Cairo University

Faculty of Engineering

Electronics and Electrical Communications Engineering Department

Third Year

Analog Communications

Term Project

MATLAB implementation of a superheterodyne receiver

Student Name: فتحى مصطفى فتحى عبدالحميد

Sec: 3 BN: 19 ID: 9210808

Student Name: محمود حمدى جابر عبدالتواب

Sec: 4 BN:10 ID:9211083

Contents

Table of figures

1. ***The transmitter***

This part contains the following tasks

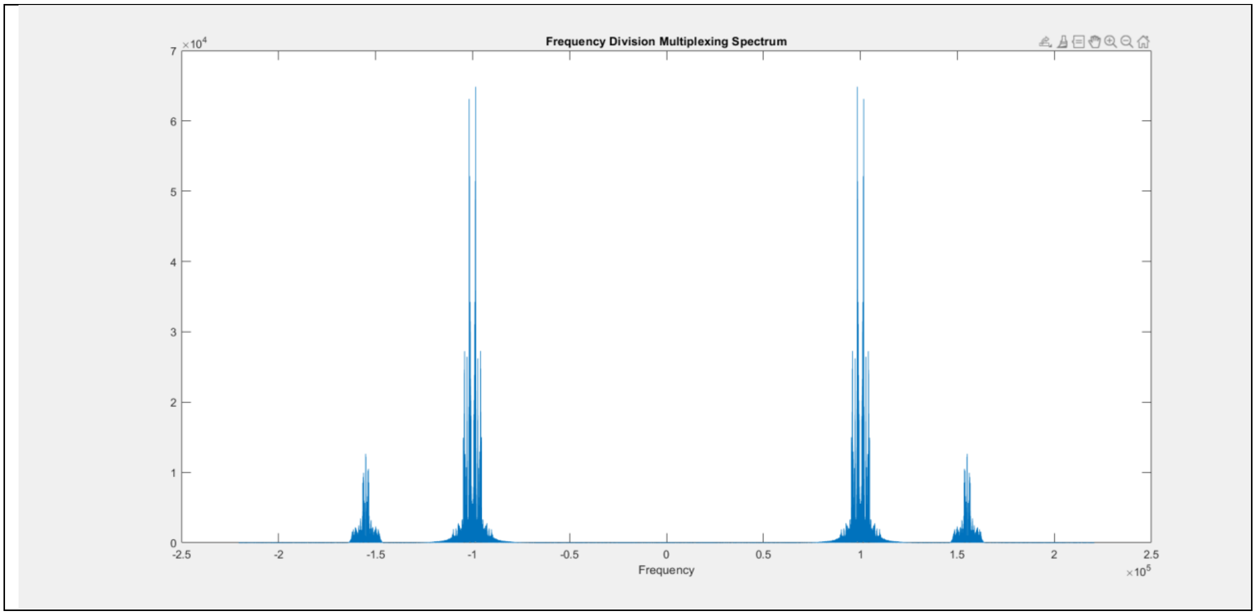
1. Reading monophonic audio signals into MATLAB.
2. Upsampling the audio signals.
3. Modulating the audio signals (each on a separate carrier).
4. Addition of the modulated signals.

Discussion

```
Firstly reading signals, then making signal monophonic channel and making
them have the same length so that I can add them together, we make up
sampling  to avoid aliasing, then  we modulate each signal with different
carrier and after multiply with carriers add them together , then make FFT
and display the spectrum and transmit the FDM Signal.
```

The figures

Figure 1: The spectrum of the output of the transmitter



2. ***The RF stage***

This part addresses the RF filter and the mixer following it.

Discussion

The RF stage is implemented as a BPF, centered at the carrier frequency. This stage is very. important as it rejects image frequency (at $F = F_{carrier} + 2F_{IF}$)noise and unwanted signals. Then comes the mixer. that shifts the required signal at the intermediate frequency (F_{IF}=27.5kHz in us case). We do that to avoid direct conversion problems, for the first signal image will be at F=155KHZ and for second signal image will be at 210KHZ so the BPF is very important .

The figures

Assume we want to demodulate the first signal (at ω_o).

Figure 2: the output of the RF filter (before the mixer)

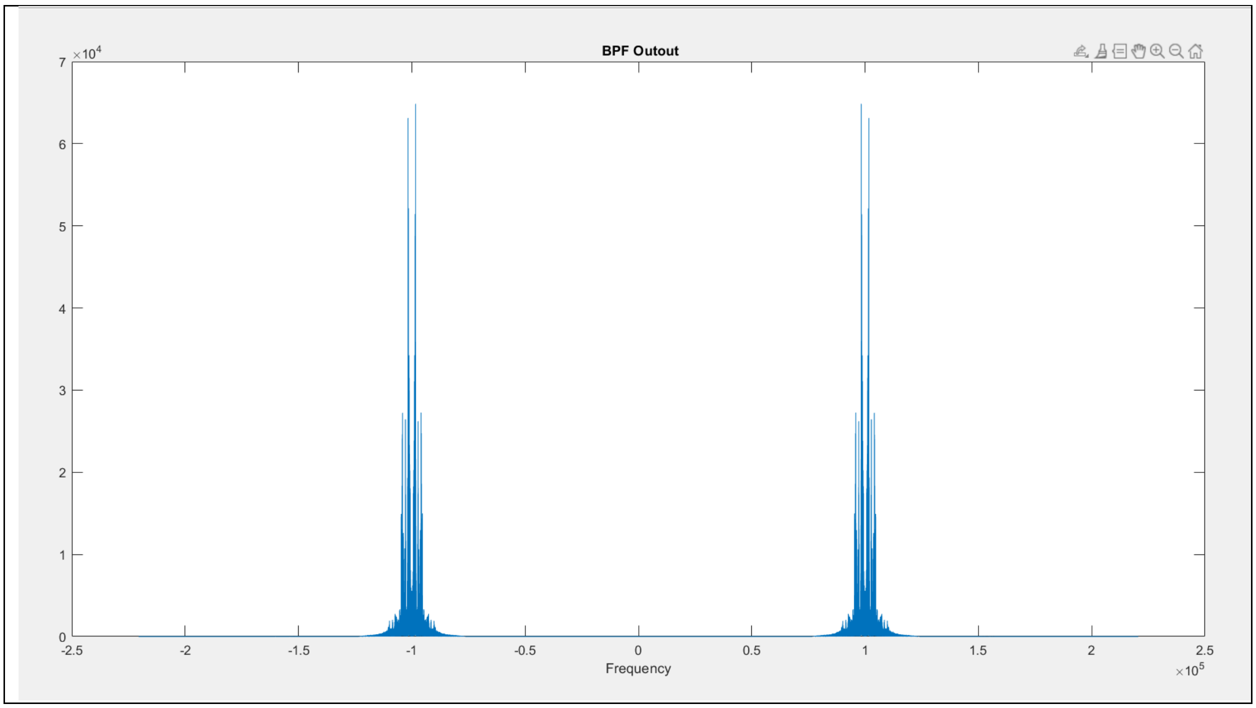
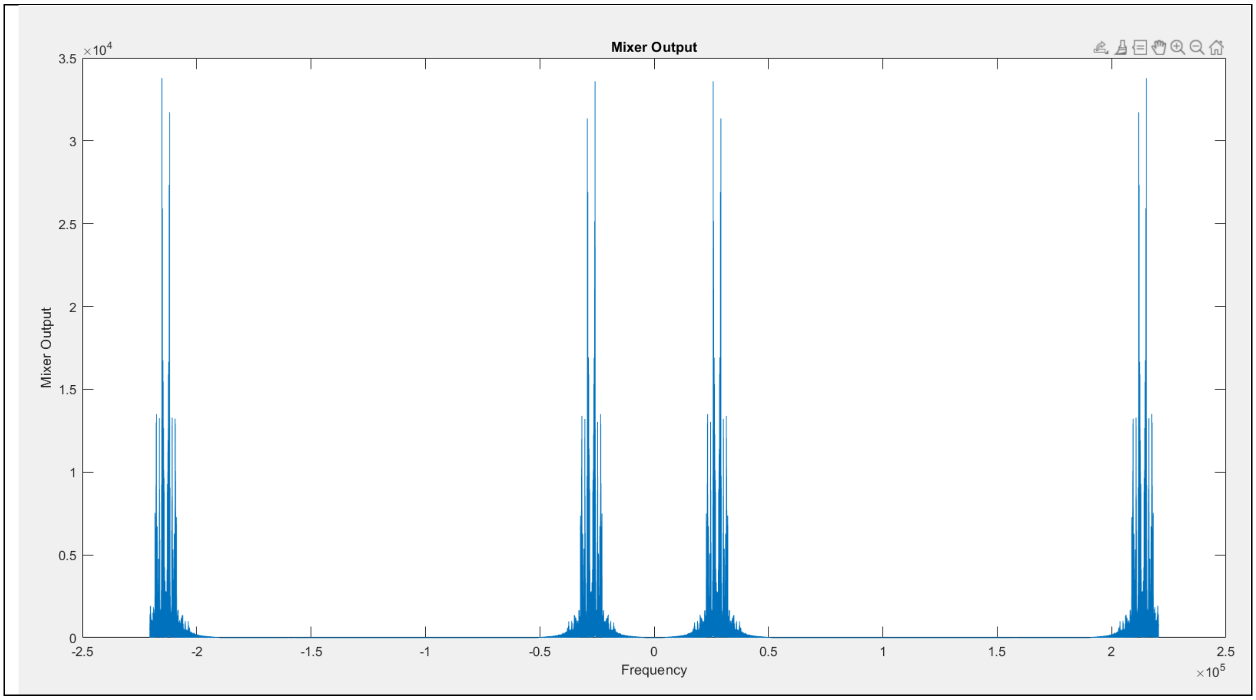


Figure 3: The output of the mixer



3. ***The IF stage***

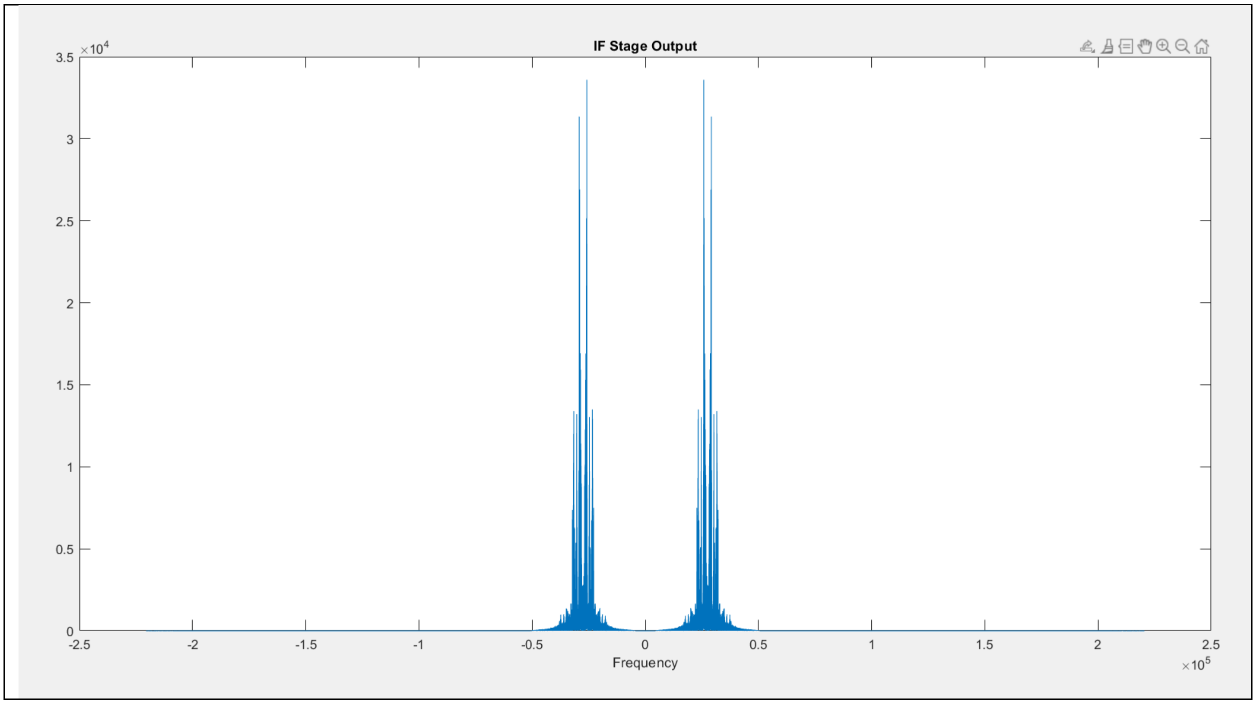
This part addresses the IF filter.

Discussion

This stage is implemented as a BPF only, center at the intermediate frequency, to select the IF signal & reject its higher frequency version, and the importance of carrying the received signal on IF before baseband to get rid of leakage & flicker noise & to improve the filters' selectivity.

The figures

Figure 4: Output of the IF filter



4. ***The baseband demodulator***

This part addresses the coherent detector used to demodulate the signal from the IF stage.

Discussion

In baseband stage: we demodulate the output signal of IF stage by a carrier signal with carrier frequency at F_{IF}, so we can have the desired signal at the baseband and at $2F_{IF}$. In LPF stage we apply low pass filter to obtain the desired signal at baseband only, after that we do downsampling to return the original sampling rate, then I can listen the desired sound

The figures

Figure 5: Output of the mixer (before the LPF)

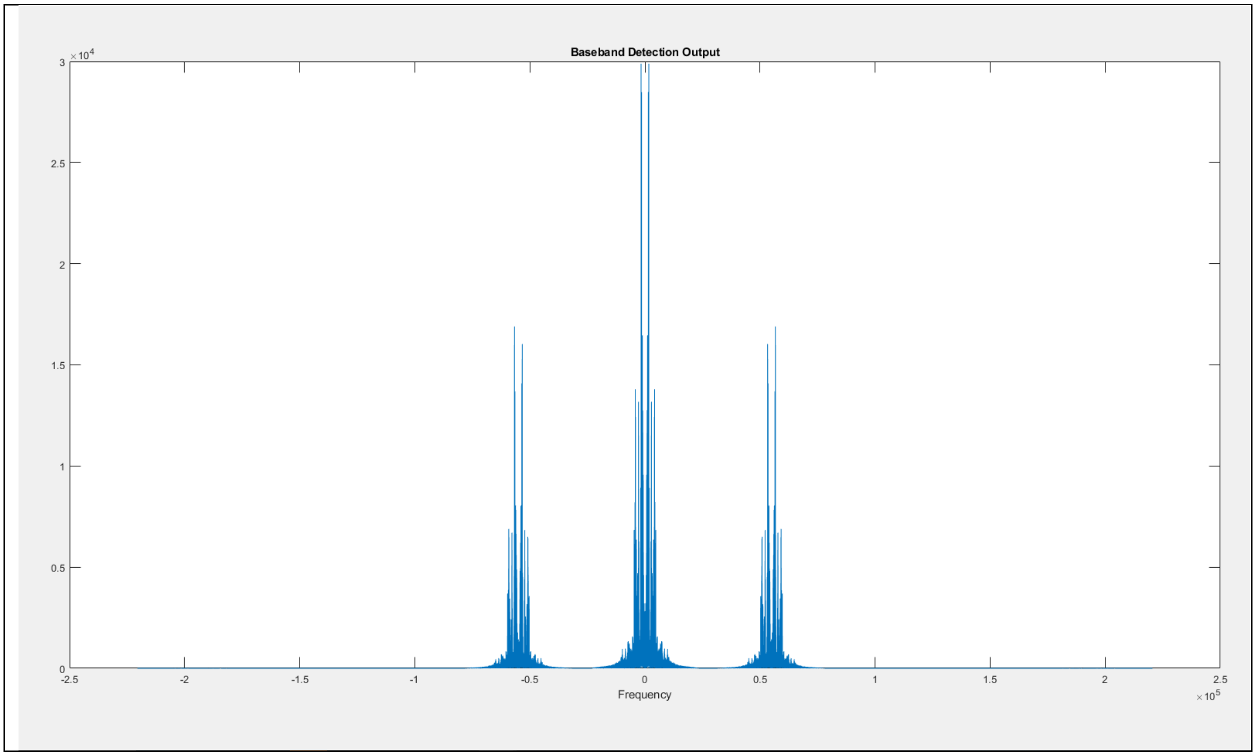
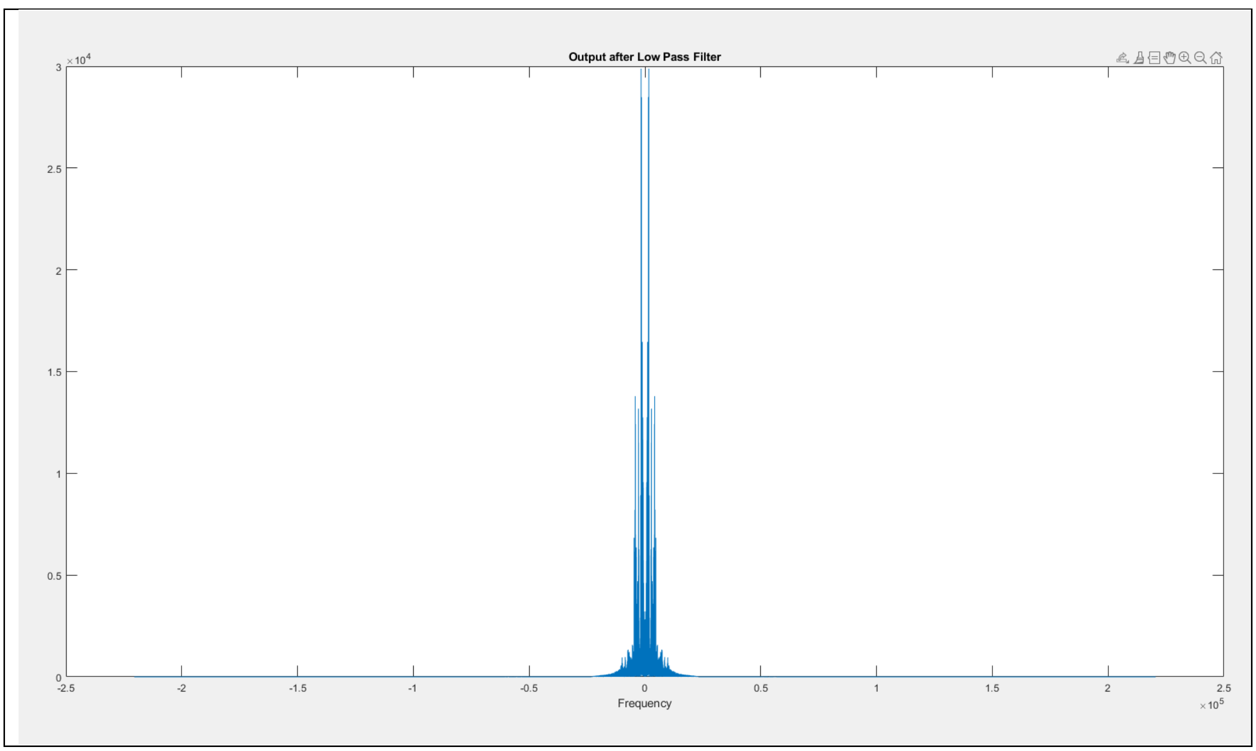


Figure 6: Output of the LPF



5. *Performance evaluation without the RF stage*

The figures

Figure 7: output of the RF mixer (no RF filter)

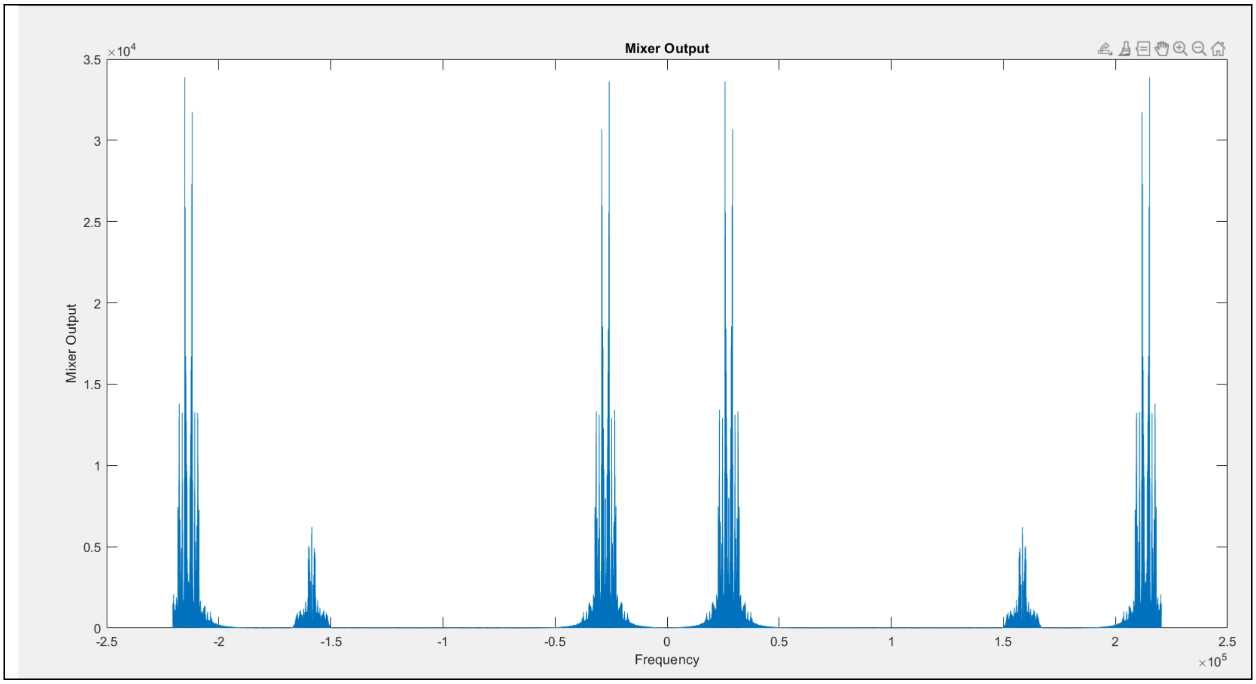


Figure 8: Output of the IF filter (no RF filter)

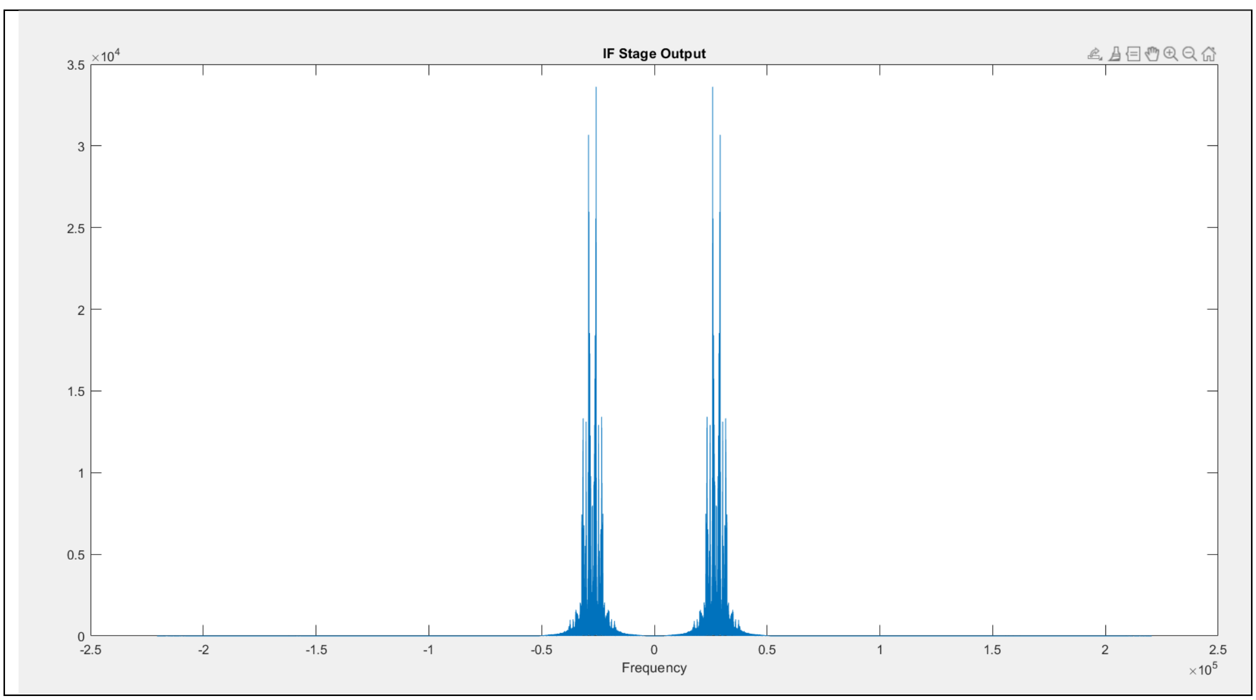


Figure 9: Output of the IF mixer before the LPF (no RF filter)

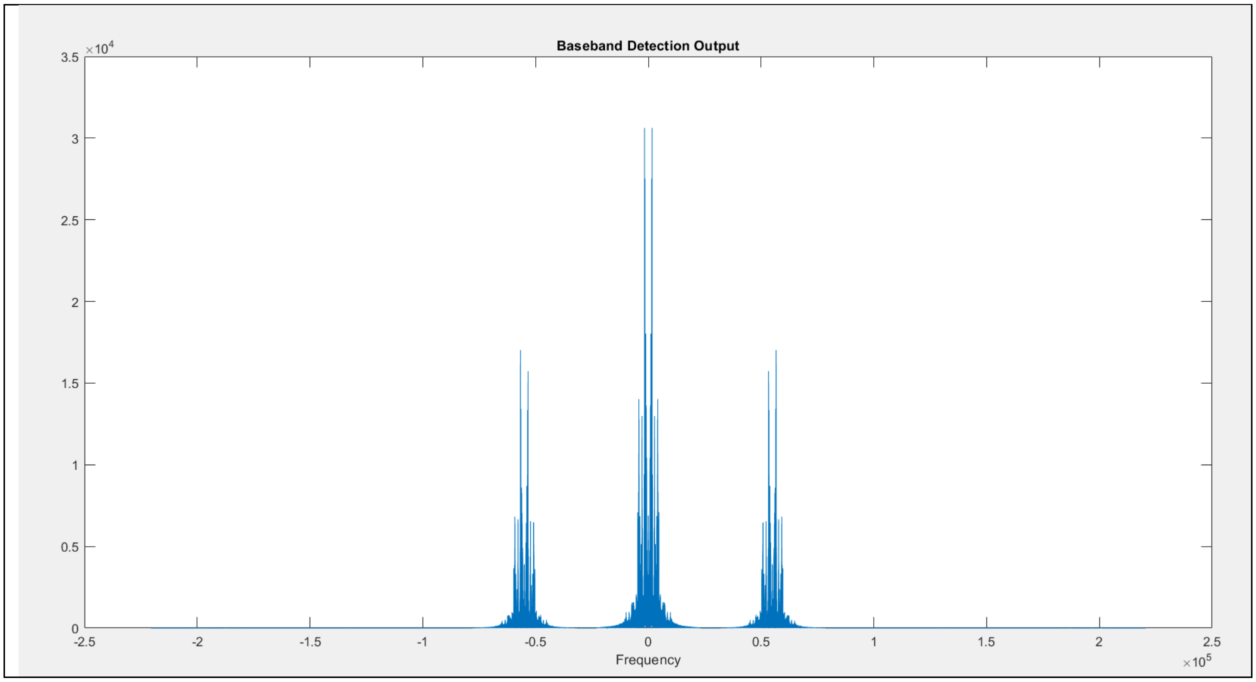
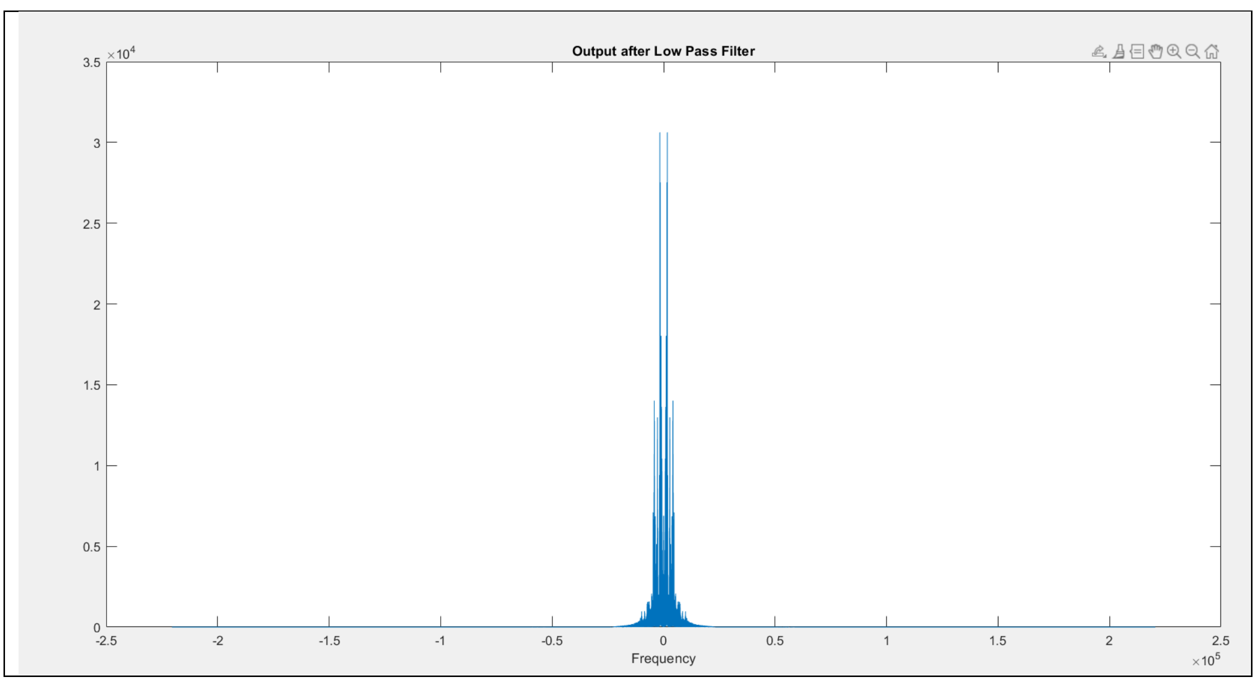


Figure 10: Ouptut of the LPF (no RF filter)



6. ***Comment on the output sound***

The output in case of RF stage existence is the same as the original audio signal without any interference with other signals. The output without RF stage sound1 is the original signal interfered with sound2 and the output audio contains the two audio signals. This happens because the second message is modulated with carrier frequency equal 155 KHZ and this frequency is image frequency of the first message $F_{Image1} = F_{carrier} + 2 * F_{IF} = 100 + 2 * 27.5 = 155KHZ$ so there is will interface with the two messages

What happens (in terms of spectrum and the sound quality) if the receiver oscillator has frequency offset by 0.1 KHz and 1 KHz

First case Offset = 0.1 KHZ: the sound was a little distorted; it is different than the original sound but still recognizable

Second case Offset = 1 KHZ: the sound was more distorted, the sound will be very bad it is different than the original and neither recognizable nor understood.

To avoid this problem: we must guarantee that there is no phase shift in the carrier frequency at the Transmitter and receiver.

The code

```
%************************** Super-heterodyne
Receiver******************%
clear all;
clc ;
close all;
% read sounds
[Sound1,FS]=audioread("Short_QuranPalestine.wav");
[Sound2,FS]=audioread("Short_SkyNewsArabia.wav");


% get sizes
Length_Sound1 = length(Sound1);
Length_Sound2 = length(Sound2);
if(Length_Sound1>Length_Sound2)
    Sound2=wextend('ar','zpd',Sound2,(Length_Sound1-
Length_Sound2),'d');
elseif (Length_Sound2>Length_Sound1)
    Sound1=wextend('ar','zpd',Sound1,(Length_Sound2-
Length_Sound1),'d');
end

%make signal Monophonic
Sound1(:,1)=Sound1(:,1)+Sound1(:,2);
Sound1(:,2) = [];
Sound2(:,1)=Sound2(:,1)+Sound2(:,2);
Sound2(:,2) = [];

%achieving Nyquist rule=10*FS
Message1=interp(Sound1 , 10 ) ;
Message2=interp(Sound2 , 10 ) ;

% F(new)=10*FS
FS = FS * 10 ;

%get n for Carrier
N = length(Message1) ;

TS = 1/FS ;      %get time
Stop_Time=N/FS; %get stop time
t = (0:TS:Stop_Time-TS)';
Carrier1 = cos(2*pi*100*1000*t) ;
%carrier2 with 100+55n KHZ
Carrier2 = cos(2*pi*(100+55)*1000*t) ;
```

```
%get frequency responce of Two Messages
Message1_Spectrum=fft(Message1);
Message2_Spectrum=fft(Message2);

k=-N/2:N/2-1;
figure
plot(k*FS/N,fftshift(abs(Message1_Spectrum)));
xlabel('Frequency');
title('Message1 Spectrum');
figure
plot(k*FS/N,fftshift(abs(Message2_Spectrum)));
xlabel('Frequency');
title('Message 2 Spectrum');

%Modulating signals
Transmitter1_Sound=Message1.*Carrier1 ;
Transmitter2_Sound=Message2.*Carrier2 ;
% Create The Frequency Division Multiplexed Signal By
Addition Of The Modulated Signals
Transmitter_Output=Transmitter1_Sound+Transmitter2_Sound;
%plotting spectrum of the channel
% perform FFT on signal
FDM = fft(Transmitter_Output );

figure
plot(k*FS/N,fftshift(abs(FDM)));
xlabel('Frequency');
title('Frequency Division Multiplexing Spectrum');

%*********RF Stage*********%
%*****choose The Channel*******%
% Choose The Required Audio Signals
disp (" ");
disp ("*********Channels******** ");
disp ("1. Short Quran Palestine: 100 KHz");
disp ("2. Short Sky News Arabia 155 KHz");
Freq_Channel = input ("Please Select The Desired Channel
Frequency in KHz: ");
Freq_Channel = 1000 * Freq_Channel;    % convert it to KHZ

%design BPF for First sound
Fstop1=Freq_Channel-24000;  % Edge of the stopband
Fpass1=Freq_Channel-22000;  % Edge of the passband
Astop1=80;     % Attenuation in the first stopband
```

```
Fpass2=Freq_Channel+22000;   % Closing edge of the
passband
Fstop2=Freq_Channel+24000;    % Edge of the second
stopband
Astop2=80;         % Attenuation in the second stopband
Apass=0.001;             % Amount of ripple allowed in the
passband
%specs of Bpf
BPF_specs=fdesign.bandpass('Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2
', ...
    Fstop1, Fpass1, Fpass2, Fstop2, Astop1,
Apass,Astop2,FS);
BPF = design(BPF_specs);
RF_Message= filter(BPF,Transmitter_Output);
RF=fft(RF_Message);
figure
plot(k*FS/N,fftshift(abs(RF)));
xlabel('Frequency');
title(' BPF Outout');


%********** Mixer*******%
Freq_IF=27500;
Mixer_Carrier=cos(2*pi*(Freq_IF+Freq_Channel)*t);
Mixer_Output=RF_Message.*Mixer_Carrier;
Mixer_Output_FFT=fft(Mixer_Output);
figure
plot(k*FS/N,fftshift(abs(Mixer_Output_FFT)));
xlabel('Frequency');
title('Mixer Output');
ylabel('Mixer Output');


%*********IF Stage**********%
%Design Baseband BPF
Fstop1=Freq_IF-24000;  % Edge of the stopband
Fpass1=Freq_IF-22000;  % Edge of the passband
Astop1=80;     % Attenuation in the first stopband
Fpass2=Freq_IF+22000;  % Closing edge of the passband
Fstop2=Freq_IF+24000;  % Edge of the second stopband
Astop2=80;     % Attenuation in the second stopband
Apass=0.001;        % Amount of ripple allowed in the
passband
```

```matlab
BPF_specs=fdesign.bandpass('Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2
', ...
    Fstop1, Fpass1, Fpass2, Fstop2, Astop1,
Apass,Astop2,FS);
BPF = design(BPF_specs);
%fvtool(BPF)     %response of filter
IF_Output= filter(BPF,Mixer_Output);
IF_Output_FFT=fft(IF_Output);

figure
plot(k*FS/N,fftshift(abs(IF_Output_FFT)));
xlabel('Frequency');
title('IF Stage Output');

%***********Baseband Detection************%
Carrier_Detection=cos(2*pi*27500*t);
Detection_Output=IF_Output.*Carrier_Detection;
Detection_Output_FFT=fft(Detection_Output);
figure
plot(k*FS/N,fftshift(abs(Detection_Output_FFT)));
xlabel('Frequency');
title('Baseband Detection Output');

%******Filter******%
F_pass = 22000; % Edge of the lowband
F_stop = 24000; % Edge of the stopband
A_pass = 0.001; % Amount of ripple allowed in the band
A_stop = 80; % Attenuation in the band
LPF_specs=fdesign.lowpass('Fp,Fst,Ap,Ast', ...
    F_pass, F_stop, A_pass, A_stop, FS);
LPF = design(LPF_specs);
LPF_Output= filter(LPF,Detection_Output);
LPF_Output_FFT= fft(LPF_Output);
figure
plot(k*FS/N,fftshift(abs(LPF_Output_FFT)));
xlabel('Frequency');
title('Output after Low Pass Filter');
LPF_Output=4.*LPF_Output;   %multiply by gain
Reciever=downsample(LPF_Output,10);   %down sampling /10
sound(Reciever,FS/10);
```

8. ***Types of Mixers***

There are two types of Mixers, they are:

I. **Multiplier:**

Such as Log-Amplifier multiplier which is designed to amplify input signal in such a way that the output signal is proportional to the logarithm of the input one, then adding the two output logarithmic signals (Message & Carrier), then anti-log them again so it will be equivalent to multiplying of them.

- **Advantages:**
  Wide dynamic range: it can handle a wide dynamic range of input signal and making them suitable for applications with varying signal magnitudes.
- **Disadvantages:**
  Complexity: it can be more complex to design and implement compared to simpler circuits.
  Cost: this complexity results in increased manufacturing costs.

II. **Switching Modulator:**

Such as Classic active mixer which Is called active refers to the use of active components such as transistors or op-amp to perform the mixing function.

- **Advantages:**
  Low-Noise operation: well designed active mixers can achieve low noise figures.
  Flexible gain control: they often provide the ability to control the gain of the input signal allowing for flexibility in adjusting the overall signal level.
- **Disadvantages:**
  Non-linearity: Active mixers, especially those using nonlinear active devices like transistors, can introduce nonlinearity and intermodulation products. These distortions may impact the quality of the output signal.
  Sensitivity to Impedance Mismatch: Active mixers can be sensitive to impedance mismatches, which can affect their performance. Proper impedance matching is often necessary for optimal operation.