EE432 Fall 2011 Project 09: Filter Design: FIR vs IIR (Due: 11/15/2011)

In this lab, you will investigate how FIR filters compare to IIR filters. Work in a group of 2 people or less. Chipman & the Buckwheat Boyz had an audio problem while recording their latest (and only) hit…some electrical feedback in the microphone caused an annoying tone to show up in the recording. Download the "PeanutButterJellyTime-noisy.wav" file and bring it into MATLAB. Listen to it, so you hear the corrupting noise.Bring the signals into MATLAB and play them using *wavplay*.

I. Finite Impulse Response (FIR) Filter: Annoying Single Tone

Chipman & the Buckwheat Boyz had an audio problem while recording their latest (and only) hit…some electrical feedback in the microphone caused an annoying tone to show up in the recording. Download the "PeanutButterJellyTime-noisy.wav" file and bring it into MATLAB. Bring the signal into MATLAB and play it using *wavplay*. Listen to it, so you hear the corrupting noise.

Design a **band stop FIR** filter using *fdatool* to remove (or at least significantly reduce) this tone while minimizing the distortion of the rest of the signal. That is, the magnitude response of the filter should be as flat as possible in the pass band, and the transition bands should be as steep as possible. Use the minimum order for your filter. Keep in mind that for FIR filters, there is the tradeoff between # coefficients to give steep transition bands, and the ripple they produce in the pass band. For your filter, do the following:

1. Write on a copy of the attached design spec plot your chosen parameters for the pass bands, stop band and transition bands. Include the type of FIR filter used (the type is up to you) and its order (which is the # of coefficients).

2. Export your filter's coefficients, then use *fvtool* to plot the magnitude response and the pole/zero plot. Label and turn in these plots. How many zeros are in the filter?

3. Filter the music clip with your filter, write out the filtered clip using *wavwrite* (be sure to include the sample frequency when you write it out), and email the wav file to the professor.

4. Is the filter stable? How do you know?

II. Infinite Impulse Response (IIR) Filter: Annoying Single Tone

Design a band stop IIR filter using *fdatool* to remove (or at least significantly reduce) this tone while minimizing the distortion of the rest of the signal. That is, the magnitude response of the filter should be as flat as possible in the pass band, and the transition bands should be as steep as possible. For your filter, do the following:

1. Use the same stop band and pass band parameters to design your IIR filter. Write on a copy of the attached design spec plot your chosen parameters for the pass bands, stop band and transition bands. Include the type of IIR filter used (the type is up to you) and its order.

2. Export your filter to the workspace as an object, NOT as coefficients. The default name for the filter object is "Hd", but you can change it if you wish. Use *fvtool* to plot the magnitude response and pole/zero plot. *fvtool* works with filter objects as well as it does with filter

coefficients…if the filter object is called "Hd", then you just type `fvtool(Hd)`. Turn in these plots. How many poles and how many zeros are in the filter?

3. Filter the music clip with your filter (note: *filter* can use filter objects just like *fvtool* can). Write out the filtered file into a wav file called "Project09-2.wav", and email it to the professor. Be sure that you write out the sample frequency.

4. Is the filter stable? How do you know?

III. Comparing FIR vs IIR

You have designed FIR filters and an IIR filter to do exactly the same thing. Now answer the following questions to compare the two.

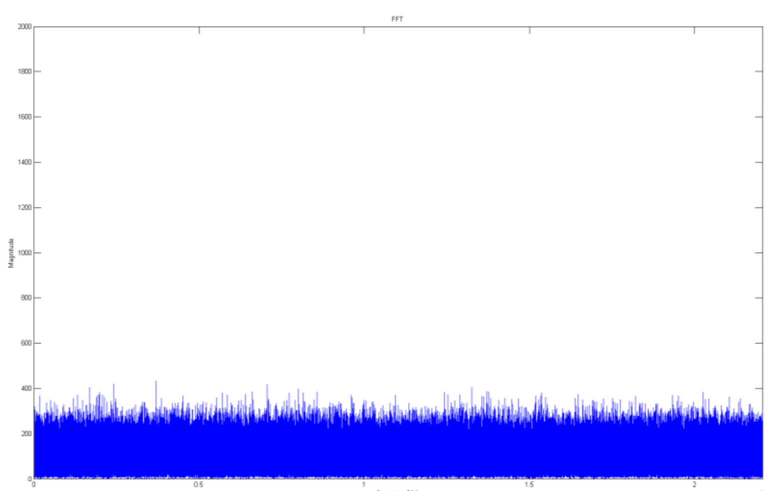
1. How do your filters compare in terms of number of poles and zeros?

2. How do your filters compare in terms of pass band flatness, and transition band width?

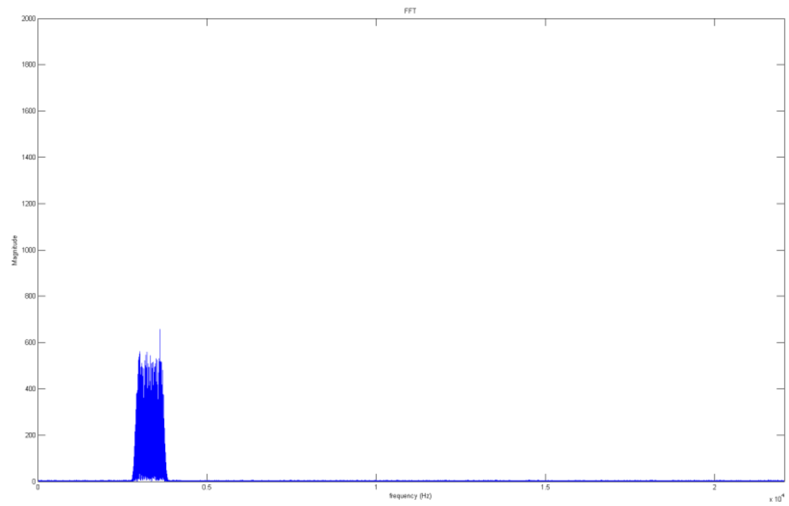
3. Which filter would seem to be the one you should use for this task?

IV. Colored Noise (use FIR or IIR, your choice)

White noise is called "white noise" because it is comparable to white light. White light is made up of all colors equally; white noise is made up of all frequencies equally. If we were to plot the magnitude of the DFT of a signal that is white noise, the magnitude would appear ~flat, since it is comprised of all frequencies equally. When the term "white noise" is used, we usually think of white Gaussian noise…that is, noise with values that are from a Gaussian random process. Below is a plot of the magnitude spectrum of a Gaussian white noise signal, with f_s = 44.1 kHz. The magnitude spectrum is essentially flat.



Colored noise is noise where the magnitude spectrum is flat, but only over a portion of the entire spectrum. This may be a large portion of the spectrum or a smaller portion. Below is the magnitude spectrum of a colored noise signal, with frequency content from ~2800 Hz to ~3800 Hz (sample frequency was 44.1 kHz).



Do the following:

1. Download the "Mozart-EineKleineNachtmusik-colored noise.wav" file and bring it into MATLAB. Listen to it, so you hear the corrupting noise.

2. The colored noise is present throughout the signal; analyzing any portion will give an indication of its frequency band, particularly a portion of the clip before the music starts playing. Using a portion of the noisy signal, determine the nature of the colored noise that is affecting this signal (record the frequency band that contains this colored noise and the average magnitude (in dB) in its frequency band):

   _______________ Hz to __________________ Hz, Magnitude: ___________dB

3. Design a band stop filter to remove or severely attenuate the band of noise and filter the noisy signal. Listen to your result. It should sound MUCH better.

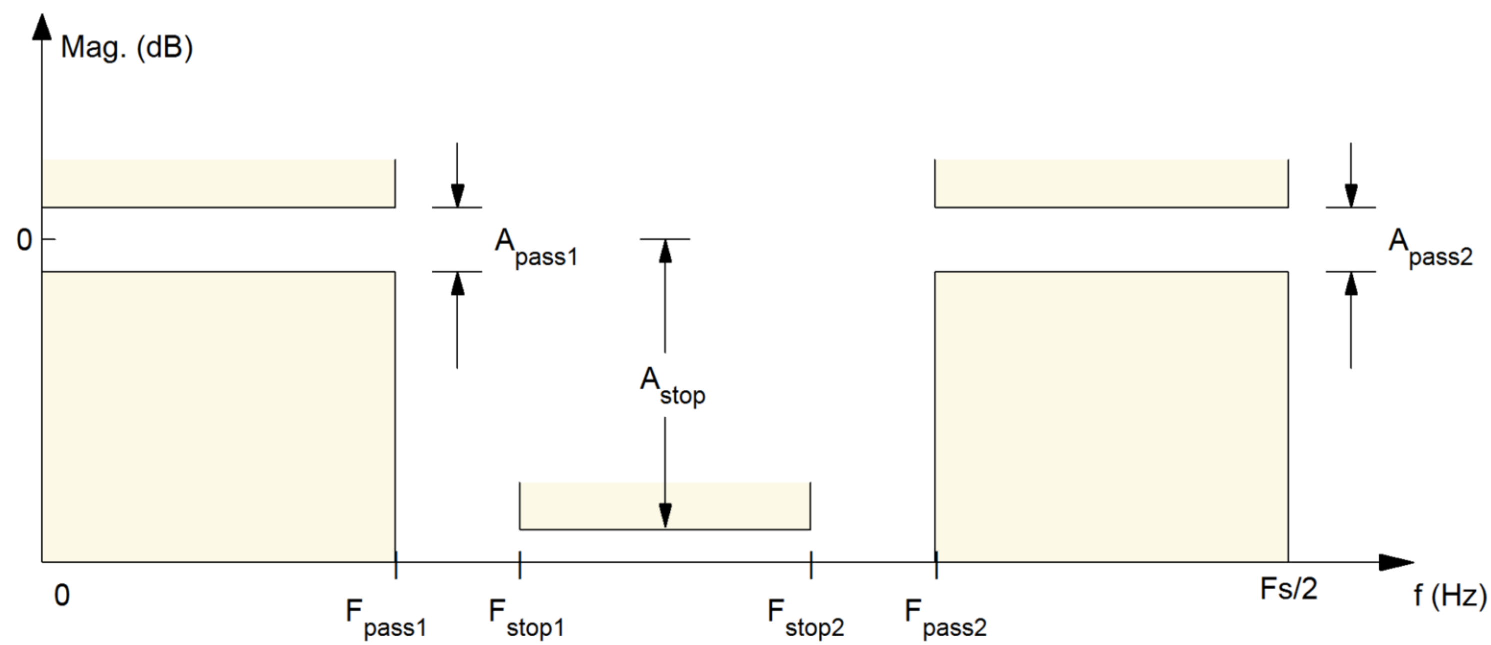
4. Create a screen capture of the *fdatool* GUI after you have designed the filter and include it with your write up, so I can see your design.

5. Write out the filtered file into a wav file called "Project09-2.wav", and email it to the professor. Be sure that you write out the sample frequency.

For this project write up, turn in the plots, fill in the attached forms for your filter designs, answer the questions, and email the wav files to the professor

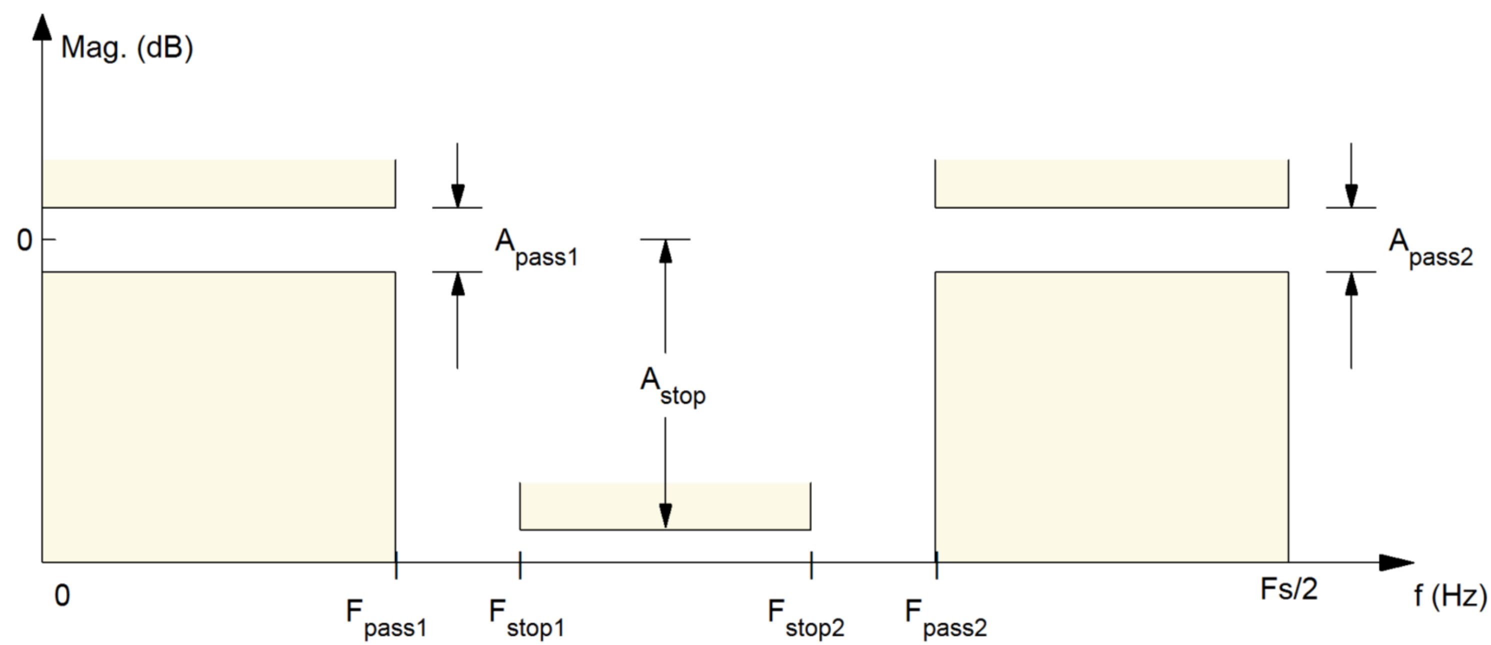
Part I: FIR Filter
Filter description:

Filter order:



Part II: IIR Filter
Filter description:

Filter order:



Filter, Part IV:
Filter description:

Filter order: