New Proposed Algorithms for nth Order Butterworth Passive Filter
Computer-Aided Design

Haider Fakher Radhi Al-Saidy

doi: 10.4156/ijact.vol2. issue5.13

Abstract

Modeling techniques are presented which allow C++ program to be used for nth order Butterworth passive LPF design. Synthesis of passive networks (ladder networks) is implemented to design any order Butterworth passive Filter. It is shown how these models can be simply interconnected to produce a complete passive LPF design program. CAD program of nth order Butterworth filter is flexible design approach of filters’ frequencies and components. New proposed algorithms are presented to achieve the highest flexibility of design the selected order, components and frequencies.

Typical examples, for various orders Butterworth passive LPF are presented. The presented algorithms are developed for fast, simple, accurate and flexible design.

Experiments show that this method is fast and capable of generating circuits which are more economical than those generated by traditional design approaches because of the high flexibility of choosing the cheapest components.

Keywords: Programming algorithms, Passive filter design, Frequency-dependent component.

1. Introduction

In this paper a new proposed algorithms for nth order Butterworth passive LPF design using synthesis of ladder networks. It is very easy to use the program of design even without background about filter design. The design program is highly flexible design approach of filters’ orders and components that leads to the best economical selection of components.

Analog electronic filters are present in just about every piece of electronic equipment [1].

The range of signal frequencies that are allowed to pass through a filter, with little or no change to the signal level, is called the passband. The range of signal frequencies that are reduced in amplitude by an amount less than the half power of the signal level in passband is called the stopband [2].

Winder [1] has written a number of simple programs to help his work [1], his programs are like “Elsie, Passive (LC) filters only”, “PC Filter” and “Filter Solutions”.

Chen [2] has provided a lot of information in filters with significant additions in the areas of computer-aided design of passive filters.

Hou [3] has presented the computer-aided design of passive filter synthesis using Genetic algorithms.

2. Filters

Filters can be considered as frequency selective networks. A filter is required to separate an unwanted signal from a mixture of unwanted and wanted signals[4].

The filter specifications are generally given in terms of cut-off frequencies or pass-band and stop-band regions. An ideal filter should pass the wanted signals with no attenuation and provide infinite attenuation for the unwanted signals[5].

The characteristics of an ideal low-pass filter are shown in the following figure:

![Figure 1. The characteristics of ideal low-pass filter](image-url)
The filter network is required to pass all frequencies under the cut-off frequency and provide infinite attenuation on all frequencies higher than the cut-off frequency. Above ideal filter specifications cannot be achieved exactly but by increasing the order of Butterworth filter, the filter will be closer and closer to ideal filter specifications[6].

3. Butterworth Filter

This filter gives maximally flat pass band. The magnitude of transfer function of this filter [7] is:

\[
|H(s)| = \frac{1}{\sqrt{1 + \omega^2 n}}
\]

where \(n\) is the order of the filter

To calculate the poles of \(n\)th order Butterworth filter we use the following equation:

\[
p_k = -\sin \left(\frac{(2k-1)\pi}{2n}\right) + j\cos \left(\frac{(2k-1)\pi}{2n}\right)
\]

for \(k=1,2,...,n\)

The poles of \(n\)th order are complex conjugates, only if the order is odd, then there is an additional pole equal to \(s = -1\). The complex conjugate poles can be calculated using the equation [8]:

\[
p_k \cdot p^*_k = s^2 + [2\sin \left(\frac{(2k-1)\pi}{2n}\right)] s + 1
\]

for \(k=1,2,\cdots,\left\lfloor \frac{n}{2} \right\rfloor\)

Butterworth approximation is a special form of Taylor series approximation in which the approximating function is identical at \(w=0\) and maximally flat. Therefore, the approximation will be better with higher order. That means better flatting in pass-band and greater attenuation in stop-band for higher order[4, 8, 9].

4. Synthesis of Passive Networks

We can synthesis a network for a given input impedance or input admittance which are real and positive in all values of \(s\). That can be done by following ladder network [4]:

![Figure 2. Ladder passive network with input impedance \(Z_{in}\)](image)

The input impedance of Figure 2 is given by Equation (4) as shown:
5. Transfer Function Synthesis

For the following network:

Using voltage divider:

\[
\frac{V_a}{V_i} = \frac{z_1}{1 + z_1}
\]  

(5)

if \( \frac{V_o}{V_a} = T(s) \)

\[
\Rightarrow \frac{V_o}{V_i} = H(s) = \frac{V_a}{V_i} \times \frac{V_o}{V_a} = \frac{z_1 \cdot T(s)}{1 + z_1}
\]  

(6)

Now let \( H(s) = \frac{1}{p(s)} = \frac{1}{O(s) + E(s)} \)

(7)

That \( H(s) \) is the transfer function of a filter, \( O(s) \) is the odd polynomial and \( E(s) \) is the even polynomial.

\[
\Rightarrow H(s) = \frac{1/O(s)}{1+E(s)/O(s)}
\]  

(8)

That leads to \( z_1 = \frac{E(s)}{O(s)} \)

(9)

If the resulted is \( O(s) \) has higher order than \( E(s) \), then we synthesize for \( y_1 \) instead of \( z_1 \). That from Eq. 10:

\[
y_1 = \frac{O(s)}{E(s)}
\]  

(10)
To prevent any loading which changes the transfer function, the designed filters can be connected to operational amplifier with suitable gain and the output of that operational amplifier stage will be the output of our designed filter.

6. Example of Synthesizing

In this example we want to synthesize the third-order prototype LPF filter. Using Butterworth equation (3) of conjugate poles we get the poles as follows: Because the order is odd, and additional \((s+1)p_1p_1' = s^2 + s + 1\)

then the transfer function

\[
H(s) = \frac{1}{s^3 + 2s^2 + 2s + 1}
\]

The odd section \(O(s) = s^3 + 2s\)

The even section \(E(s) = 2s^2 + 1\)

\[
\Rightarrow H(s) = \frac{1/O(s)}{1 + E(s)/O(s)}
\]

\[
\Rightarrow H(s) = \frac{1/(s^3 + 2s)}{1 + (2s^2 + 1)/(s^3 + 2s)}
\]

Then \(y_1 = \frac{s^3 + 2s}{2s^3 + 1}\)

Here \(y_1\) is used instead of \(z_1\) because \(O(s)\) is higher order than \(E(s)\).

Then the procedure of synthesizing is the multiple division of Equation (4) as shown:

\[
\frac{s/2}{2s^2 + 1}\frac{s^3 + s/2}{1.5s}
\]

Then \(C_1 = 0.5F\)

\[
\frac{4s/3}{1.5s}\frac{2s^2 + 1}{s^3 + 2s^2 + 1}
\]

Then \(L_1 = (4/3)H\)
Then $C_2 = 1.5F$

From above procedure of synthesizing prototype LPF we get the ladder network:

$$C_1 = \frac{1}{2}F, L_1 = \frac{4}{3}H, C_2 = 1.5F$$

7. Realization of The Prototype Filter

The prototype filter has $R=1$ and cut-off $1$ rad./sec. The impedance scaling can be made without changing the transfer function or the cut-off frequency value by multiplying by the same factor each of $R$, $Z_L$, and $Z_C$.

Therefore the transformation is by multiplying each resistance, inductance by a factor $(k)$ and dividing the capacitance by the same factor $(k)$.

In the other hand to get frequency transformation from $1$ rad./sec. to required cut-off frequency $(f)$, we must divide each inductance and capacitance by $\left(2\pi \cdot f\right)$ [5,10].

8. Frequency Transformation

In this paper we have only discussed the design of low pass (LP) filters. There are many other types of filters which find use in signal processing. Once the low-pass prototype filter has been designed, the other filters (High-pass, band-pass, and band-stop) can be obtained by a simple frequency transformation [4,6,8].

9. The First and Second Parts of The Program Filter Design

The poles of transfer function of nth order LP filter can be calculated using Equation (3) of calculating the conjugate poles. The generation of transfer function can be implemented easily using C++ program by putting the calculated conjugate poles in array and the results of cumulative multiplication of poles in another array. Flowchart 1 and Flowchart 2 show above algorithms.

10. The Third Part of the Program Filter Design

The ladder network synthesis algorithms can be represented using C++ program (using the multiple division of Equation (4)) by making the denominator as an array, the numerator as another array, and the results of division in a third array. Flowchart 3 shows the algorithms of design.
\[
a = 2 \sin \left( \frac{2k-1}{2}\pi \right)
\]

\[
i = 1
\]

\[
c[i] = b[i], d[i] = b[i]a
\]

Is \(i \leq 2k + 1\)?

\[
 j = 2
\]

\[
\]

Is \(j \leq 2k + 1\)?

**Flowchart 1.** First part of the Program Filter Design

**Flowchart 2.** Second Part of Program Filter Design
11. The Results of Program Filter Design

With cut-off frequency 1 radian per second and $R=1\Omega$ (prototype) the results for the third order Butterworth filter are:
$C_1=0.5 \, \text{F}$, $L=1.33333 \, \text{H}$, $C_2=1.5 \, \text{F}$, as shown in Figure 5:

![Flowchart 3. Third part of Program Filter Design](image)

**Figure 5.** Prototype third order LPF

If we choose realization for $R=100\Omega$ and $f_c=100\, \text{kHz}$, we get:
$L=0.212 \, \text{mH}$, $C_1=7.96 \, \text{nF}$, $C_2=23.9 \, \text{nF}$, as shown in Figure 6:
The results for the sixth order Butterworth filter with \( R = 100 \Omega \), \( fc = 100 \text{ kHz} \), are:
\[ L_1 = 41.2 \ \mu \text{H}, \ L_2 = 0.191 \ \text{mH}, \ L_3 = 0.247 \ \text{mH} \]
\[ C_1 = 12 \ \text{nF}, \ C_2 = 24.7 \ \text{nF}, \ C_3 = 24.7 \ \text{nF} \], as shown in Figure 7:

The results for tenth order Butterworth filter with \( R = 100 \), \( fc = 100 \text{ kHz} \), are:
\[ L_1 = 24.9 \ \mu \text{H}, \ L_2 = 0.121 \ \text{mH}, \ L_3 = 0.2057 \ \text{mH}, \ L_4 = 0.2686 \ \text{mH}, \ L_5 = 0.2954 \ \text{mH} \]
\[ C_1 = 7.41 \ \text{nF}, \ C_2 = 16.57 \ \text{nF}, \ C_3 = 24 \ \text{nF}, \ C_4 = 28.86 \ \text{nF}, \ C_5 = 24.9 \ \text{nF} \] as shown in Figure 8:

The results for the third order Butterworth filter with \( R = 5 \Omega \), \( fc = 100 \text{ kHz} \), are:
\[ L = 10.616 \ \mu \text{H}, \ C_1 = 159.2 \ \text{nF}, \ C_2 = 477.7 \ \text{nF} \], as shown in Figure 9:

The results for the sixth order Butterworth filter with \( R = 5 \Omega \), \( fc = 100 \text{ kHz} \), are:
\[ L_1 = 2.06 \ \mu \text{H}, \ L_2 = 9.567 \ \mu \text{H}, \ L_3 = 14 \ \mu \text{H} \]
\[ C_1 = 241.36 \ \text{nF}, \ C_2 = 494.56 \ \text{nF}, \ C_3 = 494.56 \ \text{nF} \], as shown in Figure 10:
The results for tenth order Butterworth filter with $R=5$ Ω, $f_c=100$ kHz, are:

$L_1=1.245$ $\mu$H, $L_2=6.07$ $\mu$H, $L_3=10.285$ $\mu$H,
$L_4=13.43$ $\mu$H, $L_5=14.77$ $\mu$H,
$C_1=148.2$ nF, $C_2=331.4$ nF, $C_3=480.9$ nF,
$C_4=577.1$ nF, $C_5=498.2$ nF as shown in Figure 11:

Using Electronics Workbench simulation program, above results are proved which gives the modeling technique and our CAD program the insurance of giving accurate results.

12. Conclusions

It is shown from the results that we can get fast and exact designs using CAD of nth order Butterworth filter for practical implementations with very high flexibility. It is easy to scale the results using CAD and we can easily get the required and available passive components. The design algorithms are optimized to synthesize any order without order limits.

CAD program of nth order Butterworth passive filter is very flexible design approach of filters' frequencies and components. Compared with other design methods, CAD of nth order passive filter is considered to be more economical filter design because of the high flexibility of choosing the cheapest components.

The procedure of design prototype LPF can easily extended using CAD to design any required HPF, BPF or BSF. Program size is very suitable and can transferred and carried easily by communication circuits designers.

Design results were verified using Electronics Workbench9 simulation program.

13. References