EE 451

Homework #11

Due November 5, 2001

1. Problem 7.3.

2. Design a discrete-time Butterworth lowpass filter for a sampled continuous-time system with the following characteristics:

$$F_P = 2\ kHz \quad R_P = 0.5 \text{ dB}$$
$$F_S = 8\ kHz \quad R_S = 30 \text{ dB}$$

   The sampling frequency is 48 kHz.

   (a) Find the specifications of the discrete-time filter.

   (b) Pre-warp the discrete-time frequencies to find the specifications of the continuous-time low-pass prototype.

   (c) Design the continuous-time low-pass prototype Butterworth filter.

   (d) Use the bilinear transformation to transform the CT low-pass filter to a DT low-pass filter.

   (e) Use MATLAB's `zplane()` function to plot the poles and zeros of your DT filter. Explain why this pole-zero diagram corresponds to a low-pass filter.

   (f) Use MATLAB's `freqz()` function to find the frequency response of the filter. Plot this frequency response, and verify that it meets the specifications.

3. Design a filter to meet the specifications of Problem 2 using MATLAB's `buttord()` and `butter()` functions. Show

4. Design a discrete-time Butterworth bandpass filter for a sampled continuous-time system with the following characteristics:

$$F_{pl} = 8 \text{ kHz} \quad F_{ph} = 12 \text{ kHz} \quad R_P = 0.5 \text{ dB}$$
$$F_{sl} = 2 \text{ kHz} \quad F_{sh} = 16 \text{ kHz} \quad R_S = 30 \text{ dB}$$

   The sampling frequency is 48 kHz.

   (a) Find the specifications of the discrete-time filter.

   (b) Pre-warp the discrete-time frequencies to find the specifications of the continuous-time bandpass prototype.

   (c) Use the frequency-transformation equations from Chapter 5 to convert the CT bandpass filter specifications to CT low-pass specifications.

   (d) Design the continuous-time low-pass prototype Butterworth filter.

   (e) Use the frequency-transformation equations from Chapter 5 to convert the CT lowpass filter to a CT bandpass filter.

   (f) Use the bilinear transformation to transform the CT bandpass filter to a DT band-pass filter.

(g) Use MATLAB's `zplane()` function to plot the poles and zeros of your DT filter. Explain why this pole-zero diagram corresponds to a bandpass filter.

(h) Use MATLAB's `freqz()` function to find the frequency response of the filter. Plot this frequency response, and verify that it meets the specifications.

5. Design a filter to meet the specifications of Problem 4 using MATLAB's `buttord()` and `butter()` functions.