Lab 3 Part 1: FIR Filter Design

1 Overview

1.1 Introduction

The FIR Filter Design Lab is split into two lab sessions:

I. In the first session, we will focus on the design of FIR filters in Matlab, understand how noise can be removed using a designed filter and examine the effects of quantisation on filter performance.

II. Next week, in the second part of the FIR Filter lab, we will implement our designed filters on hardware using the PYNQ-Z2 board.

1.2 Learning Outcomes

On completing this lab, you will be able to:

- Become familiar with using the filterDesigner (previously fdatool) tool in Matlab for designing FIR filters
- Obtain filter coefficients for different FIR designs specifications
- Analyse a corrupted audio file and design a suitable filter to remove the noise
- Analyse the effects of quantising the filter coefficients

2 Using Matlab's Filter Designer Tool

The Matlab filterDesigner is used for designing and analysing filters. To open the GUI, type *filterDesigner* into the command window. Figure 1 shows a screenshot of the interface that will appear on your screen. To design a filter, you can enter the desired specifications into the GUI and click the *design filter* button to implement it. The magnitude response of the filter will then be displayed in the window.

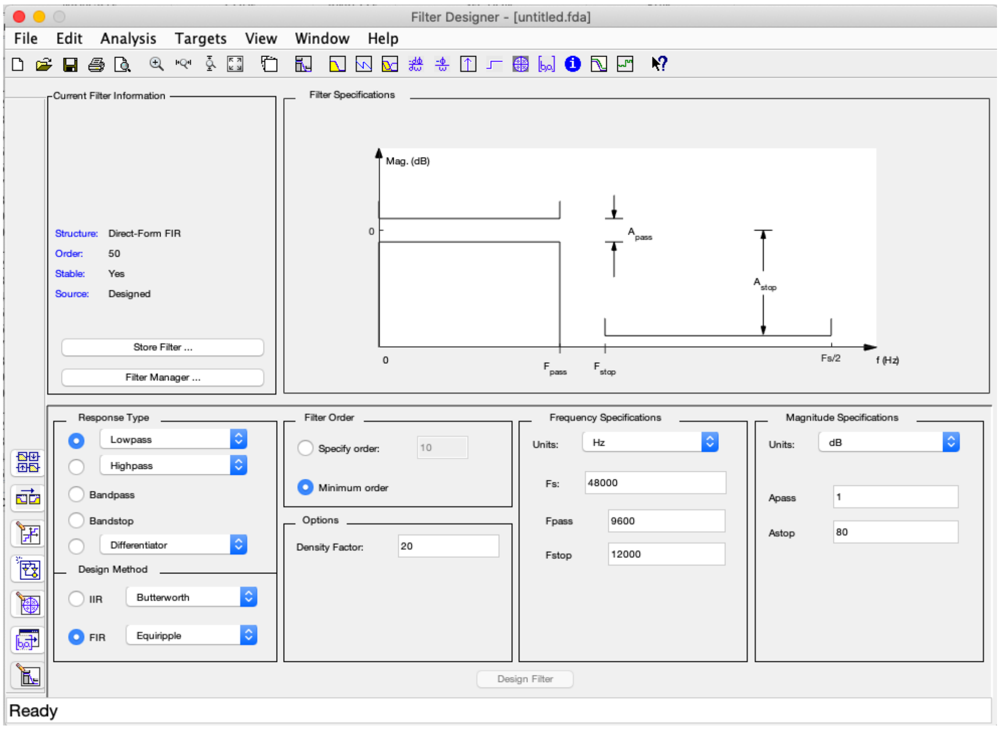

Figure 1: The Filter Designer GUI

- **Storing Filters:** You can store filters by clicking the *store filter* button and access stored filters using the *filter manager* button.

- **Generating Matlab Code:** The filter can also be exported as Matlab code by clicking File → Generate Matlab Code.

3 Filter Design Exercise

3.1 Problem Description

Design a minimum order lowpass FIR filter using the filterDesigner tool to meet the following specifications:

- Stopband attenuation > 90dB
- Passband ripple < 0.02 dB
- Passband edge frequency 3.375 kHz
- Stopband edge frequency 5.625 kHz
- Sampling frequency 20 kHz

3.2 Lab Report Questions

I. What is the order of the filter you designed? What does this mean?
II. What are the effects of altering the passband and stopband attenuation?

Include screenshots of the magnitude response of your filter design to support your answers where appropriate.

---

4 Filtering a Noisy Audio File

4.1 Problem Description

You will now design an FIR filter to filter out noise from a speech .wav file. You should be able to hear the noise when you listen to the sample (if not, please contact me). Either use the filterDesign matlab tool or write a script to design an appropriate filter.

Everyone has been assigned a speech file with a slightly different tone. Do not assume that the quality of the filtered speech file for everyone will be the same. It will depend on exactly where your interfering tone is. Some people's filter output will sound worse than others; you will be graded on your demonstrated understanding and application of filter design principles. You will need to decide what is reasonable for **your** speech file.

Please refer to the Appendix to see your assigned audio file.

4.2 Instructions

I. **Identify the Noise:** First, you need to identify the frequency range you want to filter out. The Discrete Fourier Transform (DFT) allows you to examine the frequency content of the wavefile. The fft() function in Matlab is a fast implementation of the DFT. A pure tone has been added to each file at a different frequency which you will need to remove. You should read in and analyse the frequency content of the file in Matlab.

Sample Code:

```
[x,Fs] = audioread('speech_0.wav'); % Read in audio file

% Plot to find out which frequency to remove from signal
nfft = 2^10;
X = fft(x, nfft);
fstep = Fs/nfft;
fvec = fstep*(0: nfft/2-1);
```

```
fresp = 2*abs(X(1:nfft/2));
plot(fvec,fresp)
title('Single-Sided Amplitude Spectrum of x(t)')
xlabel('Frequency (Hz)')
ylabel('|X(f)|')
```

This code will generate a figure showing the frequency content of the speech file. There should be a strong peak at a certain frequency. This is the noisy pure tone that was added. Write Matlab code to automatically capture the frequency of the tone and record this plot for your write-up (hint: max and argmax functions). Include this code in your Matlab script for submission with each line commented to demonstrate you understand its purpose.

II. **Design the FIR:** Use the filterDesigner tool in Matlab to design an FIR filter to remove the noise. You will need to define your passband(s) and stopband(s) etc. appropriately. Outline your decision process in your write-up.

Pass your audio file through the filter as demonstrated in class, then listen to the output. Is the noise gone? How clear or intelligible is the speech? If it is not clear, is there a problem with your filter?

Redo the frequency analysis on the filtered signal to verify this. Record the filter magnitude response for your write-up.

III. **Quantise the FIR Filter:**

Once you are satisfied that the filter operates correctly, experiment with quantising the coefficients. Compare the magnitude response for full precision and different levels of coefficient quantisation. What do you notice? Listen to the output. What do you notice?

At this point you may wish to reassess the order of the filter you originally designed, to account for quantisation. Discuss the trade-offs considered with your parameter choices.

Record the magnitude response at 3 different levels of quantisation for your write-up, these levels should be 'under-quantised', 'appropriately quantised' and 'over-quantised'. Indicate the final number of bits you will use for the filter coefficients. Include the steps taken to produce these in your submitted Matlab code.

4 Submission

Please submit a **brief** lab report containing:

- The lab report questions and screen captures from the first filter design exercise (refer to Part 3).
- Details of your decision process to design the noise-removing FIR filter.
- Graphs of the frequency response before and after the audio is filtered.
- Your method for quantising the filter coefficients and graphs of the frequency response after applying the 'under-quantised', 'appropriately quantised' and 'over-quantised' filters.
- Discuss the trade-offs you considered for selecting the number of bits for your final filter.

Please submit the following in a zipped folder using your name and lab3_p1 as the file name (e.g. AWalsh_lab3_p1):

- Your **brief** lab report titled in the format AWalsh_lab3_p1.pdf
- Your Matlab code used to generate filter coefficients, quantise them, estimate hardware cost, generate frequency responses and filter the data.
- Attach your filtered speech file (.wav) filtered with the full-precision FIR.
- Attach your filtered speech file (.wav) filtered with the final/chosen quantised FIR.

---

5 Appendix: Audio File Assignment

If your name does not appear in the table below please contact me to be assigned a speech file.

Student Name:		Assigned speech file
		speech_1.wav
		speech_2.wav
		speech_3.wav
		speech_4.wav
		speech_5.wav
		speech_6.wav
		speech_7.wav
		speech_8.wav

		speech_9.wav
		speech_10.wav
		speech_11.wav
		speech_12.wav
		speech_13.wav
		speech_14.wav
		speech_15.wav
		speech_16.wav
		speech_17.wav
		speech_18.wav
		speech_19.wav
		speech_20.wav
		speech_21.wav
		speech_22.wav
		speech_23.wav
		speech_24.wav
		speech_25.wav