

ECE 351 - SECTION 52

FINAL PROJECT

Filter Design

Submitted By :
Ben Bunce

Contents

1	Introduction	2
2	Equations	2
	2.1 Transfer Function	2
	2.2 Transfer function (RLC)	2
3	Methodology	2
4	Circuit	3
5	Results	4
	5.1 Input	4
	5.2 Output	4
	5.3 Filter Bode Plots	5
	5.4 FFT Plots (Filtered and Unfiltered)	6
6	Questions	8
7	Conclusion	8
8	Appendix	9
	8.1 .stem() Workaround	9

1 Introduction

Using tools Python tools learned throughout the semester, a filter is created that takes in a signal from an aircraft positioning system, and outputs a clean signal after filtering. The signal from the aircraft positioning system contains great amounts of noise that causes problems with understanding the actual data, and that noise needs filtered out.

2 Equations

2.1 Transfer Function

$$H(s) = \frac{1256.6s}{s^2 + 1256.6s + 1579043.6}$$

2.2 Transfer function (RLC)

$$H(s) = \frac{s \frac{R}{L}}{s^2 + s \frac{R}{L} + \frac{1}{LC}}$$

$$R = 1000$$

$$L = 795.8mH$$

$$C = 795.8nF$$

$$B = \frac{R}{L} = 200Hz = 400\pi rad/s$$

$$w_o^2 = \frac{1}{LC}$$

3 Methodology

First, the signal is read into an array using the 'read_csv' function. Once this function has been moved into an array, the FFT function is used to create the plots of the Fourier transform. This is done for the full range of frequencies, as well as plots of 0-1780Hz, 2010-100000Hz, and 1800-2000Hz.

Next, a bandpass filter is developed that meets the following requirements:

- Allows 1800-2000Hz
- The position measurement information is attenuated by less than -0.3dB

- The low-frequency vibration noise must be attenuated by at least -30dB
- The switching amplifier noise must be attenuated by at least -21dB
- All noise that exists at frequencies greater than 100kHz must be completely attenuated

Once the filter is created, Bode plots are generated for a variety of frequency ranges. These are the full range, 0-1780Hz, 2010-100000Hz, and 1800-2000Hz. The signal is then passed through the filter using the 'bilinear' and 'lfilter' function. This output is then plotted.

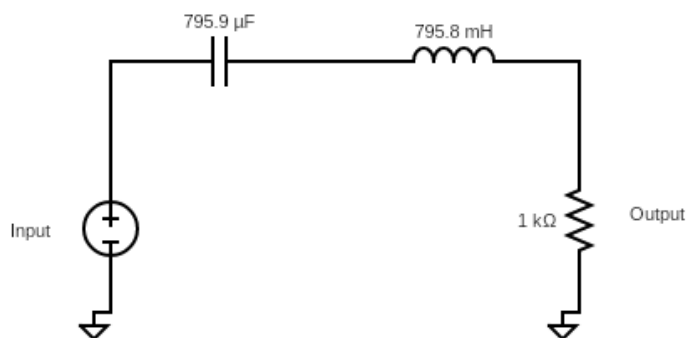
Finally, the FFT function is used to create the plots of the filtered Fourier transform. This is done for the full range of frequencies, as well as plots of 0-1780Hz, 2010-100000Hz, and 1800-2000Hz.

4 Circuit

$$R = 1000$$

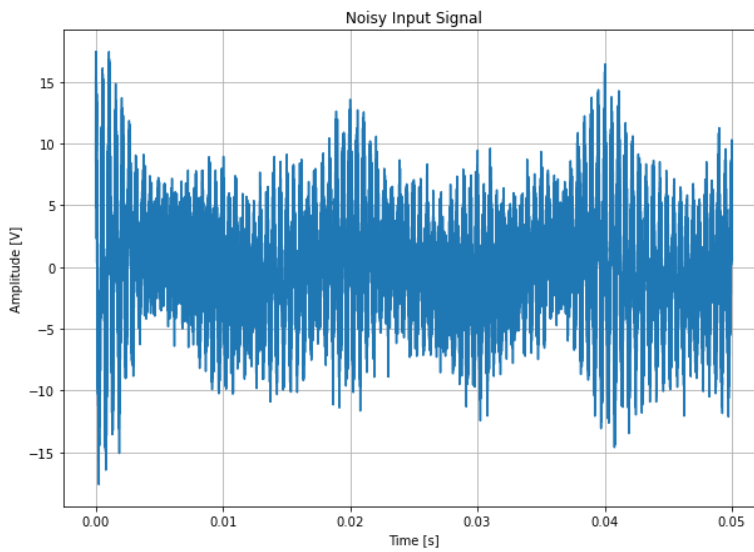
$$L = 795.8mH$$

$$C = 795.8nF$$



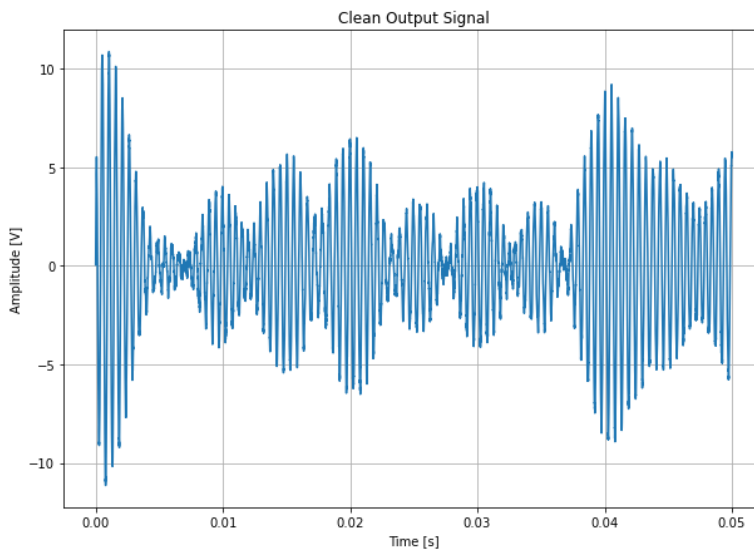
5 Results

5.1 Input



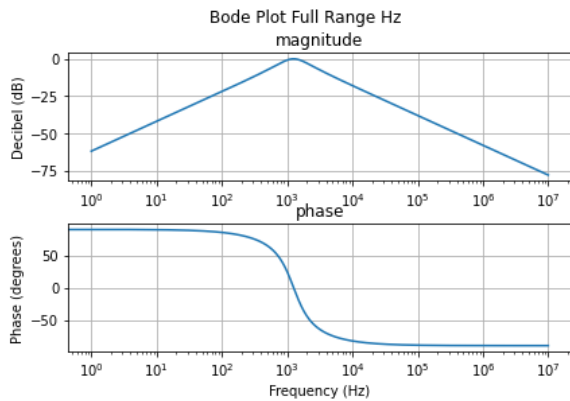
Unchanged signal received from aircraft positioning sensor.

5.2 Output

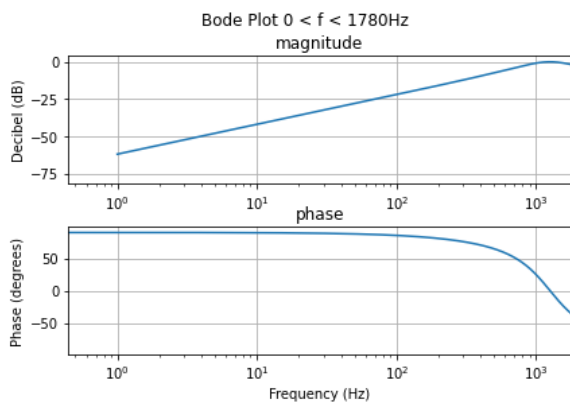


The output signal provided by the filter shows a very clean result compared to the noisy input signal. This signal has clean lines without the excess clutter that is seen in the noisy input signal, and is centered around a straight zero line, compared to the noisy input, which follows a sine curve.

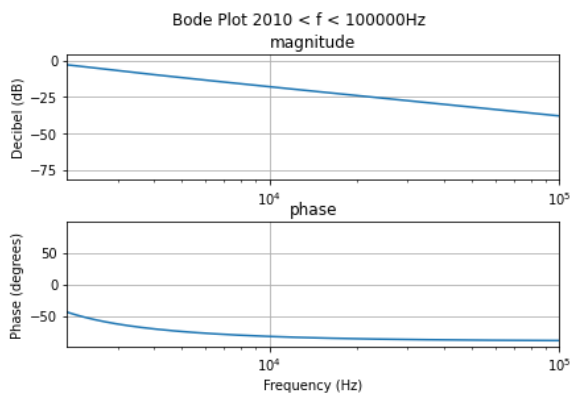
5.3 Filter Bode Plots



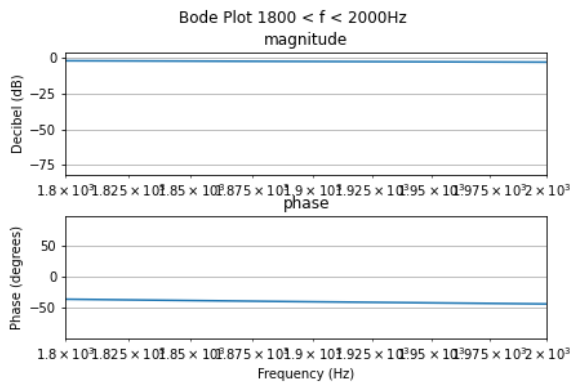
The above Bode plot shows the full range of frequencies and the effects of the filter on the frequencies.



As seen in the Bode plot above, all low frequency values (less than 100Hz) are attenuated by at least -30dB. This effectively cancels out the low frequency vibrations from the buildings ventilation system.

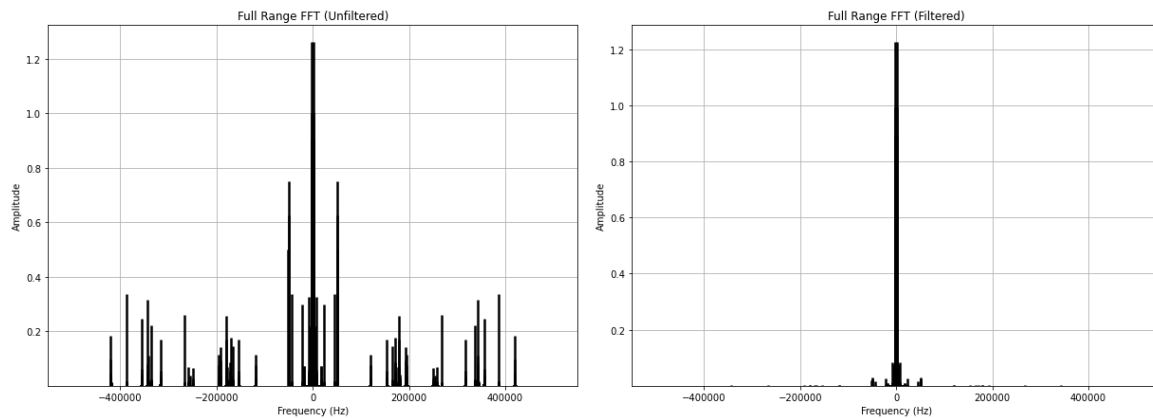


As seen in the Bode plot above, the high frequencies are attenuated by at least -25dB (frequencies greater than 10KHz). This effectively cancels out the high frequency noise from the shared ground.

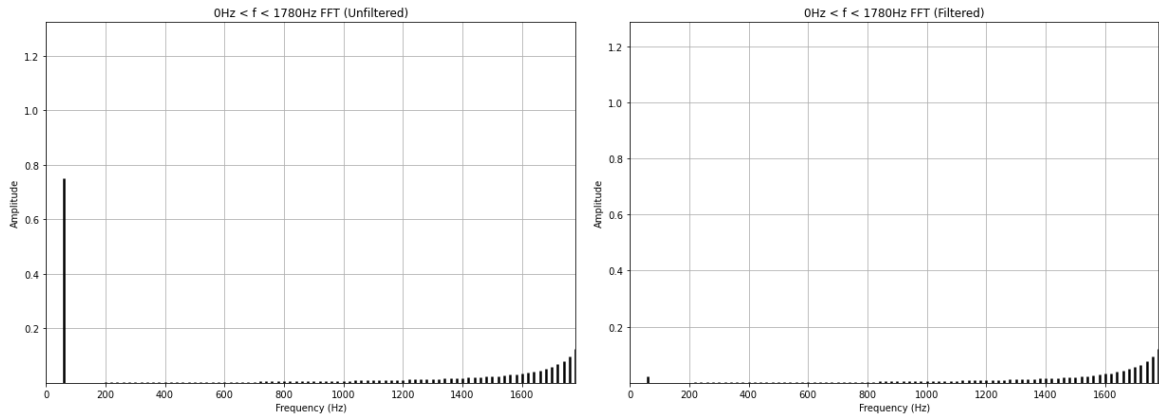


As seen in the Bode plot above, the frequencies between 1800Hz and 2000Hz are negligibly attenuated. This attenuation is less than the -0.3dB minimum of the filter, which effectively allows all frequencies between 1800Hz and 2000Hz through.

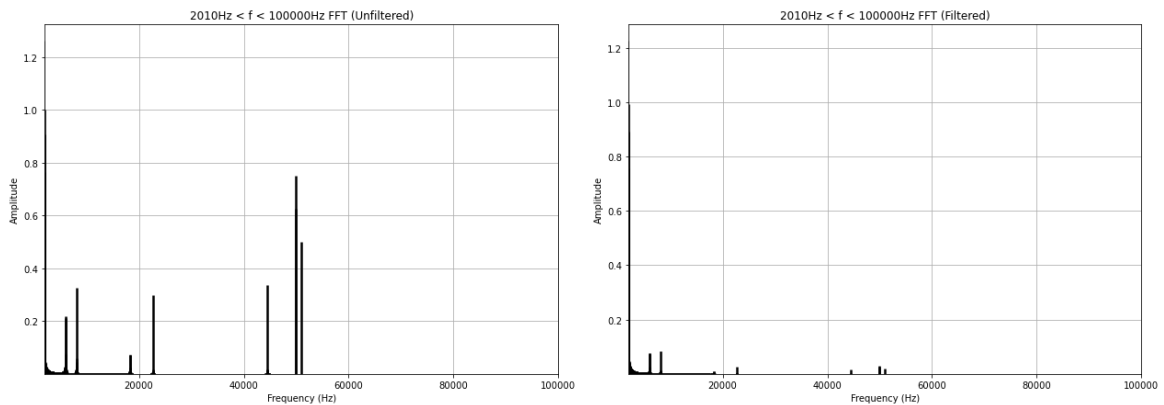
5.4 FFT Plots (Filtered and Unfiltered)



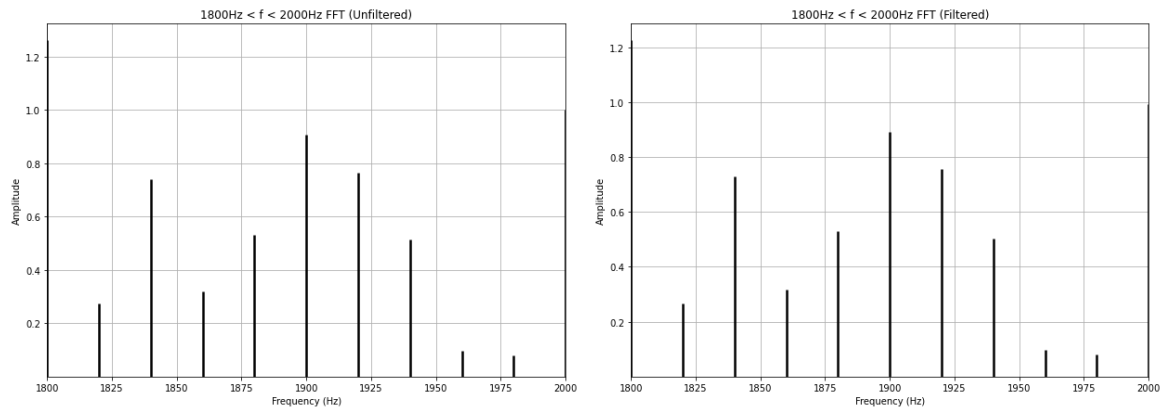
As seen in the comparison between the unfiltered (right), and filtered (left) plots, the amplitudes outside of the ideal range have been almost entirely attenuated.



As seen in the comparison between the unfiltered (right), and filtered (left) plots, the low frequency spike has been almost completely attenuated from the signal, resulting in a clean output from the filter without the initial spike.



As seen in the comparison between the unfiltered (right), and filtered (left) plots, the high frequency amplitudes have been almost completely attenuated, with all values sitting below 0.05v. This provides a clean output from the filter in the high frequency domain.



As seen in the comparison between the unfiltered (right), and filtered (left) plots, the values ranging from 1800Hz to 2000Hz have been minimally impacted when passed through the filter. There is no noticeable change between the input and output values which demonstrates the filter very little effect on the important values.

6 Questions

Earlier this semester, you were asked what you personally wanted to get out of taking this course. Do you feel like that personal goal was met? Why or why not?

I feel this class exceeded my expectations of what I was expecting. I found the labs to be useful in improving my understanding of math functions in Python and the capabilities that Python has in the electrical engineering field.

7 Conclusion

In this project, a noisy signal is recieved from an aircraft positioning system. This noisy signal is then passed through an RLC circuit, which provides a bandpass filter that only allows frequencies of 1800-2000Hz. This output signal meets a variety of criteria for the attenuation from the filter, and provides a clean signal that is usable in further processes. The output signal generated in this project proved to be a very clean filtering of the input signal provided.

8 Appendix

8.1 `.stem()` Workaround

```
def make_stem ( ax ,x ,y , color ='k', style ='solid ', label ='',linewidths =2.5 ,  
    ax . axhline ( x [0] , x [ -1] ,0 , color ='r')  
    ax . vlines (x , 0 ,y , color = color , linestyle = style , label = label , li  
    linewidths )  
    ax . set_ylim ([1.05* y . min () , 1.05* y . max () ])
```