WHOLLY COMPUTER BASED FILTER DESIGN

Fatih V. Çelebi, Tuğba Önder
Ankara Univ. Bilgisayar Müh. Bölümü, ANKARA
e-mail: fcelebi@eng.ankara.edu.tr

ABSTRACT

This article mentions about our work in C#. The goal of our work is to develop a program, based in Visual C#.NET, that simplifies the process of designing passive filters. This program is object oriented using C# capabilities for doing so. The program is able to design low-pass, high-pass, band-pass, and band-reject filters. The approximation magnitude responses can be Butterworth (maximally flat magnitude) or Chebyshev (equal-ripple).

The output is a set of data that includes the frequency response of filters and attenuation at different frequencies. Also the possible circuit schema is drawn by the program. Another output from this program is a set of magnitude plot that are readily calculated during the design. The network realizations are passive realizations. A graphical-user-interface (GUI) is developed for filter design by using Visual C#. The main menu and all the other required windows are programmed directly in Visual C#.NET. This will create a friendly environment for the user, because the program is able to find the attenuation and the frequency response graphics of the desired filter by entering the required specifications such as the maximum allowed attenuation in the pass-band, the minimum allowed stop-band attenuation, the cut-off frequencies for the pass-band and stop-band regions, and/or the filter order.

1. INTRODUCTION

A passive filter is a kind of electronic filter that is made only from passive elements -- in contrast to an active filter, it does not require an external power source. Since most filters are linear, in most cases, passive filters are composed of just the basic linear elements -- resistors, capacitors, inductors, and transformers. A passive filter has several advantages over an active filter. Some of them are guaranteed stability, better scaling to large signals, no power consumption etc.

Designing a filter is the action of determining the order of the filter and constituting the circuit elements in a possible way. Butterworth filter design can be done using attenuation at some frequencies and the cut-off frequencies whereas Chebyshev filter design requires the order of the filter and ripple values.

The design is handled in an object oriented manner. Every type of the filter in the real world is thought to be a separate class having attributes and methods to solve the required parameters. Also inheritance and polymorphism are to achieve the similar operations between filter types.

All the mathematical operations involved to solve the filter equations are done using Visual C#.NET. The values computed by filter design equations are plotted to show the frequency responses of the filters.

Lastly the powerful graphics capabilities of C#'s Graphics class, including shapes, brushes, pens, textures, gradients, line styles and general paths are used to draw the circuit elements of the filter.

2. IMPLEMENTATION

Filter design is achieved by a series of functions implemented in an object oriented way. ‘Passive Filter’ is our super class. Its sub classes ‘Butterworth’ and ‘Chebyshev’ are the parent classes of low-pass, high-pass. C# Math library is used to handle the mathematical operations.

Passive filter object has all the characteristics of the passive filters by its attributes. It has also abstract methods which are implemented in subclasses necessarily.

The Butterworth approximation to an ideal low-pass filter is based on the assumption that a flat response at zero frequency is more important than the response at other frequencies. A normalized transfer function is an all-pole type having roots which all fall on a unit circle. The attenuation is 3 dB at 1 rad/s.

The design steps of Butterworth high pass and low pass filters:

Calculating the steepness factor using equation (1) and (2)

\[ A_2 = \frac{f_2}{f_s} \]  
\[ A_3 = \frac{f_3}{f_s} \]  

Here \( f_s \) is the frequency having the minimum required stop band attenuation and \( f_3 \) is the limiting frequency or cut-off of the passband, usually the 3-dB point. The normalized curves are compared with \( A_2 \), and a design is selected that meets or exceeds the requirement. The
design is often frequency scaled so that the selected passband limit of the normalized design occurs at \( f_c \).

- Evaluating the order of the filter through the attenuation prompted by the user.
- Making out a series of points by calculating the attenuation at different frequencies and then drawing the frequency response.
- Calculating the capacitor and inductor values by using the equations (3) and (4)

\[
\sin\left(\frac{2K-\Omega}{2\pi}\right) + j \cos\left(\frac{2K-\Omega}{2\pi}\right), \quad K = 1, 2, 3, \ldots, m
\]  

(3)

\[
L_K \text{ or } C_K = 2 \sin\left(\frac{2K-\Omega}{2\pi}\right), \quad K = 1, 2, 3, \ldots, m
\]  

(4)

where \( \Omega \) is in radians.

The steps in Butterworth design are also used in Chebyshev but the user is prompted to enter the passband ripple and order of the filter. The attenuation of the filter is evaluated at different points.

The design steps of Chebyshev high pass and low pass filters are:

- Calculating the steepness factor using equation (1) and (2).
- Making out a series of points by calculating the attenuation through equations (5) through (10) at different frequencies and then drawing the frequency response.

\[
A = \frac{\varepsilon}{\pi} \sinh^{-1}\left(\frac{1}{\varepsilon}\right)
\]  

(5)

where

\[
\varepsilon = \sqrt{10^{R_{db}/10} - 1}
\]  

(6)

and \( R_{db} \) is the ripple in decibels. Figure 2 compares the frequency response of an Butterworth normalized low pass filter and the Chebyshev filter generated by applying Equations (7) and (8).

\[
k_r = \sinh A
\]  

(7)

\[
k_l = \cosh A
\]  

(8)

Figure 2: A comparison of Butterworth and Chebyshev low pass filters

The Chebyshev filter response has also been normalized so that the attenuation is 3 dB at 1 rad/s. The actual 3-dB bandwidth of a Chebyshev filter computed using equations (7) and (8) is \( \cosh A_1 \), where \( A_1 \) is given by

\[
A_1 = \frac{\varepsilon}{\pi} \cosh^{-1}\left(\frac{1}{\varepsilon}\right)
\]  

(9)

The attenuation of Chebyshev filters can be expressed as

\[
A_{db} = 10 \log \left(1 + \varepsilon^2 \cosh^{-2}(\Omega)\right)
\]  

(10)

Where \( \cosh(\Omega) \) is a Chebyshev polynomial whose magnitude oscillates between -1 and 1 for \( \Omega \leq 1 \). Table 1 lists the Chebyshev polynomials up to order \( n= 10 \). At Chebyshev polynomials have a value of unity, so the attenuation defined by Equation (5) would be equal to the ripple. The 3-dB cut-off is slightly above \( \Omega=1 \) is equal to \( \cosh A_1 \). In order to normalize the response equation (10) is computed by using the Table 3. (Because 3 dB of attenuation occurs at the \( \Omega=1 \))

Table 1: \( \Omega \) values for Chebyshev Filter

<table>
<thead>
<tr>
<th>Filter type</th>
<th>( \Omega )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low-pass</td>
<td>( (\cosh A_1) \frac{\omega_c}{\omega_p} )</td>
</tr>
<tr>
<td>High-pass</td>
<td>( (\cosh A_1) \frac{\omega_c}{\omega_s} )</td>
</tr>
<tr>
<td>Band-pass</td>
<td>( (\cosh A_1) \frac{BW_p}{BW_s} )</td>
</tr>
<tr>
<td>Band-Reject</td>
<td>( (\cosh A_1) \frac{BW_p}{BW_s} )</td>
</tr>
</tbody>
</table>
Table 2: $\Omega$ values for Chebyshev Filter

<table>
<thead>
<tr>
<th>Filter type</th>
<th>$\Omega$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low-pass</td>
<td>$(\cosh A_1) \frac{a^2}{a_f}$</td>
</tr>
<tr>
<td>High-pass</td>
<td>$(\tanh A_2) \frac{a^2}{a_n}$</td>
</tr>
<tr>
<td>Band-pass</td>
<td>$(\cosh A_2) \frac{BW_{dB}}{BW_{dB}}$</td>
</tr>
<tr>
<td>Band-Reject</td>
<td>$(\cosh A_2) \frac{BW_{dB}}{BW_{dB}}$</td>
</tr>
</tbody>
</table>

Now we will see the equations which are used to calculate the capacitor and inductor values by using following equations:

$$G_1 = \frac{2A_1 \cosh A}{Y}$$  \hspace{1cm}  (11)

$$G_k = \frac{4A_k - A_{k-1} \cosh^2 A}{B_{k-2} \Omega_{k-2}} \hspace{0.5cm} k = 2,3,4,...,n$$  \hspace{1cm}  (12)

where

$$Y = \sinh \frac{\beta}{2}$$  \hspace{1cm}  (13)

$$\beta = \ln \left( \frac{\cosh R_{dB}}{\Omega_{dB}} \right)$$  \hspace{1cm}  (14)

$$A_k = \sin \left( \frac{(2k-1)\pi}{2} \right) \hspace{0.5cm} k = 1,2,3,4,...,n$$  \hspace{1cm}  (15)

$$B_k = Y^2 + \sin^2 \left( \frac{\beta}{2} \right) \hspace{0.5cm} k = 1,2,3,4,...,n$$  \hspace{1cm}  (16)

Coefficients $G_1$ and $G_k$ are the element values.

‘Band Pass’ class is the super class of the Narrowband and Wide Band Classes.

Wideband object:

A wide band pass filter has two parts; low pass filter and high pass filter. In object oriented manner the wideband pass object is thought to have two instances of Low pass and high pass filters. Then the two objects are composed to form the Wide band Pass filter.

The application fails to give good results when the constraints are not realized. At the same time, high order passive filters can be failed by this application. Designing high order filters can be future work.

Narrowband object:

In order to design a narrow band pass filter, the following sequence of steps is involved:

- The given band pass filter requirement is converted into a normalized low-pass specification.
- A satisfactory low-pass filter is selected from the normalized frequency-response curves.
- Normalized low-pass parameters are transformed into the required band pass filter.

The bandreject object is the parent of ‘WidebandReject’ and ‘NarrowBandReject’ objects. The design steps in band pass filters are adapted to the Band Reject Filters.

Figure 1 shows the graphical user interface of the application. The radiobuttons at the top left provides the user to select the filter type and class. The textboxes at the bottom left provide to enter the cut-off frequencies, order, attenuation etc. according to the chosen filter. Figure 2 shows a screenshot of the running program. The user had chosen Low pass-Butterworth filter and entered the cut-off frequencies and attenuation. By clicking the ‘Design Solution’ button, the filter order is determined and shown in the label at the left bottom. Also user has gotten the frequency plot by clicking on the ‘PLOT’ button and the plotted points in the list boxes at the top right. At last user is able to see the circuit schema by clicking on the “DRAW” button. Figure 3 and Figure 4 are the examples of other filter types.

3. CONCLUSION

In this project we focused on modern network theory by using families of standard transfer functions that provide optimum filter performance in some desired respect.

Object oriented approach was a good solution to the filter design. Every filter is thought to be a separate class and has its necessary calculations. Some parameters are entered by the user and the desired parameters are obtained to design the filter component. To get the desired filter parameters, C# mathematical operations are used. The solution values were so near to the real world parameters.

In Electronic Filter Design Handbook [1], some Chebyshev filters are designed by looking through the tables including orders, pass band ripples and element values. These tables can be hold in a database and we can search the database to find the desired filter. This solution can be a faster and definite solution in computer based filter designs.

The iterative process of passive filter design was illustrated through a step-by-step calculation. The design is achieved by development of software with user-selectable constraints and resulting design parameters.
Figure 1: Graphical User Interface of the application

Figure 2: Experimental Result of the ‘Low Pass Butterworth Filter’ (example)
Figure 3: Experimental Result of the ‘High pass-Butterworth Filter’ (example)

Figure 4: Experimental Result of the ‘Lowpass-Chebyshev Filter’ (example)

REFERENCES