Faculty of Engineering
Cairo University

Communications Engineering Project Report

Modulation and Demodulation

Submitted by:

Yasmine Ashraf Ghanem

Yasmin Abdullah Nasser

Supervised by:

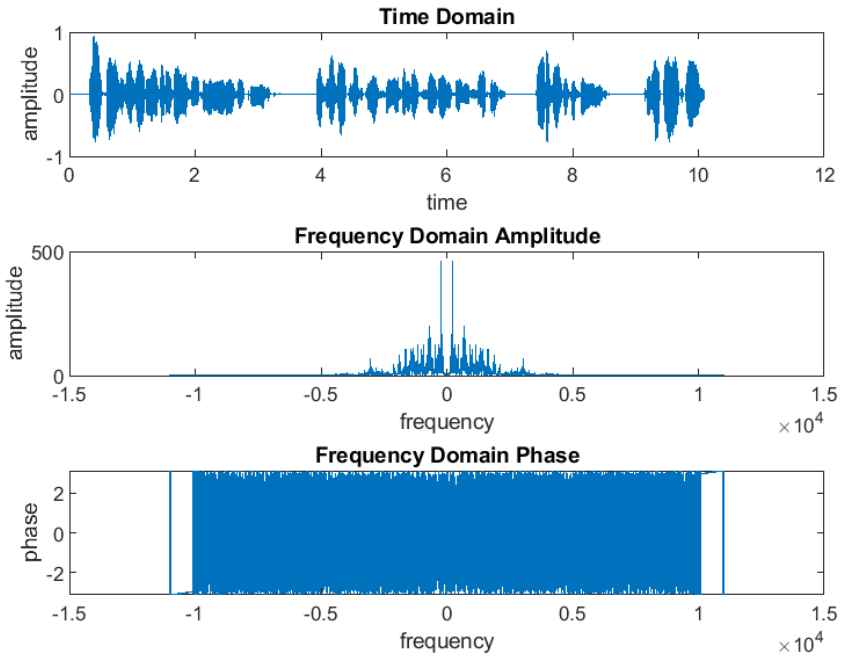
ENG. Alaa Khairalla

TEAM MEMBERS

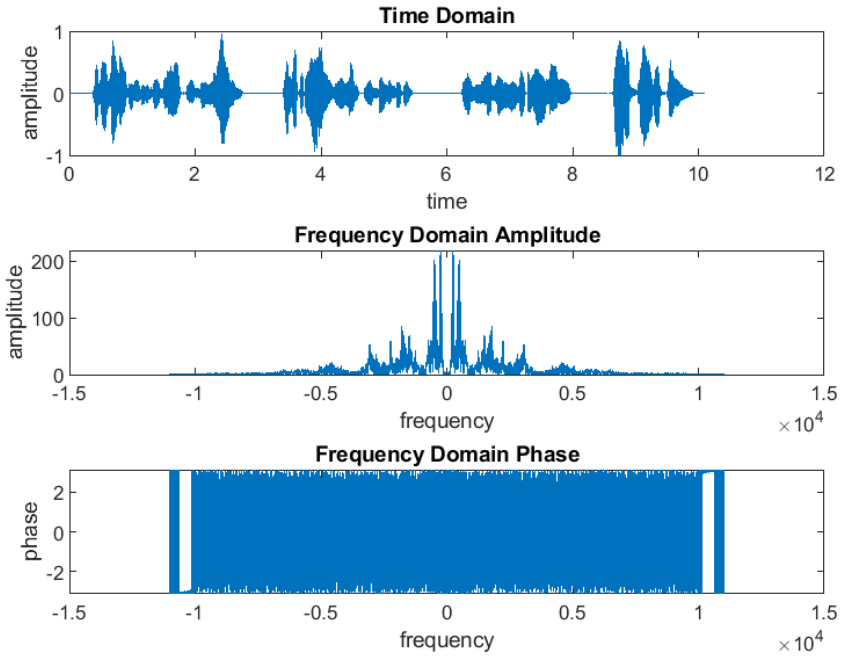
Name	Sec	BN	Email
Yasmine Ashraf Ghanem	2	37	yasmineashrafghanem@gmail.com
Yasmin Abdullah Nasser	2	38	yasminelgendi@gmail.com

SIMULATION RESULTS

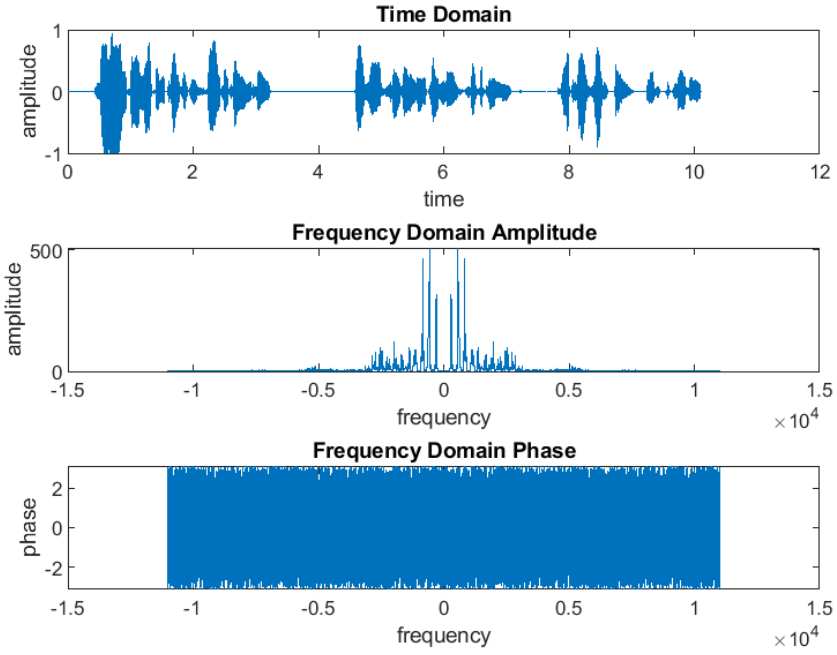
First Speech Signal:



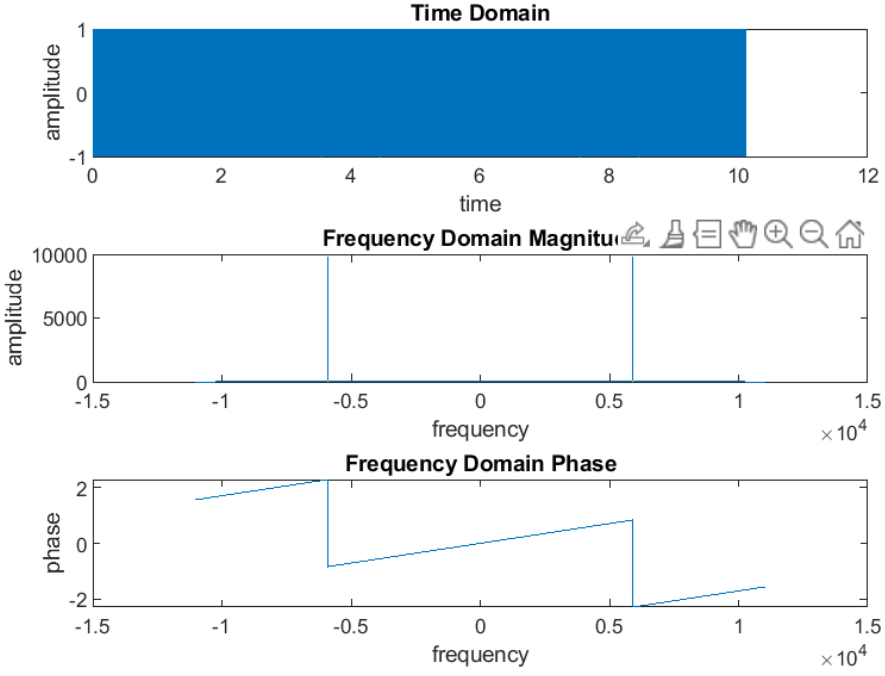
Second Speech Signal:



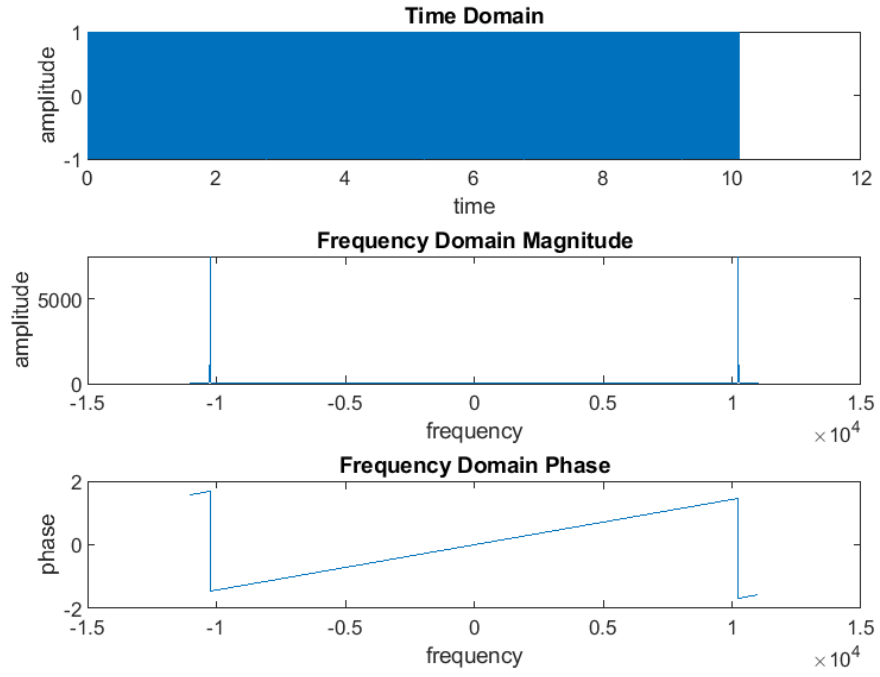
Third Speech Signal:



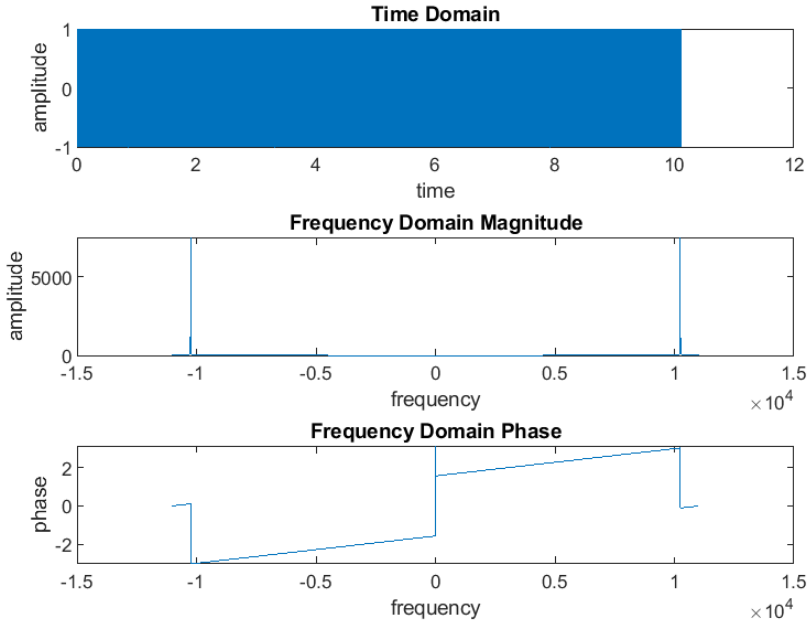
First Carrier Signal: cos($\omega_1 t$)



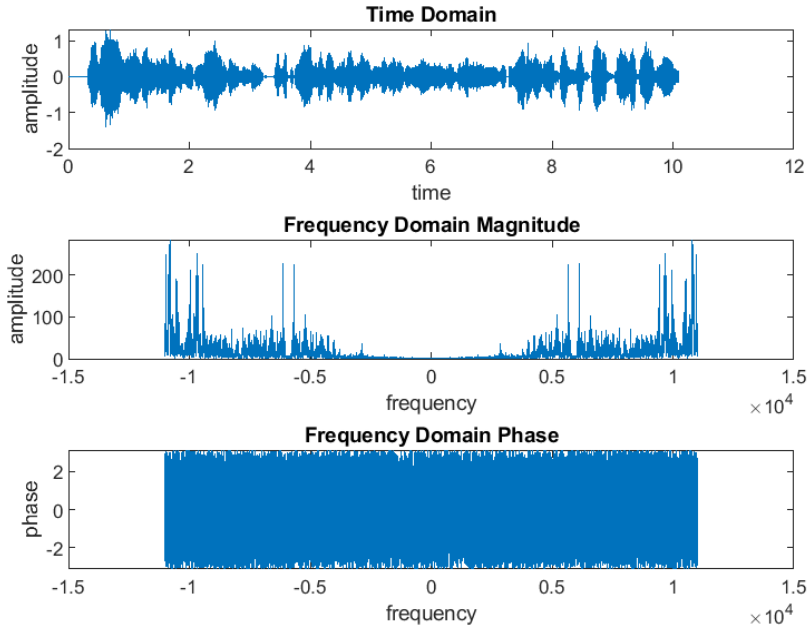
Second Carrier Signal: cos($\omega_2 t$)



Third Carrier Signal: sin($\omega_2 t$)

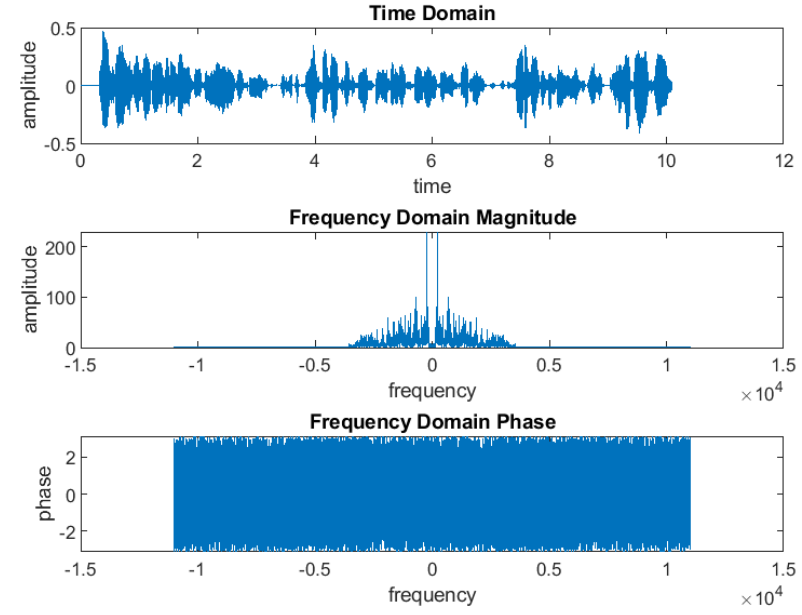


Modulated Signal:

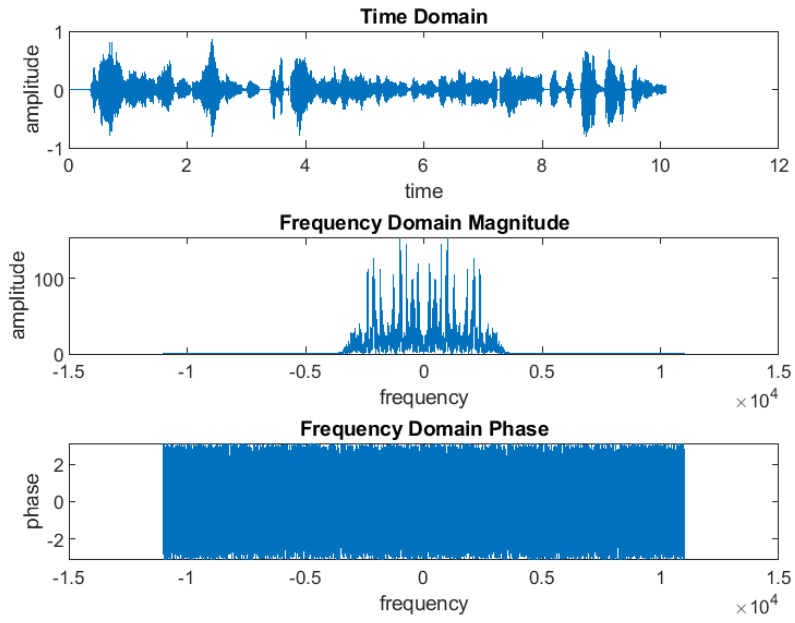


SYNCHRONOUS DEMODULATION:

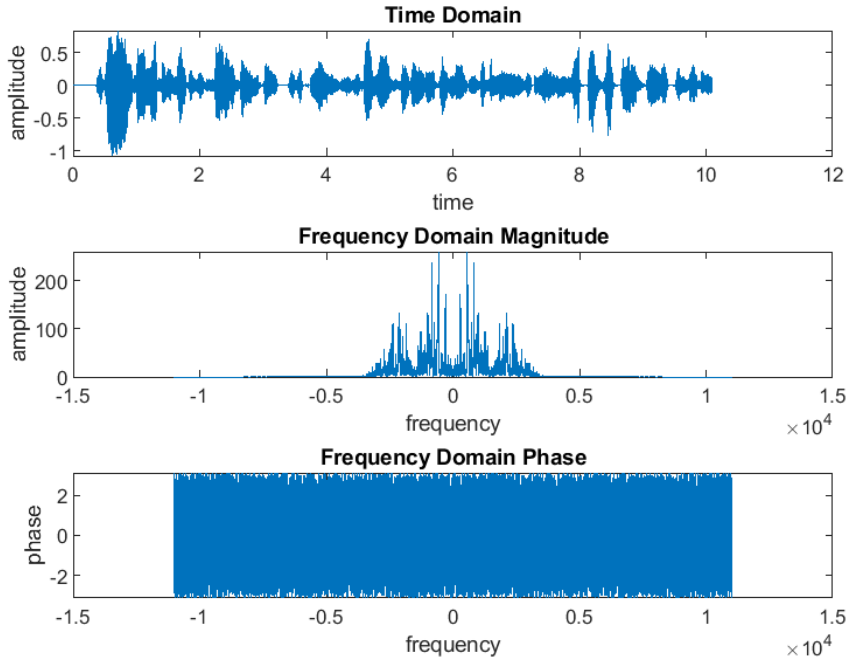
First Demodulated Signal: (first carrier)



Second Demodulated Signal: (third carrier)

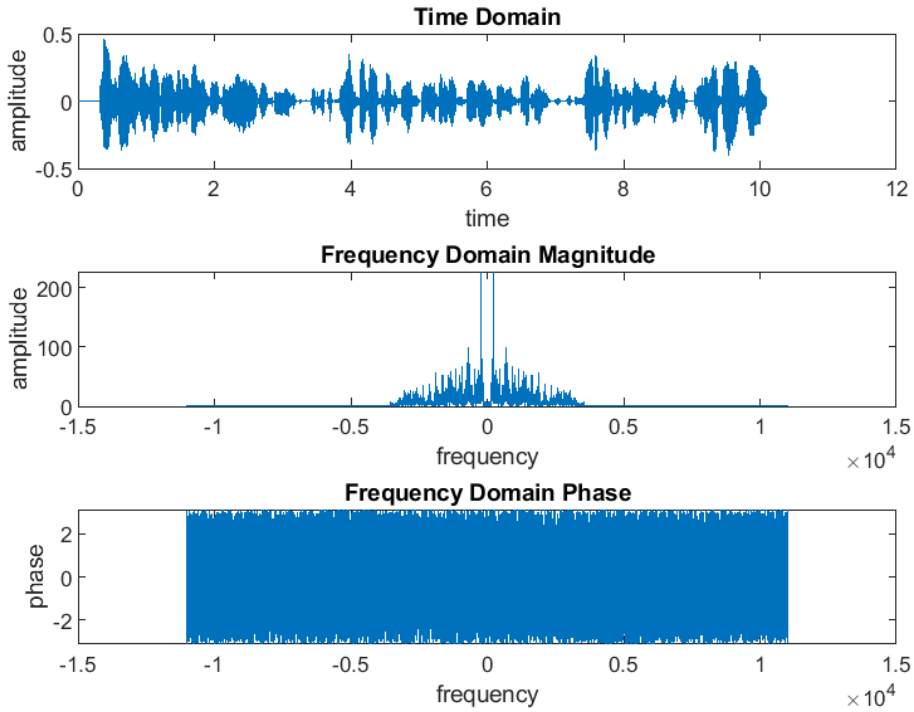


Third Demodulated Signal: (third carrier)

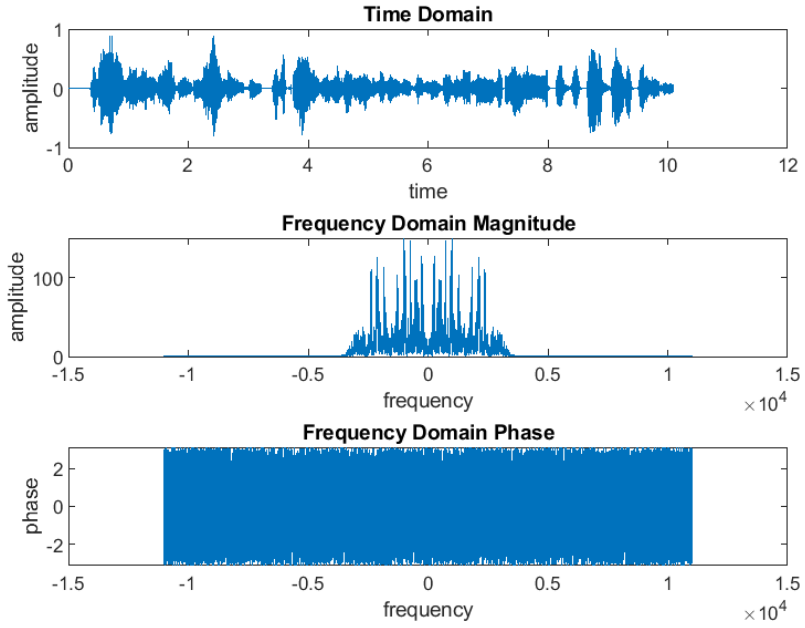


PHASE SHIFT BY 10:

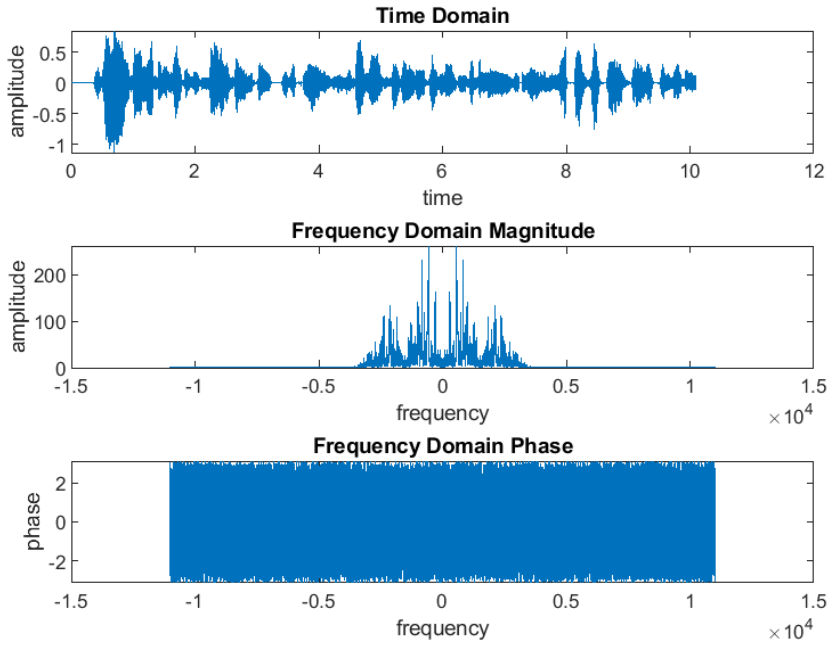
First Demodulated Signal: (first carrier)



Second Demodulated Signal: (second carrier)

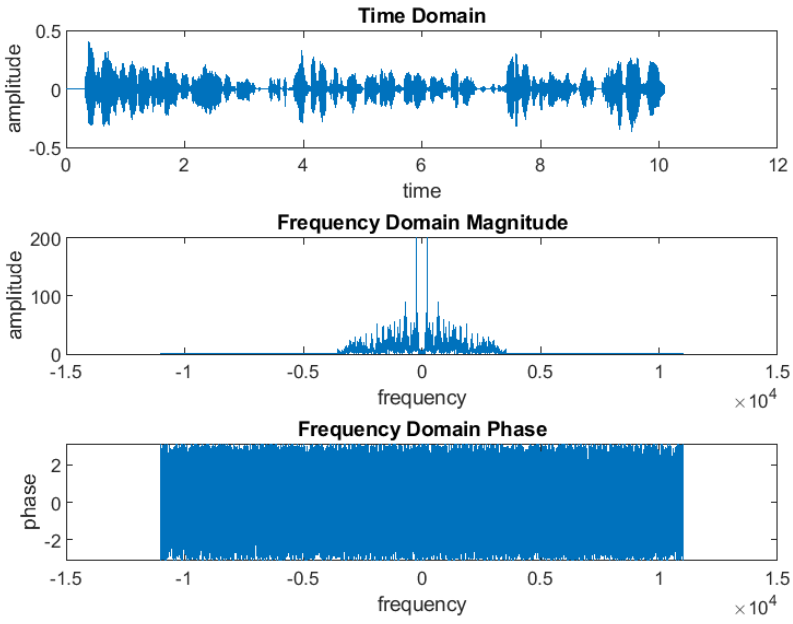


Third Demodulated Signal: (third carrier)

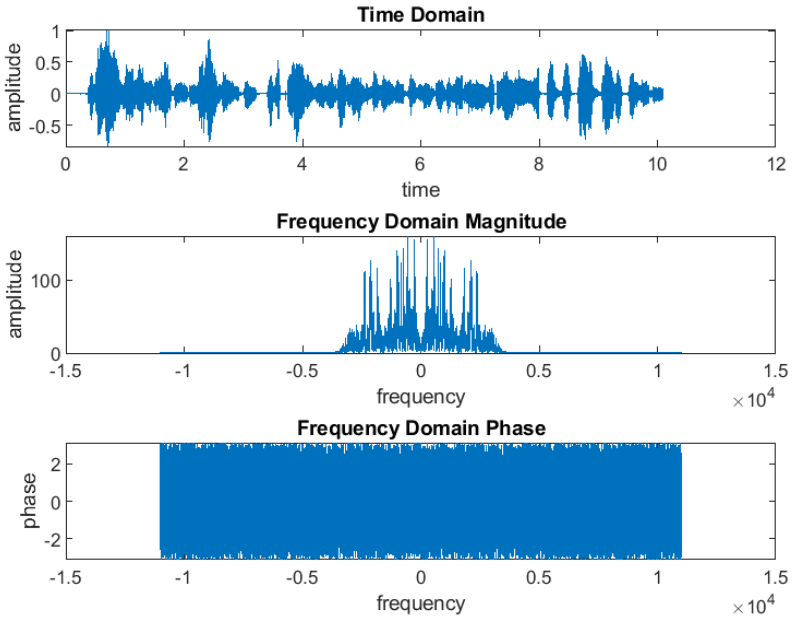


PHASE SHIFT BY 30:

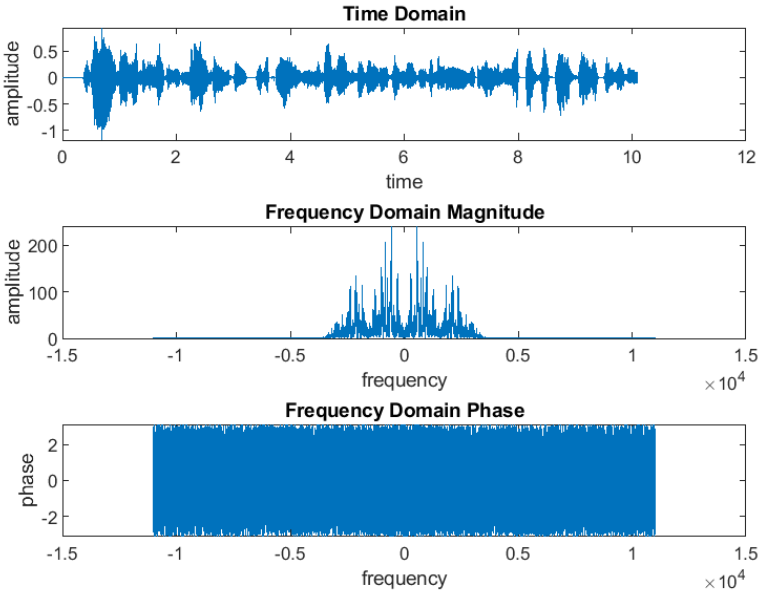
First Demodulated Signal: (first carrier)



Second Demodulated Signal: (second carrier)

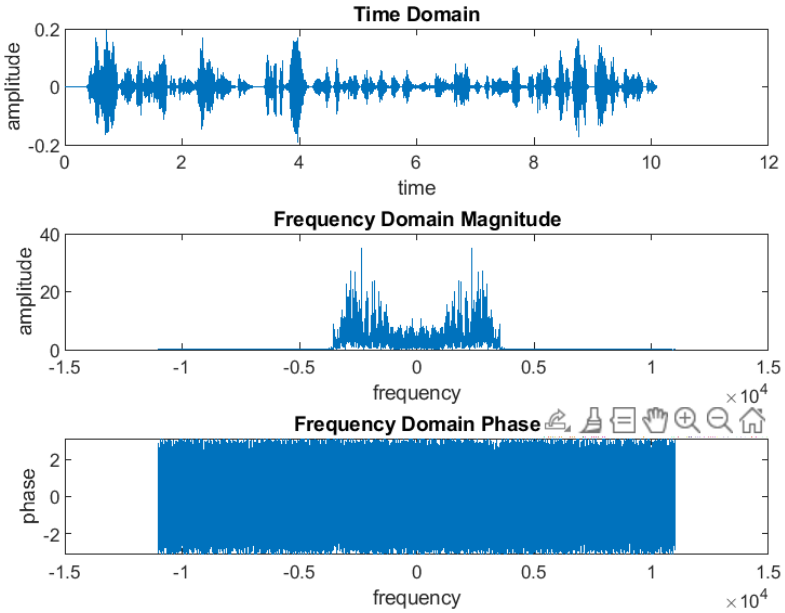


Third Demodulated Signal: (third carrier)

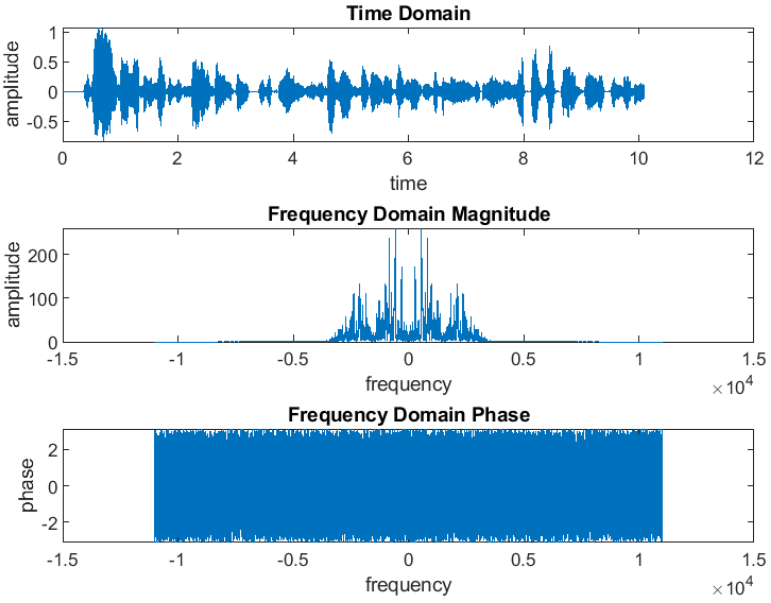


PHASE SHIFT BY 90:

First Demodulated Signal: (first carrier)



Second Demodulated Signal: (second carrier)



Third Demodulated Signal: (third carrier)

CODE

```
clear;
close all;

%Read audios

%read first signal
[signal_1,fs_1] = audioread('audios/original/sample_1.mp3'); %mt: sampled audio, fs: sampling frequency
%time domain
signal_1 = signal_1(:,1);

signal_1_audio_length = length(signal_1);
f1 = -fs_1/2:fs_1/2;
ts1 = 1/fs_1; %sampling interval
t1 = 0:ts1:(signal_1_audio_length-1)*ts1; %time interval of signal
fc1 = (fs_1/2); %carrier frequency

%frequency domain first signal
signal_1_fft_amplitude = abs(fftshift(fft(signal_1,fs_1+1)));   %obtain fourier transform of first signal
signal_1_fft_phase = angle(fftshift(fft(signal_1,fs_1+1)));

%First signal time and frequency domains plots
figure(1);
subplot(3,1,1);
plot(t1, signal_1); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f1,signal_1_fft_amplitude); title('Frequency Domain Amplitude'); xlabel('frequency'); ylabel('amplitude');
subplot(3,1,3);
plot(f1,signal_1_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');


%read second signal
[signal_2,fs_2] = audioread('audios/original/sample_2.mp3'); %mt: sampled audio, fs: sampling frequency
signal_2 = signal_2(:,1);
signal_2_audio_length = length(signal_2);

f2 = -fs_2/2:fs_2/2;
ts2 = 1/fs_2; %sampling interval
t2 = 0:ts2:(signal_2_audio_length-1)*ts2; %time interval of signal
fc2 = (fs_2/2); %carrier frequency

%frequency domain first signal
signal_2_fft_amplitude = abs(fftshift(fft(signal_2,fs_2+1)));   %obtain fourier transform of first signal
signal_2_fft_phase = angle(fftshift(fft(signal_2,fs_2+1)));
```

```
%First signal time and frequency domains plots
figure(2);
subplot(3,1,1);
plot(t2, signal_2); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f2,signal_2_fft_amplitude); title('Frequency Domain Amplitude'); xlabel('frequency'); ylabel('amplitude');
subplot(3,1,3);
plot(f2,signal_2_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%read third signal
[signal_3,fs_3] = audioread('audios/original/sample_3.mp3'); %mt: sampled audio, fs: sampling frequency
signal_3 = signal_3(:,1);
signal_3_audio_length = length(signal_3);

f3 = -fs_3/2:fs_3/2;
ts3 = 1/fs_3; %sampling interval
t3 = 0:ts3:(signal_3_audio_length-1)*ts3; %time interval of signal
fc3 = (fs_3/2); %carrier frequency

%frequency domain first signal
signal_3_fft_amplitude = abs(fftshift(fft(signal_3,fs_3+1)));   %obtain fourier transform of first signal
signal_3_fft_phase = angle(fftshift(fft(signal_3,fs_3+1)));

%First signal time and frequency domains plots
figure(3);
subplot(3,1,1);
plot(t3, signal_3); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f3,signal_3_fft_amplitude); title('Frequency Domain Amplitude'); xlabel('frequency'); ylabel('amplitude');
subplot(3,1,3);
plot(f3,signal_3_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%resample signals
final_fs = 250000;
[P, Q] = rat(final_fs/fs_1);
resampled_signal_1 = resample(signal_1, P, Q);

[P, Q] = rat(final_fs/fs_2);
resampled_signal_2 = resample(signal_2, P, Q);

[P, Q] = rat(final_fs/fs_3);
resampled_signal_3 = resample(signal_3, P, Q);

%signal modulation
% s(t) = x1(t)cos(w1*t) + x2(t)cos(w2*t) + + x3(t)cos(w2*t)
```

```
%generating carriers
w1 = 2*pi*50000;
w2 = 2*pi*100000;

%first signal carrier
time_interval_1 = (0:length(resampled_signal_1) - 1) * (1/final_fs);
carrier_1 = cos(w1*t1); %time domain
carrier_1_fft = abs(fftshift(fft(carrier_1,fs_1+1)));   %obtain fourier transform of carrier magnitude
carrier_1_fft_phase = angle(fftshift(fft(carrier_1,fs_1+1))); %obtain fourier transform of carrier phase

%first carrier time and frequency domains plots
figure(4);
subplot(3,1,1);
plot(t1, carrier_1); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f1,carrier_1_fft); title('Frequency Domain Magnitude'); xlabel('frequency'); ylabel('amplitude');
subplot(3,1,3);
plot(f1,carrier_1_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%second signal carrier
carrier_2 = cos(w2*t2);
carrier_2_fft = abs(fftshift(fft(carrier_2,fs_2+1)));   %obtain fourier transform of carrier magnitude
carrier_2_fft_phase = angle(fftshift(fft(carrier_2,fs_2+1))); %obtain fourier transform of carrier phase

%second carrier time and frequency domains plots
figure(5);
subplot(3,1,1);
plot(t2, carrier_2); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f2,carrier_2_fft); title('Frequency Domain Magnitude'); xlabel('frequency'); ylabel('amplitude');
subplot(3,1,3);
plot(f2,carrier_2_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%third signal carrier
carrier_3 = sin(w2*t3);
carrier_3_fft = abs(fftshift(fft(carrier_3,fs_3+1)));   %obtain fourier transform of carrier magnitude
carrier_3_fft_phase = angle(fftshift(fft(carrier_3,fs_3+1))); %obtain fourier transform of carrier phase

%third carrier time and frequency domains plots
figure(6);
subplot(3,1,1);
plot(t3, carrier_3); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f3,carrier_3_fft); title('Frequency Domain Magnitude'); xlabel('frequency'); ylabel('amplitude');
subplot(3,1,3);
```

```
plot(f3,carrier_3_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%modulate signals
mod_signal_1 = signal_1 .* carrier_1';
mod_signal_2 = signal_2 .* carrier_2';
mod_signal_3 = signal_3 .* carrier_3';

%get length of modulated signals
mod_signal_1_length = length(mod_signal_1);
mod_signal_2_length = length(mod_signal_2);
mod_signal_3_length = length(mod_signal_3);

%get maximum length of all three modulated signals
max_length = max(mod_signal_1_length, max(mod_signal_2_length, mod_signal_3_length));

%adjust all signal length to maximum length
s1 = [mod_signal_1;zeros(max_length-mod_signal_1_length, 1)];
s2 = [mod_signal_2;zeros(max_length-mod_signal_2_length, 1)];
s3 = [mod_signal_3;zeros(max_length-mod_signal_3_length, 1)];

%add the signals to get final modulated signal
final_fs = 22050;
final_ts = 1/final_fs;
final_t = 0:final_ts:(max_length-1)*final_ts;
final_f = -final_fs/2:final_fs/2;

modulated_signal = s1 + s2 + s3; %time domain
modulated_signal_fft = abs(fftshift(fft(modulated_signal,final_fs+1)));   %obtain fourier transform of carrier
magnitude
modulated_signal_fft_phase = angle(fftshift(fft(modulated_signal,final_fs+1))); %obtain fourier transform of
carrier phase

%modulated signal time and frequency domains plots
figure(7);
subplot(3,1,1);
plot(final_t, modulated_signal); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(final_f,modulated_signal_fft); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(final_f,modulated_signal_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%%%
%demodulate signals
%%%
```

```
%demodulate first signal
demodulated_signal_1 = modulated_signal .* carrier_1';
demodulated_signal_1_lpf = lowpass(demodulated_signal_1, 2500, final_fs);
demodulated_signal_1_fft_amplitude = abs(fftshift(fft(demodulated_signal_1_lpf,fs_1+1)));   %obtain fourier
transform of modulating signal
demodulated_signal_1_fft_phase = angle(fftshift(fft(demodulated_signal_1,fs_1+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(8);
subplot(3,1,1);
plot(t1, demodulated_signal_1_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f1, demodulated_signal_1_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f1, demodulated_signal_1_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%demodulate second signal
demodulated_signal_2 = modulated_signal .* carrier_2';
demodulated_signal_2_lpf = lowpass(demodulated_signal_2, 2500, final_fs);
demodulated_signal_2_fft_amplitude = abs(fftshift(fft(demodulated_signal_2_lpf,fs_2+1)));   %obtain fourier
transform of modulating signal
demodulated_signal_2_fft_phase = angle(fftshift(fft(demodulated_signal_2,fs_2+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(9);
subplot(3,1,1);
plot(t2, demodulated_signal_2_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f2, demodulated_signal_2_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f2, demodulated_signal_2_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%demodulate third signal
demodulated_signal_3 = modulated_signal .* carrier_3';
demodulated_signal_3_lpf = lowpass(demodulated_signal_3, 2500, final_fs);
demodulated_signal_3_fft_amplitude = abs(fftshift(fft(demodulated_signal_3_lpf,fs_3+1)));   %obtain fourier
transform of modulating signal
demodulated_signal_3_fft_phase = angle(fftshift(fft(demodulated_signal_3,fs_3+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(10);
```

```
subplot(3,1,1);
plot(t3, demodulated_signal_3_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f3, demodulated_signal_3_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f3, demodulated_signal_3_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

audiowrite('audios/demodulated/demodulated_signal_1.wav', demodulated_signal_1_lpf, fs_1+1);
audiowrite('audios/demodulated/demodulated_signal_2.wav', demodulated_signal_2_lpf, fs_2+1);
audiowrite('audios/demodulated/demodulated_signal_3.wav', demodulated_signal_3_lpf, fs_3+1);

%%%
%phase shift
%%%
%shift all carriers by 10
carrier_1 = cos(w1*final_t + ((10 * pi) / 180));
carrier_2 = cos(w2*final_t + ((10 * pi) / 180));
carrier_3 = sin(w2*final_t + ((10 * pi) / 180));

%demodulate all signals again with the shifted carriers -> 10

%demodulate first signal
demodulated_signal_1 = modulated_signal .* carrier_1';
demodulated_signal_1_lpf = lowpass(demodulated_signal_1, 2500, final_fs);
demodulated_signal_1_fft_amplitude = abs(fftshift(fft(demodulated_signal_1_lpf,fs_1+1)));   %obtain fourier
transform of modulating signal
demodulated_signal_1_fft_phase = angle(fftshift(fft(demodulated_signal_1,fs_1+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(8);
subplot(3,1,1);
plot(t1, demodulated_signal_1_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f1, demodulated_signal_1_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f1, demodulated_signal_1_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%demodulate second signal
demodulated_signal_2 = modulated_signal .* carrier_2';
demodulated_signal_2_lpf = lowpass(demodulated_signal_2, 2500, final_fs);
demodulated_signal_2_fft_amplitude = abs(fftshift(fft(demodulated_signal_2_lpf,fs_2+1)));   %obtain fourier
transform of modulating signal
```

```
demodulated_signal_2_fft_phase = angle(fftshift(fft(demodulated_signal_2,fs_2+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(9);
subplot(3,1,1);
plot(t2, demodulated_signal_2_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f2, demodulated_signal_2_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f2, demodulated_signal_2_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%demodulate third signal
demodulated_signal_3 = modulated_signal .* carrier_3';
demodulated_signal_3_lpf = lowpass(demodulated_signal_3, 2500, final_fs);
demodulated_signal_3_fft_amplitude = abs(fftshift(fft(demodulated_signal_3_lpf,fs_3+1)));   %obtain fourier
transform of modulating signal
demodulated_signal_3_fft_phase = angle(fftshift(fft(demodulated_signal_3,fs_3+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(10);
subplot(3,1,1);
plot(t3, demodulated_signal_3_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f3, demodulated_signal_3_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f3, demodulated_signal_3_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

audiowrite('audios/demodulated/demodulated_signal_1_10.wav', demodulated_signal_1_lpf, fs_1+1);
audiowrite('audios/demodulated/demodulated_signal_2_10.wav', demodulated_signal_2_lpf, fs_2+1);
audiowrite('audios/demodulated/demodulated_signal_3_10.wav', demodulated_signal_3_lpf, fs_3+1);

%shift all carriers by 30
carrier_1 = cos(w1*final_t + ((30 * pi) / 180));
carrier_2 = cos(w2*final_t + ((30 * pi) / 180));
carrier_3 = sin(w2*final_t + ((30 * pi) / 180));

%demodulate all signals again with the shifted carriers -> 30

%demodulate first signal
demodulated_signal_1 = modulated_signal .* carrier_1';
demodulated_signal_1_lpf = lowpass(demodulated_signal_1, 2500, final_fs);
```

```
demodulated_signal_1_fft_amplitude = abs(fftshift(fft(demodulated_signal_1_lpf,fs_1+1)));   %obtain fourier
transform of modulating signal
demodulated_signal_1_fft_phase = angle(fftshift(fft(demodulated_signal_1,fs_1+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(8);
subplot(3,1,1);
plot(t1, demodulated_signal_1_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f1, demodulated_signal_1_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f1, demodulated_signal_1_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%demodulate second signal
demodulated_signal_2 = modulated_signal .* carrier_2';
demodulated_signal_2_lpf = lowpass(demodulated_signal_2, 2500, final_fs);
demodulated_signal_2_fft_amplitude = abs(fftshift(fft(demodulated_signal_2_lpf,fs_2+1)));   %obtain fourier
transform of modulating signal
demodulated_signal_2_fft_phase = angle(fftshift(fft(demodulated_signal_2,fs_2+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(9);
subplot(3,1,1);
plot(t2, demodulated_signal_2_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f2, demodulated_signal_2_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f2, demodulated_signal_2_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%demodulate third signal
demodulated_signal_3 = modulated_signal .* carrier_3';
demodulated_signal_3_lpf = lowpass(demodulated_signal_3, 2500, final_fs);
demodulated_signal_3_fft_amplitude = abs(fftshift(fft(demodulated_signal_3_lpf,fs_3+1)));   %obtain fourier
transform of modulating signal
demodulated_signal_3_fft_phase = angle(fftshift(fft(demodulated_signal_3,fs_3+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(10);
subplot(3,1,1);
plot(t3, demodulated_signal_3_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
```

```
plot(f3, demodulated_signal_3_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f3, demodulated_signal_3_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

audiowrite('audios/demodulated/demodulated_signal_1_30.wav', demodulated_signal_1_lpf, fs_1+1);
audiowrite('audios/demodulated/demodulated_signal_2_30.wav', demodulated_signal_2_lpf, fs_2+1);
audiowrite('audios/demodulated/demodulated_signal_3_30.wav', demodulated_signal_3_lpf, fs_3+1);

%shift all carriers by 90
carrier_1 = cos(w1*final_t + ((90 * pi) / 180));
carrier_2 = cos(w2*final_t + ((90 * pi) / 180));
carrier_3 = sin(w2*final_t + ((90 * pi) / 180));

%demodulate all signals again with the shifted carriers -> 90

%demodulate first signal
demodulated_signal_1 = modulated_signal .* carrier_1';
demodulated_signal_1_lpf = lowpass(demodulated_signal_1, 2500, final_fs);
demodulated_signal_1_fft_amplitude = abs(fftshift(fft(demodulated_signal_1_lpf,fs_1+1)));   %obtain fourier
transform of modulating signal
demodulated_signal_1_fft_phase = angle(fftshift(fft(demodulated_signal_1,fs_1+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(8);
subplot(3,1,1);
plot(t1, demodulated_signal_1_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f1, demodulated_signal_1_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f1, demodulated_signal_1_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%demodulate second signal
demodulated_signal_2 = modulated_signal .* carrier_2';
demodulated_signal_2_lpf = lowpass(demodulated_signal_2, 2500, final_fs);
demodulated_signal_2_fft_amplitude = abs(fftshift(fft(demodulated_signal_2_lpf,fs_2+1)));   %obtain fourier
transform of modulating signal
demodulated_signal_2_fft_phase = angle(fftshift(fft(demodulated_signal_2,fs_2+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(9);
subplot(3,1,1);
plot(t2, demodulated_signal_2_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
```

```
subplot(3,1,2);
plot(f2, demodulated_signal_2_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f2, demodulated_signal_2_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

%demodulate third signal
demodulated_signal_3 = modulated_signal .* carrier_3';
demodulated_signal_3_lpf = lowpass(demodulated_signal_3, 2500, final_fs);
demodulated_signal_3_fft_amplitude = abs(fftshift(fft(demodulated_signal_3_lpf,fs_3+1)));   %obtain fourier
transform of modulating signal
demodulated_signal_3_fft_phase = angle(fftshift(fft(demodulated_signal_3,fs_3+1)));   %obtain fourier
transform of modulating signal

%plot demodulated signal in frequency domain
figure(10);
subplot(3,1,1);
plot(t3, demodulated_signal_3_lpf); title('Time Domain'); xlabel('time'); ylabel('amplitude');
subplot(3,1,2);
plot(f3, demodulated_signal_3_fft_amplitude); title('Frequency Domain Magnitude'); xlabel('frequency');
ylabel('amplitude');
subplot(3,1,3);
plot(f3, demodulated_signal_3_fft_phase) ;title('Frequency Domain Phase'); xlabel('frequency'); ylabel('phase');

audiowrite('audios/demodulated/demodulated_signal_1_90.wav', demodulated_signal_1_lpf, fs_1+1);
audiowrite('audios/demodulated/demodulated_signal_2_90.wav', demodulated_signal_2_lpf, fs_2+1);
audiowrite('audios/demodulated/demodulated_signal_3_90.wav', demodulated_signal_3_lpf, fs_3+1);
```

COMMENTS

(2) Perform synchronous demodulation to restore the three signals.

The demodulated signal resembled the original speech signal by a high percentage but there was a little bit of background noise that wasn't removed during the demodulation.

(3) Perform demodulation three times with phase shifts of 10, 30, 90 degrees for both carriers.

As the phase shift increases for each carrier, the demodulated signal contains more noise; The demodulated signal with the carrier that is shifted by 10 is less noisy than with the one shifted by 30, which was less noisy than the one with the same carrier shifted by 90. This applies for all three speech signals.

(4) For $x_1(t)$, perform demodulation two times with a local carrier frequency that is different by 2 Hz and 10 Hz from its carrier frequency.