DESIGN OF RECURSIVE DIGITAL FILTERS HAVING SPECIFIED PHASE AND MAGNITUDE CHARACTERISTICS

by

Robert E. King
Langley Research Center

and

Gregory W. Condon
Langley Directorate,
U.S. Army Air Mobility R&D Laboratory

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • APRIL 1972
**Abstract**

A method for a computer-aided design of a class of optimum filters, having specifications in the frequency domain of both magnitude and phase, is described. The method, an extension to the work of Steiglitz, uses the Fletcher-Powell algorithm to minimize a weighted squared magnitude and phase criterion. Results using the algorithm for the design of filters having specified phase as well as specified magnitude and phase compromise are presented.
DESIGN OF RECURSIVE DIGITAL FILTERS HAVING SPECIFIED PHASE AND MAGNITUDE CHARACTERISTICS

By Robert E. King
Langley Research Center

and

Gregory W. Condon
Langley Directorate, U.S. Army Air Mobility R&D Laboratory

SUMMARY

A method for a computer-aided design of a class of optimum filters, having specifications in the frequency domain of both magnitude and phase, is described. The method, an extension to the work of Steiglitz, uses the Fletcher-Powell algorithm to minimize a weighted squared magnitude and phase criterion. Results using the algorithm for the design of filters having specified phase as well as specified magnitude and phase compromise are presented.

INTRODUCTION

Recursive filters, wherein the output sequence is both a function of the input as well as past output samples, are commonly used in digital signal processing and analysis. Such digital filters in many applications offer distinct advantages of precision and versatility over their continuous or analog counterparts. There exist a number of design procedures for implementing digital filters (see ref. 1) each one of which strives to attain some analogy between discrete and continuous systems. Transform methods such as the matched-z, bilinear-z, and standard-z which lead to specific property invariances are available (see ref. 2) to the designer familiar with continuous filter design.

For frequency-domain synthesis (see refs. 3 and 4), realization is normally by means of cascade or parallel combinations of pole and zero pairs in the complex plane. The synthesis problem is, in fact, reduced to one of approximation since the filter topology is generally specified. In none of the available design procedures, which can yield filters having excellent magnitude-frequency characteristics, however, do the resultant filters, in themselves, have particularly useful phase characteristics. Indeed, in striving for particular magnitude characteristics by using any of the available design methods, there is no control over the filter phase properties.
In practice, it is often desirable to specify a digital filter in the frequency domain by its phase (see ref. 5) or even a compromise between magnitude and phase. The procedure in this paper meets these requirements through the use of an iterative computer-aided design leading to an optimum set of parameters for a specified filter topology and is an extension of the technique described by Steiglitz (see ref. 6) for determining the optimum coefficients of a cascade filter having magnitude specifications alone. The extension makes possible the design of a new class of digital filters having the prescribed phase characteristics.

**SYMBOLS**

- \( A \) filter multiplier
- \( D_k^i \) denominator of \( i \)th stage of \( H(z) \) at \( \Omega_k \)
- \( E_k^M \) magnitude error at \( \Omega_k \)
- \( E_k^\phi \) phase error at \( \Omega_k \)
- \( \bar{e}_k \) error vector at \( \Omega_k \)
- \( \partial \bar{e}_k / \partial A \) derivative of error vector at \( \Omega_k \) with respect to zero frequency gain
- \( f_k \) frequency at \( k \)th specification point, Hz
- \( f_s \) sampling frequency, Hz
- \( H(z) \) unity gain discrete transfer function
- \( |H_k| \) magnitude of \( H(z) \) at \( \Omega_k \)
- \( \bar{H}_k \) conjugate of \( H(z) \) at \( \Omega_k \)
- \( \partial |H_k| / \partial \bar{p} \) gradient vector of magnitude of \( H(z) \) at \( \Omega_k \) with respect to parameter vector
- \( I(\ ) \) imaginary part of quantity
- \( i, \ldots, N \) denotes filter stage
\[ \overline{J}_k \quad \text{Jacobian at } \Omega_k, \quad \begin{bmatrix} A^* \frac{\partial |H_k|}{\partial \overline{p}} & \frac{\partial \phi_k}{\partial \overline{p}} \end{bmatrix} \]

\( k \quad \text{sample point} \)

\( M_k \quad \text{specification magnitude at } \Omega_k \)

\( N_k^i \quad \text{numerator of ith stage of } H(z) \text{ at } \Omega_k \)

\( \overline{p} \quad \text{parameter vector} \)

\( \overline{p}_i \quad \text{set of filter parameters for the ith stage, } a_i, b_i, c_i, \text{ and } d_i \)

\( q_1^i(k) \quad \text{first system state of ith stage at kth sample point} \)

\( q_2^i(k) \quad \text{second system state of ith stage at kth sample point} \)

\( R(\ ) \quad \text{real part of quantity} \)

\( u_l(k) \quad \text{input to ith stage at kth sample point} \)

\( V \quad \text{criterion functional, that is, } V(A,\overline{p}) \)

\( V_k \quad \text{criterion functional at } \Omega_k, \text{ that is, } V_k(A,\overline{p}) \)

\( \hat{V} \quad \text{reduced criterion functional, that is, } V(A^*,\overline{p}) \)

\( \frac{\partial V}{\partial A} \quad \text{slope of criterion functional with respect to zero frequency gain} \)

\( \frac{\partial V_k}{\partial \epsilon_k} \quad \text{gradient vector of criterion functional at } \Omega_k \text{ with respect to error vector at } \Omega_k \)

\( \overline{W}_k \quad \text{weighting matrix at } \Omega_k \)

\( W_k^M \quad \text{magnitude weighting at } \Omega_k \)

\( W_k^\phi \quad \text{phase weighting at } \Omega_k \)

\( w^i(k) \quad \text{dummy variable of ith stage at kth sample point} \)
The fundamental advantages of the N-stage cascade canonical form of recursive digital filter whose signal flow graph is shown in figure 1 and which is described by the product operator

\[ Y(z) = A \prod_{i=1}^{N} \frac{1 + a_i z^{-1} + b_i z^{-2}}{1 + c_i z^{-1} + d_i z^{-2}} \]

are (1) its relative insensitivity to perturbations in the denominator coefficients, an important consideration in digital filters, especially of high order and particularly where finite register lengths (see ref. 1) are involved; (2) its simplicity of implementation; and (3) the simplicity of factoring the filter operator to determine its roots. This form has found extensive application in practical filters for signal processing, and a version employing serial arithmetic (ref. 7) is commercially available.
For completeness, an alternative description of the filter is given in terms of the system states \( q_1^i \) and \( q_2^i \) and clearly demonstrates the recursive nature of the filter. The set of difference equations describing the filter and required in developing a computer algorithm is presented. Thus, for the \( i \)th stage in figure 1 at the \( k \)th sample point

\[
w^i(k) = A_i u_1(k) - c_1 q_1^i(k) - d_1 q_2^i(k)
\]

\[
q_1^i(k + 1) = w^i(k)
\]

\[
q_2^i(k + 1) = q_1^i(k)
\]

\[
y_1(k) = w^i(k) + a_1 q_1^i(k) + b_1 q_2^i(k)
\]

where

\[
u_1(k) = y_{i-1}(k)
\]

is the input to the \( i \)th stage and is identical to the output of the \((i - 1)\) stage and

\[
A_i = \begin{cases} A & \text{(i = 1)} \\ 1 & \text{(i \neq 1)} \end{cases}
\]

The Synthesis Problem

The design problem considered in this paper can be stated as follows: When the magnitude and phase specifications \((M_k, \theta_k)\), respectively, at the \(k\)th fractional Nyquist frequencies \(f_{ik} = 2f_{s} k / f_s\) (where \(f_s\) is the sampling frequency in Hz) are known, determine the set of optimum parameters \( \bar{p}^* \) of an \(N\)-stage cascade filter having the form of equation (1) so that the resultant digital filter will have a minimum sum squared magnitude and phase error for all specified frequencies.

By constraining the filter topology, the optimum synthesis problem becomes one of parametric optimization with respect to a given criterion of fit. The composite criterion which can weight the magnitude and phase requirements independently and as functions of frequency is chosen as the inner product

\[
V(A, \bar{p}) = \sum_k \left< \bar{e}_k, \bar{W}_k \bar{e}_k \right> = \sum_k V_k
\]  

(2)
where
\[
\mathbf{e}_k = \begin{bmatrix} \mathbf{A} |H_k| - M_k \\ \phi_k - \theta_k \end{bmatrix} = \begin{bmatrix} \mathbf{E}_k^M \\ \mathbf{E}_k^\phi \end{bmatrix}
\]
is the error vector and
\[
\mathbf{W}_k = \begin{bmatrix} \mathbf{W}_k^M & 0 \\ 0 & \lambda \mathbf{W}_k^\phi \end{bmatrix}
\]
is the diagonal weighting matrix. Clearly, \( V(A, \mathbf{p}) \) is a nonlinear function of the parameter vector \( \mathbf{p} = (a_1, b_1, c_1, d_1, \ldots, a_N, b_N, c_N, d_N)^T \), which involves the \( 4N \) filter coefficients, and of the filter multiplier \( A \).

The Minimization Algorithm

Through formal differentiation of the criterion function (eq. (2)) with respect to the multiplier \( A \), the minimization procedure can be slightly simplified to that of finding the minimum of a reduced functional \( \tilde{V}(\mathbf{p}) = V(A^*, \mathbf{p}) \) involving only \( 4N \) parameters. Thus
\[
\frac{\partial V}{\partial A} = \sum_k \left\langle \frac{\partial \mathbf{e}_k}{\partial A}, \frac{\partial V_k}{\partial \mathbf{e}_k} \right\rangle = 2 \sum_k \begin{bmatrix} |H_k| \mathbf{W}_k^M & 0 \end{bmatrix} \mathbf{e}_k
\]
and \( \frac{\partial V}{\partial A} = 0 \) implies
\[
2 \sum_k |H_k| \mathbf{W}_k^M (A^* |H_k| - M_k) = 0
\]
or
\[
A^* = \frac{\sum_k |H_k| \mathbf{W}_k^M M_k}{\sum_k |H_k|^2 \mathbf{W}_k^M}
\]
(3)

An additional necessary condition for existence of an extremum is that the gradient vector be zero; thereby, the optimum parameter vector \( \mathbf{p}^* \) is obtained. From equation (2)
\[
\frac{\delta \hat{V}}{\delta p} = 2 \sum_k \langle \vec{j}_k, \vec{w}_k \vec{e}_k \rangle
\]  

(4)

where the \((4N \times 2)\) Jacobian \(\vec{j}_k\) is

\[
\vec{j}_k^T = \nabla_p \vec{e}_k = \left[ A^* \frac{\delta |H_k|}{\delta p} ; \frac{\delta \phi_k}{\delta p} \right]^T
\]  

(5)

Clearly, each element of the gradient vector is the sum of two weighted functions of the magnitude and phase error. By writing

\[
|H_k|^2 = H_k \bar{H}_k
\]

where \(\bar{H}_k\) is the conjugate of \(H_k\) evaluated at the fractional frequency \(\Omega_k\), it is readily shown (see ref. 6), where \(\bar{p}_i\) is the set of filter parameters for the \(i\)th stage, that

\[
\frac{\delta |H_k|}{\delta \bar{p}_i} = \frac{1}{|H_k|} R \left( \bar{H}_k \frac{\delta H_k}{\delta \bar{p}_i} \right)
\]

For the cascaded filter topology in terms of the elements of \(\bar{p}_i\),

\[
\frac{\delta |H_k|}{\delta a_i} = |H_k| R \left( \frac{z_k^{-1}}{N_k^i} \right)
\]

\[
\frac{\delta |H_k|}{\delta b_i} = |H_k| R \left( \frac{z_k^{-2}}{N_k^i} \right)
\]

\[
\frac{\delta |H_k|}{\delta c_i} = -|H_k| R \left( \frac{z_k^{-1}}{D_k^i} \right)
\]

and

\[
\frac{\delta |H_k|}{\delta d_i} = -|H_k| R \left( \frac{z_k^{-2}}{D_k^i} \right)
\]
where, with $z_k = e^{j\pi\Omega_k}$,

$$N_k = N_k(z_k) = 1 + a_k z_k^{-1} + b_k z_k^{-2}$$

and

$$D_k = D_k(z_k) = 1 + c_k z_k^{-1} + d_k z_k^{-2}$$

By letting

$$H_k = |H_k|e^{j\phi_k}$$

it follows that

$$\phi_k = I(\log_e H_k)$$

whence

$$\frac{\partial \phi_k}{\partial p} = I \left( \frac{\partial}{\partial p} \log_e H_k \right) = I \left( \frac{1}{H_k} \frac{\partial H_k}{\partial p} \right)$$

which takes on a particularly simple form for the cascade topology. For the $i$th stage parameters, in fact,

$$\frac{\partial \phi_k}{\partial a_i} = I \left( \frac{z_k^{-1}}{N_k} \right)$$

$$\frac{\partial \phi_k}{\partial b_i} = I \left( \frac{z_k^{-2}}{N_k} \right)$$

$$\frac{\partial \phi_k}{\partial c_i} = -I \left( \frac{z_k^{-1}}{D_k} \right)$$
and
\[
\frac{\partial \phi_k}{\partial d_1} = -I \left( \frac{z_{k-2}^{-1}}{D_k} \right)
\]

The special case of a one-stage (N = 1) filter is illustrated. Here
\[
H_k = A \frac{1 + az_k^{-1} + bz_k^{-2}}{1 + cz_k^{-1} + dz_k^{-2}}
\]

\[
\hat{V} = \sum_k \left( A^* |H_k| - M_k \right)^2 W_k^M + \lambda \sum_k (\phi_k - \theta_k)^2 W_k^\phi
\]

and
\[
\frac{\partial \hat{V}}{\partial a} = 2 \sum_k \left( E_k^M W_k^M \frac{\partial |H_k|}{\partial a} + \lambda E_k^\phi W_k^\phi \frac{\partial \phi_k}{\partial a} \right) = \sum_k \left[ Q_k^M R \left( \frac{z_{k-1}^{-1}}{N_k} \right) + \lambda R_k^\phi I \left( \frac{z_{k-1}^{-1}}{N_k} \right) \right]
\]

Similarly,
\[
\frac{\partial \hat{V}}{\partial b} = \sum_k \left[ Q_k^M R \left( \frac{z_{k-2}^{-1}}{N_k} \right) + \lambda R_k^\phi I \left( \frac{z_{k-2}^{-1}}{N_k} \right) \right]
\]

\[
\frac{\partial \hat{V}}{\partial c} = \sum_k \left[ Q_k^M R \left( \frac{z_{k-1}^{-1}}{D_k} \right) + \lambda R_k^\phi I \left( \frac{z_{k-1}^{-1}}{D_k} \right) \right]
\]

\[
\frac{\partial \hat{V}}{\partial d} = \sum_k \left[ Q_k^M R \left( \frac{z_{k-2}^{-1}}{D_k} \right) + \lambda R_k^\phi I \left( \frac{z_{k-2}^{-1}}{D_k} \right) \right]
\]

where
\[
Q_k^M = 2 E_k^M W_k^M |H_k|
\]
and

\[ R_k^\phi = 2E_k^\phi W_k^\phi \]

are the weighted errors. It is obvious that the frequency intervals of the input data (specifications) need not be uniform and may, in fact, be intentionally unequal to allow for nonuniform frequency weighting.

Complementary Root Reflection and Stability

In deriving the frequency response of a discrete operator by letting \( z_k \) lie on the unit circle \( \Gamma \), it is possible to take advantage of a unique property of the discrete transform pertaining to its magnitude when a root lying outside the unit circle is imaged or reflected into the unit circle. It is easy to show that the magnitude of a phasor \( z - z_0 \), where \( z_0 \) is a root of the discrete transform lying outside the unit circle, is equal to

\[ |z - z_0| = |z_0| |z - \frac{1}{z_0}| ; z \in \Gamma \]

Since \( z_0 \) has been assumed to be outside the unit circle, \( 1/z_0 \) must be inside, the term \( |z_0| \) correcting for magnitude changes. Thus, if in the optimization procedure a pole should stray outside the unit circle and thereby lead to an unstable filter, root reflection guarantees stability with no magnitude change. There is no analogous simple identity for the phase of a reflected root. Experience with the procedure has shown that provided the design requirements can be met by means of a stable filter, that is, that a feasible solution exists, an optimum will indeed be found through repeated application of root reflection.

The Computer Algorithm

A complete listing of the filter design algorithm, which is an adaptation of the program written by Steiglitz, is given in the appendix. The main program is termed STGZ3 which calls four principal subroutines: (1) FUNCT performs the functional and gradient computation for each iteration as well as putting out the final optimum parameters and plots, (2) FLPWL is a Fletcher-Powell conjugate gradient routine, (3) INSIDE computes root reflection, and (4) ROOTS determines the poles and zeros of the filter. Single-precision arithmetic has been employed.

When minimization of the functional has been attained in the first pass or the minimization algorithm has iterated 300 times, a test is made to ascertain that all the roots are within the unit circle, a necessary requirement for the poles for stability reasons and for the zeros to insure minimum phase. If the design should result in an unstable
configuration, the roots are reflected about the unit circle and minimization is resumed in a second pass. If a minimum does indeed exist and all the roots then lie within the unit circle, the program computes and prints out the frequency response and commences plotting.

Minimization is deemed to be achieved when the absolute difference in functionals between successive iterations $\epsilon = |\hat{V}_{\text{new}} - \hat{V}_{\text{old}}|$ or the norm of the gradient vector falls below preassigned limits. Convergence is generally fast for magnitude or phase filters but can be very slow for the case of compromise filters.

When the design specifications cannot be met after $LIM$ iterations (see appendix), the program will stop; this situation indicates that the optimum could not be found and the resultant characteristic which may be unusable is plotted. Generally, feasible designs have been determined in less than 2000 iterations.

Minimization of the criterion function does not guarantee determination of a global minimum but rather determination of a local minimum. Depending upon the parameter vector utilized for initialization of the algorithm computation, different minima may be achieved. Experience has shown that stage-by-stage optimization, that is, utilization of the $i$th-stage optimum parameter vector as the initial parameter vector for the $(i + 1)$ stage of an $N$-stage filter, yields lower minimum values of the criterion function than does single-pass optimization.

APPLICATIONS

Linear-Phase Filter

This example considers a digital filter having application as a phase discriminator with a linear phase characteristic and arbitrary magnitude characteristic and is shown in figure 2. In this example all magnitude weights were set to zero and all phase weights to unity, the multiplier $A$ being arbitrarily made unity since it has no effect on the phase characteristic.

The phase requirements were $\theta_k = 1 - 2\Omega_k \left(0 \leq \Omega_k \leq 1\right)$, and a two-stage filter was specified. When an initial parameter vector $\bar{p} = (0, 0, 0, 0.25, 0, 0, 0, 0)^T$ was used, the algorithm converged to the optimum, with $\epsilon = 10^{-4}$, in 52 iterations and a Control Data 6600 computer time of 14 seconds. The optimum parameter values computed were to four places

$$A = 1.0$$

$$a_1 = 0 \quad b_1 = -0.9871 \quad c_1 = 0 \quad d_1 = 0.0395$$

$$a_2 = 0 \quad b_2 = -0.9871 \quad c_2 = 0 \quad d_2 = -0.0127$$
It is interesting to note that the phase requirements were met to within $0.008 \pi$ radian for approximately 95 percent of the frequency range.

Constant-Phase Filters

Two cases were considered to obtain filters having constant phases of $-\pi/2$ and $\pi/2$ radians over a frequency range $0.3 \leq \Omega_k \leq 0.7$. As in the previous case, the form of the magnitude characteristic was of no concern; hence, zero magnitude weighting was specified. With the same initial parameter state used in the previous example, the first case (lag network) optimized in 1673 iterations and 42 seconds to yield a hyperbolic magnitude characteristic and phase errors of less than $0.0003 \pi$ radian throughout the specified band.

The computed parameters for the lag case were

$$A = 1.0$$

$$a_1 = 0.5580 \quad b_1 = -0.1857 \quad c_1 = -0.4752 \quad d_1 = 0.0363$$

$$a_2 = 0.5580 \quad b_2 = -0.1857 \quad c_2 = -0.3712 \quad d_2 = -0.5686$$

The positive phase filter (lead network), however, took only 165 iterations and 17 seconds to yield the desired phase characteristic with errors nowhere exceeding $0.001 \pi$ radian in the specified band.

The optimum filter parameters for this second case were determined to be

$$A = 1.0$$

$$a_1 = -0.4768 \quad b_1 = -0.1548 \quad c_1 = 0.5022 \quad d_1 = -0.1082$$

$$a_2 = -0.4768 \quad b_2 = -0.1548 \quad c_2 = 0.4515 \quad d_2 = -0.2008$$

It is noted that for both cases, the phase weights outside the specified band were set to zero, and thereby allowed for arbitrary phase in these regions. Figures 3(a) and 3(b) show the resultant frequency characteristics for the lag and lead cases, respectively, of two-stage filters. The combination of the two filters, although they have antagonistic magnitude characteristics, suggests the possibility of a phase-splitting digital network.
Limited-Band Constant-Gain Linear-Phase Filter

The third example demonstrates a compromise design of a digital filter having constant-magnitude and linear-phase characteristics, over a limited frequency band, typical of phase discriminators. Here, except for \( \lambda = 0 \), the specifications were stated as

\[
M_k = \begin{cases} 
1 & (0.3 \leq \Omega_k \leq 0.7) \\
0 & \text{(Elsewhere)}
\end{cases}
\]

\[
\theta_k = \begin{cases} 
1 - 2\Omega_k & (0.3 \leq \Omega_k \leq 0.7) \\
0 & \text{(Elsewhere)}
\end{cases}
\]

Equal error and frequency weights were employed and the effects of changes in \( \lambda \) are shown in figure 4 for a two-stage design. Figure 4(a) shows the case of \( \lambda = 0 \), that is, a magnitude-only filter being specified, and coincidentally yields the linear-phase-filter characteristic derived in the first example. (See fig. 2.) Figures 4(b) and 4(c) show the magnitude and phase characteristics for the cases of \( \lambda = 10 \) and \( \lambda = 1000 \), respectively. The increasing weight on phase and resultant degradation in the magnitude characteristic are shown. The optimum parameters were

\( \lambda = 0 \):  
\[
A = 0.2063 \\
a_1 = 0.0000 \quad b_1 = -1.0000 \quad c_1 = 0.0000 \quad d_1 = 0.1539 \\
a_2 = 0.0000 \quad b_2 = -1.0000 \quad c_2 = 0.0000 \quad d_2 = 0.1539
\]

\( \lambda = 10 \):  
\[
A = 0.3658 \\
a_1 = -0.9754 \quad b_1 = 0.7300 \quad c_1 = 0.4529 \quad d_1 = 0.7211 \\
a_2 = 0.8632 \quad b_2 = 0.5632 \quad c_2 = -0.6119 \quad d_2 = 0.7443
\]
\( \lambda = 1000: \)

\[
A = 0.4232 \\
a_1 = -1.1739 \quad b_1 = 0.8489 \quad c_1 = 0.7596 \quad d_1 = 0.6691 \\
a_2 = 1.1739 \quad b_2 = 0.8489 \quad c_2 = -0.7596 \quad d_2 = 0.6691
\]

**Low-Pass Zero-Phase Filter**

The fourth example considers a compromise filter, having two and three stages, with specifications that are intentionally conflicting. A filter described by

\[
M_k = \begin{cases} 
1.0 & (0.0 \leq \Omega_k < 0.5) \\
0.5 & (\Omega_k = 0.5) \\
0.0 & (0.5 < \Omega_k \leq 1.0)
\end{cases}
\]

\[
\phi_k = \begin{cases} 
0 & (0.0 \leq \Omega_k \leq 0.5) \\
\text{Unspecified} & (\text{Elsewhere})
\end{cases}
\]

is specified.

Figures 5 and 6 show the results for the two- and three-stage designs, respectively, with figures 5(a) and 6(a) showing the magnitude-only (\( \lambda = 0 \)) case. The degradation in the magnitude characteristics when greater emphasis is placed on the phase specifications is evident in figures 5(b) and 6(b) for \( \lambda = 10 \) and in figures 5(c) and 6(c) for \( \lambda = 1000 \). Comparison of figure 6 with figure 5 demonstrates the improvement brought about by increasing the number of stages. The optimum parameters for the two-stage filter were \( \lambda = 0: \)

\[
A = 0.1196 \\
a_1 = 1.0240 \quad b_1 = 1.0000 \quad c_1 = -0.1713 \quad d_1 = 0.7676 \\
a_2 = 1.0240 \quad b_2 = 1.0000 \quad c_2 = -0.5324 \quad d_2 = 0.2286
\]
$\lambda = 10$:

\[
A = 0.4879 \\
a_1 = 0.2018 \\nb_1 = 0.6684 \\
c_1 = 0.3560 \\nd_1 = 0.4612 \\
a_2 = 0.6597 \\nb_2 = 0.4335 \\
c_2 = 0.0806 \\nd_2 = 0.7671
\]

$\lambda = 1000$:

\[
A = 0.5343 \\
a_1 = 0.0205 \\nb_1 = 0.7169 \\
c_1 = -0.0836 \\nd_1 = 0.6255 \\
a_2 = 0.6286 \\nb_2 = 0.7905 \\
c_2 = 0.2123 \\nd_2 = 0.6681
\]

The optimum parameters for the three-stage filter were

$\lambda = 0$:

\[
A = 0.0510 \\
a_1 = 0.8537 \\nb_1 = 1.0000 \\
c_1 = -0.1068 \\nd_1 = 1.0000 \\
a_2 = 0.8537 \\nb_2 = 1.0000 \\
c_2 = -0.4046 \\nd_2 = 0.5990 \\
a_3 = 0.8537 \\nb_3 = 1.0000 \\
c_3 = -0.6799 \\nd_3 = 0.2069
\]

$\lambda = 10$:

\[
A = 0.5109 \\
a_1 = 1.3302 \\nb_1 = 0.5515 \\
c_1 = -0.1731 \\nd_1 = 0.8097 \\
a_2 = 0.6844 \\nb_2 = 0.7157 \\
c_2 = 1.1880 \\nd_2 = 0.5850 \\
a_3 = -0.0373 \\nb_3 = 0.7012 \\
c_3 = 0.3825 \\nd_3 = 0.5262
\]
\( \lambda = 1000: \)

\[
\begin{align*}
A &= 0.4515 \\
\alpha_1 &= 1.5107 \\
\beta_1 &= 0.5286 \\
\gamma_1 &= -0.1771 \\
\delta_1 &= 0.8972 \\
\alpha_2 &= 0.5825 \\
\beta_2 &= 0.7490 \\
\gamma_2 &= 1.3094 \\
\delta_2 &= 0.4191 \\
\alpha_3 &= -0.1663 \\
\beta_3 &= 0.7485 \\
\gamma_3 &= 0.2002 \\
\delta_3 &= 0.6393
\end{align*}
\]

A three-stage design of this example is used to demonstrate the existence of two distinct local minima, dependent upon the initial parameter vector. In the first case, a single-pass optimization was accomplished with \( \vec{p} = (0, 0, 0, 0.25, 0, 0, 0, 0)^T \) for the initial parameter vector and resulted in the optimum filter shown in figure 6(a). In the second case, a stage-by-stage optimization was accomplished by utilizing the optimum parameter vector from a two-stage design for the initial parameter vector of a three-stage design and resulted in the optimum filter shown in figure 7. Comparison of these results demonstrates the existence of two distinct local minima, the stage-by-stage minimization yielding superior results.

**CONCLUDING REMARKS**

A method has been developed for a computer-aided design of cascade canonical digital filters having prescribed magnitude or phase characteristics or a compromise between the two. The method, which uses an unconstrained minimization algorithm, allows for arbitrary error and frequency weighting. Representative designs of phase and compromise filters have demonstrated the utility of the technique. Although convergence is generally fast for magnitude phase filters, it may be slow for the case of compromise filters.

Langley Research Center,
National Aeronautics and Space Administration,
Hampton, Va., February 17, 1972.
APPENDIX

PROGRAM LISTING

This appendix contains a program listing written for the Control Data 6600 computer at the Langley Research Center, Hampton, Virginia, and is an adaptation of that written by Kenneth Steiglitz at Princeton University for the design of specified magnitude-only filters.

```
PROGRAM STG73
  INPUT, OUTPUT, TAPE5 = INPUT, TAPE6 = OUTPUT, PUNCH
  EXTERNAL FUNCTION
  DIMENSION H(141), X(141), G(141)
  COMMON/M(1001), Y(1001), PHASED(1001), ALAMDA, FR, WTM(1001),
  C WT(1001), XTXP
  COMMON/RAW1/CALL, KCALL, LINE
  CALL CALCMP
  CALL LFRY
  WRITE(*,-1)
  51 FORMAT(* INPUT DATA*)
  M=0
  000012 36 Y=M+1
  000014 FORMAT(2111(M), 1(Y), PHASED(M), WTM(M), XTXP(M))
  000031 21 FORMAT(5F10.5)
  000032 WRITE(6,22) M, WTM(M), PHASED(M), XTXP(M)
  000033 22 FORMAT(* M=13, WTM= F7.4, PHASED= F7.4, XTXP= F7.4)
  000051 IF (M .LT. 11) GOTO 30
  000054 DO 15 J=1,16
  000056 15 X(I,J)=0.00
  000062 X(1,J)=.25
  000078 FORMAT(5,5.5) L,LIN,FST, FPS, MAX, ALAMDA, FR, XTXP
  000094 60 FORMAT(215,5F10.5,17)
  000106 IF (FR .LT. .001) FR=1.
  000112 N4=4.
  000113 IF (FR .LT. 5) G00, PAA
  000117 488 CONTINUE
  000117 488 WRITE(*,511) L,LIN,FST, FPS, MAX, ALAMDA
  000141 61 FORMAT(* L= ,12, L= *15., EST= *, F10.5, FPS= *, F10.5,
  000158  MAX= *, F10.5, FREQUENCY= *, F10.5, LAMBDA= *, F10.5)
  000141 ICALL= 0
  000142 ICALL= 0
  000143 CALL LFRY(L, FST, FPS, MAX, ALAMDA)
  000155 CALL WRITE(N,X)
  000157 CALL INSIDE1(X,KFLAG)
  000167 WRITE(*,261) KFLAG, ICALL, KCALL
  000176 IF (ICALL .NE. 15, KFLAG= 15, ICALL= IF , KCALL= 15)
  000176 ICALL = 015
  000175 ICALL = 015
  000179 IF (KFLAG .LE. 6, CALL, L, IF , X) G01
  000179 CALL L (N, X)
  000185 ICALL = 10
  000194 CALL L, K, FST, FPS, MAX, ALAMDA
  000194 CALL WRITE(N,X)
  000203 CALL M, IF , X, F, G, HMAX
  000224 STOP
  000226 STOP
  000230 END
```
APPENDIX – Continued

```
003427  DC 4?? J=1.X
003429  J=1(J=1-1)4
003428
003429  E=.,#3V(T/I-I/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1/1...
APPENDIX - Continued

```plaintext
001721  IF( PHASE = YMAX3 ) 301, 3C1, 4C1
001726  4C1  YMAX3 = PHASE
001730  3C1  CONTINUE
001730  4C2  IF( PHASE = YMIN3 ) 402, 3C2, 3C2
001733  4C2  YMIN3 = PHASE
001735  3C2  CONTINUE
001735  4C3  IF( A1 = YMAX11 ) 3C3, 3CC, 4C0
001740  4C0  YMAX11 = A1
001742  3C0  CONTINUE

C SCALE COMPUTATIONS

001761  IF( HMAX, EQ, 0 ) GO TO 332
001764  YMAX = FLAT( YMAX11 )
001766  IF( HMAX, EQ, 0 ) GO TO 332
001767  YMAX = HMAX
001770  333  CONTINUE
001770  YMAX3 = YMAX3
001773  YMAX3 = FLAT( YMAX3 )
001775  YMIN3 = YMIN3
001780  AXM = AXM
001790  AYM = AYM
001800  APM = APM
001805  NFR = NFR
001810  AF = AF
001815  CALL INDFPLTL1, NFR, APM, APM, APM, APM, APM, APM, APM, APM, APM
001820  CALL INDFPLTL1, NFR, APM, APM, APM, APM, APM, APM, APM, APM, APM
001825  CALL INDFPLTL1, NFR, APM, APM, APM, APM, APM, APM, APM, APM, APM
001830  CALL INDFPLTL1, NFR, APM, APM, APM, APM, APM, APM, APM, APM, APM

C MAGNITUDE (COMPUTED) - FREQUENCY PLOT

001837  CALL INDFPLTL( NFR, 1, AMAG, 1, C, FR, C, YMAX, C, 5, -10, AXM, -10, AYM, 11, 1, C, AMAG, 1, C, FR, C, YMAX, C, 5, -10, AXM, -10, AYM, 11, 1) 0217
001840  CALL INDFPLTL( NFR, 1, AMAG, 1, C, FR, C, YMAX, C, 5, -10, AXM, -10, AYM, 11, 1) 0218

C MAGNITUDE (DESIRABLE) - FREQUENCY PLOT

001847  CALL INDFPLTL( NFR, 1, AMAG, 1, C, FR, C, YMAX, C, 5, -10, AXM, -10, AYM, 11, 1) 0220
001852  CALL INDFPLTL( NFR, 1, AMAG, 1, C, FR, C, YMAX, C, 5, -10, AXM, -10, AYM, 11, 1) 0221

C PHASE FREQUENCY PLOT

001859  CALL INDFPLTL( NFR, 1, APHASE, 1, C, PHASE, 1, C, 5, -10, AXM, -10, AYM, 11, 1) 0223
001864  CALL INDFPLTL( NFR, 1, APHASE, 1, C, PHASE, 1, C, 5, -10, AXM, -10, AYM, 11, 1) 0224

C PHASE (DISPERSION) - FREQUENCY PLOT

001871  CALL INDFPLTL( NFR, 1, APHASE, 1, C, PHASE, 1, C, 5, -10, AXM, -10, AYM, 11, 1) 0226
001876  CALL INDFPLTL( NFR, 1, APHASE, 1, C, PHASE, 1, C, 5, -10, AXM, -10, AYM, 11, 1) 0227
001883  CALL INDFPLTL( NFR, 1, APHASE, 1, C, PHASE, 1, C, 5, -10, AXM, -10, AYM, 11, 1) 0228
001888  CALL INDFPLTL( NFR, 1, APHASE, 1, C, PHASE, 1, C, 5, -10, AXM, -10, AYM, 11, 1) 0229
```

20
APPENDIX – Continued

00001 7  H(KL)=0.00  0246
00004 6  K=KL+1  0247
00007 5  K Tân.< Count+1  0248
00010 4  WRITE('K =',51) KCount  0249
00013 4  K = K-1  0250
00016 5  Ktan=Ktan+1  0251
00019 4  Ich=1  0252
00022 3  Ich=Ich+1  0253
00025 3  Ich=Ich+1  0254
00028 3  Ich=Ich+1  0255
00031 3  Ich=Ich+1  0256
00034 3  Ich=Ich+1  0257
00037 3  Ich=Ich+1  0258
00040 3  Ich=Ich+1  0259
00043 3  Ich=Ich+1  0260
00046 3  Ich=Ich+1  0261
00049 3  Ich=Ich+1  0262
00052 3  Ich=Ich+1  0263
00055 3  Ich=Ich+1  0264
00058 3  Ich=Ich+1  0265
00061 3  Ich=Ich+1  0266
00064 3  Ich=Ich+1  0267
00067 3  Ich=Ich+1  0268
00070 3  Ich=Ich+1  0269
00073 3  Ich=Ich+1  0270
00076 3  Ich=Ich+1  0271
00079 3  Ich=Ich+1  0272
00082 3  Ich=Ich+1  0273
00085 3  Ich=Ich+1  0274
00088 3  Ich=Ich+1  0275
00091 3  Ich=Ich+1  0276
00094 3  Ich=Ich+1  0277
00097 3  Ich=Ich+1  0278
00100 3  Ich=Ich+1  0279
00103 3  Ich=Ich+1  0280
00106 3  Ich=Ich+1  0281
00109 3  Ich=Ich+1  0282
00112 3  Ich=Ich+1  0283
00115 3  Ich=Ich+1  0284
00118 3  Ich=Ich+1  0285
00121 3  Ich=Ich+1  0286
00124 3  Ich=Ich+1  0287
00127 3  Ich=Ich+1  0288
00130 3  Ich=Ich+1  0289
00133 3  Ich=Ich+1  0290
00136 3  Ich=Ich+1  0291
00139 3  Ich=Ich+1  0292
00142 3  Ich=Ich+1  0293
00145 3  Ich=Ich+1  0294
00148 3  Ich=Ich+1  0295
00151 3  Ich=Ich+1  0296
00154 3  Ich=Ich+1  0297
00157 3  Ich=Ich+1  0298
00160 3  Ich=Ich+1  0299
00163 3  Ich=Ich+1  0300
00166 3  Ich=Ich+1  0301
00169 3  Ich=Ich+1  0302
00172 3  Ich=Ich+1  0303
00175 3  Ich=Ich+1  0304
00178 3  Ich=Ich+1  0305
00181 3  Ich=Ich+1  0306
00184 3  Ich=Ich+1  0307
00187 3  Ich=Ich+1  0308
00190 3  Ich=Ich+1  0309

Continued
APPENDIX – Continued

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>000327</td>
<td>DALFA=CALFA<em>DALFA-DX/ALFA</em>CY/ALFA</td>
<td>0310</td>
</tr>
<tr>
<td>000333</td>
<td>IF(CALFA) I=1,25,25</td>
<td>0311</td>
</tr>
<tr>
<td>000335</td>
<td>25 W=ALFA/ SQRT(CALFA)</td>
<td>0312</td>
</tr>
<tr>
<td>000340</td>
<td>ALFA= (DY+W-1)<em>AMBDA/(DY+2. COX</em>W-DX)</td>
<td>0313</td>
</tr>
<tr>
<td>000351</td>
<td>DO 26 I=1,N</td>
<td>0314</td>
</tr>
<tr>
<td>000356</td>
<td>26 XI=I(X(I)+T-ALFA)*H(I)</td>
<td>0315</td>
</tr>
<tr>
<td>000366</td>
<td>CALL FUNCT(N, X, F, G, H, FAX)</td>
<td>0320</td>
</tr>
<tr>
<td>000490</td>
<td>IF(KCALL.GT.300) GO TO 725</td>
<td>0321</td>
</tr>
<tr>
<td>000493</td>
<td>IF(KCALL.GT.LIM) GO TO 725</td>
<td>0322</td>
</tr>
<tr>
<td>000496</td>
<td>GO TO 919</td>
<td>0323</td>
</tr>
<tr>
<td>000497</td>
<td>725 IER=3</td>
<td>0324</td>
</tr>
<tr>
<td>000410</td>
<td>RFLPN</td>
<td>0325</td>
</tr>
<tr>
<td>000419</td>
<td>915 IF(F-FX) 27, 27.28</td>
<td>0326</td>
</tr>
<tr>
<td>00043</td>
<td>IF(F-FY) 36, 36.29</td>
<td>0327</td>
</tr>
<tr>
<td>000416</td>
<td>28 DALFA=0.00</td>
<td>0328</td>
</tr>
<tr>
<td>000417</td>
<td>DC 29 I=1,N</td>
<td>0329</td>
</tr>
<tr>
<td>000421</td>
<td>29 DALFA=DALFA+G(I)*H(I)</td>
<td>0330</td>
</tr>
<tr>
<td>000430</td>
<td>IF(CALFA) 30, 33, 33</td>
<td>0331</td>
</tr>
<tr>
<td>000431</td>
<td>30 IF(F-FX) 32, 32.33</td>
<td>0332</td>
</tr>
<tr>
<td>000444</td>
<td>31 IF(X-DALFA) 32, 36.37</td>
<td>0333</td>
</tr>
<tr>
<td>000436</td>
<td>32 FX=F</td>
<td>0334</td>
</tr>
<tr>
<td>000437</td>
<td>DX=CALFA</td>
<td>0335</td>
</tr>
<tr>
<td>000440</td>
<td>T=ALFA</td>
<td>0336</td>
</tr>
<tr>
<td>000442</td>
<td>AMBDA=ALFA</td>
<td>0337</td>
</tr>
<tr>
<td>000443</td>
<td>GN TO 23</td>
<td>0338</td>
</tr>
<tr>
<td>000444</td>
<td>33 IF(FY-F) 35, 34.35</td>
<td>0339</td>
</tr>
<tr>
<td>000445</td>
<td>34 IF(FY-DALFA) 35, 36.35</td>
<td>0340</td>
</tr>
<tr>
<td>000447</td>
<td>35 FY=F</td>
<td>0341</td>
</tr>
<tr>
<td>000450</td>
<td>DY=CALFA</td>
<td>0342</td>
</tr>
<tr>
<td>000451</td>
<td>AMBDA=AMBDA-ALFA</td>
<td>0343</td>
</tr>
<tr>
<td>000454</td>
<td>GO TO 22</td>
<td>0344</td>
</tr>
<tr>
<td>000454</td>
<td>36 DO 37 J=1,N</td>
<td>0345</td>
</tr>
<tr>
<td>000456</td>
<td>K=K+N</td>
<td>0346</td>
</tr>
<tr>
<td>000457</td>
<td>H(K)=G(IJ)-H(K)</td>
<td>0347</td>
</tr>
<tr>
<td>000463</td>
<td>K=K+N</td>
<td>0348</td>
</tr>
<tr>
<td>000464</td>
<td>37 H(K)=X(IJ)-H(K)</td>
<td>0349</td>
</tr>
<tr>
<td>000477</td>
<td>IF(MDF-F+FPS) 51, 38, 38</td>
<td>0350</td>
</tr>
<tr>
<td>000478</td>
<td>38 IFR=0</td>
<td>0351</td>
</tr>
<tr>
<td>000477</td>
<td>IF(KOUNT-N) 42, 39, 39</td>
<td>0352</td>
</tr>
<tr>
<td>000501</td>
<td>39 T=0.00</td>
<td>0353</td>
</tr>
<tr>
<td>000507</td>
<td>Z=0.00</td>
<td>0354</td>
</tr>
<tr>
<td>000503</td>
<td>40 DC 40 J=1,N</td>
<td>0355</td>
</tr>
<tr>
<td>000504</td>
<td>K=N+J</td>
<td>0356</td>
</tr>
<tr>
<td>000505</td>
<td>W=H(K)</td>
<td>0357</td>
</tr>
<tr>
<td>000506</td>
<td>K=K+N</td>
<td>0358</td>
</tr>
<tr>
<td>000511</td>
<td>T=T+l FPS(W(K))</td>
<td>0359</td>
</tr>
<tr>
<td>000514</td>
<td>40 T=2.4+H(K)</td>
<td>0360</td>
</tr>
<tr>
<td>000522</td>
<td>IF(HVR- FPS) 41, 41.42</td>
<td>0361</td>
</tr>
<tr>
<td>000526</td>
<td>41 IF(T FPS) 55, 56, 42</td>
<td>0362</td>
</tr>
<tr>
<td>000531</td>
<td>42 IF(KOUNT-LIMIT) 42, 45, 5C</td>
<td>0363</td>
</tr>
<tr>
<td>000534</td>
<td>43 ALFA=0.00</td>
<td>0364</td>
</tr>
<tr>
<td>000535</td>
<td>DO 47 J=1,N</td>
<td>0365</td>
</tr>
<tr>
<td>000537</td>
<td>K=J+N</td>
<td>0366</td>
</tr>
<tr>
<td>000540</td>
<td>W=C,00</td>
<td>0367</td>
</tr>
<tr>
<td>000542</td>
<td>DC 45 L=1,N</td>
<td>0368</td>
</tr>
<tr>
<td>000543</td>
<td>X=H+L</td>
<td>0369</td>
</tr>
<tr>
<td>000544</td>
<td>W=W+H(KL)*H(K)</td>
<td>0370</td>
</tr>
<tr>
<td>000547</td>
<td>IF(L-J) 44, 45, 45</td>
<td>0371</td>
</tr>
<tr>
<td>000554</td>
<td>44 K=K+N-L</td>
<td>0372</td>
</tr>
<tr>
<td>000557</td>
<td>45 GO TO 46</td>
<td>0373</td>
</tr>
<tr>
<td>000557</td>
<td>46 CONTINUE</td>
<td>0374</td>
</tr>
<tr>
<td>000565</td>
<td>K=N+J</td>
<td>0375</td>
</tr>
</tbody>
</table>
APPENDIX – Continued

SUBROUTINE INSIDE (N, X, KFLAG)
DIMENSION X(16)
J = 1
KFLAG = 0
10 J = J + 2
000006 IF (J .GT. N) RETURN
000007 E = -500 * X(J)
000008 C = X(J+1)
000009 DISC = E - B - C
000010 IF (DISC .LE. 0.00) GOTO 20

C......REAL ROOTS

000024 DISC = SQRT(DISC)
000025 R1 = B + DISC
000027 R2 = B - DISC
000031 R1 = ABS(R1)
000033 R2 = ABS(R2)
000035 IF (DR1.LE.1.00).AND.(DR2.LE.1.00) GOTO 10
000051 KFLAG = 1
000054 IF (DR1.GT.1.00).AND.(DR2.LE.1.00) GOTO 10
000060 X(J) = -1.00*(R1+R2)
000064 X(J+1) = R1*R2
000066 GOTO 10

C......COMPLEX ROOTS

20 IF (C.LT.1.00) GOTO 10
000071 KFLAG = 1
000074 X(J+1) = X(J) * C
000076 GOTO 10
000077 END
APPENDIX – Concluded

```plaintext
00005 SUBROUTINE ROOTS(N,X)
00010 DIMENSION X(16)
00013 WRITE(6,40)
00016   40 FORMAT(* ROOTS*/6X,*REAL*,11X,*IMAG*,11X,*REAL*,11X,*IMAG*)
00019 J=1
00023  10 J=J+2
00027   IF(J.GT.N)RETURN
00031   B=-.500*X(J)
00035   C=X(J+1)
00039 DISC=B*B-C
00043 IF(DISC.LE.0.00)GOTO20
00047   R=-.500*X(J)
00051   C=X(J+1)
00055 DISC=B*B-C
00059 IF(DISC.LE.0.00)GOTO20
00063 R=-.500*X(J)
00067 C=X(J+1)
00071 DISC=B*B-C
00075 WRITE(6,50)R1,R2
00079 50 FORMAT(* *F15.8*15X~F15.8*)
00083 GO TO 10
00087 END
```

```plaintext
C*****REAL ROOTS
000027 DISC = SORT(DISC)
000030 R1=R+DISC
000032 R2=R-DISC
000035 WRITE(6,30)R1,R2
000044 30 FORMAT(* *F15.8*15X~F15.8*)
000044 GO TO 10
000048 END
```

```plaintext
C*****COMPLEX ROOTS
000046 20 DISC= SORT(-1.00*DISC)
000052 DISCM=-1.00*DISC
000055 WRITE(6,50)B,DISC,B,DISCM
000070 50 FORMAT(* *F15.8*)
000070 GO TO 10
000072 END
```
REFERENCES


Figure 1.- Signal flow graph of cascaded digital filter.
\[ \theta_k = 1 - 2 \Omega_k \quad (0 \leq \Omega_k \leq 1) \]

\[ M_k = \text{Unspecified} \quad (0 \leq \Omega_k \leq 1) \]

Figure 2.- Two-stage linear-phase filter.
\[ \theta_k = \begin{cases} -\pi/2 & (0.3 \leq \Omega_k \leq 0.7) \\ \text{Unspecified} & \text{(elsewhere)} \end{cases} \]

\[ M_k = \text{Unspecified} & (0 \leq \Omega_k \leq 1) \]

Figure 3.- Two-stage constant-phase filters.

(a) Lag filter.
\[ \theta_k = \begin{cases} \pi/2 & (0.3 \leq \Omega_k \leq 0.7) \\ \text{Unspecified} & \text{elsewhere} \end{cases} \]

\[ M_k = \text{Unspecified} \quad (0 \leq \Omega_k \leq 1) \]

(b) Lead filter.

Figure 3.- Concluded.
\[
M_k = \begin{cases} 
1 & (0.3 \leq \Omega_k \leq 0.7) \\
0 & \text{(elsewhere)}
\end{cases}
\]

\[
\theta_k = \text{Unspecified} \quad (0 \leq \Omega_k \leq 1)
\]

(a) Unspecified phase filter. \( \lambda = 0 \).

Figure 4. - Two-stage limited-band constant-gain filters.
$$M_k = \begin{cases} 1 & (0.3 \leq \Omega_k \leq 0.7) \\ 0 & \text{(elsewhere)} \end{cases}$$

$$\theta_k = \begin{cases} 1-2\Omega_k & (0.3 \leq \Omega_k \leq 0.7) \\ 0 & \text{(elsewhere)} \end{cases}$$

(b) Linear-phase filter. $\lambda = 10$.

Figure 4.- Continued.
\[ M_k = \begin{cases} 1 & (0.3 \leq \Omega_k \leq 0.7) \\ 0 & \text{(elsewhere)} \end{cases} \]

\[ \theta_k = \begin{cases} 1 - 2\Omega_k & (0.3 \leq \Omega_k \leq 0.7) \\ 0 & \text{(elsewhere)} \end{cases} \]

(c) Linear-phase filter. \( \lambda = 1000 \).

Figure 4.- Concluded.
\( \theta_k = \text{Unspecified} \quad (0 \leq \Omega_k \leq 1) \)

\[
M_k = \begin{cases} 
1.0 & (0.0 \leq \Omega_k < 0.5) \\
0.5 & (\Omega_k = 0.5) \\
0.0 & (0.5 < \Omega_k \leq 1.0) 
\end{cases}
\]

(a) Unspecified-phase filter. \( \lambda = 0 \).

Figure 5.- Two-stage low-pass filters.
\[
\theta_k = \begin{cases} 
0 & (0.0 \leq \Omega_k \leq 0.5) \\
\text{Unspecified} & \text{(elsewhere)} 
\end{cases}
\]

\[
M_k = \begin{cases} 
1.0 & (0.0 \leq \Omega_k < 0.5) \\
0.5 & (\Omega_k = 0.5) \\
0.0 & (0.5 < \Omega_k \leq 1.0) 
\end{cases}
\]

(b) Zero-phase filter. \( \lambda = 10 \).

Figure 5.- Continued.
\[ \theta_k = \begin{cases} 0 & (0.0 \leq \Omega_k \leq 0.5) \\ \text{Unspecified} & \text{(elsewhere)} \end{cases} \]

\[ M_k = \begin{cases} 1.0 & (0.0 \leq \Omega_k < 0.5) \\ 0.5 & \Omega_k = 0.5 \\ 0.0 & (0.5 < \Omega_k \leq 1.0) \end{cases} \]

(c) Zero-phase filter. \( \lambda = 1000. \)

Figure 5. - Concluded.
\[ \theta_k = \text{Unspecified} \quad (0 \leq \Omega_k \leq 1) \]

\[ M_k = \begin{cases} 1.0 & (0.0 \leq \Omega_k < 0.5) \\ 0.5 & (\Omega_k = 0.5) \\ 0.0 & (0.5 < \Omega_k \leq 1.0) \end{cases} \]

(a) Unspecified-phase filter. \( \lambda = 0 \).

Figure 6.- Three-stage low-pass filters.
\[
\theta_k = \begin{cases} 
0 & (0.0 \leq \Omega_k \leq 0.5) \\
\text{Unspecified} & (\text{elsewhere})
\end{cases}
\]

\[
M_k = \begin{cases} 
1.0 & (0.0 \leq \Omega_k < 0.5) \\
0.5 & (\Omega_k = 0.5) \\
0.0 & (0.5 < \Omega_k \leq 1.0)
\end{cases}
\]

Figure 6. - Continued.

(b) Zero-phase filter. \( \lambda = 10 \).
\[
\theta_k = \begin{cases} 
0 & (0.0 \leq \Omega_k \leq 0.5) \\
\text{Unspecified} & \text{(elsewhere)} 
\end{cases}
\]

\[
M_k = \begin{cases} 
1.0 & (0.0 \leq \Omega_k < 0.5) \\
0.5 & (\Omega_k = 0.5) \\
0.0 & (0.5 < \Omega_k \leq 1.0) 
\end{cases}
\]

(c) Zero-phase filter. \( \lambda = 1000. \)

Figure 6. - Concluded.
\[ \theta_k = \text{Unspecified} \quad (0 \leq \Omega_k \leq 1) \]

\[ M_k = \begin{cases} 
1.0 & (0.0 \leq \Omega_k < 0.5) \\
0.5 & (\Omega_k = 0.5) \\
0.0 & (0.5 < \Omega_k \leq 1.0) 
\end{cases} \]

Figure 7. - Three-stage low-pass filter.
"The aeronautical and space activities of the United States shall be conducted so as to contribute ... to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—National Aeronautics and Space Act of 1958

**NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS**

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

**SCIENTIFIC AND TECHNICAL INFORMATION OFFICE**

**NATIONAL AERONAUTICS AND SPACE ADMINISTRATION**

Washington, D.C. 20546