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: Zeinab Mohammed**

**Table of figures**

Figure 1: The spectrum of the output of the transmitter 3

Figure 2: the output of the RF filter (before the mixer) 3

Figure 3: The output of the mixer 3

Figure 4: Output of the IF filter 4

Figure 5: Output of the mixer (before the LPF) 4

Figure 6: Output of the LPF 4

Figure 7: output of the RF mixer (no RF filter) 5

Figure 8: Output of the IF filter (no RF filter) 5

Figure 9: Output of the IF mixer before the LPF (no RF filter) 5

Figure 10: Ouptut of the LPF (no RF filter) 5

* ***The transmitter***

This part contains the following tasks

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

**Discussion**

|  |
| --- |
| . Reading the auidios using audioread function and use audioinfo function to check that signals have(2-channel) and get FS by using the ‘wavread’ command.  . Adding the two channels of the signals into one channel.  . making all signals with the same length by using padarry function then add interpulation that multiply fs by 12 since FC>>fs/2 .  .we modulate on separate carrier to prevent interfacing between signals.  .for multiplexing "add the modulated signals". |

**The figures**

**Figure 1: The spectrum of the output of the transmitter**

|  |
| --- |
| % Please insert your figure in this box. |

* ***The RF stage***

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

**Discussion**

|  |
| --- |
| Add 3-4 lines explaining the role of this stage and why we need it.  . RF\_stage (it isnot a sharp BPF because it 's compliex to take sharp filter at high frequency) but we using it to cancle the image massage which has frequency [F=Fc+Fif] ,to prevent the interfacing between signal and it 's image .So it reject image message. |

**The figures**

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

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

|  |
| --- |
| % Please insert a figure here. |

**Figure 3: The output of the mixer**

|  |
| --- |
| % Please insert a figure here. |

**Comments**

|  |
| --- |
| % Please add a comment on the order of the BPF you used and compare to practical filters (refer to the second point in page 3 of your project description).  .for signal\_1:'Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',70000/(6\*Fs),89340/(6\*Fs),108500/(6\*Fs),130000/(6\*Fs).  .for signal\_2:'Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',70000/(6\*Fs),149000/(6\*Fs),170000/(6\*Fs),180000/(6\*Fs).  .for signal\_3:'Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',70000/(6\*Fs),210000/(6\*Fs),230000/(6\*Fs),240000/(6\*Fs). |

* ***The IF stage***

This part addresses the IF filter.

**Discussion**

|  |
| --- |
| Add 3-4 lines explaining the role of this stage and why we need it.  .for "Super\_Hetrodyne Reciver",we using oscillator of cos(Wc+Wif)in IF mixer stage to move signal to at Wif so we can take a sharp filter here. |

**The figures**

**Figure 4: Output of the IF filter**

|  |
| --- |
| % Please insert a figure here. |

* ***The baseband demodulator***

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

**Discussion**

|  |
| --- |
| Add 3-4 lines explaining the role of this stage.  .since we take a sharp BPF at [IF\_stage] which signal at [Fif=30 k.hz] as pervious;now we want to use the coherent demodulator to move signal to the baseband using mixer of(cos wif) then take LPF so we can hear it clear. |

**The figures**

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

|  |
| --- |
| % Please insert a figure here. |

**Figure 6: Output of the LPF**

|  |
| --- |
| % Please insert a figure here. |

**Comments**

|  |
| --- |
| % Please add a comment on the order of the BPF you used and compare to practical filters (refer to the second point in page 3 of your project description).  .IF\_bPF IS same for all signals:'Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',1/(6\*Fs),10000/(6\*Fs),29000/(6\*Fs),50000/(6\*Fs)' |

* ***Performance evaluation without the RF stage***

**The figures**

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

|  |
| --- |
| % figure here. |

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

|  |
| --- |
| % figure here. |

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

|  |
| --- |
| % figure here. |

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

|  |
| --- |
| % figure here. |

* ***Comment on the output sound***

|  |
| --- |
| Comment on the sound you heard at the output of your designed Superheterodyne receiver in both cases of the existence and absence of the RF stage  .in case of non\_RF\_stage;the sound is not clear because interfacing happenes between signals.But in case of RF\_stage there is Filtering which separate the desired signal from other  modulated signals;So no interfacing here  . |

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

Answer of the previous question explaining your answer.



* ***The code***

|  |
| --- |
| Please insert the code of your project here. A good code needs to be readable and understandable. Use clear names for variables. Use comments as much as you want.  %% -----------------Reading the auidios-----------------------%%  [y1,Fs] = audioread('Short\_BBCArabic2.wav');  [y2,Fs] = audioread('Short\_FM9090.wav');  [y3,Fs] = audioread('Short\_QuranPalestine.wav');  [y4,Fs] = audioread('Short\_RussianVoice.wav');  [y5,Fs] = audioread('Short\_SkyNewsArabia.wav');  [y6,Fs] = audioread('Short\_WRNArabic.wav');  %% ------------------Adding the two channels of the signals into one channel---------------%%  x1=y1(:,1)+y1(:,2);  x2=y2(:,1)+y2(:,2);  x3=y3(:,1)+y3(:,2);  x4=y4(:,1)+y4(:,2);  x5=y5(:,1)+y5(:,2);  x6=y6(:,1)+y6(:,2);  %% ------------------ making all signals with the same length then add interpulation -----------------------%%  sig1 = interp(padarray(x1,[896 0],0,'post'),12);  sig2 = interp(padarray(x2,[43904 0],0,'post'),12);  sig3 = interp(padarray(x3, [2240 0],0,'post'),12);  % sig4 = interp(padarray(x4,[38080 0],0,'post'),12);  % sig5 = interp(padarray(x5,[29568 0],0,'post'),12);  % sig6= interp(x6,10);  %%-------------------- 1st modulated signal----------------------%%  t1=0:1/(12\*Fs):(length(sig1)-1)/(12\*Fs);  carrier=cos(2\*pi\*100000\*t1);  carrier\_trans=transpose(carrier);  modulated\_sig1=sig1.\*carrier\_trans;  modulated\_sig1\_fd=fft(modulated\_sig1);  fr=linspace(-6\*Fs,6\*Fs,length(modulated\_sig1\_fd));  modulated\_sig1\_fd1=fftshift(abs(modulated\_sig1\_fd));  %%--------------------- 2nd modulated signal---------------------%%  t2=0:1/(12\*Fs):(length(sig2)-1)/(12\*Fs);  carrier2=cos(2\*pi\*160000\*t2);  carrier\_trans2=transpose(carrier2);  modulated\_sig2=sig2.\*carrier\_trans2;  modulated\_sig2\_fd=fft(modulated\_sig2);  fr2=linspace(-6\*Fs,6\*Fs,length(modulated\_sig2\_fd));  modulated\_sig2\_fd1=modulated\_sig1\_fd1+fftshift(abs(modulated\_sig2\_fd));  %%--------------------- 3rd modulated signal----------------------%%  t3=0:1/(12\*Fs):(length(sig3)-1)/(12\*Fs);  carrier3=cos(2\*pi\*220000\*t3);  carrier\_trans3=transpose(carrier3);  modulated\_sig3=sig3.\*carrier\_trans3;  modulated\_sig3\_fd=fft(modulated\_sig3);  fr3=linspace(-6\*Fs,6\*Fs,length(modulated\_sig3\_fd));  figure  modulated\_sig3\_fd1=plot(fr3,modulated\_sig2\_fd1+fftshift(abs(modulated\_sig3\_fd)));  title('Figure 1: The spectrum of the output of the transmitter')  ylabel('Amplitude')  xlabel('Frequency')  %% ----------------------------spectrums of 6 the signals-----------------------------%%  % subplot(3,2,1);  % plot(fftshift(abs(fft(x1))))  % title(' Spectrum of sig 1')  % subplot(3,2,2);  % plot(fftshift(abs(fft(x2))))  % title(' Spectrum of sig 2')  % subplot(3,2,3);  % plot(fftshift(abs(fft(x3))))  % title(' Spectrum of sig 3')  % subplot(3,2,4);  % plot(fftshift(abs(fft(x4))))  % title(' Spectrum of sig 4')  % subplot(3,2,5);  % plot(fftshift(abs(fft(x5))))  % title(' Spectrum of sig 5')  % subplot(3,2,6);  % plot(fftshift(abs(fft(x6))))  % title(' Spectrum of sig 6')  %% ------------------(RF)filter of modulated\_signal\_1-------------------------------%%  multiplexed\_time=modulated\_sig1+modulated\_sig2+modulated\_sig3;  RF = fdesign.bandpass('Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',70000/(6\*Fs),89340/(6\*Fs),108500/(6\*Fs),130000/(6\*Fs),60,1,60);  Hd = design(RF,'equiripple');  FILTER1=filter(Hd,multiplexed\_time);  order\_filter=order(Hd);  figure  plot(fr,fftshift(abs(fft(FILTER1))));  title('Figure 2: the output of the RF filter 1 (before the mixer)')  ylabel('Amplitude')  xlabel('Frequency')  %% -----------------(RF)filter of modulated\_signal\_2-----------------------%%  RF2 = fdesign.bandpass('Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',70000/(6\*Fs),149000/(6\*Fs),170000/(6\*Fs),180000/(6\*Fs),60,1,60);  Hd2 = design(RF2,'equiripple');  FILTER2=filter(Hd2,multiplexed\_time);  order\_filter2=order(Hd2);  figure  plot(fr2,fftshift(abs(fft(FILTER2))));  title('Figure 2: the output of the RF filter 2 (before the mixer)')  ylabel('Amplitude')  xlabel('Frequency')  %% -----------------(RF)filter of modulated\_signal\_3------------------------%%  RF3 = fdesign.bandpass('Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',70000/(6\*Fs),210000/(6\*Fs),230000/(6\*Fs),240000/(6\*Fs),60,1,60);  Hd3 = design(RF3,'equiripple');  FILTER3=filter(Hd3,multiplexed\_time);  order\_filter3=order(Hd3);  figure  plot(fr3,fftshift(abs(fft(FILTER3))));  title('Figure 2: the output of the RF filter 3 (before the mixer)')  ylabel('Amplitude')  xlabel('Frequency')  %% -----------------mix band pass filter of signal\_1 with carrier of Wc+WiF----------------%%  % t1\_FILTER=0:1/(12\*Fs):(length( multiplexed\_time)-1)/(12\*Fs); in case of no RF  t1\_FILTER=0:1/(12\*Fs):(length(FILTER1)-1)/(12\*Fs);  carrier\_FILTER=cos(2\*pi\*130000\*t1\_FILTER);  carrier\_FILTER\_trans=transpose(carrier\_FILTER);  % modulated\_FILTER1= multiplexed\_time.\*carrier\_FILTER1\_trans; in case of no RF  modulated\_FILTER1=FILTER1.\*carrier\_FILTER\_trans;  modulated\_FILTER1\_fd=fft(modulated\_FILTER1);  f\_filter1=linspace(-6\*Fs,6\*Fs,length(modulated\_FILTER1\_fd));  figure  plot(f\_filter1,fftshift(abs(modulated\_FILTER1\_fd)));  title('Figure 3: The output of the mixer for signal 1')  ylabel('Amplitude')  xlabel('Frequency')  %% ----------------- mix band pass filter of signal\_2 with carrier of Wc+WiF----------------%%  % t2\_FILTER=0:1/(12\*Fs):(length( multiplexed\_time)-1)/(12\*Fs); in case of no RF  t2\_FILTER=0:1/(12\*Fs):(length(FILTER2)-1)/(12\*Fs);  carrier\_FILTER2=cos(2\*pi\*190000\*t2\_FILTER);  carrier\_FILTER2\_trans=transpose(carrier\_FILTER2);  % modulated\_FILTER2= multiplexed\_time.\*carrier\_FILTER2\_trans; in case of no RF  modulated\_FILTER2=FILTER2.\*carrier\_FILTER2\_trans;  modulated\_FILTER2\_fd=fft(modulated\_FILTER2);  f\_filter2=linspace(-6\*Fs,6\*Fs,length(modulated\_FILTER2\_fd));  figure  plot(f\_filter2,fftshift(abs(modulated\_FILTER2\_fd)));  title('Figure 3: The output of the mixer for signal 2')  ylabel('Amplitude')  xlabel('Frequency')  %% ---------------- mix band pass filter of signal\_3 with carrier of Wc+WiF-----------------%%  % t3\_FILTER=0:1/(12\*Fs):(length( multiplexed\_time)-1)/(12\*Fs); in case of no RF  t3\_FILTER=0:1/(12\*Fs):(length(FILTER3)-1)/(12\*Fs);  carrier\_FILTER3=cos(2\*pi\*190000\*t3\_FILTER);  carrier\_FILTER3\_trans=transpose(carrier\_FILTER3);  % modulated\_FILTER3= multiplexed\_time.\*carrier\_FILTER3\_trans; in case of no RF  modulated\_FILTER3=FILTER3.\*carrier\_FILTER3\_trans;  modulated\_FILTER3\_fd=fft(modulated\_FILTER3);  f\_filter3=linspace(-6\*Fs,6\*Fs,length(modulated\_FILTER3\_fd));  figure  plot(f\_filter3,fftshift(abs(modulated\_FILTER3\_fd)));  title('Figure 3: The output of the mixer for signal 3')  ylabel('Amplitude')  xlabel('Frequency')  %% -----------------------(IF) band pass filter2 of signal 1--------------------%%  RF\_1 = fdesign.bandpass('Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',1/(6\*Fs),10000/(6\*Fs),29000/(6\*Fs),50000/(6\*Fs),60,1,60);  Hd\_1 = design(RF\_1,'equiripple');  FILTER\_WIF\_1=filter(Hd\_1,modulated\_FILTER1);  f\_s\_filter1=linspace(-6\*Fs,6\*Fs,length(fft( FILTER\_WIF\_1)));  order\_filter\_1=order(Hd\_1);  figure  plot(f\_s\_filter1,fftshift(abs(fft( FILTER\_WIF\_1))))  title('Figure 4: Output of the IF filter1')  ylabel('Amplitude')  xlabel('Frequency')  %% ----------------------(IF) band pass filter2 of signal 2 --------------------%%  RF\_2 = fdesign.bandpass('Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',1/(6\*Fs),10000/(6\*Fs),29000/(6\*Fs),50000/(6\*Fs),60,1,60);  Hd\_2 = design(RF\_2,'equiripple');  FILTER\_WIF\_2=filter(Hd\_2,modulated\_FILTER2);  f\_s\_filter2=linspace(-6\*Fs,6\*Fs,length(fft( FILTER\_WIF\_2)));  order\_filter\_2=order(Hd\_2);  figure  plot(f\_s\_filter2,fftshift(abs(fft( FILTER\_WIF\_2))))  title('Figure 4: Output of the IF filter2')  ylabel('Amplitude')  xlabel('Frequency')  %% ----------------------(IF) band pass filter2 of signal 3 ---------------------%%  RF\_3 = fdesign.bandpass('Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',1/(6\*Fs),10000/(6\*Fs),29000/(6\*Fs),50000/(6\*Fs),60,1,60);  Hd\_3 = design(RF\_3,'equiripple');  FILTER\_WIF\_3=filter(Hd\_3,modulated\_FILTER3);  f\_s\_filter3=linspace(-6\*Fs,6\*Fs,length(fft( FILTER\_WIF\_3)));  order\_filter\_3=order(Hd\_3);  figure  plot(f\_s\_filter3,fftshift(abs(fft( FILTER\_WIF\_3))))  title('Figure 4: Output of the IF filter3')  ylabel('Amplitude')  xlabel('Frequency')  %% -------------------- mix the signal\_1 after second band pass filter with cos WIF ----------------------%%  t1\_second\_FILTER=0:1/(12\*Fs):(length(FILTER\_WIF\_1)-1)/(12\*Fs);  carrier\_second\_FILTER=cos(2\*pi\*30000\*t1\_second\_FILTER);  carrier\_second\_FILTER\_trans=transpose(carrier\_second\_FILTER);  modulated\_second\_FILTER1=FILTER\_WIF\_1.\*carrier\_second\_FILTER\_trans;  f\_second\_filter1=linspace(-6\*Fs,6\*Fs,length(modulated\_second\_FILTER1));  figure  plot(f\_second\_filter1,fftshift(abs(fft(modulated\_second\_FILTER1))))  title('Figure 5: Output of the mixer for signal 1 (before the LPF)')  ylabel('Amplitude')  xlabel('Frequency')  %% --------------------- low pass filter for signal 1 ------------------------%%  d1=fdesign.lowpass('Fp,Fst,Ap,Ast',20000/(6\*Fs),25000/(6\*Fs),1,60);  Hd\_LPF1 = design(d1,'equiripple');  LPF1=filter(Hd\_LPF1,modulated\_second\_FILTER1);  f\_LPF1=linspace(-6\*Fs,6\*Fs,length(modulated\_second\_FILTER1));  order\_filter\_lPF1=order(Hd\_LPF1);  figure  plot(f\_LPF1,fftshift(abs(fft(LPF1))));  title('Figure 6: Output of the LPF1)')  ylabel('Amplitude')  xlabel('Frequency')  lpf=downsample(LPF1,12);  sound(lpf,Fs)  %% --------------------- mix the signal\_2 after second band pass filter with cos WIF -------------%%  t2\_second\_FILTER=0:1/(12\*Fs):(length(FILTER\_WIF\_2)-1)/(12\*Fs);  carrier\_second\_FILTER2=cos(2\*pi\*30000\*t2\_second\_FILTER);  carrier\_second\_FILTER\_trans2=transpose(carrier\_second\_FILTER2);  modulated\_second\_FILTER2=FILTER\_WIF\_2.\*carrier\_second\_FILTER\_trans2;  f\_second\_filter2=linspace(-6\*Fs,6\*Fs,length(modulated\_second\_FILTER2));  figure  plot(f\_second\_filter2,fftshift(abs(fft(modulated\_second\_FILTER2))))  title('Figure 5: Output of the mixer for signal 2 (before the LPF)')  ylabel('Amplitude')  xlabel('Frequency')  %% --------------------- low pass filter for signal 2 ------------------------%%  d2=fdesign.lowpass('Fp,Fst,Ap,Ast',20000/(6\*Fs),25000/(6\*Fs),1,60);  Hd\_LPF2 = design(d2,'equiripple');  LPF2=filter(Hd\_LPF2,modulated\_second\_FILTER2);  f\_LPF2=linspace(-6\*Fs,6\*Fs,length(modulated\_second\_FILTER2));  order\_filter\_lPF2=order(Hd\_LPF2);  figure  plot(f\_LPF2,fftshift(abs(fft(LPF2))));  title('Figure 6: Output of the LPF2)')  ylabel('Amplitude')  xlabel('Frequency')  lpf2=downsample(LPF2,12);  %sound(lpf2,Fs)  %% -------------------- mix the signal\_3 after second band pass filter with cos WIF -------------%%  t3\_second\_FILTER=0:1/(12\*Fs):(length(FILTER\_WIF\_3)-1)/(12\*Fs);  carrier\_second\_FILTER3=cos(2\*pi\*30000\*t3\_second\_FILTER);  carrier\_second\_FILTER\_trans3=transpose(carrier\_second\_FILTER3);  modulated\_second\_FILTER3=FILTER\_WIF\_3.\*carrier\_second\_FILTER\_trans3;  f\_second\_filter3=linspace(-6\*Fs,6\*Fs,length(modulated\_second\_FILTER3));  figure  plot(f\_second\_filter3,fftshift(abs(fft(modulated\_second\_FILTER3))))  title('Figure 5: Output of the mixer for signal 3 (before the LPF)')  ylabel('Amplitude')  xlabel('Frequency')  %% --------------------- low pass filter for signal 3 ------------------------%%  d3=fdesign.lowpass('Fp,Fst,Ap,Ast',20000/(6\*Fs),25000/(6\*Fs),1,60);  Hd\_LPF3 = design(d3,'equiripple');  LPF3=filter(Hd\_LPF3,modulated\_second\_FILTER3);  f\_LPF3=linspace(-6\*Fs,6\*Fs,length(modulated\_second\_FILTER3));  order\_filter\_lPF3=order(Hd\_LPF3);  figure  plot(f\_LPF3,fftshift(abs(fft(LPF3))));  title('Figure 6: Output of the LPF3)')  ylabel('Amplitude')  xlabel('Frequency')  lpf3=downsample(LPF3,12);  %sound(lpf3,Fs) |