<u>ECE320-001</u>

SIGNALS AND SYSTEMS II

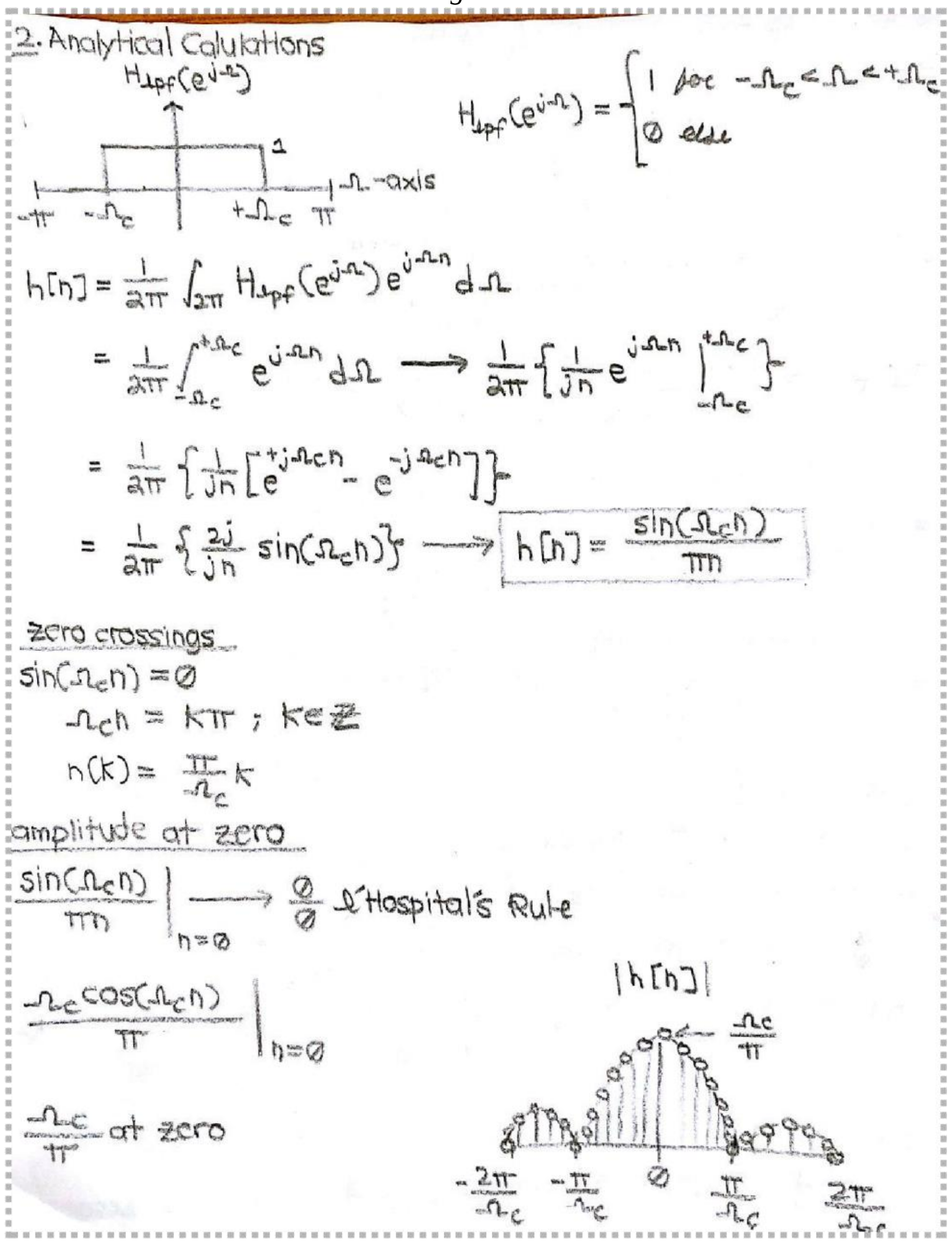
<u>MATLAB PROJECT II - FIR FILTER DESIGN WITH AUDIO FILTERING</u>

<u>Team Members:</u> Darima Batueva-Islam
Ahad Hassan

2. Analytical Calculations

(a) Using the inverse Fourier transform, determine the impulse response for the ideal lowpass filter.

Figure 1



2. Analytical Calculations

$$H_{lpf}(e^{j\Omega}) = \begin{cases} 1 & \text{for } -\Omega_c < \Omega < +\Omega_c \\ 0 & \text{else} \end{cases}$$

$$h[n] = \frac{1}{2\pi}\int_{2\pi} H_{lpf}(e^{j\Omega})e^{j\Omega n}\,d\Omega$$

$$= \frac{1}{2\pi}\int_{-\Omega_c}^{+\Omega_c} e^{j\Omega n}\,d\Omega \longrightarrow \frac{1}{2\pi}\left\{\frac{1}{jn}e^{j\Omega n}\Big|_{-\Omega_c}^{+\Omega_c}\right\}$$

$$= \frac{1}{2\pi}\left\{\frac{1}{jn}\left[e^{+j\Omega_c n} - e^{-j\Omega_c n}\right]\right\}$$

$$= \frac{1}{2\pi}\left\{\frac{2j}{jn}\sin(\Omega_c n)\right\} \longrightarrow \boxed{h[n] = \frac{\sin(\Omega_c n)}{\pi n}}$$

zero crossings

$$\sin(\Omega_c n) = 0$$

$$\Omega_c n = k\pi \ ; \ k \in \mathbb{Z}$$

$$n(k) = \frac{\pi}{\Omega_c}k$$

amplitude at zero

$$\frac{\sin(\Omega_c n)}{\pi n}\Big|_{n=0} \longrightarrow \frac{0}{0} \text{ l'Hospital's Rule}$$

$$\frac{\Omega_c \cos(\Omega_c n)}{\pi}\Big|_{n=0}$$

$\frac{\Omega_c}{\pi}$ at zero

While not instructed, we sketched the absolute value of the impulse response in figure 1. We chose to do this so it was easier to draw.

3. FIR Filter Design via the Window Method

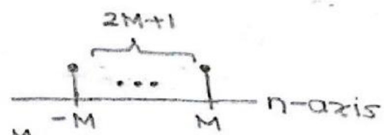
(a) Analytically determine the transform of the rectangular window $w_{rect}[n]$. Make a sketch (by hand) of this transform. Comment on how the transform changes as the length of the window increases.

Figure 2

3. FIR Filter Design via Window Method

a) transform $w_{rect}[n]$

$$w_{rect}[n] = \begin{cases} 1 & -M \le n \le +M \\ 0 & \text{otherwise} \end{cases}$$



$$W(e^{j\Omega}) = \sum_{n=-M}^{+M} x[n]e^{-j\Omega n} \longrightarrow \sum_{n=-M}^{M} (1)e^{-j\Omega n} \text{ geometric series}$$

$$W(e^{j\Omega}) = \frac{(e^{+j\Omega M} - e^{-j\Omega(M+1)})}{1-e^{-j\Omega}} \longrightarrow \frac{e^{+j\Omega M} - e^{-j\Omega M}e^{-j\Omega(1)}}{1-e^{-j\Omega}}$$

$$= \frac{\cancel{e^{j\Omega(-\frac{1}{2})}}\left[e^{+j\Omega M}e^{j\Omega(\frac{1}{2})} - e^{-j\Omega M}e^{-j\Omega(\frac{1}{2})}\right]}{\cancel{e^{j\Omega(-\frac{1}{2})}}\left[e^{j\Omega(+\frac{1}{2})} - e^{j\Omega(-\frac{1}{2})}\right]}$$

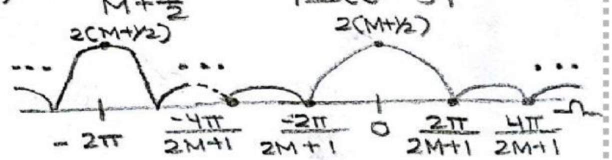
$$= \frac{e^{+j\Omega(M+\frac{1}{2})} - e^{-j\Omega(M+\frac{1}{2})}}{e^{j\Omega(-\frac{1}{2})} - e^{j\Omega(\frac{1}{2})}} \text{ so... } W(e^{j\Omega}) = \frac{\sin(\Omega(M+\frac{1}{2}))}{\sin(\frac{\Omega}{2})}$$

zero crossings

$\sin(\Omega(M+\frac{1}{2})) = k\pi \; ; \; k \in \mathbb{Z}$

$\Omega(M+\frac{1}{2}) = k\pi$ so... $\Omega_{zeros}(k) = \dfrac{k\pi}{M+\frac{1}{2}} \rightarrow \dfrac{2\pi k}{2M+1}$

amplitude at origen l'Hospital



$$\left.\frac{\sin(\Omega(M+\frac{1}{2}))}{\sin(\frac{\Omega}{2})}\right|_{\Omega=0} \longrightarrow \frac{0}{0}$$

$$\left.\frac{\cos(\Omega(M+\frac{1}{2}))\cdot(M+\frac{1}{2})}{\frac{1}{2}\cos(\frac{\Omega}{2})}\right|_{\Omega=0} \longrightarrow \frac{1\cdot(M+\frac{1}{2})}{\frac{1}{2}} \text{ so... } 2(M+\tfrac{1}{2}) \text{ at zero}$$

In figure 2, the frequency response of the window is dependent on *M*(**note to grader:** it is understood by both project partners that the frequency response is periodic every 2π.While the work expressed above only draws periodicity negative 2π , we ran out of paper to show positive for 2π.). The overall length of the window *L* is equal to *L=2M+1*, meaning that an 11-point window spans from -5 to 5 (-5, -4, . . . 0, . . . 4, 5). Thus, the value for $\Omega_{rect-win} = 0$ depends on the overall length of the window. An

11-point window will have a height of 11 at Ω = 0.The longer the window, the greater the value at Ω = 0

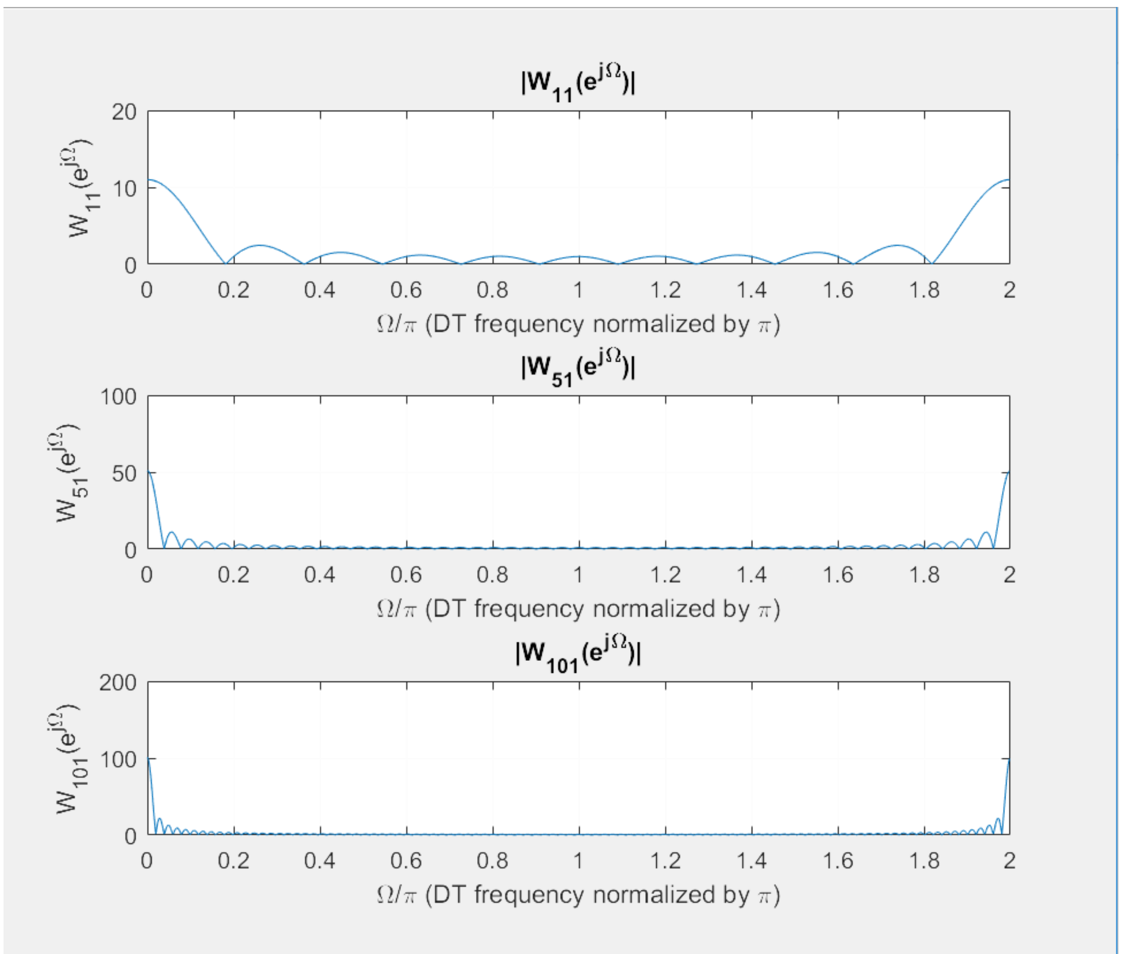
3. FIR Filter Design via the Window Method

(a) continued

The zero crossings are dependent on M by $2\pi/(2M + 1)$. The longer the window is, the greater the value M is. Thus, when we increase M, we will have more zero crossings for longer windows. This is verified in the plots below.

(b) **Generate rectangular windows of length 11, 51, and 101 points. Plot the frequency response magnitudes for each of the windows on a single plot. How do they compare? Were your predictions of how the transform changes with the length of the window correct?**

Figure 3



As shown above in figure 3, we can see that our analytical calculations were correct. The window height at Ω=0 is based on the length of the window. We can also see that as the window gets longer, we have more zero crossings on the Ω-*axis*. So, our predictions (based on our calculations) were correct.

3. FIR Filter Design via the Window Method

(b) continued

Figure 4

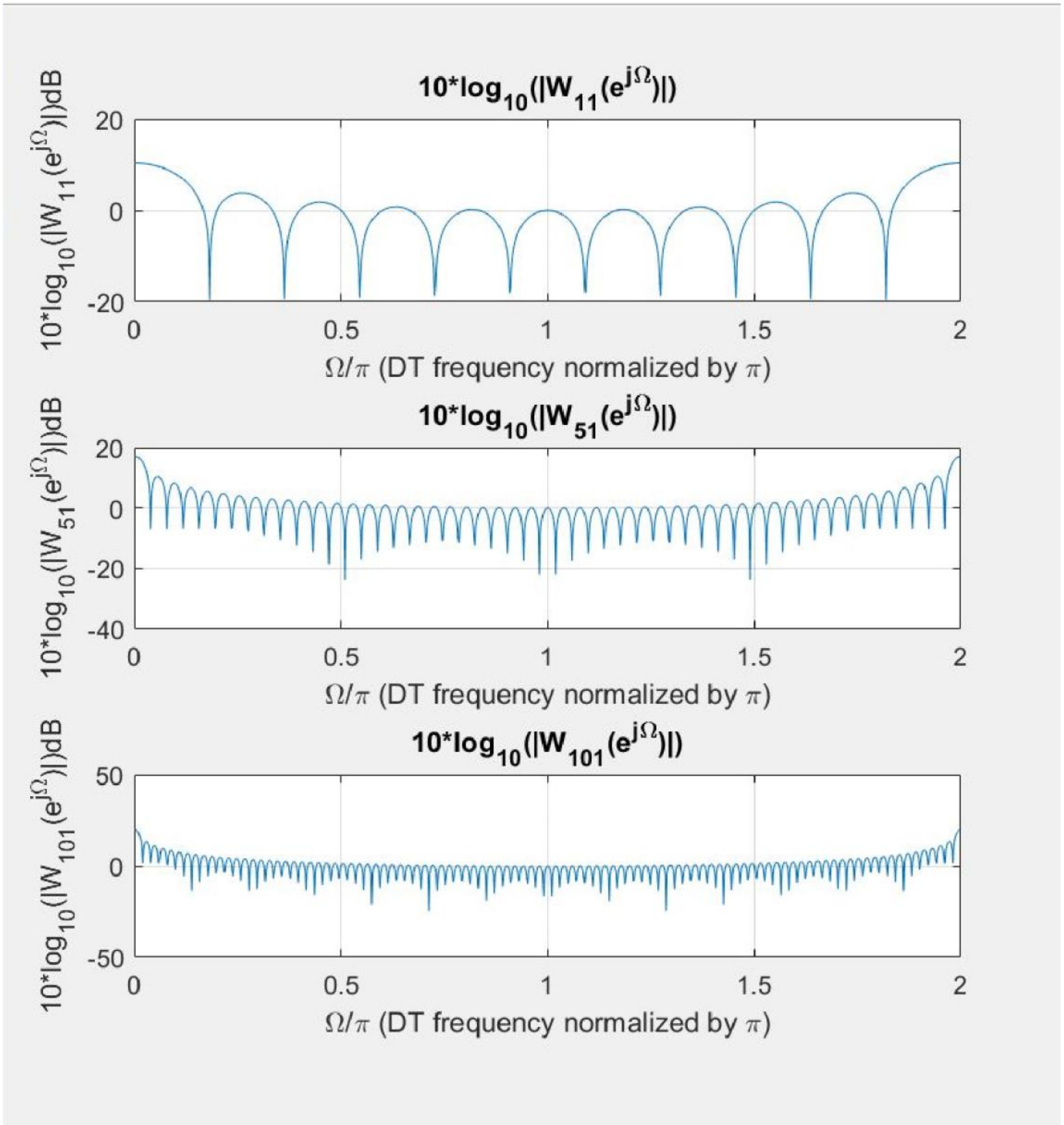


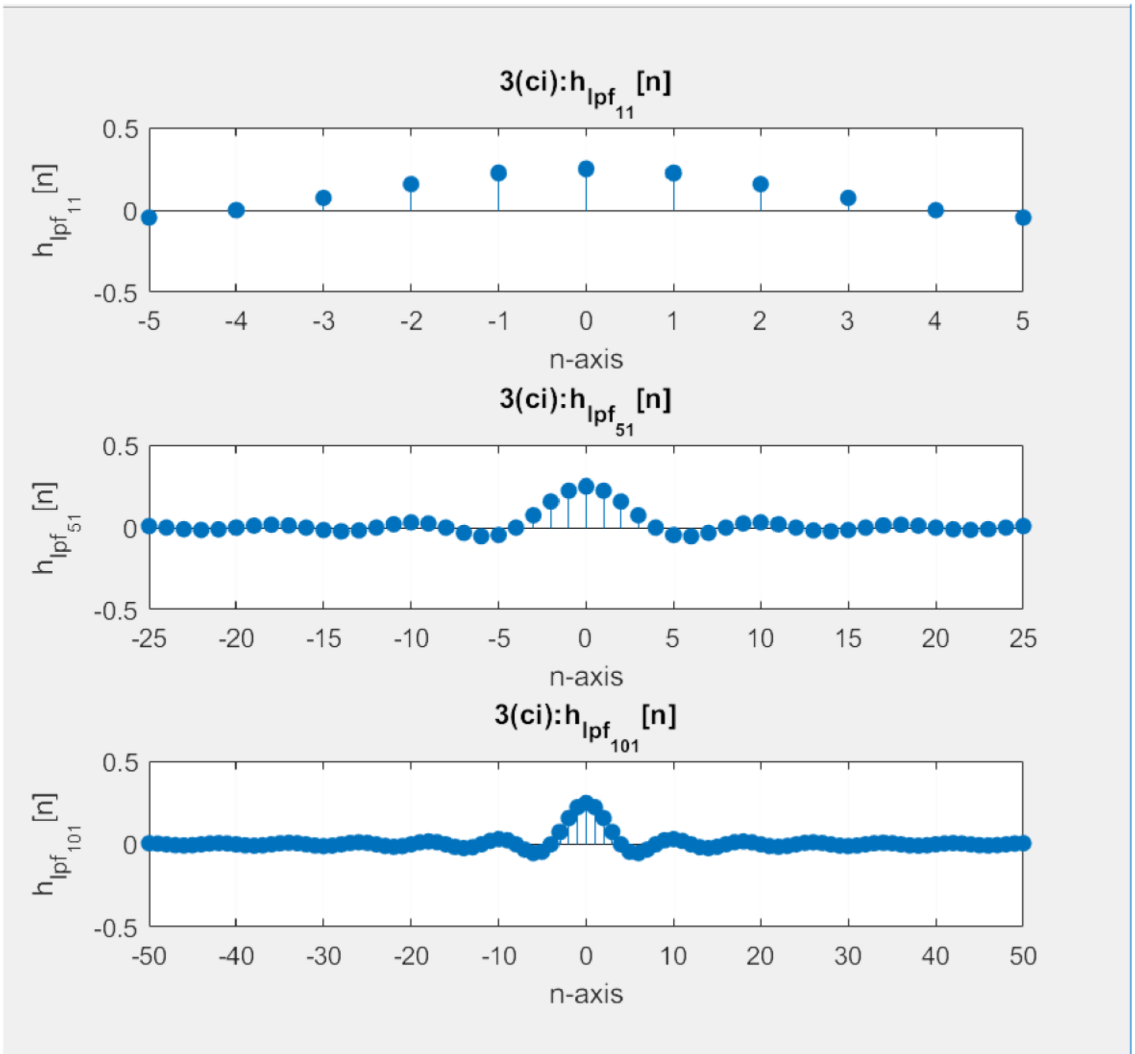
Figure 4 used for comparisons on a 10 log base 10 scale versus Blackman - While not instructed, both partners plotted the windows on a log scale since the Blackman window was instructed to be plotted on a log scale as shown above in figure 4. The longer the window, the shorter the transition band. Not only that, but we also have more ripples. The more ripples we have in our window frequency response, the better our window is at attenuating signals in its stopband when we multiply the window with an ideal sinc function.

3. FIR Filter Design via the Window Method

(c) Create 3 lowpass filters using 3 windows with a cutoff frequency of pi/4

(i) Use the stem and subplot commands to plot the impulse response of the 3 filters you defined.

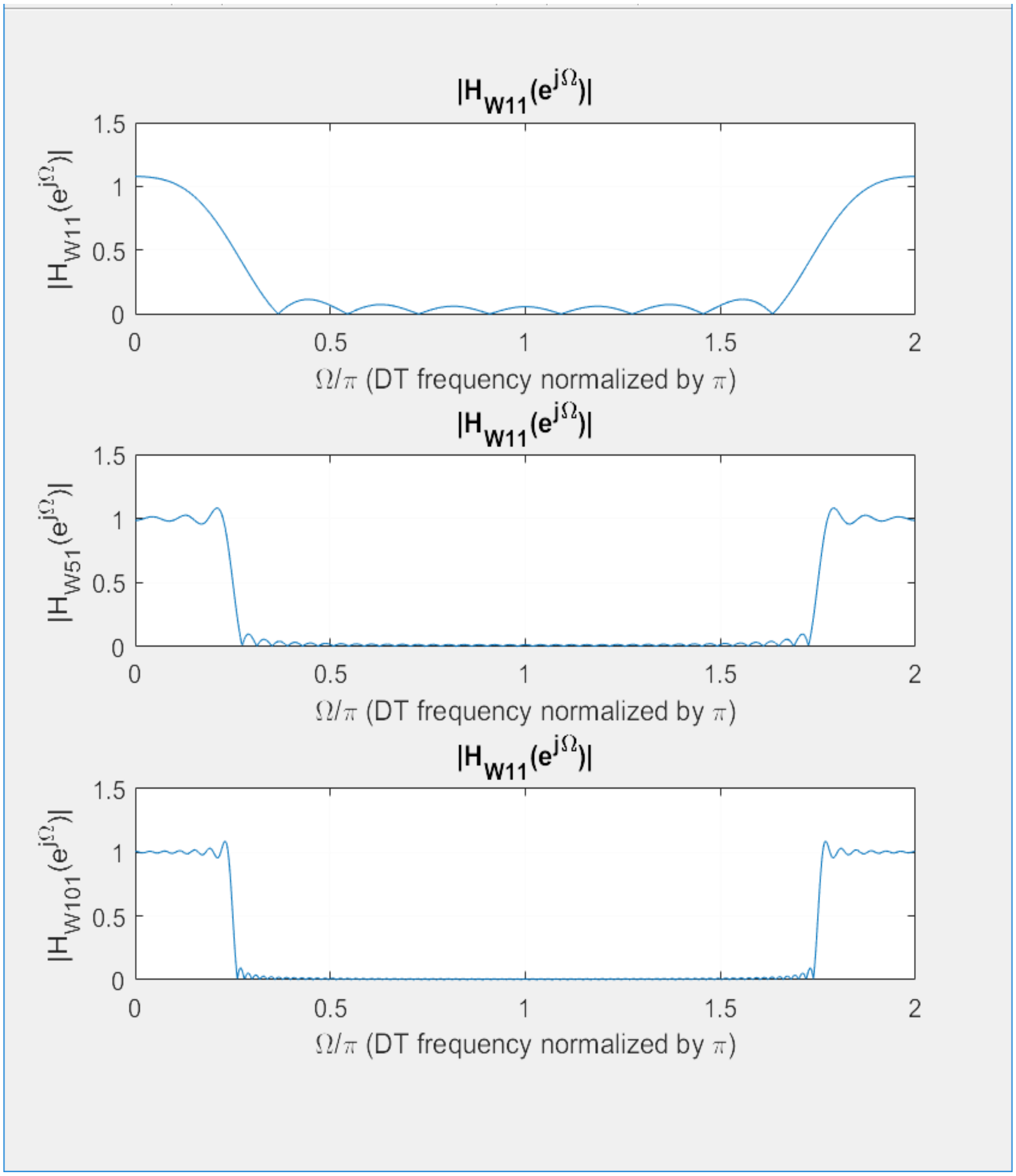
Figure 5



3. FIR Filter Design via the Window Method

(ii)Use the fft or freqz command to find the frequency response of the 3 filters. Plot these on 3 subplots.

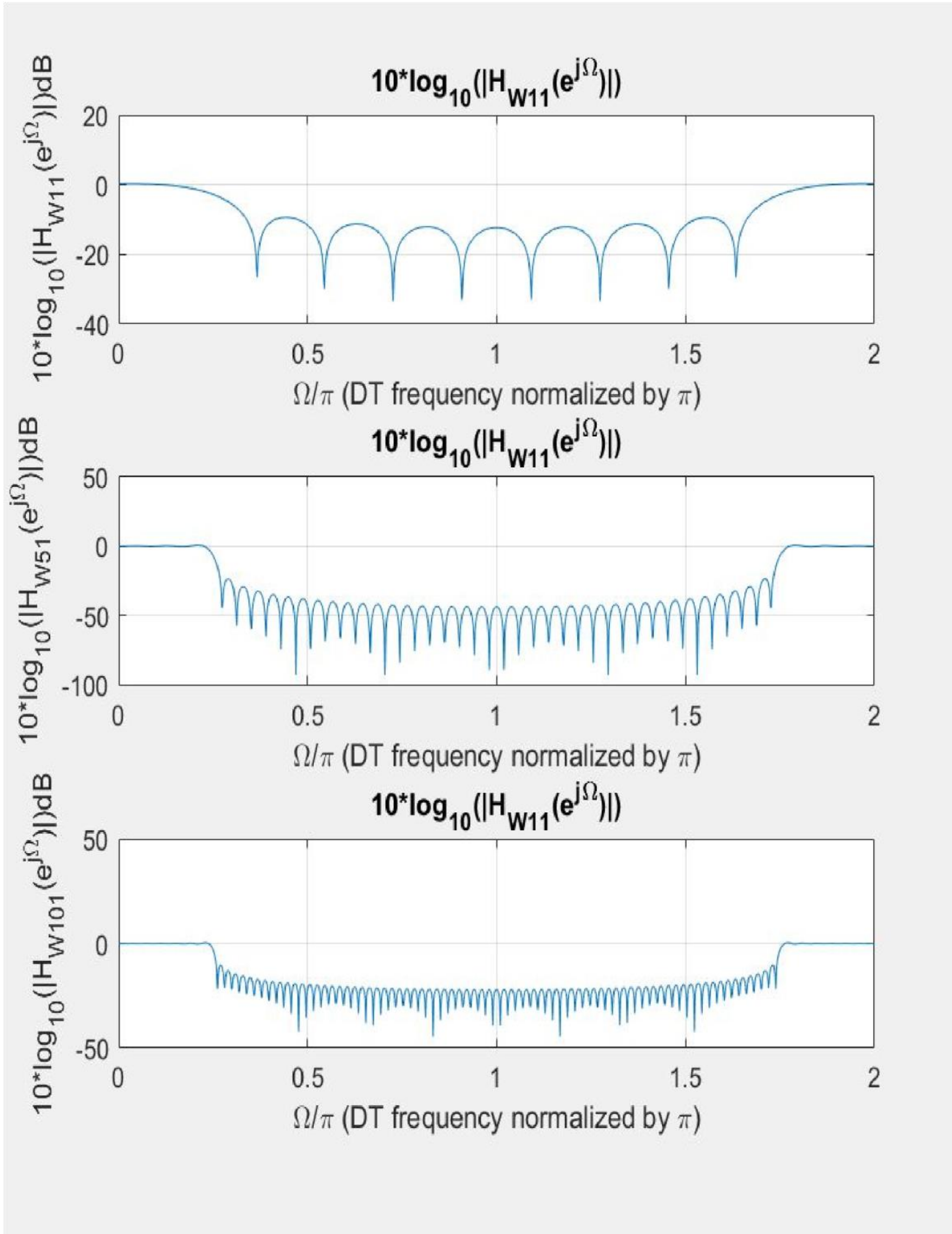
Figure 6



3. FIR Filter Design via the Window Method

(ii)*continued*

Figure 7



10*log10(|HW11(ejΩ)|)
10*log10(|HW11(ejΩ)|)dB
20
0
-20
-40
0
0.5
1
1.5
2
Ω/π (DT frequency normalized by π)
10*log10(|HW11(ejΩ)|)
10*log10(|HW51(ejΩ)|)dB
50
0
-50
-100
0
0.5
1
1.5
2
Ω/π (DT frequency normalized by π)
10*log10(|HW11(ejΩ)|)
10*log10(|HW101(ejΩ)|)dB
50
0
-50
0
0.5
1
1.5
2
Ω/π (DT frequency normalized by π)

3. FIR Filter Design via the Window Method

(iii)Compare the 3 filters and describe their frequency responses. Are the results what you were expecting? How does the length of the window affect the frequency response?

As shown in the figure above, the frequency response looks more and more "box" like the longer the window becomes. Thus, it can be concluded that the the more data points we take for a sinc function (i.e., the longer the window is when we multiply it with an infinitely long sinc function), the more it will look like an ideal lowpass filter. However, such a function cannot be implemented in real life. We cannot have an infinitely long sinc function in MATLAB. So, we do the next best thing and make a finite sinc function.

We can see that the shorter the window is, the less ideal it looks when compared to the ideal lowpass filter. This is to be expected; since if we do not have enough data points for our sinc, the frequency response will not appear as a lowpass filter (or to be more precise, it would not be a very good lowpass filter.),

Increasing the length of window from 11 points to 51 points dramatically improves the frequency resolution. $H_{W51}(e^{j\Omega})$looks much more like an ideal lowpass filter than $H_{W11}(e^{j\Omega})$ does. The transition band from $H_{W51}(e^{j\Omega})$ compared to $H_{W11}(e^{j\Omega})$is much sharper than the wide transition band from $H_{W11}(e^{j\Omega})$. When we look at $H_{W101}(e^{j\Omega})$the transition band is ever sharper than $H_{W101}(e^{j\Omega})$. Thus, it can be concluded that the longer the window is, the sharper the transition band will be when we multiply a long window to an infinite sinc function to truncate it.

When we view the response on a 10 times log base 10 scale, the differences become even more apparent between the different length filters. The cutoff of the frequency response is 0.25, which is what we expected. The longer the window, the more apparent the cutoff frequency becomes. We can see that when we increase the length of the frequency response (meaning, when we multiply our infinite sinc with a longer window) the ripples in the stopband become much more narrow. This is also to be expected, as we know from our analytical calculations from the window that we obtain more zero crossings in the frequency response of the window the wider we make it.

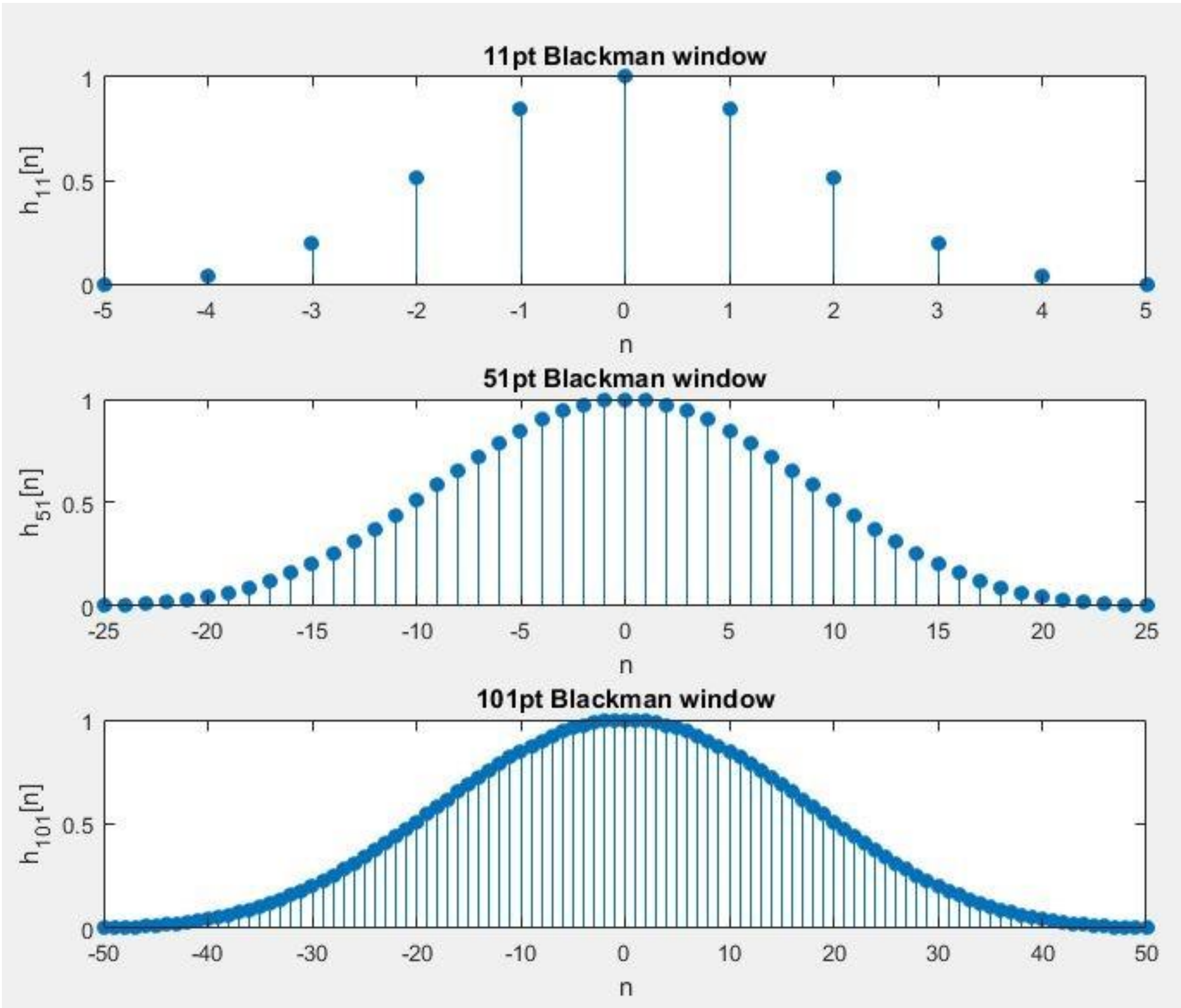
3. FIR Filter Design via the Window Method

(d) Designing Lowpass filter with Blackman Window.

i) Generate 3 Blackman windows of lengths 11, 51, and 101. Comment on how these windows look compared to the rectangular window.

Figure 8 shows impulse responses of 3 Blackman windows of lengths 11,51 and 101 points. The Blackman windows have a bell shape to them compared to the rectangular windows. Blackman windows have a "bell" shape, with gradual transitions from low to high. Rectangular windows look like "boxes", with sharp transitions.

Figure 8



(d) Designing Lowpass filter with Blackman Window.

ii) Plot the frequency response magnitudes of the 3 Blackman windows and compare them to the frequency responses of the rectangular windows you made previously.

Figure 9 shows the plots of the frequency response magnitudes of three Blackman windows. Comparing these plots to frequency response of rectangular windows figure 3 we can tell that frequency response of Blackman windows is “smoother” than frequency response of rectangular windows. The main lobes are wider and there are no ripples in stopband.

3. FIR Filter Design via the Window Method

Figure 9

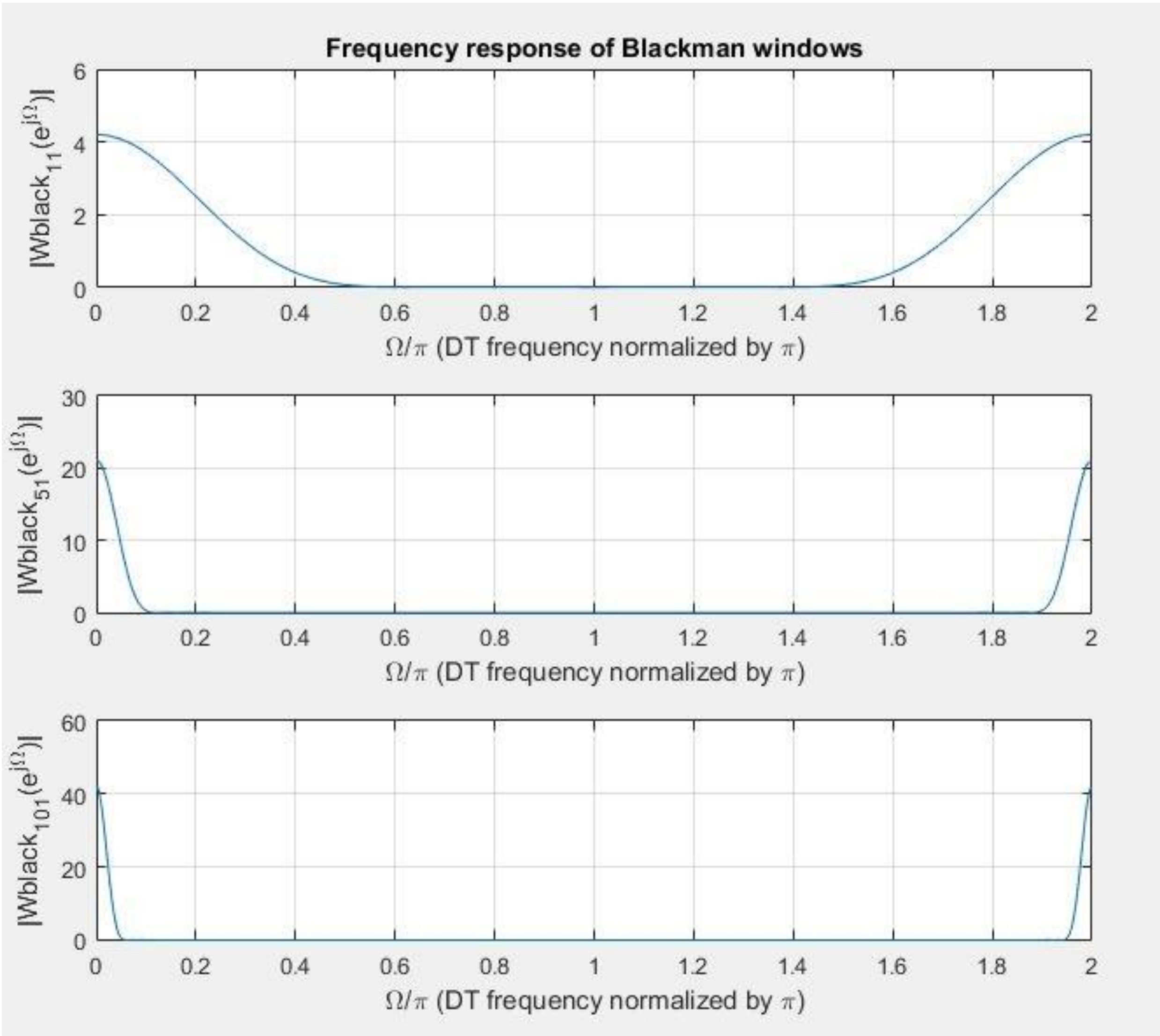
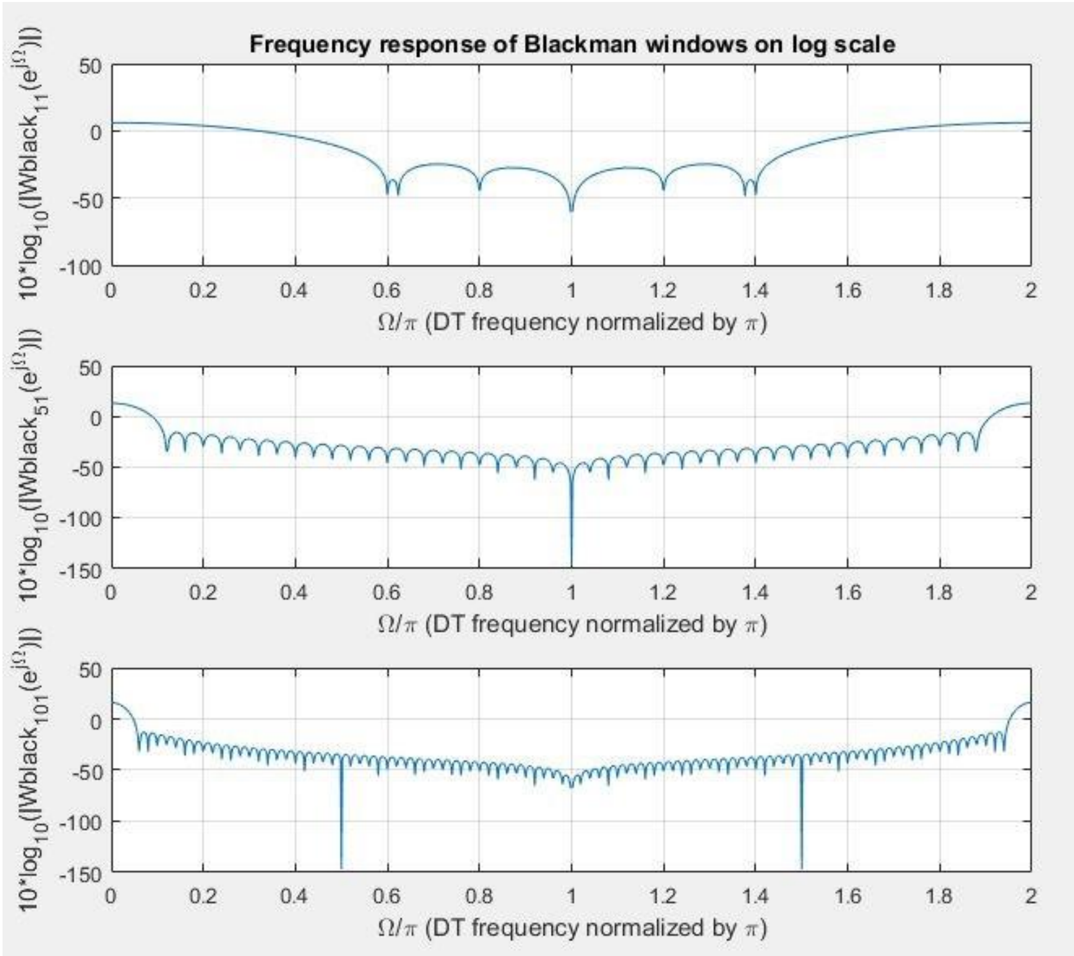


Figure 10 shows the plots of same frequency response but on a log scale. When we compare figure 11 with the log scale of the windows shown in figure 4, we can see that the Blackman frequency response on a log scale is "smoother." The transition band is much wider for the Blackman window, and there are less ripples in the stopband than in figure 4 log scale frequency response. The stopband of the log Blackman window dips to -25dB, while the rectangular window in a log scale stays around -10dB to -15dB depending on the length of the rectangular window.

Figure 10



(d) Designing Lowpass filter with Blackman Window.

iii)Use the 3 Blackman windows to design lowpass filters with a cutoff of pi/4 and lengths 11, 51, and 101. Plot the frequency response and compare them to the rectangular windows. Were your predictions correct?

Figure 11 shows impulse response of lowpass filters designed with Blackman windows.

3. FIR Filter Design via the Window Method

Figure 11

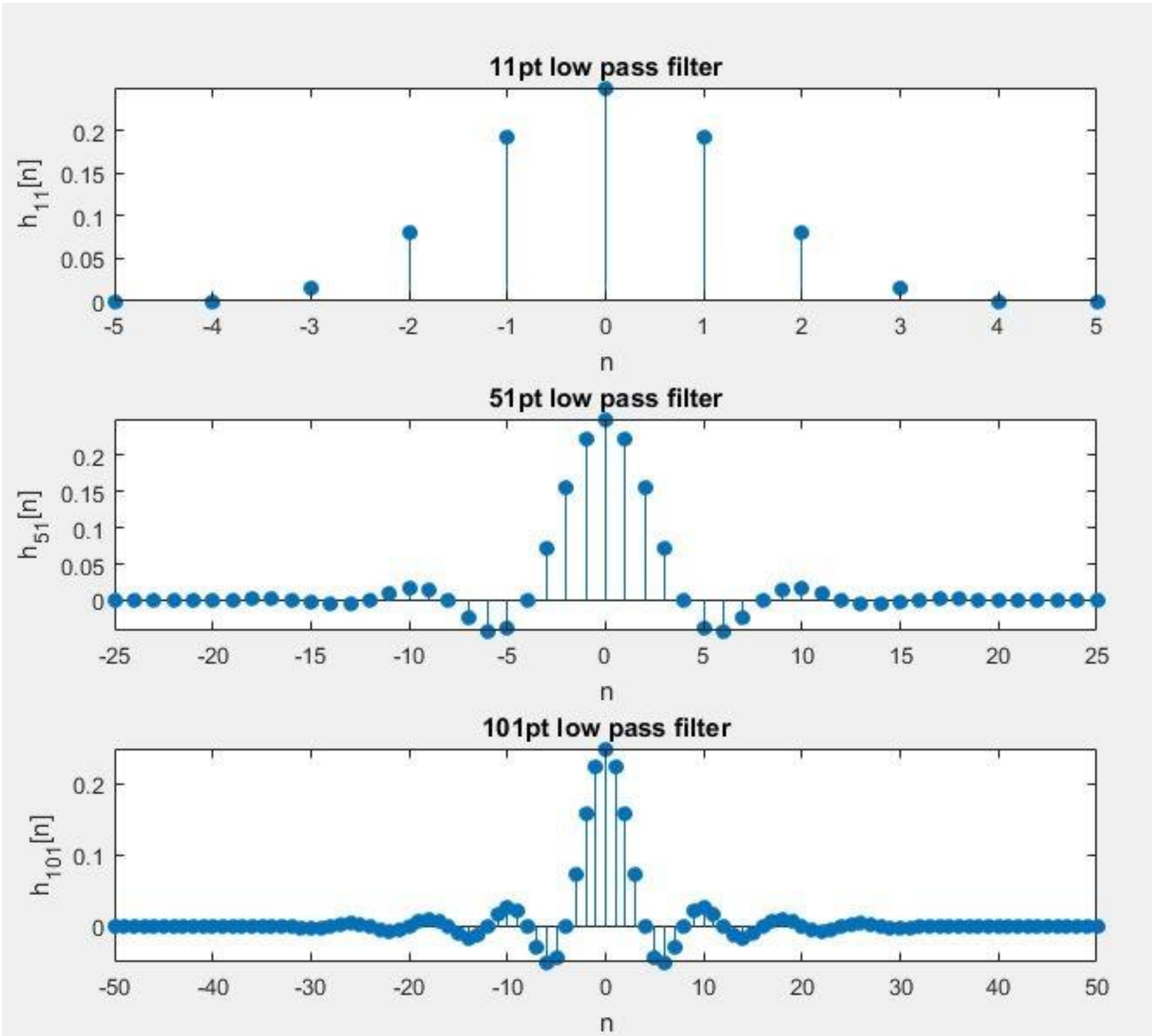


Figure 12 shows the frequency response magnitudes of lowpass filters designed with Blackman windows. The cutoff frequency is ?/4. As expected, these plots look “smoother” comparing to those of rectangular window in figure 6. There are less ripples in both passband and stopband. Comparing the 3 plots to each other, we can tell that the transition band of the filter decreases with the length of the window increasing. Also for 11 points window, the magnitude of the frequency response of the filter does not reach 1 in passband.

Figure 12

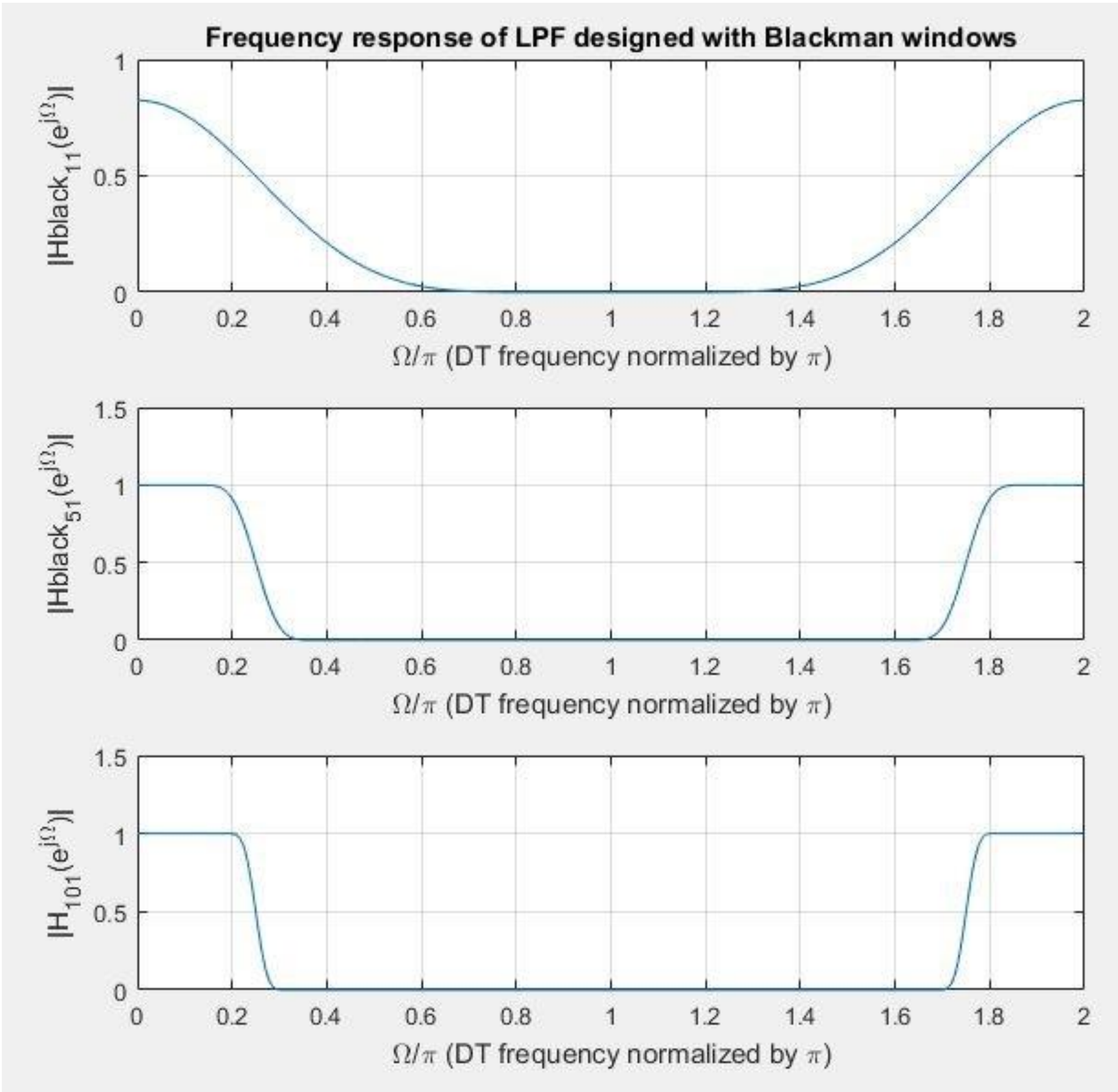
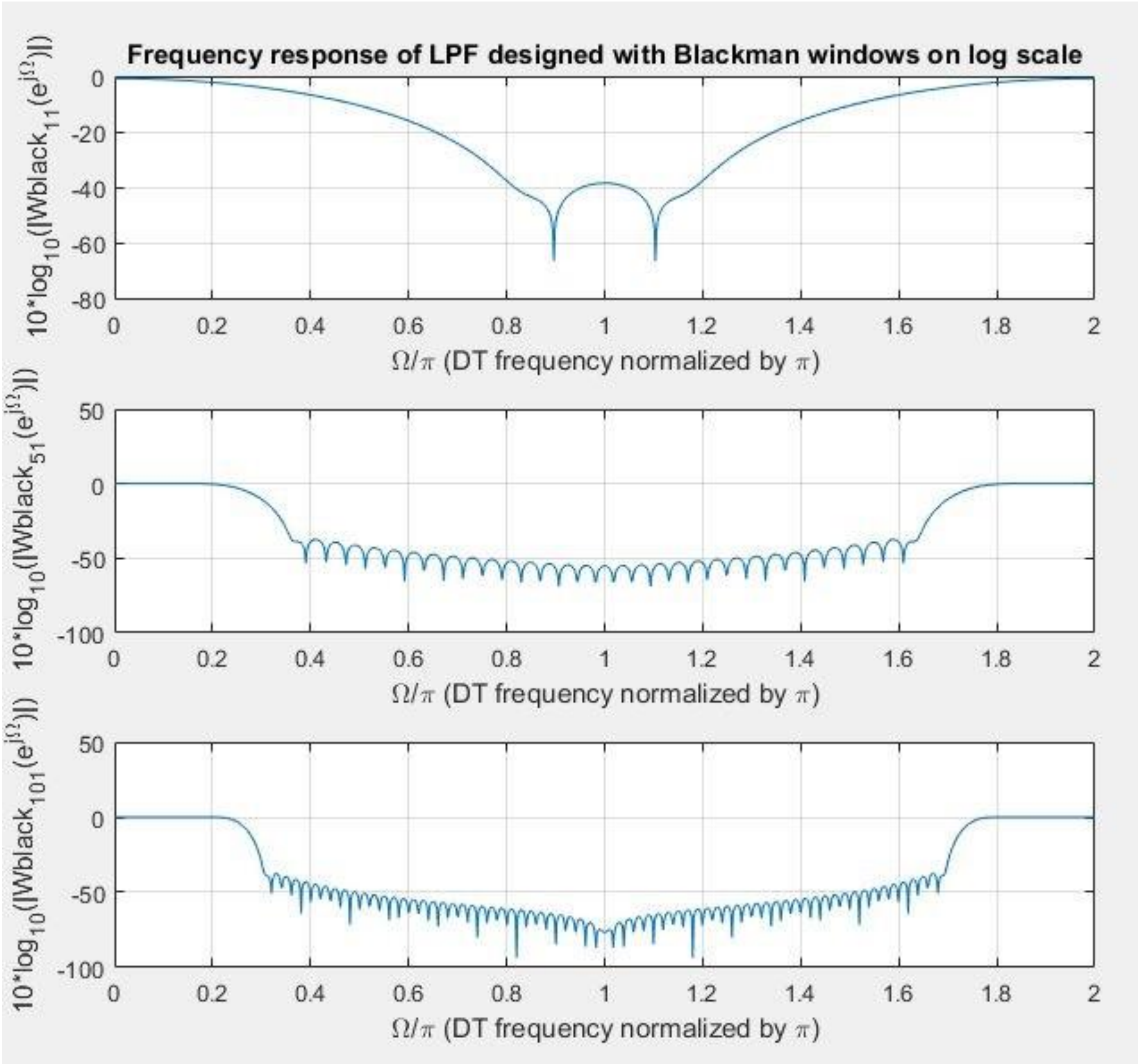


Figure 13 shows the plots of same frequency response shown in figure 12, but on a log scale. When we compare figure 13 to figure 7 (the log scale LPF of the rectangular window), we can make similar conclusions like we did between figure's 12 and 6. The transition band for the Blackman window is much wider, even on a log scale, than the LPF rectangular window.

The stopband for figure 13 dips to roughly to -50dB to -75dB depending on the length of the LPF Blackman window length. When we compare this to figure 7, the stopband only dips between -20dB to -40dB depending on the length of the window and where you look at the ripples.

Figure 13



iv)Discuss the relative merits of the rectangular window design versus the Blackman window design.Does one type of filter perform better than the other? If so, how? Comment on how you would decide which window to use.

Comparing lowpass filters designed with rectangular windows and with Blackman windows, it can be seen that frequency response of LPFs designed with rectangular windows have higher ripples in both passband and stopband. On the other hand, LPFs designed with Blackman windows have longer transition band. Hence, in case when we need lower ripples in passband and stopband we will choose Blackman window design. In case if we need smaller transition band, meaning that we need the filter to be more accurate around cutoff frequency, we will prefer the rectangular window design.

3. FIR Filter Design via the Window Method

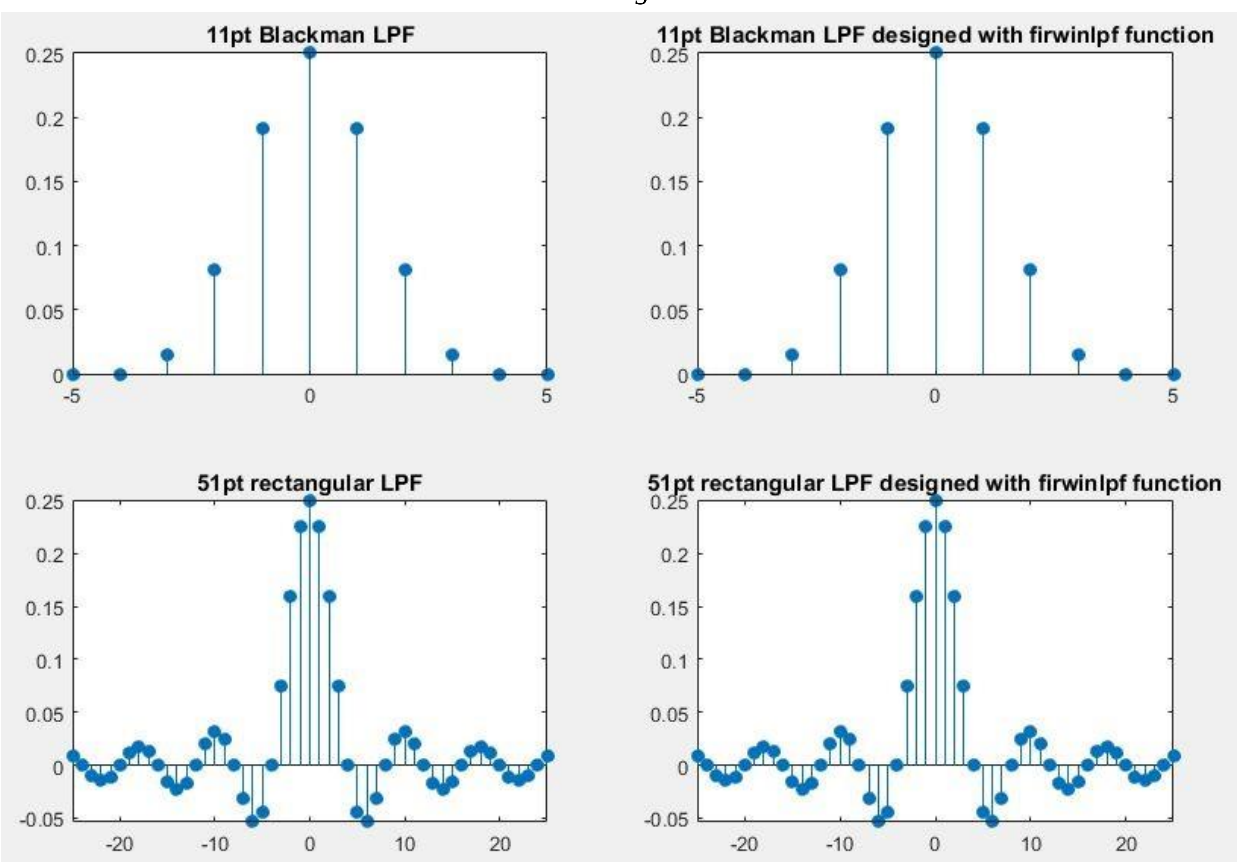
(e) Write a function to implement window design of lowpass filters. Following is the MATLAB code for firwinlpf function.

```
function [h,n] = firwinlpf(wc,M,winfxn
)     n=-M:1:M;     win=
feval(winfxn,2*M+1);
hlpf=sin(wc*n)./(pi*n);
hlpf(M+1)=(wc/pi);
h=transpose(win).*hlpf; end
```

(f) Test your function

To test the function, we called it to create the same filters that we created in the previous parts of this project. Then we compared the impulse responses of these filters. Figure 14 shows the results of the testing. On the left we have the impulse response of the filters designed earlier and on the right - impulse response of filters designed with function firwinlpf. As we can see these plots match. Hence, the function works as expected.

Figure 14



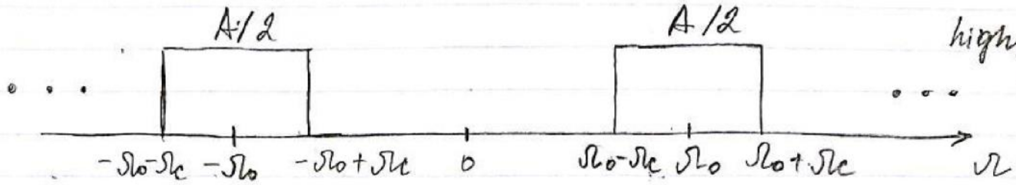
4. Design of Bandpass and Highpass FIR Filters

a) **Determine an analytical expression for $H_{new}(e^{j\Omega})$ in terms of $H(e^{j\Omega})$**

Figure 15

a) $h_{new}[n] = A\, h[n] \cos(\Omega_0 n)$

$$H_{new}(e^{j\Omega}) = \frac{1}{2\pi} A\, H(e^{j\Omega}) * [\pi(\delta(\Omega-\Omega_0) + \delta(\Omega+\Omega_0))]$$



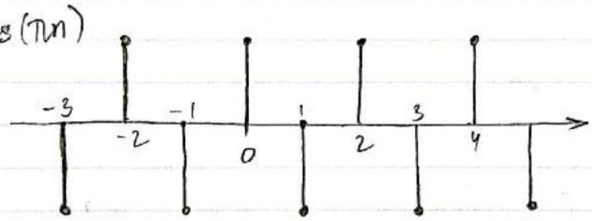
highpass filter

$$H_{new}(e^{j\Omega}) = \frac{A}{2}[H(e^{j(\Omega-\Omega_0)}) + H(e^{j(\Omega+\Omega_0)})]$$

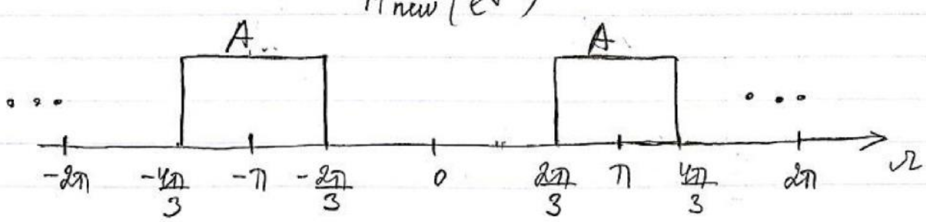
b) **Assume $\Omega_c = \pi/3$ and $\Omega_0 = \pi$, find the frequency response**

b) $\Omega_c = \frac{\pi}{3}$ $\quad \Omega_0 = \pi$

$\cos(\pi n)$



$H_{new}(e^{j\Omega})$



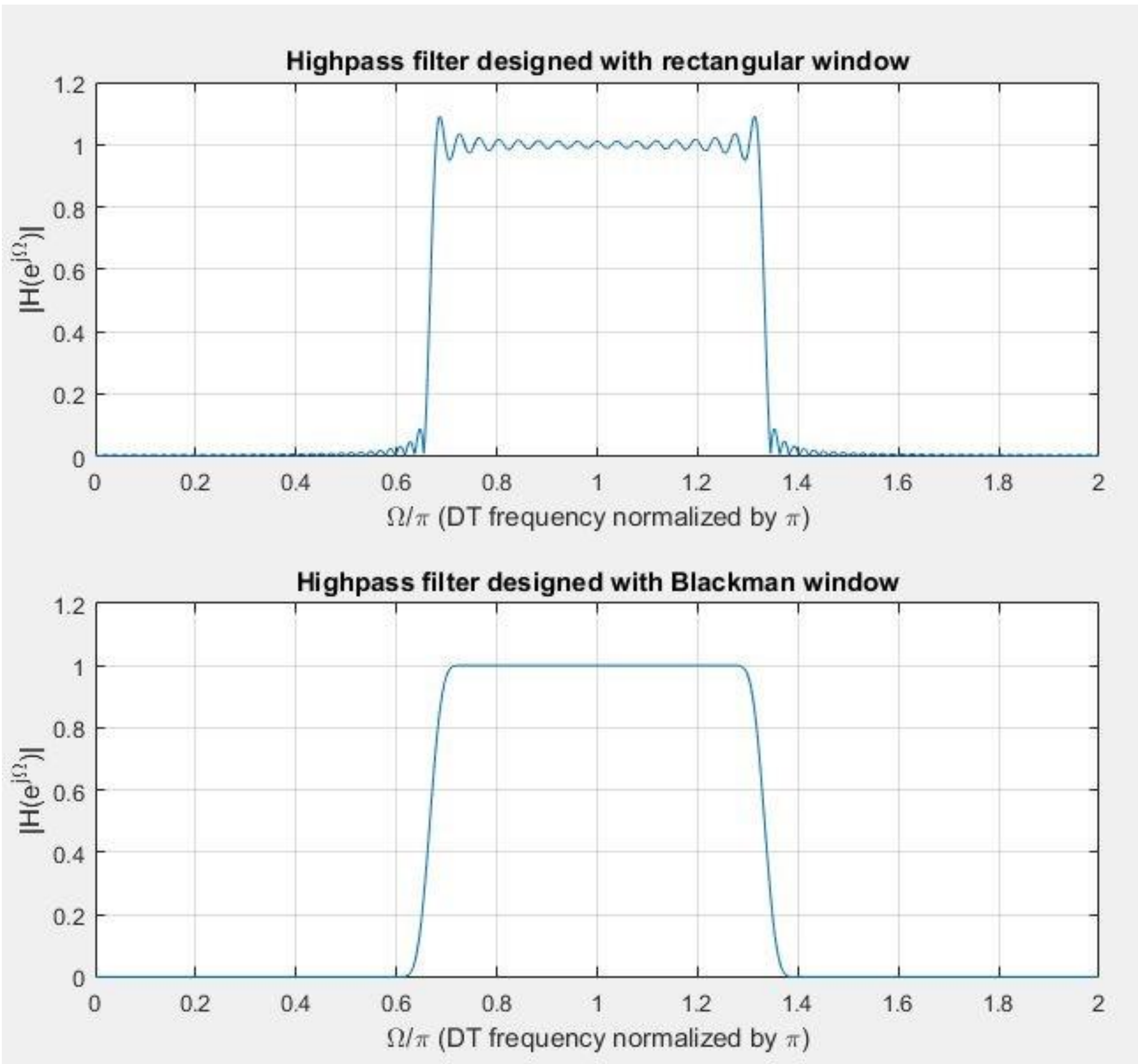
Since $H_{new}(e^{j\Omega})$ is 2π periodic we will get two identical "boxes" around π. Adding them up will get amplitude A

4. Design of Bandpass and Highpass FIR Filters

c) Use the *firwinlpf* to design a 101 point lowpass filter

Since the instructions didn't specify which window design to use, we generated lowpass filters with both rectangular window and Blackman window. Figure 16 shows the magnitude of frequency response of these filters. We can see that in both cases filter passes the frequencies around ⍰, which considered to be the highest frequency for DTFT. The cutoff frequency is 2⍰/3. These results match our analytical solutions.

Figure 16

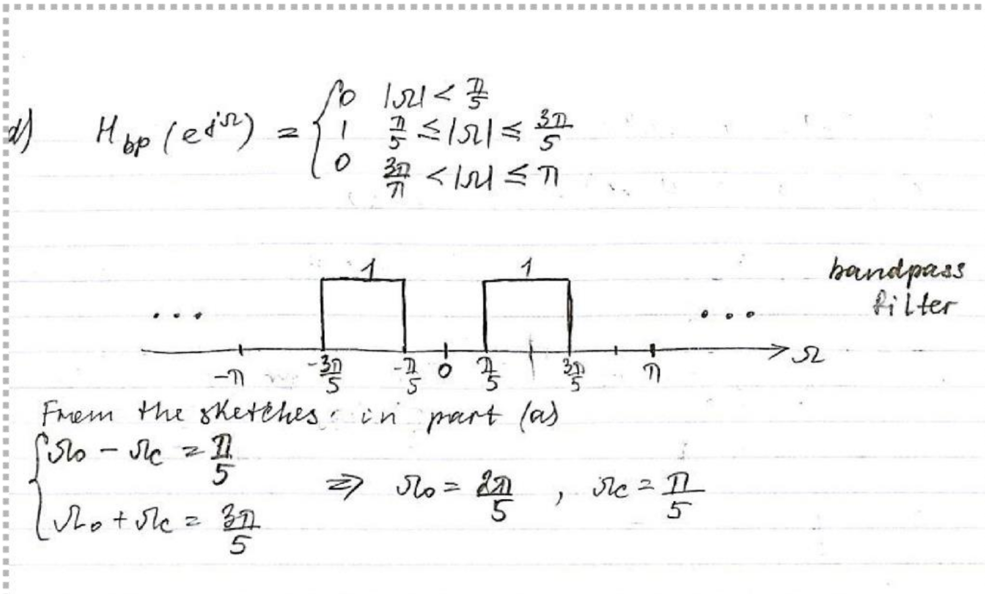


4. Design of Bandpass and Highpass FIR Filters

d) Bandpass and Highpass Filters

i) Sketch $H_{BP}(e^{j\Omega})$ for $-\pi \le \Omega \le \pi$

Figure 17



ii) What cutoff frequency and modulation frequency would you choose?

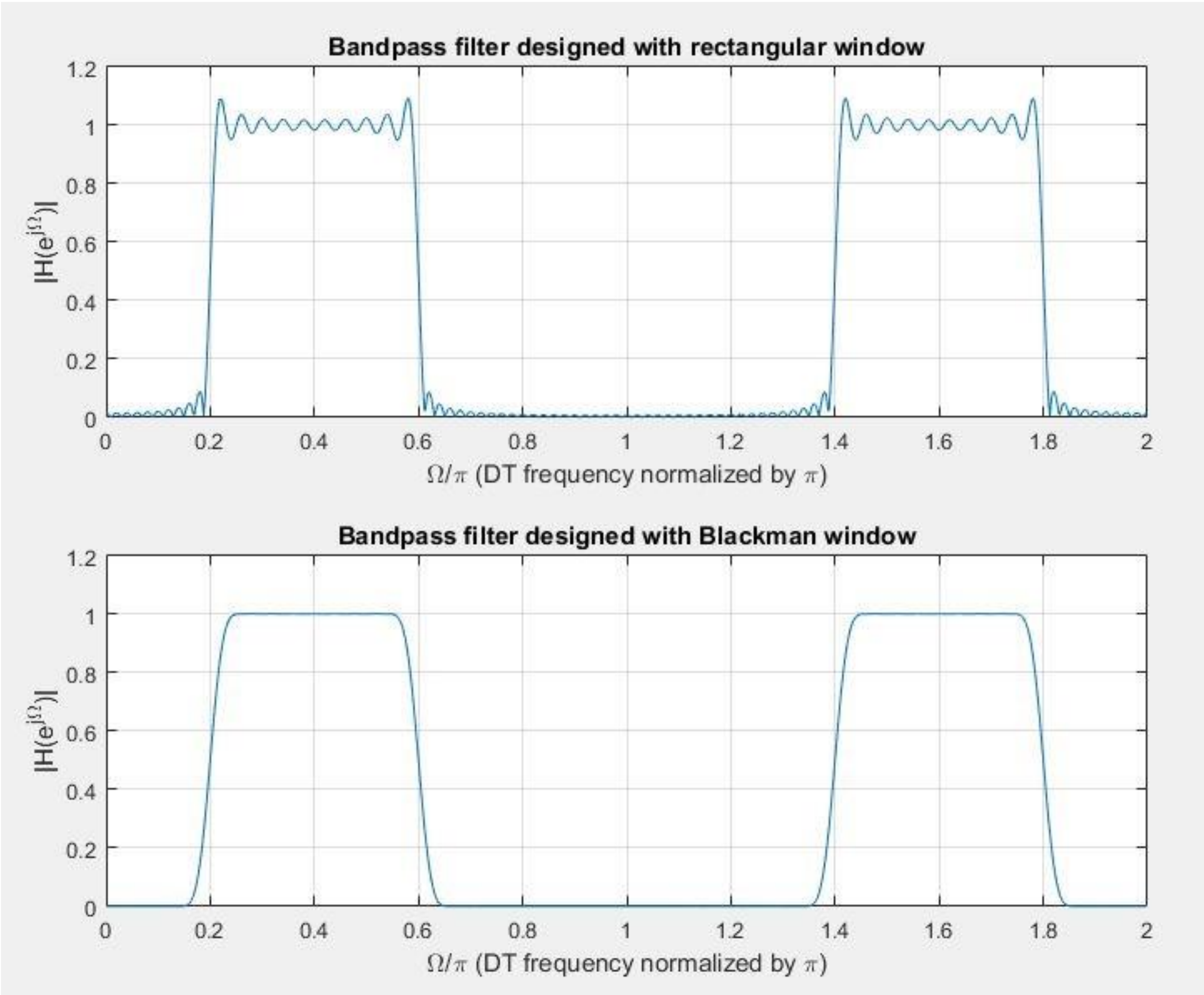
In figure 17 w want to modulate the signal at 2pi/5; meaning, we want to create two copies of our lowpass filter centered at 2pi/5 that extends to 3pi/5. To extend to 3pi/5 and pi/5, we choose the cutoff frequency of our sinc to be pi/5.

iii) Using *firwinlpf*, create the necessary filter expressed from the previous section

For a bandpass filter we determined that the cutoff frequency Ω_c of a lowpass filter must be ⍰/5 and modulation frequency Ω_0 = 2⍰/5. We also calculated that scale factor A = 2. Figure 18 Shows the plots of frequency response magnitude of two bandpass filters designed with rectangular and Blackman windows. These plots match our analytical solutions expressed in figure 17.

4. Design of Bandpass and Highpass FIR Filters

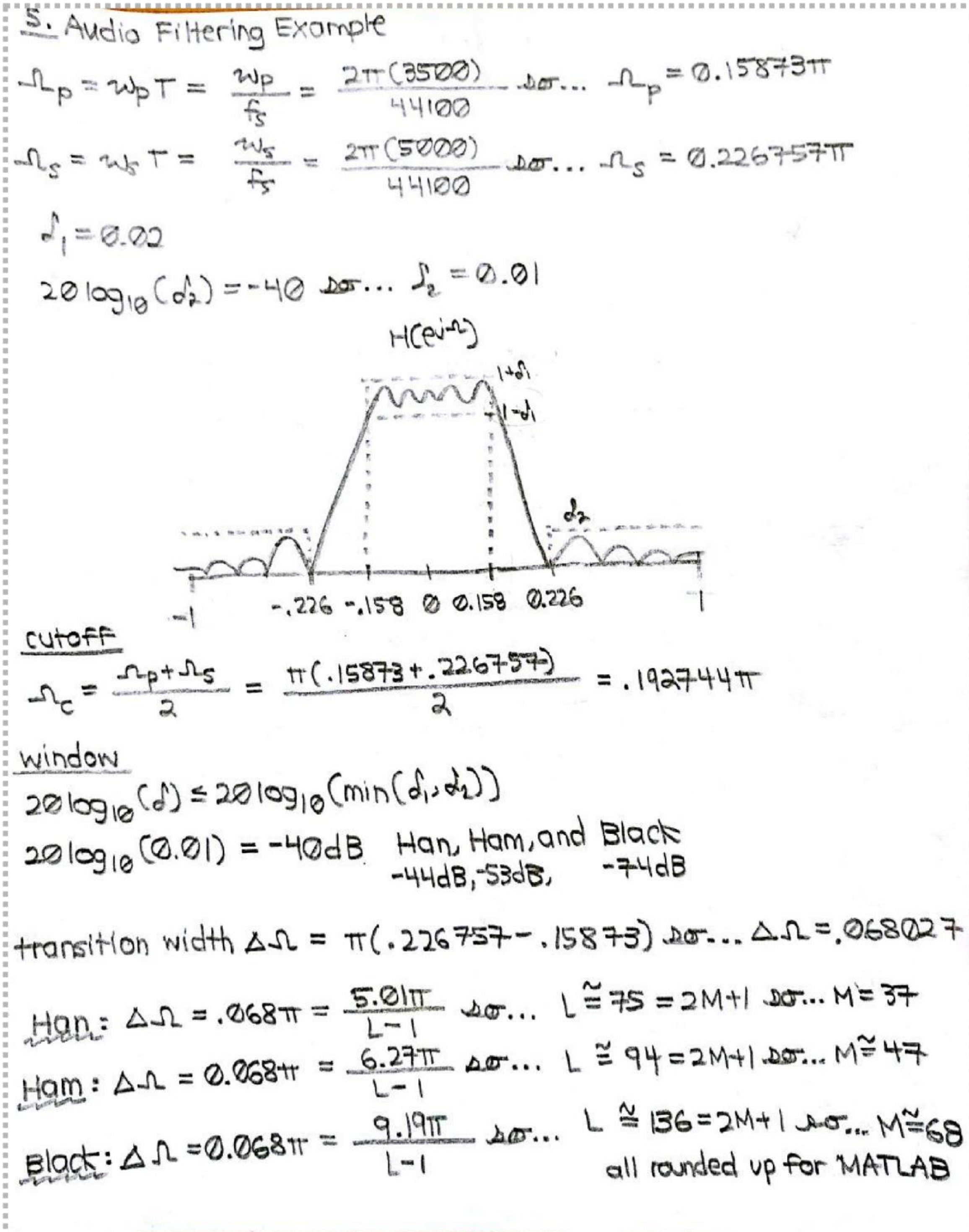
Figure 18



5. Audio Filtering Example

(a) Determine and sketch the specifications of the discrete-time lowpass filter. Sketch the tolerances for $H(e^{j\Omega})$ **for** $-\pi \leq \Omega \leq +\pi$

Figure 19



5. Audio Filtering Example

$\Omega_p = \omega_p T = \frac{\omega_p}{f_s} = \frac{2\pi(3500)}{44100}$ so... $\Omega_p = 0.15873\pi$

$\Omega_s = \omega_s T = \frac{\omega_s}{f_s} = \frac{2\pi(5000)}{44100}$ so... $\Omega_s = 0.226757\pi$

$\delta_1 = 0.02$

$20\log_{10}(\delta_2) = -40$ so... $\delta_2 = 0.01$

cutoff

$\Omega_c = \frac{\Omega_p + \Omega_s}{2} = \frac{\pi(.15873 + .226757)}{2} = .192744\pi$

window

$20\log_{10}(\delta) \leq 20\log_{10}(\min(\delta_1, \delta_2))$

$20\log_{10}(0.01) = -40dB$ Han, Ham, and Black
-44dB, -53dB, -74dB

transition width $\Delta\Omega = \pi(.226757 - .15873)$ so... $\Delta\Omega = .068027$

Han: $\Delta\Omega = .068\pi = \frac{5.01\pi}{L-1}$ so... $L \cong 75 = 2M+1$ so... $M = 37$

Ham: $\Delta\Omega = 0.068\pi = \frac{6.27\pi}{L-1}$ so... $L \cong 94 = 2M+1$ so... $M \cong 47$

Black: $\Delta\Omega = 0.068\pi = \frac{9.19\pi}{L-1}$ so... $L \cong 136 = 2M+1$ so... $M \cong 68$

all rounded up for MATLAB

5. Audio Filtering Example

b) Window Design - Hanning, Hamming, and Blackman

From the requirements for our filters: $20*\log_{10}(\min(\delta_1, \delta_2) = -40$ dB. Based on the table in figure 20, we concluded that Rectangular and Bartlett windows cannot satisfy this requirement. Thus, we only designed lowpass filters using Hanning, Hamming and Blackman windows. The calculations for length of the windows are included in analytical part above.

Figure 20

Window Type	Peak Sidelobe	Approximate ML width	Peak approx. error $20\log_{10}\delta$	Equiv. Trans. width (Kaiser)
Rectangular	-13 dB	$4\pi/L$	-21 dB	$1.81\pi/(L-1)$
Bartlett	-25 dB	$8\pi/(L-1)$	-25 dB	$2.37\pi/(L-1)$
Hanning	-31 dB	$8\pi/(L-1)$	-44 dB	$5.01\pi/(L-1)$
Hamming	-41 dB	$8\pi/(L-1)$	-53 dB	$6.27\pi/(L-1)$
Blackman	-57 dB	$12\pi/(L-1)$	-74 dB	$9.19\pi/(L-1)$

Figure 21 shows the magnitude and phase of frequency response of lowpass filters designed with Hanning, Hamming and Blackman windows.

Figure 21

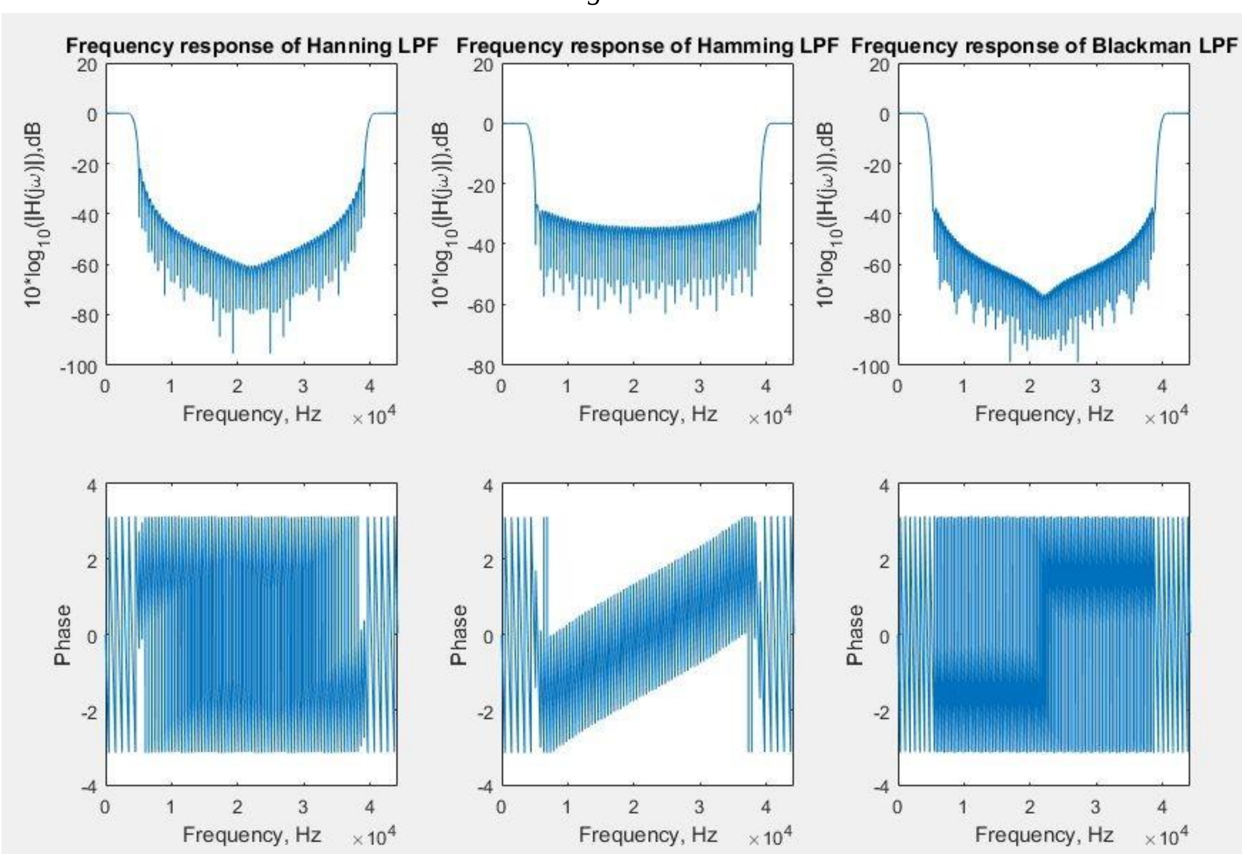
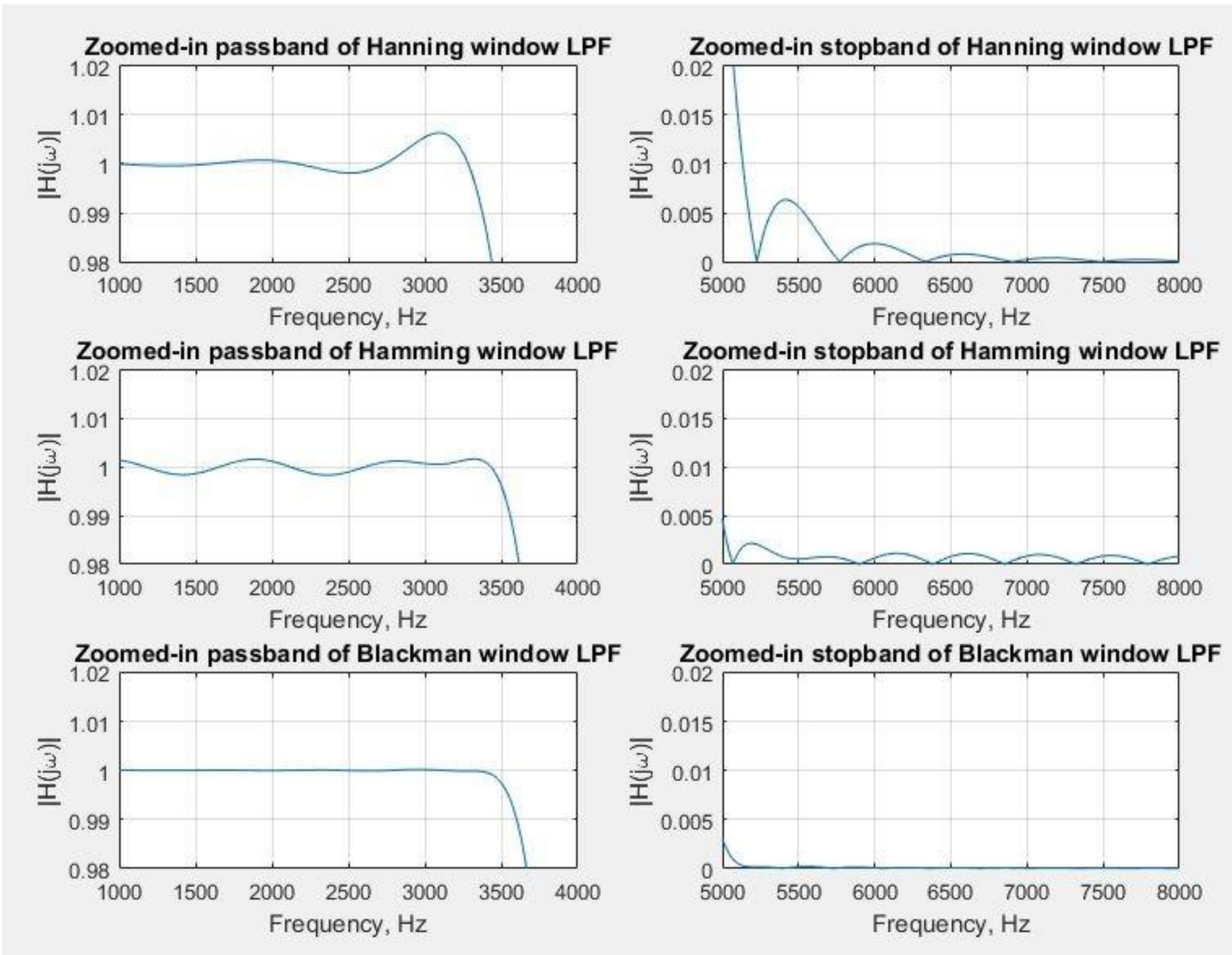


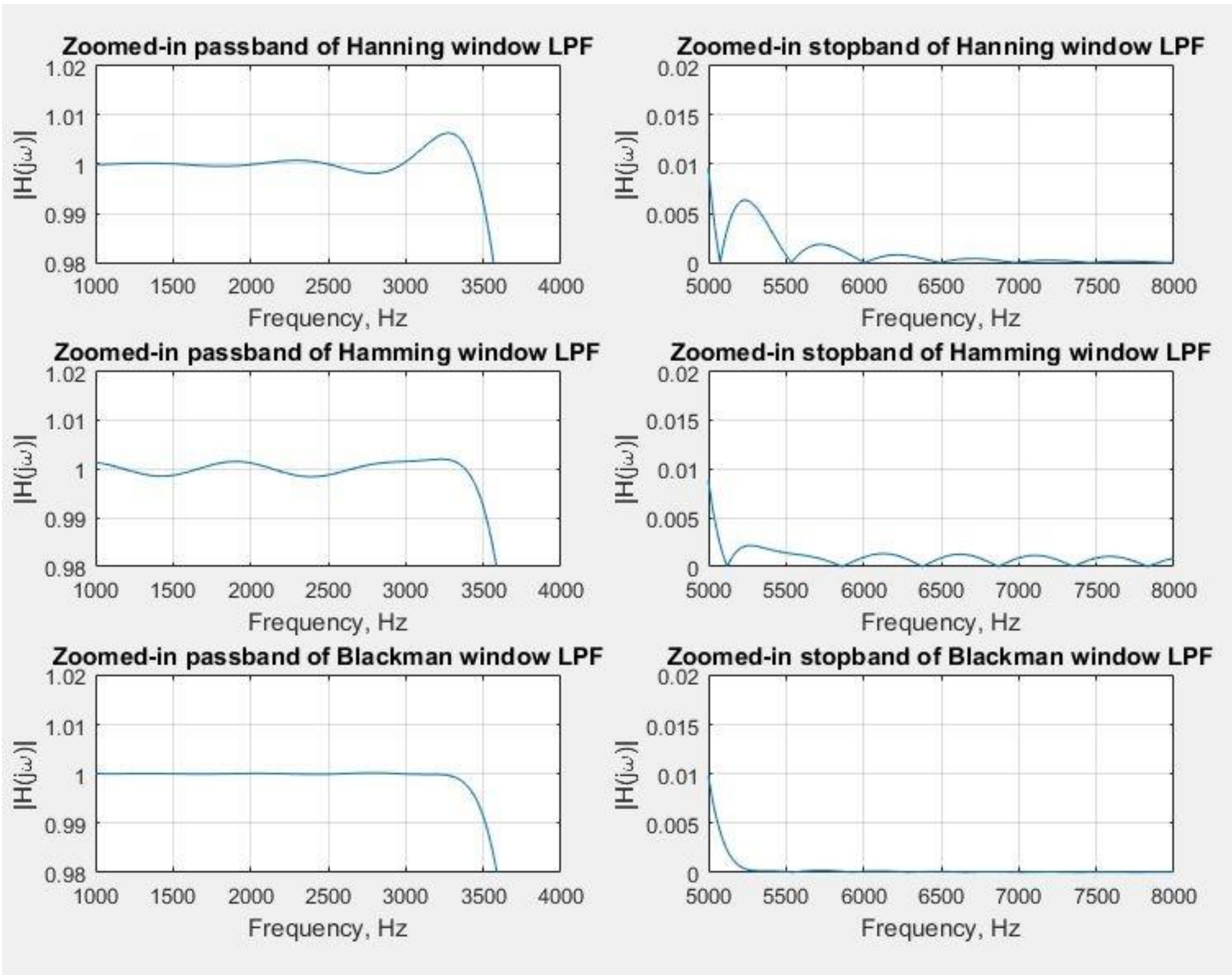
Figure 22 shows the zoomed in passband and stopband of Hanning, Hamming and Blackman window LPFs. For these plots we used the window lengths calculated in analytical part. However, we had to adjust these numbers to satisfy the requirements. For Hanning window we had to increase the length, because the initial value didn't satisfy the requirement for stopband distortion, δ_2. For Hamming and Blackman windows δ_2 for calculated length was much lower than required, which allowed us to decrease the length of the windows.

Figure 22



The new values are L=89 for Hanning window, L=91 for Hamming window, and L=121 for Blackman window. Figure 23 shows the plots for updated values. We can see that these filters satisfy the requirements for both passband and stopband distortion $\square_1$ and $\square_2$.

Figure 23



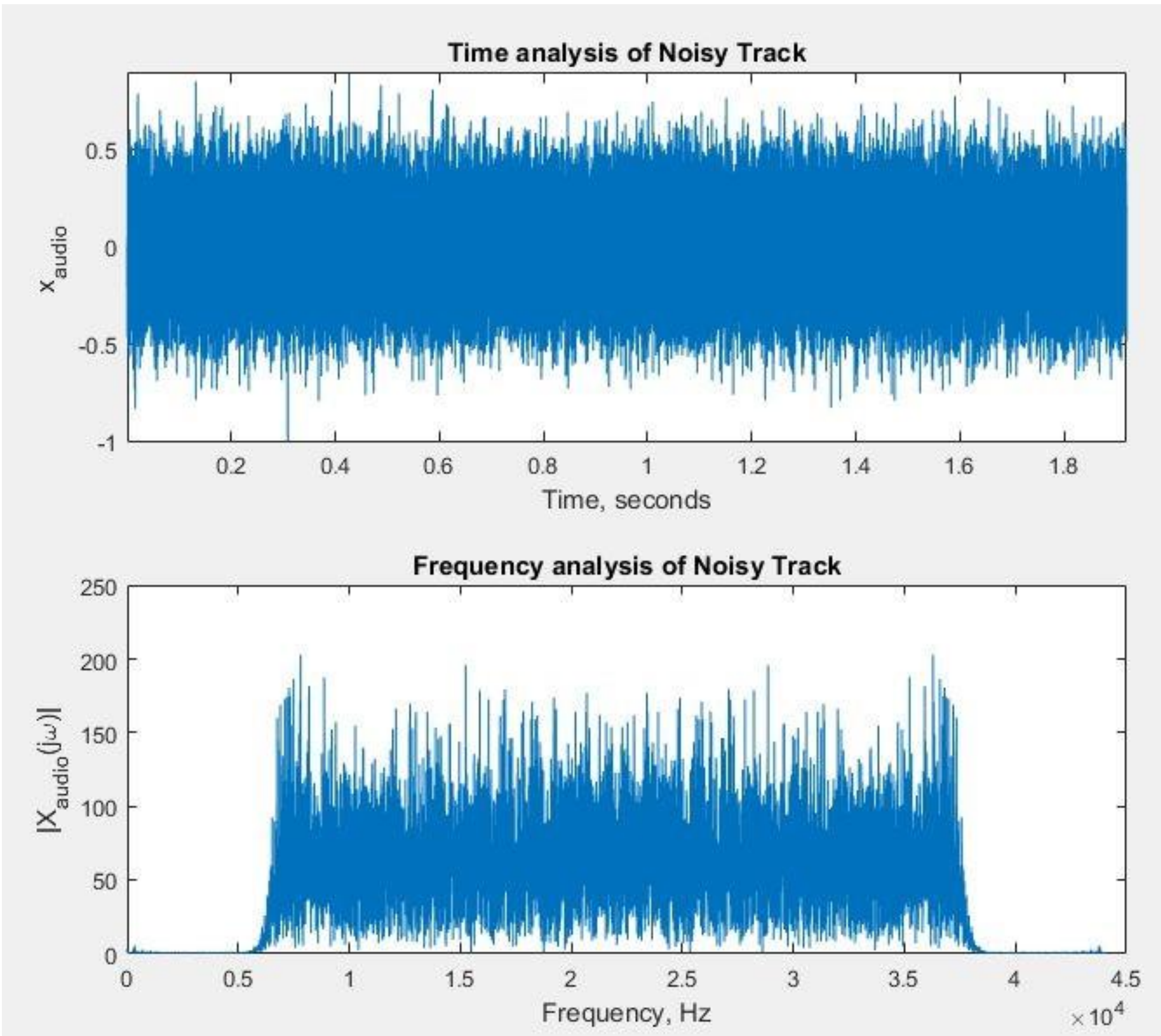
5. Audio Filtering Example

c) Application: filtering the noisy audio signal

(i) load the noisy signal

To understand better the problem we have analyzed Noisy Track in both time domain and frequency domain. It can be seen from figure 24 that the track has a lot of noise, and the magnitude of the high frequency signals(above 5 kHz) is much higher than those of the low frequencies.

Figure 24



5. Audio Filtering Example

c) Application: filtering the noisy audio signal

(ii) Filter the signal with each of the filters you designed in part b. Use the filter command to implement the filtering operation. What does the filtered signal sound like? (Be specific and descriptive)

For all 3 filters, we filter out a significant amount of noise. However, each filter left some degree of noise in the noisy track after we applied the filter. The Hanning window had some noise left in the sound file. It was not very apparent, but it was a low hissing sound. The Hamming window left even more noise in the filtered signal. It was a noticeable hissing sound. The Blackman window was the best at filtering out the noise. When we played the audio file at max volume on both computer speakers and headphones, we heard virtually no hissing noise. The Blackman filter was the best at filtering out the noise compared to the Hanning and Hamming filters.

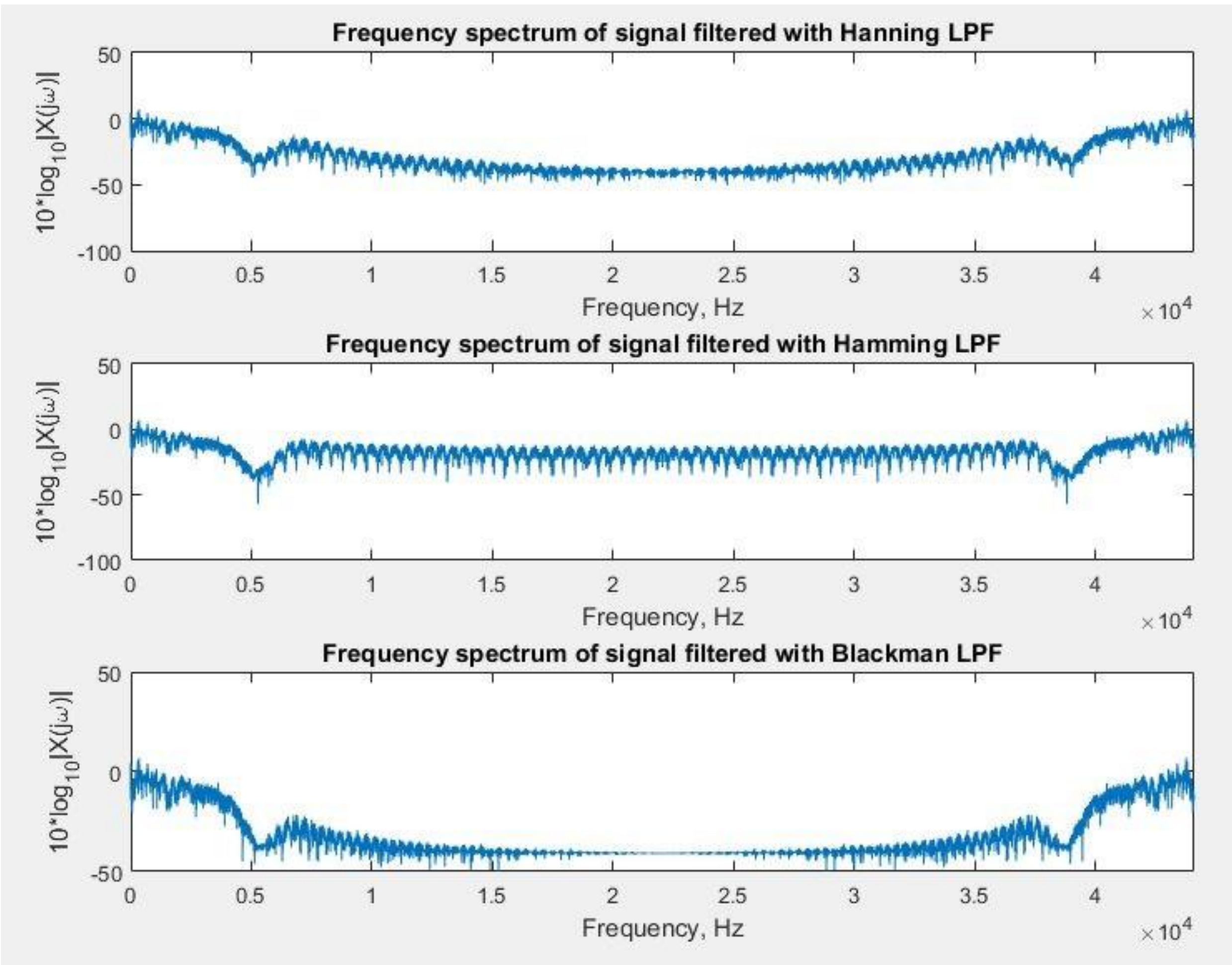
5. Audio Filtering Example

c) Application: filtering the noisy audio signal

(iii) How did your filters perform? Do they remove all the noise? Does one filter perform better than the others or do they all produce identical results? Write up your comparison of the filter performance. If you notice differences, try to explain them based on what you know about the impulse and frequency response of your filters.

While the Hanning and Hamming filters left some noise in the signal, the Blackman filter eliminated most of the noise. Looking at the figure 25, that shows frequency spectrum of three filtered signals, the difference in the filtered audio become very apparent. Hanning and Hamming filters did not attenuate frequencies in stopband as much as Blackman filter, which explains the higher level of noise in the filtered track.

Figure 25



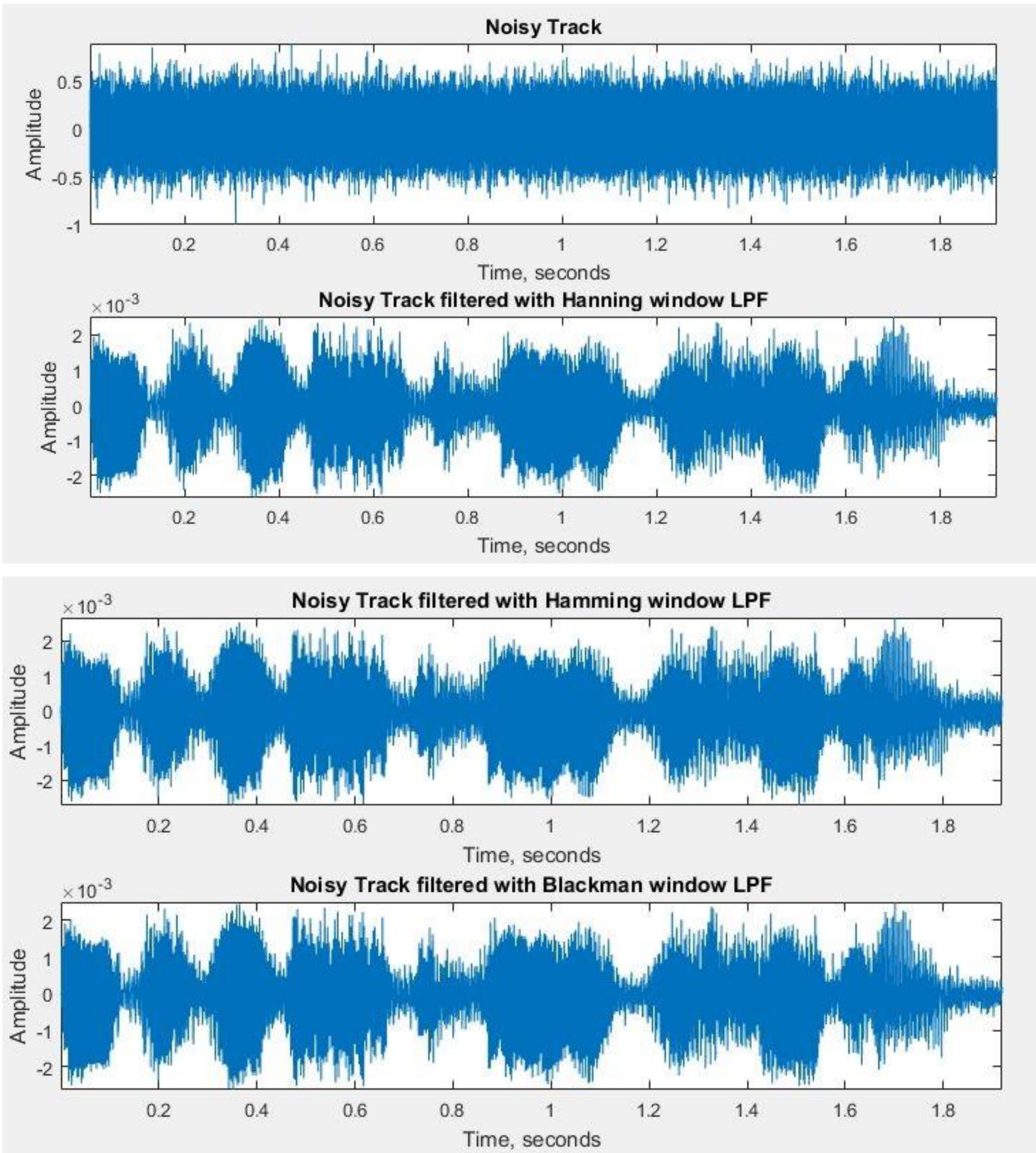
This could be predicted from the frequency response of three filters in figures 23. We can see that Hanning and Hamming filters have higher ripples in stopband than Blackman window filter, which means that they will still pass frequencies in stopband although with a very low gain. Since the magnitude of high frequencies in the noisy track was much higher than those of low frequencies (see figure 24), that gain was not sufficient, and the noise in filtered signals was still noticeable.

5. Audio Filtering Example

c) Application: filtering the noisy audio signal

(iv) Please include plots of the noisy signal and the filtered signals in your report. You should plot them versus an appropriate time vector (remember, the signals are sampled at a rate of fs Hz).

Figure 26



References

MATLAB CODE - Ahad Hassan

```
clear all;
close all;

%------------------------------------------------------------------------
%3 FIR Filter Design via the Window Method
%PART (b) START - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
%declare rect. windows
w11=rectwin(11);
w51=rectwin(51);
w101=rectwin(101);  %plot
rect. windows n_11=-5:5;
n_51=-25:25; n_101=-50:50;
figure(1);  subplot(3,1,1),
stem(n_11,w11);  title('FIG.1-
w_{11}[n]');  xlabel('n-axis');
ylabel('output');
subplot(3,1,2), stem(n_51,w51);
title('FIG.1-w_{51}[n]');
xlabel('n-axis');  ylabel('output');
subplot(3,1,3), stem(n_101,w101);
title('FIG.1-w_{101}[n]'); xlabel('n-axis');
ylabel('output');
%FFTs of rect. windows
W11=fft(w11,1024);
W51=fft(w51,1024);
W101=fft(w101,1024);
%plot FFTs of rect. windows individually
k = 0:1023;  omega = 2*pi*k/1024;
figure(); %figure 2
plot(omega/pi,abs(W11));
title('FIG.2-|W_{11}(e^{j\Omega})|'); xlabel('\Omega/\pi (DT frequency normalized by \pi)');
ylabel('W_{11}(e^{j\Omega})');  grid;  figure(); %figure 3  plot(omega/pi,abs(W51));
title('FIG.3-|W_{51}(e^{j\Omega})|'); xlabel('\Omega/\pi (DT frequency normalized by \pi)');
ylabel('H_{W51}(e^{j\Omega})');  grid;
figure(); %figure 4  plot(omega/pi,abs(W101));
title('FIG.4-|W_{101}(e^{j\Omega})|'); xlabel('\Omega/\pi (DT frequency normalized by \pi)');
ylabel('W_{101}(e^{j\Omega})');  grid;
%subplot the FFTs of the windows
figure(); %figure 5
subplot(3,1,1), plot(omega/pi,(abs(W11)));
```

```
grid; title('|W_{11}(e^{j\Omega})|'); xlabel('\Omega/\pi (DT frequency normalized by \pi)');
ylabel('W_{11}(e^{j\Omega})');  subplot(3,1,2), plot(omega/pi,(abs(W51)));
grid; title('|W_{51}(e^{j\Omega})|'); xlabel('\Omega/\pi (DT frequency normalized by \pi)');
ylabel('W_{51}(e^{j\Omega})');  subplot(3,1,3), plot(omega/pi,(abs(W101)));
grid; title('|W_{101}(e^{j\Omega})|'); xlabel('\Omega/\pi (DT frequency normalized by \pi)');
ylabel('W_{101}(e^{j\Omega})');  %plot all 3 rect. windows together  figure(); %figure 6
plot(omega/pi,abs(W11),'b-',omega/pi,abs(W51),'r--',omega/pi,abs(W101),'g-.');
grid; title('FIG.6-|W_{11}(e^{j\Omega})|, |W_{51}(e^{j\Omega})|, |W_{101}(e^{j\Omega})|');
xlabel('\Omega/\pi (DT frequency normalized by \pi)'); ylabel('W_{11}(e^{j\Omega}),
W_{51}(e^{j\Omega}), W_{101}(e^{j\Omega})');
legend('11pt rectwin','51pt rectwin','101pt rectwin');  xlim([0
0.25]);
%plot all 3 rect. windows on a 10*log_10 scale for comparison  figure();
%figure 7
subplot(3,1,1), plot(omega/pi,10*log10(abs(W11)));
grid; title('10*log_{10}(|W_{11}(e^{j\Omega})|)'); xlabel('\Omega/\pi (DT frequency normalized by
\pi)'); ylabel('10*log_{10}(|W_{11}(e^{j\Omega})|)dB');  subplot(3,1,2),
plot(omega/pi,10*log10(abs(W51)));
grid; title('10*log_{10}(|W_{51}(e^{j\Omega})|)'); xlabel('\Omega/\pi (DT frequency normalized by
\pi)'); ylabel('10*log_{10}(|W_{51}(e^{j\Omega})|)dB');  subplot(3,1,3),
plot(omega/pi,10*log10(abs(W101)));
grid; title('10*log_{10}(|W_{101}(e^{j\Omega})|)'); xlabel('\Omega/\pi (DT frequency normalized by
\pi)'); ylabel('10*log_{10}(|W_{101}(e^{j\Omega})|)dB');

%PART (b) END - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - %-------------------------------------------
----------------------------

%------------------------------------------------------------------------
%PART (c) START - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
%lowpass filter parameters
cutoff=(pi/4);  n_5=-
5:5; n_25=-25:25;
n_50=-50:50;
%creating the 3 lowpass filters
h_11=(sin((cutoff).*n_5))./(pi.*n_5); h_11(6)=0.25;
%value at zero
h_51=(sin((cutoff).*n_25))./(pi.*n_25);
h_51(26)=0.25;
h_101=(sin((cutoff).*n_50))./(pi.*n_50);  h_101(51)=0.25;
%(i)plotting the filters  figure();
%figure 8
subplot(3,1,1), stem(n_5,h_11,'filled');
```

```
grid; title('3(c{i}):h_{lpf_{11}}[n]'); xlabel('n-axis'); ylabel('h_{lpf_{11}}[n]');  subplot(3,1,2),
stem(n_25,h_51,'filled');
grid; title('3(c{i}):h_{lpf_{51}}[n]'); xlabel('n-axis'); ylabel('h_{lpf_{51}}[n]');  subplot(3,1,3),
stem(n_50,h_101,'filled');
grid; title('3(c{i}):h_{lpf_{101}}[n]'); xlabel('n-axis'); ylabel('h_{lpf_{101}}[n]');
%(ii)frequency response of filters
H11=fft(h_11,1024);
H51=fft(h_51,1024);
H101=fft(h_101,1024);  figure();
%figure 9
subplot(3,1,1), plot(omega/pi,(abs(H11)));
grid; title('|H_{W11}(e^{j\Omega})|'); xlabel('\Omega/\pi (DT frequency normalized by \pi)');
ylabel('|H_{W11}(e^{j\Omega})|');  subplot(3,1,2), plot(omega/pi,(abs(H51)));
grid; title('|H_{W11}(e^{j\Omega})|'); xlabel('\Omega/\pi (DT frequency normalized by \pi)');
ylabel('|H_{W51}(e^{j\Omega})|');  subplot(3,1,3), plot(omega/pi,(abs(H101)));
grid; title('|H_{W11}(e^{j\Omega})|'); xlabel('\Omega/\pi (DT frequency normalized by \pi)');
ylabel('|H_{W101}(e^{j\Omega})|');
%(iii)discussion

%talk about something here

%PART (c) END - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - %---------------------------------------------
----------------------------

%------------------------------------------------------------------------
%PART (d) START - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
%(i)making the blackman windows
blackman11=blackman(11);
blackman51=blackman(51);
blackman101=blackman(101);
%plotting blackman windows  figure();
%figure 10
subplot(3,1,1), stem(n_5,blackman11);
grid; title('FIG.10-11pt BMW'); xlabel('n-axis'); ylabel('BMW_{11}[n]');  subplot(3,1,2),
stem(n_25,blackman51);
grid; title('FIG.10-51pt BMW'); xlabel('n-axis'); ylabel('BMW_{51}[n]');  subplot(3,1,3),
stem(n_50,blackman101);
grid; title('FIG.10-101pt BMW'); xlabel('n-axis'); ylabel('BMW_{101}[n]');
%(ii)plot the frequency reponse of the 3 blackman windows
%FFTs of blackman windows
BM11=fft(blackman11,1024);
BM51=fft(blackman51,1024);
```

BM101=fft(blackman101,1024); %plotting
FFTs of blackman windows
figure(); %figure 11
subplot(3,1,1), plot(omega/pi,10*log10(abs(BM11)));
grid; title('10*log_{10}(|BMW_{11}(e^{j\Omega})|)'); xlabel('\Omega/\pi (DT frequency normalized by \pi)'); ylabel('10*log_{10}(|BMW_{11}(e^{j\Omega})|)'); subplot(3,1,2),
plot(omega/pi,10*log10(abs(BM51)));
grid; title('10*log_{10}(|BMW_{51}(e^{j\Omega})|)'); xlabel('\Omega/\pi (DT frequency normalized by \pi)'); ylabel('10*log_{10}(|BMW_{51}(e^{j\Omega})|)'); subplot(3,1,3),
plot(omega/pi,10*log10(abs(BM101)));
grid; title('10*log_{10}(|BMW_{101}(e^{j\Omega})|)'); xlabel('\Omega/\pi (DT frequency normalized by \pi)'); ylabel('10*log_{10}(|BMW_{101}(e^{j\Omega})|)'); %(iii)%designing the lowpass filters with blackman windows
hlpf_BMW11=transpose(blackman11).*h_11; hlpf_BMW51=transpose(blackman51).*h_51;
hlpf_BMW101=transpose(blackman101).*h_101;
%frequency response
HLPF_BMW11=fft(hlpf_BMW11,1024);
HLPF_BMW51=fft(hlpf_BMW51,1024);
HLPF_BMW101=fft(hlpf_BMW101,1024);
%plotting figure();
%figure 12
subplot(3,1,1),plot(omega/pi,20*log10(abs(HLPF_BMW11)));
grid; title('20*log_{10}(|H_{BMW11}(e^{j\Omega})|)'); xlabel('\Omega/\pi (DT frequency normalized by \pi)'); ylabel('20*log_{10}(|H_{BMW11}(e^{j\Omega})|)');
subplot(3,1,2),plot(omega/pi,20*log10(abs(HLPF_BMW51)));
grid; title('20*log_{10}(|H_{BMW51}(e^{j\Omega})|)'); xlabel('\Omega/\pi (DT frequency normalized by \pi)'); ylabel('20*log_{10}(|H_{BMW51}(e^{j\Omega})|)');
subplot(3,1,3),plot(omega/pi,20*log10(abs(HLPF_BMW101)));
grid; title('20*log_{10}(|H_{BMW101}(e^{j\Omega})|)'); xlabel('\Omega/\pi (DT frequency normalized by \pi)'); ylabel('20*log_{10}(|H_{BMW101}(e^{j\Omega})|)');
%(iv)

%come back to this part later when you understand the directions

%PART (d) END -

%PART (e) START -

%see script attached

%PART (e) END -

%PART (f) START - winfxn='blackman';

```
M=5; %from -5 to 5, an 11 point blackman window
[h,n]=firwinlpf((pi/4),M,winfxn);
figure(); %figure 13  stem(n,h,'filled');
grid; title('FIG.13-firwinlpf test:\Omega_{c}=\pi/4,M=5,blackman'); xlabel('n-axis');
ylabel('h_{BMW11}[n]');

%PART (f) END - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - %---------------------------------------------
----------------------------
%/////////////////////////////////////////////////////////////////////////

%/////////////////////////////////////////////////////////////////////////
%------------------------------------------------------------------------
%4 Design of a Bandpass and Highpass FIR Filters
%PART (a) START - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%see hand calculations

%PART (a) END - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%see hand calculations

%PART (b) START - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%see hand calculations

%PART (b) END - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%PART (c) START - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
%apply the firwinlpf function  winfxn='rectwin';
[h_bp,n_bp]=firwinlpf((pi/3),50,winfxn);
figure(); %figure 14 stem(n_bp,h_bp,'filled');
grid; title('4(c):FIG.14-w_{101}[n]*h_{ideal_{lpf}}[n]\Omega_c=\pi/3'); xlabel('n-axis');
ylabel('h_{w_{101}}[n]');  %modulate by a cosine function
h_bp_101=1*transpose(h_bp).*cos(pi*n_bp);  %H_BP_101=fft(h_bp_101,1024);
[mag,omega]=freqz(h_bp_101,1,1024,'whole');
figure(); %figure 15
%freqz(h_bp_101,1024,'whole');  plot(omega/pi,abs(mag));  grid; title('4(c):FIG.15-
|H_{HP_{W101}}(e^{j\Omega})|~\Omega_c=\pi/3,\Omega_0=\pi');
xlabel('\Omega/\pi (DT frequency normalized by \pi)'); ylabel('H_{HP_{W101}}(e^{j\Omega})');
%PART (c) END - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%PART (d) START - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - %(i)
```

```
%see hand calculations

%(ii)

%see hand calculations

%(iii)
%parameters for the firwinlpf fuction
winfxn='rectwin';  cutoff=pi/5;
M=50; %for a 101 point rectwin
[h_101_2,n_101_2]=firwinlpf(cutoff,M,winfxn);
figure(); %figure 16 stem(n_101_2,h_101_2,'filled');
grid; title('4d(iii):w_{101}[n]*h_{ideal_{lpf}}[n]:\Omega_c=\pi/5,M=101,rectwin \Omega_c=\pi/5');
xlabel('n-axis'); ylabel('h_{w_{101}}[n]');
%modulate by a cosine function
h_bp_101_2=2*transpose(h_101_2).*cos(((2*pi)/5)*n_101_2);
%H_BP_101_2=fft(h_bp_101_2,1024); [mag,phase]=freqz(h_bp_101_2,1,1024,'whole');
figure(); %figure 17  plot(phase/pi,abs(mag));
grid; title('4d(iii):|H_{BP_{W101}}(e^{j\Omega})|~\Omega_c=\pi/5,\Omega_0=2\pi/5');
xlabel('\Omega/\pi (DT frequency normalized by \pi)'); ylabel('H_{BP_{W101}}(e^{j\Omega})');
%PART (d) END - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -  %--------------------------------------------
-----------------------------

%-------------------------------------------------------------------------
%5 AUDIO FILTERING EXAMPLE
%PART (a) START - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%see analytical calculations

%PART (a) END - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%PART (b) START - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%see analytical calculations

%PART (b) START- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%see analytical calculations

%PART (b) END - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

%PART (c) START - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
%(i)PLAY THE NOISY SIGNAL XAUDIO WITH FS
```

```
load noisytrack.mat  %soundsc(xaudio,fs);
%(ii)CREATE THE FILTERS AND FILTER THE AUDIO
%lengths for the windows
%M_han=37;
M_han=44; %new length used to meet the requirements
M_ham=47;  M_black=68;
window_cutoff=0.192744*pi;
%use the firwinlpf function
[h_han,m_han]=firwinlpf(window_cutoff,M_han,'hanning');
[h_ham,m_ham]=firwinlpf(window_cutoff,M_ham,'hamming');
[h_black,m_black]=firwinlpf(window_cutoff,M_black,'blackman');
%freqz of of the impulse respnose
[han_h,han_freq]=freqz(h_han,1,8192,'whole',fs);
[ham_h,ham_freq]=freqz(h_ham,1,8192,'whole',fs);
[black_h,black_freq]=freqz(h_black,1,8192,'whole',fs);

%plotting the frequency response
figure();
subplot(3,1,1), plot(han_freq,abs(han_h));
grid; title('5(c):FIG.-|H_{HAN}(e^{j\Omega})|'); xlabel('Frequency(Hz)');
ylabel('|H_{HAN}(e^{j\Omega})|');  subplot(3,1,2),
plot(ham_freq,abs(ham_h));
grid; title('5(c):FIG.-|H_{HAM}(e^{j\Omega})|'); xlabel('Frequency(Hz)');
ylabel('|H_{HAM}(e^{j\Omega})|');  subplot(3,1,3),
plot(black_freq,abs(black_h));
grid; title('5(c):FIG.-|H_{BLACK}(e^{j\Omega})|'); xlabel('Frequency(Hz)');
ylabel('|H_{BLACK}(e^{j\Omega})|');

%plotting the passband of each filter  figure();
subplot(3,1,1),plot(han_freq,abs(han_h));
ylim([0.9900,1.010]);  xlim([0 5000]);
grid; title('5(c):FIG.-Passband of |H_{HAN}(e^{j\Omega})|'); xlabel('Frequency(Hz)');
ylabel('H_{HAN}(e^{j\Omega})');

subplot(3,1,2),plot(ham_freq,abs(ham_h));
ylim([0.9900,1.010]);  xlim([0 5000]);
grid; title('5(c):FIG.-Passband of |H_{HAM}(e^{j\Omega})|'); xlabel('Frequency(Hz)');
ylabel('H_{HAM}(e^{j\Omega})');

subplot(3,1,3),plot(black_freq,abs(black_h));
ylim([0.9900,1.010]);  xlim([0 5000]);
grid; title('5(c):FIG.-Passband of |H_{BLACK}(e^{j\Omega})|'); xlabel('Frequency(Hz)');
ylabel('H_{BLACK}(e^{j\Omega})');
```

```
%plotting the stopband
figure();
subplot(3,1,1), plot(han_freq,abs(han_h));
xlim([4900 7000]);  ylim([0 0.010]);
grid; title('5(c):FIG.-Stopband of |H_{HAN}(e^{j\Omega})|'); xlabel('Frequency(Hz)');
ylabel('H_{HAN}(e^{j\Omega})');

subplot(3,1,2), plot(ham_freq,abs(ham_h));
xlim([4900 7000]);  ylim([0 0.010]);
grid; title('5(c):FIG.-Stopband of |H_{HAM}(e^{j\Omega})|'); xlabel('Frequency(Hz)');
ylabel('H_{HAM}(e^{j\Omega})');

subplot(3,1,3), plot(black_freq,abs(black_h));
xlim([4900 7000]);  ylim([0 0.010]);
grid; title('5(c):FIG.-Stopband of |H_{BLACK}(e^{j\Omega})|'); xlabel('Frequency(Hz)');
ylabel('H_{BLACK}(e^{j\Omega})');

%plotting the phase
figure();
subplot(3,2,1), plot(han_freq,angle(han_h));
grid; title('5(c):FIG.-Phase |H_{HAN}(e^{j\Omega})|'); xlabel('Frequency(Hz)'); ylabel('Phase
H_{HAN}(e^{j\Omega})');
subplot(3,2,2), plot(han_freq,unwrap(angle(han_h)));
grid; title('5(c):FIG.-Unwrapped Phase H_{HAN}(e^{j\Omega})'); xlabel('Frequency(Hz)');
ylabel('|H_{HAN}(e^{j\Omega})|');

subplot(3,2,3), plot(ham_freq,angle(ham_h));
grid; title('5(c):FIG.-Phase |H_{HAM}(e^{j\Omega})|'); xlabel('Frequency(Hz)'); ylabel('Phase
H_{HAM}(e^{j\Omega})');
subplot(3,2,4), plot(ham_freq,unwrap(angle(ham_h)));
grid; title('5(c):FIG.-Unwrapped Phase H_{HAM}(e^{j\Omega})'); xlabel('Frequency(Hz)');
ylabel('|H_{HAM}(e^{j\Omega})|');

subplot(3,2,5), plot(black_freq,angle(black_h));
grid; title('5(c):FIG.-Phase |H_{BLACK}(e^{j\Omega})|'); xlabel('Frequency(Hz)'); ylabel('Phase
H_{BLACK}(e^{j\Omega})');
subplot(3,2,6), plot(black_freq,unwrap(angle(black_h)));
grid; title('5(c):FIG.-Unwrapped Phase H_{BLACK}(e^{j\Omega})'); xlabel('Frequency(Hz)');
ylabel('|H_{BLACK}(e^{j\Omega})|');

%apply the filters  y_han=filter(h_han,1,xaudio);
y_ham=filter(h_ham,1,xaudio);
```

```
y_blackman=filter(h_black,1,xaudio);

%transform the filtered signal
[X_han,F_han]=freqz(y_han,1,8196,'whole',fs);
[X_ham,F_ham]=freqz(y_ham,1,8196,'whole',fs);
[X_bmw,F_bmw]=freqz(y_blackman,1,8196,'whole',fs);

%plot transforms
figure();
subplot(3,1,1),plot(F_han,20*log10(abs(X_han)));
grid; title('5(c)_{iii}20*log_{10}(|H_{XAUDIO_{HAN}}(e^{j\Omega})|)');
xlabel('Frequency(Hz)'); ylabel('H(e^{j\Omega}) dB');
subplot(3,1,2),plot(F_ham,20*log10(abs(X_ham)));  grid;
title('5(c)_{iii}20*log_{10}(|H(e^{j\Omega})|)');
xlabel('Frequency(Hz)'); ylabel('H(e^{j\Omega}) dB');
subplot(3,1,3),plot(F_bmw,20*log10(abs(X_bmw)));
grid; title('5(c)_{iii}20*log_{10}(|H_{XAUDIO_{BMW}}(e^{j\Omega})|)');  xlabel('Frequency(Hz)');
ylabel('H(e^{j\Omega}) dB');

%(iii)PLOTS
%PART (c) END - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

References

MATLAB CODE - Darima Batueva-Islam

```
%% 5 Audio Filtering Example
close all clear all fs=44100;
N=8192; pb=[1000 4000 0.98
1.02]; sb=[5000 8000 0 0.02];
% specifying 3 windows
[han, n1]=firwinlpf(0.193*pi,44,'hanning'); %min=44
[H_han,F1]=freqz(han,1,N,'whole',fs);
[ham, n2]=firwinlpf(0.193*pi,45, 'hamming');%min 45
[H_ham,F2]=freqz(ham,1,N,'whole',fs);
[bmw, n3]=firwinlpf(0.193*pi,60, 'blackman'); %min=60
[H_bmw,F3]=freqz(bmw,1,N,'whole',fs);

figure % frequency response of 3 filters subplot(231);
plot(F1,10*log10(abs(H_han))); xlabel('Frequency,
Hz'); xlim([0,fs])
ylabel('10*log_1_0(|H(j\omega)|),dB');
title('Frequency response of Hanning LPF')
subplot(232); plot(F2,10*log10(abs(H_ham)));
xlabel('Frequency, Hz'); xlim([0,fs])
ylabel('10*log_1_0(|H(j\omega)|),dB');
title('Frequency response of Hamming LPF')
subplot(233); plot(F3,10*log10(abs(H_bmw)));
xlabel('Frequency, Hz'); xlim([0,fs])
ylabel('10*log_1_0(|H(j\omega)|),dB');
title('Frequency response of Blackman LPF')
subplot(234); plot(F1,angle(H_han));
xlabel('Frequency, Hz'); ylabel('Phase'); xlim([0,fs])
subplot(235); plot(F2,angle(H_ham));
xlabel('Frequency, Hz'); ylabel('Phase'); xlim([0,fs])
subplot(236); plot(F3,angle(H_bmw));
xlabel('Frequency, Hz'); ylabel('Phase'); xlim([0,fs])

figure % plotting zoomed-in passband and stopband
subplot(321); plot(F1,abs(H_han)); xlabel('Frequency,
Hz'); ylabel('|H(j\omega)|') title('Zoomed-in passband
of Hanning window LPF')
axis(pb); grid on subplot(322);
plot(F1,abs(H_han));
xlabel('Frequency,
Hz');ylabel('|H(j\omega)|')
```

```
title('Zoomed-in stopband of
Hanning window LPF')
axis(sb); grid on subplot(323); plot(F2,abs(H_ham));
xlabel('Frequency, Hz'); ylabel('|H(j\omega)|')
title('Zoomed-in passband of Hamming window LPF')
axis(pb); grid on subplot(324);
plot(F2,abs(H_ham)); xlabel('Frequency, Hz');
ylabel('|H(j\omega)|') title('Zoomed-in stopband of
Hamming window LPF')
axis(sb); grid on subplot(325); plot(F3,abs(H_bmw));
xlabel('Frequency, Hz'); ylabel('|H(j\omega)|')
title('Zoomed-in passband of Blackman window LPF')
axis(pb); grid on subplot(326);
plot(F3,abs(H_bmw)); xlabel('Frequency,
Hz');ylabel('|H(j\omega)|') title('Zoomed-in stopband
of Blackman window LPF')
axis(sb); grid on

%% Application: Filtering the noisy audio signal
load noisytrack.mat %soundsc(xaudio,fs)

x_han=filter(han,1,xaudio); x_ham=filter(ham,
1,xaudio); x_bmw=filter(bmw,1,xaudio);
%soundsc(x_han,fs)
%soundsc(x_bmw,fs)
%soundsc(x_bmw,fs)

t=1:length(xaudio); [H,F]=freqz(xaudio,1,N,'whole',fs);
figure subplot(211); plot(t/fs,xaudio) title('Time analysis
of Noisy Track') xlabel('Time, seconds');
ylabel('x_a_u_d_i_o'); axis tight subplot(212); plot(F,
abs(H))
xlabel('Frequency, Hz'); ylabel('|X_a_u_d_i_o(j\omega)|');
title('Frequency analysis of Noisy Track')

figure % plotting filtered signals in time subplot(211);
plot(t/fs,xaudio) title('Noisy Track')
xlabel('Time, seconds'); ylabel('Amplitude'); axis tight
subplot(212); plot(t/fs,x_han) title('Noisy Track
filtered with Hanning window LPF') xlabel('Time,
seconds'); ylabel('Amplitude'); axis tight figure
subplot(211); plot(t/fs,x_ham) title('Noisy Track
filtered with Hamming window LPF') xlabel('Time,
seconds'); ylabel('Amplitude'); axis tight subplot(212);
```

```
plot(t/fs,x_bmw) title('Noisy Track filtered with
Blackman window LPF')
xlabel('Time, seconds'); ylabel('Amplitude'); axis tight

% Frequency spectrum of filtered signal
[X_han,F1]=freqz(x_han,1,N,'whole',fs);
[X_ham,F2]=freqz(x_ham,1,N,'whole',fs);
[X_bmw,F3]=freqz(x_bmw,1,N,'whole',fs);

figure  % plotting frequency response of filtered signal
subplot(311); plot(F1,abs(X_han)); xlabel('Frequency, Hz');
ylabel('|X(e^{j\Omega})|'); xlim([0,fs]) title('Frequency
spectrum of signal filtered with Hanning LPF') subplot(312);
plot(F2, abs(X_ham)); xlabel('Frequency, Hz');
ylabel('|X(e^{j\Omega})|'); xlim([0,fs]) title('Frequency
spectrum of signal filtered with Hamming LPF') subplot(313);
plot(F3,abs(X_bmw)); xlabel('Frequency, Hz');
ylabel('|X(e^{j\Omega})|'); xlim([0,fs])
title('Frequency spectrum of signal filtered with Blackman LPF')

figure
subplot(311); plot(F1,10*log10(abs(X_han))); xlabel('Frequency,
Hz'); ylabel('10*log_1_0|X(j\omega)|'); xlim([0,fs]) title('Frequency
spectrum of signal filtered with Hanning LPF') subplot(312); plot(F2,
10*log10(abs(X_ham))); xlabel('Frequency, Hz');
ylabel('10*log_1_0|X(j\omega)|'); xlim([0,fs]) title('Frequency
spectrum of signal filtered with Hamming LPF') subplot(313);
plot(F3,10*log10(abs(X_bmw))); xlabel('Frequency, Hz');
ylabel('10*log_1_0|X(j\omega)|'); xlim([0,fs]) title('Frequency
spectrum of signal filtered with Blackman LPF')
```