# Lab #7: FIR and IIR Filter Design

## Purpose

The purpose of this lab is to experiment and hear the results of implementing a digital low & high pass FIR filters and a band pass IIR filter. In a previous semester students created sound files that consisted of mixing speech with noise to create a "secret message". In some of these files, the message cannot be discerned and in others you can easily determine the message being spoken. We will focus on trying to apply filters to the truly secret message files but they are all available online for listening purposes.

The point of the experiment here is to see if we can improve the intelligibility of the message when using a filter technique. For example on message files that appear to contain high frequency noise, we'll employ a low pass filter. For files that appear to have low frequency noise we'll use a high pass filter. Since speech is band limited in a range discussed below, we'll also try a band pass filter.

Finally, you will be asked to run the sound files I have generated via Matlab that contain music + noise through a filter type. Throughout this testing, compare & contrast the filter results.

## Part I. Filtering the Secret Message with a Simple Low Pass FIR Filter

1. Here is a nice link that shows the spectrum of human speech:

***http://www.bnoack.com/index.html?http&&&www.bnoack.com/audio/speech-level.html***

Notice the big dB drop below 100 Hz and above 1.5KHz. The bulk of the energy appears to be somewhere between 300 - 1 KHz. Design the following digital Low Pass Filter therefore to remove noise energy above 1K Hz:

***LPF, Pass Band (3 dB ripple) 0-1400 Hz, Stop Band (-40 dBs) 3K-24K, Fs = 48 KHz;***

2. Implement your FIR on your DSP in **'C' code** **and use DSP Mode on the Codec to optimize the sampling**. Sweep a sine wave to verify the correct pass, transition and stop bands. Record the dB gain or attenuation on paper for at least two points in the Pass Band, Transition Band and Stop Band. For each of these frequencies, record the phase shift observed at the output (in degrees and radians).

3. Input & play out the following "secret message" sound files to see if there is an improvement in the ability to decode the message: Travis_Wise_SecretMessage.mp3, DanielKellySecretMessage.wav, Kevin_French_SecretTunnel.wav

**Special Note:** Since we are sampling at 48KHz and the sound files have been generated with 44KHz sampling, there should not be a problem with aliasing. However, if you find that you have to lower the sampling rate due to your filter execution time being too long then you will need to play out the sound file using a PC Equalization (EQ) Ap that can be used to turn down the gain on all EQ frequencies above your Fs/2 sampling rate. i.e. You sample at 32 KHz but our signal has frequency content up to 20 KHz, turn down all the EQ sliders from say 12K Hz and up on the ap playing out the sound to kill the frequencies above your Nyquist sampling frequency.

4. Now play out the file that I created which contains music + high frequency noise with and without your filter found online: HighFreqNoise_Crash_Test1.mp3 How do the results compare with this file versus the sound files made by the students?

5. Time the FIR code in your timer interrupt by toggling a port pin and measuring the FIR execution time with your DAD.

# Lab #7: FIR and IIR Filter Design

**Part II. Filtering the Sound Files with a High Pass FIR Filter**

1. Design a High Pass Filter to **eliminate the speech** with the following specifications:

***HPF, Stop Band (-30 dBs) 0 - 2000 Hz, Pass Band (3 dB ripple) 4000-24000 Hz, Fs = 48 KHz;***

2. Implement your FIR on the DSP using 'C' code. Sweep a sine wave to verify the correct pass, transition and stop bands. Record the dB gain or attenuation on paper for at least two points in the Stop, Pass and Transition Bands. For each of these sine inputs, record the output phase shift as was done earlier.

Once again time your interrupt code via pulsing an I/O pin and measure the execution time using your DAD. If possible, increase the sampling rate, lower the stop band attenuation (by increasing the number of filter taps) to create a more robust filter.

3. Input & play out these files (with & without filtering) to observe the effect of your HPF: Eric (the kid) SchwartzSecretMessage.wav and Hannah Mellott's .wav file. Is the speech attenuated with your filter? Finally, then run LowFreqNoise_Loops1.mp3 to hear the effect of removing the low frequency noise content.

**Part III. Filtering the Sound Files with a Band Pass IIR Filter**

1. Design an IIR Band Pass Filter with the following specifications:

***BPF, Stop Band (-40 dBs) 0 - 100 Hz, Pass Band (3 dB ripple) 400-800 Hz, Stop Band 1.2K-24K, Fs = 48 KHz***

2. Implement your IIR on the **DSP in 'C' code** **and use DSP Mode on the Codec to optimize the sampling.** Sweep a sine wave to verify the correct pass, transition and stop bands. Record the dB gain or attenuation on paper for at least two points in the Stop Band, Pass Band and Stop Band. For each of these Sine inputs, record the phase shift observed at the output (in degrees and radians). Input & play out the previously mentioned sound files to test your filter.

**Parts I - III. Pre-Lab Requirements**

Printed out in hard copy form all code written along with the filter coefficient information.

**Parts I - III. In-Lab Requirements.**

Demonstrate to your TA a functional filters via sine sweep. Show the interrupt timing and phase results. Filter/play out the requested sound files as mentioned earlier in this document.

Write down your high level results and bring to class. i.e. Which filters worked on which sound files? Which filters didn't seem to work at all? Did the filtering improve speech clarity? Did it remove speech for a particular sound file?

**Lab Point Break-down**

| | | |
|---|---|---|
| In-Lab Quiz | 30% | |
| Part I - IV. Pre-lab Materials | | ; print out your Main & Interrupt Service Routines for each Part I-III. |
| Functional Part I In-Lab | 20% | ; show code, working filter, record timing and phase results for the sine sweep test |
| Functional Part II In-Lab | 20% | ; show code, working filter, timing/phase measurements/demo sound |
| Functional Part III In-Lab | 30% | ; show code, working filter, timing/phase measurements/demo sound |