Lab Objectives

- Become familiar with the MATLAB toolbox routines for designing IIR filters and MATLAB window functions for designing FIR filters.
- Use the FFT algorithm to accurately examine periodic signals.

Assignment

Task 1: Filter Design:

**FIR Window Method** Write a set of MATLAB functions/scripts to aid you in designing Linear Phase FIR filters to meet a set of stated design specs. You may use any MATLAB window functions, but you are not allowed to use the FIR design functions in the signal processing toolbox.

Using your design tool, design a linear phase low-pass filter with less than 0.1 dB of ripple in the passband \((0 < |f| < 0.07)\), and more than 80 dB of rejection in the stopband \((0.11 < |f| < 0.5)\). Try to minimize the required filter order.

Your scripts/functions should be flexible enough to design a wide variety of filter magnitude responses, including responses with non-flat passband characteristics. FIR filter designs must have real coefficients, and have linear phase characteristics. Your end filter coefficients must be accessible within the Matlab workspace (provide the impulse response in a vector with length equal to the number of terms in your design).

**IIR Designs** Write a set of MATLAB functions to aid you in designing IIR filters to meet a set of stated design specs. You may use MATLAB toolbox routines or the `fdatool` for designing IIR filters. In particular, you should be able to use design elliptic, Chebyshev Type I or Type II, or Butterworth filters (see MATLAB’s ‘ellip’, ‘cheby1’, ‘cheby2’, and ‘butter’ functions.)

Once you have the IIR design tools at hand, design a band-pass filter for a system with a sampling frequency of 48 ksps. The magnitude response of the filter must meet the specifications:

<table>
<thead>
<tr>
<th>Condition</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>(9.9 \text{ dB} \leq 20\log_{10}(</td>
<td>H(F)</td>
</tr>
<tr>
<td>(2\log_{10}(</td>
<td>H(F)</td>
</tr>
<tr>
<td>(2\log_{10}(</td>
<td>H(F)</td>
</tr>
<tr>
<td>(12 \text{ kHz} &lt;</td>
<td>F</td>
</tr>
<tr>
<td>(</td>
<td>F</td>
</tr>
</tbody>
</table>

As for the FIR case, try to minimize the required filter order.

Your end filter design coefficients must be accessible within the Matlab workspace (Provide two vectors: one holding the denominator coefficients for \(H(z)\), and the other holding the numerator coefficients. Biquad or other filter structures will not be graded.)

You will be asked to demonstrate a random filter design during your group meeting. All group members are expected to be familiar with the filter design methodology, and the use of your scripts. Lower filter orders will receive better grades.

Task 2: FFT Analysis of Periodic Signals Become familiar with the use of the DFT to analyze a sampled signal containing multiple periodic waveforms. By using window functions and zero-padding, you should be able to accurately determine the frequency and amplitude of sampled signal(s). Essentially, for you’ll be asked to demonstrate the techniques that you developed in Homework 4, Problem 4.

You’ll be asked to analyze an array of sampled data during your lab demonstration period. For this demonstration, you’ll have access to Matlab, as well as any Matlab scripts or functions that you submit as a part of this lab.

Task 3: Make Lab 4 Work If you were unable to demonstrate working lab 4 software on the STM development board, you must complete the lab as a part of Lab 5.
What should be handed in:

Writeup: Your writeup should include the following:

(a) A short description of the functions and scripts that you created and how they are used.

(b) A description of the use of your routines to design the above required filters, and plots showing the required filter specifications given above have been met. State the resulting filter order for your designs, and indicate the delay associated with your designs. (For the IIR design, provide a plot of the group delay for the filter.)

(c) Provide a written description of why one would use zero-padding or window functions in the DFT analysis of periodic signals. Your description should include illustrative plots showing the effectiveness of these techniques.

(d) Include listings of any scripts or functions that you submitted.

Program listings and Documentation will be collected at the beginning of class on the due date. Write-ups handed in after class begins, but before 1:00pm on the due date, will receive a deduction of one letter grade for the “written lab summary” portion the lab grade. Labs handed in after 1:00pm will not be accepted.

Matlab Code: All programs must be archived in your group’s GIT repository the code due time. Any code archived after the due time will not be graded. Please include all files in the “lab5” subdirectory of your group’s project. Instructions for submitting your programs are provided on the course web site.

The emphasis in this lab is for you to create any scripts which allow you to effectively perform the above tasks. You are NOT expected to create a polished professional filter design package. Your writeup must show that you understand the design process, and your code should be sufficient for you to effectively perform any requested design.

Demonstration Please bring your laptop to your lab demonstration meeting in the week of the due date. You’ll be given access to (only) Matlab and the scripts/functions that your group created and checked into your git repository. You’ll be asked to perform filter designs or analyze sampled data sequences during your demonstration.

If you were unable to successfully demonstrate Lab 4, be prepared to show that you have working lab 4 implementations at your lab 5 demonstration.