A GENERAL METHOD AND FORTRAN PROGRAM FOR THE DESIGN OF RECURSIVE DIGITAL FILTERS

Joseph J. Wolcin

Naval Underwater Systems Center
New London, Connecticut

25 September 1973
A general method for the design of digital filters is presented, and a FORTRAN program that follows the steps of the method is described. The computer program can design any low-pass, high-pass, bandpass, or band-reject digital filter by using either Butterworth, Chebyshev, or elliptic approximations to the ideal rectangular type frequency response. The user need only provide the
20. program with a set of structured specifications, and as output he obtains the transfer function of the minimum order filter that meets his requirements. For convenience, the transfer function is expressed in a form that allows immediate implementation in a cascade type realization.
PREFACE

This report was prepared under a local indirect funding project (NUSC Job Order 300200, Cost Center SPA400).

The Technical Reviewer for this report was Dr. Albert H. Nuttall (Code TC).

REVIEWED AND APPROVED: 25 September 1973

R. M. Dunlap
Director for Plans and Analysis

Inquiries concerning this report may be addressed to the author, New London Laboratory, Naval Underwater Systems Center, New London, Connecticut 06320
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF ILLUSTRATIONS</td>
<td>iii</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>USE OF THE COMPUTER PROGRAM</td>
<td>2</td>
</tr>
<tr>
<td>THEORY AND PROGRAMMING METHOD</td>
<td>11</td>
</tr>
<tr>
<td>Preliminaries</td>
<td>11</td>
</tr>
<tr>
<td>Analog Critical Frequencies</td>
<td>14</td>
</tr>
<tr>
<td>Determination of Minimum Order</td>
<td>15</td>
</tr>
<tr>
<td>Determination of $s$-Plane Pole-Zero Pattern</td>
<td>18</td>
</tr>
<tr>
<td>Calculation of Pole-Zero Pattern of the Desired Digital Filter</td>
<td>19</td>
</tr>
<tr>
<td>Calculation of the Constant Multiplier</td>
<td>21</td>
</tr>
<tr>
<td>Manipulation into Cascade Form</td>
<td>23</td>
</tr>
<tr>
<td>The Magnitude and Phase of the Frequency Response</td>
<td>25</td>
</tr>
<tr>
<td>The Unit Sample Response</td>
<td>25</td>
</tr>
<tr>
<td>SUMMARY</td>
<td>27</td>
</tr>
<tr>
<td>APPENDIX - LISTING OF THE COMPUTER PROGRAM</td>
<td>29</td>
</tr>
</tbody>
</table>
# LIST OF ILLUSTRATIONS

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Illustrative Frequency Characteristics (Elliptic)</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Sample Printout</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>Frequency Characteristic (Magnitude)</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>Frequency Characteristic (Phase)</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>Unit Sample Response</td>
<td>10</td>
</tr>
</tbody>
</table>
A GENERAL METHOD AND FORTRAN PROGRAM
FOR THE DESIGN OF RECURSIVE DIGITAL FILTERS

INTRODUCTION

A very tedious problem often confronting engineers is the design of a frequency selective digital filter that meets a desired set of specifications. The task is especially laborious and the theory quite complex when the filter has to be designed to satisfy very demanding requirements. Considerations such as these have often caused the designer to settle for a filter with characteristics further from the ideal rectangular type response than is desired. Described herein is a general method for the design of digital filters and a versatile FORTRAN program (listed in the appendix) that will carry out the design of any low-pass, high-pass, bandpass, or band-reject digital filter in any of the standard forms (i.e., Butterworth, Chebyshev, or elliptic). The user need only provide the program with a set of structured specifications, and as output he obtains the transfer function of the minimum order filter that meets his requirements. For convenience, the transfer function is expressed in a form that allows immediate implementation in a cascade type realization. The user also has the option of obtaining plots of the frequency response (both magnitude and phase) and the unit sample response of the filter.

Complicated design problems will no longer demand so much time and effort, as has especially been the case with elliptic filters. Moreover, it is suspected that elliptic filters may be preferred now for many applications since they normally require a lower order filter than that required by the currently more common Butterworth and Chebyshev filters. The advantage lies in the fact that a lower order implies fewer computations in the recursive scheme used in implementing the filter. The principal argument against elliptic filters in the past has been the time consuming and complicated design process.

In this report instructions for the use of the computer program are discussed first and then its use by example is illustrated. Discussion of the underlying theory is deferred to the latter part of the report since its comprehension is not necessary in order to successfully use the program and the resulting filter.
USE OF THE COMPUTER PROGRAM

One data card, which provides the program with the desired set of specifications, should be used for each filter to be designed. The structure of the data card is shown in table 1. Also, figure 1, which illustrates how the various input parameters should be interpreted on plots of typical frequency curves, should be helpful to the user. More detailed descriptions of the input parameters and other special notes now follow.

Table 1. Data Card Structure

<table>
<thead>
<tr>
<th>Input Variable</th>
<th>Format</th>
<th>Columns</th>
<th>Brief Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRATE</td>
<td>F8.0</td>
<td>1-8</td>
<td>Sampling rate</td>
</tr>
<tr>
<td>FLOW</td>
<td>F8.0</td>
<td>9-16</td>
<td>Lower cutoff frequency</td>
</tr>
<tr>
<td>FCHIGH</td>
<td>F8.0</td>
<td>17-24</td>
<td>Higher cutoff frequency</td>
</tr>
<tr>
<td>RIPPLE</td>
<td>F6.0</td>
<td>25-30</td>
<td>Passband ripple</td>
</tr>
<tr>
<td>FSLOW</td>
<td>F8.0</td>
<td>31-38</td>
<td>Lower stopband boundary</td>
</tr>
<tr>
<td>FSHIGH</td>
<td>F8.0</td>
<td>39-46</td>
<td>Higher stopband boundary</td>
</tr>
<tr>
<td>STLVL</td>
<td>F4.0</td>
<td>47-50</td>
<td>Minimum stopband attenuation</td>
</tr>
<tr>
<td>IKIND</td>
<td>I1</td>
<td>51</td>
<td>Kind of filter desired (i.e., Chebyshev, Butterworth, elliptic)</td>
</tr>
<tr>
<td>ITYPE</td>
<td>I1</td>
<td>52</td>
<td>Type of filter desired (e.g., low-pass, or high-pass,)</td>
</tr>
<tr>
<td>NPOLES</td>
<td>I2</td>
<td>53-54</td>
<td>Order desired; set equal to zero for calculation of order</td>
</tr>
<tr>
<td>IPLOT</td>
<td>I1</td>
<td>56</td>
<td>Set ≠ 0 for plot of magnitude of frequency response</td>
</tr>
<tr>
<td>NPTS</td>
<td>I4</td>
<td>57-60</td>
<td>Number of points in unit sample response</td>
</tr>
<tr>
<td>FLOW</td>
<td>F10.0</td>
<td>61-70</td>
<td>Lowest frequency in plots</td>
</tr>
<tr>
<td>FHIGH</td>
<td>F10.0</td>
<td>71-80</td>
<td>Highest frequency in plots</td>
</tr>
</tbody>
</table>
Figure 1. Illustrative Frequency Characteristics (Elliptic)
The first input parameter, SRATE, is the sampling rate and is normally the number of data samples per second. However, any time unit can be used, and in fact, one might choose a fictitious value here with the intention of constructing a filter whose cutoff is a certain percentage of the Nyquist rate. For example, a low-pass filter with a cutoff that is one-quarter of Nyquist could be obtained by choosing SRATE to be 200 Hz and the cutoff to be 25 Hz. Note that this same filter, if applied to data sampled at some other rate, would no longer have cutoff at 25 Hz, but its cutoff would shift to one-quarter of the new Nyquist rate.

The quantities FCLOW and FCHIGH are the cutoff frequencies of the filter, and must be given in the same units as SRATE. Note that if a low-pass or high-pass filter is desired, there is only one cutoff frequency and this value should be read into FCLOW.

The quantity RIPPLE is the maximum passband ripple (in decibels) to be allowed. In the case of a Butterworth filter, where the response is monotonic, RIPPLE is not used. Note also that RIPPLE should always be a positive quantity.

The parameters FSLOW and FSHIGH are similar to FCLOW and FCHIGH, but they mark the boundaries between stop-band and transition band. Here too the quantity FSHIGH is not used in the low-pass or high-pass case.

The next specification, STPLVL, is the minimum stopband attenuation (in decibels) the user will tolerate. This should also always be a positive quantity.

The value 1, 2, or 3 of the variable IKIND determines which kind of filter will be designed, that is, Chebyahev, Butterworth, or elliptic, respectively.

Similarly, the value 1, 2, 3, or 4 of ITYPE determines the shape of the filter that will be designed, that is, low-pass, high-pass, bandpass, or band-reject, respectively.
The value of NPOLES should normally be read in as zero, and the correct order will be calculated. However, the user may (except in the elliptic case) choose a positive integer here if he desires a filter of some specific order. This integer should be the order of the basic low-pass structure desired before transformation. In converting a low-pass filter to either a bandpass or band-reject filter, the method doubles the number of poles; thus, to get a band-reject filter of order 18, NPOLES should be set equal to 9. If NPOLES is given as nonzero, the quantities FSLOW, FSHIGH, and STOLVL are not used since the given order determines these quantities uniquely. NPOLES is restricted to be less than or equal to 20, since it is anticipated that filters of these orders can meet any reasonable set of requirements. However, if necessary, one could merely increase the array dimensions used in the program in order to obtain filters of higher order.

The remaining input parameters control the plots produced by the program. If the user does not have access to the Stromberg Carlson 4060 Integrated Graphics System, he should remove the corresponding code or replace it by a code that is compatible with his plotter.

If for some reason the user does not want any of the three available plots, he should set the parameter IPLOT to zero. The number of points desired in the unit sample response should be read into NPTS. This quantity must not be greater than 1000 and has a default value of 100 if the corresponding field in the data card is left blank. The values given to the quantities FLOW and FHIGH merely restrict the plots of magnitude and phase to the range between these two frequencies. All information can be seen by plotting from zero to Nyquist, since the curve is periodic with period SRATE and is symmetric about Nyquist. However, one can expand a small frequency range for closer examination.

If more than one filter is desired, a similar data card for each should be used and a blank card should be placed last.

The following typical example illustrates the points just discussed. Suppose that an elliptic bandpass digital filter with cutoffs at 2000 Hz and 3000 Hz is desired. Assume that the sampling rate is 10,000 Hz, that the filter is to be at least 30 dB down by 1800 Hz and 3200 Hz, and that the tolerable ripple in the passband is 0.5 dB. The following is a list of the proper input parameters for this problem:
Figures 2 through 5 show the printed and plotted outputs for the example in question. Note that the range of the magnitude plot is from 0 dB to -50 dB, which can easily be increased if desired.
INPUT SPECIFICATIONS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNATE</td>
<td>10000.00</td>
</tr>
<tr>
<td>FCHW</td>
<td>2000.00</td>
</tr>
<tr>
<td>FCHGH</td>
<td>3000.00</td>
</tr>
<tr>
<td>RIPPLE</td>
<td>.500</td>
</tr>
<tr>
<td>FSLL</td>
<td>1800.00</td>
</tr>
<tr>
<td>FS耦</td>
<td>3000.00</td>
</tr>
<tr>
<td>STLVL</td>
<td>30.00</td>
</tr>
<tr>
<td>IKIND</td>
<td>3</td>
</tr>
<tr>
<td>IYPL</td>
<td>3</td>
</tr>
<tr>
<td>NPOLES</td>
<td>0</td>
</tr>
</tbody>
</table>

The minimum order filter which meets these specifications is order 8.

The transfer function \( H(z) \) of the desired filter can be expressed in the following form:

\[
H(z) = \sum_{i=0}^{N} \left( \frac{a_i z^{-i}}{1 + \sum_{j=0}^{M} b_j z^{-j}} \right)
\]

Where the coefficients are given by:

\[
\begin{align*}
&c_{-1} = 0.31310351423646001 \\
&a(1) = 0.14079198291840001 \\
&a(2) = 0.10000000000000001 \\
&a(3) = 0.10000000000000001 \\
&a(4) = 0.10000000000000001 \\
&a(5) = 0.10000000000000001 \\
&a(6) = 0.10000000000000001 \\
&a(7) = 0.10000000000000001 \\
&a(8) = 0.10000000000000001
\end{align*}
\]

The transfer function is expressed in the above form so that the coefficients can be used as given in a cascade implementation of the filter.

Figure 2. Sample Printout
Figure 3. Frequency Characteristic (Magnitude)
Figure 4. Frequency Characteristic (Phase)
THEORY OF PROGRAMMING METHOD

PRELIMINARIES

Each of the major steps presented in the following synopsis of operations performed by the program will be discussed in greater detail in succeeding sections:

a. Certain critical frequencies in the digital z-plane are given as input parameters. These must be transformed to continuous s-plane critical frequencies since much of the design process is carried out in the s-plane.

b. The minimum order filter of the appropriate kind that will meet the specifications is computed.

c. The s-plane pole-zero pattern of a unity bandwidth low-pass analog filter of proper order is determined.

d. These s-plane poles and zeros are mapped to the z-plane poles and zeros of the required filter.

e. The poles and zeros of the desired transfer function determine the function up to some constant factor; the required constant is obtained by forcing the maximum value of the amplitude characteristic to be unity (0 dB).

f. Next, the transfer function is manipulated into a form that can be immediately implemented in cascade form.

g. The magnitude and phase of the transfer function evaluated on the unit circle in the z-plane (i.e., the frequency response) is plotted in the frequency range specified by the user.

h. Finally, the unit sample response of the desired filter is plotted.
A few facts from filter theory are now presented as background for later discussion. The three kinds of filters considered in this report have magnitude characteristics in the analog plane defined by the following equations:

**Butterworth**

\[ |H(j\Omega)|^2 = \frac{1}{1 + \Omega^{2n}} \]  \hspace{1cm} (1a)

**Chebyshev**

\[ |H(j\Omega)|^2 = \frac{1}{1 + \epsilon^2 V_n^2(\Omega)} \]  \hspace{1cm} (1b)

**Elliptic**

\[ |H(j\Omega)|^2 = \frac{1}{1 + \epsilon^2 \psi_n^2(\Omega)} \]  \hspace{1cm} (1c)

where

- \( H(\cdot) \) = the transfer function of the unity bandwidth low-pass filter
- \( \Omega \) = radian frequency
- \( n \) = the order of the filter
- \( \epsilon^2 = 10^{R/10} - 1 \), where \( R \) is the amount of passband ripple (in decibels)
- \( V_n(\cdot) \) = Chebyshev polynomial of order \( n \)
- \( \psi_n(\Omega) \) = complete elliptic integral of the first kind

\[ \psi_n(\Omega) = \begin{cases} 
\text{sn} \left[ n \frac{K(k_1)}{K(k)} \text{sn}^{-1}(\Omega, k), k_1 \right], & \text{if } n \text{ odd} \\
\text{sn} \left[ K(k_1) + n \frac{K(k)}{K(k_1)} \text{sn}^{-1}(\Omega, k), k_1 \right], & \text{if } n \text{ even} 
\end{cases} \]

where

- \( \text{sn}(\cdot, \cdot) \) = Jacobian elliptic function
- \( K(\cdot) \) = complete elliptic integral of the first kind
\[ k = 1/\Omega^S, \quad \Omega^S \text{ being the start of the stopband} \]

\[ k_1 = \epsilon/\sqrt{A^2 - 1}, \text{ with } A = 10^{S/20} \text{ and } S \text{ equal to the minimum stopband attenuation (in decibels)}. \]

Two types of mappings that play a major role in the method will now be presented. The first is a map that converts a unity bandwidth low-pass analog filter, say \( H(s) \), into an analog filter of another type with different cutoff(s).

If \( H(s) \) is such a filter, then we see that

\[ H(s/\Omega) \]

is low-pass with cutoff \( \Omega \) \hspace{1cm} (2a)

\[ H(\Omega/s) \]

is high-pass with cutoff \( \Omega \) \hspace{1cm} (2b)

\[ H \left[ \frac{s^2 + \Omega_1^c \Omega_2^c}{s (\Omega_2^c - \Omega_1^c)} \right] \]

is bandpass with cutoffs \( \Omega_1^c \text{ and } \Omega_2^c \) \hspace{1cm} (2c)

and

\[ H \left[ \frac{s(\Omega_2^c - \Omega_1^c)}{s^2 + \Omega_1^c \Omega_2^c} \right] \]

is band-reject with cutoffs \( \Omega_1^c \text{ and } \Omega_2^c \). \hspace{1cm} (2d)

A second mapping, called the bilinear transformation, maps the \( s \)-plane to the \( z \)-plane and can be used to convert an analog filter to a digital filter. The mapping is given by

\[ z = \frac{1 + s}{1 - s}. \hspace{1cm} (3) \]

Note that the imaginary axis in the \( s \)-plane is mapped to the unit circle in the \( z \)-plane and that the left half of the \( s \)-plane is mapped inside the unit circle in the \( z \)-plane. Thus, a stable analog filter will be mapped to a stable digital filter and the digital frequency response from zero to Nyquist will take on exactly the same values as the analog frequency response from zero to infinity. Also, note that the aliasing problem inherent in filters designed by the method of impulse invariance is not present here since the mapping is invertible. It is often stated that this method has the drawback of warping the frequency scale. However, no real problem exists, since the critical frequencies in a design problem can be "prewarped," as described in the next section.
In the literature, the terms Butterworth, Chebyshev, and elliptic are used in the description of low-pass analog filters. In this report these terms are also used in describing filters of other types that have been obtained by the previous two mappings.

ANALOG CRITICAL FREQUENCIES

The given digital critical frequencies must be transformed to the analog plane in such a way that when they are later mapped back to the z-plane by the bilinear transformation they will be mapped to the proper values. This "prewarping" of the critical frequencies is carried out by the map

\[ \Omega = \tan \left( \frac{\omega T}{2} \right), \]  

where

\[ \Omega = \text{continuous radian frequency} \]
\[ \omega = \text{digital radian frequency} \]
\[ T = \text{time between samples}. \]

The following argument will convince the reader of this result. Using the inverse of the bilinear transformation (3), we can relate any z-plane point to its corresponding s-plane point, i.e.,

\[ s = \frac{z - 1}{z + 1}. \]

For \( z = e^{j\omega T} \), at which points we obtain the digital frequency response, the corresponding s-plane point is

\[ s = \frac{e^{j\omega T} - 1}{e^{j\omega T} + 1} = j \tan \left( \frac{\omega T}{2} \right). \]

But the analog frequency response is obtained by evaluating the transfer function at \( s = j\Omega \). Thus, we have the correspondence dictated by (4). Now when we apply the bilinear transformation to our s-plane filter, we can be sure we will end up with a filter with proper digital critical frequencies.
DETERMINATION OF MINIMUM ORDER

We will first solve the problem for the low-pass case and then show how any of the other cases can be reduced to an equivalent problem.

Let $\omega^c$ and $\Omega^c$ be the desired digital and analog radian cutoff frequencies, respectively, and let $\omega^s$ and $\Omega^s$ be the frequencies at which the stopband begins. By (4), these frequencies are connected by the relations

$$\Omega^c = \tan\left(\frac{\omega^c T}{2}\right)$$

and

$$\Omega^s = \tan\left(\frac{\omega^s T}{2}\right),$$

where

$$\omega^c = 2\pi f^c$$

$$\omega^s = 2\pi f^s$$

$f^c$ and $f^s$ are the given input specifications in hertz.

Next, define the "transition ratio," an important factor in determining the required order, by

$$\Omega^T = \frac{\Omega^s}{\Omega^c}.$$  

Using the expressions (1a)-(1c) given previously for the three magnitude characteristics, we can determine $N$, the minimum order that will satisfy the filter requirements. Omitting details of the algebra, we now present general expressions for calculation of $N$.

If a Butterworth filter is desired, $N$ can be computed as the smallest integer that is greater than

$$\frac{\log_{10} \left(10^{S/10} - 1\right)}{2 \log_{10} \Omega^T}.$$  

15
For a Chebyshev filter, we compute the required order by finding the smallest value of $N$:

$$V_N(\Omega) \geq \left(\frac{10^{S/10} - 1}{10^{R/10} - 1}\right)^{1/2}.$$  

Finally, if an elliptic filter is desired, we determine $N$ by finding the smallest integer that is greater than

$$\frac{K\left(\sqrt{1 - k_1^2}\right) K(1/\Omega^T)}{K(k_1) K\left(\sqrt{1 - (1/\Omega^T)^2}\right)}.$$  

All the parameters used in these expressions have been defined in this or previous sections.

The only additional problem arises in the elliptic case and results from the fact that the expression above will ordinarily not be an integer and must be rounded upwards. Since it is required that

$$N = \frac{K\left(\sqrt{1 - k_1^2}\right) K(1/\Omega^T)}{K(k_1) K\left(\sqrt{1 - (1/\Omega^T)^2}\right)},$$

the value of $k_1$ must be recalculated after $N$ is determined. A convenient formula* for recomputation of $k_1$ is

$$k_1 = -\frac{2q^{0.25} \left\{1 + \sum_{i=1}^{\infty} q^{i(i+1)}\right\}^2}{1 + 2 \sum_{i=1}^{\infty} q^{i2}},$$

where

$$q = \exp\left\{-\frac{N*K(\sqrt{1 - k^2})}{K(k)}\right\}.$$  

Note here that $k = 1/\Omega^T$.

---

Now the problem of determination of minimum order for the low-pass case is solved. For any other case we define $\Omega^T$ differently but proceed exactly as in the low-pass case. For the high-pass case the appropriate definition is

$$\Omega^T = \frac{\Omega^c}{\Omega^s}. $$

For the bandpass case choose

$$\Omega^T = \min \left( \frac{1}{\Omega_1^T}, \frac{1}{\Omega_2^T} \right),$$

and for the band-reject case choose

$$\Omega^T = \min \left( \frac{1}{\Omega_1}, \frac{1}{\Omega_2} \right),$$

where the $\Omega_i^T$ are given by

$$\Omega_i^T = \frac{(\Omega_i^c)^2 - \Omega_1^c \Omega_2^c}{\Omega_i^s (\Omega_2^c - \Omega_1^c)}, \quad i = 1, 2.$$ 

By $\Omega_1^c$ and $\Omega_2^c$ we mean the two cutoff frequencies, and by $\Omega_1^s$ and $\Omega_2^s$ we mean the two boundaries between stopband and transition band.

These rules for determination of $\Omega^T$ and, hence, $N$, for filter types other than low-pass, follow directly from (2a)-(2d), the transformations that convert a low-pass filter to one of another type. Note, however, that in the bandpass and band-reject cases, the computed $N$ will equal the order of the basic low-pass structure before transformation to the filter of proper type. The actual order of the final filter will be $2N$.

In the above we have chosen the minimum order filter which meets the user's requirements. It should be mentioned, however, that the user may find another order more appealing in certain special cases. For example, for band-reject Chebyshev or elliptic filters that are derived from even order low-pass structures, we find that the magnitude characteristic is asymptotic to $-R \, \text{dB}$ in
the passband, and is actually near -R dB over a large percentage of the frequency range. Thus, a filter derived from an odd order low-pass structure may be preferred here, since its frequency response is asymptotic to 0 dB, the ideal value in the passband.

DETERMINATION OF S-PLANE POLE-ZERO PATTERN

A lengthy but straightforward task is the computation of the pole-zero pattern of a continuous unity bandwidth low-pass filter of order N. The techniques are adequately reviewed by Gold and Rader* and will not be repeated here. However, it is not necessary to keep track of all poles and zeros since they always occur in complex conjugate pairs, and it is not necessary to compute any zeros, except in the elliptic case, since they are always at infinity in the s-plane.

Also, the complete elliptic integral of the first kind, which is needed in the elliptic design problem, is given exactly by

\[
K(k) = \int_{0}^{1} \frac{1}{\sqrt{(1 - t^2)(1 - k^2 t^2)}} dt, \]

but in the program is approximated by the formula†

\[
K(m) \doteq (a_0 + a_1 m_1 + \cdots + a_4 m_1^4) + (b_0 + b_1 m_1 + \cdots + b_4 m_1^4) \ln \left(\frac{1}{m_1}\right),
\]


where

\[ a_0 = 1.38629436112 \quad b_0 = 0.5 \]
\[ a_1 = 0.09666344259 \quad b_1 = 0.12498593597 \]
\[ a_2 = 0.03590092383 \quad b_2 = 0.06880248576 \]
\[ a_3 = 0.0374563713 \quad b_3 = 0.0332355346 \]
\[ a_4 = 0.01451196212 \quad b_4 = 0.00441787012 \]

and

\[ m = k^2 \]
\[ m_1 = 1 - m. \]

The maximum error in this formula is \(2 \times 10^{-8}\) when \(0 \leq m < 1\), in which range \(m\) will always be.

Also, other approximations* are used in the calculation of elliptic poles and zeros. Iterative and/or series calculations can be used to replace these approximations but the results obtained with them were deemed sufficiently accurate.

**CALCULATION OF POLE-ZERO PATTERN OF THE DESIRED DIGITAL FILTER**

Once the s-plane poles and zeros of a unity bandwidth low-pass continuous filter of proper order and kind have been determined, it is possible to map these to the poles and zeros of the desired digital filter, which may be of any type with any cutoff(s).

Although the calculations vary by type of filter desired, the method is basically the same, and will be shown for the bandpass case only. If \(H(s)\) is the transfer function of a unity bandwidth low-pass continuous filter of order \(N\), then

\[ H_A(s) = H \left[ \frac{s^2 + \Omega_1^c \Omega_2^c}{s (\Omega_2^c - \Omega_1^c)} \right] \]

*Gold and Rader, op. cit.*

19
is the transfer function of a continuous bandpass filter of order $2N$ with cutoffs $\Omega_1^c$ and $\Omega_2^c$.

Now applying the bilinear transformation (3), we see that

$$
H_D(z) = H \left[ \frac{z-1}{z+1} \right] = H \left[ \frac{(\frac{z-1}{z+1})^2 + \Omega_1^c \Omega_2^c}{(\frac{z-1}{z+1})(\Omega_2^c - \Omega_1^c)} \right]
$$

is the transfer function of a bandpass digital filter of order $2N$ with cutoffs at

$$
\omega_1^c = \frac{2}{T} \tan^{-1} \left( \Omega_1^c \right)
$$

and

$$
\omega_2^c = \frac{2}{T} \tan^{-1} \left( \Omega_2^c \right).
$$

Now if we solve the equation

$$
s = \frac{(\frac{z-1}{z+1})^2 + \Omega_1^c \Omega_2^c}{(\frac{z-1}{z+1})(\Omega_2^c - \Omega_1^c)}
$$

for $z$ in terms of $s$, we obtain

$$
z = \frac{(p - 1) \pm (s^2 d^2 - 4p)^{1/2}}{sd - 1 - p},
$$

where

$$
p = \Omega_1^c \Omega_2^c
$$

$$
d = \Omega_2^c \Omega_1^c.
$$
We can merely substitute the previously computed s-plane poles and zeros into this expression to obtain the poles and zeros of the desired digital bandpass transfer function $H_D(z)$. Note that each point in the s-plane is mapped to two z-plane points, and thus the number of poles and zeros doubles. Actually, the computations are not carried out for the zeros unless we are working with an elliptic filter since $H_D(z)$ will always have $N$ zeros at $z = 1$ and $N$ zeros at $z = -1$. This follows from the fact that the s-plane zeros are all at infinity.

When we have calculated the z-plane poles and zeros, we have determined the desired transfer function up to some constant factor. In the next section we will determine that constant.

**CALCULATION OF THE CONSTANT MULTIPLIER**

Thus far we have determined the transfer function, $H_D(z)$, of the desired filter as

$$H_D(z) = \frac{K \prod_{i=1}^{N} (z-z_i)}{\prod_{i=1}^{N} (z-p_i)}$$

where

- $K = \text{constant multiplier to be determined}$
- $N = \text{order of filter}$
- $z_i = \text{calculated zeros}$
- $p_i = \text{calculated poles}$.

Knowing the value of $H_D(z)$ for any $z$ ($z \neq z_i$ or $p_i$) will enable us to determine $K$ uniquely.
Let $E$ be defined by

$$E = \begin{cases} 1 & \text{for } N \text{ odd or Butterworth} \\ \frac{1}{\sqrt{1 + \epsilon^2}} & \text{otherwise} \end{cases}$$

Now for any low-pass continuous filter of the kinds considered in this report, we have

$$H_A(s)\bigg|_{s=0} = E$$

and, in particular, this holds true for the unity bandwidth low-pass filter constructed previously. Thus, if we knew the point $z_0$ in the $z$-plane to which the point $s = 0$ is mapped, we could merely set

$$H_D(z)\bigg|_{z=z_0} = E$$

and solve for the quantity $K$.

From the theory of the previous section it follows that the proper values for $z_0$ are

- low-pass $z_0 = 1$
- high-pass $z_0 = -1$
- bandpass $z_0 = \frac{1 - p}{1 - p} \pm \frac{2\sqrt{p}}{1 + p}$, where $p = \Omega_1^c \Omega_2^c$, as before,
- band-reject $z_0 = \pm 1$.

For example, in the bandpass case we know that any point $s_0$ in the $s$-plane will be mapped to the point

$$z_0 = \frac{(p - 1) \pm (s_0 \, d^2 - 4p)^{1/2}}{s_0 \, d - 1 - p}.$$
Hence, the point \( s_0 = 0 \) is mapped to

\[
z_0 = \frac{1 - p}{1 + p} \pm j \frac{2\sqrt{p}}{1 + p}.
\]

Note that in the bandpass and band-reject cases we have a choice of two values for \( z_0 \), since each is an image of the point \( s = 0 \).

No matter what type filter we are designing, it follows that we can determine the quantity \( K \) by

\[
K = \frac{E \prod_{i=1}^{N} (z_o - p_i)}{\prod_{i=1}^{N} (z_o - z_i)}.
\]

In computing this expression, computational techniques make it unnecessary to work with complex quantities.

**MANIPULATION INTO CASCADE FORM**

By combining the complex conjugate poles and zeros of the transfer function \( H_D(z) \) into second order factors with real coefficients, we can write the transfer function as a product of terms of the form

\[
\frac{z^2 + b_1 z + b_2}{z^2 + c_1 z + c_2}.
\]

There will also be one factor of the form

\[
\frac{z + b}{z + c},
\]

if the order of the filter is odd, corresponding to the real pole. However, when one actually implements a filter, some recursive scheme is needed to calculate the present filtered data point. With the transfer function in the above form we
can immediately write a set of difference equations that describes the filter in the time domain, and this set will correspond to a cascade realization. In the general case where the transfer function is

\[
H(z) = \frac{Y(z)}{X(z)} = K \left( \frac{z + b_1}{z + c_1} \right) \left( \frac{z^2 + b_2 z + b_3}{z^2 + c_2 z + c_3} \right) \cdots \left( \frac{z^{N-1} + b_{N-1} z + b_N}{z^{N-1} + c_{N-1} z + c_N} \right),
\]

the following set of difference equations equivalently describes the filter:

\[
\begin{align*}
y_1(n) &= x(n) - c_1 y_1(n-1) \\
y_2(n) &= y_1(n) + b_1 y_1(n-1) \\
y_3(n) &= y_2(n) - c_2 y_3(n-1) - c_3 y_3(n-1) \\
y_4(n) &= y_3(n) + b_2 y_3(n-1) + b_3 y_3(n-2) \\
y_5(n) &= y_4(n) - c_4 y_5(n-1) - c_5 y_5(n-2) \\
y_6(n) &= y_5(n) + b_4 y_5(n-1) + b_5 y_5(n-2) \\
&\vdots \\
y_N(n) &= y_{N-1}(n) - c_{N-1} y_N(n-1) - c_N y_N(n-2) \\
y(n) &= K(y_N(n) + b_{N-1} y_N(n-1) + b_N y_N(n-2)),
\end{align*}
\]

where \( x(n) \) and \( y(n) \) are the filter inputs and outputs, respectively, and \( y_1(n), y_1(n), \ldots, y_N(n) \) are intermediate values determined by the above equations.

The above filter could also be described by a single difference equation of order \( N \), which would be an entirely equivalent time domain representation, if all coefficients could be represented exactly. However, due to the necessary quantization of coefficients, this so-called "direct form" realization is not recommended since it is much more sensitive to coefficient accuracy than the corresponding cascade implementation (5).
THE MAGNITUDE AND PHASE
OF THE FREQUENCY RESPONSE

To obtain the frequency response of a digital filter, we must evaluate the transfer function on the unit circle of the complex z-plane. In particular, $|H_D(e^{j\omega T})|$, the magnitude of the frequency response, is usually of primary interest.

Thus far we have $H_D(z)$ expressed as a product of terms of the form

$$
\frac{z^2 + b_1 z + b_2}{z^2 + c_1 z + c_2}
$$

The magnitude and phase of this term for $z = e^{j\omega T}$ can easily be expressed as a trigonometric function of $\omega T$ and can be simply evaluated for any $\omega$. The product of the magnitudes of all such terms and the constant multiplier is then the magnitude of the frequency response. Similarly, the sum of the phases of all such terms gives the phase of the frequency response. The possibility of a single term of the form

$$
\frac{z + b}{z + c}
$$

poses no additional problem. In the program we compute the magnitude and phase as above, convert the magnitude to decibels and phase to degrees, and then provide plots in the frequency range desired.

THE UNIT SAMPLE RESPONSE

The rate of decay of the unit sample response (i.e., the response of the filter to a pulse of unit height at time zero) of a digital filter is often of interest to the designer. For this reason a plot is provided if requested.

Since the transfer function, $H_D(z)$, can be viewed as the $Z$-transform of $h(n)$, the unit sample response, we can write

$$
h(n) = Z^{-1} \left[ H_D(z) \right],
$$
or

\[ h(n) = \frac{1}{2\pi j} \oint_C H_D(z) z^{n-1} dz, \]

where \( C \) is any contour in the complex plane enclosing all poles of the integrand. Replacing \( H_D(z) \) by its pole-zero factorization, we obtain

\[ h(n) = \frac{K}{2\pi j} \oint_C \prod_{i=1}^{N} \frac{(z - z_i)}{(z - p_i)} z^{n-1} dz. \]

Since the \( p_i, i = 1, 2, \ldots, N \) will always be simple poles, the residue at a specific \( p_j \) of the integrand is

\[ \prod_{i=1}^{N} \frac{(p_j - z_i)}{(p_j - p_i)} p_j^{n-1}. \]

For the special case \( n = 0 \) there is also a pole at \( z = 0 \), but the complications introduced by the additional pole need not be considered since we know by examining (5) that \( h(0) \) must equal \( K \). Defining

\[ \beta_j = \prod_{i=1}^{N} \frac{(p_j - z_i)}{(p_j - p_i)}, \]

we can then easily compute the unit sample response as...
h(n) = \begin{cases} 
K \sum_{i=1}^{N} \beta_i p_i^{n-1}, & n \geq 1 \\
K, & n = 0 
\end{cases}

Although the unit sample response is a discrete function of time, the plot produced by the program is a continuous function for ease in viewing.

**SUMMARY**

Butterworth, Chebyshev, and elliptic digital filters are realizable networks that approximate physically unrealizable filters having rectangular magnitude characteristics. Each of these filters come arbitrarily close to the ideal response as the order of the approximating network becomes large. Thus, the question of the order necessary to meet a set of specifications is usually critical. The computer program described in this report computes the minimum order filter that meets the particular requirements of the user, and then proceeds to design that filter. The program also produces plots of the magnitude and phase of the frequency response and the unit sample response of the filter. The reader need not fully understand the theory presented in this report in order to use the program. Finally, for those unfamiliar with digital signal processing techniques, the instruction on how the filter transfer function can be implemented as a set of difference equations should be useful.
Appendix

LISTING OF THE COMPUTER PROGRAM
CHEBYSHEV BUTTERWORTH OR ELLIPTIC DIGITAL FILTER DESIGN PROGRAM

BUTTERWORTH BY J.J.KOLT.

THIS PROGRAM WILL DESIGN ANY LOW-PASS, HIGH-PASS, BANDPASS OR
BAND-REJECT FILTER BY BILINEAR TRANSFORMATION TECHNIQUES

LAST UPDATED 6 JUNE 1973

* * * * * * * * INSTRUCTIONS FOR INPUT SPECIFICATIONS * * * * * * * *

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>TYPE</th>
<th>COLUMNS</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>RATE</td>
<td>REAL</td>
<td>1-8</td>
<td>THE NUMBER OF DATA SAMPLES PER TIME UNIT (USUALLY SECONDS)</td>
</tr>
<tr>
<td>FCLOW</td>
<td>REAL</td>
<td>9-16</td>
<td>THE LOWER CUTOFF FREQUENCY (IN CYCLES/TIME UNIT)</td>
</tr>
<tr>
<td>FCHIGH</td>
<td>REAL</td>
<td>17-24</td>
<td>THE UPPER CUTOFF FREQUENCY (IN CYCLES/TIME UNIT)</td>
</tr>
<tr>
<td>RIPPLE</td>
<td>REAL</td>
<td>25-30</td>
<td>THE LOW-PASS ROLL-OFF ROLL-OFF IN DB)</td>
</tr>
<tr>
<td>FSLIM</td>
<td>REAL</td>
<td>31-36</td>
<td>THE LOWER FREQUENCY MARKING THE BOUNDARY BETWEEN STOPBAND AND TRANSITION BAND (IN CYCLES/TIME UNIT)</td>
</tr>
<tr>
<td>FSHIGH</td>
<td>REAL</td>
<td>39-46</td>
<td>THE HIGHER FREQUENCY MARKING THE BOUNDARY BETWEEN STOPBAND AND TRANSITION BAND (IN CYCLES/TIME UNIT)</td>
</tr>
<tr>
<td>STPURL</td>
<td>REAL</td>
<td>47-50</td>
<td>THE MINIMAL STOPBAND ATTENUATION (IN DB)</td>
</tr>
<tr>
<td>KINOD</td>
<td>INTEGER</td>
<td>51</td>
<td>THE KIND OF FILTER DESIRED I.E.</td>
</tr>
<tr>
<td>TYPE</td>
<td>INTEGER</td>
<td>52</td>
<td>TYPE OF FILTER DESIRED I.E.</td>
</tr>
<tr>
<td>WPOLES</td>
<td>INTEGER</td>
<td>53-54</td>
<td>SETS THE ORDER OF THE FILTER</td>
</tr>
<tr>
<td>IFLUT</td>
<td>INTEGER</td>
<td>56</td>
<td>SETS THE FILTER TYPE TO LOW-PASS, HIGH-PASS, BANDPASS, OR BAND-REJECT</td>
</tr>
</tbody>
</table>

THE NUMBER OF DATA SAMPLES PER TIME UNIT (USUALLY SECONDS)
THE LOWER CUTOFF FREQUENCY (IN CYCLES/TIME UNIT)
THE UPPER CUTOFF FREQUENCY (IN CYCLES/TIME UNIT)
THE LOW-PASS ROLL-OFF (IN DB)
THE LOWER FREQUENCY MARKING THE BOUNDARY BETWEEN STOPBAND AND TRANSITION BAND (IN CYCLES/TIME UNIT)
THE HIGHER FREQUENCY MARKING THE BOUNDARY BETWEEN STOPBAND AND TRANSITION BAND (IN CYCLES/TIME UNIT)
THE MINIMAL STOPBAND ATTENUATION (IN DB)
THE KIND OF FILTER DESIRED I.E. 1 CHEBYSHEV, 2 BUTTERWORTH, 3 ELLIPTIC
TYPE OF FILTER DESIRED I.E. 1 LOW-PASS, 2 HIGH-PASS, 3 BANDPASS, 4 BAND-REJECT
SET TO CALCULATE THE ORDER OF THE FILTER |
SET TO DESIRED FOR BASIC LOW-PASS STRUCTURE BEFORE TRANSFORMATION
SET TO PLOT THE MAGNITUDE OF H(Z)
SET TO NO PLOT
### READ INPUT SPECIFICATIONS OF FILTER TO BE DESIGNED

- Heade(16)
- State: FLOW, FSHIGH, HIGHF, FLOW, FSHIGH, STPLVL, INKIND
- IF ITYPE=1, POLES, PLOT, HTPE, HPPT, FLW, FSHIHF
- E HEAT=1/38, POLES, PLOT, HTPE, HPPT, FLW, FSHIHF
- IF ITYPE=2, POLES, PLOT, HTPE, HPPT, FLW, FSHIHF
- IF ITYPE=3, POLES, PLOT, HTPE, HPPT, FLW, FSHIHF
- IF ITYPE=4, POLES, PLOT, HTPE, HPPT, FLW, FSHIHF

### WRITE OUT INPUT SPECIFICATIONS ON PRINTER

- Heade(17)
- State: FLOW, FSHIGH, HIGHF, FLOW, FSHIGH, STPLVL, INKIND
- IF ITYPE=1, POLES, PLOT, HTPE, HPPT, FLW, FSHIHF
- IF ITYPE=2, POLES, PLOT, HTPE, HPPT, FLW, FSHIHF
- IF ITYPE=3, POLES, PLOT, HTPE, HPPT, FLW, FSHIHF
- IF ITYPE=4, POLES, PLOT, HTPE, HPPT, FLW, FSHIHF

### Note

- IF NTYPES.NT=0 (I.E., THE ORDER IS SPECIFIED), FLOW, FSHIGH, AND STPLVL ARE NOT USED (UNLESS ELLIPTIC) SINCE THE GIVEN ORDER DETERMINES THESE QUANTITIES
- IF LOW-PASS OR HIGH-PASS, FSHIGH AND FSHIGH ARE NOT USED
- IF BUTTERWORTH IS DESIRED, RIPPLE IS NOT USED
- THE QUANTITY RIPPLE AND STPLVL SHOULD ALWAYS BE GIVEN, I.E., 0
- REPEAT DATA CARD UP TO THIS TYPE FOR EACH FILTER TO BE DESIGNED
- PLACE A 'BLANK' DATA CARD LAST

### Standards

UO1: TEGEP 37-60 NO. OF PTS IN UNIT SAMPLE RESPONSE (DEFAULT VALUE = 100)
- UNI: H2AL 0770 LOW END FREQUENCY OF PLOT
- H2IN: H2AL 7I-80 HIGH END FREQUENCY OF PLOT

### Reproduced from best available copy.
FD Flow = 1.0, V2 = 1.0, V3 = 1.0, V4 = 1.0, V5 = 1.0, V6 = 1.0, V7 = 1.0, V8 = 1.0, V9 = 1.0,
V10 = 1.0, V11 = 1.0, V12 = 1.0, V13 = 1.0, V14 = 1.0, V15 = 1.0, V16 = 1.0, V17 = 1.0, V18 = 1.0,
V19 = 1.0, V20 = 1.0, V21 = 1.0, V22 = 1.0, V23 = 1.0, V24 = 1.0, V25 = 1.0, V26 = 1.0,
V27 = 1.0, V28 = 1.0, V29 = 1.0, V30 = 1.0, V31 = 1.0, V32 = 1.0, V33 = 1.0, V34 = 1.0,
V35 = 1.0, V36 = 1.0, V37 = 1.0, V38 = 1.0, V39 = 1.0, V40 = 1.0, V41 = 1.0, V42 = 1.0,
V43 = 1.0, V44 = 1.0, V45 = 1.0, V46 = 1.0, V47 = 1.0, V48 = 1.0, V49 = 1.0, V50 = 1.0,
V51 = 1.0, V52 = 1.0, V53 = 1.0, V54 = 1.0, V55 = 1.0, V56 = 1.0, V57 = 1.0, V58 = 1.0,
V59 = 1.0, V60 = 1.0, V61 = 1.0, V62 = 1.0, V63 = 1.0, V64 = 1.0, V65 = 1.0, V66 = 1.0,
V67 = 1.0, V68 = 1.0, V69 = 1.0, V70 = 1.0, V71 = 1.0, V72 = 1.0, V73 = 1.0, V74 = 1.0,
V75 = 1.0, V76 = 1.0, V77 = 1.0, V78 = 1.0, V79 = 1.0, V80 = 1.0, V81 = 1.0, V82 = 1.0,
V83 = 1.0, V84 = 1.0, V85 = 1.0, V86 = 1.0, V87 = 1.0, V88 = 1.0, V89 = 1.0, V90 = 1.0,
V91 = 1.0, V92 = 1.0, V93 = 1.0, V94 = 1.0, V95 = 1.0, V96 = 1.0, V97 = 1.0, V98 = 1.0,
V99 = 1.0, V100 = 1.0, V101 = 1.0, V102 = 1.0, V103 = 1.0, V104 = 1.0, V105 = 1.0, V106 = 1.0,
IF(OMEGA<0.1,VIU) GO TO 25
VIU = 123*U1
V1(1) = 0.0+OMEGA*V1(1-1)-V1(1-2)
IF(V1(1)<0.0,VIU) GO TO 20
NPULSES=L-1
GO TO 30
20 CONTINUE
21 NINT4(N*22)
26 NINT4(INDA) THE SPECIFICATIONS REQUIRE A FILTER OF ORDER .Ut. 20)
GO TO 20
20 NPULSES
GO TO 30
30 NPULSES=ALOG10((1.0+STRIVL/10.0)-1.0)/(2.*ALOG10(OMEGA))+1
IF(NPULSES<0.20) GO TO 21
NPULSES=NPULSES+1
GO TO 30
40 AI=0.0+STRIVL/20.
B=PSD/(AI*AI)
C1=3.0/((2.0-OMEGA)
ELLIPIC(1,-A)
ELLIPIC(1,-C)
ELLIPIC(T0)
ELLIPIC(1)
NPULSES=NPULSES+1
IF(NPULSES<0.20) GO TO 21
NPULSES=NPULSES+1
GO TO 40
50 FE0,
GO TO 47 101+100
WS=AI-1)
IF(DL<1.1+30) GO TO 48
GO TO 10
47 CONTINUE
40 BI=(2.0+0.25A)/(1.0+2.0F)
C=(2.0+0.25A)/(1.0+2.0F)
ELLIPIC(1,-F)
F=ELLIPI(1,-F)
GO TO 30
NPULSES=(1.0+11)/2
WHILE(NPULSES) 11
26 NINT4(INDA) THE MAXIMUM ORDER FILTER WHICH MEETS THESE SPECIFICATIONS IS ORDER 1,12.
30 CONTINUE

CALCULATE THF SECOND QUADRANT POLE POSITIONS IN THE S-PLANE
OF A UNITY UNAMPSWIDTH LOW PASS FILTER OF THE REQUIRED ORDER.
REALIZING THAT THEIN COMPLEX CONJUGATES ARE ALSO POLES
40 NINT4(NPULSES-1)
NPULSES=NPOLE+1
11
GO TO 33
LSTM(TL,0,0)/NPOLL
IF(INIнов,Ev,3) 60 TO 90

ONbALElXPI*EXPNT
A1=0
B1=0
IF((iniEv,Ev,2) 60 TO 70
C=USNMT(1.000+1.000/EPS2)+1.000/USNRT(EPS2)**EXPNT
AS,500+(C,1.000/C)
B5,500+(C,1.000/C)
70 ANGLE=0.0
IF(100*EV,0) ANGLE=0.0
U0 80 121
POLES(i+1)=1.000*xPOES(ANGLE)
POLES(i+2)=BDUSIN(ANGLE)
ANGLE=ANGR.+ANGLE
80 CUNITUE
60 TO 110
90 if=(6/F)*USO((600+TL,1+7,2)1+1)/USO(EPS2)
F=USNMT(1.+C)

* 90 USELLF:(IFF)
BE_fff:(1,-F,F)
USELF:=(FF<FF)
ANGRT=(FF/FF)
S=0
U0 94 U0 1000
(94*EUS(J,2)/11.-USO(1011))
SUSIN=USNMT(120+1+2)*ANGL
IF((U1,1000) 60 TO 95
94 CONTINUE
95 S=US(J,11,1,1,F)/F
60 TO 10 (9F*W1)ENDL

* 90 S=US
CN/USNRT(1,-USIN(1))
USDUSNRT(1,-(F*USIN)**)
95 CONTINUE
90 S=US
CN/USNRT(1,-USIN(1))
USDUSNRT(1,-(F*USIN)**)
121
60 TO 92
90 Sn=US
CN/USNRT(1,-USIN(1))
USDUSNRT(1,-(F*USIN)**)
121
POLES(i+1)=SUSIN(1.000+1.000/EPS2)+1.000/USNRT(EPS2)**EXPNT
AS,500+(C,1.000/C)
B5,500+(C,1.000/C)
110 ANGLE=0.0
IF(100*Ev,0) ANGLE=0.0
U0 80 121
POLES(i+1)=1.000*xPOES(ANGLE)
POLES(i+2)=BDUSIN(ANGLE)
ANGLE=ANGR.+ANGLE
100 CUNITUE
60 TO 92
HAPPY THE S-PLANE POLES TO THE Z-PLANE USING THE BILINEAR TRANSFORMATION TAKING THE CUTOFF FREQUENCY AND TYPE OF FILTER INTO ACCOUNT.

ALSO, IF ELLIPTIC IS DESIRED CARRY OUT THE SAME PROCEDURE FOR THE ZEROS.

110 IF(I=ITHE,61,2) GO TO 170
    IF(I=ITHE,61,1) GO TO 140
    PULEZ(J1,1)=(POLES(1,1)-WCUT1)/(POLES(1,1)+WCUT1)
    PULEZ(J1,2)=0.0
    IF(PULES(J1,1)) GO TO 300
    U=1

120 L7=L

    UV-I=1+K
    A=POLES(1,1)+WCUT1
    B=POLES(1,1)
    UVU=POLE(1,1)+WCUT1
    PULEZ(J1,1)=(A+WCUT1)/((A-D)APOLZ(J1,1))
    PULEZ(J1,2)=(A-D)/((A-D)APOLZ(J1,2))
    PULEZ(J1,1)=POLE(J1,1)
    PULEZ(J1,2)=POLE(J1,2)
    UVU=POLE(J1,1)
    IF(U=1) GO TO 110

140 C1=0.000/WCUT1
    PULEZ(J1,1)=POLES(1,1)+C=POLES(1,1)
    PULEZ(J1,2)=0.0
    IF(PULES(J1,1)) GO TO 300
    U=2

150 L7=L

    UV-I=1+K
    D=POLES(1,1)+C
    E=POLES(1,1)
    UVU=POLE(1,1)+WCUT1
    PULEZ(J1,1)=(D+WCUT1)/((D-D)APOLZ(J1,1))
    PULEZ(J1,2)=(D-D)/((D-D)APOLZ(J1,2))
    PULEZ(J1,1)=POLE(J1,1)
    PULEZ(J1,2)=POLE(J1,2)
    UVU=POLE(J1,1)
    IF(U=1) GO TO 110
    ZEHOS(J1,1)=ZEHOS(J1,1)-WCUT1/WCUT1
    ZEHOS(J1,2)=ZEHOS(J1,2)
    ZEHOS(J1,1)=ZEHOS(J1,1)+WCUT1/WCUT1
    ZEHOS(J1,2)=ZEHOS(J1,2)

170 U=1

180 ZEHOS(J1,1)=ZEHOS(J1,1)+WCUT1/WCUT1
    ZEHOS(J1,2)=ZEHOS(J1,2)
    ZEHOS(J1,1)=ZEHOS(J1,1)-WCUT1/WCUT1
    ZEHOS(J1,2)=ZEHOS(J1,2)

190 U=2
160 CONTINUE
GO TO 300

170 IF (ITYPE.EQ.3) GO TO 190
DU 180 11, K
APOLEX(I)
POLES(I) = POLES(I-1) + POLES(I-2)
POLES(I) = POLES(I-2) + POLES(I-1)
IF (K.I.EQ.INC) GO TO 190
IF (1.EQ.1.AND.100.EQ.1) GO TO 180
AZ<0) = (1)
ZRNOS(I) = ZEROS(I) + ZEROS(I)
ZRNOS(I) = ZEROS(I) + ZEROS(I)
180 CONTINUE

190 INO = 0
190 CONTINUE
JE 1

200 11, K
IF (1.EQ.1.AND.100.EQ.1) GO TO 198
IF (1.EQ.1.AND.3.EQ.1) GO TO 300
INCR2 = 1
POLEZ(I) = POLEZ(I-1) + POLEZ(I-2)
POLEZ(I) = POLEZ(I-2) + POLEZ(I-1)
IF (INCR2.EQ.1) GO TO 200
ZRNOS(I) = ZEROS(I) + ZEROS(I)
ZRNOS(I) = ZEROS(I) + ZEROS(I)
200 CONTINUE

1 IF (K.I.NC.3.OR.1.NDC, EQ.1) GO TO 300
DU 210 11, K
POLES(I) = ZEROS(I)

36
PULIFZ3(1,2)ZEROS3(1,2)

410 continue
in1=c1
uu to 190

300 continue
if(i1=0,i2,3,an1,10,0)
if(i1=0,i2,3)
46=1

--- Fill up the zeros array with the z-plane zeros (unless elliptic) in which case the zeros have been calculated. ---

uu 400 121124
uu to (310,320,330,340)*1type
310 ZEH02(1,2)=1.
ZEH2(1,2)=0.
UU 10 400
320 ZEH02(1,2)=1.
ZEH2(1,2)=0.
UU 10 400
330 ZEH02(1,2)=(-1.1))=(-1)
ZEH2(1,2)=0.
UU 10 400
340 ZEH02(1,2)=(-1,1))=((1)
ZEH2(1,2)=0.
UU 10 400
350 ZEH02(1,2)=(-1,1))=(2)
ZEH2(1,2)=1.
UU 10 400
360 ZEH02(1,2)=(-1,1))=(3)
ZEH2(1,2)=0.
UU 10 400
370 ZEH02(1,2)=(-1,1))=(4)
ZEH2(1,2)=0.
UU 10 400
380 ZEH02(1,2)=(-1,1))=(5)
ZEH2(1,2)=0.
UU 10 400
390 ZEH02(1,2)=(-1,1))=(6)
ZEH2(1,2)=0.
UU 10 400
400 continue

--- Determine the coefficients in the denominator polynomial of the transfer function by combining the complex conjugate pole paths into real second order factors. If the order is odd there will also be one term of order one, corresponding to the real pole. Follow the same procedure to determine the numerator coefficients. ---

410 u2
if(i1type.0)=1)
uu to 450
CULF(1,1)=POLEZ(1,1)
CULF(1,2)=CPLF(1,1)
CULF(1,2)=ZEROS(1,1)
CULE(1,1)=1.
if(npules.ne.1)
uu to 440
L1=0
420 u2
if(i1type.0)=1)
uu to 450
CULF(1,1)=CPLF(1,1)+CULE(1,1)
CULF(1,2)=CPLF(1,1)+CULE(1,1)
UU to 300
440 u2
450 uu
if(i1type.0)=1)
CULF(1,1)=POLEZ(1,1)
CULF(1,2)=POLEZ(1,1)+POLEZ(1,1)
CULF(1,2)=POLEZ(1,1)+POLEZ(1,1)
CONTINUE

HOW THAT WE HAVE THE POLE-ZERO PATTERN OF THE DESIRED TRANSFER
FUNCTION, H(z), WE HAVE DETERMINED H(z) UP TO SOME CONSTANT.
NEXT DETERMINE THAT CONSTANT.

ES1.000
IF (IONU.EQ.0.AND.WAND.NE.1) ES=1.000/OSQRT(1.000+EPS2)
CS=QUOT
GO TO (470,480,490,470,1)YPE

470 QU 475 [21;M
AS=COEF(1,1)+COEF(1,2)
BS=1.000+COEF(1,1)+COEF(1,2)
CS=QUOT
470 CONTINUE
GO TO 300

480 QU 480 [21;K
AS=COEF(1,1)+COEF(1,2)
BS=1.000+COEF(1,1)+COEF(1,2)
CS=QUOT
480 CONTINUE
GO TO 300

490 METHD4DATA(2,0.OSQRT(4.0D1),1,-PROU)
G=2.0COS(2.0)*META
G=2.0COS(THETA)
G=2.0SIN(2.0)*META
G=2.0SIN(THETA)
DO 404 1=1.INPOLES
BS=500(F+COEF(1,1)+COEF(1,2)+COEF(1,1)*0+COEF(1,1)+2)
AS=500(F+COEF(1,1)+COEF(1,2)+COEF(1,1)*0+COEF(1,1)+2)
CS=QUOT
490 CONTINUE

500 CONTINUE

510 CSTGSE4C

CALCULATIONS ARE COMPLETE, WRITE OUT THE TRANSFER FUNCTION
AS A RATIO OF POLYNOMIALS WHICH CAN BE IMMEDIATELY
IMPLEMENTED IN 'CASCADE' FORM

WRITE(4,510)

510 FORMAL (///X.8 THE TRANSFER FUNCTION,H(z), OF THE DESIRED FILTER
AS A RATIO OF POLYNOMIALS WHICH CAN BE IMMEDIATELY
IMPLEMENTED IN 'CASCADE' FORM

WRITE(4,520)
IF (IONU.EQ.0.AND.WAND.NE.1) WRITE(4,530)
WRITE(4,540)
WRITE(4,550)
WRITE(4,560)
WRITE(4,570)
WRITE(4,580)
WRITE(4,590)
WRITE(4,510)
WRITE(4,520)
WRITE(4,530)
WRITE(4,540)
WRITE(4,550)
WRITE(4,560)
WRITE(4,570)
WRITE(4,580)
WRITE(4,590)

38
530 FORMAT(//,5X,'Z**2*RA(1)+2*RA(2)*X+7X**2+2*RA(3)*X+8*RA(4)*X+9*RA(5)*X+10*RA(6))

540 FORMAT(//,1X,'H(1)'+SN(5)') CNST

550 FORMAT(//,1X,'The CUEfficicients ARE Given By: H(1)'+SN(5)') CNST = 

560 FORMAT(1X,10E16)

570 FORMAT(//,1X,'The Transfer Function is Expressed in the Above FOR 

580 FORMAT(//,1X,'Gu that H(1)'+SN(5)') CUNTENT la Given in a Cascade 

590 FORMAT(//,1X,'IMPLEMENTATION OF THE FILTER')

600 FORMAT(//,1X,'Go To 5')

610 FORMAT(//,1X,'Go To 5B')

620 FORMAT(//,1X,'Go To 5A')

630 FORMAT(//,1X,'Go To 5B')

640 FORMAT(//,1X,'Go To 5B')

650 FORMAT(//,1X,'Go To 5B')

660 FORMAT(//,1X,'Go To 5B')

670 FORMAT(//,1X,'Go To 5B')

680 FORMAT(//,1X,'Go To 5B')

690 FORMAT(//,1X,'Go To 5B')

700 FORMAT(//,1X,'Go To 5B')
CALL NUSDU(Z)  
CALL NUSDU(Z+0.0,50.0,0.0,050.0,0)  
CALL NUSDU(Z,50.0,0.0,0.0,0)  
CALL LABELU(Z+0+7,0+0,0+0,0,0)  
CALL LEDEL(Z+1+0,0.0,0.0,0,1)  
CALL LUSDU(Z+100.0 X+7)  

CALL TITLEU(Z,29,"FREQUENCY IN CYCLES/TIME UNIT",.0,"DECIBELS",.0)  

CALL UGCIU(Z,200.0,40.0,LO", "PASS")  
GO TO 705  

CALL UGCIU(Z,200.0,50.0,495.0,3050.0)  
CALL UGCIU(Z+0.0,1045.1,100.1)  
CALL LONGLU(Z+200.0,211)"DESCRIPTION OF FILTER")  
GO TO 701,702,711,712;TIME  

CALL UGCIU(Z+200.0,80.0,40.0,80.0)  
CALL UGCIU(Z,200.0,40.0,40.0,80.0)  
CALL NUMPtr(Z+50.0,50.0,0.0,0.0,0)  
CALL NUMPtr(Z+100.0,100.0,0.0,0.0,0)  
CALL NUMPtr(Z+150.0,150.0,0.0,0.0,0)  
CALL NUMPtr(Z+200.0,200.0,0.0,0.0,0)  

CALL UGCIU(Z,200.0,50.0,40.0,50.0)  
GO TO 700  

CALL LONGLU(Z,200.0,70.0,2.0,2.0,1)"AMPLITUDE"

CALL UGCIU(Z,200.0,50.0,40.0,50.0)  
GO TO 700  

CALL UGCIU(Z,200.0,50.0,40.0,50.0)  

CALL LONGLU(Z,200.0,75.0,2.0,2.0,1)"AMPLITUDE"

GO TO 700  

CALL UGCIU(Z,200.0,50.0,40.0,50.0)  
GO TO 700  

CALL LONGLU(Z,200.0,75.0,2.0,2.0,1)"AMPLITUDE"
CALL UJUZU(Z'0,1,1J)
CALL PPGW(2'00,.0.0.0)
TR 4629

I$$(\text{MAX} \gt \text{LT} 0.002) \quad \text{MAX}=0.002$$
$$\text{MAX}=0.002$$
$$\text{MAX}=2.0$$
$$\text{MAX}=10$$
$$\text{MAX}=20.0$$

CALL U0U(2,0,0,0,0,0,0,0,0,0)
CALL U0U(2,0,0,0,0,0,0,0,0,0)
CALL U0U(2,0,0,0,0,0,0,0,0,0)
CALL U0U(2,0,0,0,0,0,0,0,0,0)
CALL U0U(2,0,0,0,0,0,0,0,0,0)
CALL U0U(2,0,0,0,0,0,0,0,0,0)
CALL U0U(2,0,0,0,0,0,0,0,0,0)

CALL INPUTU(2,0,0,0,0,0,0,0,0,0)
CALL INPUTU(2,0,0,0,0,0,0,0,0,0)
CALL INPUTU(2,0,0,0,0,0,0,0,0,0)
CALL INPUTU(2,0,0,0,0,0,0,0,0,0)
CALL INPUTU(2,0,0,0,0,0,0,0,0,0)
CALL INPUTU(2,0,0,0,0,0,0,0,0,0)
CALL INPUTU(2,0,0,0,0,0,0,0,0,0)

CALL TITLE(12,15)"RESPONSE NUMBER",12,"OUTPUT VALUE",36,"PLOT OF U"
CALL PAGE(2,2,1)
60 TO C
1000 CONTINUE
CALL EXIT(2)
END