LAB #09

Modeling Frequency Division Multiplexing/DE-multiplexing

Fall 2023

CSE-402L Digital Signal Processing Lab

Submitted by: MUHAMMAD SADEEQ

Registration No.: 21PWCSE2028

Section: C

"On my honor, as a student of the University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work"

Submitted to:

Dr. Yasir Saleem Afridi

(7 Jan 2024)

Department of Computer systems engineering
University of Engineering and Technology,
Peshawar

CODE:

```
close all;
clear all;

bandwidth = 4000;
guard_band = 300;
signal_to_noise_ratio = 20;
ssb_modulation = 1;

carrier_freq1 = bandwidth * 3;
carrier_freq2 = bandwidth * 4;
carrier_freq3 = bandwidth * 5;

sampling_freq = carrier_freq3 * 2 + 5000;
cutoff_freq = 2500;

show_graphics = 1;
play_sound = 1;

[B,A] = butter(3,cutoff_freq/(sampling_freq/2));
low_pass_filter = @(signal) filter(B,A,signal);

[C1,D1] = butter(2,[bandwidth*2+guard_band,bandwidth*3-
guard_band]/(sampling_freq/2));
band_filter3=@(signal) filter(C1,D1,signal);

[C2,D2] = butter(2,[bandwidth*3+guard_band,bandwidth*4-
guard_band]/(sampling_freq/2));
band_filter4=@(signal) filter(C2,D2,signal);

[C3,D3] = butter(2,[bandwidth*4+guard_band,bandwidth*5-
guard_band]/(sampling_freq/2));
band_filter5=@(signal) filter(C3,D3,signal);

signal1 = audioread("Sound1.m4a");
length_signal1 = length(signal1);

signal2 = audioread("Sound2.m4a");
length_signal2 = length(signal2);

signal3 = audioread("Sound3.m4a");
length_signal3 = length(signal3);

beep_sound = audioread("beep.mp3");
beep_player = audioplayer(beep_sound,44100);

min_length = min([length_signal1,length_signal2]);

time = linspace(0,5,min_length);

signal1 = signal1(1:min_length);
signal2 = signal2(1:min_length);
signal3 = signal3(1:min_length);
```

```
flag = input("Step2, the signals are reproduced as they arrive");

if(play_sound>0)
    player1 = audioplayer(signal1,44100);
    playblocking(player1);
    playblocking(beep_player);

    player2 = audioplayer(signal2,44100);
    playblocking(player2);
    playblocking(beep_player);

    player3 = audioplayer(signal3,44100);
    playblocking(player3);
end

flag = input("Step 3 plot the spectra of the signals as they arrive");

if(show_graphics>0)
    figure

    spectrum1 = abs(fft(signal1));
    subplot(3,1,1),plot(spectrum1),grid on;zoom,title('Spectrum of filtered
signal_1');

    spectrum2 = abs(fft(signal2));
    subplot(3,1,2),plot(spectrum2),grid on,zoom,title('Spectrum of filtered
signal_2');

    spectrum3 = abs(fft(signal1));
    subplot(3,1,3),plot(spectrum3),grid on,zoom,title('Spectrum of filtered
signal_3');

end

flag = input('Step 4 reproduce the signals after passing them through the filter');

if(play_sound>0)
    beep_player = audioplayer(beep_sound,44100);

    player1 = audioplayer(signal1,44100);
    playblocking(player1);
    playblocking(beep_player);

    player2 = audioplayer(signal2,44100);
    playblocking(player2);
    playblocking(beep_player);

    player3 = audioplayer(signal1,44100);
    playblocking(player3);
    playblocking(beep_player);

end

flag = input('Step 5 the signals are modulated to different carriers');
```

```
if(ssb_modulation>0)
    modulated_signal1 = ssbmod(signal1,carrier_freq1,sampling_freq);
    modulated_signal2 = ssbmod(signal2,carrier_freq2,sampling_freq);
    modulated_signal3 = ssbmod(signal3,carrier_freq3,sampling_freq);

else
    modulated_signal1 = ammod(signal1,carrier_freq1,sampling_freq);
    modulated_signal2 = ammod(signal2,carrier_freq2,sampling_freq);
    modulated_signal3 = ammod(signal3,carrier_freq3,sampling_freq);
end

if(show_graphics>0)
    figure

    spectrum1 = abs(fft(modulated_signal1));
    subplot(3,1,1),plot(spectrum1),grid on,zoom,title('Spectrum of modulated
signal_1');

    spectrum2 = abs(fft(modulated_signal2));
    subplot(3,1,2),plot(spectrum2),grid on,zoom,title('Spectrum of modulated
signal_2');

    spectrum3 = abs(fft(modulated_signal3));
    subplot(3,1,3),plot(spectrum1),grid on,zoom,title('Spectrum of modulated
signal_3');

end
flag = input('Step 6 the modulated signals are filtered in the defined bands and
added');

filtered_signal1 = band_filter3(modulated_signal1);
filtered_signal2 = band_filter4(modulated_signal2);
filtered_signal3 = band_filter5(modulated_signal3);

complete_signal = filtered_signal1 + filtered_signal2 + filtered_signal3;

if(show_graphics > 0)
    figure
    spectrum1 = abs(fft(filtered_signal1));
    subplot(4,1,1),plot(spectrum1),grid on,zoom,title('Spectrum signal_1 modulated
and filtered');

    spectrum2 = abs(fft(filtered_signal2));
    subplot(4,1,2),plot(spectrum2),grid on,zoom,title('Spectrum signal_2 modulated
and filtered');

    spectrum3 = abs(fft(filtered_signal3));
    subplot(4,1,3),plot(spectrum3),grid on,zoom,title('Spectrum signal_3 modulated
and filtered');


    total_spectrum = abs(fft(complete_signal));
    subplot(4,1,4),plot(total_spectrum),grid on,zoom,title('Summed Spectrum');
end
```

```
if(show_graphics > 0)
    figure

    total_spectrum = abs(fft(complete_signal));
    subplot(2,1,1),plot(total_spectrum),grid on,zoom,title('Full signal spectrum
without noise');
end

flag = input('Step 8 upon arrival each band is filtered');

demod_signal1 = band_filter3(complete_signal);
demod_signal2 = band_filter4(complete_signal);
demod_signal3 = band_filter5(complete_signal);

if(show_graphics > 0)
    figure

    spectrum1 = abs(fft(demod_signal1));
    subplot(3,1,1),plot(spectrum1),grid on,zoom,title('Spectrum signal_1 filtered');

    spectrum2 = abs(fft(demod_signal2));
    subplot(3,1,2),plot(spectrum2),grid on,zoom,title('Spectrum signal_2 filtered');

    spectrum3 = abs(fft(demod_signal3));
    subplot(3,1,3),plot(spectrum3),grid on,zoom,title('Spectrum signal_3 filtered');
end

flag = input('Step 9 each recovered band is demodulated to return the signal to the
baseband frequency');

if(ssb_modulation >0)
    demod_signal1 = ssbdemod(demod_signal1,carrier_freq1,sampling_freq);
    demod_signal2 = ssbdemod(demod_signal2,carrier_freq2,sampling_freq);
    demod_signal3 = ssbdemod(demod_signal3,carrier_freq3,sampling_freq);
else
    demod_signal1 = amdemod(demod_signal1,carrier_freq1,sampling_freq);
    demod_signal2 = amdemod(demod_signal2,carrier_freq2,sampling_freq);
    demod_signal3 = amdemod(demod_signal3,carrier_freq3,sampling_freq);
end

if(show_graphics > 0)
    figure

    spectrum1 = abs(fft(demod_signal1));
    subplot(3,1,1),plot(spectrum1),grid on,zoom,title('Spectrum of demodulated
signal_1');

    spectrum2 = abs(fft(demod_signal2));
    subplot(3,1,2),plot(spectrum2),grid on,zoom,title('Spectrum of demodulated
signal_2');

    spectrum3 = abs(fft(demod_signal3));
    subplot(3,1,3),plot(spectrum3),grid on,zoom,title('Spectrum of demodulated
signal_3');
```

```
end

flag = input('Step 10 the recovered signal is passed through a low pass filter');

demod_signal1 = low_pass_filter(demod_signal1);
demod_signal2 = low_pass_filter(demod_signal2);
demod_signal3 = low_pass_filter(demod_signal3);

if(show_graphics > 0)
    figure

    spectrum1 = abs(fft(demod_signal1));
    subplot(3,1,1),plot(spectrum1),grid on,zoom,title('Spectrum signal_1
demodulated');

    spectrum2 = abs(fft(demod_signal2));
    subplot(3,1,2),plot(spectrum2),grid on,zoom,title('Spectrum signal_2
demodulated');

    spectrum3 = abs(fft(demod_signal3));
    subplot(3,1,3),plot(spectrum3),grid on,zoom,title('Spectrum signal_3
demodulated');

end

flag = input('Step 11 play the reproduced signal after transmission');

player4 = audioplayer(demod_signal1,44100);
playblocking(player4);
playblocking(beep_player);

player5 = audioplayer(demod_signal2,44100);
playblocking(player5);
playblocking(beep_player);

player6 = audioplayer(demod_signal3,44100);
playblocking(player6);
playblocking(beep_player);
```

Output:

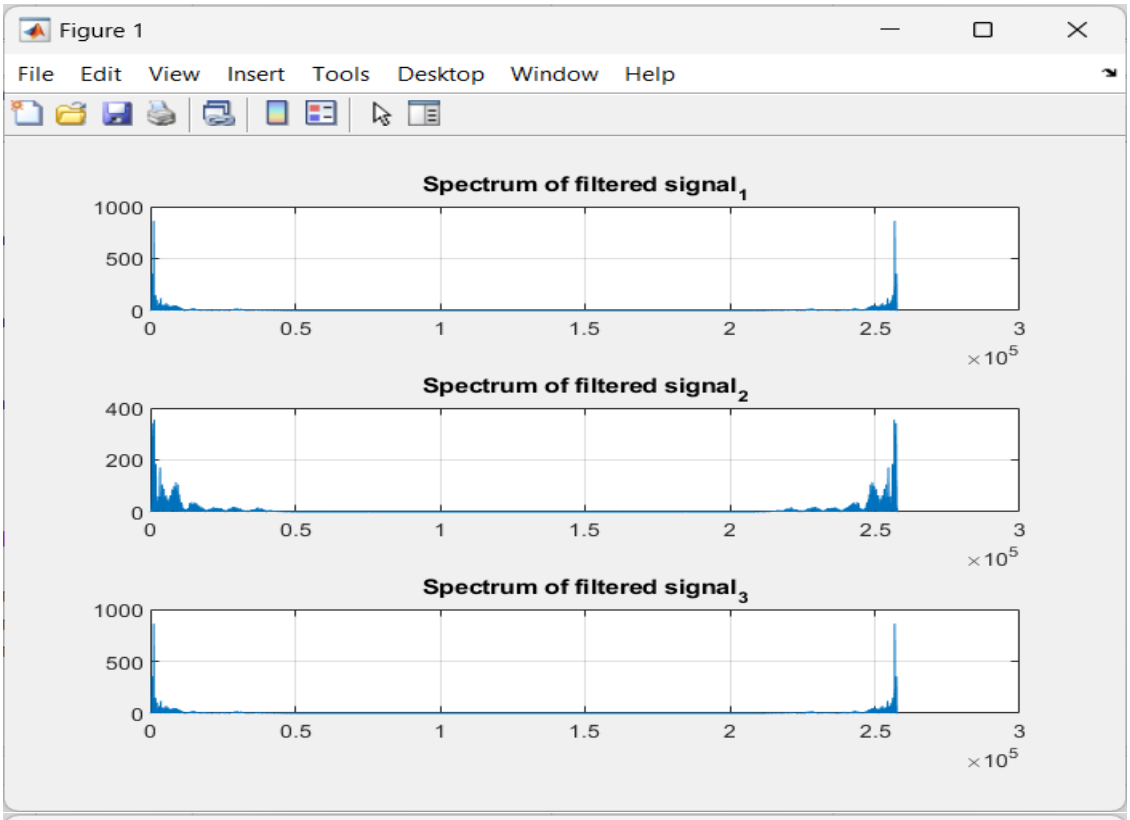
Figure 1
File Edit View Insert Tools Desktop Window Help
Spectrum of filtered signal1
1000
500
0
0 0.5 1 1.5 2 2.5 3
×10^5
Spectrum of filtered signal2
400
200
0
0 0.5 1 1.5 2 2.5 3
×10^5
Spectrum of filtered signal3
1000
500
0
0 0.5 1 1.5 2 2.5 3
×10^5

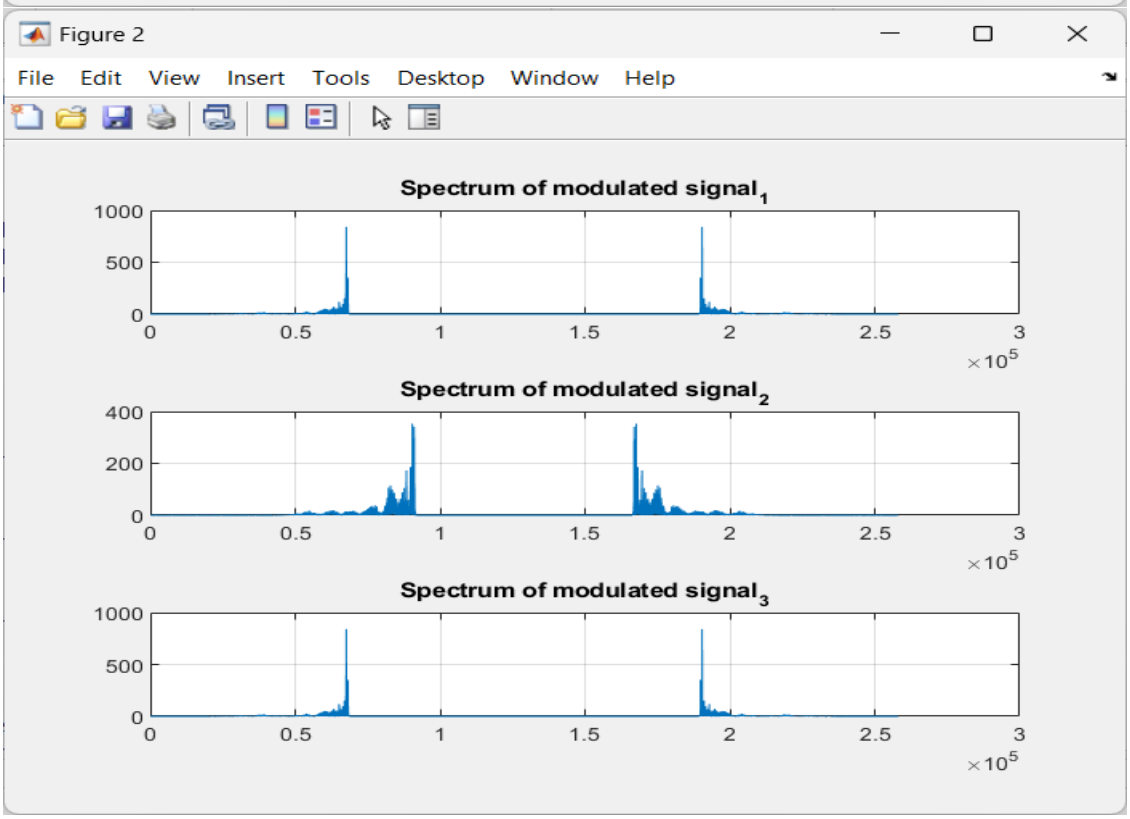
Figure 2
File Edit View Insert Tools Desktop Window Help
Spectrum of modulated signal1
1000
500
0
0 0.5 1 1.5 2 2.5 3
×10^5
Spectrum of modulated signal2
400
200
0
0 0.5 1 1.5 2 2.5 3
×10^5
Spectrum of modulated signal3
1000
500
0
0 0.5 1 1.5 2 2.5 3
×10^5

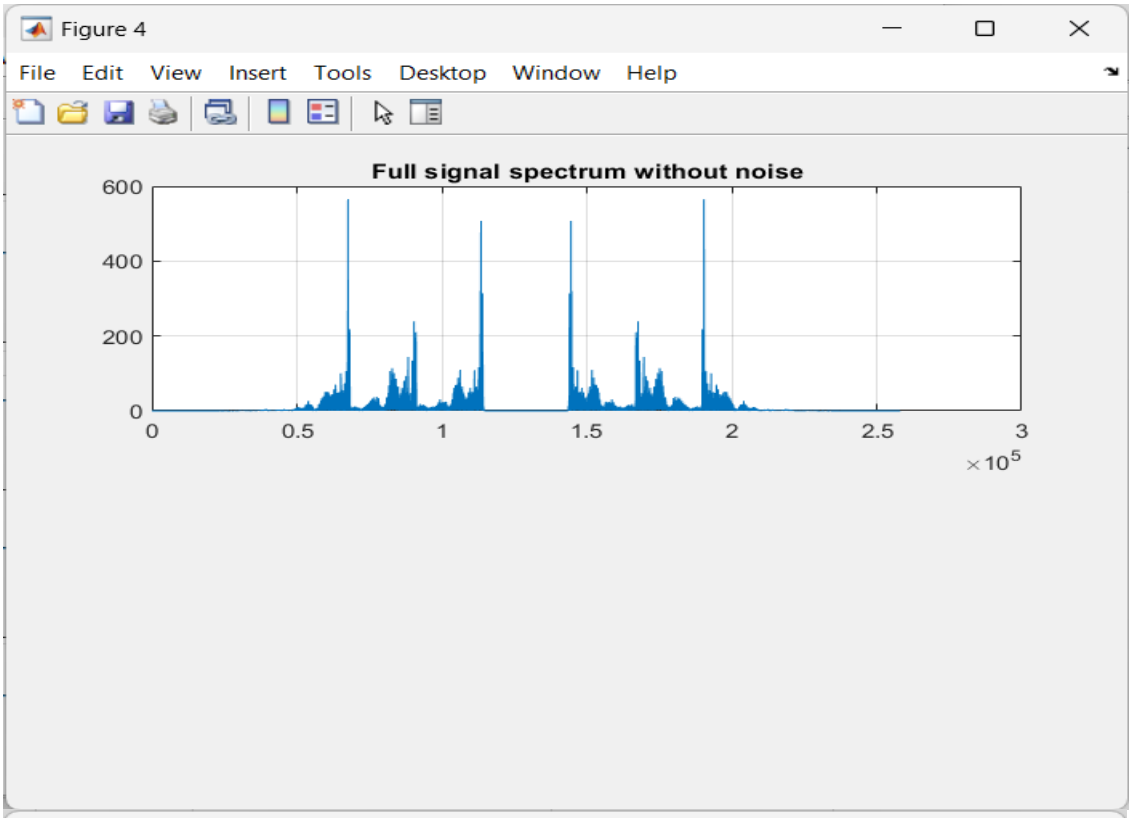
Figure 4
File Edit View Insert Tools Desktop Window Help
Full signal spectrum without noise
600
400
200
0
0 0.5 1 1.5 2 2.5 3
×10^5

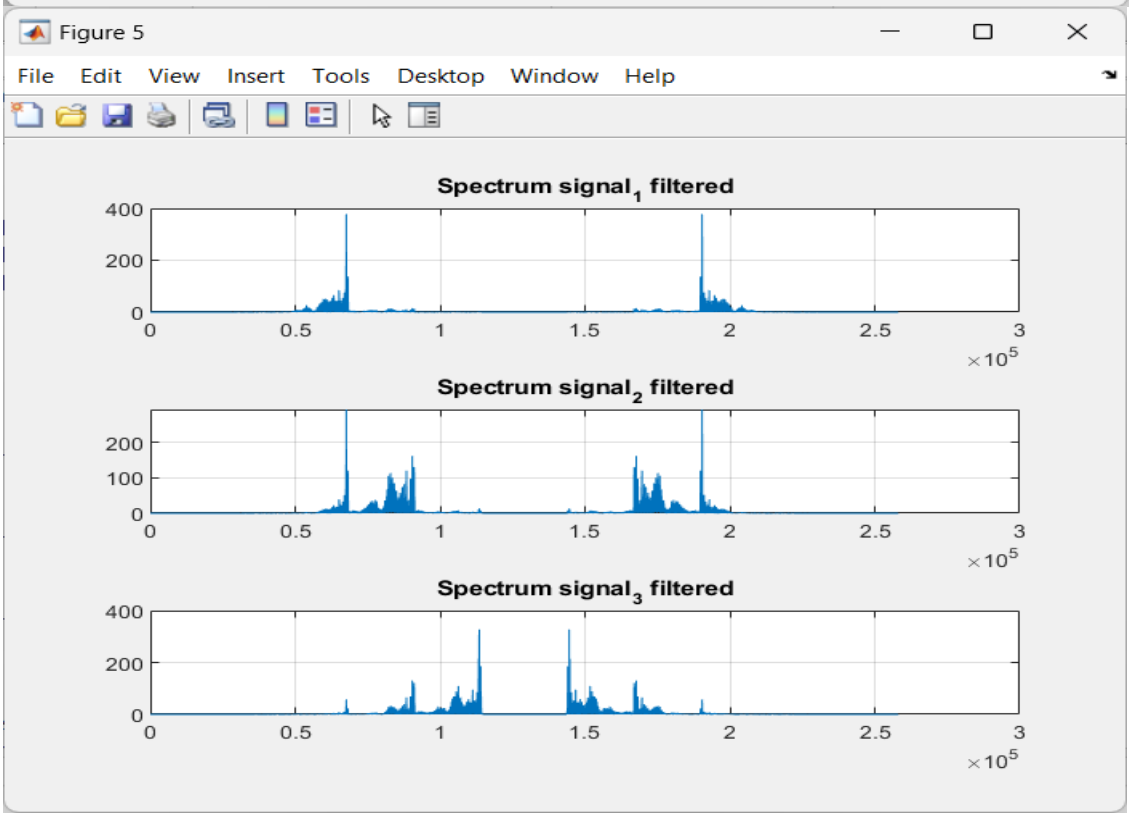
Figure 5
File Edit View Insert Tools Desktop Window Help
Spectrum signal1 filtered
400
200
0
0 0.5 1 1.5 2 2.5 3
×10^5
Spectrum signal2 filtered
200
100
0
0 0.5 1 1.5 2 2.5 3
×10^5
Spectrum signal3 filtered
400
200
0
0 0.5 1 1.5 2 2.5 3
×10^5

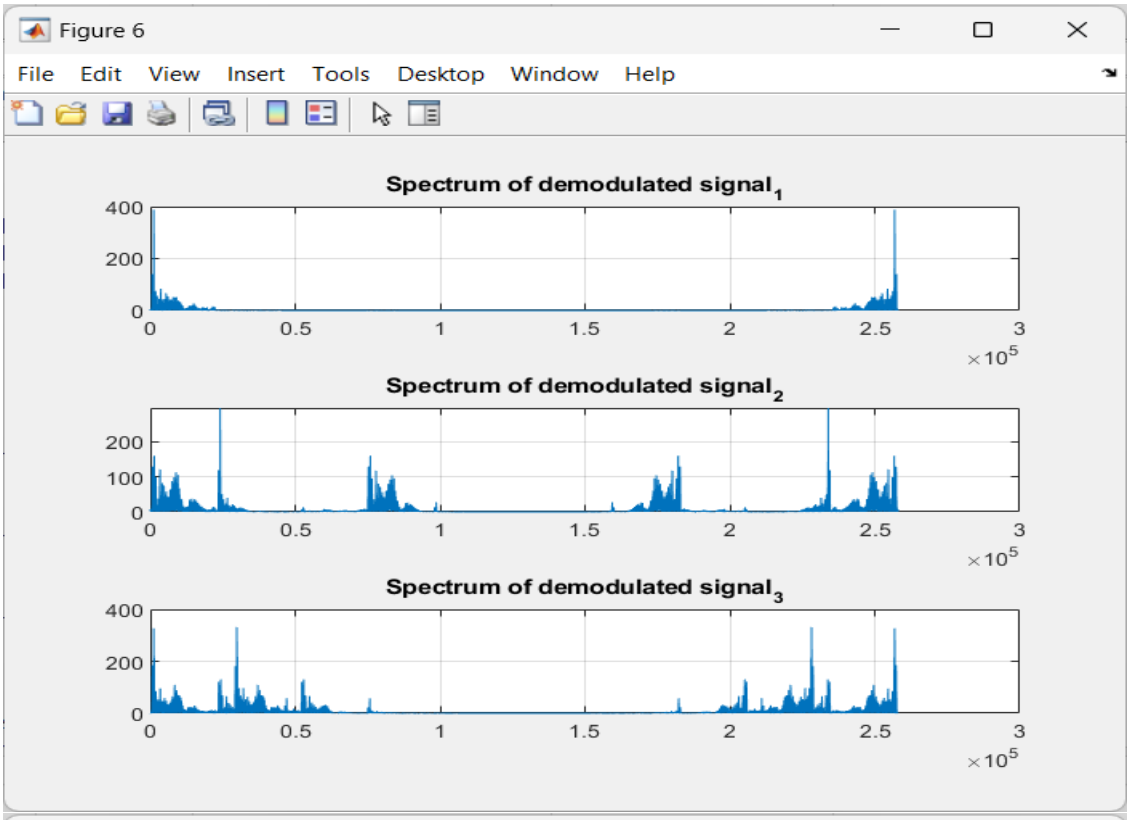
Figure 6
File Edit View Insert Tools Desktop Window Help
Spectrum of demodulated signal1
Spectrum of demodulated signal2
Spectrum of demodulated signal3
×10^5

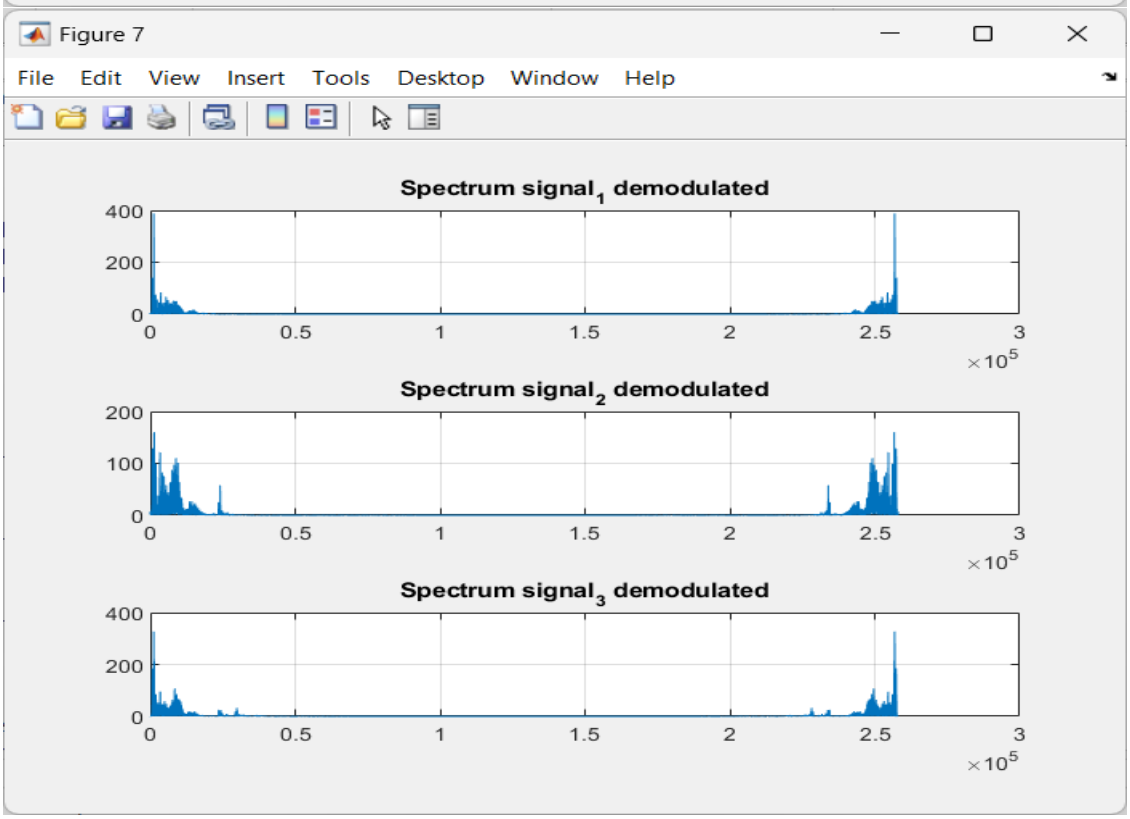
Figure 7
File Edit View Insert Tools Desktop Window Help
Spectrum signal1 demodulated
Spectrum signal2 demodulated
Spectrum signal3 demodulated
×10^5