DE-43(EE) Syndicate:A,C

Semester Project EE-232

Design and Implementation of a 5-band Graphic Equalizer

Part I

The goal of this project is to design a 10-band graphic equalizer and then to implement it - first employing Simulink to check the design and then designing a GUI.

Matlab can be employed to design the required filters and then Simulink can be used to implement the graphic equalizer in real time. Most commercial equalizers use either 1/3 octave or 2/3 octave bandpass filters but to keep this from becoming too large we will employ one octave bandpass filters. Following are the design specifications for the equalizer:

1. Employing Matlab, design 5 different bandpass filters with center frequencies of 63 Hz, 250 Hz, 1000 Hz, 4000 Hz, and 16000 Hz. These center frequencies correspond to the ISO (International Standards Organization) standard for graphic equalizer center frequencies.

2. The bandwidth of each filter is the frequency difference $\Delta f = f_2 - f_1$, where f_1 and f_2 correspond to the frequencies where the gain is 3 dB less than the maximum gain at the center frequency. It also is necessary to choose f_1 and f_2 such that the center frequency, f_c, is equal to the geometric mean of f_1 and f_2, i.e. $f_c = (f_1 f_2)^{1/2}$. We also have to choose the bandwidth of each filter so that we get a flat frequency response when all filter gains are equal and added together.
3. You can use Butterworth filters; however you are free to choose the order of the filters. The Matlab help file for the Butterworth filter is the following:

   [B,A] = butter(N,Wn) designs an Nth order lowpass digital Butterworth filter and returns the filter coefficients in length N+1 vectors B (numerator) and A (denominator). The coefficients are listed in descending powers of z. The cutoff frequency Wn must be 0.0 < Wn < 1.0, with 1.0 corresponding to half the sample rate.

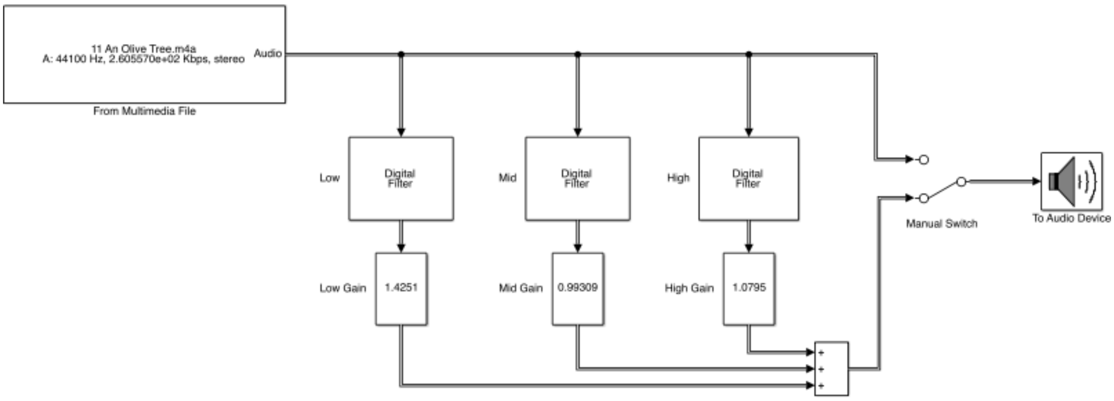
   If Wn is a two-element vector, Wn = [W1 W2], butter returns an order 2N bandpass filter with passband W1 < W < W2.

   [B,A] = butter(N,Wn,'high') designs a highpass filter. [B,A] = butter(N,Wn,'low') designs a lowpass filter.
   [B,A] = butter(N,Wn,'stop') is a bandstop filter if Wn = [W1 W2].

4. Write a Matlab m-file to compute the set of filter coefficients and plot the combination (sum) of all filter frequency responses. Note that you can use the 'freqz' command to easily find the frequency response of a filter defined by the filter coefficient arrays B and A. Your goal is to achieve as flat of a frequency response as you can when all the frequency response of all filters are added ± 1 dB is a good goal. Remember that the center frequency of each filter must be fixed to one of the five values given above and the upper and lower cutoff frequencies f_2 & f_1 must satisfy $f_c = (f_1 f_2)^{1/2}$. Your goal is to find the

Δf value for each filter that achieves a flat frequency response when all filters are combined with equal weights. [Hint: the filters should all be constant Q, where $Q = f_c / (f_2 - f_1)$, so once you find the right value for Q all filters should have the same Q.]

The simplified 3-band graphic equalizer is shown below, where the filters are in parallel and each one is followed by a gain (using the Matlab slider gain block). Your mixer will have 5 filters in parallel.



We would like to be able to adjust the gain of each band by ± 12 dB. Remember that 6dB corresponds to approximately a factor of 2x, so 12 dB is about 4x. So +12 dB is like multiplying by 4 and -12 dB is like multiplying by ¼.Use these values as the limits for the slider gain blocks.

One final note: The "From Multimedia File" and "To Audio Device" blocks can be found in the DSP Toolbox.

Part II

Design a GUI with the following provisions

1. A 'load' button that can load an audio file of your choice (Suggestion: choose a file with large frequency range e.g a symphony).
2. Display for the input and output signals.
3. Display for the input and output spectrum.
4. Adjustable gain sliders for the filters.
5. A 'play' button that can playback the output file.

P.S. You are welcome to get help from the internet, for example https://www.mathworks.com/matlabcentral/fileexchange/23982-digital-audio-equalizer