ECE 535 DIGITAL SIGNAL PROCESSING

Matlab Project 4

Spring 2001

Issued: Tuesday, April 10, 2001 **Due:** Tuesday, April 24, 2001

Part I: Filter Design

An analog signal consists of the sum of two components $x_1(t)$ and $x_2(t)$. The approximate spectral characteristics of $x(t)$ are shown in the sketch of Figure 1. The signal $x(t)$ is bandlimited to 40 kHz and it is sampled at a rate of 100 kHz to yield the sequence $x[n]$. Note that the figure is for illustration purposes only; it does not define the signal $x(t)$.

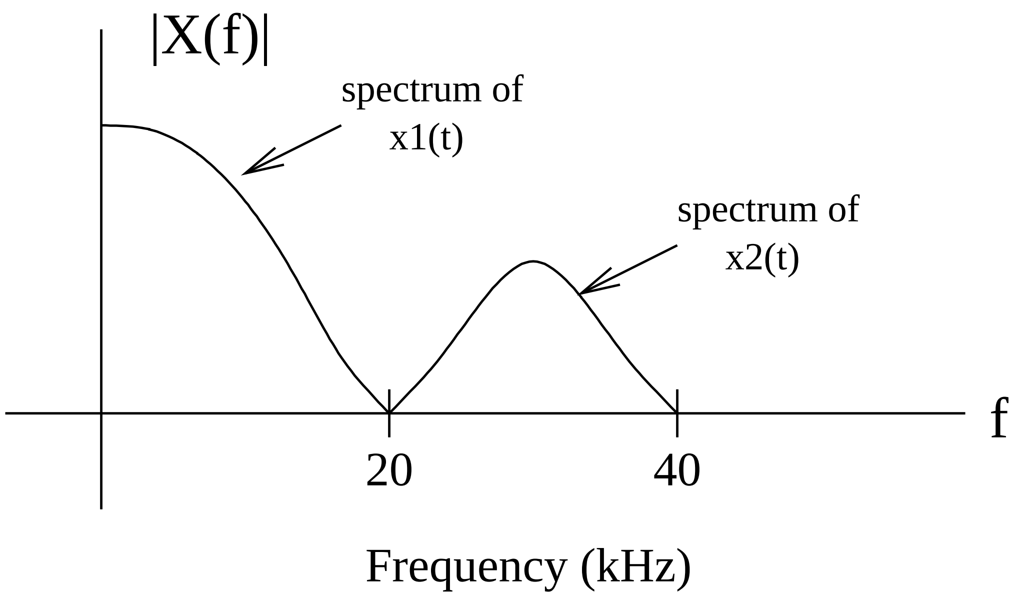

Figure 1: Spectrum of the signal $x(t)$

Suppose that we wish to suppress the signal $x_2(t)$ by lowpass-filtering the sequence $x[n]$. Assume that the allowable amplitude distortion (introduced by the DT processing) on the spectrum $X_1(f)$ is $\pm 2\%$ over the frequency range $0 \le |f| \le 15$ kHz. Above 20 kHz the filter must have an attenuation of at least 40 dB. The objective of this project is to explore several ways of designing a discrete-time lowpass filter to meet these specifications. Parts (a)-(f) below lead you through the design process for three different filters. For each design your writeup should include:

- Name of the Matlab function used in the design;
- Input design parameters;
- Plots of the magnitude and phase characteristics of the filter;
- Brief discussion of actual specifications (*e.g.*, passband ripple) achieved by the design.

(a) Determine and sketch the specifications for the discrete-time filter.

(b) Use the Parks-McClellan algorithm to design the minimum-order linear-phase FIR filter that meets the specifications.

(c) For the order M obtained in part (b), design an FIR digital lowpass filter using the window technique and the Hamming window. Does this filter meet the specifications? If not, find the smallest order Hamming window design that has the desired characteristics.

(d) Use the Kaiser window method to design a filter that meets the specifications.

(e) Design the minimum order elliptic filter that meets the given amplitude specifications. Compare the frequency response of the elliptic filter with that of the FIR filter in part (a).

(f) Compare the complexity of implementing the FIR filters with the IIR filters obtained in part (e). Assume that the FIR filters are implemented in the direct form and that the elliptic filter is implemented as a cascade of first- and second-order sections. Use storage requirements and the number of multiplications per output point in the comparison of complexity.

(g) Briefly discuss the relative merits of each of your designs. Your writeup for this part be a short memo, written for a boss who needs the information but does not have much time to read it.

Optional

If you have extra time, you may want to experiment with the Butterworth and Chebychev IIR designs. If you do, include them in your comparisons in part (g).

Hints:

- Before starting the FIR design problem, you may want to determine approximate required order using Eq. 7.104 on page 502 of Oppenheim and Schafer's text.
- You will probably want to make use of the following Matlab commands:

`remez`	Parks-McClellan optimal equiripple FIR filter design
`fir1`	Filter design using the window method
`hamming`	Hamming window
`ellip`	Elliptic (Cauer) digital/analog filter design
`ellipord`	Elliptic filter order selection
`butter`	Butterworth digital/analog filter design
`buttord`	Butterworth filter order selection
`cheby1, cheby2`	Chebyshev digital/analogy filter design
`cheby1ord, cheby2ord`	Chebyshev filter order selection
`freqz`	Frequency response
`type`	Types out a listing of an M-file

Part II: Filtering a Periodic Signal

In this part you will design a highpass filter so that you can model the periodic signal filtering problem (problem 2) on Exam 2. The purpose is to get more practice with filter design and to reinforce important Fourier series and filtering concepts.

(a) On Exam 2 you computed the discrete Fourier series representation for the signal with period $N = 4$ shown in Figure 2 below. Use the Fourier series synthesis equation to generate a vector containing $x_1[n]$ for $n = 0$ to $n = 200$. If you think carefully about it, you should be able to generate the signal using a matrix multiply (rather than using a `for` loop to implement the summation). Plot your results using the `stem` command and verify that your vector is equal to 201 samples of the periodic sequence. Note: your $x_1[n]$ vector will probably contain a *very* small imaginary part. You may ignore this by setting `x1=real(x1)`.

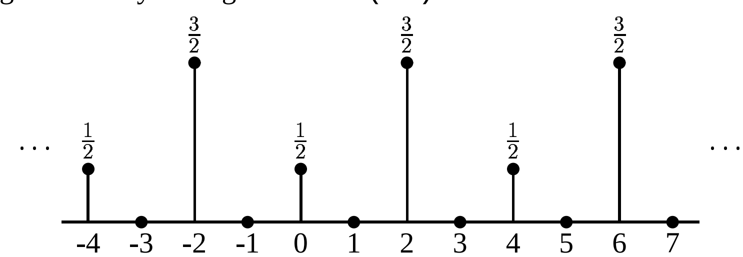

Figure 2: Periodic signal $x_1[n]$

(b) Design an FIR highpass filter using the window design method (Matlab's `fir1` program) using the ideal frequency response shown in Figure 3 below. The choice of the window and the length is up to you. Please comment on how you make these choices. (Note: you may want to try the filtering in part c with several different filter designs before deciding on one.)

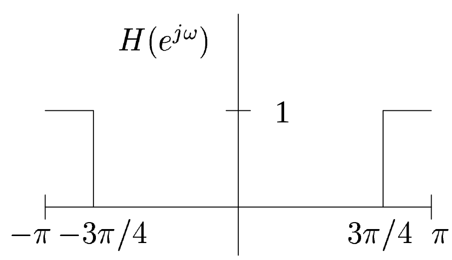

Figure 3: Frequency response of system for Problem 2d

(c) Compute $y[n]$, the output of the highpass filter, given $x_1[n]$ as the input. How does your result differ from the analytical solution for in problem 2, part d?

(d) **Extra credit** Design an IIR highpass filter to meet the specifications given in part a. Use it to filter the periodic signal and compare/contrast the results with those for the FIR filter.