LAB 5

Discrete Time Impulse Response and Convolution

Jordan Hayes

EGR323: Signals and Systems Analysis

6/15/2021

Prof. Krug

**Objective**

The objectives of this laboratory are to further practice signal processing by generating filters to process signals by convoluting them using functions in MATLAB and output their plots, and to filter audio files using generated filters as well to output their ‘cleaner’ sound. DT signals are always related to a sampled version of a CT analogy, done so through knowledge of the sampling frequency:

f

Nyquist theory states that the highest frequency component that can be represented in the sampled DT signal is

To determine its CT analogous frequency, the period of this signal is for which we conclude that its frequency is .

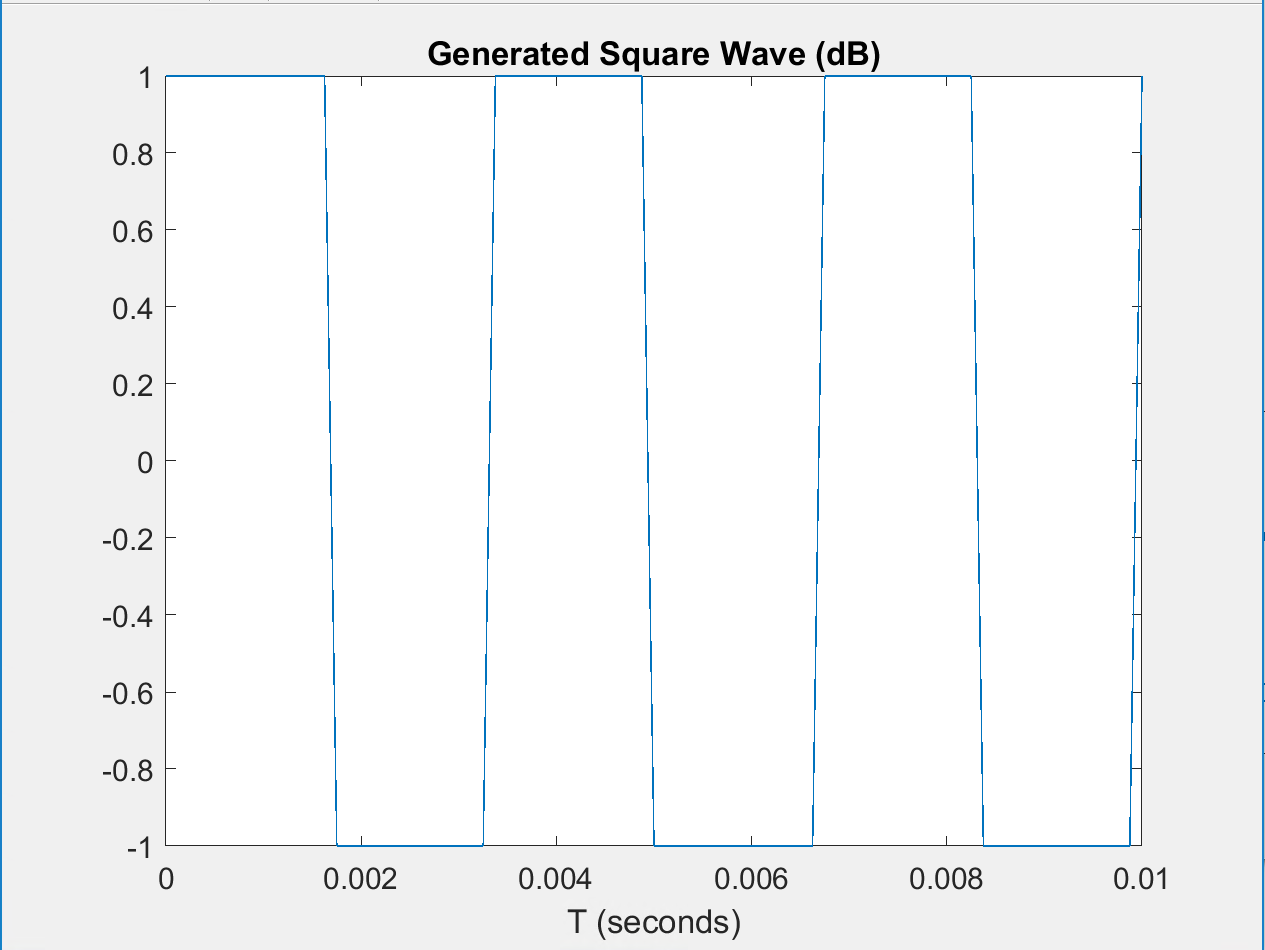
**Procedure and Results**

**Part I:**

This part involves generating a square wave in MATLAB of 300Hz; to generate a DT sinusoid at a particular frequency given a sampling frequency , Eq. (1) is used below.

(1)

3 cycles of this wave are generated in MATLAB and shown below.



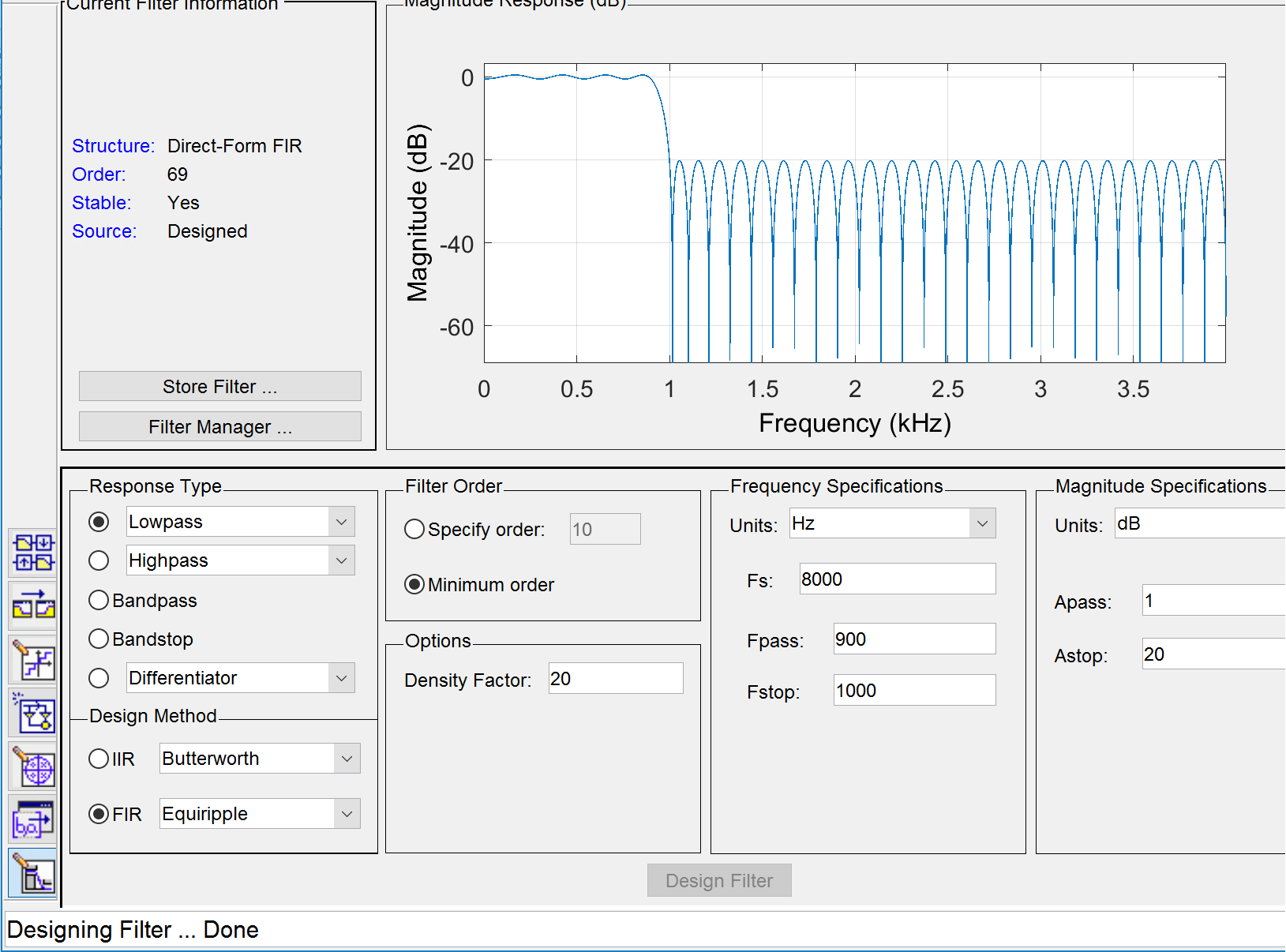
**Figure 1: MATLAB Generated 300Hz Square Wave**

FilterDesigner is opened in MATLAB, and an FIR Lowpass Filter (LPF) was generated to process the square wave. The design choices for the filter were given in Table 1 below.

**Table 1: Design Choices for LPF**

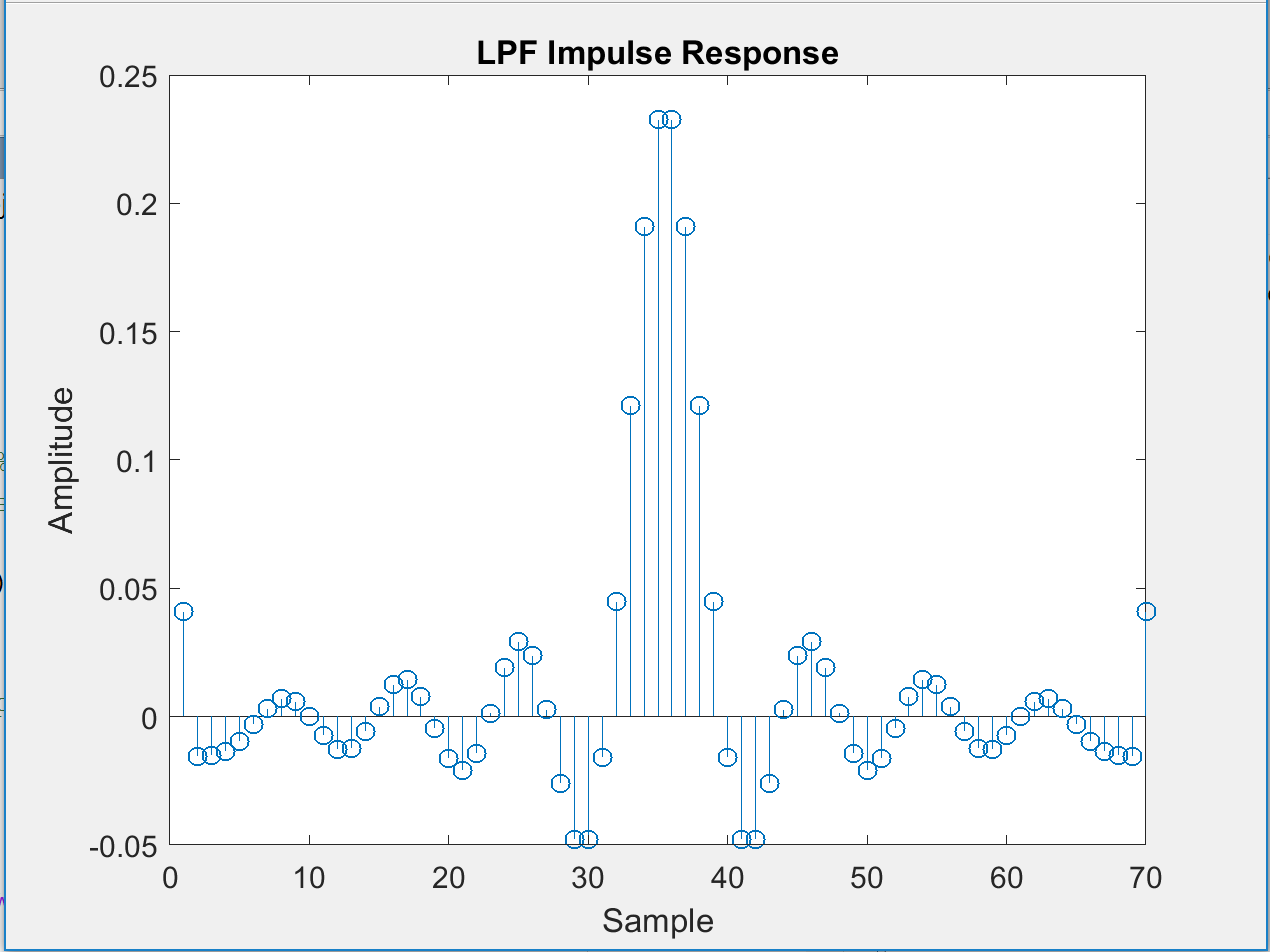
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Design Method | Filter Order | Passband Edge (Hz) | Stopband Edge (Hz) | Sampling Frequency (Hz) | Stopband Attenuation (dB) |
| FIR | Minimum | 900 | 1000 | 8000 | 20 |

The output window of filterDesigner magnitude response is shown below.



**Figure 2: filterDesigner Window with Magnitude Response of Generated LPF**

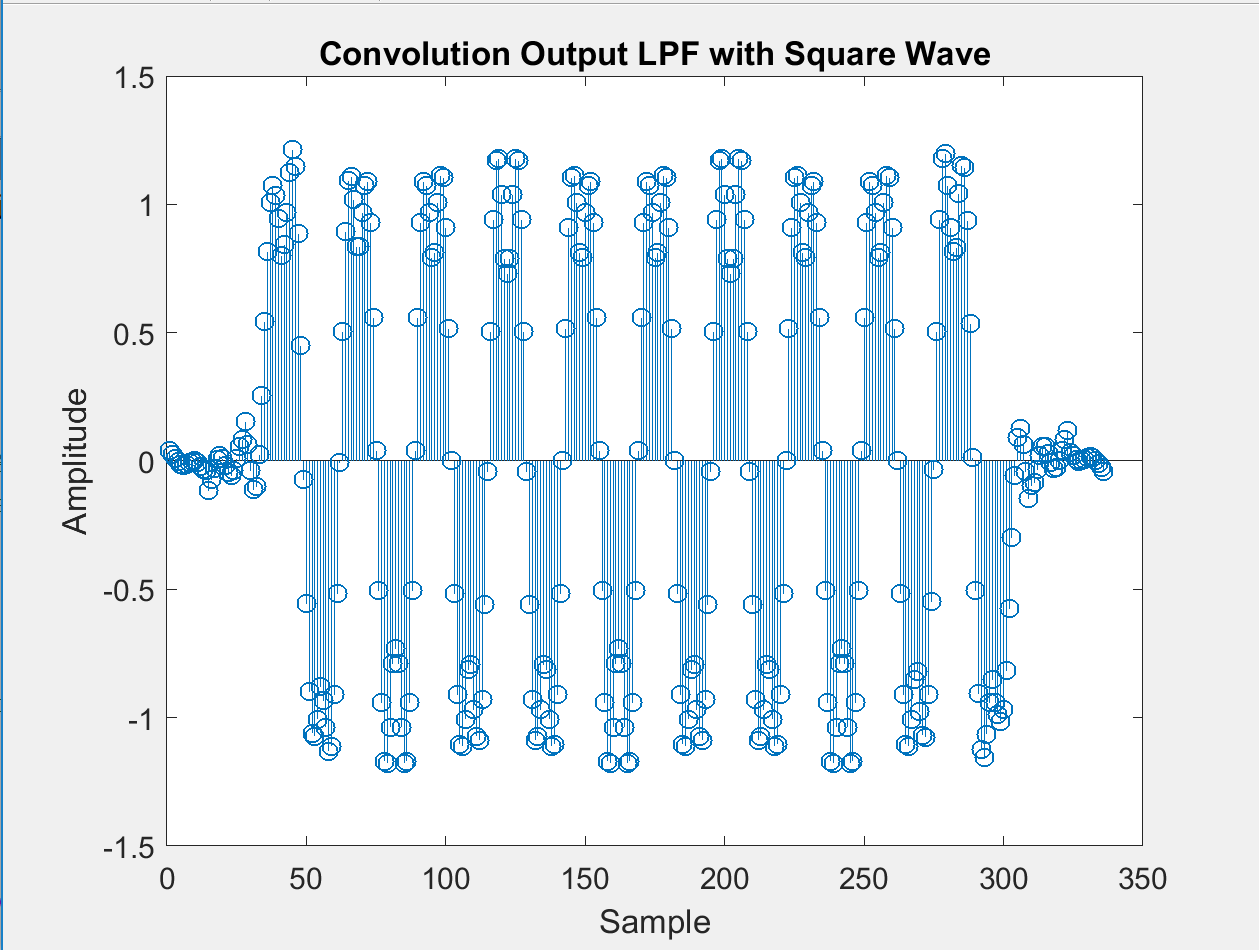
The magnitude plot in the figure above shows that frequencies over 1000 Hz of any signal convolved with the Lowpass Filter will be attenuated by 20 dB, and only frequencies lower will pass through and be captured by the convolution plot. The impulse response of this filter is plotted in MATLAB using the *stem()* command.



**Figure 3: Impulse Response of LPF**

The impulse response in MATLAB is graphed in the plot above such that the sample number is on the x-axis, and amplitude is on the y-axis. The plot looks like a typical Lowpass impulse response, which takes the form of *sinc().*

The *conv()* function is used to process the input square wave with the LPF. The resulting output signal is plotted using *stem()* shown in Figure 4.



**Figure 4: Convolution of LPF and Square Wave**

The plot above shows the output of the convolved square wave with the LPF, giving the plot a rounded look on the humps of the graph.

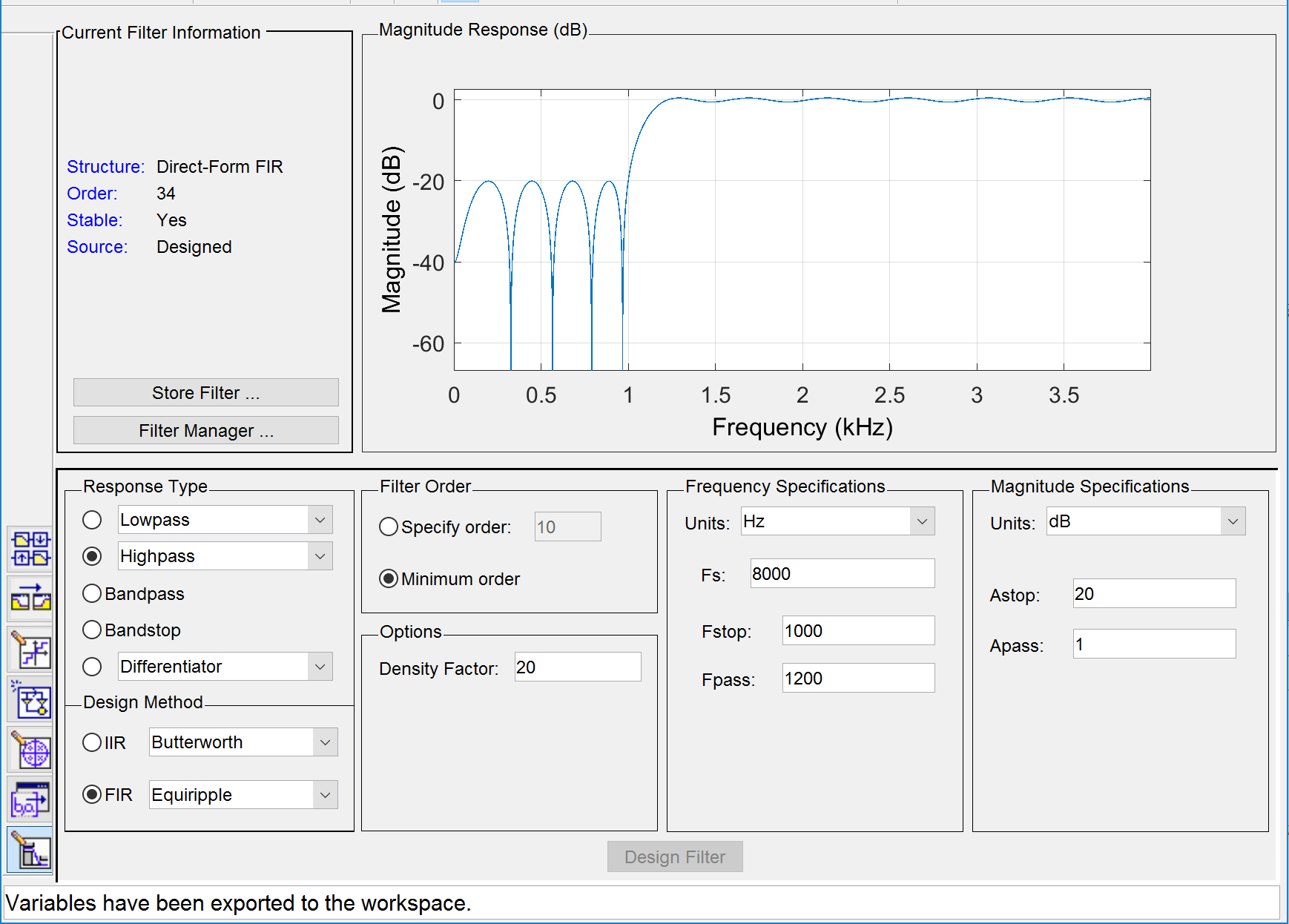
3) Does it look the way you would expect? Explain.

*Yes, Figure 4 does look as expected because the Lowpass Filter will filter out high frequency signals, which are located at the edges of the square wave where it becomes vertical and the corners. The points shown in Figure 4 are low frequency points, which represent the more or less horizontal parts of the input signal. When a sampled signal is passed through a low pass filter, reconstructed output signal is the sum of copies of impulse response shifted by integral multiples of , the sampling vector, and multiplied by the value of x(t) at the corresponding integral multiple of . This particular convolution have adjacent points that provide an averaging affect that smooths out sharp corners or accelerations.*

The above process was repeated for a Highpass FIlter (HPF).

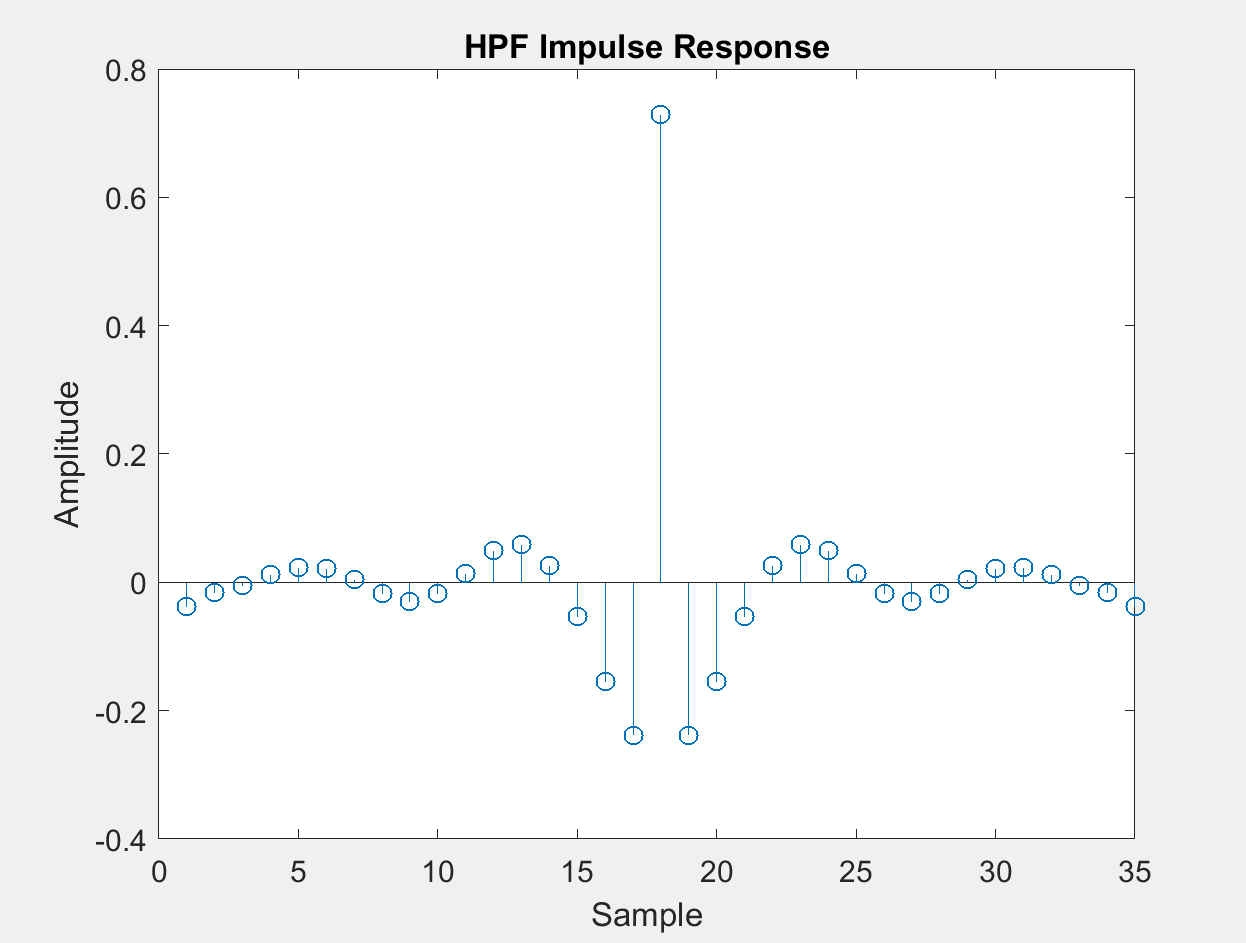
**Table 2: Design Choices for HPF**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Design Method | Filter Order | Passband Edge (Hz) | Stopband Edge (Hz) | Sampling Frequency (Hz) | Stopband Attenuation (dB) |
| FIR | Minimum | 1200 | 1000 | 8000 | 20 |



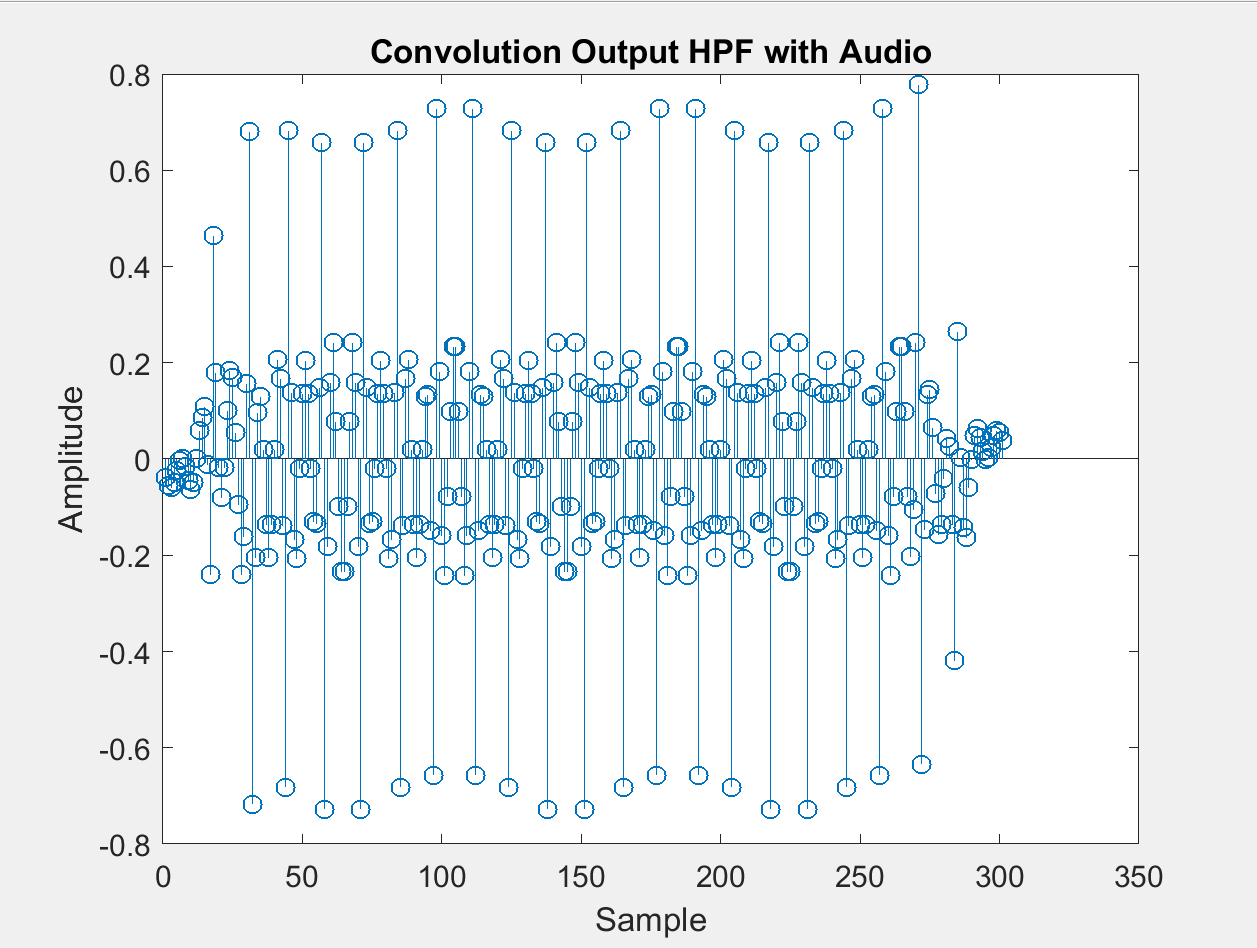
**Figure 5: filterDesigner Window with Magnitude Response of Generated HPF**

The magnitude plot in the figure above shows that frequencies over 1000 Hz of any signal convolved with the Highpass Filter will pass through and exist in the convolution. Frequencies less than 1000 Hz will be attenuated by 20 dB.



**Figure 6: Impulse Response of HPF**

The plot above looks like a typical Highpass kernel, or filter impulse response, which are formed from subtracting the corresponding Lowpass filter kernel from a delta function. The result of the convolution is shown in Figure 7 below.



**Figure 7: Convolution of HPF and Square Wave**

3) Does it look the way you would expect? Explain.

*Yes, Figure 7 does look as expected because the Highpass Filter will filter out the low frequency points, which are located (on this graph above) between tallest, vertical points. The horizontal portions of the square wave are filtered out. A high density of points not filtered out by the Highpass Filter are condensed in areas where the corners and vertical portions of the square input wave exist.*

5) Compare the LPF and HPF impulse response plots. What similarities and differences

do you see? Can you postulate why these filters might have these shapes in order to

achieve the desired filtering?

*The plots (Figure 3 and 6) are similar in a way such that they both allow a certain range of frequencies to pass through convolution of the input signal and filter characteristics. They both capture some portion of the signal, but the portions they capture are nearly compliments; if you added the output of the convolutions, the plot would look a lot like a reconstruction of the original input, with all of the frequencies now captured above and below 1000 Hz.*

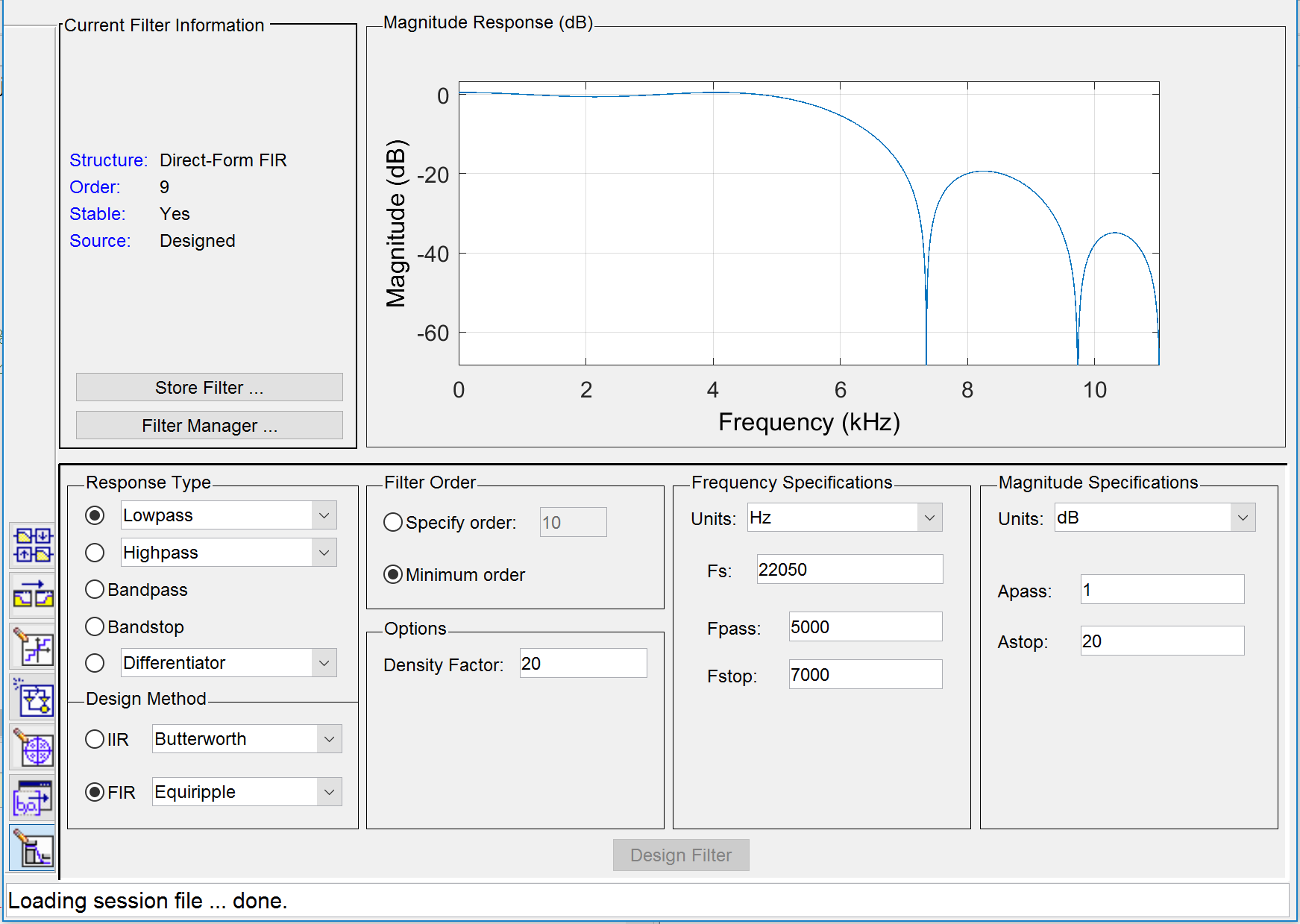
*Both impulses seem to haver sinusoidal tendencies and very similar shape; however, the axis are different. The impulse function of a low pass filter turns out to be a sinc() function, and the Highpass filter is simply the value of the Lowpass subtracted from a delta function, which is why adding up the signals should reconstruct the original.*

**Part II:**

For this part, noise corrupted speech files are downloaded and imported using the *audioread()* command. The file was sampled at 22050Hz. The unwanted noise is present at 8kHz and above; therefore, a LPF was designed was reduce the noise and improve the speech quality. The design choices are listed in Table 1 below, using a stopband value lower than 8kHz.

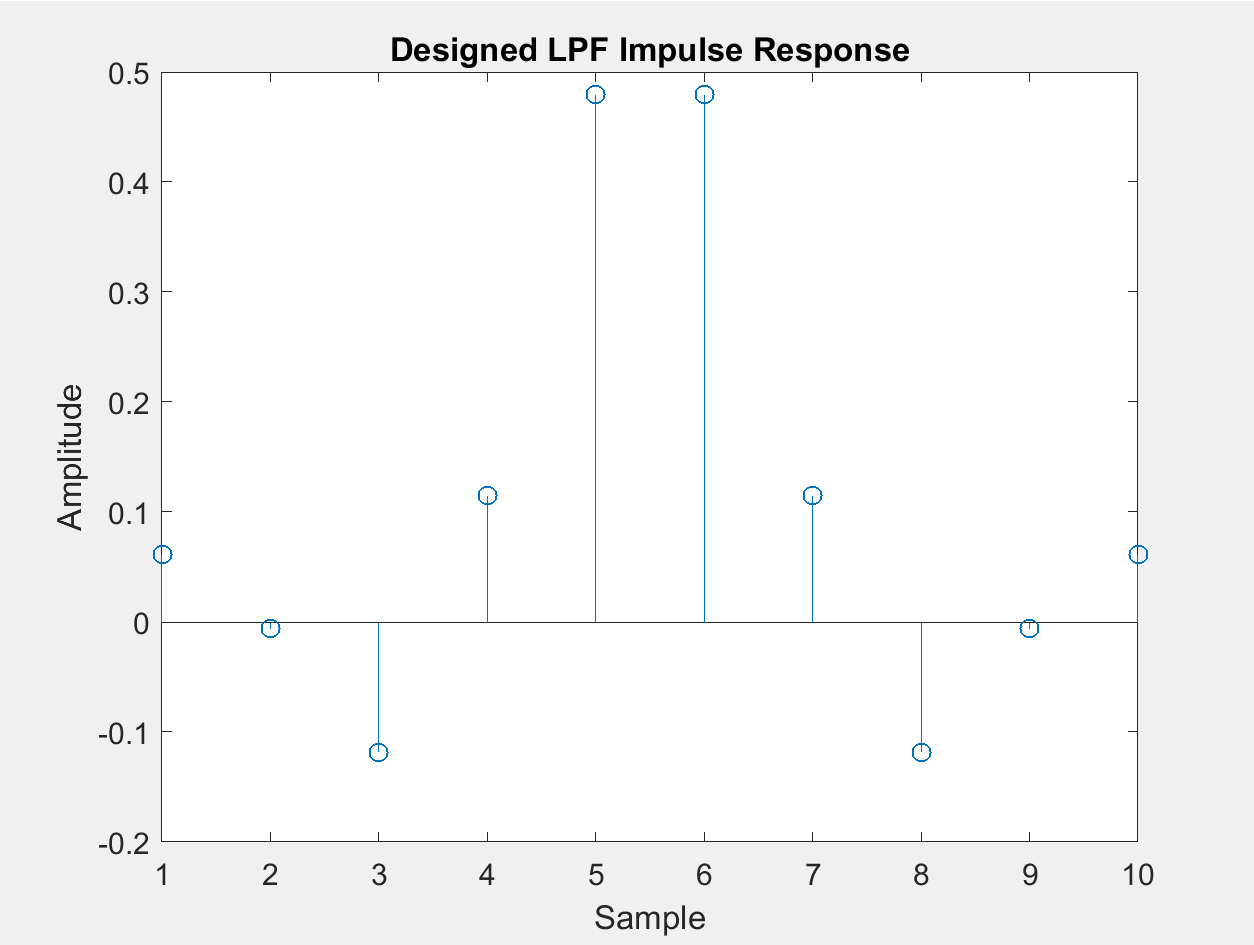
**Table 3: Design Choices for Designed LPF**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Design Method | Filter Order | Passband Edge (Hz) | Stopband Edge (Hz) | Sampling Frequency (Hz) | Stopband Attenuation (dB) |
| FIR | Minimum | 5000 | 7000 | 22050 | 20 |



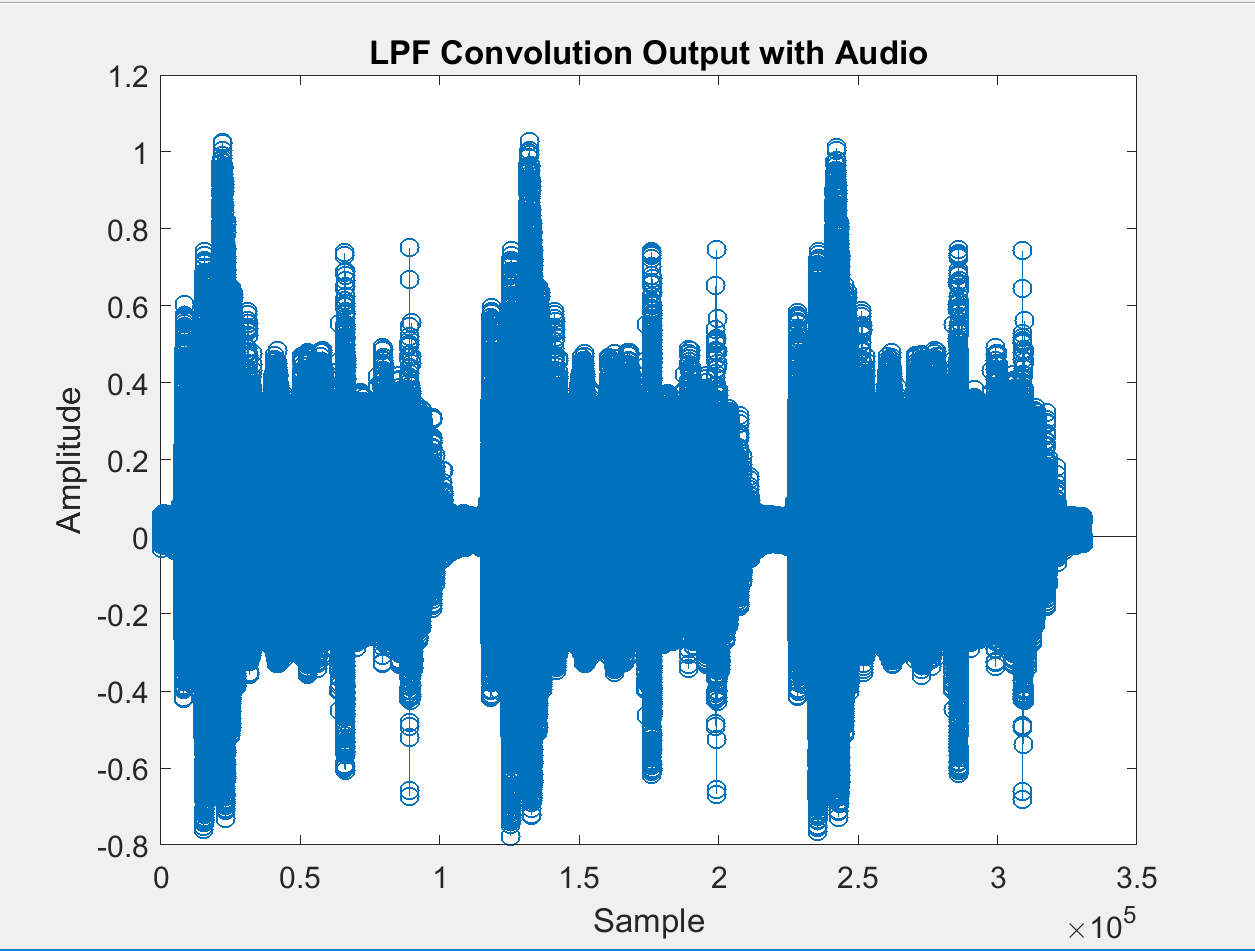
**Figure 8: filterDesigner Window with Magnitude Response of Designed LPF**

The magnitude plot in the figure above shows that frequencies over 7000 Hz of any signal convolved with the Lowpass Filter will be attenuated by 20 dB.



**Figure 9: Impulse Response of Designed LPF**

Figure 9 shows the impulse response of the designed Lowpass Filter, showing the amplitude vs. the sample number. The *sinc()* shape still forms which is used to separate one band of frequencies from another.



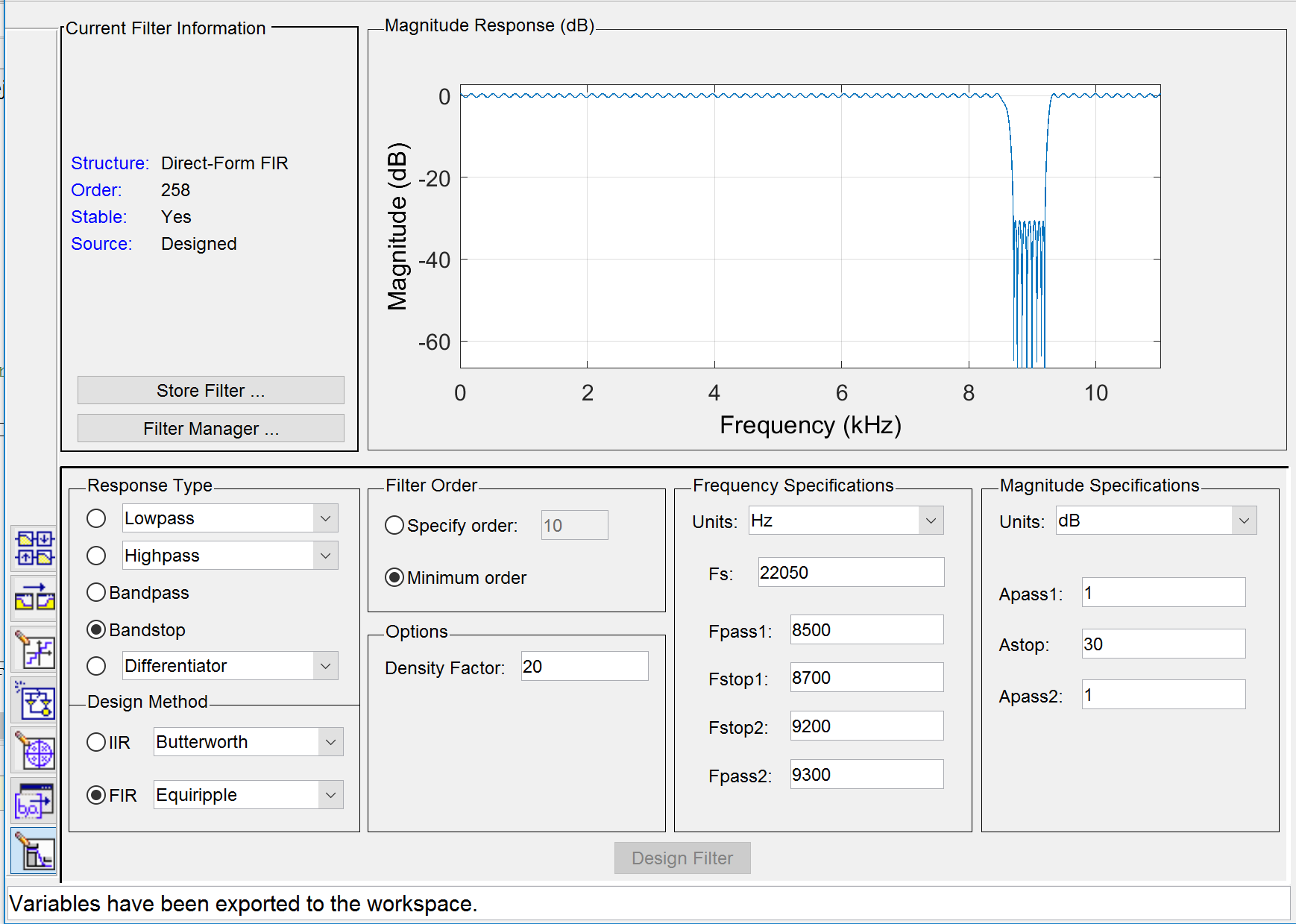
**Figure 10: Convolution of Designed LPF and Audio**

Although the output plot of the convolution may not seem too useful, listening to the filtered audio file compared with the original proves there is a clear difference in tones and frequencies.

Another wave file was imported, renamed *Jordans\_test.wav,* which has been corrupted by an interfering tone. A FIR Bandstop Filter (BSF) is designed to attempt to remove the tone. The design choices and filter coefficients are shown below.

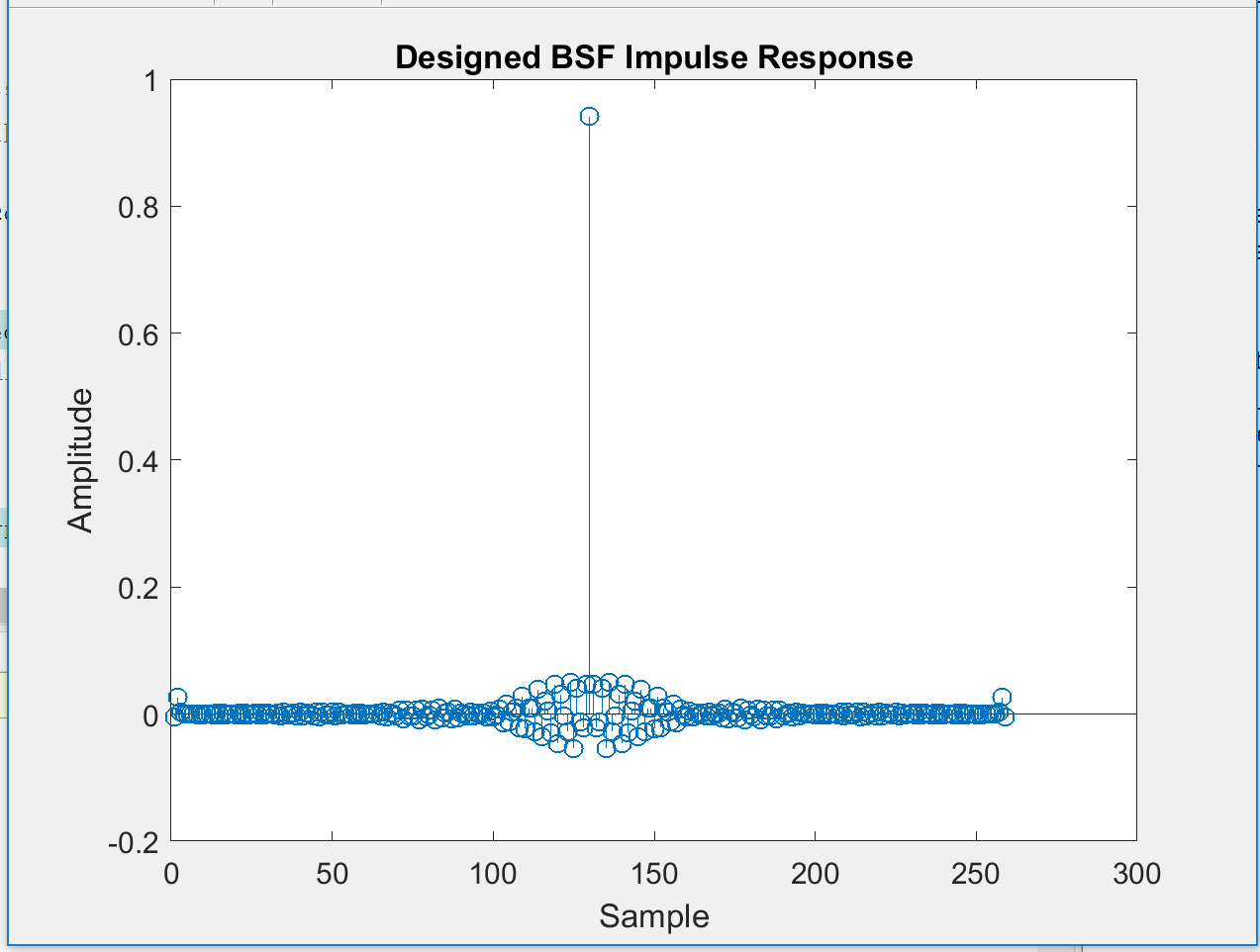
**Table 4: Design Choices for Designed BSF**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Design Method | Filter Order | Passband Edge 1 (Hz) | Stopband Edge 1 (Hz) | Stopband Edge 2 (Hz) | Passband Edge 2 (Hz) | Sampling Frequency (Hz) | Stopband Attenuation (dB |
| DIR | Minimum | 8500 | 8700 | 9200 | 9300 | 22050 | 30 |



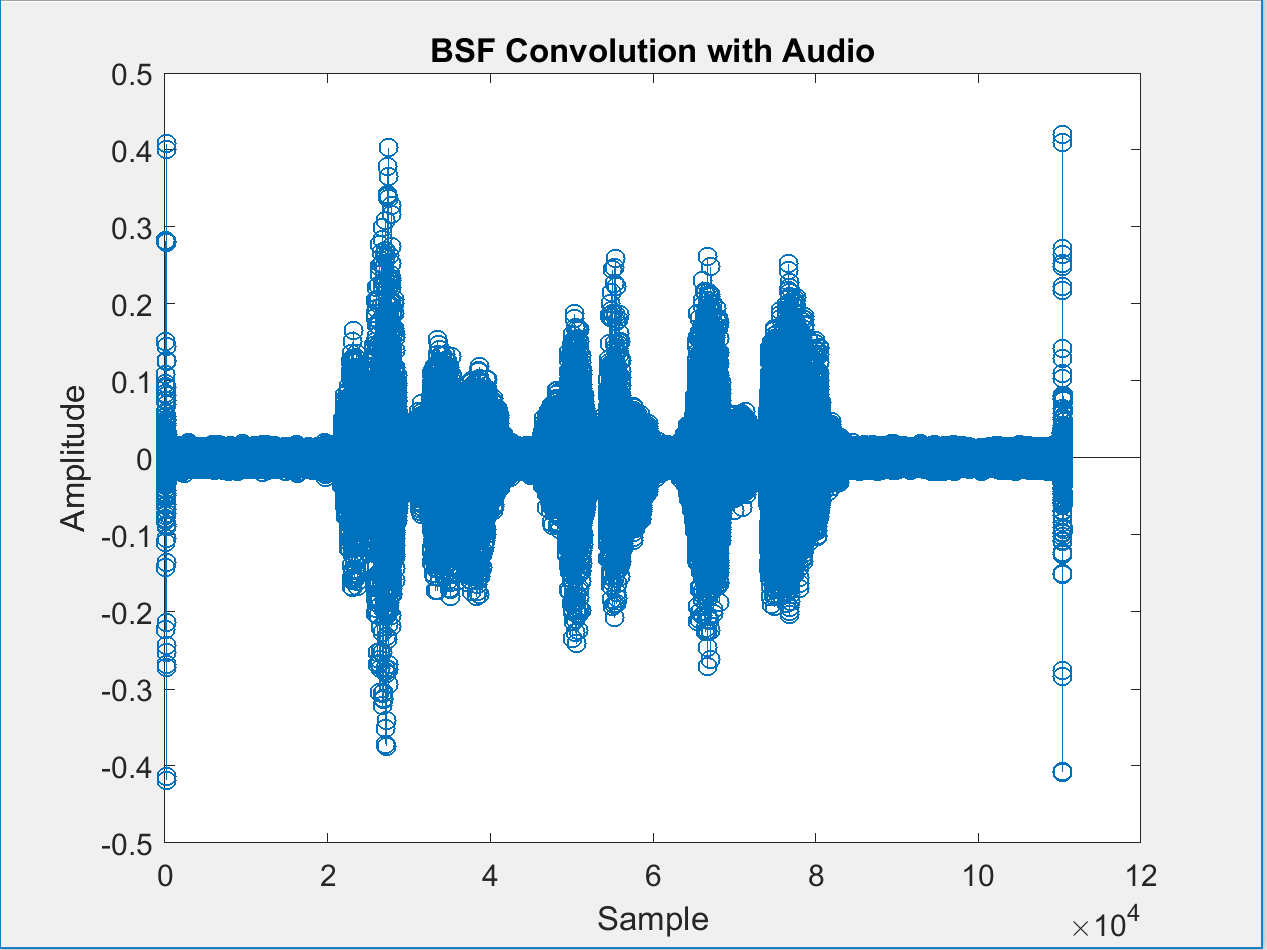
**Figure 11: filterDesigner Window with Magnitude Response of Designed BSF**

The magnitude plot in the figure above shows that the frequencies of an input signal between 8700 and 9200 Hz convolved with the designed Bandstop filter will be attenuated by 30 dB.



**Figure 12: Impulse Response of Designed BSF**

Figure 12 shows the impulse response of the designed Bandstop Filter, showing the amplitude vs. the sample number.



**Figure 13: Convolution of Designed BSF and Audio**

Looking at the plot and listening to the audio file, it is found that much of the noise and ringing is found inside the stopband range and is therefore attenuated. The voice heard is much louder and clearer, and the ringing is reduced to a much more tolerable level.

**Conclusion**

The objectives of this laboratory are completed by practicing signal processing by generating filters to process signals and convoluting them using functions in MATLAB. Their plots and design characteristics were recorded and/or plotted. The audio files are filtered using designed filters as well to output the ‘cleaner’ signal. An important observation in this procedure is seen by convoluting the generated square wave and audio files. Areas of the square wave with high frequencies, the corners and vertical legs, were filtered out by a lowpass, but captured by a highpass, and vice versa for low frequency areas or the graph, near DC where the line of the wave is horizontal. The ringing in the audio files was filtered by iterative analysis, repeatedly designing filters with new characteristics to identify the range of frequencies the noise existed in, and using the impulse function of the filters to convolve and filter out the unwanted noise.