('AM Spectrum');
xlabel('{\it f} (Hz)');
ylabel('{\it |S|(f)}');
sgtitle('JACOB V SANOJ (PES1UG20EC083)');
figure;
subplot(221)
plot(t,m1t);
title('Coherently demodulated signal');
xlabel('{\it t} (sec)');
ylabel('{\it m1(t)}');
subplot(223)
plot(t,m1ht);
title('Envelope-detected Signal');
xlabel('{\it t} (sec)');
ylabel('{\it m1h(t)}');
subplot(222)
plot(f,abs(M1f));
title('Spectrum of m_1(t)');
xlabel('{\it f} (Hz)');
ylabel('{\it |M1|(f)}');
subplot(224)
plot(f,abs(M1hf));
title('Spectrum of m_1h(t)');
```

```
xlabel('{\it f} (Hz)');
ylabel('{\it |M1h|(f)}');
sgtitle('JACOB V SANOJ (PES1UG20EC083)');
```





OVER MODULATION

Code:

```
% To demonstrate standard AM modulation
close all
% Set the parameters for the simulation
Ts = 0.001; % Time resolution
Ac = 10;    % Carrier amplitude
fc = 200;   % Carrier frequency in Hz
Nlpf = 50;  % Length of the FIR LPF at the receiver, for coherent demodulation
Bm = 150;   % Bandwidth of the FIR LPF
ka = 0.9;   % Amplitude sensitivity
fm = 25;    % Modulating frequency
% Generating m(t)
t = [0:Ts:0.5];            % The time range for displaying the signals
mt = 2*cos(2*pi*fm*t);
% Multiplying m(t) with the carrier to generate s(t)
st = (1 + ka * mt) .* (Ac * cos(2*pi*fc*t));
% To compute and plot the spectra of m(t) and s(t). We will use the fft command
to compute the spectrum
Nfft = length(t);                 % Find the length of m(t)
Nfft = 2^(ceil(log2(Nfft)));      % Set the FFT length as the next higher power
of 2
```

```
f = ((-Nfft/2):(Nfft/2)-1)/(Nfft*Ts); % Set the frequency scale, to display the
FFT output in terms of analog frequency (in Hz)
Mf = fft(mt,Nfft);              % Spectrum of m(t)
Mf = fftshift(Mf);              % Circularly shift the FFT output to bring the
dc component to the center, so that the spectrum plot will be from -pi to pi
Sf = fft(st,Nfft);              % Spectrum of s(t)
Sf = fftshift(Sf);
% Demodulation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Coherent demodulation:
% Multiply s(t) with the local carrier to generate v(t)
vt = st .* cos(2*pi*fc*t);
% We need to apply an LPF on v(t) to obtain m1(t), the recovered version of
% m(t). We will use an FIR LPF for this purpose.
% Design the LPF using fir1. The cutoff frequency is to be specified as a
% fraction of the sampling frequency (fs = 1/Ts)
h = fir1(Nlpf, 2*Bm*Ts);
% Filter v(t) with the LPF
m1t = filter(h,1,vt);
% Its spectrum:
M1f = fft(m1t,Nfft);
M1f = fftshift(M1f);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Envelope detection:
% Here, we use "hilbert", which returns the pre-envelope of the signal. Its
% absolute value is the natural envelope of the signal
m1h = hilbert(st);
m1ht = abs(m1h);
% Spectrum:
M1hf = fft(m1ht,Nfft);
M1hf = fftshift(M1hf);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Plot the results
figure;
subplot(221)
plot(t,mt);
title('Message Signal');
xlabel('{\it t} (sec)');
ylabel('{\it m(t)}');
subplot(223)
plot(t,st);
title('AM Signal');
xlabel('{\it t} (sec)');
ylabel('{\it s(t)}');
subplot(222)
plot(f,abs(Mf));
title('Message Spectrum');
xlabel('{\it f} (Hz)');
ylabel('{\it |M|(f)}');
```

```
subplot(224)
plot(f,abs(Sf));
title('AM Spectrum');
xlabel('{\it f} (Hz)');
ylabel('{\it |S|(f)}');
sgtitle('JACOB V SANOJ (PES1UG20EC083)');
figure;
subplot(221)
plot(t,m1t);
title('Coherently demodulated signal');
xlabel('{\it t} (sec)');
ylabel('{\it m1(t)}');
subplot(223)
plot(t,m1ht);
title('Envelope-detected Signal');
xlabel('{\it t} (sec)');
ylabel('{\it m1h(t)}');
subplot(222)
plot(f,abs(M1f));
title('Spectrum of m_1(t)');
xlabel('{\it f} (Hz)');
ylabel('{\it |M1|(f)}');
subplot(224)
plot(f,abs(M1hf));
title('Spectrum of m_1h(t)');
xlabel('{\it f} (Hz)');
ylabel('{\it |M1h|(f)}');
sgtitle('JACOB V SANOJ (PES1UG20EC083)');
```