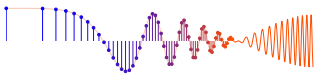


Signal Processing First

Lab 08: Frequency Response: Bandpass and Nulling Filters

Pre-Lab and Warm-Up: You should read at least the Pre-Lab and Warm-up sections of this lab assignment and go over all exercises in the Pre-Lab section before going to your assigned lab session.

Verification: The Warm-up section of each lab must be completed **during your assigned Lab time** and the steps marked *Instructor Verification* must also be signed off **during the lab time**. One of the laboratory instructors must verify the appropriate steps by signing on the **Instructor Verification** line. When you have completed a step that requires verification, simply demonstrate the step to the TA or instructor. Turn in the completed verification sheet to your TA when you leave the lab.

Lab Report: It is only necessary to turn in a report on Section 3 with graphs and explanations. You are asked to **label** the axes of your plots and include a title for every plot. In order to keep track of plots, include your plot *inlined* within your report. If you are unsure about what is expected, ask the TA who will grade your report.

1 Pre-Lab

The goal of this lab is to study the response of FIR filters to inputs such as complex exponentials and sinusoids. In the experiments of this lab, you will use `firfilt()`, or `conv()`, to implement filters and `freqz()` to obtain the filter's frequency response.[1] As a result, you should learn how to characterize a filter by knowing how it reacts to different frequency components in the input.

This lab also introduces two practical filters: bandpass filters and nulling filters. Bandpass filters can be used to detect and extract information from sinusoidal signals, e.g., tones in a touch-tone telephone dialer. Nulling filters can be used to remove sinusoidal interference, e.g., jamming signals in a radar.

1.1 Frequency Response of FIR Filters

The output or *response* of a filter for a complex sinusoid input, $e^{j\hat{\omega}n}$, depends on the frequency, $\hat{\omega}$. Often a filter is described solely by how it affects different input frequencies—this is called the *frequency response*.

For example, the frequency response of the two-point averaging filter

$$y[n] = \tfrac{1}{2}x[n] + \tfrac{1}{2}x[n-1]$$

can be found by using a general complex exponential as an input and observing the output or response.

$$x[n] = Ae^{j(\hat{\omega}n+\phi)} \tag{1}$$

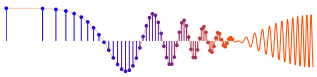
$$y[n] = \tfrac{1}{2}Ae^{j(\hat{\omega}n+\phi)} + \tfrac{1}{2}Ae^{j(\hat{\omega}(n-1)+\phi)} \tag{2}$$

$$= Ae^{j(\hat{\omega}n+\phi)}\tfrac{1}{2}\left\{1+e^{-j\hat{\omega}}\right\} = Ae^{j(\hat{\omega}n+\phi)} \cdot H(e^{j\hat{\omega}}) \tag{3}$$

In (3) there are two terms, the original input, and a term that is a function of $\hat{\omega}$. This second term is the frequency response and it is commonly denoted by $H(e^{j\hat{\omega}})$, which in this case is

$$H(e^{j\hat{\omega}}) = \tfrac{1}{2}\left\{1+e^{-j\hat{\omega}}\right\} \tag{4}$$

[1] If you are working at home and do not have the function `freqz.m`, there is a substitute available called `freekz.m` in the SP First toolbox.



Once the frequency response, $H(e^{j\hat{\omega}})$, has been determined, the effect of the filter on any complex exponential may be determined by evaluating $H(e^{j\hat{\omega}})$ at the corresponding frequency. The output signal $y[n]$, will be a complex exponential whose complex amplitude has a constant magnitude and phase. The phase describes the phase change of the complex sinusoid and the magnitude describes the gain applied to the complex sinusoid.

The frequency response of a general FIR linear time-invariant system is

$$H(e^{j\hat{\omega}}) = \sum_{k=0}^{M} b_k e^{-j\hat{\omega}k} \tag{5}$$

In the example above, $M = 1$, and $b_0 = \frac{1}{2}$ and $b_1 = \frac{1}{2}$.

1.1.1 MATLAB Function for Frequency Response

MATLAB has a built-in function called `freqz()` for computing the frequency response of a discrete-time LTI system. The following MATLAB statements show how to use `freqz` to compute and plot both the magnitude (absolute value) and the phase of the frequency response of a two-point averaging system as a function of $\hat{\omega}$ in the range $-\pi \leq \hat{\omega} \leq \pi$:

```
bb = [0.5, 0.5];          %-- Filter Coefficients
ww = -pi:(pi/100):pi;     %-- omega hat
HH = freqz(bb, 1, ww);    %<--freekz.m is an alternative
subplot(2,1,1);
plot(ww, abs(HH))
subplot(2,1,2);
plot(ww, angle(HH))
xlabel('Normalized Radian Frequency')
```

For FIR filters, the second argument of `freqz( _, 1, _ )` must always be equal to `1`.[2] The frequency vector `ww` should cover an interval of length 2π for $\hat{\omega}$, and its spacing must be fine enough to give a smooth curve for $H(e^{j\hat{\omega}})$. Note: we will always use capital `HH` for the frequency response.

1.2 Periodicity of the Frequency Response

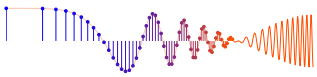
The frequency responses of discrete-time filters are *always* periodic with period equal to 2π. Explain why this is the case by stating a definition of the frequency response and then considering two input sinusoids whose frequencies are $\hat{\omega}$ and $\hat{\omega} + 2\pi$.

$$x_1[n] = e^{j\hat{\omega}n} \qquad \text{versus} \qquad x_2[n] = e^{j(\hat{\omega}+2\pi)n}$$

Consult Chapter 6 for a mathematical proof that the outputs from each of these signals will be identical (basically because $x_1[n]$ is equal to $x_2[n]$.)

The implication of periodicity is that a plot of $H(e^{j\hat{\omega}})$ only needs to extend over the interval $-\pi \leq \hat{\omega} \leq \pi$ or any other interval of length 2π.

[2] If the output of the `freqz` function is not assigned, then plots are generated automatically; however, the magnitude is given in decibels which is a logarithmic scale. For linear magnitude plots a separate call to `plot` is necessary.



1.3 Frequency Response of the Four-Point Averager

In Chapter 6 we examined filters that average input samples over a certain interval. These filters are called "running average" filters or "averagers" and they have the following form for the L-point averager:

$$y[n] = \frac{1}{L}\sum_{k=0}^{L-1} x[n-k] \tag{6}$$

(a) Use Euler's formula and complex number manipulations to show that the frequency response for the 4-point running average operator is given by:

$$H(e^{j\hat{\omega}}) = \frac{2\cos(0.5\hat{\omega}) + 2\cos(1.5\hat{\omega})}{4}\, e^{-j1.5\hat{\omega}} \tag{7}$$

(b) Implement (7) directly in MATLAB. Use a vector that includes 400 samples between $-\pi$ and π for $\hat{\omega}$. Since the frequency response is a complex-valued quantity, use `abs()` and `angle()` to extract the magnitude and phase of the frequency response for plotting. Plotting the real and imaginary parts of $H(e^{j\hat{\omega}})$ is not very informative.

(c) In this part, use `freqz.m` in MATLAB to compute $H(e^{j\hat{\omega}})$ numerically (from the filter coefficients) and plot its magnitude and phase versus $\hat{\omega}$. Write the appropriate MATLAB code to plot both the magnitude and phase of $H(e^{j\hat{\omega}})$. Follow the example in Section 1.1.1. The filter coefficient vector for the 4-point averager is defined via:

```
bb = 1/4*ones(1,4);
```

Note: the function `freqz(bb,1,ww)` evaluates the frequency response for all frequencies in the vector `ww`. It uses the summation in (5), not the formula in (7). The filter coefficients are defined in the assignment to vector `bb`. How do your results compare with part (b)?

1.4 The MATLAB FIND Function

Often signal processing functions are performed in order to extract information that can be used to make a decision. The decision process inevitably requires logical tests, which might be done with `if-then` constructs in MATLAB. However, MATLAB permits vectorization of such tests, and the `find` function is one way to do lots of tests at once. In the following example, `find` extracts all the numbers that "round" to 3:

```
xx = 1.4:0.33:5, jkl = find(round(xx)==3), xx(jkl)
```

The argument of the `find` function can be any logical expression. Notice that `find` returns a list of indices where the logical condition is true. See `help` on `relop` for information.
Now, suppose that you have a frequency response:

```
ww = -pi:(pi/500):pi; HH = freqz( 1/4*ones(1,4), 1, ww );
```

Use the `find` command to determine the indices where `HH` is zero, and then use those indices to display the list of frequencies where `HH` is zero. Since there might be round-off error in calculating `HH`, the logical test should probably be a test for those indices where the magnitude (absolute value in MATLAB) of `HH` is less than some rather small number, e.g., 1×10^{-6}. Compare your answer to the frequency response that you plotted for the four-point averager in Section 1.3.

2 Warm-up

The first objective of this warm-up is to use a MATLAB GUI to demonstrate nulling. This demo, `dltidemo`, is on the *SP First CD-ROM* under Chapter 6 demos.

CD-ROM

Chap 6 Demo DLTI

2.1 LTI Frequency Response Demo

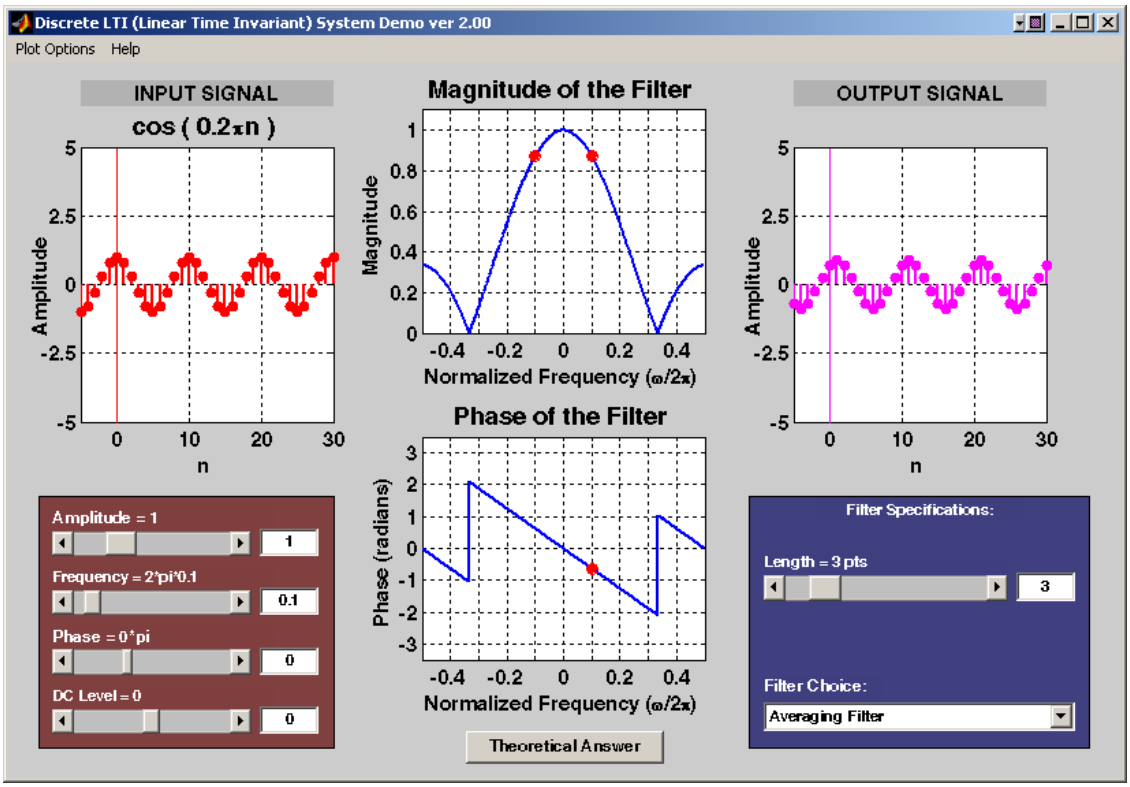


Figure 1: Discrete-time LTI demo interface.

The `dltidemo` illustrates the "sinusoid-IN gives sinusoid-OUT" property of discrete-time LTI systems. In this demo, you can change the amplitude, phase and frequency of an input sinusoid, $x[n]$, and you can change the digital filter that processes the signal. Then the GUI will show the output signal, $y[n]$, which is also a sinusoid (at the same frequency). Figure 1 shows the interface for the `dltidemo` GUI. It is possible to see the formula for the output signal, if you click on the `Theoretical Answer` button located at the bottom-middle part of the window. The digital filter can be changed by choosing different options in the `Filter Specifications` box in the lower right-hand corner.
In the Warm-up, you should perform the following steps with the `dltidemo` GUI:

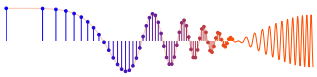
(a) Set the input to $x[n] = 1.5\cos(0.1\pi(n-4))$

(b) Set the digital filter to be a 9-point averager.

(c) Determine the formula for the output signal and write it in the form: $y[n] = A\cos(\hat{\omega}_0(n - n_d))$.

(d) Using n_d for $y[n]$ and the fact that the input signal had a peak at $n = 4$, determine the amount of delay through the filter. In other words, how much has the peak of the cosine wave shifted?

Instructor Verification (separate page)



(e) Now, determine the length of the averaging filter so that the output will be zero, i.e., $y[n] = 0$. Use the GUI to show that you have the correct filter to zero the output. If the length is more than 15, you will have to enter the "Filter Specifications" with the `user Input` option.

(f) When the output is zero, the filter acts as a *Nulling Filter*, because it eliminates the input at $\hat{\omega} = 0.1\pi$. Which other frequencies $\hat{\omega}$ are also nulled? Find at least one.

Instructor Verification (separate page)

2.2 Cascading Two Systems

More complicated systems are often made up from simple building blocks. In Fig. 2, two FIR filters are shown connected "in cascade."

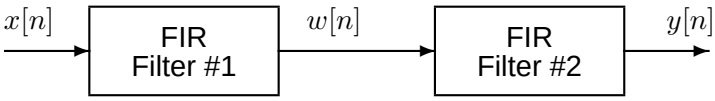


Figure 2: Cascade of two FIR filters.

Assume that the system in Fig. 2 is described by the two equations

$$w[n] = \sum_{\ell=0}^{M} \alpha^{\ell} x[n-\ell] \qquad \text{(FIR FILTER \#1)}$$

$$y[n] = w[n] - \alpha\, w[n-1] \qquad \text{(FIR FILTER \#2)}$$

(a) Use `freqz()` in MATLAB to get the frequency responses for the case where $\alpha = 0.8$ and $M = 9$. Plot the magnitude and phase of the frequency response for Filter #1, and also for Filter #2. Which one of these filters is a *lowpass filter*?

(b) Plot the magnitude and phase of the frequency response of the overall cascaded system.

(c) Explain how the individual frequency responses in part(a) are combined to get the overall frequency response in part(b). Comment on the magnitude combinations as well as the phase combinations.

Instructor Verification (separate page)

2.3 Deconvolution

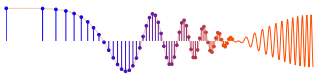
In Lab 7, the two filters from Section 2.2 were used in an image deblurring experiment. You should now re-interpret how that experiment worked by explaining what happens *in the frequency domain.*

(a) If a single filter has a frequency response $H(e^{j\hat{\omega}}) = 1$, how is the output of the filter $y[n]$ related to the input $x[n]$?

(b) Ideally, a "deconvolved" output should look exactly like the input prior to blurring. If Filter #1 (in Fig. 2) has a frequency response $H_1(e^{j\hat{\omega}})$, and Filter #2 is $H_2(e^{j\hat{\omega}})$, explain why the condition $H_1(e^{j\hat{\omega}})H_2(e^{j\hat{\omega}}) = 1$ will guarantee "perfectly deconvolution."

(c) The filters in Section 2.2 do not perform a perfect deconvolution (for the case $\alpha = 0.8$ and $M = 9$). Use the frequency response from Section 2.2(b) to explain deviations from a perfect result.

Instructor Verification (separate page)



3 Lab Exercises

3.1 Nulling Filters for Rejection

Nulling filters are filters that completely eliminate some frequency component. If the frequency is $\hat{\omega} = 0$ or $\hat{\omega} = \pi$, then a two-point FIR filter will do the nulling. The simplest possible general nulling filter can have as few as three coefficients. If $\hat{\omega}_{\rm n}$ is the desired nulling frequency, then the following length-3 FIR filter

$$y[n] = x[n] - 2\cos(\hat{\omega}_{\rm n})x[n-1] + x[n-2] \tag{8}$$

will have a zero in its frequency response at $\hat{\omega} = \hat{\omega}_{\rm n}$. For example, a filter designed to completely eliminate signals of the form $Ae^{j0.5\pi n}$ would have the following coefficients because we would pick the desired nulling frequency to be $\hat{\omega}_{\rm n} = 0.5\pi$.

$$b_0 = 1, \quad b_1 = -2\cos(0.5\pi) = 0, \quad b_2 = 1.$$

(a) Design a filtering system that consists of the *cascade of two FIR nulling filters* that will eliminate the following input frequencies: $\hat{\omega} = 0.44\pi$, and $\hat{\omega} = 0.7\pi$. For this part, derive the filter coefficients of both nulling filters.

(b) Generate an input signal $x[n]$ that is the sum of three sinusoids:

$$x[n] = 5\cos(0.3\pi n) + 22\cos(0.44\pi n - \pi/3) + 22\cos(0.7\pi n - \pi/4)$$

Make the input signal 150 samples long over the range $0 \leq n \leq 149$.

(c) Use `firfilt` (or `conv`) to filter the sum of three sinusoids signal $x[n]$ through the filters designed in part (a). Show the MATLAB code that you wrote to implement the cascade of two FIR filters.

(d) Make a plot of the output signal—show the first 40 points. Determine (by hand) the exact mathematical formula (magnitude, phase and frequency) for the output signal for $n \geq 5$.

(e) Plot the mathematical formula determined in (d) with MATLAB to show that it matches the filter output from `firfilt` over the range $5 \leq n \leq 40$.

(f) Explain why the output signal is different for the first few points. How many "start-up" points are found, and how is this number related to the lengths of the filters designed in part (a)? Hint: consider the length of a single FIR filter that is equivalent to the cascade of two length-3 FIRs.

3.2 Simple Bandpass Filter Design

The L-point averaging filter is a lowpass filter. Its passband width is controlled by L, being inversely proportional to L. In fact, you can use the GUI `dltidemo` to view the frequency response for different averagers and measure the passband widths. It is also possible to create a filter whose passband is centered around some frequency other than zero. One simple way to do this is to define the impulse response of an L-point FIR as:

$$h[n] = \frac{2}{L}\cos\left(\hat{\omega}_c n\right), \qquad 0 \leq n < L$$

where L is the filter length, and $\hat{\omega}_c$ is the center frequency that defines the frequency location of the passband. For example, we would pick $\hat{\omega}_c = 0.44\pi$ if we want the peak of the filter's passband to be centered at 0.44π. The bandwidth of the bandpass filter is controlled by L; the larger the value of L, the narrower the bandwidth. This particular filter is also discussed in the section on useful filters in Chapter 7 of the text.

(a) Generate a bandpass filter that will pass a frequency component at $\hat{\omega} = 0.44\pi$. Make the filter length (L) equal to 10. Since we are going to be filtering the signal defined in section 3.1(b), measure the gain of the filter at the three frequencies of interest: $\hat{\omega} = 0.3\pi$, $\hat{\omega} = 0.44\pi$ and $\hat{\omega} = 0.7\pi$.

(b) The *passband* of the BPF filter is defined by the region of the frequency response where $|H(e^{j\hat{\omega}})|$ is close to its maximum value. If we define the maximum to be $H_{\max}$, then the passband width is defined as the length of the frequency region where the ratio $|H(e^{j\hat{\omega}})|/H_{\max}$ is greater than $1/\sqrt{2} = 0.707$. Figure 3 shows how to define the passband and stopband. Note: you can use MATLAB's `find` function to locate those frequencies where the magnitude satisfies $|H(e^{j\hat{\omega}})| \geq 0.707 H_{\max}$.

The *stopband* of the BPF filter is defined by the region of the frequency response where $|H(e^{j\hat{\omega}})|$ is close to zero. In this case, we will define the stopband as the region where $|H(e^{j\hat{\omega}})|$ is less than 25% of the maximum.

Make a plot of the frequency response for the $L = 10$ bandpass filter from part (a), and determine the passband width (at the 0.707 level). Repeat the plot for $L = 20$ and $L = 40$, so you can explain how the width of the passband is related to filter length L, i.e., what happens when L is doubled or halved.

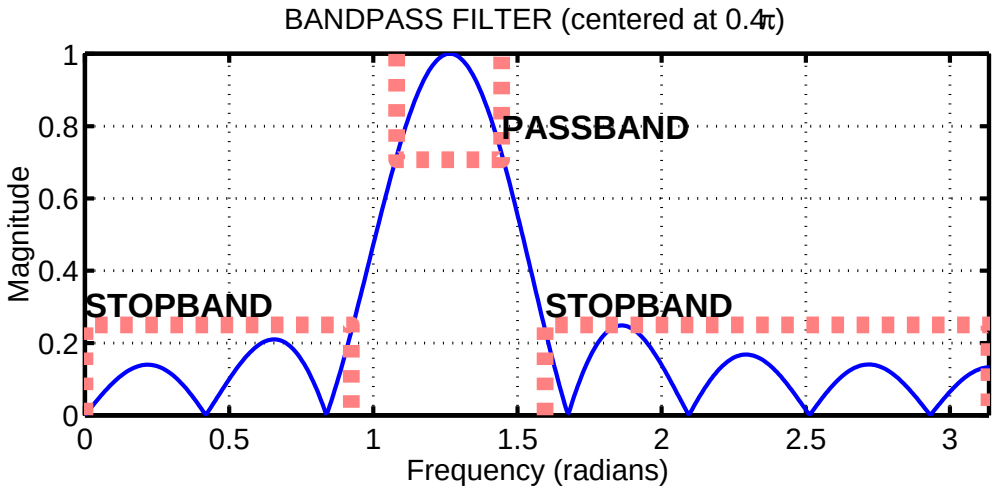


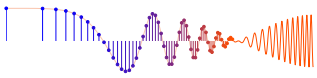
Figure 3: Passband and Stopband for a typical FIR bandpass filter. In this case, the maximum value is 1, the passband is the region where the frequency response is greater than $1/\sqrt{2} = 0.707$, and the stopband is defined as the region where the frequency response is less than 25% of the maximum.

(c) Comment on the selectivity of the $L = 10$ bandpass filter. In other words, which frequencies are "passed by the filter?" Use the frequency response to explain how the filter can pass one component at $\hat{\omega} = 0.44\pi$, while reducing or rejecting the others at $\hat{\omega} = 0.3\pi$ and $\hat{\omega} = 0.7\pi$.

(d) Generate a bandpass filter that will pass the frequency component at $\hat{\omega} = 0.44\pi$, but now make the filter length (L) long enough so that it will also *greatly* reduce frequency components at (or near) $\hat{\omega} = 0.3\pi$ and $\hat{\omega} = 0.7\pi$. Determine the *smallest* value of L so that

- Any frequency component satisfying $|\hat{\omega}| \leq 0.3\pi$ will be reduced by a factor of 10 or more.[3]
- Any frequency component satisfying $0.7\pi \leq |\hat{\omega}| \leq \pi$ will be reduced by a factor of 10 or more.

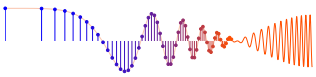
[3]For example, the input amplitude of the 0.7π component is 22, so its output amplitude must be less than 2.2.



This can be done by making the passband width very small.

(e) Use the filter from the previous part to filter the "sum of 3 sinusoids" signal from Section 3.1. Make a plot of 100 points of the input and output signals, and explain how the filter has reduced or removed two of the three sinusoidal components.

(f) Make a plot of the frequency response (magnitude only) for the filter from part (d), and explain how $H(e^{j\hat{\omega}})$ can be used to determine the relative size of each sinusoidal component in the output signal. In other words, connect a mathematical description of the output signal to the values that can be obtained from the frequency response plot.



Lab 08
INSTRUCTOR VERIFICATION PAGE

For each verification, be prepared to explain your answer and respond to other related questions that the lab TA's or professors might ask. Turn this page in at the end of your lab period.

Name: _____ Date of Lab: _____

Part 2.1(d) Use the `dltidemo` to illustrate the operation of a 9-point averaging filter. Determine the amount of delay through the filter, and write your answer in the space below.

Verified:_____ Date/Time:_____

Part 2.1(f) Use the `dltidemo` to find a digital FIR filter that will null the input signal. Determine the filter length, and write your answer in the space below.

Verified:_____ Date/Time:_____

Part 2.2 Plot the frequency response of the two filters in the cascade combination, and then explain how the magnitudes are combined and how the phases are combined to get the overall filter. Check the range of frequencies $(\hat{\omega})$ used for the plot.

Verified:_____ Date/Time:_____

Part 2.3 Explain how close the frequency response of the cascaded system is to the desired frequency response for "ideal deconvolution."

Verified:_____ Date/Time:_____

McClellan, Schafer, and Yoder, *Signal Processing First*, ISBN 0-13-065562-7.
Prentice Hall, Upper Saddle River, NJ 07458.