NASA SP-3049

# COMPUTER-AIDED FILTER DESIGN MANUAL



GUSSOW and WEATHERS



NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

NASA SP-3049

# COMPUTER-AIDED FILTER DESIGN MANUAL

By Sidney Gussow and Glenn Weathers

Prepared under contract for NASA
by Sperry Rand Corporation, Huntsville Alabama

*Scientific and Technical Information Division*
OFFICE OF TECHNOLOGY UTILIZATION 1969
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
*Washington, D.C.*

For sale by the Superintendent of Documents
U.S. Government Printing Office, Washington, D.C. 20402
Price $1.25
*Library of Congress Catalog Card Number 75–601301*

# Preface

This manual presents a computerized filter design procedure that provides both the experienced and inexperienced filter designer with a comprehensive filter design capability. The design procedure covers selection, analysis, and synthesis of electric-wave filters for telemetry applications. The manual was developed and prepared by the Space Support Division of Sperry Rand Corp., Huntsville, Ala., under NASA contract number NAS8–20055 for the Instrumentation and Communication Division of the Astrionics Laboratory at George C. Marshall Space Flight Center, Huntsville, Ala.

# Table of Contents

# List of Computer Programs

All the main-line computer programs utilized in this document are listed below along with the section number in which they are described.



A brief description of the purpose of each of these computer programs is given below:

ML1A — Frequency Analysis — Program ML1A calculates the frequency response curves of the filter's voltage transfer ratio. The response curves can be calculated over any desired frequency range and include the filter's amplitude response, phase response, group delay response, and time delay response. These responses are the curves most commonly used for describing a filter.

ML1B — Transient Analysis — Program ML1B calculates the output time response of a filter excited by a unit step sinusoidal signal. A typical use for the program would be to determine the time required for the transient effect on the filter to disappear.

ML2 — Darlington Synthesis — Darlington synthesis allows the realization of a filter with a doubly terminated ladder network. That is, the program calculates the values of the circuit components required to build the specified filter.

ML3 — Cauer Synthesis — Program ML3 is similar to ML2 in that it calculates the actual circuit components required to realize the filter. The difference is that Cauer synthesis realizes the filter for a singly terminated ladder network.

ML4 — Transformation Calculation — Program ML4 calculates data which assist in the use of the filter selection charts. The program transforms filter specifications from requirements on a given filter to requirements on its lowpass equivalent.

ML5 — Chart Plotter — Program ML5 plots the filter selection charts for given attenuation levels. A set of these charts is included in the filter design manual. However, for some application where the given charts are not extensive enough, program ML5 can be used to produce additional charts.

ML6 — Active Filter Design — Program ML6 supplies a set of acceptable amplifier parameters and calculates all the component values required to build the filter with active networks. The program also calculates the frequency response curves of networks.

ML7 — Active Filter Synthesis — Program ML7 allows the designer to specify the amplifier parameters and then calculates the remaining component values required to build the filter with active networks. The filter response curves are also calculated. Both programs ML6 and ML7 supply limiting values for the amplifier parameters which assists in selection of the amplifier used with the active networks.

ML8 — Temperature Dependent Filter Response — Program ML8 calculates the active network response curves of amplitude, phase, group delay, and time delay for any temperature value specified by the operator.

SECTION 1

# Introduction

There is a need in communications and telemetry for a simplified, accurate method of filter design and synthesis. To satisfy this need, a computerized procedure has been developed by the Space Support Division of Sperry Rand under the sponsorship of the National Aeronautics and Space Administration. A main objective of the design procedure was to provide engineers having a minimum of filter design experience with a comprehensive filter design capability.

The computerized filter design technique presented here offers advantages and capabilities beyond those of conventional computer-aided design programs. The design procedure provides a designer totally inexperienced in filter design with the capability of accurate filter design or of analyzing an existing design. The program also offers the experienced designer the normal computer-aided design advantages of performing laborious and complex calculations.

The filter design procedure consists of selection, analysis, and synthesis. Computer programs have been developed to assist the designer in each of these areas. The filter functions considered are the Butterworth and Chebyshev in lowpass, highpass, bandpass, and bandstop configurations. Most filters for telemetry applications are restricted to one of these filter types because of phase requirements.

The filter selection procedure includes the use of charts that assist in the rapid selection of a filter configuration. The charts specify the type, order, and ripple factor of lowpass filters meeting the specified amplitude and group delay requirements. Computer-supplied format transformations allow the designer to specify the filter performance requirements in terms of lowpass parameters; by utilizing the charts, he can select the optimum filter for his application.

The synthesis programs calculate the component circuit values required for the specified filter. Synthesis procedures are provided for realizing the filter in either passive or active form. The synthesis programs include the ability to design with dissipative components. The designer need only specify the $Q$ factors of the circuit elements he plans to use, and the program will predistort automatically the pole-zero plot of the filter such that synthesis with the dissipative elements will yield the proper response.

The passive synthesis programs use Cauer realization techniques for singly loaded networks and Darlington realization techniques for doubly loaded networks. Other programs are provided to assist the designer in tuning the filter.

Active synthesis is accomplished through the use of active quadratic networks that are connected in cascade. Active synthesis design techniques were developed for this design procedure that are far more accurate than are obtainable with conventional active designs. Accurate mathematical models were derived that more accurately describe the active filter networks than the simplified ideal models that are conventionally used.

The active design cannot be used accurately for narrow bandwidth requirements or for a frequency much beyond 500 kHz. These limitations are expounded upon in the active synthesis section. Within its usable limits, the active design can be used conveniently. The advantages of active designs over passive designs include: elimination of inductive components, ease of circuit assembly because of elimination of complex tuning procedures, and the ability to fabricate the filter in integrated circuit form.

The analysis programs calculate filter response characteristics including amplitude response, phase response, time delay response, and group delay response. The response curves can be calculated for the theoretical filter function and for the transfer function of the networks used to approximate the filter function. For active designs, the response curves can be obtained for any desired temperature value.

The design procedure also includes the ability to determine the transient response of the filter to a sinusoidal step input.

The filter analysis and synthesis programs are intended to relieve the filter designer of the laborious task of numerical calculation involved in obtaining response characteristics and circuit parameters for the filter design. Because the object of the design procedure is to give engineers with a minimum of background in filter design a capability in this area, the programs are written in such a way that a minimum amount of information is required to be read into the computer. All the computer programs and subroutines are written in FORTRAN IV for the IBM 1130 computer but can be adapted easily to any general-purpose digital computer using FORTRAN IV or FORTRAN V.

## SECTION 2

# General Filter Theory

This section contains a brief review of the filter theory pertaining to the use of the filter design programs presented in this document. The purpose is to provide the reader with a basic knowledge of the design concepts and terminology used so as to enable him to better understand and utilize the design programs. A large portion of the design theory is presented in the sections in which it is applicable. Only the theory that is applicable to the majority of the sections is presented here.

No attempt has been made to cover adequately the theory and techniques of general filter design. This material can be found easily in numerous texts if desired but is not necessary for use of the design procedures presented here.

### 2.1 DEFINITIONS OF FILTER TERMS

The terms commonly used throughout this handbook and in general filter design are defined as follows:

*Passband*—The frequency range in which the filter is intended to pass signals; in the passband, the attenuation is minimized

*Stopband*—The frequency range in which it is intended to reject or attenuate all signals as much as possible

*Cutoff frequency*—The frequency at which the signal amplitude is attenuated 3 db from its passband value

*Quadratic or second-order function*—Function of the form $S^2+2\zeta\omega_n S+\omega_n^2$, where $S=j\omega$, $\zeta$ is the damping ratio, and $\omega_n$ is the natural frequency; the normalized quadratic equation has the form $S^2+2\zeta S+1$, where $S=j\ (\omega/\omega_n)$. For values of the damping ratio below 0.5, the reciprocal of the function has a peak magnitude near its natural frequency

*Voltage transfer ratio*—The function describing the ratio of the output voltage to the input voltage of a filter network (also called the voltage transfer function); the function has the general form $E_{\text{out}}/E_{\text{in}}=(GS^2+DS+E)/(AS^2+BS+C)$

*Amplitude roll-off*—The increase in attenuation or the decrease in magnitude of the voltage transfer ratio as the frequency moves through the bandstop range away from the passband range

*Bandwidth*—The width of the passband between the 3-db cutoff frequencies of a bandpass filter; this term is also used in describing lowpass filters where it is the frequency range between zero frequency and the cutoff frequency or simply the cutoff frequency

*Center frequency*—The geometric mean frequency between the 3-db cutoff frequencies of a bandpass or bandstop filter; this equals the square root of the product of the two cutoff frequencies

*Lowpass filter*—A filter that passes low frequencies and rejects high frequencies

*Highpass filter*—A filter that passes high frequencies and rejects low frequencies

*Bandpass filter*—A filter that passes a defined band of frequencies and rejects both higher and lower frequencies

*Bandstop filter*—A filter that rejects a defined band of frequencies and passes both higher and lower frequencies

*Q factor*—Term used to specify the degree of dissipation associated with capacitive and inductive components; mathematically it is defined as the ratio of real-to-reactive impedance for the capacitor ($Q_c=R/X_c=\omega RC$) and the ratio of reactive-to-real impedance for the inductor ($Q_L=X_L/R=\omega_L/R$). Also the ratio of the center frequency to the bandwidth for a bandpass or bandstop filter.

*Filter responses*—The frequency responses of the filter voltage transfer ratio that includes: amplitude, phase, time delay, and group delay response. The amplitude response concerns the magnitude of the voltage transfer ratio, and the phase response gives the phase angle. The time delay is the delay that a sinusoidal signal encounters in passing through the filter and is defined as the phase angle divided by the negative of the signal frequency. The group delay is the relative delay of the envelope components of a modulated signal passing through the filter and is defined as the negative of the derivative of the phase angle with respect to the signal frequency. The group delay is sometimes called the envelope delay.

## 2.2 BUTTERWORTH AND CHEBYSHEV FILTERS

The two main filter functions considered are the Butterworth and Chebyshev filters. These are the most frequently used filters for telemetry applications because of their balanced amplitude-phase relationship. The Butterworth and Chebyshev filters are approximations to the so-called ideal "brick-wall" magnitude response, which is defined as being constant within the passband frequency range and zero over the rest of the frequency range. The higher the order of either filter, the more closely it approximates the ideal response. The general shape of the magnitude response for these two filters is shown in figure 2–1.

The Butterworth filter is characterized by a maximally flat response in the passband and monotonically increasing attenuation in the stopband. The form of the Butterworth transfer function is

$$G_{12}(j\omega)=\frac{1}{\sqrt{1+\omega^{2n}}}$$

The slope of the amplitude response in the stopband approaches 6$n$ db per octave, where $n$ is the order of the filter. The response is down 3 db at the cutoff frequency for all order filters.

The poles of the normalized lowpass Butterworth filter are uniformly spaced along a semicircle in the complex frequency plane. The pole locations for the normalized lowpass Butterworth are given by $S=-\sigma_\kappa \pm j\omega_\kappa$, where

$$\sigma_\kappa=\cos\frac{(2K-1+n)\pi}{2n}$$

$$\omega_\kappa=\sin\frac{(2K-1+n)\pi}{2n}$$

$$K=1, 2, \ldots, 2n,$$

where $n$ = order of the filter.

Rejecting the right half plane poles, the left half plane poles form the filter response function.

The Chebyshev filter is characterized by an equal magnitude ripple in the passband and monotonically increasing attenuation in the stopband. The number of the maxima and minima ripple peaks in the passband is equal to the order of the filter.

The form of the Chebyshev transfer function is

$$G_{12}(j\omega)=\frac{1}{\sqrt{1+\epsilon^2 Tn^2(\omega)}}$$

where $\epsilon$ is a real constant, $\epsilon<1$, and $Tn(\omega)$ is the

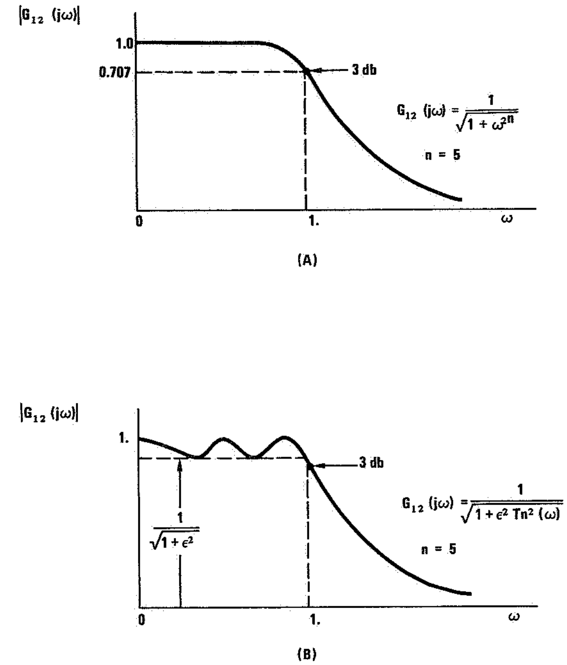



FIGURE 2–1.—Normalized lowpass magnitude response curves for (A) Butterworth and (B) Chebyshev filters.

$n^{th}$ order Chebyshev polynomial defined as $Tn(\omega)=\cos(n \cos^{-1} Z)$, where $Z$ is a real variable $Z \leq 1$. The distance between the maxima and minima ripple in the passband is $1-1(1+\epsilon^2)^{-\frac{1}{2}}$ and is approximated for small $\epsilon$ by $\epsilon^2/2$. As the ripple and order of the filter are increased, the rate of attenuation in the stopband increases.

The Chebyshev filter has a sharper roll-off, or squarer amplitude response, than the Butterworth of the same order but has less desirable phase and group delay characteristics. The poles of the Chebyshev filter are located on an ellipse in the complex frequency plane. The pole locations for the normalized lowpass Chebyshev are given by $S=\sigma_\kappa \pm j\omega_\kappa$, where

$$\sigma_\kappa = \sin h\, a \sin \frac{(2K-1)\pi}{2n}$$

$$\omega_\kappa = \cos h\, a \cos \frac{(2K-1)\pi}{2n}$$

$$K=1, 2, \ldots, 2n,$$

where

$a=1/n \sin h^{-1} 1/\epsilon$
$n$=order of the filter.

The magnitude of the Chebyshev filter at the cutoff frequency is defined by the passband ripple value, which means that a filter with $\pm 1$ db ripple will be 1 db down at the cutoff frequency. The cutoff frequency is not equal to the 3-db frequency unless the ripple is 3 db. The Butterworth response is always 3 db down at the cutoff frequency.

To make the design of these two filters more compatible, the poles of the Chebyshev filter are normalized to make the magnitude response 3 db down at the cutoff frequency for all values of passband ripple. The poles are normalized by dividing $\sigma_\kappa$ and $\omega_\kappa$ by the factor $\chi_0$, where $\chi_0 = \cos h\, [(1/n) \cos h^{-1}(1/\epsilon)]$.

For all design procedures used in this document, the Chebyshev response is 3 db down at its cutoff frequency.

After obtaining the normalized lowpass response function from the pole plot, the function can be transformed into an unnormalized lowpass, highpass, bandpass, or bandstop function. The transformation is accomplished by replacing the $S$ operator of the normalized lowpass with one of the following operators:

$$\frac{S}{\omega_c} \text{ Lowpass,}$$

$$\frac{\omega_c}{S} \text{ Highpass,}$$

$$\frac{\omega_0}{BW}\left(\frac{S}{\omega_0}+\frac{\omega_0}{S}\right) \text{ Bandpass,}$$

$$\frac{BW}{\omega_0}\left(\frac{1}{\frac{S}{\omega_0}+\frac{\omega_0}{S}}\right) \text{ Bandstop,}$$

where
$\omega_c$ =3-db cutoff frequency,
$\omega_0$ =center frequency,
$BW$=bandwidth,
$S$ =complex operator.

The transformations preserve the nature of the amplitude versus frequency characteristics both in the passband and in the stopband. For example, when a lowpass filter is transformed into a bandpass filter, the attenuation bandwidth ratios are unchanged. However, the transformed highpass, bandpass, and bandstop transient responses bear no obvious relationship to that of the corresponding lowpass filter.

# SECTION 3

# Filter Selection Procedure

## 3.1 INTRODUCTION

This section provides charts and instructions describing a filter selection procedure developed for Chebyshev and Butterworth approximation-type filters. The so-called "brick-wall" amplitude response characteristic can be approximated to any desired accuracy by using a polynomial of sufficient order. This is the basis of the approximation problem as related to filter design. The rapid selection of the type and order of filter required to meet a desired level of approximation to the brick-wall response is an advantage of this design procedure.

The filter data or parameters assumed to be specified in a given application are as follows:

(1) Maximum passband ripple,

(2) Center frequency, bandwidth, or cutoff frequencies,

(3) Attenuation levels at specified frequencies,

(4) Maximum group delay allowable across the passband.

## 3.2 INSTRUCTIONS

The following procedure can be used to select the optimum Butterworth or Chebyshev filter for a given filtering problem.

(1) The foregoing parameters should be selected or specified for the filter to be designed. It is not necessary to have all of the foregoing parameters to define a filter. For example, a filter may be defined without any group delay specifications. This parameter may be carried through the selection procedure, however, if it has been specified.

(2) Program ML4 should be used to transform the specified data to equivalent frequency data as used in the charts (refer to section 3.4). The output of program ML4 includes the equivalent lowpass frequency for a given frequency of interest. For example, if a bandpass filter with a center frequency of 90 kHz were being designed and it were of interest to know the equivalent lowpass frequency for 120 kHz when the bandwidth is 40 kHz, the program would calculate 1.313. The program also would calculate a multiplication factor that could be used to estimate the group delay across the filter and the variation in group delay across the passband of the filter.

(3) Filter selection may be achieved by using the equivalent lowpass frequency and maximum passband ripple to define a group of filters that will meet a set of specifications. If several attenuation-frequency requirements are specified, program ML4 should be run for each. The filter charts are used to select a design that appears in the "region of acceptable configuration" for each requirement.

When the maximum ripple and attenuation at a definite frequency are specified, program ML4 is used to find the equivalent lowpass frequency. On the chart for the specific attenuation of interest, a vertical line should be drawn through the equivalent frequency value. Through the maximum ripple a horizontal line is drawn, intersecting the previous vertical line. The rectangle formed by these two lines and the chart axis is the region of acceptable configuration for the given requirements. A filter can be selected from this group. Usually the lowest-order configuration that lies within the rectangle is the optimum filter selection. When this procedure for all specified frequency-attenuation pairs is repeated, the filter selected must be within the acceptable region for each case.

(4) To find an approximation of the variation in group delay across a filter, the delay charts are used in conjunction with the multiplication factor calculated in ML4. The charts give the equivalent lowpass group delays at equivalent frequencies of 1 and 0. The delay at bandedge is found by running ML4 for the filter cutoff frequency and multiplying the calculated "multiplication factor" by the values of delay given by figure 7–27. The delay at band center may be calculated by running ML4 for that frequency and multiplying this calculated multiplication factor by the value for the specific filter from figure 7–26.

Referencing the filter selection charts, a complete set of filters can be shown to fall within the region of acceptable configuration. The final selection is left to the designer. Usually, the optimum filter will be the one of lowest order within the acceptable set because lower-order filters require fewer sections and are easier to tune.

## 3.3 EXAMPLE OF DELAY CALCULATIONS

Consider a bandpass filter with its center frequency at 230 kHz and cutoff frequencies at 190 kHz and 278 kHz. To find the group delay at 230, 190, and 278 kHz, program ML4 is run for each of these frequency values, and charts for delay at 0, 1, and 1 are used, respectively. The equations are

The variation in delay across the lower passband is then the difference in the first two quantities calculated. The variation in delay across the upper passband is the difference in the first and last.

Figure 7–28 supplements the charts; it gives the intersection of the family of curves with the abscissa and corresponds to the Butterworth filter.

Programs ML5A and ML5B provide additional filter selection charts. These programs must be run on a computer with a plotter and stored plotter subroutines.

## 3.4 TRANSFORMATION CALCULATION PROGRAM ML4

### Purpose

The purpose of ML4 is to transform filter specifications from requirements for the given filter to requirements for its lowpass prototype. This allows the design charts to be used for selecting the optimum filter for a given application.

This program allows two forms of input data. They are frequency of interest-center frequency-bandwidth and frequency of interest-upper cutoff frequency-lower cutoff frequency. The second form can be used only in the case of a bandpass or bandstop filter. The output of this program includes

(1) The equivalent frequency, the number to be used in the filter selection charts.

(2) A multiplier factor used to estimate the variation in group delay over the passband (or stopband) of the filter.

$$\text{Delay (230 kHz)} = \begin{pmatrix} \text{Multiplication} \\ \text{factor for 230 kHz} \\ \end{pmatrix} \times \begin{pmatrix} \text{Value from} \\ \text{“delay at 0”} \\ \text{chart} \end{pmatrix}$$

$$\text{Delay (190 kHz)} = \begin{pmatrix} \text{Multiplication} \\ \text{factor for 190 kHz} \\ \end{pmatrix} \times \begin{pmatrix} \text{Value from} \\ \text{“delay at 1”} \\ \text{chart} \end{pmatrix}$$

and

$$\text{Delay (278 kHz)} = \begin{pmatrix} \text{Multiplication} \\ \text{factor for 278 kHz} \\ \end{pmatrix} \times \begin{pmatrix} \text{Value from} \\ \text{“delay at 1”} \\ \text{chart} \end{pmatrix}$$

### Operation

To use ML4, the input data should be punched on three cards with the format given in table I.

TABLE I.—*ML4 Input Data Format*

| Card | Parameter | Column |
|---|---|---|
| No. 1 | Frequency of interest (Hz) | 1–10 |
| No. 2 | Data factor | 1–10 |
| No. 3 | $Q$ factor | 1–10 |
| No. 4 | Configuration factor | 1–10 |
| Format 1 | Center frequency (Hz) | 11–20 |
| | Bandwidth (Hz) | 21–30 |
| | or | |
| No. 4 | Configuration factor | 1–10 |
| Format 2 | Upper cutoff frequency (Hz) | 11–20 |
| | Lower cutoff frequency (Hz) | 21–30 |

The data factor controls whether the first or second format shown should be used for card 4. All input data are floating-point numbers.

$$\text{Data factor}=\begin{cases}<0 & \text{Specify Format 1}\\ >0 & \text{Specify Format 2}\end{cases}$$

The configuration factor is defined as follows:

$$\text{Configuration factor}=\begin{cases}\text{1. Lowpass}\\ \text{2. Highpass}\\ \text{3. Bandpass}\\ \text{4. Bandstop}\end{cases}$$

The output of this program prints the type of filter being considered, the equivalent lowpass frequency of the frequency of interest, and a multiplier factor used in conjunction with the group delay charts to estimate variation in group delay.

### Description

Table II gives the transformation equations used by ML4. ML4 calls no subroutines.

TABLE II.—*ML4 Transformation Equations*

| Configuration | Frequency transformation | Multiplier factor |
|---|---|---|
| Lowpass | $S/\omega_0$ | $1/\omega_0$ |
| Highpass | $\omega_0/S$ | $\omega_0/S^2$ |
| Bandpass | $\omega_0/B(S/\omega_0+\omega_0/S)$ | $\omega_0/B(1/\omega_0-\omega_0/S^2)$ |
| Bandstop | $B/[\omega\phi(S/\omega_0+\omega_0/S)]$ | $-(1/\omega_0-\omega_0/S^2)/[\omega_0/B(S/\omega_0+\omega_0/S)^2]$ |

$\omega\phi$=Center frequency
$B$=Bandwidth
$S$=Frequency being transformed

Figure 3–1 gives an example of the output of ML4. The data should be added to the program as shown in figure 3–2.

## 3.5 CHART PLOTTER PROGRAM ML5A AND ML5B

### Purpose

The purpose of ML5A and ML5B is to plot

```
FEATURES SUPPORTED
 ONE WORD INTEGERS
 IOCS

CORE REQUIREMENTS FOR
 COMMON        0  VARIABLES      32  PROGRAM      500

END OF COMPILATION


// XEQ

    BAND PASS FILTER, CENTER FREQUENCY 0.100000E 02               BANDWIDTH 0.500000E 01

    EQUIVALENT FREQUENCY  0.300000E 01           MULTIPLIER  0.157079E 01

    FREQUENCY OF INTEREST         0.200000E 02
```

FIGURE 3–1.—Example of output of ML4

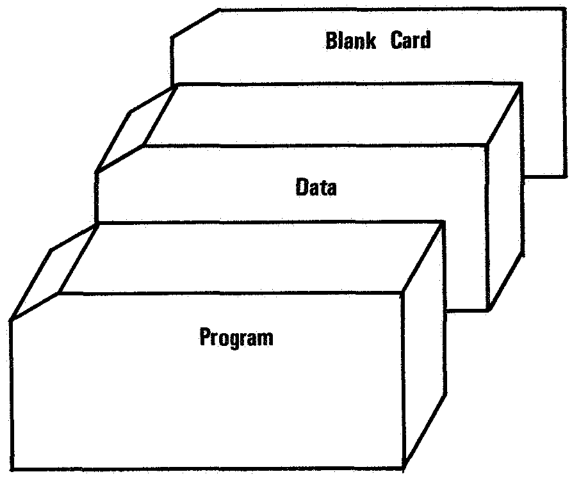



FIGURE 3-2.—Arrangement of main line program and data

the design charts shown in section 7. In some applications, the charts may not be extensive enough to allow for convenient filter design. In such cases, ML5A and ML5B can be used to generate additional charts for particular design problems. For example, if it were specified that the attenuation of a filter must be greater than 90 db above a given frequency, this program could then be used to generate charts for that attenuation level. The two programs are used as follows:

ML5A—High ripple levels (0.1 to 3 db)
ML5B—Low ripple levels (0 to 0.1 db)

The input data to this program include scale information for the plotter and the value of the attenuation for which the chart is being generated. The output is the design chart. This program uses the plotter facility and cannot be used unless a plotter is available. However, the program could be modified easily to allow the computer to print the data, and the charts could be made by hand.

### Operation

To use ML5A and ML5B, the input data should be punched on one card with the format given in table III.

TABLE III.—*ML5A and ML5B Input Data Format*

| Card | Parameter | Column |
|---|---|---|
| No. 1 | Attenuation level (db) | 1–10 |
| | Maximum equivalent frequency (Hz) | 11–20 |

The maximum equivalent frequency can be chosen for convenience. The input data are in floating-point numbers.

Before loading the program, the plotter pin should be placed in the lower right corner of the chart. The design charts given were generated using this program.

### Description

This program plots contours of constant attenuation relating the ripple and equivalent frequency parameters of the lowpass prototype filter. The amplitude response of the Chebyshev filter is given as

$$f(\omega)=\frac{1}{\sqrt{1+\epsilon^2T_n^2(X_0\omega)}}$$

where $\epsilon$ is the ripple factor, $T_n$ $(X_0\omega)$ is the n$^{\text{th}}$ order Chebyshev polynomial, and $X_0$ is the normalization frequency; $X_0$ is chosen so that the lowpass prototype filter has its cutoff (3-db attenuation) at $\omega=1$. Under this condition,

$$X_0=\cos h\left[\frac{1}{n}\cos h^{-1}\left(\frac{1}{\epsilon}\right)\right]$$

and the equation for the contour is

$$\omega=\frac{\cos h\left\{\frac{1}{n}\cos h^{-1}\left[\frac{f}{\epsilon}\left(\frac{1}{\epsilon^2}-1\right)^{1/2}\right]\right\}}{\cos h\left[\frac{1}{n}\cos h^{-1}\left(\frac{1}{\epsilon}\right)\right]}$$

SECTION 4

# Analysis Programs

## 4.1 INTRODUCTION

There are two programs available in this computer-aided design technique that may be used to analyze filters. The first is a frequency analysis program. It can be used to calculate filter amplitude response, phase response, time delay, and group delay. It can be used to analyze Butterworth and Chebyshev filters in the highpass, lowpass, bandpass, and bandstop configurations.

The listed parameters are calculated and printed at intervals and over frequency ranges controlled by the input data. Specifically, the maximum and minimum frequencies of interest are required as input data to the program, along with the magnitude of the frequency interval between calculation points.

The second analysis program is the transient analysis program. It calculates the response, in the time domain, of a lowpass filter to a unit step sinusoidal input. This program can be used for lowpass Butterworth and Chebyshev filters by specifying the order of the filter, ripple in decibels, cutoff frequency (3-db point), frequency of the sinusoidal input, maximum and minimum times of interest, and magnitude of the time interval between calculation points.

## 4.2 ANALYSIS PROCEDURE

### Frequency Analysis

The following information must be obtained for a frequency analysis:

Filter order
Ripple factor
$Q$ factor
Configuration factor
Center frequency
Bandwidth
Maximum frequency of interest
Minimum frequency of interest
Frequency step (between calculation points)

Note that for lowpass and highpass filters the bandwidth parameter does not apply.

Transfer the foregoing data to cards using the format given with the information of program ML1A (see Section 4.3). Add the data cards to program ML1A as shown in figure 3–2 and load into the computer. The computer output includes a listing of frequency, amplitude, phase, time delay, and group delay. An example of this output is shown in figure 4–1.

### Transient Analysis

The following information must be obtained for a transient analysis:

Filter order
Cutoff frequency
Ripple factor
Frequency of input signal (unit step sinusoidal)
Maximum time of interest
Minimum time of interest
Time step (between calculation points)

Transfer the foregoing data to cards using the format given with the explanation of program ML1B (refer to section 4.4). Add the data cards to program ML1B as shown in figure 3–2 and load into the computer. The computer output includes a listing of time and filter output. An example of this output is shown in figure 4–2.



353-813 O—69——2

```
// XEQ

N ODD
 FC = 0.00000E 00  AMP = 0.10000E 01  PHASE = 0.00000E 00  PHASE/W = 0.00000E 00  DELAY = 0.32360E-03
 FC = 0.50000E 03  AMP = 0.10000E 01  PHASE =-0.92736E 01  PHASE/W =-0.18547E 03  DELAY = 0.32391E-03
 FC = 0.10000E 04  AMP = 0.10000E 01  PHASE =-0.18565E 02  PHASE/W =-0.18565E 03  DELAY = 0.32485E-03
 FC = 0.15000E 04  AMP = 0.10000E 01  PHASE =-0.27892E 02  PHASE/W =-0.18595E 03  DELAY = 0.32643E-03
 FC = 0.20000E 04  AMP = 0.10000E 01  PHASE =-0.37275E 02  PHASE/W =-0.18637E 03  DELAY = 0.32871E-03
 FC = 0.25000E 04  AMP = 0.10000E 01  PHASE =-0.46734E 02  PHASE/W =-0.18693E 03  DELAY = 0.33175E-03
 FC = 0.30000E 04  AMP = 0.99999E 00  PHASE =-0.56291E 02  PHASE/W =-0.18763E 03  DELAY = 0.33565E-03
 FC = 0.35000E 04  AMP = 0.99998E 00  PHASE =-0.65974E 02  PHASE/W =-0.18849E 03  DELAY = 0.34053E-03
 FC = 0.40000E 04  AMP = 0.99994E 00  PHASE =-0.75814E 02  PHASE/W =-0.18953E 03  DELAY = 0.34662E-03
 FC = 0.45000E 04  AMP = 0.99983E 00  PHASE =-0.85849E 02  PHASE/W =-0.19077E 03  DELAY = 0.35418E-03
 FC = 0.50000E 04  AMP = 0.99951E 00  PHASE =-0.96125E 02  PHASE/W =-0.19225E 03  DELAY = 0.36359E-03
 FC = 0.55000E 04  AMP = 0.99873E 00  PHASE =-0.10670E 03  PHASE/W =-0.19400E 03  DELAY = 0.37532E-03
 FC = 0.60000E 04  AMP = 0.99699E 00  PHASE =-0.11765E 03  PHASE/W =-0.19609E 03  DELAY = 0.38990E-03
 FC = 0.65000E 04  AMP = 0.99333E 00  PHASE =-0.12907E 03  PHASE/W =-0.19857E 03  DELAY = 0.40782E-03
 FC = 0.70000E 04  AMP = 0.98617E 00  PHASE =-0.14105E 03  PHASE/W =-0.20151E 03  DELAY = 0.42925E-03
 FC = 0.75000E 04  AMP = 0.97298E 00  PHASE =-0.15369E 03  PHASE/W =-0.20493E 03  DELAY = 0.45362E-03
 FC = 0.80000E 04  AMP = 0.95028E 00  PHASE =-0.16705E 03  PHASE/W =-0.20882E 03  DELAY = 0.47894E-03
 FC = 0.85000E 04  AMP = 0.91406E 00  PHASE =-0.18111E 03  PHASE/W =-0.21307E 03  DELAY = 0.50123E-03
```

FIGURE 4–1.—Example of output of ML1A

```
// XEQ

  2  POLE FILTER

 WO =     0.10000E 01              R =  0.000000E 00

 W =  0.100000E 01


 T=     0.000000E 00     FT=     -0.230065E-06
 T=     0.999999E-01     FT=     -0.792084E-02
 T=     0.199999E 00     FT=     -0.157649E-01
 T=     0.299999E 00     FT=     -0.235308E-01
 T=     0.399999E 00     FT=     -0.312164E-01
 T=     0.499999E 00     FT=     -0.388201E-01
 T=     0.599999E 00     FT=     -0.463392E-01
 T=     0.699999E 00     FT=     -0.537721E-01
 T=     0.799999E 00     FT=     -0.611169E-01
 T=     0.899999E 00     FT=     -0.683714E-01
 T=     0.999999E 00     FT=     -0.755339E-01
 T=     0.109999E 01     FT=     -0.826022E-01
 T=     0.119999E 01     FT=     -0.895744E-01
 T=     0.129999E 01     FT=     -0.964487E-01
 T=     0.139999E 01     FT=     -0.103223E 00
 T=     0.149999E 01     FT=     -0.109895E 00
 T=     0.159999E 01     FT=     -0.116465E 00
 T=     0.169999E 01     FT=     -0.122928E 00
 T=     0.179999E 01     FT=     -0.129284E 00
 T=     0.189999E 01     FT=     -0.135531E 00
 T=     0.199999E 01     FT=     -0.141668E 00
 T=     0.209999E 01     FT=     -0.147691E 00
 T=     0.219999E 01     FT=     -0.153601E 00
 T=     0.229999E 01     FT=     -0.159394E 00
 T=     0.239999E 01     FT=     -0.165069E 00
 T=     0.249999E 01     FT=     -0.170625E 00
 T=     0.259999E 01     FT=     -0.176060E 00
 T=     0.269999E 01     FT=     -0.181373E 00
 T=     0.279999E 01     FT=     -0.186561E 00
 T=     0.289999E 01     FT=     -0.191623E 00
 T=     0.299999E 01     FT=     -0.196559E 00
 T=     0.309999E 01     FT=     -0.201365E 00
 T=     0.319999E 01     FT=     -0.206042E 00
 T=     0.329999E 01     FT=     -0.210587E 00
 T=     0.339999E 01     FT=     -0.214999E 00
 T=     0.349999E 01     FT=     -0.219277E 00
 T=     0.359998E 01     FT=     -0.223419E 00
 T=     0.369998E 01     FT=     -0.227425E 00
 T=     0.379998E 01     FT=     -0.231293E 00
 T=     0.389998E 01     FT=     -0.235021E 00
 T=     0.399998E 01     FT=     -0.238610E 00
 T=     0.409998E 01     FT=     -0.242057E 00
 T=     0.419998E 01     FT=     -0.245362E 00
 T=     0.429998E 01     FT=     -0.248523E 00
 T=     0.439998E 01     FT=     -0.251540E 00
 T=     0.449998E 01     FT=     -0.254412E 00
```

FIGURE 4–2.—Example of output of ML1B

## 4.3 FREQUENCY ANALYSIS PROGRAM ML1A

### Purpose

The purpose of the frequency analysis program is to calculate important filter characteristics as a function of frequency. These characteristics include amplitude response, phase response, time delay, and group delay. These response characteristics are printed for frequency values at discrete intervals over a range of frequencies. The value of the frequency step and range is controlled by the operator.

The input data required by this program include specification of the order of the filter, ripple factor, $Q$ factor, filter configuration, filter center frequency, bandwidth, frequency range, and frequency step. The program can be used for Butterworth and Chebyshev filters in lowpass, highpass, bandpass, and bandstop configurations.

### Operation

To use ML1A, the input data should be punched on three cards with the format given in table IV.

The filter order is a fixed-point number; all other numbers are floating point. The ripple factor is specified in decibels. If the $Q$ factor is unknown, punch no number in columns 21–30 on the first data card. The configuration factor is defined as follows:

$$\text{Configuration factor} = \begin{cases} 1.\ \text{Lowpass} \\ 2.\ \text{Highpass} \\ 3.\ \text{Bandpass} \\ 4.\ \text{Bandstop} \end{cases}$$

The computer will calculate the response

TABLE IV.—*ML1A Input Data Format*

| Card | Parameter | Column |
|---|---|---|
| No. 1 | Filter order | 4 and 5 |
| | Ripple factor | 6–20 |
| | *Q* factor | 21–30 |
| No. 2 | Configuration factor | 1–10 |
| | Center frequency (Hz) | 11–20 |
| | Bandwidth (Hz) | 21–30 |
| No. 3 | Maximum frequency (Hz) | 1–10 |
| | Minimum frequency (Hz) | 11–20 |
| | Frequency step (Hz) | 21–30 |

between the maximum frequency and the minimum frequency at intervals determined by the frequency step. For example if

Maximum frequency = 15000,
Minimum frequency = 10000,
Frequency step = 100,

the response characteristics are calculated and printed at frequencies of 10000, 10100, 10200 . . . , 14900, and 15000.

The output format for this program is very straightforward. On each line of the output, there are printed the response characteristics for one frequency in the set of frequency points. Also, on each line, the parameter printed is defined. Each line, therefore, includes a listing of frequency, amplitude response, phase response, time delay, and group delay.

### Description

From the filter specifications given, the program calculates a transfer function of the form

$$F(S)=\frac{S^m+a_1S^{m-1}+a_2S^{m-2}+\ldots+a_m}{S^n+b_1S^{n-1}+b_2S^{n-2}+\ldots+b_n}.$$

To calculate the response characteristics, the transformation is made $S=j\omega$, and the parameters calculated are:

(1) Amplitude response, the absolute magnitude of $F(j\omega)$, $[|F(j\omega)|]$;

(2) Phase response, the phase of $F(j\omega)$, $(\theta)$;

(3) Time delay, the ratio of phase-to-angular frequency $(\theta/\omega)$;

(4) Group delay, the negative derivative of phase with respect to frequency $(-d\theta/d\omega)$.

The subroutines called by this program include:

(1) CARIP—Calculate ripple constant.

(2) CAPOE—Calculate poles even for the filter specified.

(3) CAPOO—Calculate poles odd for the filter specified.

(4) CARES—Set up polynomial ratio, substitute $S=j\omega$, and calculate response characteristics.

Figure 4–3 gives an example of the output of ML1A.

## 4.4 TRANSIENT ANALYSIS PROGRAM ML1B

### Purpose

The purpose of the transient analysis program is to calculate filter response, in the time domain, to a unit step sinusoidal input. This pro-

```
// XEQ
      FC = 0.90000E 04  AMP = 0.86108E 00  PHASE =-0.19569E 03  PHASE/W =-0.21743E 03  DELAY = 0.51482E-03
      FC = 0.95000E 04  AMP = 0.79088E 00  PHASE =-0.21047E 03  PHASE/W =-0.22155E 03  DELAY = 0.51425E-03
      FC = 0.10000E 05  AMP = 0.70710E 00  PHASE =-0.22499E 03  PHASE/W =-0.22499E 03  DELAY = 0.49721E-03
      FC = 0.10500E 05  AMP = 0.61675E 00  PHASE =-0.23882E 03  PHASE/W =-0.22745E 03  DELAY = 0.46604E-03
      FC = 0.11000E 05  AMP = 0.52750E 00  PHASE =-0.25162E 03  PHASE/W =-0.22874E 03  DELAY = 0.42636E-03
      FC = 0.11500E 05  AMP = 0.44519E 00  PHASE =-0.26322E 03  PHASE/W =-0.22889E 03  DELAY = 0.38406E-03
      FC = 0.12000E 05  AMP = 0.37289E 00  PHASE =-0.27364E 03  PHASE/W =-0.22803E 03  DELAY = 0.34338E-03
      FC = 0.12500E 05  AMP = 0.31138E 00  PHASE =-0.28294E 03  PHASE/W =-0.22635E 03  DELAY = 0.30652E-03
      FC = 0.13000E 05  AMP = 0.26006E 00  PHASE =-0.29124E 03  PHASE/W =-0.22403E 03  DELAY = 0.27421E-03
      FC = 0.13500E 05  AMP = 0.21766E 00  PHASE =-0.29869E 03  PHASE/W =-0.22125E 03  DELAY = 0.24634E-03
      FC = 0.14000E 05  AMP = 0.18280E 00  PHASE =-0.30540E 03  PHASE/W =-0.21814E 03  DELAY = 0.22243E-03
      FC = 0.14500E 05  AMP = 0.15414E 00  PHASE =-0.31147E 03  PHASE/W =-0.21480E 03  DELAY = 0.20192E-03
      FC = 0.15000E 05  AMP = 0.13056E 00  PHASE =-0.31699E 03  PHASE/W =-0.21133E 03  DELAY = 0.18425E-03
      FC = 0.15500E 05  AMP = 0.11108E 00  PHASE =-0.32205E 03  PHASE/W =-0.20777E 03  DELAY = 0.16896E-03
      FC = 0.16000E 05  AMP = 0.94936E-01  PHASE =-0.32669E 03  PHASE/W =-0.20418E 03  DELAY = 0.15563E-03
      FC = 0.16500E 05  AMP = 0.81495E-01  PHASE =-0.33098E 03  PHASE/W =-0.20059E 03  DELAY = 0.14394E-03
      FC = 0.17000E 05  AMP = 0.70255E-01  PHASE =-0.33495E 03  PHASE/W =-0.19703E 03  DELAY = 0.13363E-03
      FC = 0.17500E 05  AMP = 0.60814E-01  PHASE =-0.33865E 03  PHASE/W =-0.19351E 03  DELAY = 0.12447E-03
      FC = 0.18000E 05  AMP = 0.52848E-01  PHASE =-0.34209E 03  PHASE/W =-0.19005E 03  DELAY = 0.11629E-03
      FC = 0.18500E 05  AMP = 0.46097E-01  PHASE =-0.34532E 03  PHASE/W =-0.18666E 03  DELAY = 0.10895E-03
      FC = 0.19000E 05  AMP = 0.40353E-01  PHASE =-0.34834E 03  PHASE/W =-0.18334E 03  DELAY = 0.10234E-03
      FC = 0.19500E 05  AMP = 0.35444E-01  PHASE =-0.35119E 03  PHASE/W =-0.18009E 03  DELAY = 0.96348E-04
```

FIGURE 4–3.—Example of output of ML1A

gram is for lowpass filters with any cutoff frequency. The data required include specification of the order of the filter, the cutoff frequency, the ripple factor (in db), the range of time for which the response is to be calculated, and the time interval.

The program can be used for a unit step sinusoidal of any frequency, and this parameter must also be provided in the input data. The program is designed for Butterworth and Chebyshev filters.

### Operation

To use ML1B, the input data should be punched on three cards with the format given in table V.

TABLE V.—*ML1B Input Data Format.*

| Card | Parameter | Column |
|---|---|---|
| No. 1 | Filter order | 4 and 5 |
| No. 2 | Cutoff frequency (Hz) | 1–10 |
| | Ripple factor (in db) | 11–20 |
| No. 3 | Sinusoidal input frequency (Hz) | 1–10 |
| | Maximum time of interest (sec) | 11–20 |
| | Minimum time of interest (sec) | 21–30 |
| | Time step (sec) | 31–40 |

The filter order is a fixed-point number; all other numbers are floating point.

The computer will calculate the response between the maximum time and minimum time in intervals determined by the time step. For example, if

Maximum time=20 msec,
Minimum time=10 msec,
Time step=0.1 msec,

the response characteristics are calculated and printed at times of 10.0, 10.1, 10.2, . . . , 19.9, and 20.0 msec.

The output format for ML1B is very similar to that for ML1A. In the transient analysis program, a listing of the order of the filter, the ripple factor, cutoff frequency of the filter, and frequency of the sinusoidal unit step input precedes the response data. The response data include a listing of the time and filter output voltage.

### Description

From the filter specifications given, the program calculates a transfer function of the form

$$F(S)=\frac{S^m+a_1S^{m-1}+a_2S^{m-2}+\ldots+a_m}{S^mb_1S^{n-1}+b_2S^{n-2}+\ldots+b_n}$$

The Laplace transform of the unit step sinusoidal,

$$e(t)=\mu(t)\sin(\omega t)$$

is given as

$$E(S)=\frac{\omega}{S^2+\omega^2}$$

The response, given by

$$R(S)=F(S)E(S)=\frac{\omega(S^m+aS^{m-1}+\ldots+a_m)}{(S^2+\omega^2)(S^m+b_1S^{m-1}+\ldots b_m)}$$

is expanded in a partial fraction

$$R(S)=\frac{A_1}{(S-P_1)}+\frac{A_2}{(S-P_2)}+\cdots$$

where $P_i$ is the $i^{\text{th}}$ pole of the function $R(S)$. The constants are determined by calculating the residue at each of the poles. Each of the partial fractions are then transformed back to the time domain and summed to give the response function $r(t)$.

The subroutines called by this program include:

(1) CARIP—Calculate ripple constant

(2) ATRAN—Calculate the transfer function, expand in partial fractions, and take the inverse Laplace transform to find the response in the time domain.

# SECTION 5

# Passive Filter Synthesis

## 5.1 INTRODUCTION

Chebyshev and Butterworth filters may be synthesized in a Cauer or Darlington configuration as shown in figures 5–1 and 5–2. The Darlington circuit is a doubly terminated ladder network, and the Cauer circuit is a singly terminated ladder network. One advantage of the Cauer synthesis procedure is the ability to compensate for dissipative effects in the filter by predistortion. This is done automatically by the computer; the designer merely provides an estimate of the $Q$ factor of the components to be used in constructing the filter. The ability to predistort is important because parasitic dissipative effects help to deteriorate a filter response. Parasitic dissipation results in decreasing ripple, decreasing roll-off rate, and increasing flat loss.

In making the choice between a Cauer or Darlington configuration, these characteristics should be considered:

*Darlington*—High $Q$ elements required; doubly terminated network

*Cauer*—Lower $Q$ can be used by utilizing the predistortion ability of the program; singly terminated network

## 5.2 PROCEDURE

The following procedure can be used in synthesizing a filter.

(1) Obtain the following information:
  Filter order
  Ripple factor
  $Q$ factor
  Configuration factor
  Center frequency
  Bandwidth
  Terminal impedance

(2) Decide which network configuration—either the Darlington or the Cauer—is to be used. Normally the Cauer synthesis should be used. If Cauer synthesis is used, go to step 3. If Darlington synthesis is used, go to step 4.

(3) Transfer the data mentioned in step 1 to cards using the format given with the explanation of program ML3 (refer to section 5.6). Add the data cards to program ML3 shown in

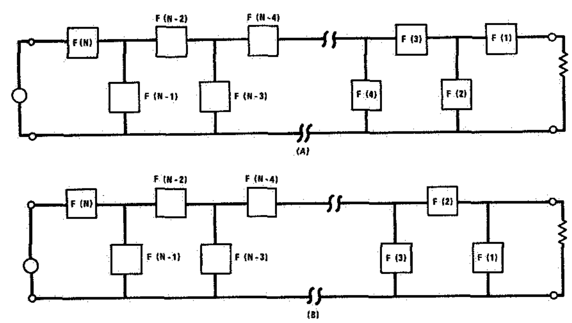



FIGURE 5–1.—Cauer configurations (A) odd order and (B) even order

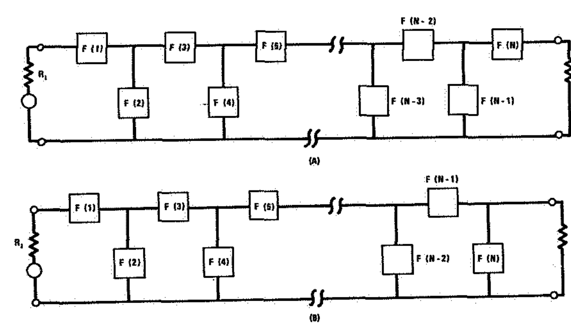



FIGURE 5–2.—Darlington configurations (A) odd order and (B) even order

figure 3–2 and load into the computer. The computer output will include the circuit elements required to realize the filter in a Cauer configuration. Also, information will be provided by the computer that will prove useful in constructing the filter. An example of the computer output for Cauer synthesis is shown in figure 5–3.

The outputs are as follows:

*Ripple*—The same as read in.

*Theoretical poles*—For the lowpass prototype, for reference only.

*Branch bandwidth*—This parameter is printed for bandpass and bandstop filters only, where the ladder branches are resonant circuits; it is the frequency between 3–db points for any branch in the ladder.

*Filter bandwidth*—This is printed for bandpass and bandstop filters only and is a listing of the bandwidth specified in the input data.

*Center frequency*—The filter center frequency listed in the input data; if the filter is a lowpass or highpass type, this parameter corresponds to the 3–db frequency.

*Tap ratio*—Provides information needed if it is desired to use tapped coils in a bandpass filter; to use tapped coils and the information generated by this program, refer to section 5.3.

```
// XEQ           1
*LOCALML3,CAPOO,PRDSO,XOPOL,CAPOE,PRDSE,XEPOL,SYCAS

        RIPPLE =     0.000000E 00

    THEORETICAL POLES

       1       -0.707106E 00          0.707107E 00
    N EVEN

    BRANCH BANDWIDTH         0.711100E 03

    FILTER BANDWIDTH   0.110000E 05

    CENTER FREQUENCY         0.142220E 05


    COIL         TAP 2 RATIO          TAP 3 RATIO

       1            0.103595E 01                 0.100000E 01


     COIL         FC. DEF. 2                     FC. DEF. 3

       1            0.821392E 04                 0.901472E 04

    FLAT LOSS =      0.107320E 01

    CAUER CIRCUIT ELEMENTS

    I= 1         F=     0.851519E-07
    I= 2         F=     0.157836E-02
    TERMINAL RESISTANCE =     0.100000E 03

    N =    2   Q =      20.000003   E =   0.000000E 00
```

FIGURE 5–3.—Example of output of ML3

*Frequencies difference*—Parameter needed in tuning a tapped bandpass filter; this information is used to adjust the tap point to compensate for non-ideal coupling in the tapped coil.

*Flat loss*—The loss, in db across the filter caused by dissipative effects.

*Cauer circuit elements*—The circuit elements for Cauer realization are given in the output, branch by branch; to reference the $F(i)$ terms shown to the ladder network, reference should be made to figure 5–1 and section 5.3.

*Terminal resistance*—The value of the terminal resistance, an input parameter, is listed for reference.

*Order and Q factor*—Listed for reference.

(4) Transfer the data mentioned in step 1 to cards, using the format given with the explanation of programs ML2A and ML2B (refer to section 5.5). If the order of the filter to be synthesized is even, use program ML2A; if the order is odd, use program ML2B. Add the data cards to the program as shown in figure 3–2 and load into the computer. The computer output will include the circuit elements required to realize the filter in a Darlington configuration. An example of the computer output for Darlington synthesis is shown in figure 5–4. The outputs are as follows:

*Theoretical poles* and *theoretical zeroes*—Of the reflection coefficient function are listed for reference

*Predistorted poles* and *predistorted zeroes*—Should be the same as the theoretical values for Darlington synthesis

*Denominator polynomial Numerator polynomial Denominator polynomial after ADPOL Numerator polynomial after APPOC*—These are reference parameters used for checking subroutines only and are listed here for reference

*Z input resistance Darlington circuit elements*—The circuit elements for Darlington realization are given in the output, branch by branch. To reference the $F(i)$ terms shown to the ladder network, reference should be made to figure 5–2 and section 5.3.

```
// XEQ          1
*LOCALML2.ANOD.ANEV.AFIN

   N ODD


   THEORETICAL POLES
  SPOL =  -0.100000E 01
  I =  1   APHA =  -0.499999E 00   BETA =   0.866025E 00

   THEORETICAL ZEROES
  SOL =   0.000000E 00
  I =  1   AHA =   0.000000E 00   BTA =   0.866025E-15

   PREDISTORTED POLES
  SPOL =  -0.999999E 00
  I =  1   APHA =  -0.499998E 00   BETA =   0.866025E 00

   PREDISTORTED ZEROES
  SOL =   0.100000E-05
  I =  1   AHA =   0.100000E-05   BTA =   0.866025E-15

   DENOMINATOR POLYNOMIAL
  AP( 1) =        0.199999E 01
  AP( 2) =        0.199999E 01
  AP( 3) =        0.999997E 00

   NUMERATOR POLYNOMIAL
  AZ( 1) =      -0.300000E-05
  AZ( 2) =       0.299999E-11
  AZ( 3) =      -0.999999E-18

   DENOMINATOR POLYNOMIAL AFTER ADPOL
  AD( 1) =    0.199999E 01
  AD( 2) =    0.199999E 01
  AD( 3) =    0.999997E 00

   NUMERATOR POLYNOMIAL AFTER ADPOL
  AN( 1) =    0.199999E 01
  AN( 2) =    0.199999E 01
  AN( 3) =    0.999997E 00
     I= 1          F=     0.111907E-02
     I= 2          F=     0.223816E-06
     I= 3          F=     0.111907E-02
     I= 4          F=     0.100000E 03
     Z=      0.100000E 03
  F( 1) =    0.1119077E-02
  F( 2) =    0.2238160E-06
  F( 3) =    0.1119075E-02
  F( 4) =    0.1000000E 03


   N =    3   Q = 1000000.126464   E =   0.000000E 00
```

FIGURE 5-4.—Example of output of ML2B

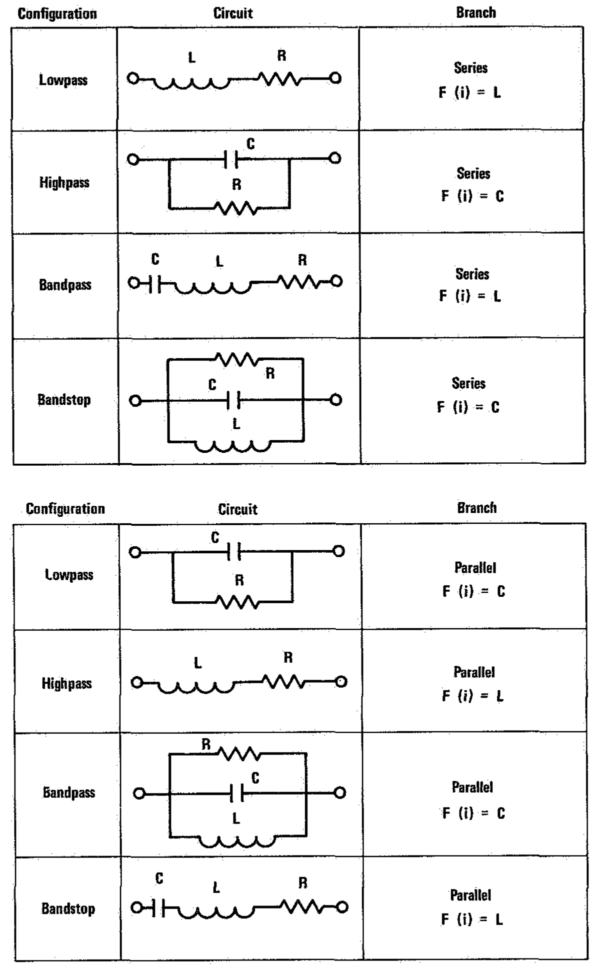



FIGURE 5-5.—Configurations of $F(i)$ for various filter types

## 5.3 REALIZATION, IMPEDANCE, SCALING, AND TAPPING

The outputs of the Darlington and Cauer synthesis programs are a set of subscripted variables $F(i)$. These variables refer to each branch of the respective ladder networks shown in figures 5–1 and 5–2. Figure 5–5 relates the branch configurations for the series and parallel branches to the filter configuration—lowpass, highpass, bandpass, or bandstop. Figure 5–5 also relates the value $F(i)$ to the branch configuration. For example, in a lowpass filter, the values of $F(i)$ for the series are actual inductances, and for the parallel branches the $F(i)$ values are actual capacitances.

The following rules are to be followed in determining the other circuit values in each branch.

(1) The resistances are chosen to give each branch the $Q$ value specified by the design requirements. This resistance is usually chosen during tuning.

(2) In the bandpass and bandstop design configurations, the adjacent reactive element is chosen to make the branch resonant at the filter center frequency.

After the synthesis procedure is complete and if the circuit values are found to be impractical physically (i.e., capacitances and inductances too large or too small), it may be possible to achieve practical circuit component values by scaling the terminal resistance up or down. For example, scaling up resistance scales up the inductances and scales down the capacitances. The best procedure is to choose a new terminal resistance and reload the program with these new data. The component values can now be checked to determine if they are physically realizable.

Another capability of the computer, calculating tap points, can be used to achieve a more practical realization of a given filter. Figure 5–6 gives an example of a filter and its realization as a tapped-coil configuration. Figure 5–7 shows details of a branch featuring tapped coils. This branch is part of a bandpass filter. If the filter is realized in this manner, all branches have the same circuit value as $F(1)$ in the computer output data. The advantage of using tapped coils is that all circuit elements are the same, i.e., same size capacitors, and same total turns on each coil.

## 5.4 FILTER TUNING

Passive filters, in general, must be tuned to achieve an acceptable response. The computer program gives information that is useful in performing the tuning. This section describes information generated that can be used in tuning a bandpass filter.

The calculated parameter, branch bandwidth, can be used to adjust the resistances to achieve the proper $Q$ value for the branch. This is naturally important to realize the proper response.

If tapped coils are used, the following procedure can be used to compensate for imperfect coupling in the coil. The frequency difference parameter printed in the program output is the theoretical difference between two peaks in the frequency response at the tap point. To adjust the tap point, one series branch and one parallel branch are connected, driven through the series branch, and the voltage at the tap point is observed as a function of frequency. If the frequency difference between the peak points is different from that calculated by the computer, the tap point is adjusted up or down (i.e., a turn added or subtracted) until the theoretical frequency difference is attained. See figure 5–7

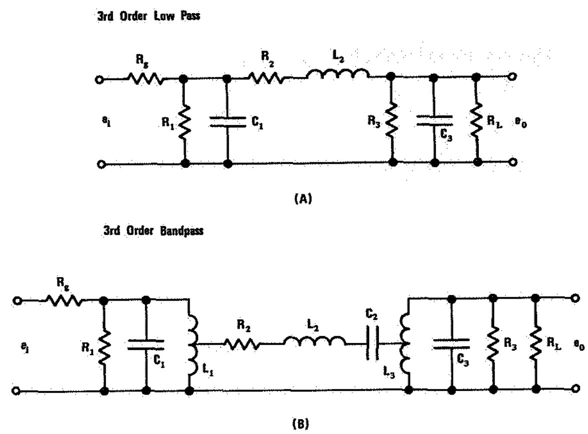



FIGURE 5–6.—Realization of a filter in a tapped coil configuration (A) nontapped and (B) tapped

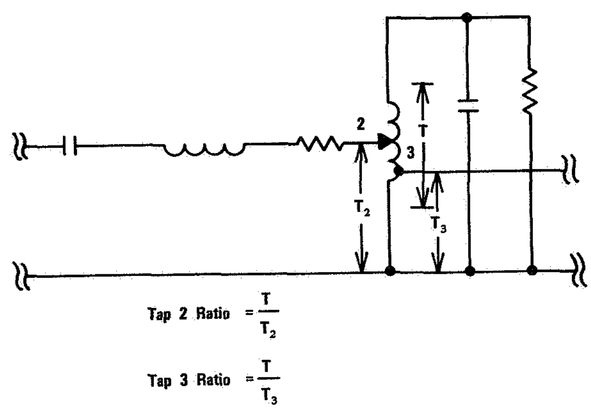



FIGURE 5–7.—Example of a bandpass series and parallel branch with tapped coils

for an example of a series and parallel branch coupled by tapped coils. Figure 5–6 shows a bandpass filter and its realization as a tapped structure.

## 5.5 DARLINGTON SYNTHESIS PROGRAMS ML2A AND ML2B

### Purpose

The purpose of ML2A and ML2B is to synthesize a filter by the Darlington synthesis method. The ML2A program is used for even-order filters, and the ML2B for odd-order cases. The program covers Chebyshev and Butterworth filters, and each program can be used to synthesize filters in the lowpass, highpass, bandpass, and bandstop configurations. Darlington synthesis results in a doubly terminated ladder network (see fig. 5–2). This Darlington synthesis program is for filters that will use circuit elements with low dissipation factors (high $Q$).

The input data required for the program include specification of the filter order, ripple factor, $Q$ factor, configuration factor, center frequency, bandwidth, and terminal impedance. The program output is the values of the circuit elements to be used in the branches ($F_1$, $F_2$, . . . , $F_n$ as defined in fig. 5–2).

#### Operation

To use ML2A and ML2B, the input data should be punched on the cards in the format given in table VI.

TABLE VI.—*ML2A and ML2B Input Data Format*

| Card | Parameter | Column |
|---|---|---|
| No. 1 | Filter order | 4 and 5 |
| | Ripple factor (in db) | 6–20 |
| | $Q$ factor | 21–30 |
| No. 2 | Configuration factor | 1–10 |
| | Center frequency (Hz) | 11–20 |
| | Bandwidth (Hz) | 21–30 |
| | Terminal impedance (ohms) | 31–40 |

The filter order is a fixed-point number; the remaining input data are all in floating point. The $Q$ factor should be any number larger than $1 \times 10^6$. The configuration factor is defined as follows:

$$\text{Configuration factor} = \begin{cases} 1.\ \text{Lowpass} \\ 2.\ \text{Highpass} \\ 3.\ \text{Bandpass} \\ 4.\ \text{Bandstop} \end{cases}$$

The terminal impedance, in ohms, can be any arbitrary value and can be varied to obtain reasonable values for the circuit parameters. The output format includes printing the circuit factors as a subscripted variable, i.e., $F(1)=$ $F(2)=$etc.

#### Description

Darlington synthesis is based on the calculation of a network's reflection coefficient and commensurate determination of its driving point impedance. A four-terminal network with input impedance $Z_i(S)$ and resistive terminations $R_1$ and $R_2$ has the reflection coefficient

$$\Gamma(S) = \frac{Z_i(S) - R_1}{Z_i(S) + R_1}.$$

The relation existing between the reflection coefficient and transfer function is

$$\Gamma(S)\Gamma(-S) = 1 - \frac{4R_1R_2}{(R_1+R_2)^2}\frac{1}{P(S)},$$

where $P(S)$ is the transfer function. The basis of Darlington synthesis is then

(1) Determine $\Gamma(S)$.

(2) Find the driving-point impedance.

(3) Compute the circuit components required to realize this driving-point impedance by performing a continued fractional expansion of the polynomial ratio defining this parameter. The subroutines called by these two programs are CARIP, ANEV, ANOD, and AFIN.

### 5.6 CAUER SYNTHESIS PROGRAM ML3

#### Purpose

The purpose of ML3 is to synthesize a filter by the Cauer synthesis procedure. Cauer synthesis results in a singly terminated ladder network (see fig. 5–1). This program covers Chebyshev and Butterworth approximation-type filters in the lowpass, highpass, bandpass, and bandstop configurations.

The main advantage of the Cauer synthesis program is its ability to predistort the filter's pole—zero–plot to compensate for dissipative effects in the filter circuit elements. This predistortion is automatic in the program; the operator merely must supply the expected $Q$ factor of the circuit elements to be used in realizing the filter.

The input data required by this program include specification of order of the filter, the ripple factor, the element $Q$ factor, the configuration factor, center frequency, bandwidth, and impedance of the termination in ohms. The program output is the values of the circuit elements to be used in the branches ($F(1)$, $F(2)$, . . ., $F(N)$ as defined in figure 5–1).

### Operation

To use ML3, the input data should be punched on two cards with the format given in table VII.

TABLE VII.—*ML3 Input Data Format*

| Card | Parameter | Column |
|---|---|---|
| No. 1 | Filter order | 4 and 5 |
| | Ripple factor (db) | 6–20 |
| | *Q* factor | 21–30 |
| No. 2 | Configuration factor | 1–10 |
| | Center frequency (Hz) | 11–20 |
| | Bandwidth (Hz) | 21–30 |
| | Terminal impedance (ohms) | 31–40 |

The filter order is a fixed-point number; the remaining input data are all floating point. The configuration factor is defined as follows:

$$\text{Configuration factor}=\begin{cases}1.\ \text{Lowpass}\\2.\ \text{Highpass}\\3.\ \text{Bandpass}\\4.\ \text{Bandstop}\end{cases}$$

The terminal impedance, in ohms, can be any arbitrary value and can be varied to obtain reasonable numbers for the circuit parameters. The output format includes printing the circuit factors as a subscripted variable, i.e., $F(1)=$, $F(2)=$, etc.

### Description

Cauer synthesis used here allows the realization of a network transfer function as a resistively terminated ladder network. The transfer emittance of such a network is

$$\frac{I_2}{V_1}=-Y_{12}=\frac{-Y_{12}G_2}{G_2+Y_{22}}$$

where $G_2$ is the admittance of the termination. If $G_2$ is normalized to 1, this equation takes the form

$$-Y_{12}=\frac{-Y_{12}}{1+Y_{22}}$$

The transfer function (voltage ratio) of the network is given as

$$G_{12}=\frac{-Y_{12}}{G_2+Y_{22}}$$

and in the normalized case, it is seen that

$$G_{12}=\frac{-Y_{12}}{1+Y_{22}}$$

If the transfer function is written in the form

$$G_{12}=\frac{P(S)}{q_1(S)+q_2(S)}$$

then

$$-Y_{12}=\frac{P(S)}{q_1(S)+q_2(S)}=\frac{P(S)/q_1(S)}{1+q_2(S)/q_1(S)}=\frac{-Y_{12}}{1+Y_{22}}$$

and

$$Y_{22}=q_2(S)/q_1(S)$$

We select $q_1$ $(S)$ as the polynomial composed of the even parts of the polynomial in the denominators of the transfer functions, and $q_2(S)$ as the odd parts. If the choice were not made in this manner, the network would not be realizable as a grounded structure.

The network components required for the filter can be determined from a continued fractional expansion of the polynomial ratio defining $Y_{22}(S)$. Program ML3 calculates the components required to realize a filter by:

(1) Calculating the filter transfer function,

(2) Isolating the output admittance $Y_{22}$ by assuming a normalized resistive termination,

(3) Calculating the circuit elements by performing a continued fractional expansion of this polynomial ratio. The subroutines called by ML3 include CARIP, CAPOO, PRDSO, CARES, XOPOL, CAPOE, PRDSE, XEPOL, SYCAS, and FITUM.

One important capability of this program is predistortion. The $Q$ value of the circuit components to be used in the filter is read in, and the pole positions of the filter transfer function are distorted so that the effect of dissipation on the shape of the response curve is compensated. The complex operators transformation for predistortion is given as

$$S^*=S+\frac{1}{Q}$$

where $S^*$ is the dissipative complex operator.

SECTION 6

# Active Filter Synthesis

## 6.1 ACTIVE FILTER DESIGN THEORY

### Introduction

For many filter applications, either an active or passive filter can satisfy the design specifications. However, the active filter is rarely used although it offers several advantages over passive filters. One reason for this is that passive design techniques are well developed and documented and have therefore become the conventional design method. A second and more important reason is that telemetry systems require accurate filters for a range of frequencies extending into megacycles, and most active design techniques that have been developed are valid for only extremely low frequencies, typically below 1 kHz. Numerous "cookbook" formulas and circuits have been developed for active filters, but none are applicable beyond 1 kHz. Most are grossly inaccurate for frequencies above 10 kHz.

This accuracy limitation has been rectified for this design procedure by the development of an active filter design technique that is highly accurate for frequencies up to 500 kHz. This covers most of the frequency range required for FM telemetry applications. The design technique basically consists of taking ideal active networks used in conventional low-frequency design and extending their analysis to include higher frequency effects. This involved deriving accurate mathematical filter models that took all pertinent circuit parameters into consideration. Equations describing these networks were derived and used to calculate the component values required to realize the network for a given response. The resulting synthesis design can be used to realize a filter network with sufficient accuracy to assure that the resultant frequency response will be within a few percent of specified cutoff frequencies.

Active networks require the use of resistors and capacitors in conjunction with active elements and eliminate the need for any inductive components. Since resistors and capacitors can be specified accurately, complex and time-consuming tuning procedures can be eliminated. By eliminating the complex tuning requirements, the time required to construct the filter is greatly reduced. Since active filter synthesis eliminates the need for inductive components, the active filter can be fabricated in integrated-circuit form, which permits a smaller package size and higher reliability than that possible with passive filters. An active filter in integrated-circuit form can have a higher reliability than an equivalent passive filter using discrete components.

The ideal networks of figure 6–1 are the conventional ideal models used for low-frequency filter design. These networks are helpful in visualizing the construction of active filters. Active synthesis is accomplished through the use of cascaded lowpass and highpass quadratic (second order) sections. Any lowpass, highpass, or bandpass filter of any order can be obtained by cascading these sections. For example, a sixth-order bandpass filter would use three lowpass and three highpass quadratic sections. For odd-order filters, a first-order section consisting of a resistor and capacitor is added to the last quadratic section. The amplifier element, having a gain $k$, can be any high-gain, feedback-type amplifier such as an emitter follower, Darlington, or operational amplifier.

The denominator of the voltage transfer ratio for both the lowpass and the highpass

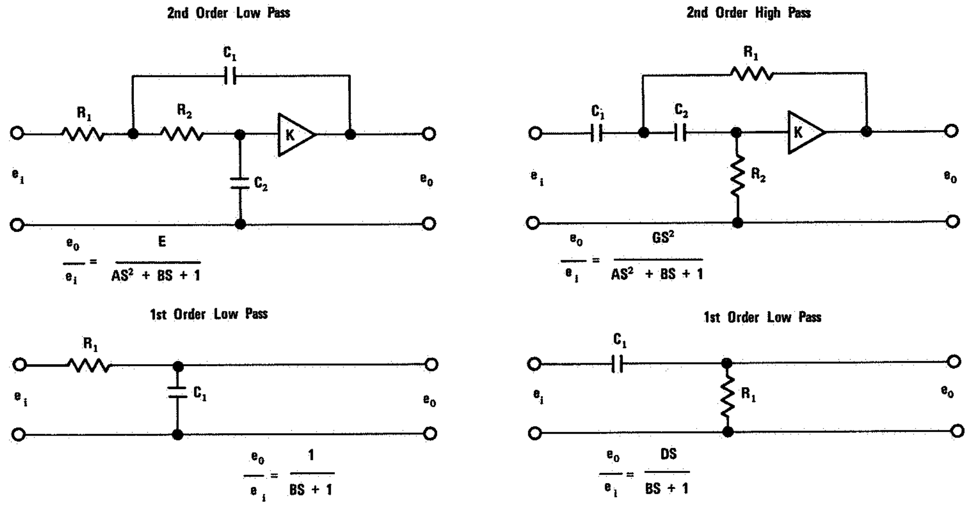



FIGURE 6–1.—Ideal filter networks

quadratic networks are of the form

$$\left(\frac{S}{\omega_n}\right)^2 + B\left(\frac{S}{\omega_n}\right) + 1$$

where $B$ is twice the damping ratio, and $\omega_n$ is the natural frequency in radians per second. One quadratic network is needed to realize a second-order filter. The parameters that need to be specified are $B$ and $\omega_n$. For a fourth-order bandpass filter, two lowpass and two highpass quadratic sections are required. The fourth-order bandpass can be considered as a combination of a fourth-order lowpass together with a fourth-order highpass. The computer program calculates the required values of $B$ and $\omega_n$ for the type filter specified. These quantities ordinarily would be sufficient data for the computer to calculate the discrete component values required to realize the filter. However, since the conventional network models of figure 6–1 are limited in accuracy, other parameters related to a more accurate model also must be specified.

For accurate synthesis at higher frequencies, other circuit parameters must be taken into consideration to obtain an accurate, realistic model. These additional parameters were found to be amplifier input capacitance ($C_A$), output resistance ($R_O$), and gain ($K$), and the source impedance of the input generator ($R_g$). Network models taking these parameters into consideration are shown in figure 6–2. These models are used by the computer program to synthesize the filter. The programs calculate the component values of $R_1$, $R_2$, $C_1$, and $C_2$ shown on each network. The elements $R_O$, $R_g$, and $C_A$ are an integral part of the amplifier and are not discrete components.

These parameters add additional terms to the numerator of the voltage transfer functions and alter the coefficients of the existing terms. An $R_O$ as low as 5 ohms can seriously distort the desired response if not taken into consideration. Even when these parameters are considered, the extra terms in the numerator cause some degradation of the response by limiting the amplitude attenuation far out in the reject-frequency band. However, this seldom causes any problem since the attenuation is usually far below the desired value before the roll-off is limited.

Two separate programs for the design of active filters are provided. Both programs print out the minimum and maximum values of the additional parameters of the realistic network model. The first program (ML6) automatically assigns acceptable values to these additional parameters ($K$, $R_O$, $R_g$, $C_A$) and calculates the component values of $R_1$, $R_2$, $C_1$, and $C_2$ required to synthesize the specified filter. The second program (ML7) allows the operator to assign values to these additional

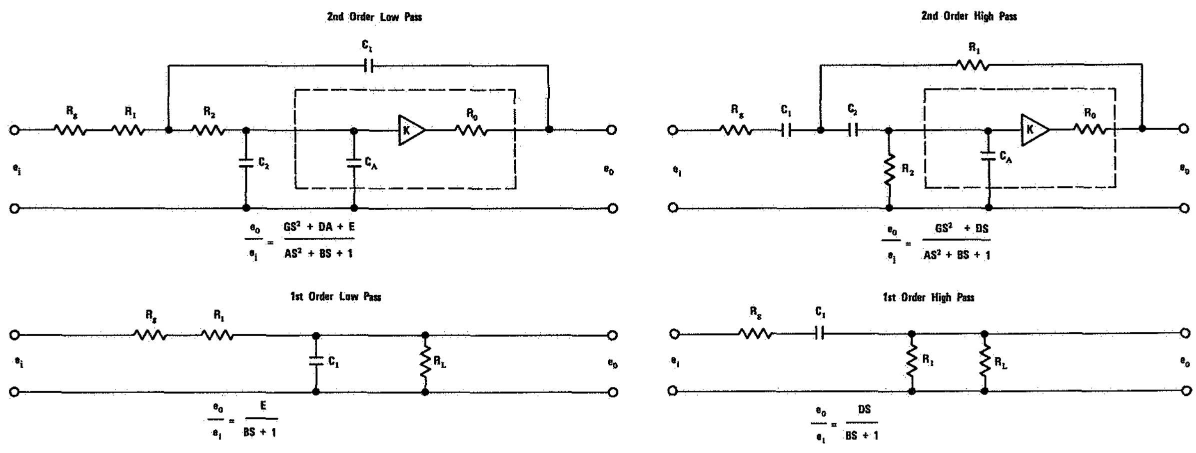


FIGURE 6–2.—Actual filter networks

parameters. The operator can choose, within the specified limits, any set of desired values. The program then calculates $R_1$, $R_2$, $C_1$, and $C_2$.

The capacitive elements of the network can have dissipative components associated with them so that low-$Q$ capacitors can be used. When dissipative components are used, the computer program will predistort the pole-zero plot of the filter such that synthesis with the dissipative elements will yield the proper response.

Both programs calculate the frequency response curves for the voltage transfer ratio ($E_{out}/E_{in}$) of the networks used to synthesize the filter. The response curves are calculated for amplitude, phase, time delay, and group delay. The amplitude response is for the magnitude of the voltage transfer ratio, and the phase response gives the phase angle in degrees. The time delay is the delay in seconds that a sinusoidal signal encounters in passing through the network. The time delay is defined as the negative of the phase angle divided by the signal frequency. The group delay is the relative delay in seconds of the envelope components of a modulated signal passing through the network and is therefore sometimes called the envelope delay. The group delay is defined as the negative of the derivative of the phase angle with respect to the signal frequency.

The specially developed active filter design equations used in the programs are not included in this manual, except in the program source listings, because they are not needed for effective use of the programs. If desired, a thorough development of the design theory and equations can be obtained from a Thesis entitled, "Analysis and Synthesis of Active Quadratic Filter Networks for Telemetry Applications", University of Alabama in Huntsville, 1969, by Sidney Gussow. The Thesis essentially is a detailed design manual for active filters using quadratic networks and was the source for much of the material used in this section.

### Amplifier Requirements

The amplifier chosen for the active element of the quadratic network is dictated by the restrictions on the values of the amplifier parameters. These parameters are amplifier output resistance ($R_O$), input capacitance ($C_A$), and gain ($K$). The restrictions on these parameters vary for each quadratic network and depend upon the filter requirements. Both computer programs print out, for each quadratic network, values for the maximum gain permissible ($K_{max}$), the minimum gain permissible ($K_{min}$), and the maximum allowable output resistance ($R_{Omax}$). These limits however are interrelated; i.e., if the limits on one parameter are further restricted, the limits on the other parameters are extended. The set of limits calculated is for an input capacitance ($C_A$) of 10 picofarads. If $C_A$ is increased, the limits are further restricted. The limits on any parameters can be exceeded a small amount by tightening the restrictions on another parameter.

$K_{max}$ varies from 2.5 to 1.0, and $K_{min}$ varies from 0.6 to 1.0. For high-order filters, or bandpass filters with narrow bandwidths, $K_{max}$ and $K_{min}$ quickly converge toward unity; i.e., an amplifier with a stable gain near unity is required. As $K_{min}$ approaches extremely close to unity ($K>0.995$), the stability of the gain chosen becomes more critical. For example, if $K_{min}$ were 0.99990, the gain of the amplifier used would have to be within 0.01 percent of the gain value it was designed for. Although this is an extreme case, the gain requirement could be satisfied by the use of a high-gain operational amplifier.

The amplifier choice is restricted to those amplifiers having parameters that lie within the specified limits of $K$ and $R_o$. Three recommended amplifier types along with their typical ranges of parameter values are shown in figure 6-3. This can be used as a guide in selecting an amplifier that meets the requirements.

For the amplifier of figure 6-3(A), figures 6-4 and 6-5 aid in determining the actual values of $R_O$ and $K$. Figure 6-4 shows a graph of $R_O$ versus emitter current ($I_E$). The current $I_E$ is determined by the value of the chosen emitter resistor $R_s$ and the negative supply voltage. After $R_O$ is found, the gain $K$ can be determined from figure 6-5, which shows a plot of $K$ versus $R_s$ for various values of $R_O$. Figure 6-6, which shows a similar graph of $R_O$ versus $I_E$, can be used to find $R_O$ for the amplifier of figure 6-3(B). Figure 6-7 can then be used to

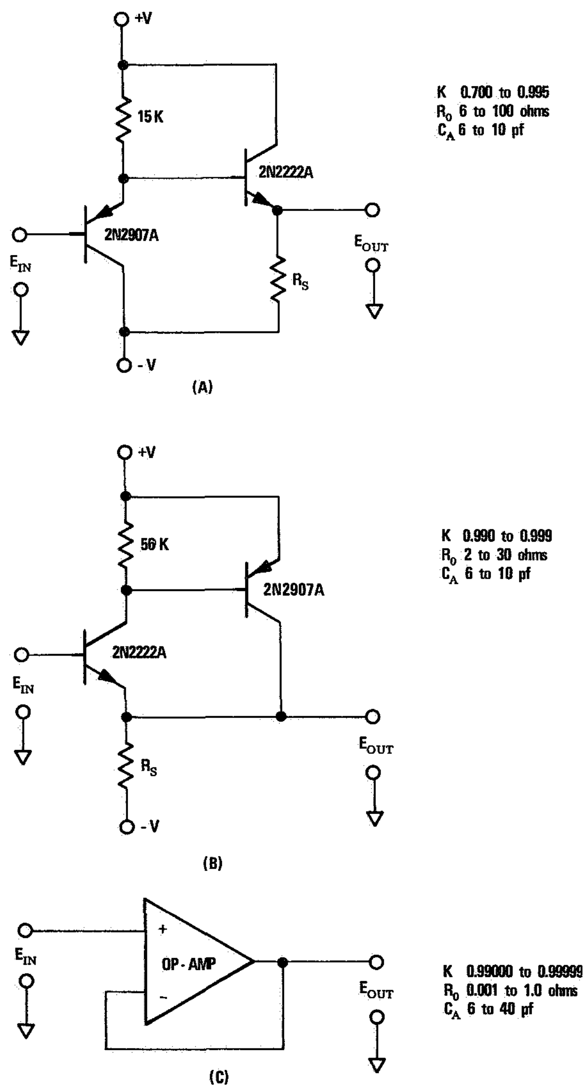



FIGURE 6-3.—Recommended amplifier types for second order networks

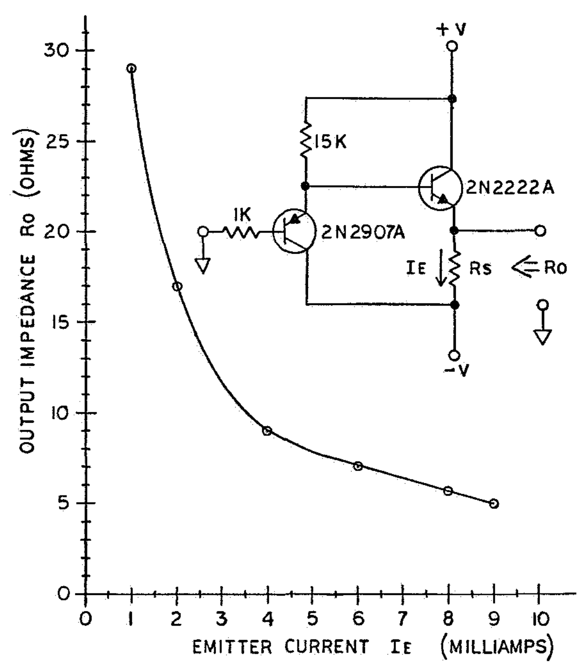



FIGURE 6-4.—Amplifier output resistance chart

find the gain $K$. These curves vary slightly from one transistor to another, but values obtained directly from these curves are usually accurate enough to obtain good design results. If other transistors or amplifier circuits are used, the gain and output resistance of the amplifier must be measured.

Operational amplifiers having a high open-loop gain can be used to obtain a gain as close to unity as 0.99999 and an output resistance as low as 0.001 ohm. For filters requiring amplifiers having parameters beyond these limits, the filter cannot be accurately realized. When using operational amplifiers, two other characteristics should be observed. One is that the frequency roll-off of the amplifier itself must occur outside the filter range so as not to influence the filter roll-off. The other is the input capacitance ($C_A$) of the amplifier. For some operational amplifiers, $C_A$ can be as high as 50 picofarads, which may be too high for the network.

After the amplifier parameters are selected and fed into the computer, the program prints out guide values that are used to determine how much more the parameters can be increased or decreased and still meet the network require-

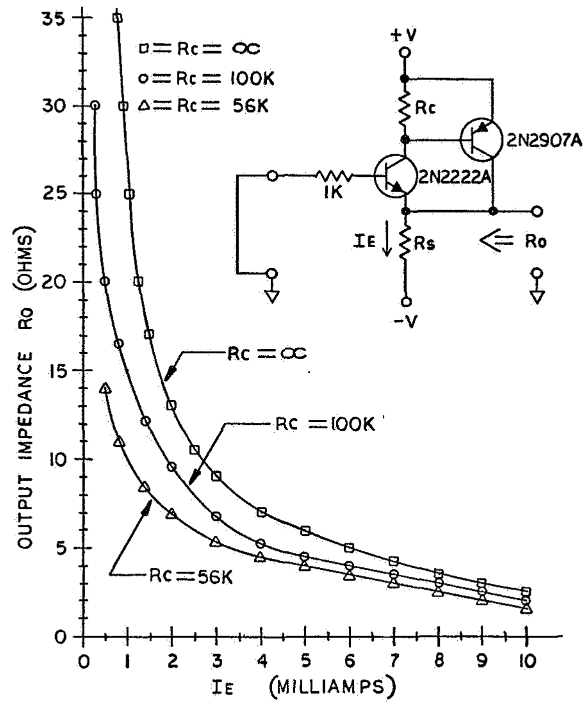



Figure 6-6.—Amplifier output resistance chart

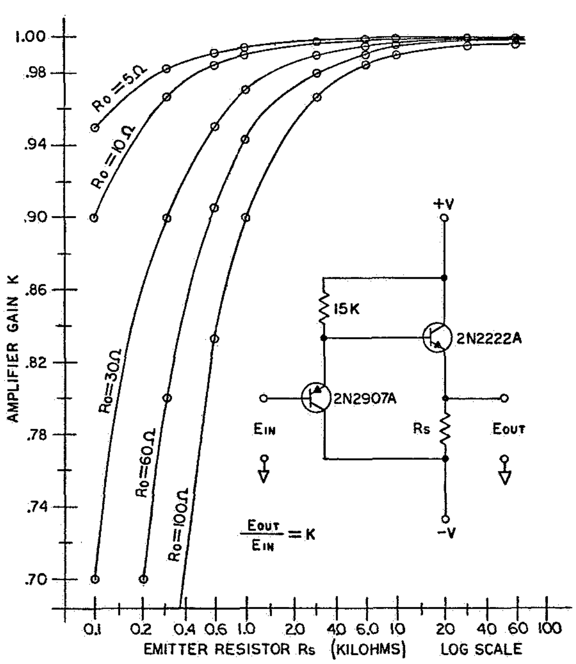



Figure 6–5.—Amplifier gain chart

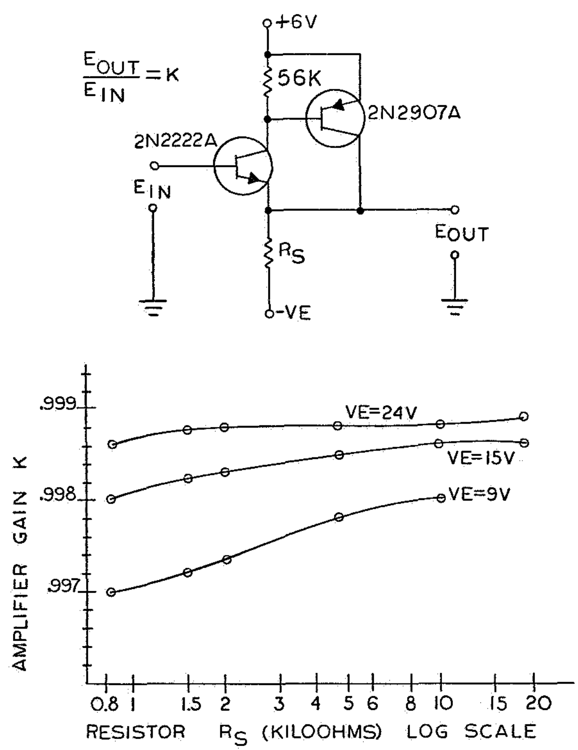



Figure 6–7.—Amplifier gain chart

ments. If unacceptable parameter values are used, the word "error" will appear in front of the printout for that network in which the values are unacceptable.

When either of several amplifier types can be used to satisfy the network requirements, the amplifier can be selected on the basis of its power consumption. Each type of amplifier has a different power requirement range. As the power level of a particular amplifier is decreased, its parameter values change such that higher valued resistors and capacitors are required to realize the network. Typically, a broad range of component values is acceptable, and the power consumption can be held to a low value. Power requirements for each active quadratic section range from 2 to 150 milliwatts. A more typical range would be 10 to 60 milliwatts.

### Connection of Cascaded Networks

The quadratic networks are connected in cascade with the output of one network feeding directly into the input of the following network. It is important to assure that the amplifiers are biased properly. The input to the amplifier of the lowpass quadratic is dc coupled and can therefore obtain its bias from the preceding stage. The input to the amplifier of the highpass network is ac coupled and therefore must supply its own bias.

The quadratic networks can be connected in any sequence. Any lowpass or highpass section can be connected to any other lowpass or highpass section. However, certain guidelines should be followed. The output resistance of one network is the source resistance ($R_g$) for the following network. It should be noted that the output resistance of a network is usually much higher than the output resistance of the amplifier itself. For values of the coefficient $B$ below 1.414, the output impedance peaks sharply at the natural frequency of the network. The lower the value of $B$, the higher the output impedance. At the peak value, the output impedance is purely resistive. Graphs of output impedance versus frequency for various values of $B$ are shown in figures 6–8 through 6–11 for lowpass and highpass quadratic networks. The frequency scale is normalized to have the natural frequency ($F_n$) always appear at unity.

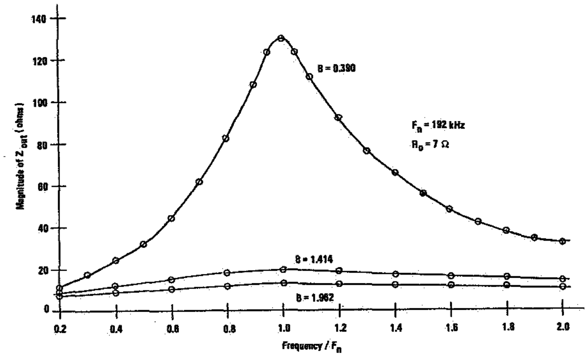



FIGURE 6–8.—Magnitude of output impedance for lowpass second order networks

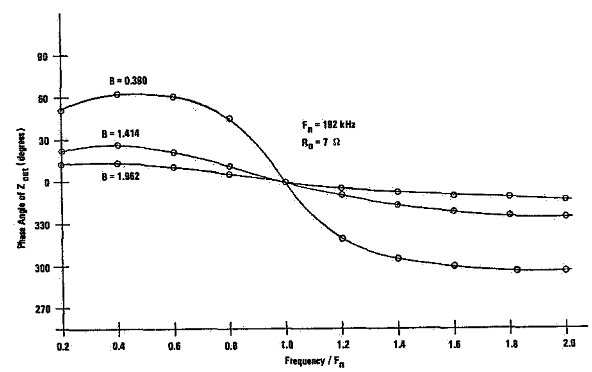



FIGURE 6–9.—Phase angle of output impedance for lowpass second order networks

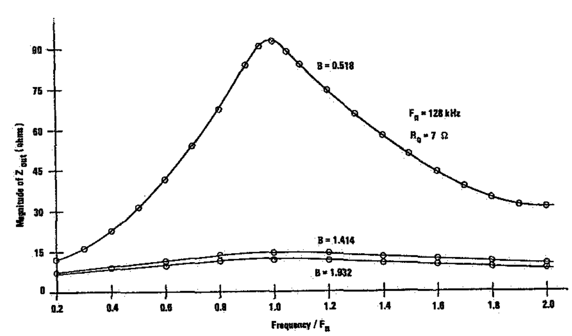



FIGURE 6–10.—Magnitude of output impedance for highpass second order networks

The magnitude of the output impedance varies from the value of amplifier output impedance up to a peak value at its natural frequency and then back down. The computer program prints out the peak value of the output impedance ($Z_o$) for each quadratic network. If a network

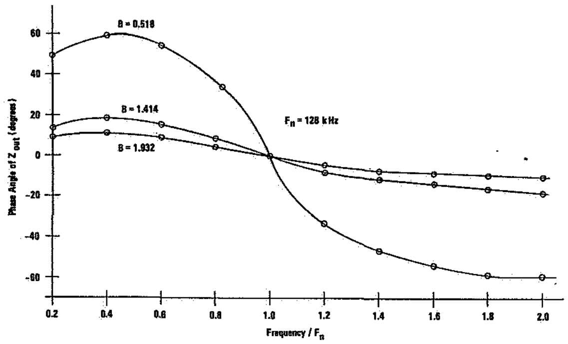



FIGURE 6–11.—Phase angle of output impedance for highpass second order networks

has the same natural frequency as the preceding connected network, then the source resistance ($R_g$) is made equal to $Z_o$ of the preceding network. If the network has a natural frequency different from the preceding network, the value of $R_g$ at the network's natural frequency is roughly estimated from $Z_o$ of the preceding network by referring to the output impedance curves. The natural frequency is the most critical frequency for the network, and it is at this frequency that the network should be defined accurately.

One quantity not considered in the model of the quadratic network is its output load. The effect of the load was not included because it was found to complicate excessively the synthesis equations and did not normally need to be considered. It does need to be considered when the input impedance of the following network is less than five times the output impedance of the preceding or driving network. By proper sequencing of the networks, this case can usually be avoided; if not, the load on the network must be reduced by using a buffer amplifier, usually an emitter-follower, at the output. An emitter-follower buffer consisting of a PNP transistor can very conveniently be added to the output of the amplifier of figure 6–3(B). Here, the combination of the amplifier and buffer has no dc offset voltage between input and output. It should be noted that when a buffer is used, the feedback component that is connected to the output of the amplifier remains connected there and is not moved to the output of the buffer.



The value of the input impedance ($Z_{in}$) of each network at its natural frequency is supplied by the programs. Graphs of input impedance versus normalized frequency are shown in figures 6–12 through 6–15 for values of the coefficient $B$ above and below 1.414 for lowpass and highpass quadratic networks.

If a first-order network is required, it should be connected to the output of the last second-order network. The value of $R_g$ for the first-order network is determined by the same method as for the quadratic network. The first-order network, however, considers the effects of a load resistance ($R_L$) at its output. The program supplies the maximum possible value for $R_g$ and the minimum value of $R_L$ for each first-order network.

## 6.2 FILTER DESIGN PROGRAM ML6

### Purpose

The purpose of ML6 is the active synthesis of a filter. The program supplies a typical set of acceptable amplifier-dependent parameters and calculates the component values of the networks used to realize the filter along with the response curves of the networks.

### Description

The program covers Butterworth and Chebyshev-type filters in the lowpass, highpass, and even-order bandpass configurations.

The synthesis portion of the program calculates $R_1$, $R_2$, $C_1$, and $C_2$ for the networks shown in figure 6–2. Note that $R_g$, $R_O$ and $C_A$ are internal parameters of the amplifier and are not

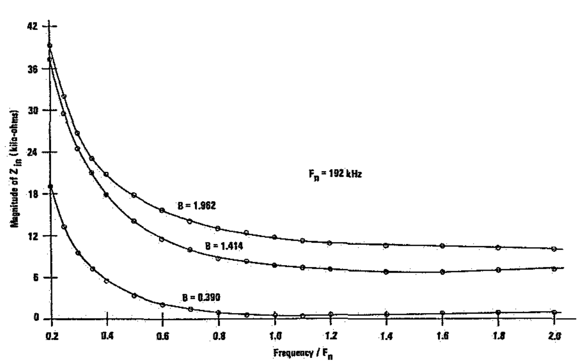



FIGURE 6–12.—Magnitude of input impedance for lowpass second order networks

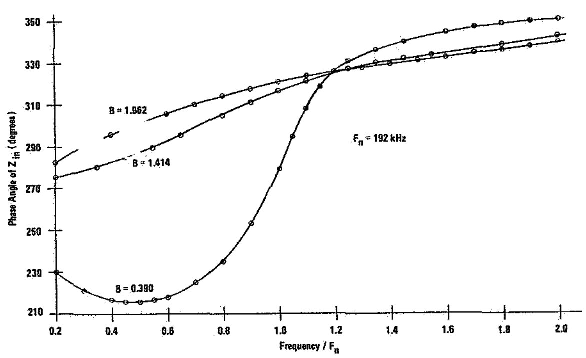



FIGURE 6–13.—Phase angle of input impedance for lowpass second order networks

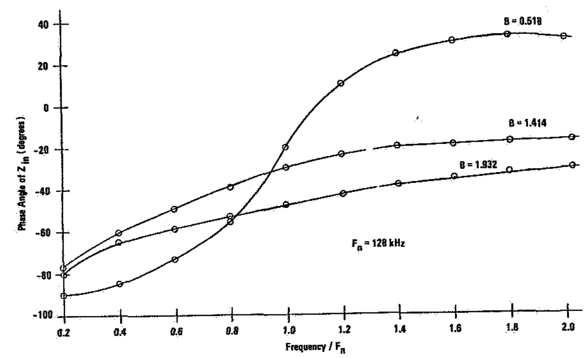



FIGURE 6–15.—Phase angle of input impedance for highpass second order networks

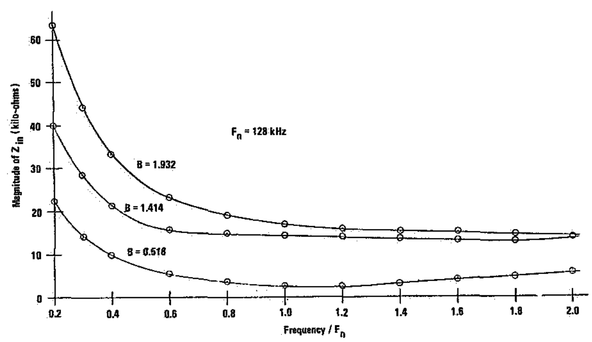



FIGURE 6–14.—Magnitude of input impedance for highpass second order networks

externally added components. Also calculated are the input and output impedances of the networks along with limits on the amplifier parameters that assist in selection of the type amplifier required.

The analysis portion of the program calculates the frequency response for each network and the total responses of the combined networks. The response curves are calculated for amplitude, phase, time delay, and group delay. These response curves are given for the actual networks and can differ slightly from the theoretical curves obtained from program ML1A.

The program has the ability to predistort the filter's pole-zero plot to compensate for the dissipative effects of low-$Q$ capacitors. The predistortion occurs automatically when a value for $Q$ is specified for the capacitors.

The input data required for filter synthesis are entered on two cards and include order of the filter, passband ripple, quality factor of the capacitors, configuration factor, center frequency for bandpass or 3-db cutoff frequency for lowpass and highpass, and bandwidth. The input data required for the analysis portion are entered on a third card and include the maximum frequency of interest, the minimum frequency, the size of the frequency step between successive frequencies of interest, and a magnitude scale factor that divides into the actual amplitude response values. If the scale factor is set equal to the maximum amplitude value, obtained from a previous run, the new maximum amplitude value will be unity. If the response curves are not desired, the third data card is left blank.

A bandpass filter can be realized with either of two different sets of networks. The input-to-output voltage gain of the combined networks is much greater for one set than for the other. Both of these sets of networks are printed out for the bandpass case so that the operator can make his own choice. The two sets of bandpass networks are labeled "High Gain" and "Low Gain." A separate set of response curves is calculated for each case. Typically, the low-gain filter will have a voltage gain between 0.5 and 2, while the high-gain filter will have a voltage gain between 2 and 15. Usually, it is more desirable to choose the low-gain networks because they have less stringent amplifier parameter limits. Also, lower power supply voltages can be used since the signal amplitude is lower.

### Operation

To use ML6, the input data should be punched on three cards with the format given in table VIII. Any number of sets of these three cards can be stacked at the end of the program deck. Two blank cards must be placed behind the last data card.

TABLE VIII.—*ML6 Input Data Format*

| Card | Parameter | Column |
|---|---|---|
| No. 1 | Filter order | 4 and 5 |
| | Passband ripple (db) | 6–20 |
| | $Q$ factor | 21–30 |
| No. 2 | Configuration factor | 1–10 |
| | Center or cutoff frequency (Hz) | 11–20 |
| | Bandwidth (Hz) | 21–30 |
| No. 3 | Maximum frequency (Hz) | 1–10 |
| | Minimum frequency (Hz) | 11–20 |
| | Frequency step (Hz) | 21–30 |
| | Amplitude scale factor | 31–40 |

If there is to be no predistortion, the columns allocated for the $Q$ factor should be left blank; if there is to be no scaling of the amplitude response, the columns allocated for the amplitude scale factor should be left blank. The filter order is a fixed-point number; all other input data are floating-point numbers.

The configuration factor is defined as follows:

$$\text{Configuration factor}=\begin{cases}1. \text{ Lowpass}\\2. \text{ Highpass}\\3. \text{ Bandpass}\end{cases}$$

The subroutines called by this program are CARIP, CAPOO, CAPOE, CMPRL, CMPRH, CPFRL, CPFRH, BPBCW, CARSP, PDSEV, and PSDOD.

### Output Format

A sample computer printout of the first two pages for a lowpass, fifth-order Butterworth filter having a 3-db cutoff frequency at 100 kHz is shown in figure 6–16. The printout for a second-order, bandpass Butterworth having a bandwidth of 40 kHz at a center frequency of 45.826 kHz is shown in figure 6–17. The synthesis section and response curve section are shown for both examples. Only the first page of the response curve values is shown.

First shown is a printout of the input data in the same arrangement as punched on the input data cards. Next are the parameter values for each type network needed to synthesize the filter. Each network is defined by the coefficient $B$, which is equal to twice the damping ratio, and the natural frequency $F$. FLIM is the frequency at which the amplitude roll-off slows down and begins to deviate severely from the theoretical response.

The discrete network components $R_1$, $R_2$, $C_1$, and $C_2$ and the amplifier parameters $K$, $R_O$, and $C_A$ are for the networks shown in figure 6–2; $R_g$ is the output resistance of the preceding network and is not added as a separate component. In simpler terms, the calculated values of $R_1$, $R_2$, $C_1$, and $C_2$ can be considered to belong to the ideal networks of figure 6–1 where the parameters $K$, $R_O$, $R_g$, and $C_A$ are internal to the networks.

$K_{min}$, $K_{max}$, and $R_{Omax}$ are the limits on the amplifier parameters $K$ and $R_O$; $Z_o$ and $Z_{in}$ are the output resistance and the magnitude of input impedance at the natural frequency.

CMIN is the minimum allowable value of $C_1$ and $C_2$ for the highpass second-order networks. A and BCAL are program checkout parameters; A should always be equal to unity, and BCAL should equal B.

C2M is a guide parameter that must always be greater than $C_2$. The amount that C2M exceeds $C_2$ gives an indication of how much $C_1$ can be decreased and still satisfy the network requirements. For the minimum value of $C_1$, C2M and $C_2$ are equal.

The format of the equations used to calculate $R_1$ and $R_2$ is listed last. The quantity within the square root (SQRT) sections must always be equal to, or greater than, zero. The amount that this quantity exceeds zero gives an indication of how much the calculated amplifier parameter limits can be exceeded. The value of these guide parameters will become more apparent when using the next computer program ML7, which has the same output format as this program.

The limits RGMAX and RLMIN for the first-order networks are for the value of $C_1$ used.

```
                         ACTIVE FILTER DESIGN PROGRAM  ML 6

     INPUT DATA

     N =    5           R(DB) = 0.00000E 00          Q = 0.00000E 00
    YY = 0.10000E 01       FO = 0.10000E 06         BW = 0.00000E 00
  FMAX = 0.20000E 06     FMIN = 0.40000E 05         DF = 0.10000E 05      SCAL = 0.00000E 00


  I = 1       LOW-PASS  SECOND ORDER

       B = 0.61803E 00          F = 0.99999E 05          Q = 0.10000E 26       FLIM = 0.70356E 06
       K = 0.97877E 00         RO = 0.64048E 02         RG = 0.00000E 00         CA = 0.10000E-10
      R1 = 0.31129E 04         R2 = 0.32081E 04         C1 = 0.22159E-08         C2 = 0.99999E-10

    KMIN = 0.97877E 00       KMAX = 0.11273E 01      ROMAX = 0.64048E 02        C2M = 0.10442E-09
    BCAL = 0.61803E 00          A = 0.99999E 00         ZO = 0.49444E 03        ZIN = 0.20546E 04

      R1 =  X1 + Z1   =   X1 + SQRT( X1**2 - Y1 )
      X1 = 0.26705E 04         Z1 = 0.44234E 03      X1**2 = 0.71320E 07         Y1 = 0.69363E 07

      R2 =  X2 - Z2   =   X2 - SQRT( X2**2 - Y2 )
      X2 = 0.38396E 04         Z2 = 0.63144E 03      X2**2 = 0.14742E 08         Y2 = 0.14343E 08


  I = 2       LOW-PASS  SECOND ORDER

       B = 0.16180E 01          F = 0.99999E 05          Q = 0.10000E 26       FLIM = 0.95317E 06
       K = 0.90000E 00         RO = 0.10000E 03         RG = 0.00000E 00         CA = 0.10000E-10
      R1 = 0.99092E 04         R2 = 0.11541E 05         C1 = 0.19764E-09         C2 = 0.99999E-10

    KMIN = 0.73819E 00       KMAX = 0.18726E 01      ROMAX = 0.38144E 03        C2M = 0.10000E-09
    BCAL = 0.16180E 01          A = 0.10000E 01         ZO = 0.16767E 03        ZIN = 0.13717E 05

      R1 =  X1 + Z1   =   X1 + SQRT( X1**2 - Y1 )
      X1 = 0.98387E 04         Z1 = 0.70427E 02      X1**2 = 0.96801E 08         Y1 = 0.96796E 08

      R2 =  X2 - Z2   =   X2 - SQRT( X2**2 - Y2 )
      X2 = 0.11624E 05         Z2 = 0.82945E 02      X2**2 = 0.13512E 09         Y2 = 0.13512E 09


  I = 3       LOW-PASS  FIRST ORDER

       B = 0.10000E 01          F = 0.10000E 06          W = 0.62831E 06          Q = 0.10000E 26
      R1 = 0.15172E 04         C1 = 0.10000E-08         RG = 0.10000E 03         RL = 0.10000E 06

   C1MAX = 0.79577E-08      RGMAX = 0.10020E 03      RLMIN = 0.31831E 04


  FILTER RESPONSE CURVES


   I        A            B            C            G            D            E            F

   1   0.99999E 00  0.61803E 00  0.10000E 01  0.19773E-01  0.89176E-01  0.97877E 00  0.10000E 06
   2   0.10000E 01  0.16180E 01  0.10000E 01  0.99059E-02  0.12418E-01  0.90000E 00  0.10000E 06
   3   0.00000E 00  0.10000E 01  0.10000E 01  0.00000E 00  0.00000E 00  0.98408E 00  0.10000E 06

FREQ =  0.400000E 05
   I = 1     A=  0.11149E 01    P= -0.14305E 02    P/W= -0.99342E-06    D=  0.13419E-05
   I = 2     A=  0.84724E 00    P= -0.37297E 02    P/W= -0.25900E-05    D=  0.26344E-05
   I = 3     A=  0.91369E 00    P= -0.21801E 02    P/W= -0.15139E-05    D=  0.13720E-05
            AT=  0.86310E 00   PT= -0.73403E 02   PT/W= -0.50974E-05   DT=  0.53484E-05

FREQ =  0.500000E 05
   I = 1     A=  0.12017E 01    P= -0.19771E 02    P/W= -0.10984E-05    D=  0.17217E-05
   I = 2     A=  0.81359E 00    P= -0.46771E 02    P/W= -0.25984E-05    D=  0.26228E-05
   I = 3     A=  0.88019E 00    P= -0.26565E 02    P/W= -0.14758E-05    D=  0.12732E-05
            AT=  0.86063E 00   PT= -0.93107E 02   PT/W= -0.51726E-05   DT=  0.56178E-05

FREQ =  0.600000E 05
   I = 1     A=  0.13156E 01    P= -0.26936E 02    P/W= -0.12470E-05    D=  0.22973E-05
   I = 2     A=  0.77095E 00    P= -0.56129E 02    P/W= -0.25985E-05    D=  0.25680E-05
   I = 3     A=  0.84384E 00    P= -0.30963E 02    P/W= -0.14335E-05    D=  0.11702E-05
            AT=  0.85591E 00   PT= -0.11402E 03   PT/W= -0.52791E-05   DT=  0.60356E-05

FREQ =  0.700000E 05
   I = 1     A=  0.14520E 01    P= -0.36621E 02    P/W= -0.14532E-05    D=  0.31280E-05
   I = 2     A=  0.72067E 00    P= -0.65202E 02    P/W= -0.25874E-05    D=  0.24645E-05
   I = 3     A=  0.80619E 00    P= -0.34992E 02    P/W= -0.13885E-05    D=  0.10681E-05
            AT=  0.84365E 00   PT= -0.13681E 03   PT/W= -0.54292E-05   DT=  0.66607E-05

FREQ =  0.800000E 05
   I = 1     A=  0.15839E 01    P= -0.49717E 02    P/W= -0.17263E-05    D=  0.41626E-05
   I = 2     A=  0.66518E 00    P= -0.73821E 02    P/W= -0.25632E-05    D=  0.23171E-05
   I = 3     A=  0.76844E 00    P= -0.38659E 02    P/W= -0.13423E-05    D=  0.97045E-06
            AT=  0.80965E 00   PT= -0.16219E 03   PT/W= -0.56319E-05   DT=  0.74502E-05

FREQ =  0.900000E 05
   I = 1     A=  0.16436E 01    P= -0.66375E 02    P/W= -0.20486E-05    D=  0.50018E-05
   I = 2     A=  0.60742E 00    P= -0.81848E 02    P/W= -0.25261E-05    D=  0.21386E-05
   I = 3     A=  0.73146E 00    P= -0.41987E 02    P/W= -0.12959E-05    D=  0.87931E-06
            AT=  0.73028E 00   PT= -0.19021E 03   PT/W= -0.58707E-05   DT=  0.80198E-05

FREQ =  0.100000E 06
   I = 1     A=  0.15584E 01    P= -0.84687E 02    P/W= -0.23524E-05    D=  0.49975E-05
   I = 2     A=  0.55016E 00    P= -0.89200E 02    P/W= -0.24778E-05    D=  0.19445E-05
   I = 3     A=  0.69585E 00    P= -0.45000E 02    P/W= -0.12500E-05    D=  0.79577E-06
            AT=  0.59660E 00   PT= -0.21888E 03   PT/W= -0.60802E-05   DT=  0.77379E-05
```

FIGURE 6-16.—Sample output of program ML6 for a fifth order lowpass Butterworth filter

For a different value of $C_1$, these limits will change; C1MAX is the maximum allowable value of $C_1$ for the lowpass first-order networks. If any of the calculations for a network are in error, the word "error" will appear on the left side of the network number.

The filter response curves are calculated from the coefficients of the network voltage transfer ratios that have the general form

$$\frac{E_{out}}{E_{in}}=\frac{GS^2+DS+E}{AS^2+BS+C},$$

where $S=j(\omega/\omega_n)$ and $\omega_n$ is the natural frequency in radians per second. The coefficients $A$, $B$, $C$, $G$, $D$, and $E$ are printed out for each network along with its natural frequency $F$ in hertz.

The response curves are calculated for each network from the lowest frequency of interest through the highest frequency of interest. The symbol representation used is:

$A$ =Amplitude
$P$ =Phase angle in degrees
$P/W$=Time delay in seconds
$D$ =Group delay in seconds

$AT$, $PT$, $PT/W$, and $DT$ are the total responses of the combined networks and closely approximate the theoretical filter response.

## 6.3 FILTER SYNTHESIS PROGRAM ML7

### Purpose

The purpose of ML7 is to synthesize active filter networks from the characteristics of the networks voltage transfer function. The operator supplies the coefficient $B$, the natural frequency, and the amplifier parameters; the program calculates the component values and the response curves of the network.

### Description

The networks covered by the program, along with their ideal voltage transfer functions, are:

$$\text{First-order lowpass } \frac{E}{BS+1},$$

$$\text{First-order highpass } \frac{DS}{BS+1},$$

$$\text{Second-order lowpass } \frac{E}{S^2+BS+1},$$

$$\text{Second-order highpass } \frac{GS^2}{S^2+BS+1},$$

where $S=j(\omega/\omega_n)$, where $\omega_n$ is the natural frequency in radians per second. As can be seen, $B$ is the coefficient of the $S$ term in the denominator of the first order transfer functions and is equal to twice the damping ratio for the second-order functions.

The synthesis portion of the program allows the operator to supply the values of the coefficient $B$, the natural frequency $F$, the amplifier parameters $K$, $R_O$, and $C_A$, the input resistance $R_g$, and the capacitor $C_2$ for the second-order networks or capacitor $C_1$ for the first-order networks. These parameters are for the networks shown in figure 6–2. The parameters required can be obtained from program ML6, which calculates these parameters from the filter specifications.

Program ML7 also calculates guide parameters and limits for the amplifier parameters that assist in obtaining the optimum set of parameter values.

The four networks considered can be used to synthesize filters in the lowpass, highpass, bandpass, and bandstop configurations.

The analysis portion of the program calculates the frequency responses for each network and the total responses of the combined networks. The response curves are calculated for amplitude, phase, time delay, and group delay.

The input data required for network synthesis are the values of $B$, $F$, $R_g$, $R_O$, $K$, $C_A$, and $C_2$ for the second-order networks and $B$, $F$, $R_g$, $R_L$ and $C_1$ for the first-order networks. For the second-order highpass networks, $C_1$ and $C_2$ are always equal. Each parameter has an assigned code number that allows the computer to identify it. This allows the computer to transfer the parameter value from one network case to another until it is modified by the operator.

The input data required for the response curves are entered on a single card that is inserted behind all the network data cards for which the response curves pertain. The data entered on the response analysis card include a code number that calls the analysis subprogram, the maximum frequency of interest, the minimum frequency, the size of the frequency step

```
                     ACTIVE FILTER DESIGN PROGRAM   ML 6

     INPUT DATA

   N =    2           R(DB) = 0.00000E 00        Q = 0.00000E 00
  YY = 0.30000E 01       FO = 0.45826E 05       BW = 0.40000E 05
FMAX = 0.15000E 06     FMIN = 0.10000E 05       DF = 0.10000E 05      SCAL = 0.00000E 00


          BAND-PASS FILTER NETWORKS ( HIGH GAIN )

I = 1     LOW-PASS  SECOND ORDER

     B = 0.58735E 00          F = 0.62949E 05        Q = 0.10000E 26     FLIM = 0.46551E 06
     K = 0.98083E 00         RO = 0.87152E 02       RG = 0.00000E 00       CA = 0.10000E-10
    R1 = 0.46874E 04         R2 = 0.48233E 04       C1 = 0.24793E-08       C2 = 0.99999E-10

  KMIN = 0.98083E 00       KMAX = 0.11149E 01    ROMAX = 0.87152E 02      C2M = 0.10455E-09
  BCAL = 0.58735E 00          A = 0.10000E 01       ZO = 0.74347E 03      ZIN = 0.29485E 04

    R1 =  X1 + Z1  =  X1 + SQRT( X1**2 - Y1 )
    X1 = 0.40146E 04         Z1 = 0.67275E 03    X1**2 = 0.16117E 08       Y1 = 0.15665E 08

    R2 =  X2 - Z2  =  X2 - SQRT( X2**2 - Y2 )
    X2 = 0.57866E 04         Z2 = 0.96338E 03    X2**2 = 0.33485E 08       Y2 = 0.32557E 08


I = 2     HIGH-PASS  SECOND ORDER

     B = 0.58736E 00          F = 0.33360E 05        Q = 0.10000E 26     FLIM = 0.35999E 03
     K = 0.98083E 00         RO = 0.10000E 03       RG = 0.10000E 03       CA = 0.10000E-10
    R1 = 0.17271E 04         R2 = 0.50672E 05       C1 = 0.48575E-09       C2 = 0.48575E-09

  KMIN = 0.98083E 00       KMAX = 0.11149E 01    ROMAX = 0.32048E 03     CMIN = 0.48575E-09
  BCAL = 0.58736E 00          A = 0.10000E 01       ZO = 0.95809E 03      ZIN = 0.50483E 04

    R1 =  X1 + Z1  =  X1 + SQRT( X1**2 - Y1 )
    X1 = 0.13191E 04         Z1 = 0.40797E 03    X1**2 = 0.17402E 07       Y1 = 0.15738E 07

    R2 =  X2 - Z2
    X2 = 0.50763E 05         Z2 = 0.90894E 02


FILTER RESPONSE CURVES     *


 I      A             B             C             G             D             E             F

 1   0.10000E 01   0.58735E 00   0.10000E 01   0.17935E-01   0.85466E-01   0.98083E 00   0.62949E 05
 2   0.10000E 01   0.58736E 00   0.10000E 01   0.94353E 00   0.10181E-01   0.00000E 00   0.33360E 05

FREQ =  0.100000E 05
  I = 1    A=  0.10012E 01    P= -0.46743E 01    P/W= -0.12984E-05    D=  0.13672E-05
  I = 2    A=  0.91513E-01    P=  0.16698E 03    P/W=  0.46386E-04    D=  0.29815E-05
          AT=  0.91631E-01   PT=  0.16231E 03   PT/W=  0.45087E-04   DT=  0.43487E-05

FREQ =  0.200000E 05
  I = 1    A=  0.10666E 01    P= -0.10137E 02    P/W= -0.14079E-05    D=  0.17177E-05
  I = 2    A=  0.46399E 00    P=  0.15017E 03    P/W=  0.20857E-04    D=  0.69857E-05
          AT=  0.49491E 00   PT=  0.14003E 03   PT/W=  0.19449E-04   DT=  0.87034E-05

FREQ =  0.300000E 05
  I = 1    A=  0.11892E 01    P= -0.17521E 02    P/W= -0.16223E-05    D=  0.24742E-05
  I = 2    A=  0.13583E 01    P=  0.10922E 03    P/W=  0.10113E-04    D=  0.15995E-04
          AT=  0.16154E 01   PT=  0.91701E 02   PT/W=  0.84909E-05   DT=  0.18470E-04

FREQ =  0.400000E 05
  I = 1    A=  0.13862E 01    P= -0.28852E 02    P/W= -0.20036E-05    D=  0.39886E-05
  I = 2    A=  0.16360E 01    P=  0.57625E 02    P/W=  0.40017E-05    D=  0.98993E-05
          AT=  0.22679E 01   PT=  0.28772E 02   PT/W=  0.19981E-05   DT=  0.13888E-04

FREQ =  0.500000E 05
  I = 1    A=  0.16337E 01    P= -0.47644E 02    P/W= -0.26469E-05    D=  0.66166E-05
  I = 2    A=  0.13890E 01    P=  0.34822E 02    P/W=  0.19345E-05    D=  0.38840E-05
          AT=  0.22693E 01   PT= -0.12822E 02   PT/W= -0.71236E-06   DT=  0.10500E-04

FREQ =  0.600000E 05
  I = 1    A=  0.17063E 01    P= -0.75888E 02    P/W= -0.35133E-05    D=  0.85773E-05
  I = 2    A=  0.12347E 01    P=  0.24956E 02    P/W=  0.11554E-05    D=  0.19262E-05
          AT=  0.21069E 01   PT= -0.50931E 02   PT/W= -0.23579E-05   DT=  0.10503E-04

FREQ =  0.700000E 05
  I = 1    A=  0.13868E 01    P= -0.10424E 03    P/W= -0.41367E-05    D=  0.66492E-05
  I = 2    A=  0.11478E 01    P=  0.19614E 02    P/W=  0.77837E-06    D=  0.11441E-05
          AT=  0.15918E 01   PT= -0.84632E 02   PT/W= -0.33584E-05   DT=  0.77934E-05

FREQ =  0.800000E 05
  I = 1    A=  0.99051E 00    P= -0.12297E 03    P/W= -0.42701E-05    D=  0.39133E-05
  I = 2    A=  0.10950E 01    P=  0.16256E 02    P/W=  0.56447E-06    D=  0.76149E-06
          AT=  0.10846E 01   PT= -0.10672E 03   PT/W= -0.37056E-05   DT=  0.46748E-05

FREQ =  0.900000E 05
  I = 1    A=  0.71053E 00    P= -0.13381E 03    P/W= -0.41301E-05    D=  0.22753E-05
  I = 2    A=  0.10605E 01    P=  0.13936E 02    P/W=  0.43013E-06    D=  0.54620E-06
          AT=  0.75357E 00   PT= -0.11988E 03   PT/W= -0.37000E-05   DT=  0.28215E-05

FREQ    0.100000E 06
  I = 1    A=  0.52915E 00    P= -0.14025E 03    P/W= -0.38960E-05    D=  0.13912E-05
  I = 2    A=  0.10367E 01    P=  0.12227E 02    P/W=  0.33965E-06    D=  0.41272E-06
          AT=  0.54862E 00   PT= -0.12803E 03   PT/W= -0.35564E-05   DT=  0.18039E-05
```

FIGURE 6–17.—Sample output of program ML6 for a second order bandpass Butterworth filter

```
              BAND-PASS FILTER NETWORKS ( LOW GAIN )

I = 1         LOW-PASS  SECOND ORDER

       B = 0.58736E 00         F = 0.33360E 05         Q = 0.10000E 26      FLIM = 0.32500E 06
       K = 0.98083E 00        RO = 0.10000E 03        RG = 0.00000E 00        CA = 0.10000E-10
      R1 = 0.94901E 04        R2 = 0.11004E 05        C1 = 0.19431E-08        C2 = 0.99999E-10

    KMIN = 0.98083E 00      KMAX = 0.11149E 01     ROMAX = 0.16445E 03       C2M = 0.10079E-09
    BCAL = 0.58736E 00         A = 0.10000E 01        ZO = 0.73853E 03       ZIN = 0.59286E 04

      R1 =  X1 + Z1   =  X1 + SQRT( X1**2 - Y1 )
      X1 = 0.88429E 04        Z1 = 0.64720E 03     X1**2 = 0.78198E 08        Y1 = 0.77779E 08

      R2 =  X2 - Z2   =  X2 - SQRT( X2**2 - Y2 )
      X2 = 0.11870E 05        Z2 = 0.86631E 03     X2**2 = 0.14091E 09        Y2 = 0.14016E 09


I = 2         HIGH-PASS  SECOND ORDER

       B = 0.58735E 00         F = 0.62949E 05         Q = 0.10000E 26      FLIM = 0.12790E 04
       K = 0.98083E 00        RO = 0.10000E 03        RG = 0.10000E 03        CA = 0.10000E-10
      R1 = 0.81167E 03        R2 = 0.28521E 05        C1 = 0.48575E-09        C2 = 0.48575E-09

    KMIN = 0.98083E 00      KMAX = 0.11149E 01     ROMAX = 0.16984E 03      CMIN = 0.48575E-09
    BCAL = 0.58735E 00         A = 0.10000E 01        ZO = 0.10085E 04       ZIN = 0.26754E 04

      R1 =  X1 + Z1   =  X1 + SQRT( X1**2 - Y1 )
      X1 = 0.64129E 03        Z1 = 0.17037E 03     X1**2 = 0.41125E 06        Y1 = 0.38222E 06

      R2 =  X2 - Z2
      X2 = 0.28606E 05        Z2 = 0.85709E 02


FILTER RESPONSE CURVES


 I        A             B             C             G             D             E             F

 1   0.10000E 01   0.58736E 00   0.10000E 01   0.10334E-01   0.40729E-01   0.98083E 00   0.33360E 05
 2   0.10000E 01   0.58735E 00   0.10000E 01   0.94559E 00   0.19212E-01   0.00000E 00   0.62949E 05


FREQ =  0.100000E 05
   I = 1     A=  0.10571E 01    P= -0.10234E 02    P/W= -0.28429E-05    D=  0.33550E-05
   I = 2     A=  0.24567E-01    P=  0.16724E 03    P/W=  0.46456E-04    D= -0.41506E-06
            AT=  0.25971E-01   PT=  0.15700E 03   PT/W=  0.43613E-04   DT=  0.29400E-05

FREQ =  0.200000E 05
   I = 1     A=  0.13371E 01    P= -0.27366E 02    P/W= -0.38008E-05    D=  0.69286E-05
   I = 2     A=  0.10416E 00    P=  0.16461E 03    P/W=  0.22863E-04    D=  0.14322E-05
            AT=  0.13928E 00   PT=  0.13724E 03   PT/W=  0.19062E-04   DT=  0.83609E-05

FREQ =  0.300000E 05
   I = 1     A=  0.17323E 01    P= -0.67932E 02    P/W= -0.62900E-05    D=  0.15856E-04
   I = 2     A=  0.26150E 00    P=  0.15764E 03    P/W=  0.14597E-04    D=  0.24711E-05
            AT=  0.45300E 00   PT=  0.89717E 02   PT/W=  0.83071E-05   DT=  0.18327E-04

FREQ =  0.400000E 05
   I = 1     A=  0.11664E 01    P= -0.11896E 03    P/W= -0.82614E-05    D=  0.97283E-05
   I = 2     A=  0.54306E 00    P=  0.14612E 03    P/W=  0.10147E-04    D=  0.40860E-05
            AT=  0.63347E 00   PT=  0.27158E 02   PT/W=  0.18859E-05   DT=  0.13814E-04

FREQ =  0.500000E 05
   I = 1     A=  0.62885E 00    P= -0.14111E 03    P/W= -0.78398E-05    D=  0.36950E-05
   I = 2     A=  0.10031E 01    P=  0.12688E 03    P/W=  0.70491E-05    D=  0.67622E-05
            AT=  0.63083E 00   PT= -0.14233E 02   PT/W= -0.79073E-06   DT=  0.10457E-04

FREQ =  0.600000E 05
   I = 1     A=  0.38442E 00    P= -0.15027E 03    P/W= -0.69573E-05    D=  0.17238E-05
   I = 2     A=  0.15147E 01    P=  0.98062E 02    P/W=  0.45399E-05    D=  0.87508E-05
            AT=  0.58229E 00   PT= -0.52215E 02   PT/W= -0.24173E-05   DT=  0.10474E-04

FREQ =  0.700000E 05
   I = 1     A=  0.25951E 00    P= -0.15486E 03    P/W= -0.61456E-05    D=  0.92977E-06
   I = 2     A=  0.16835E 01    P=  0.69044E 02    P/W=  0.27398E-05    D=  0.68412E-05
            AT=  0.43689E 00   PT= -0.85825E 02   PT/W= -0.34057E-05   DT=  0.77710E-05

FREQ =  0.800000E 05
   I = 1     A=  0.18699E 00    P= -0.15743E 03    P/W= -0.54664E-05    D=  0.53499E-06
   I = 2     A=  0.15791E 01    P=  0.49595E 02    P/W=  0.17220E-05    D=  0.41193E-05
            AT=  0.29529E 00   PT= -0.10783E 03   PT/W= -0.37444E-05   DT=  0.46543E-05

FREQ =  0.900000E 05
   I = 1     A=  0.14088E 00    P= -0.15891E 03    P/W= -0.49048E-05    D=  0.30670E-06
   I = 2     A=  0.14427E 01    P=  0.37995E 02    P/W=  0.11726E-05    D=  0.24928E-05
            AT=  0.20326E 00   PT= -0.12092E 03   PT/W= -0.37321E-05   DT=  0.27995E-05

FREQ =  0.100000E 06
   I = 1     A=  0.10961E 00    P= -0.15973E 03    P/W= -0.44371E-05    D=  0.15877E-06
   I = 2     A=  0.13357E 01    P=  0.30751E 02    P/W=  0.85420E-06    D=  0.16189E-05
            AT=  0.14641E 00   PT= -0.12898E 03   PT/W= -0.35829E-05   DT=  0.17777E-05

FREQ =  0.110000E 06
   I = 1     A=  0.87351E-01    P= -0.16011E 03    P/W= -0.40431E-05    D=  0.53637E-07
   I = 2     A=  0.12578E 01    P=  0.25890E 02    P/W=  0.65379E-06    D=  0.11253E-05
```

FIGURE 6-17.—Concluded.

between successive frequencies of interest, and a magnitude scale factor that divides into the actual amplitude response values. If the scale factor is set equal to the maximum amplitude value, obtained from a previous run, the new maximum amplitude value will be unity.

### Operation

To use ML7, the input data cards are prepared in the following format:

(1) *Title card*—Must be present but can be blank; any descriptive information can be entered in columns 1–80

(2) *Network code card*—Calls the desired network; code number used columns 1 and 2.

(3) *Parameter value cards*—Each network parameter value is punched on a separate card along with its code number.

| | |
|---|---|
| Code number | Column 1 |
| Parameter value | Columns 2-12 |

(4) *Blank card*

(5) *Parameter value cards*—For the next case of the same type network; only the parameters that are changed from the preceding case need to be entered. As many cases as desired can be obtained by separating the new parameter values with a blank card.

| | |
|---|---|
| Code number | Column 1 |
| Parameter value | Columns 2–12 |

(6) *Blank card*

(7) *Network exit card*—Exits the called network; exit number is 9 in column 1.

(8) *Network code card*—Calls the next network desired; repeat the previous sequence of cards for as many network types as required. Be sure to end each sequence with a network exit card.

(9) *Response analysis card*—If the response curves are not desired, this card is omitted.

| | *Columns* |
|---|---|
| Code number is 15 | 1 and 2 |
| Maximum frequency (Hz) | 3–12 |
| Minimum frequency (Hz) | 13–22 |
| Frequency step (Hz) | 23–32 |
| Amplitude scale factor | 33–42 |

(10) *Blank card*

(11) *Blank card*

The network code numbers are defined as follows:

*Network Code No.*

| | |
|---|---|
| 11 | Lowpass first order |
| 12 | Lowpass second order |
| 13 | Highpass first order |
| 14 | Highpass second order |
| 15 | Call analysis program |

The parameter code numbers are defined in table IX. These parameters are for the networks of figure 6–2, where $F$ is the resonant frequency in Hz.

TABLE IX.—*Definement of Parameter Code Numbers*

| Parameter Code No. | Lowpass 2nd order | Highpass 2nd order | Lowpass 1st order | Highpass 1st order |
|---|---|---|---|---|
| 1 | B | B | B | B |
| 2 | F | F | F | F |
| 3 | C2 | C2 | C1 | C1 |
| 4 | RG | RG | RG | RG |
| 5 | K | K | RL | RL |
| 6 | RO | RO | | |
| 7 | CA | CA | | |
| 8 | C1 | | | |

The network and parameter code numbers are fixed-point numbers, and all other input data are floating-point numbers. The capacitor values are more conveniently expressed in exponential floating-point form. As an example, 100 picofarads would be expressed as 1.0E–10 or 100.0E–12. When in this form, the number must end in the last column allocated for the parameter. The resistance values are entered in ohms, and the capacitance value in farads.

The value of the parameters, with the exception of $C_1$ and $R_L$, are carried over from one network case to another and from one network type to another and need not be re-entered unless they are changed.

The parameter values of $C_1$ for the lowpass second-order network and $R_L$ for the first-order networks do not have to be specified. The computer will automatically assign the minimum acceptable value for $C_1$ and an infinitely high value for $R_L$. If a standard capacitor value is desired, $C_1$ can be specified at a value above its minimum value.

If a printout of the coefficients of the network transfer functions is desired without having the response curves, the response analysis card is used with number 15 in columns 1 and 2, and the rest of the card left blank. A sample listing of input data cards follows for a fifth-order, lowpass Butterworth filter. All data start in column 1. The computer printout for these input data is shown in figures 6–18 and 6–19.

Title Card
11
1 1.0
2 100000.
3 2200.0E–12
4 34
Blank card
9
12
1 0.618
3 560.00E–12
4 20.
5 0.998
6 15.
7 10.000E–12
8 8200.0E–12
Blank card
1 1.618
4 140.
8 1000.0E–12
Blank card
9
15 200000. 40000. 10000.
Blank card
Blank card

The subroutines called by this program are CASEL, CASEH, CAFIL, CAFIH, and CARSP.

### Output Format

A sample computer printout for a fifth-order' lowpass Butterworth filter having a 3-db cutoff frequency at 100 kHz is shown in figure 6–18. This printout was obtained by using the sample deck of data cards listed in section 6.3.

A second sample printout for a second-order, bandpass Butterworth filter having a bandwidth of 40 kHz at a center frequency of 45.826 kHz is shown in figure 6–19. The input parameter values for both printouts were obtained from the sample printouts of program ML6.

The output format for these programs is the same as for program ML6 with the exception that the $Q$ factor is replaced by $\omega$, which is the natural frequency in radians per second. Rather than repeat the same material here, the reader is requested to refer to program ML6, output format, section 6.2, where the output format is discussed in detail.

The data from these two sample printouts were used to construct active filters of the type indicated in the printout. The responses of the filters were measured and found to coincide closely with the theoretical response. The filter schematics and comparison of the filter response curves are shown as part of the design examples given in section 8.

## 6.4 TEMPERATURE-DEPENDENT FILTER RESPONSE PROGRAM ML8

### Purpose

The purpose of ML8 is to calculate the network response curves for any temperature specified by the operator.

### Description

All the parameters of each network considered are required as input data. Also required are the temperature coefficients of components $R_1$, $R_2$, $C_1$, and $C_2$ in units of percentage of change per degree Celsius for each network. For a specified value of temperature, the program will calculate the response curves for each network and the response curves for the total combined networks. The response curves are calculated for amplitude, phase, time delay, and group delay.

The control frequencies for the response range are punched on a single card in the same format as for program ML6. The data consist of the maximum frequency of interest, the minimum frequency, the size of the frequency step between successive frequencies of interest, and a magnitude scale factor that divides into the actual amplitude response values. If the scale factor is set equal to the maximum amplitude value, obtained from a previous program run, the new maximum amplitude value will be unity. The columns allocated for the scale factor are normally left blank.

```
                         ACTIVE FILTER SYNTHESIS PROGRAM  ML 7

      FIFTH ORDER LOW-PASS BUTTERWORTH FILTER


         LOW-PASS FIRST ORDER FILTER NETWORKS

I = 1

      B = 0.10000E 01         F = 0.10000E 06         W = 0.62831E 06      BCAL = 0.10000E 01
     R1 = 0.68943E 03        C1 = 0.22000E-08        RG = 0.34000E 02        RL = 0.10000E 26

  C1MAX = 0.79577E-08     RGMAX = 0.10000E 03     RLMIN = 0.14468E 04         E = 0.10000E 01


         LOW-PASS SECOND ORDER FILTER NETWORKS

I = 2

      B = 0.61800E 00         F = 0.10000E 06         W = 0.62831E 06      FLIM = 0.79729E 06
      K = 0.99800E 00        RO = 0.15000E 02        RG = 0.20000E 02        CA = 0.10000E-10
     R1 = 0.89620E 03        R2 = 0.56722E 03        C1 = 0.82000E-08        C2 = 0.55999E-09

   KMIN = 0.97878E 00      KMAX = 0.11273E 01     ROMAX = 0.12358E 02       C2M = 0.59874E-09
   BCAL = 0.61799E 00         A = 0.10000E 01        ZO = 0.12746E 03       ZIN = 0.54413E 03

     R1 =  X1 + Z1  =  X1 + SQRT( X1**2 - Y1 )
     X1 = 0.73356E 03        Z1 = 0.18263E 03     X1**2 = 0.53812E 06        Y1 = 0.50477E 06

     R2 =  X2 - Z2  =  X2 - SQRT( X2**2 - Y2 )
     X2 = 0.75510E 03        Z2 = 0.18788E 03     X2**2 = 0.57018E 06        Y2 = 0.53488E 06


I = 3

      B = 0.16180E 01         F = 0.10000E 06         W = 0.62831E 06      FLIM = 0.14335E 07
      K = 0.99800E 00        RO = 0.15000E 02        RG = 0.14000E 03        CA = 0.10000E-10
     R1 = 0.29020E 04        R2 = 0.14387E 04        C1 = 0.10000E-08        C2 = 0.55999E-09

   KMIN = 0.73820E 00      KMAX = 0.18726E 01     ROMAX = 0.73607E 02       C2M = 0.64482E-09
   BCAL = 0.16179E 01         A = 0.99999E 00        ZO = 0.32596E 02       ZIN = 0.33293E 04

     R1 =  X1 + Z1  =  X1 + SQRT( X1**2 - Y1 )
     X1 = 0.22378E 04        Z1 = 0.80418E 03     X1**2 = 0.50079E 07        Y1 = 0.43612E 07

     R2 =  X2 - Z2  =  X2 - SQRT( X2**2 - Y2 )
     X2 = 0.22457E 04        Z2 = 0.80701E 03     X2**2 = 0.50434E 07        Y2 = 0.43921E 07

FILTER RESPONSE CURVES

 I         A            B            C            G            D            E            F

 1    0.00000E 00  0.10000E 01  0.10000E 01  0.00000E 00  0.00000E 00  0.10000E 01  0.10000E 06
 2    0.10000E 01  0.61800E 00  0.10000E 01  0.15699E-01  0.77283E-01  0.99800E 00  0.10000E 06
 3    0.99999E 00  0.16180E 01  0.10000E 01  0.48563E-02  0.94247E-02  0.99800E 00  0.10000E 06

FREQ =  0.400000E 05
   I = 1    A=  0.92847E 00    P= -0.21801E 02    P/W= -0.15139E-05    D=  0.13720E-05
   I = 2    A=  0.11374E 01    P= -0.14619E 02    P/W= -0.10152E-05    D=  0.13640E-05
   I = 3    A=  0.94042E 00    P= -0.37396E 02    P/W= -0.25969E-05    D=  0.26414E-05
           AT=  0.99316E 00   PT= -0.73817E 02   PT/W= -0.51262E-05   DT=  0.53775E-05

FREQ =  0.500000E 05
   I = 1    A=  0.89442E 00    P= -0.26565E 02    P/W= -0.14758E-05    D=  0.12732E-05
   I = 2    A=  0.12264E 01    P= -0.20165E 02    P/W= -0.11203E-05    D=  0.17440E-05
   I = 3    A=  0.90357E 00    P= -0.46896E 02    P/W= -0.26053E-05    D=  0.26299E-05
           AT=  0.99117E 00   PT= -0.93627E 02   PT/W= -0.52015E-05   DT=  0.56471E-05

FREQ =  0.600000E 05
   I = 1    A=  0.85749E 00    P= -0.30963E 02    P/W= -0.14335E-05    D=  0.11702E-05
   I = 2    A=  0.13430E 01    P= -0.27411E 02    P/W= -0.12690E-05    D=  0.23199E-05
   I = 3    A=  0.85679E 00    P= -0.56279E 02    P/W= -0.26055E-05    D=  0.25751E-05
           AT=  0.98677E 00   PT= -0.11465E 03   PT/W= -0.53081E-05   DT=  0.60653E-05

FREQ =  0.700000E 05
   I = 1    A=  0.81923E 00    P= -0.34992E 02    P/W= -0.13885E-05    D=  0.10681E-05
   I = 2    A=  0.14830E 01    P= -0.37178E 02    P/W= -0.14753E-05    D=  0.31510E-05
   I = 3    A=  0.80156E 00    P= -0.65378E 02    P/W= -0.25943E-05    D=  0.24717E-05
           AT=  0.97384E 00   PT= -0.13754E 03   PT/W= -0.54583E-05   DT=  0.66909E-05

FREQ =  0.800000E 05
   I = 1    A=  0.78086E 00    P= -0.38659E 02    P/W= -0.13423E-05    D=  0.97045E-06
   I = 2    A=  0.16185E 01    P= -0.50358E 02    P/W= -0.17485E-05    D=  0.41861E-05
   I = 3    A=  0.74052E 00    P= -0.74023E 02    P/W= -0.25702E-05    D=  0.23244E-05
           AT=  0.93594E 00   PT= -0.16304E 03   PT/W= -0.56611E-05   DT=  0.74810E-05

FREQ =  0.900000E 05
   I = 1    A=  0.74329E 00    P= -0.41987E 02    P/W= -0.12959E-05    D=  0.87931E-06
   I = 2    A=  0.16805E 01    P= -0.67101E 02    P/W= -0.20710E-05    D=  0.50259E-05
   I = 3    A=  0.67693E 00    P= -0.82077E 02    P/W= -0.25332E-05    D=  0.21460E-05
           AT=  0.84556E 00   PT= -0.19116E 03   PT/W= -0.59001E-05   DT=  0.80512E-05

FREQ =  0.100000E 06
   I = 1    A=  0.70710E 00    P= -0.45000E 02    P/W= -0.12500E-05    D=  0.79577E-06
   I = 2    A=  0.15943E 01    P= -0.85501E 02    P/W= -0.23750E-05    D=  0.50222E-05
   I = 3    A=  0.61383E 00    P= -0.89456E 02    P/W= -0.24848E-05    D=  0.19520E-05
           AT=  0.69204E 00   PT= -0.21995E 03   PT/W= -0.61099E-05   DT=  0.77700E-05
```

FIGURE 6-18.—Sample output of program ML7 for a fifth order lowpass Butterworth filter

```
                    ACTIVE FILTER SYNTHESIS PROGRAM   M L  7

      SECOND ORDER BUTTERWORTH BAND-PASS FILTER      HIGH GAIN


         LOW-PASS SECOND ORDER FILTER NETWORKS

I = 1

      B = 0.58736E 00        F = 0.62950E 05        W = 0.39552E 06     FLIM = 0.61394E 06
      K = 0.99800E 00       RO = 0.15000E 02       RG = 0.20000E 02       CA = 0.10000E-10
     R1 = 0.13728E 04       R2 = 0.95659E 03       C1 = 0.82000E-08       C2 = 0.55999E-09

   KMIN = 0.98083E 00     KMAX = 0.11149E 01    ROMAX = 0.16818E 02      C2M = 0.58431E-09
   BCAL = 0.58735E 00        A = 0.99999E 00       ZO = 0.12889E 03      ZIN = 0.78593E 03

     R1 =  X1 + Z1  =  X1 + SQRT( X1**2 - Y1 )
     X1 = 0.11611E 04       Z1 = 0.23169E 03    X1**2 = 0.13481E 07       Y1 = 0.12945E 07

     R2 =  X2 - Z2  =  X2 - SQRT( X2**2 - Y2 )
     X2 = 0.11949E 04       Z2 = 0.23836E 03    X2**2 = 0.14279E 07       Y2 = 0.13711E 07


         HIGH-PASS SECOND ORDER FILTER NETWORKS

I = 2

      B = 0.58736E 00        F = 0.33360E 05        W = 0.20960E 06     FLIM = 0.61044E 02
      K = 0.99800E 00       RO = 0.15000E 02       RG = 0.60000E 02       CA = 0.10000E-10
     R1 = 0.21336E 04       R2 = 0.32562E 05       C1 = 0.56000E-09       C2 = 0.56000E-09

   KMIN = 0.98083E 00     KMAX = 0.11149E 01    ROMAX = 0.27799E 03      CMIN = 0.28298E-09
   BCAL = 0.58735E 00        A = 0.10000E 01       ZO = 0.11232E 03      ZIN = 0.43706E 04

     R1 =  X1 + Z1  =  X1 + SQRT( X1**2 - Y1 )
     X1 = 0.12214E 04       Z1 = 0.91218E 03    X1**2 = 0.14919E 07       Y1 = 0.65989E 06

     R2 =  X2 - Z2
     X2 = 0.32620E 05       Z2 = 0.5753[illegible]E 02


FILTER RESPONSE CURVES


 I         A             B             C             G             D             E             F

 1    0.99999E 00   0.58736E 00   0.10000E 01   0.10492E-01   0.48649E-01   0.99800E 00   0.62950E 05
 2    0.10000E 01   0.58736E 00   0.10000E 01   0.96219E 00   0.17606E-02   0.00000E 00   0.33360E 05

FREQ =  0.100000E 05
   I = 1     A=  0.10189E 01    P= -0.50239E 01    P/W= -0.13955E-05    D=  0.14644E-05
   I = 2     A=  0.93267E-01    P=  0.16870E 03    P/W=  0.46861E-04    D=  0.34566E-05
            AT=  0.95034E-01   PT=  0.16367E 03   PT/W=  0.45466E-04   DT=  0.49210E-05

FREQ =  0.200000E 05
   I = 1     A=  0.10858E 01    P= -0.10837E 02    P/W= -0.15052E-05    D=  0.18154E-05
   I = 2     A=  0.47311E 00    P=  0.15102E 03    P/W=  0.20975E-04    D=  0.71048E-05
            AT=  0.51373E 00   PT=  0.14018E 03   PT/W=  0.19470E-04   DT=  0.89203E-05

FREQ =  0.300000E 05
   I = 1     A=  0.12115E 01    P= -0.18575E 02    P/W= -0.17199E-05    D=  0.25728E-05
   I = 2     A=  0.13851E 01    P=  0.10979E 03    P/W=  0.10165E-04    D=  0.16049E-04
            AT=  0.16781E 01   PT=  0.91216E 02   PT/W=  0.84460E-05   DT=  0.18621E-04

FREQ =  0.400000E 05
   I = 1     A=  0.14134E 01    P= -0.30263E 02    P/W= -0.21016E-05    D=  0.40883E-05
   I = 2     A=  0.16682E 01    P=  0.58052E 02    P/W=  0.40313E-05    D=  0.99287E-05
            AT=  0.23580E 01   PT=  0.27788E 02   PT/W=  0.19297E-05   DT=  0.14017E-04

FREQ =  0.500000E 05
   I = 1     A=  0.16677E 01    P= -0.49416E 02    P/W= -0.27453E-05    D=  0.67178E-05
   I = 2     A=  0.14164E 01    P=  0.35163E 02    P/W=  0.19535E-05    D=  0.39029E-05
            AT=  0.23623E 01   PT= -0.14253E 02   PT/W= -0.79183E-06   DT=  0.10620E-04

FREQ =  0.600000E 05
   I = 1     A=  0.17444E 01    P= -0.78028E 02    P/W= -0.36124E-05    D=  0.86806E-05
   I = 2     A=  0.12591E 01    P=  0.25241E 02    P/W=  0.11686E-05    D=  0.19393E-05
            AT=  0.21965E 01   PT= -0.52786E 02   PT/W= -0.24438E-05   DT=  0.10620E-04

FREQ =  0.700000E 05
   I = 1     A=  0.14201E 01    P= -0.10676E 03    P/W= -0.42366E-05    D=  0.67551E-05
   I = 2     A=  0.11705E 01    P=  0.19859E 02    P/W=  0.78806E-06    D=  0.11538E-05
            AT=  0.16623E 01   PT= -0.86904E 02   PT/W= -0.34485E-05   DT=  0.79089E-05

FREQ =  0.800000E 05
   I = 1     A=  0.10163E 01    P= -0.12588E 03    P/W= -0.43708E-05    D=  0.40220E-05
   I = 2     A=  0.11166E 01    P=  0.16470E 02    P/W=  0.57189E-06    D=  0.76891E-06
            AT=  0.11349E 01   PT= -0.10941E 03   PT/W= -0.37989E-05   DT=  0.47909E-05

FREQ =  0.900000E 05
   I = 1     A=  0.73069E 00    P= -0.13711E 03    P/W= -0.42319E-05    D=  0.23872E-05
   I = 2     A=  0.10815E 01    P=  0.14126E 02    P/W=  0.43599E-06    D=  0.55206E-06
            AT=  0.79027E 00   PT= -0.12298E 03   PT/W= -0.37959E-05   DT=  0.29392E-05

FREQ =  0.100000E 06
   I = 1     A=  0.54551E 00    P= -0.14396E 03    P/W= -0.39990E-05    D=  0.15067E-05
   I = 2     A=  0.10572E 01    P=  0.12398E 02    P/W=  0.34440E-06    D=  0.41747E-06
            AT=  0.57677E 00   PT= -0.13156E 03   PT/W= -0.36546E-05   DT=  0.19241E-05
```

FIGURE 6-19.—Sample output of program ML7 for a second order bandpass Butterworth filter

The program also can be used to determine the sensitivity of a network to a change in one or more of its component values. Any component can be made to deviate a specified percentage by properly choosing its temperature coefficient along with a temperature change. The response curves of the network are obtained for each of the component value deviations. By comparing the response curves for each case, the sensitivity can be determined.

### Operation

To use ML8, the input data cards are prepared in the format shown in table X.

The data on card 1 are fixed-point numbers, and all other input data are floating-point numbers. The capacitor values are more conveniently expressed in exponential floating-point form. As an example, 100 picofarads would be expressed as 1.0E–10 or 100.0E–12. When this form is used, the number must end in the last column allocated for the parameter.

The temperature coefficients of the component values are specified in units of percentage of change per °C. This value can be related to units of parts per million (ppm), which is another common means of specifying temperature coefficients. For example, 200 ppm equal 0.02 percent/°C.

The temperature is entered in units of degrees Celsius. Both the temperature and the temperature coefficients can be entered in either positive or negative values. If the temperature coefficients are negative, they should be entered as negative values.

The subroutines called by this program are TEMP and CARSP.

### Output Format

Three samples of the computer printout for this program are shown in figures 6–20 through 6–22. The input parameter values are for a second-order, bandpass Butterworth filter having a bandwidth of 40 kHz at a center frequency of 45.826 kHz; the values were obtained from the sample printout of figure 6–19. The temperatures specified for these three examples are 25° C, −40° C, and 125° C as shown on the printouts.

The input data are printed out first. The component values and the temperature coefficients of each network are printed out in the order entered. The temperature coefficients of $R_1$, $R_2$, $C_1$, and $C_2$ are listed as PR1, PR2, PC1, and PC2. The nominal reference temperature at which the temperature coefficients have no effect is 25° C.

The filter response curves are calculated from the coefficients of the network voltage transfer ratios, which have the general form of

$$\frac{E_{out}}{E_{in}}=\frac{GS^2+DS+E}{AS^2+BS+C},$$

TABLE X.—*ML8 Input Data Card Format*

| Card | Parameter | Columns |
|---|---|---|
| No. 1 | No. of L.P. 1st Order Networks | 4 and 5 |
| | No. of L.P. 2nd Order Networks | 9 and 10 |
| | No. of H.P. 1st Order Networks | 14 and 15 |
| | No. of H.P. 2nd Order Networks | 19 and 20 |
| No. 2 | Temperature (°C) | 1–10 |
| No. 3 | $R_G$ (ohms) | 1–10 |
| | $R_O$ (ohms) | 11–20 |
| | $K$ | 21–30 |
| | $C_A$ (farads) | 31–45 |
| | $F$ (Hz) | 46–60 |
| No. 4 | $R_1$ (ohms) | 1–10 |
| | $R_2$ (ohms) | 11–20 |
| | $C_1$ (farads) | 21–35 |
| | $C_2$ (farads) | 36–50 |
| No. 5 | Temperature coefficient of $R_1$ (% change/°C) | 1–10 |
| | Temperature coefficient of $R_2$ (% change/°C) | 11–20 |
| | Temperature coefficient of $C_1$ (% change/°C) | 21–30 |
| | Temperature coefficient of $C_2$ (% change/°C) | 31–40 |

Repeat cards 3 through 5 for each of the other networks. The networks should be entered in the same order as shown on card 1.

| | | |
|---|---|---|
| Response card | Maximum frequency (Hz) | 1–10 |
| | Minimum frequency (Hz) | 11–20 |
| | Frequency step (Hz) | 21–30 |
| | Amplitude scale factor | 31–40 |
| Blank card | | |

```
                    TEMPERATURE DEPENDENT FILTER RESPONSE PROGRAM    ML8

  INPUT DATA

  LP 1ST ORDER = 0      LP 2ND ORDER = 1
  HP 1ST ORDER = 0      HP 2ND ORDER = 1             TEMP = 0.25000E 02

  RG = 0.20000E 02      RO = 0.15000E 02      K = 0.99800E 00      CA = 0.10000E-10      F = 0.62950E 05
  R1 = 0.13730E 04      R2 = 0.95660E 03     C1 = 0.82000E-08      C2 = 0.56000E-09
 PR1 = 0.50000E-01     PR2 = 0.50000E-01    PC1 = 0.10000E 00     PC2 = 0.10000E 00

  RG = 0.60000E 02      RO = 0.15000E 02      K = 0.99800E 00      CA = 0.10000E-10      F = 0.33360E 05
  R1 = 0.21336E 04      R2 = 0.32560E 05     C1 = 0.56000E-09      C2 = 0.56000E-09
 PR1 = 0.50000E-01     PR2 = 0.50000E-01    PC1 = 0.10000E 00     PC2 = 0.10000E 00

  FILTER RESPONSE CURVES

   I       A             B             C             G             D             E             F

   1   0.10001E 01   0.58740E 00   0.10000E 01   0.10492E-01   0.48649E-01   0.99800E 00   0.62950E 05
   2   0.99983E 00   0.58733E 00   0.10000E 01   0.96197E 00   0.17606E-02   0.00000E 00   0.33360E 05

FREQ =  0.100000E 05
   I = 1    A=  0.10189E 01    P= -0.50243E 01    P/W= -0.13956E-05    D=  0.14645E-05
   I = 2    A=  0.93244E-01    P=  0.16870E 03    P/W=  0.46861E-04    D=  0.34563E-05
           AT=  0.95011E-01   PT=  0.16367E 03   PT/W=  0.45466E-04   DT=  0.49209E-05

FREQ =  0.200000E 05
   I = 1    A=  0.10858E 01    P= -0.10838E 02    P/W= -0.15053E-05    D=  0.18156E-05
   I = 2    A=  0.47297E 00    P=  0.15102E 03    P/W=  0.20976E-04    D=  0.71034E-05
           AT=  0.51359E 00   PT=  0.14019E 03   PT/W=  0.19470E-04   DT=  0.89191E-05

FREQ =  0.300000E 05
   I = 1    A=  0.12115E 01    P= -0.18577E 02    P/W= -0.17201E-05    D=  0.25732E-05
   I = 2    A=  0.13847E 01    P=  0.10980E 03    P/W=  0.10167E-04    D=  0.16045E-04
           AT=  0.16777E 01   PT=  0.91228E 02   PT/W=  0.84470E-05   DT=  0.18619E-04

FREQ =  0.400000E 05
   I = 1    A=  0.14135E 01    P= -0.30267E 02    P/W= -0.21019E-05    D=  0.40892E-05
   I = 2    A=  0.16682E 01    P=  0.58064E 02    P/W=  0.40322E-05    D=  0.99309E-05
           AT=  0.23580E 01   PT=  0.27797E 02   PT/W=  0.19303E-05   DT=  0.14020E-04

FREQ =  0.500000E 05
   I = 1    A=  0.16678E 01    P= -0.49425E 02    P/W= -0.27458E-05    D=  0.67193E-05
   I = 2    A=  0.14164E 01    P=  0.35170E 02    P/W=  0.19539E-05    D=  0.39039E-05
           AT=  0.23623E 01   PT= -0.14254E 02   PT/W= -0.7[illegible]191E-06   DT=  0.10623E-04

FREQ =  0.600000E 05
   I = 1    A=  0.17443E 01    P= -0.78041E 02    P/W= -0.36130E-05    D=  0.86812E-05
   I = 2    A=  0.12591E 01    P=  0.25246E 02    P/W=  0.11688E-05    D=  0.19398E-05
           AT=  0.21963E 01   PT= -0.52795E 02   PT/W= -0.24442E-05   DT=  0.10621E-04

FREQ =  0.700000E 05
   I = 1    A=  0.14199E 01    P= -0.10677E 03    P/W= -0.42371E-05    D=  0.67541E-05
   I = 2    A=  0.11704E 01    P=  0.19862E 02    P/W=  0.78819E-06    D=  0.11540E-05
           AT=  0.16620E 01   PT= -0.86912E 02   PT/W= -0.34489E-05   DT=  0.79082E-05

FREQ =  0.800000E 05
   I = 1    A=  0.10161E 01    P= -0.12588E 03    P/W= -0.43711E-05    D=  0.40211E-05
   I = 2    A=  0.11166E 01    P=  0.16473E 02    P/W=  0.57198E-06    D=  0.76905E-06
           AT=  0.11346E 01   PT= -0.10941E 03   PT/W= -0.37991E-05   DT=  0.47902E-05

FREQ =  0.900000E 05
   I = 1    A=  0.73055E 00    P= -0.13712E 03    P/W= -0.42321E-05    D=  0.23866E-05
   I = 2    A=  0.10814E 01    P=  0.14128E 02    P/W=  0.43606E-06    D=  0.55216E-06
           AT=  0.79008E 00   PT= -0.12299E 03   PT/W= -0.37960E-05   DT=  0.29388E-05

FREQ =  0.100000E 06
   I = 1    A=  0.54542E 00    P= -0.14397E 03    P/W= -0.39991E-05    D=  0.15064E-05
   I = 2    A=  0.10572E 01    P=  0.12400E 02    P/W=  0.34445E-06    D=  0.41754E-06
           AT=  0.57663E 00   PT= -0.13157E 03   PT/W= -0.36547E-05   DT=  0.19239E-05

FREQ =  0.110000E 06
   I = 1    A=  0.42232E 00    P= -0.14841E 03    P/W= -0.37478E-05    D=  0.10072E-05
   I = 2    A=  0.10397E 01    P=  0.11068E 02    P/W=  0.27951E-06    D=  0.32793E-06
           AT=  0.43912E 00   PT= -0.13734E 03   PT/W= -0.34683E-05   DT=  0.13351E-05

FREQ =  0.120000E 06
   I = 1    A=  0.33689E 00    P= -0.15145E 03    P/W= -0.35058E-05    D=  0.70293E-06
   I = 2    A=  0.10267E 01    P=  0.10007E 02    P/W=  0.23165E-06    D=  0.26506E-06
           AT=  0.34591E 00   PT= -0.14144E 03   PT/W= -0.32742E-05   DT=  0.96800E-06
```

FIGURE 6–20.—Sample output of program ML8 showing response curves of bandpass filter at 25° C

```
                    TEMPERATURE DEPENDENT FILTER RESPONSE PROGRAM    ML8

 INPUT DATA

 LP 1ST ORDER = 0      LP 2ND ORDER = 1
 HP 1ST ORDER = 0      HP 2ND ORDER = 1                  TEMP =-0.40000E 02

  RG = 0.20000E 02    RO = 0.15000E 02     K = 0.99800E 00    CA = 0.10000E-10    F = 0.62950E 05
  R1 = 0.13730E 04    R2 = 0.95660E 03    C1 = 0.82000E-08    C2 = 0.56000E-09
 PR1 = 0.50000E-01   PR2 = 0.50000E-01   PC1 = 0.10000E 00   PC2 = 0.10000E 00

  RG = 0.60000E 02    RO = 0.15000E 02     K = 0.99800E 00    CA = 0.10000E-10    F = 0.33360E 05
  R1 = 0.21336E 04    R2 = 0.32560E 05    C1 = 0.56000E-09    C2 = 0.56000E-09
 PR1 = 0.50000E-01   PR2 = 0.50000E-01   PC1 = 0.10000E 00   PC2 = 0.10000E 00

 FILTER RESPONSE CURVES

   I       A             B             C             G             D             E             F

   1   0.82053E 00   0.53246E 00   0.10000E 01   0.87990E-02   0.45487E-01   0.99800E 00   0.62950E 05
   2   0.80736E 00   0.53485E 00   0.10000E 01   0.77454E 00   0.16462E-02   0.00000E 00   0.33360E 05

FREQ =  0.100000E 05
   I = 1    A=  0.10151E 01    P= -0.45216E 01    P/W= -0.12560E-05    D=  0.13069E-05
   I = 2    A=  0.73946E-01    P=  0.16978E 03    P/W=  0.47162E-04    D=  0.29765E-05
           AT=  0.75065E-01   PT=  0.16526E 03   PT/W=  0.45906E-04   DT=  0.42834E-05

FREQ =  0.200000E 05
   I = 1    A=  0.10692E 01    P= -0.96202E 01    P/W= -0.13361E-05    D=  0.15603E-05
   I = 2    A=  0.35742E 00    P=  0.15548E 03    P/W=  0.21595E-04    D=  0.53985E-05
           AT=  0.38217E 00   PT=  0.14586E 03   PT/W=  0.20259E-04   DT=  0.69589E-05

FREQ =  0.300000E 05
   I = 1    A=  0.11688E 01    P= -0.16074E 02    P/W= -0.14883E-05    D=  0.20827E-05
   I = 2    A=  0.10560E 01    P=  0.12567E 03    P/W=  0.11636E-04    D=  0.11976E-04
           AT=  0.12344E 01   PT=  0.10960E 03   PT/W=  0.10148E-04   DT=  0.14059E-04

FREQ =  0.400000E 05
   I = 1    A=  0.13275E 01    P= -0.25173E 02    P/W= -0.17481E-05    D=  0.30747E-05
   I = 2    A=  0.16842E 01    P=  0.75826E 02    P/W=  0.52657E-05    D=  0.12606E-04
           AT=  0.22359E 01   PT=  0.50653E 02   PT/W=  0.35176E-05   DT=  0.15681E-04

FREQ =  0.500000E 05
   I = 1    A=  0.15481E 01    P= -0.39160E 02    P/W= -0.21755E-05    D=  0.48477E-05
   I = 2    A=  0.15232E 01    P=  0.44492E 02    P/W=  0.24717E-05    D=  0.54984E-05
           AT=  0.23582E 01   PT=  0.53318E 01   PT/W=  0.29621E-06   DT=  0.10346E-04

FREQ =  0.600000E 05
   I = 1    A=  0.17453E 01    P= -0.60853E 02    P/W= -0.28173E-05    D=  0.71709E-05
   I = 2    A=  0.13348E 01    P=  0.30763E 02    P/W=  0.14242E-05    D=  0.26128E-05
           AT=  0.23298E 01   PT= -0.30089E 02   PT/W= -0.13930E-05   DT=  0.97838E-05

FREQ =  0.700000E 05
   I = 1    A=  0.16688E 01    P= -0.88480E 02    P/W= -0.35111E-05    D=  0.76125E-05
   I = 2    A=  0.12221E 01    P=  0.23657E 02    P/W=  0.93878E-06    D=  0.14903E-05
           AT=  0.20395E 01   PT= -0.64822E 02   PT/W= -0.25723E-05   DT=  0.91029E-05

FREQ =  0.800000E 05
   I = 1    A=  0.13126E 01    P= -0.11230E 03    P/W= -0.38994E-05    D=  0.54335E-05
   I = 2    A=  0.11532E 01    P=  0.19345E 02    P/W=  0.67171E-06    D=  0.96355E-06
           AT=  0.15138E 01   PT= -0.92960E 02   PT/W= -0.32277E-05   DT=  0.63971E-05

FREQ =  0.900000E 05
   I = 1    A=  0.96395E 00    P= -0.12785E 03    P/W= -0.39462E-05    D=  0.33505E-05
   I = 2    A=  0.11085E 01    P=  0.16438E 02    P/W=  0.50737E-06    D=  0.67710E-06
           AT=  0.10686E 01   PT= -0.11142E 03   PT/W= -0.34388E-05   DT=  0.40276E-05

FREQ =  0.100000E 06
   I = 1    A=  0.71711E 00    P= -0.13745E 03    P/W= -0.38181E-05    D=  0.20978E-05
   I = 2    A=  0.10778E 01    P=  0.14336E 02    P/W=  0.39823E-06    D=  0.50409E-06
           AT=  0.77296E 00   PT= -0.12311E 03   PT/W= -0.34199E-05   DT=  0.26019E-05

FREQ =  0.110000E 06
   I = 1    A=  0.55056E 00    P= -0.14360E 03    P/W= -0.36263E-05    D=  0.13825E-05
   I = 2    A=  0.10558E 01    P=  0.12738E 02    P/W=  0.32167E-06    D=  0.39131E-06
           AT=  0.58133E 00   PT= -0.13086E 03   PT/W= -0.33046E-05   DT=  0.17738E-05

FREQ =  0.120000E 06
   I = 1    A=  0.43561E 00    P= -0.14774E 03    P/W= -0.34201E-05    D=  0.95533E-06
   I = 2    A=  0.10395E 01    P=  0.11477E 02    P/W=  0.26568E-06    D=  0.31346E-06
           AT=  0.45284E 00   PT= -0.13627E 03   PT/W= -0.31544E-05   DT=  0.12688E-05
```

FIGURE 6–21.—Sample output of program ML8 showing response curves of bandpass filter at −40° C

```
                    TEMPERATURE DEPENDENT FILTER RESPONSE PROGRAM     ML8

  INPUT DATA

  LP 1ST ORDER = 0      LP 2ND ORDER = 1
  HP 1ST ORDER = 0      HP 2ND ORDER = 1             TEMP = 0.12500E 03

   RG = 0.20000E 02      RO = 0.15000E 02      K = 0.99800E 00      CA = 0.10000E-10      F = 0.62950E 05
   R1 = 0.13730E 04      R2 = 0.95660E 03     C1 = 0.82000E-08      C2 = 0.56000E-09
  PR1 = 0.50000E-01     PR2 = 0.50000E-01    PC1 = 0.10000E 00     PC2 = 0.10000E 00

   RG = 0.60000E 02      RO = 0.15000E 02      K = 0.99800E 00      CA = 0.10000E-10      F = 0.33360E 05
   R1 = 0.21336E 04      R2 = 0.32560E 05     C1 = 0.56000E-09      C2 = 0.56000E-09
  PR1 = 0.50000E-01     PR2 = 0.50000E-01    PC1 = 0.10000E 00     PC2 = 0.10000E 00

  FILTER RESPONSE CURVES

  I        A             B             C             G             D             E             F

  1   0.13295E 01   0.67191E 00   0.10000E 01   0.13096E-01   0.53514E-01   0.99800E 00   0.62950E 05
  2   0.12959E 01   0.66808E 00   0.10000E 01   0.12503E 01   0.19367E-02   0.00000E 00   0.33360E 05

FREQ =  0.100000E 05
   I = 1    A=  0.10261E 01    P= -0.58142E 01    P/W= -0.16150E-05    D=  0.17214E-05
   I = 2    A=  0.12401E 00    P=  0.16693E 03    P/W=  0.46370E-04    D=  0.42533E-05
           AT=  0.12724E 00   PT=  0.16111E 03   PT/W=  0.44755E-04   DT=  0.59747E-05

FREQ =  0.200000E 05
   I = 1    A=  0.11178E 01    P= -0.12873E 02    P/W= -0.17880E-05    D=  0.22870E-05
   I = 2    A=  0.67307E 00    P=  0.14299E 03    P/W=  0.19859E-04    D=  0.10459E-04
           AT=  0.75239E 00   PT=  0.13011E 03   PT/W=  0.18071E-04   DT=  0.12746E-04

FREQ =  0.300000E 05
   I = 1    A=  0.12960E 01    P= -0.23174E 02    P/W= -0.21457E-05    D=  0.36133E-05
   I = 2    A=  0.16776E 01    P=  0.85330E 02    P/W=  0.79009E-05    D=  0.17960E-04
           AT=  0.21743E 01   PT=  0.62155E 02   PT/W=  0.57551E-05   DT=  0.21573E-04

FREQ =  0.400000E 05
   I = 1    A=  0.15768E 01    P= -0.40707E 02    P/W= -0.28269E-05    D=  0.64415E-05
   I = 2    A=  0.15264E 01    P=  0.42788E 02    P/W=  0.29714E-05    D=  0.65755E-05
           AT=  0.24069E 01   PT=  0.20808E 01   PT/W=  0.14450E-06   DT=  0.13017E-04

FREQ =  0.500000E 05
   I = 1    A=  0.17769E 01    P= -0.70732E 02    P/W= -0.39295E-05    D=  0.99112E-05
   I = 2    A=  0.13017E 01    P=  0.27591E 02    P/W=  0.15328E-05    D=  0.26745E-05
           AT=  0.23131E 01   PT= -0.43140E 02   PT/W= -0.23966E-05   DT=  0.12585E-04

FREQ =  0.600000E 05
   I = 1    A=  0.14665E 01    P= -0.10501E 03    P/W= -0.48620E-05    D=  0.81330E-05
   I = 2    A=  0.11858E 01    P=  0.20578E 02    P/W=  0.95269E-06    D=  0.14202E-05
           AT=  0.17389E 01   PT= -0.84441E 02   PT/W= -0.39093E-05   DT=  0.95533E-05

FREQ =  0.700000E 05
   I = 1    A=  0.99715E 00    P= -0.12729E 03    P/W= -0.50512E-05    D=  0.44743E-05
   I = 2    A=  0.11211E 01    P=  0.16545E 02    P/W=  0.65658E-06    D=  0.88479E-06
           AT=  0.11179E 01   PT= -0.11074E 03   PT/W= -0.43946E-05   DT=  0.53591E-05

FREQ =  0.800000E 05
   I = 1    A=  0.68467E 00    P= -0.13935E 03    P/W= -0.48388E-05    D=  0.24701E-05
   I = 2    A=  0.10814E 01    P=  0.13906E 02    P/W=  0.48287E-06    D=  0.60819E-06
           AT=  0.74046E 00   PT= -0.12545E 03   PT/W= -0.43559E-05   DT=  0.30783E-05

FREQ =  0.900000E 05
   I = 1    A=  0.49502E 00    P= -0.14627E 03    P/W= -0.45147E-05    D=  0.14845E-05
   I = 2    A=  0.10553E 01    P=  0.12032E 02    P/W=  0.37137E-06    D=  0.44619E-06
           AT=  0.52244E 00   PT= -0.13424E 03   PT/W= -0.41433E-05   DT=  0.19306E-05

FREQ =  0.100000E 06
   I = 1    A=  0.37462E 00    P= -0.15058E 03    P/W= -0.41828E-05    D=  0.95790E-06
   I = 2    A=  0.10372E 01    P=  0.10625E 02    P/W=  0.29514E-06    D=  0.34270E-06
           AT=  0.38858E 00   PT= -0.13995E 03   PT/W= -0.38877E-05   DT=  0.13006E-05

FREQ =  0.110000E 06
   I = 1    A=  0.29371E 00    P= -0.15343E 03    P/W= -0.38745E-05    D=  0.64871E-06
   I = 2    A=  0.10241E 01    P=  0.95257E 01    P/W=  0.24054E-06    D=  0.27228E-06
           AT=  0.30078E 00   PT= -0.14390E 03   PT/W= -0.36339E-05   DT=  0.92099E-06

FREQ =  0.120000E 06
   I = 1    A=  0.23661E 00    P= -0.15538E 03    P/W= -0.35969E-05    D=  0.45216E-06
   I = 2    A=  0.10142E 01    P=  0.86467E 01    P/W=  0.20001E-06    D=  0.22202E-06
           AT=  0.23998E 00   PT= -0.14674E 03   PT/W= -0.33969E-05   DT=  0.67419E-06
```

FIGURE 6-22.—Sample output of program ML8 showing response curves of bandpass filter at 125° C where $S=j(\omega/\omega_n)$, and where $\omega_n$ is the natural frequency in radians per second. Coefficients $A$, $B$, $C$, $G$, $D$, and $E$ are printed out for each network along with its natural frequency $F$ in hertz.

The response curves are calculated for each network from the lowest frequency of interest through the highest frequency of interest. The symbol representation used is

$A$=Amplitude,
$P$=Phase angle in degrees,
$P/W$=Time delay in seconds,
$D$=Group delay in seconds.

$AT$, $PT$, $PT/W$, and $DT$ are the total reponses of the combined networks.

The change in the network coefficients for the changes in temperature can be observed by comparing the coefficients shown in the three sample printouts. The temperature coefficients for the examples were chosen at 0.05 percent/° C for the resistors and 0.1 percent/° C for the capacitors. These values are high but were chosen for demonstration purposes.

For the 25° C curve, the amplitude values at the cutoff frequencies of 30 kHz and 70 kHz are equal, as they should be. For the −40° C and 125° C curves, the amplitude values are seen to be quite different.

SECTION 7

# Selection Charts

Figures 7–1 through 7–28 give the filter selection charts and group delay calculation charts. The following chart listing is given here for reference.

| *Figure* | *Attenuation/delay* |
|---|---|
| 7–1 | 4 db |
| 7–2 | 6 db |
| 7–3 | 8 db |
| 7–4 | 10 db |
| 7–5 | 12 db |
| 7–6 | 14 db |
| 7–7 | 16 db |
| 7–8 | 18 db |
| 7–9 | 20 db |
| 7–10 | 22 db |
| 7–11 | 24 db |
| 7–12 | 26 db |
| 7–13 | 28 db |
| 7–14 | 30 db |
| 7–15 | 32 db |
| 7–16 | 34 db |
| 7–17 | 36 db |
| 7–18 | 38 db |
| 7–19 | 40 db |
| 7–20 | 45 db |
| 7–21 | 50 db |
| 7–22 | 55 db |
| 7–23 | 60 db |
| 7–24 | 70 db |
| 7–25 | 80 db |
| 7–26 | Delay at 0 |
| 7–27 | Delay at 1 |
| 7–28 | 0-db ripple equivalent frequencies |



353–813 O—69——4

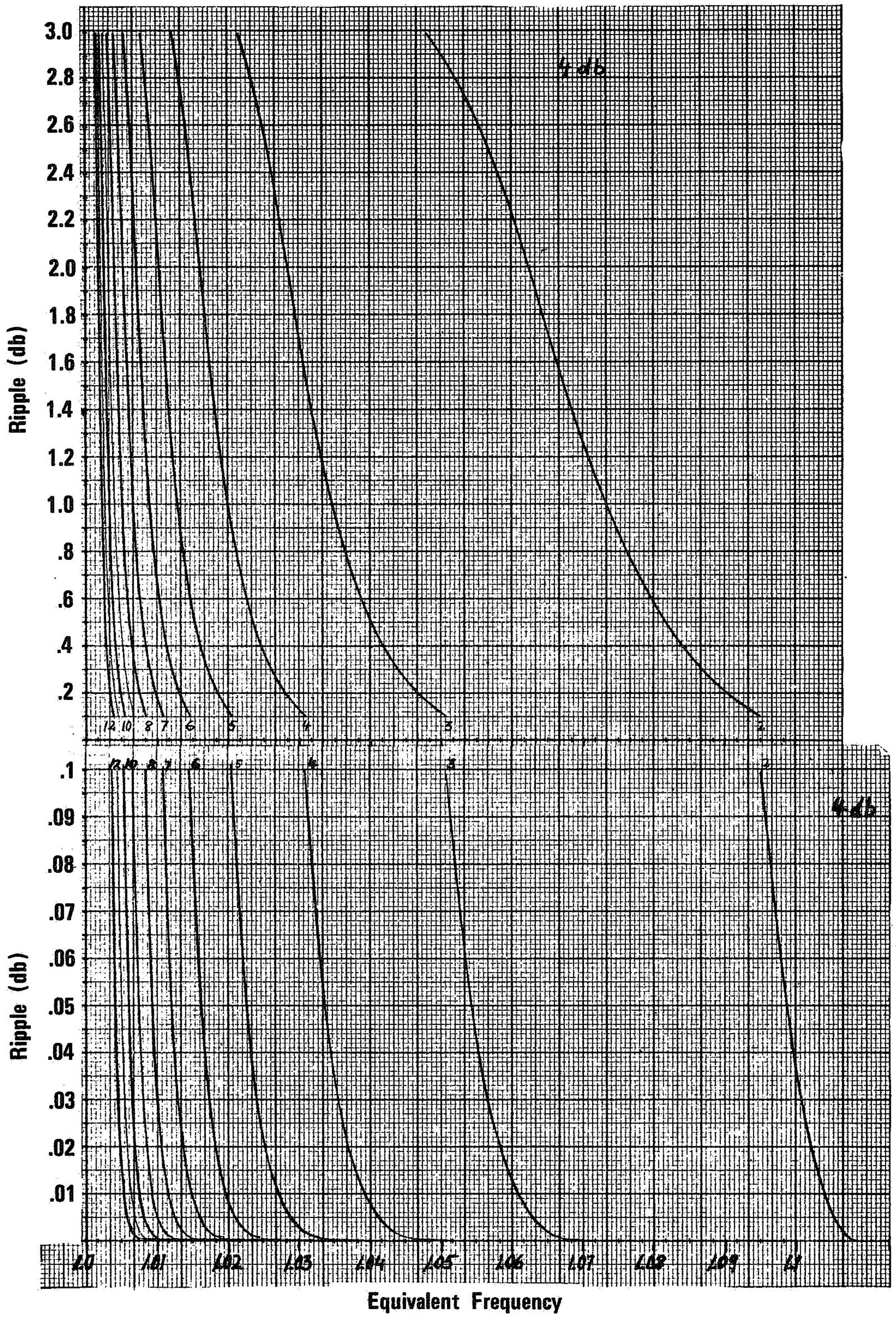



FIGURE 7-1.—Filter selection chart, 4 db attenuation

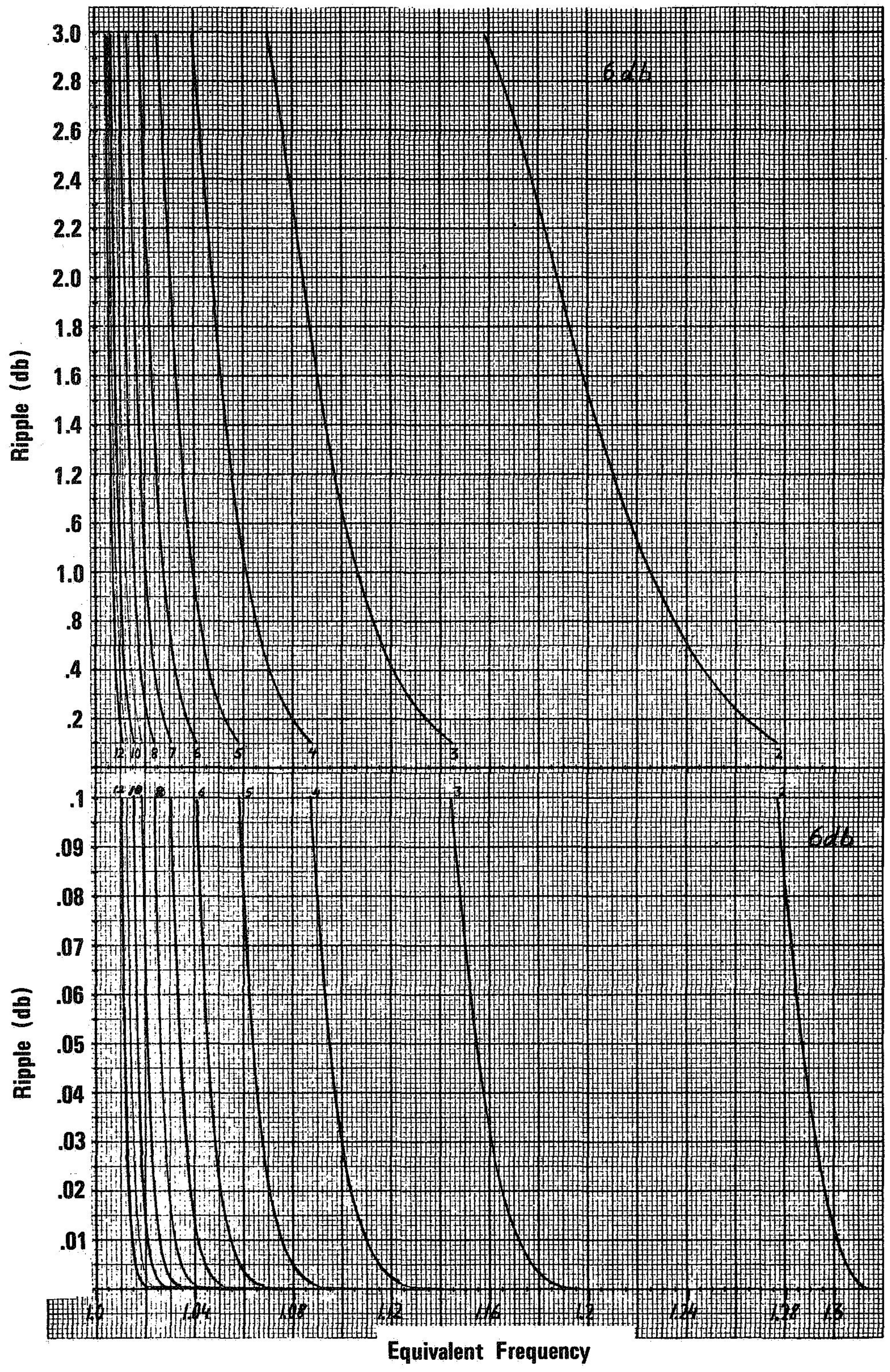



FIGURE 7–2.—Filter selection chart, 6 db attenuation

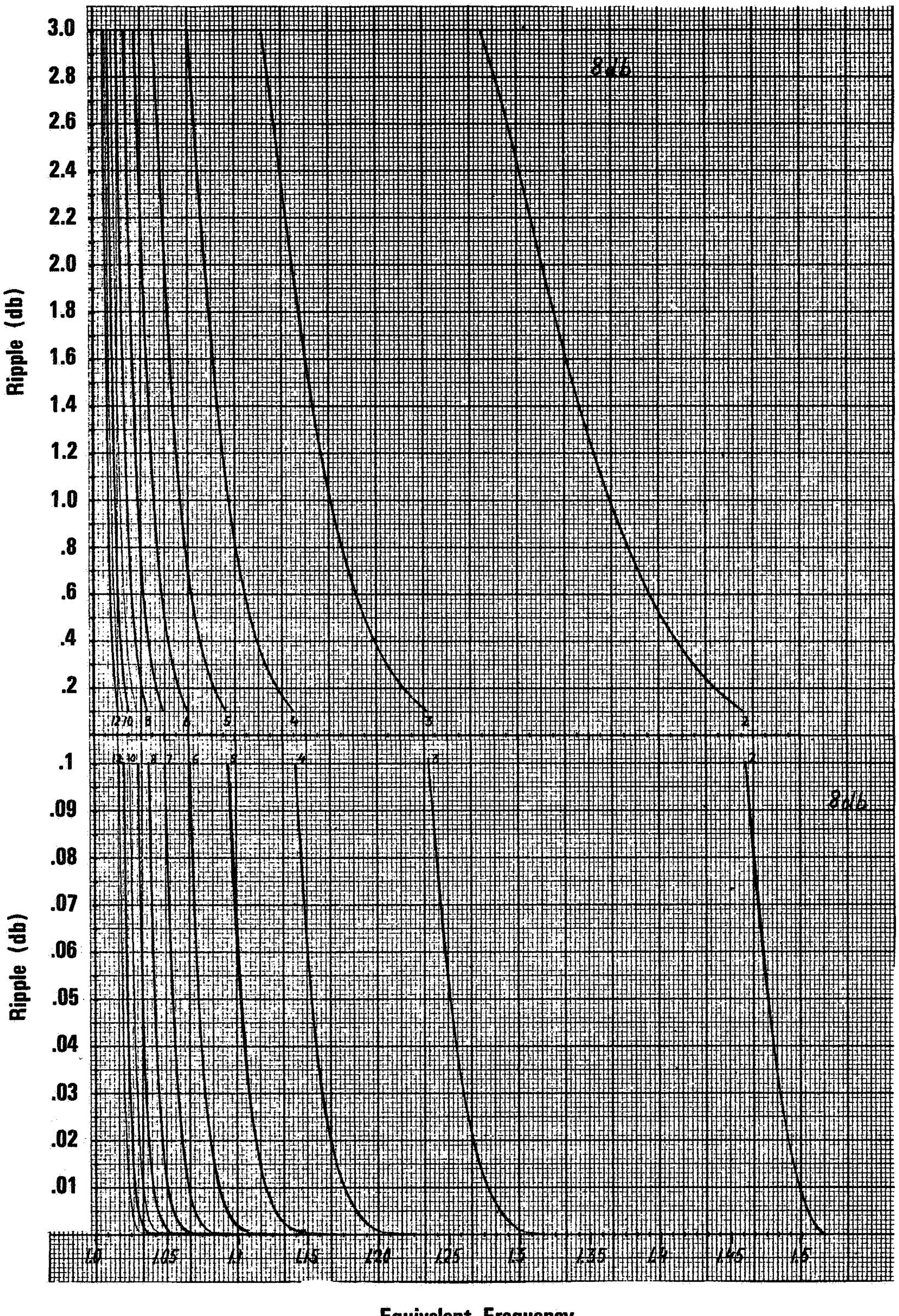



FIGURE 7–3.—Filter selection chart, 8 db attenuation

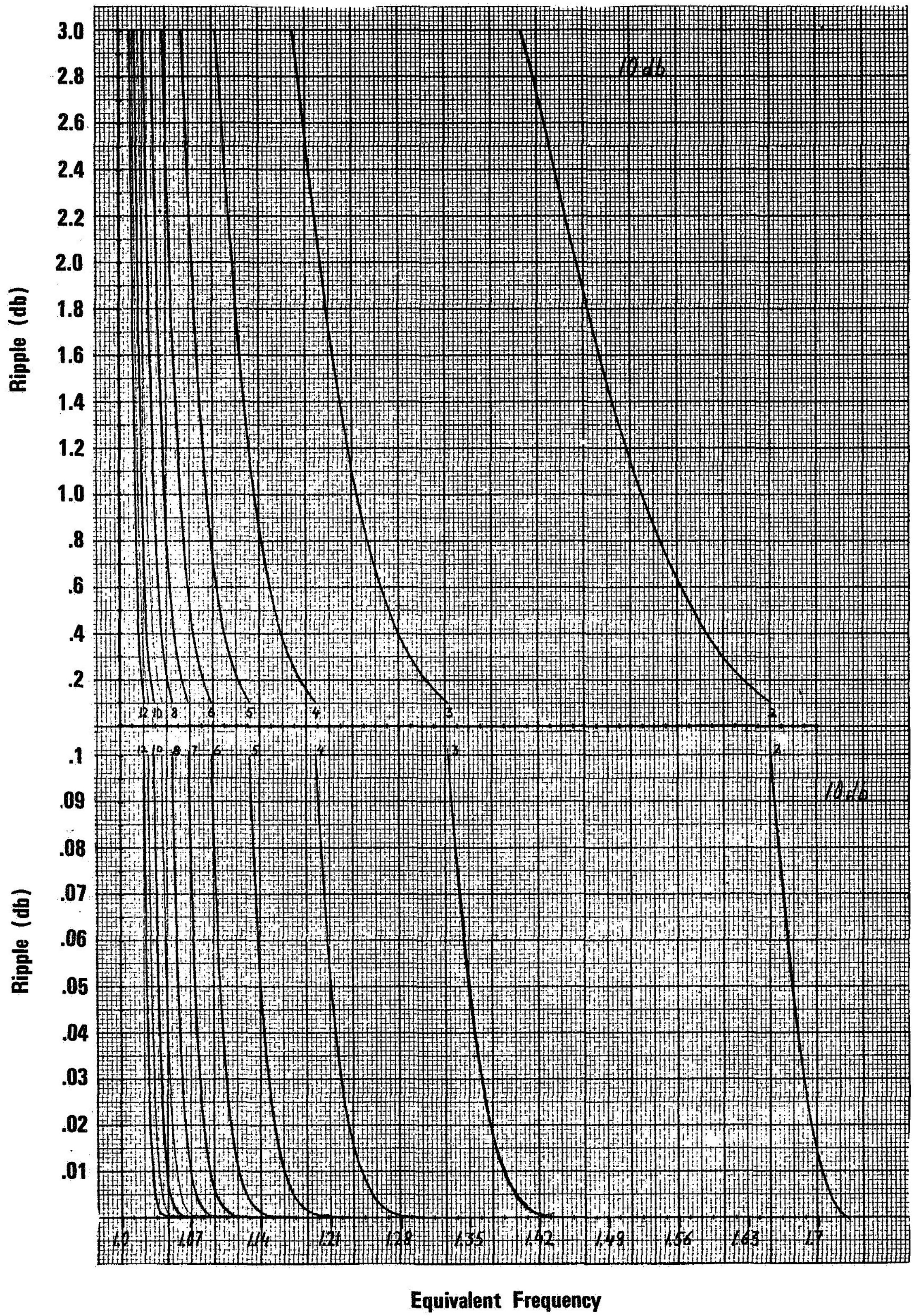



FIGURE 7–4.—Filter selection chart, 10 db attenuation

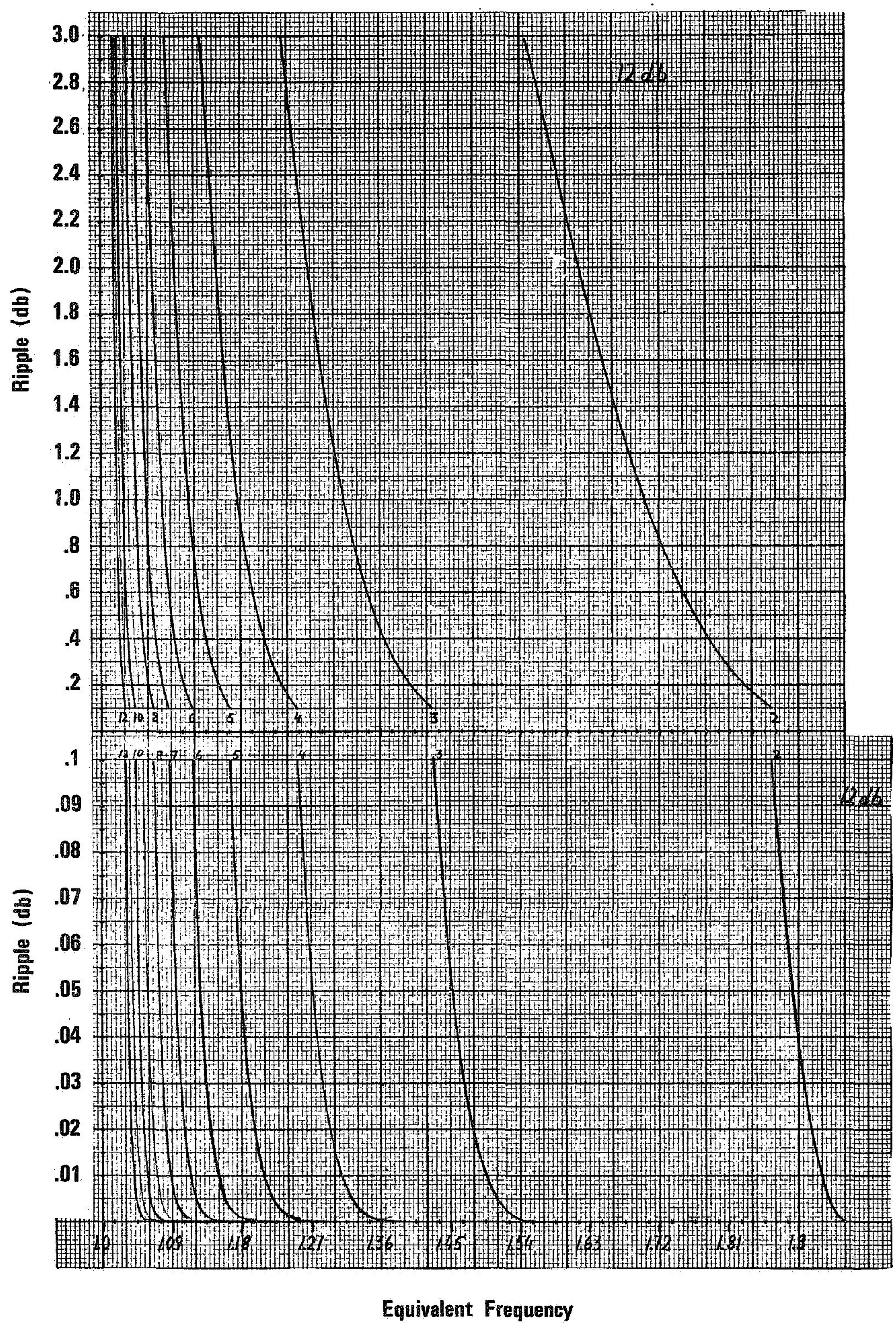



FIGURE 7–5.—Filter selection chart, 12 db attenuation

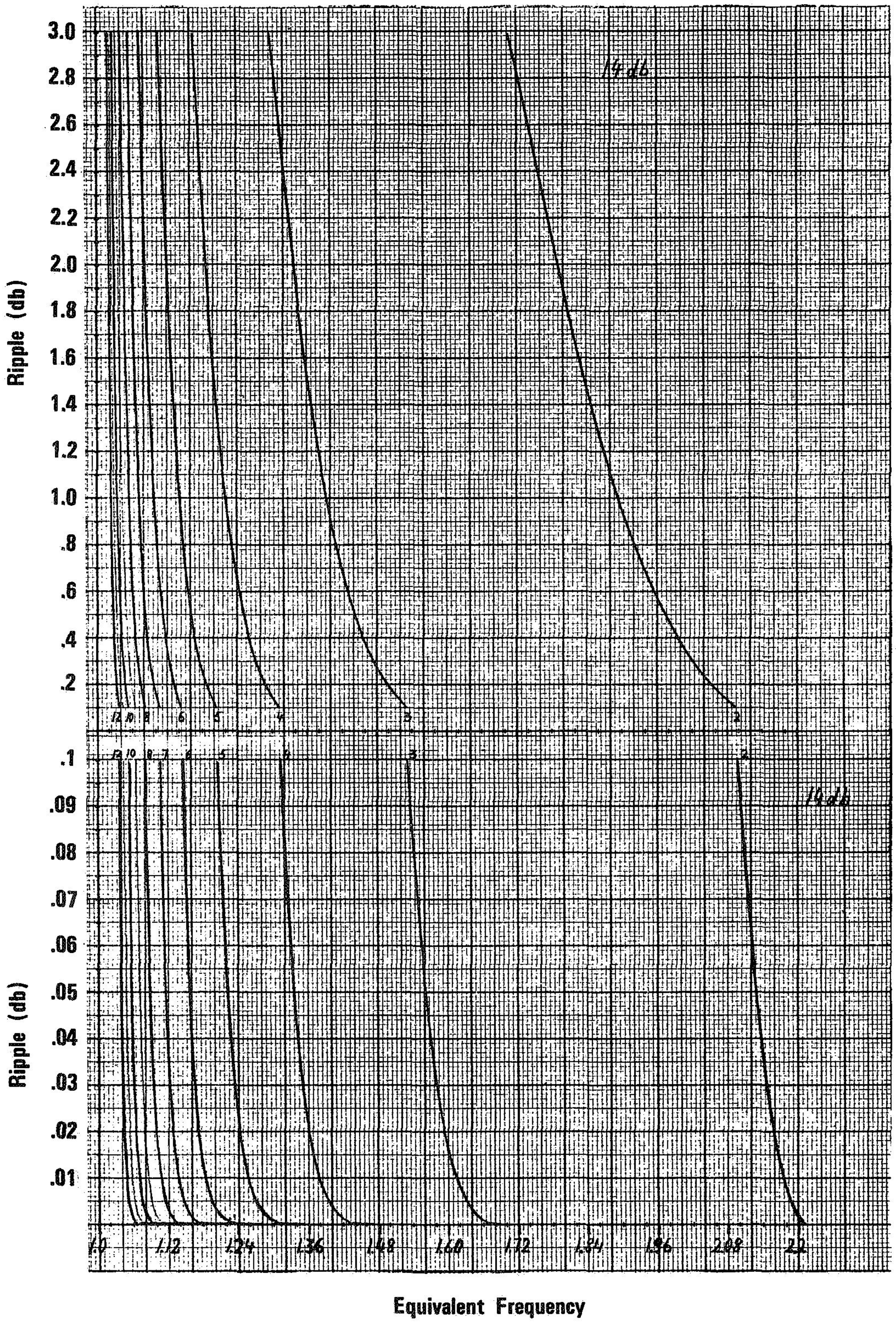



Figure 7–6.—Filter selection chart, 14 db attenuation

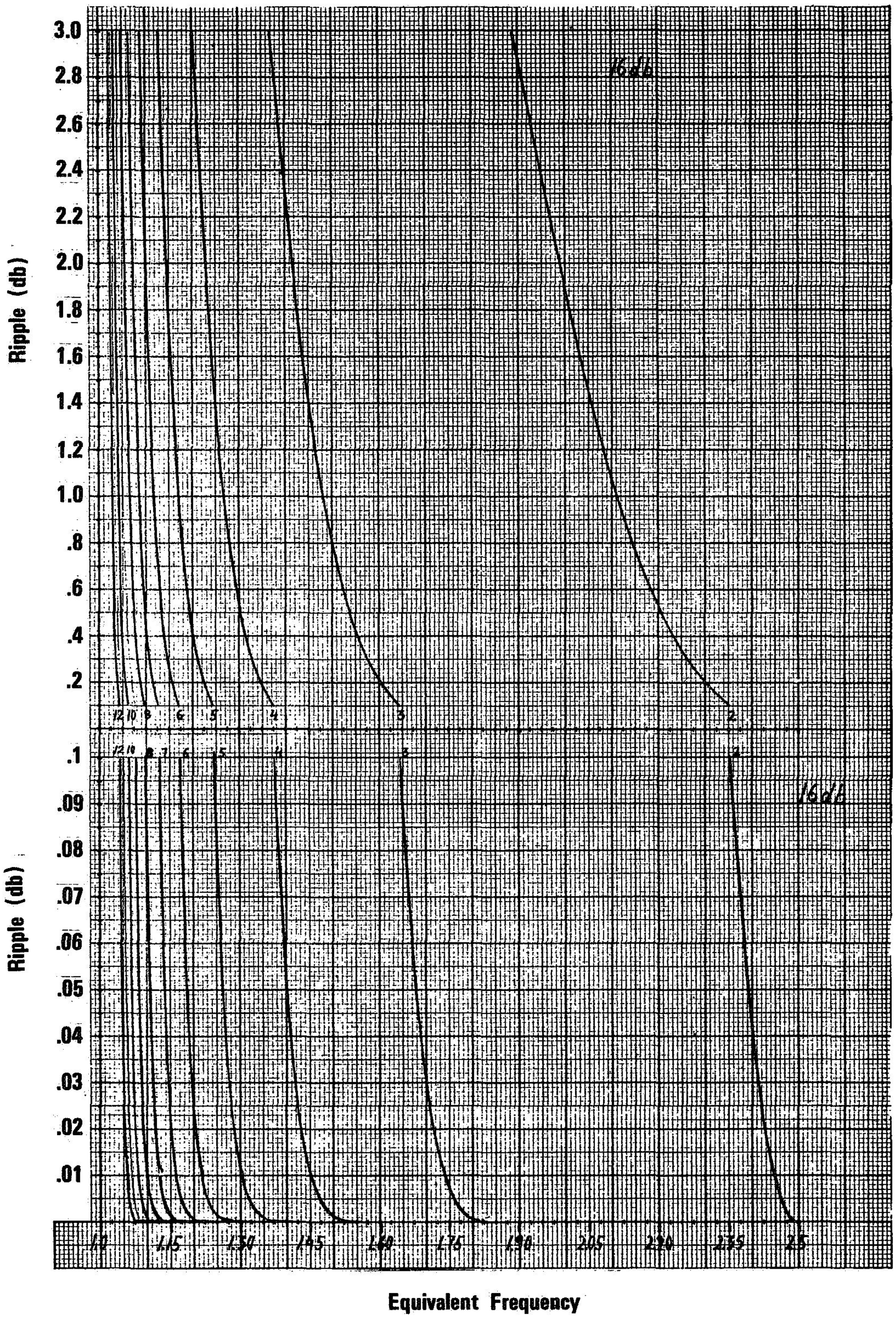



FIGURE 7-7.—Filter selection chart, 16 db attenuation

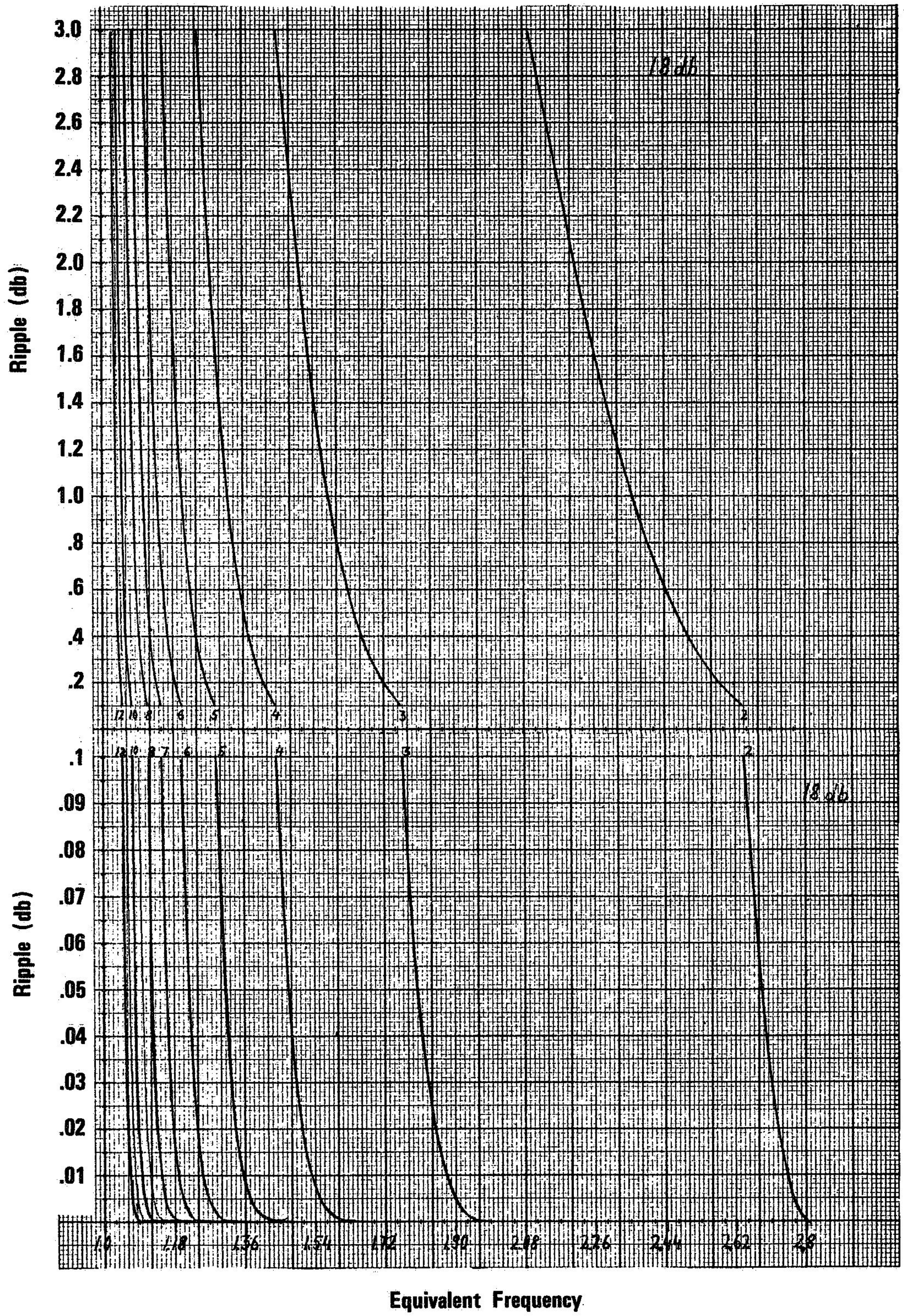



FIGURE 7–8.—Filter selection chart, 18 db attenuation

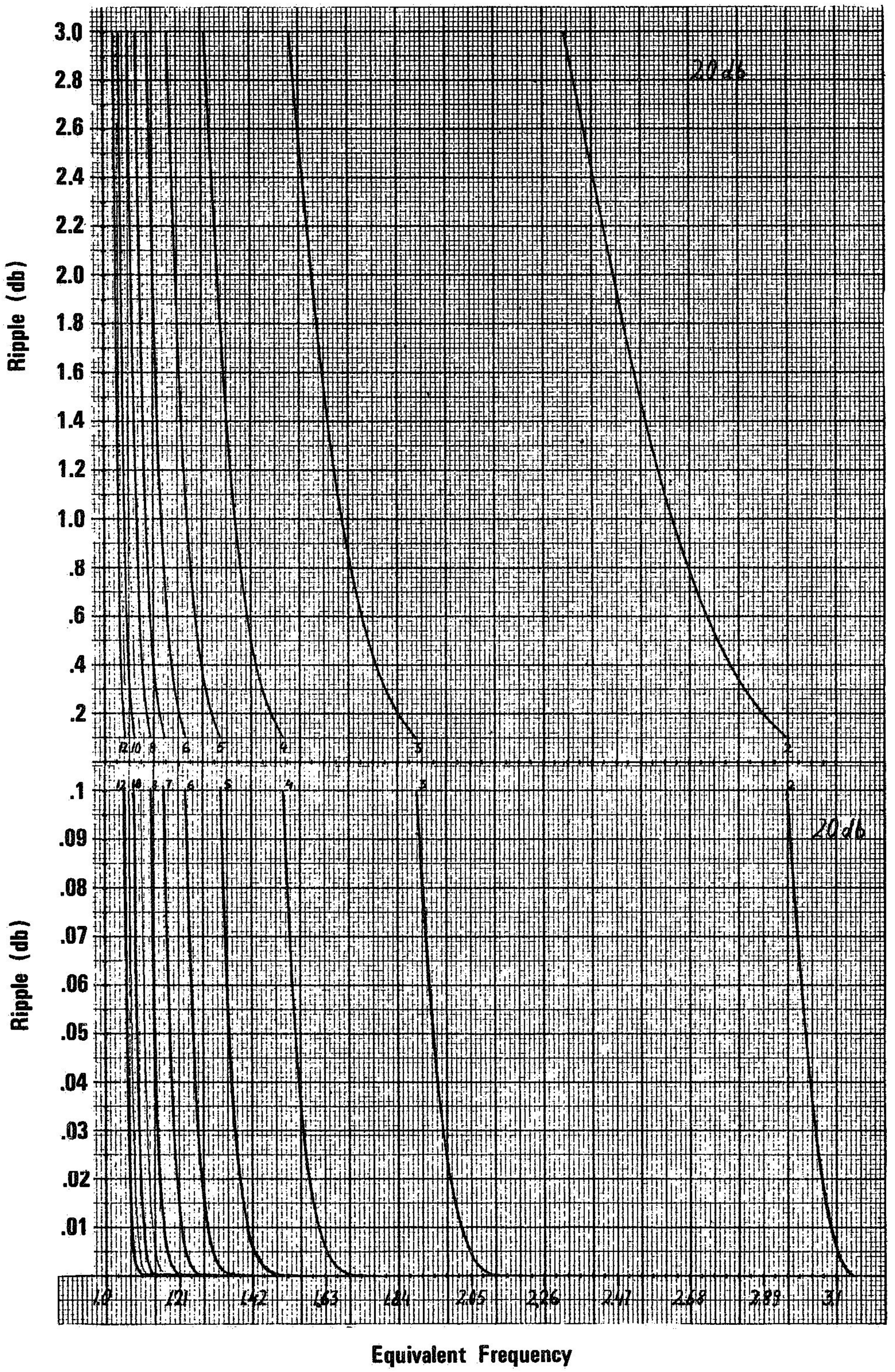



FIGURE 7–9.—Filter selection chart, 20 db attenuation

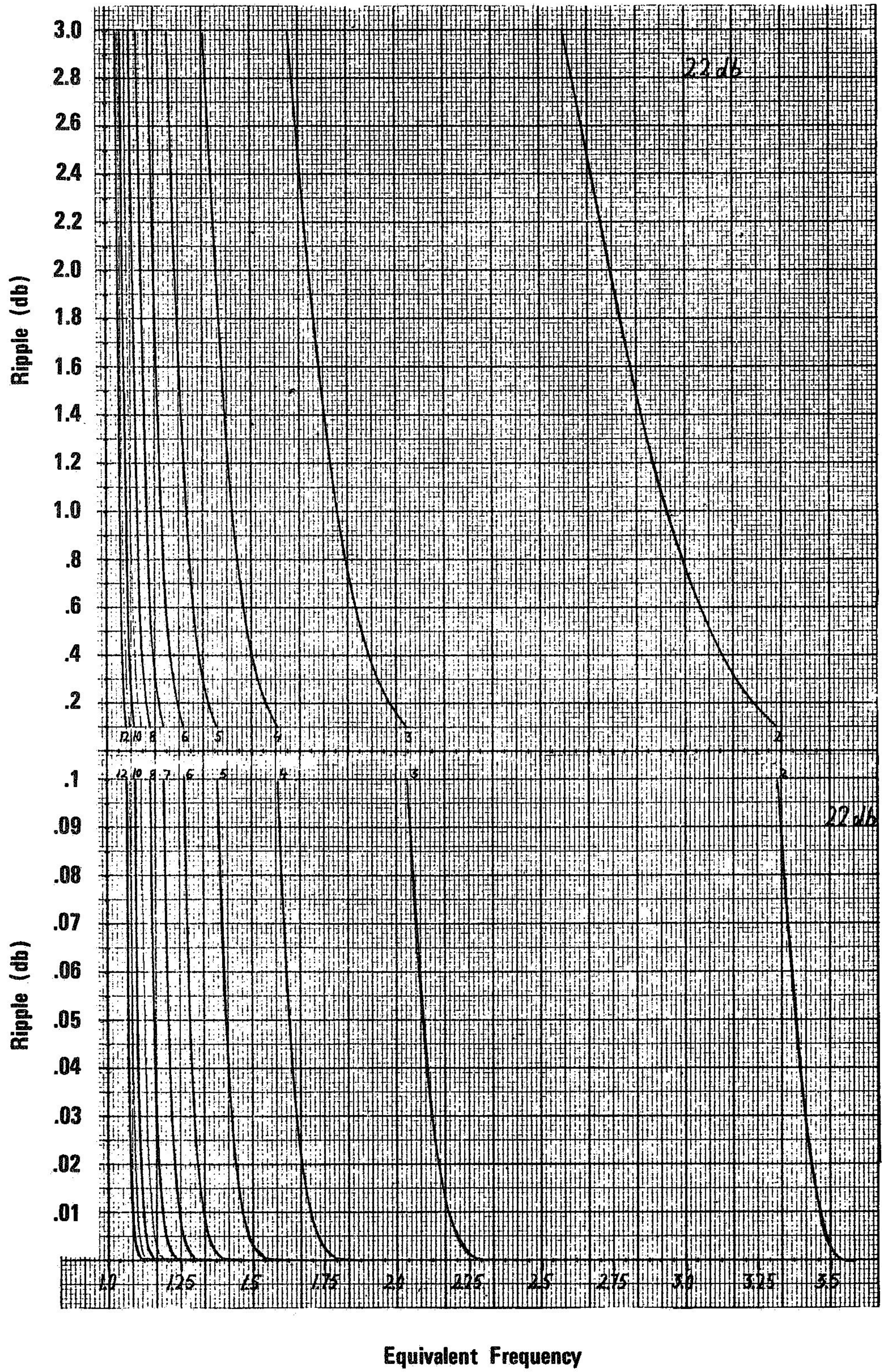



FIGURE 7–10.—Filter selection chart, 22 db attenuation

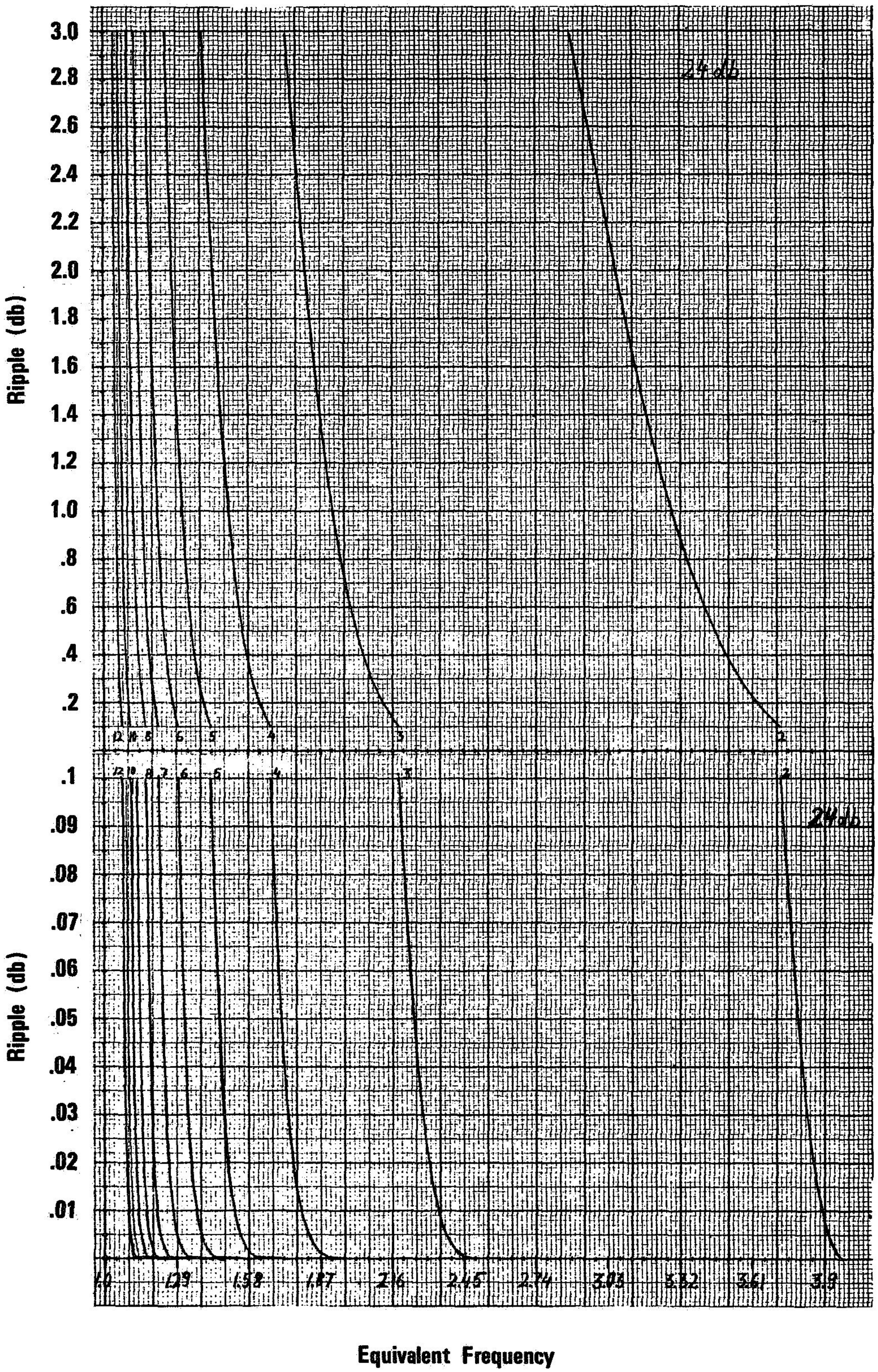



FIGURE 7–11.—Filter selection chart, 24 db attenuation

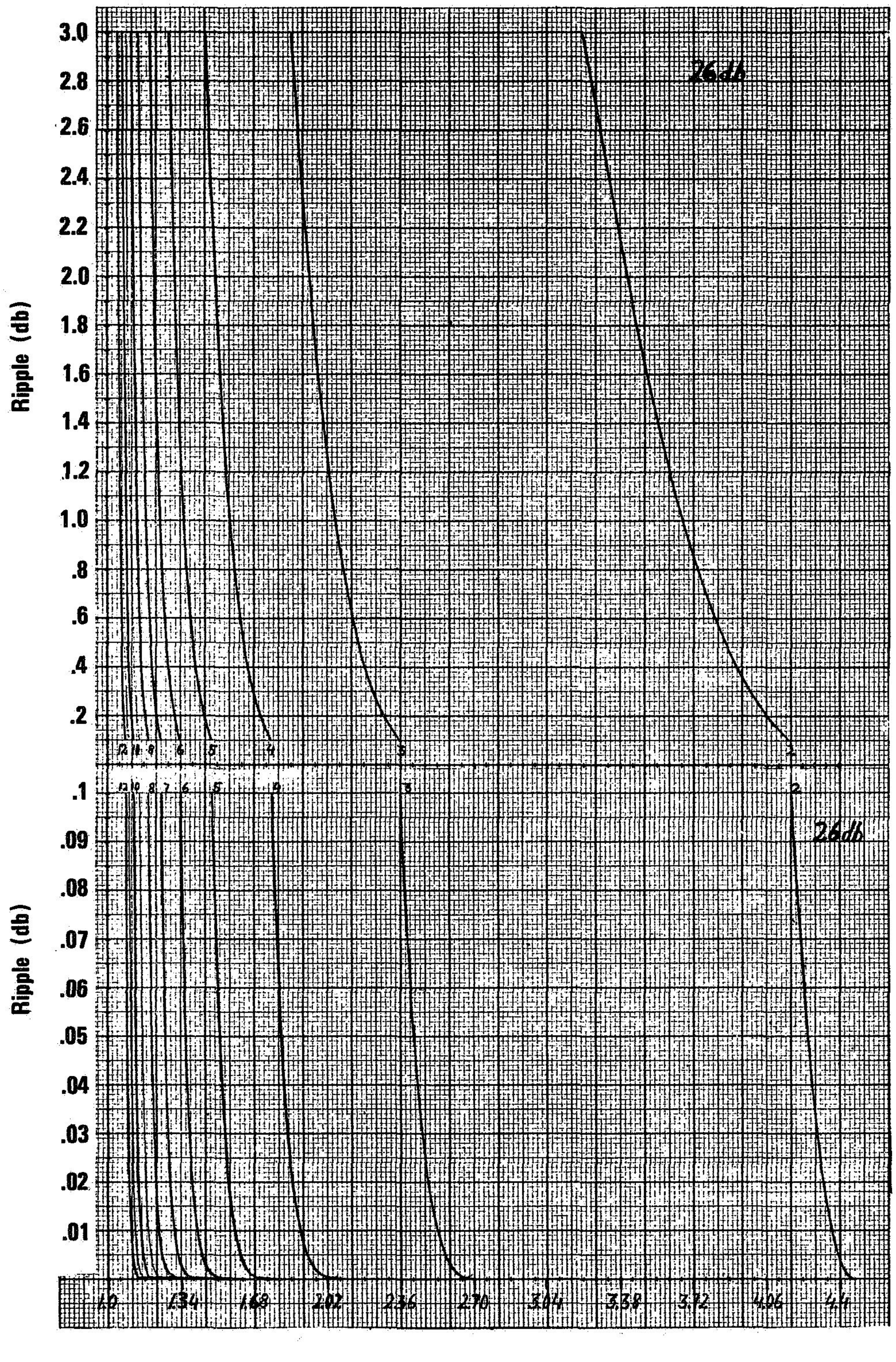



FIGURE 7-12.—Filter selection chart, 26 db attenuation

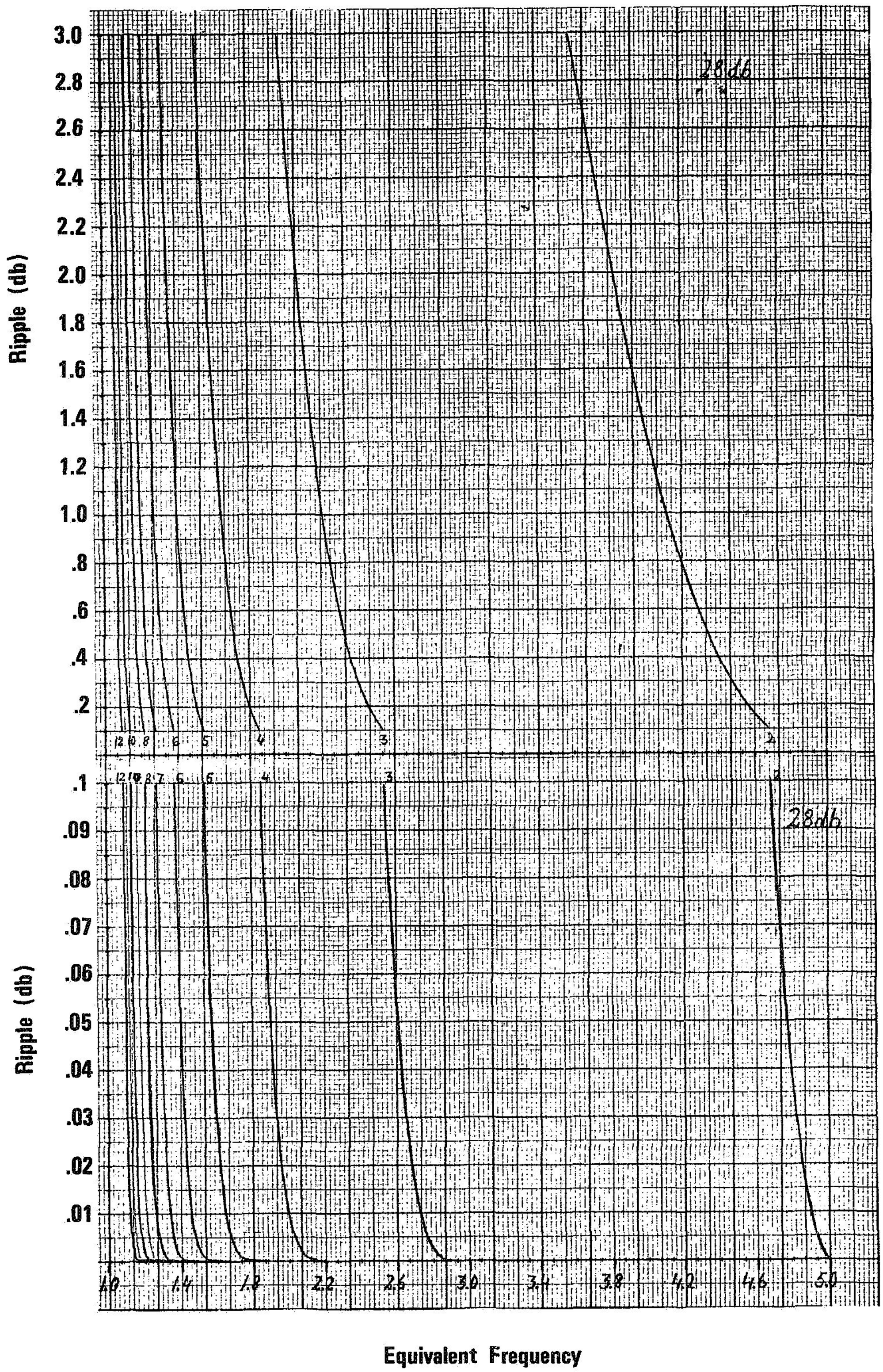



FIGURE 7-13.—Filter selection chart, 28 db attenuation

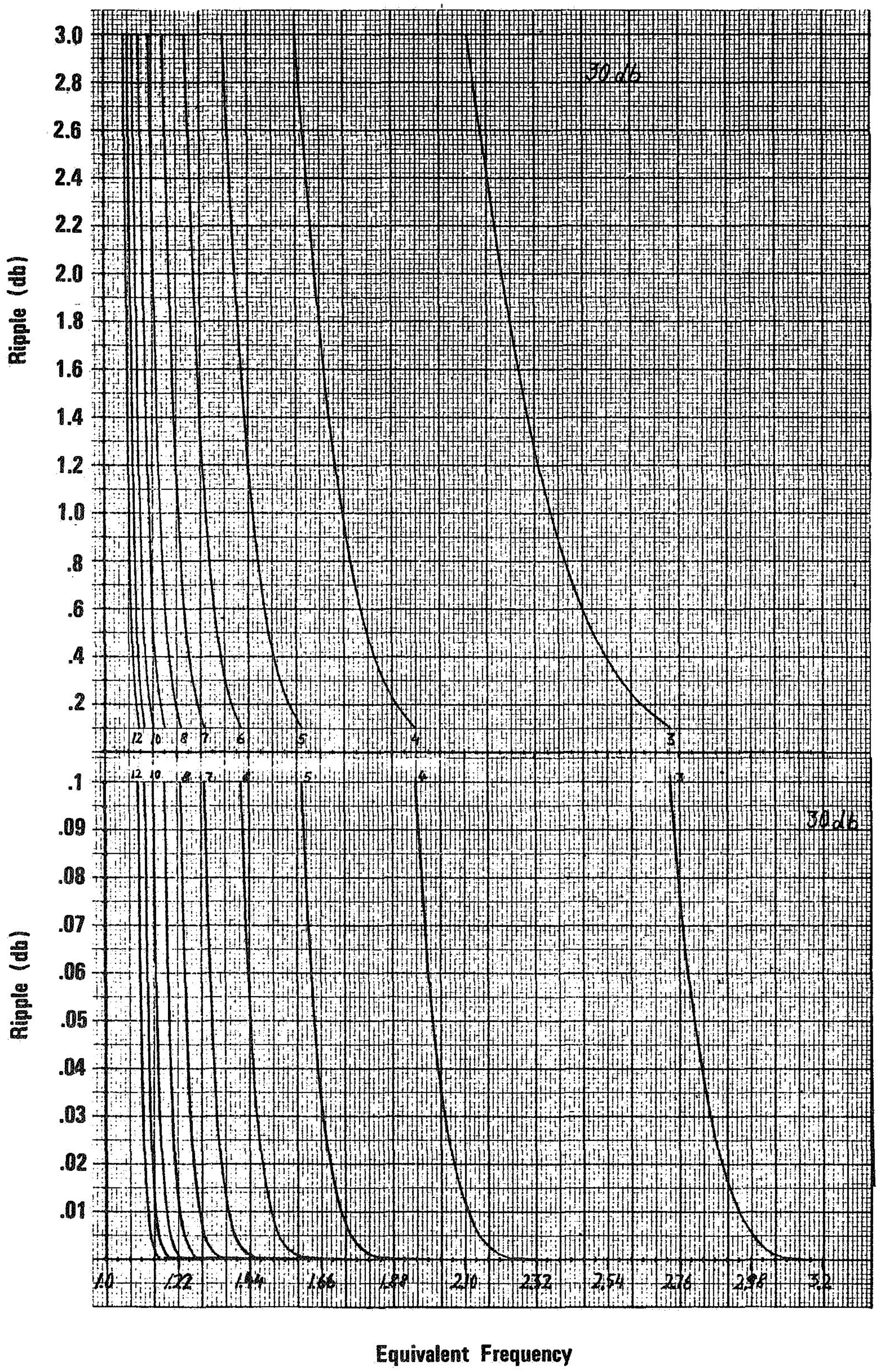



FIGURE 7-14.—Filter selection chart, 30 db attenuation

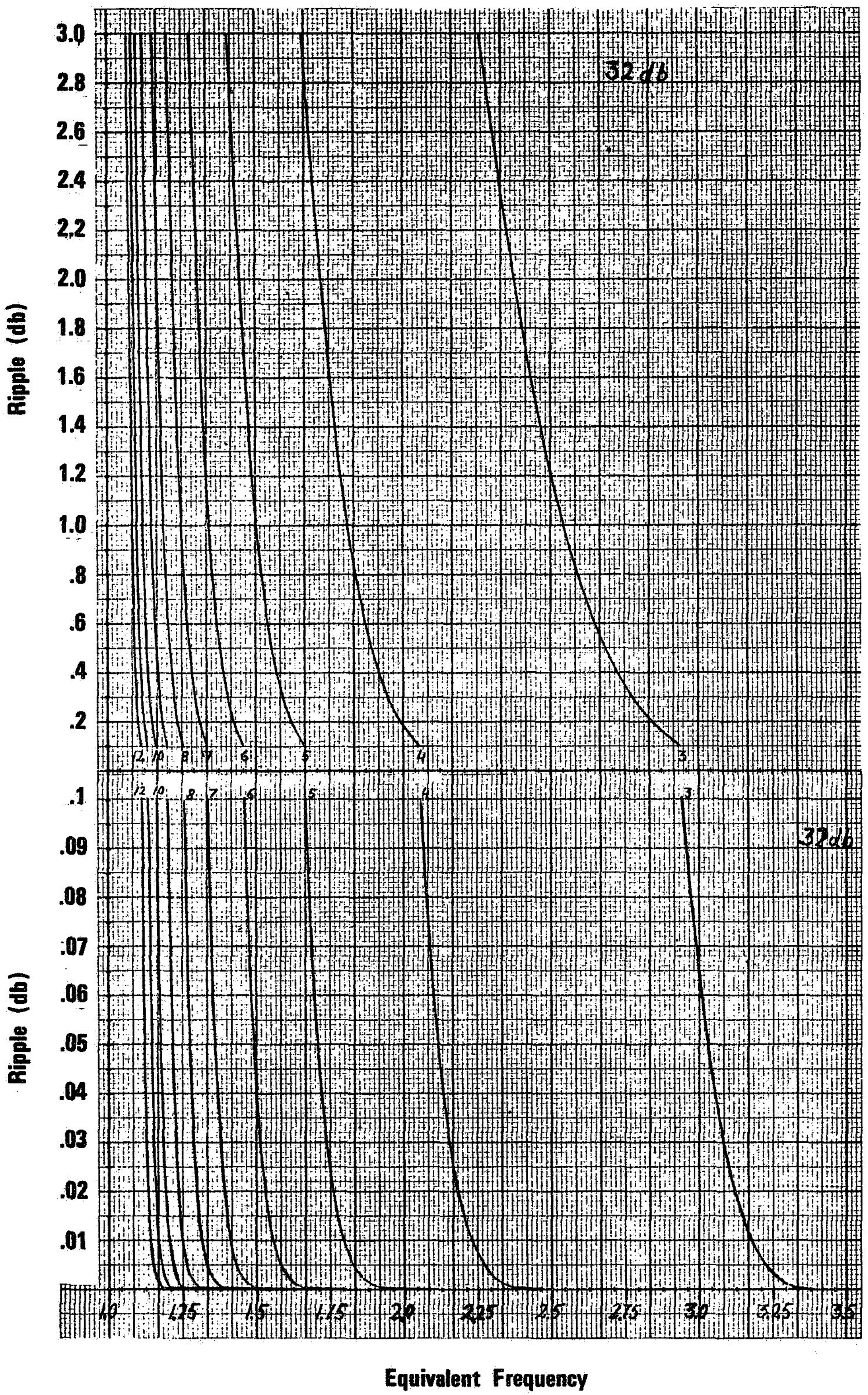



FIGURE 7–15.—Filter selection chart, 32 db attenuation

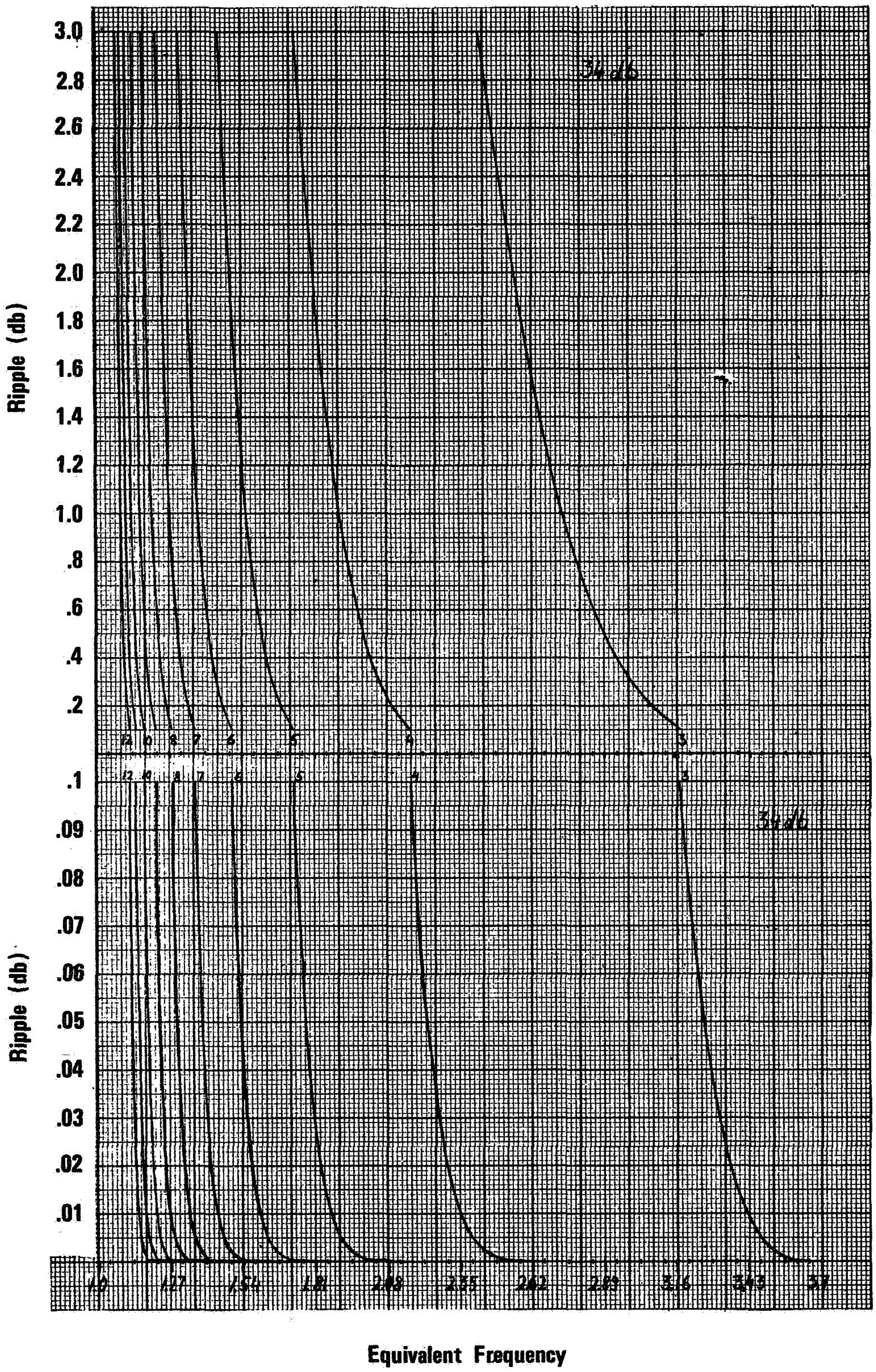



FIGURE 7–16.—Filter selection chart, 34 db attenuation

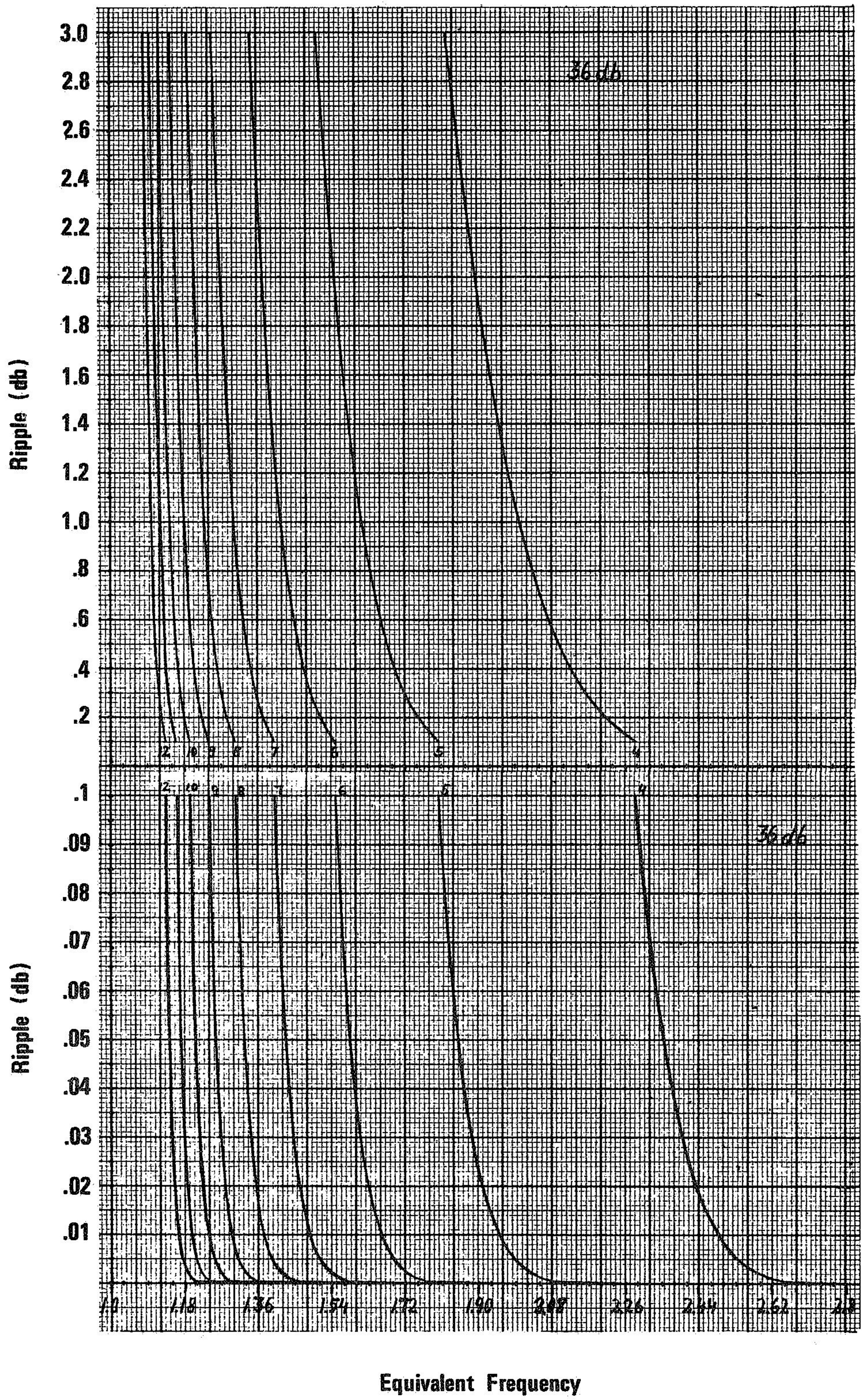



FIGURE 7-17.—Filter selection chart, 36 db attenuation

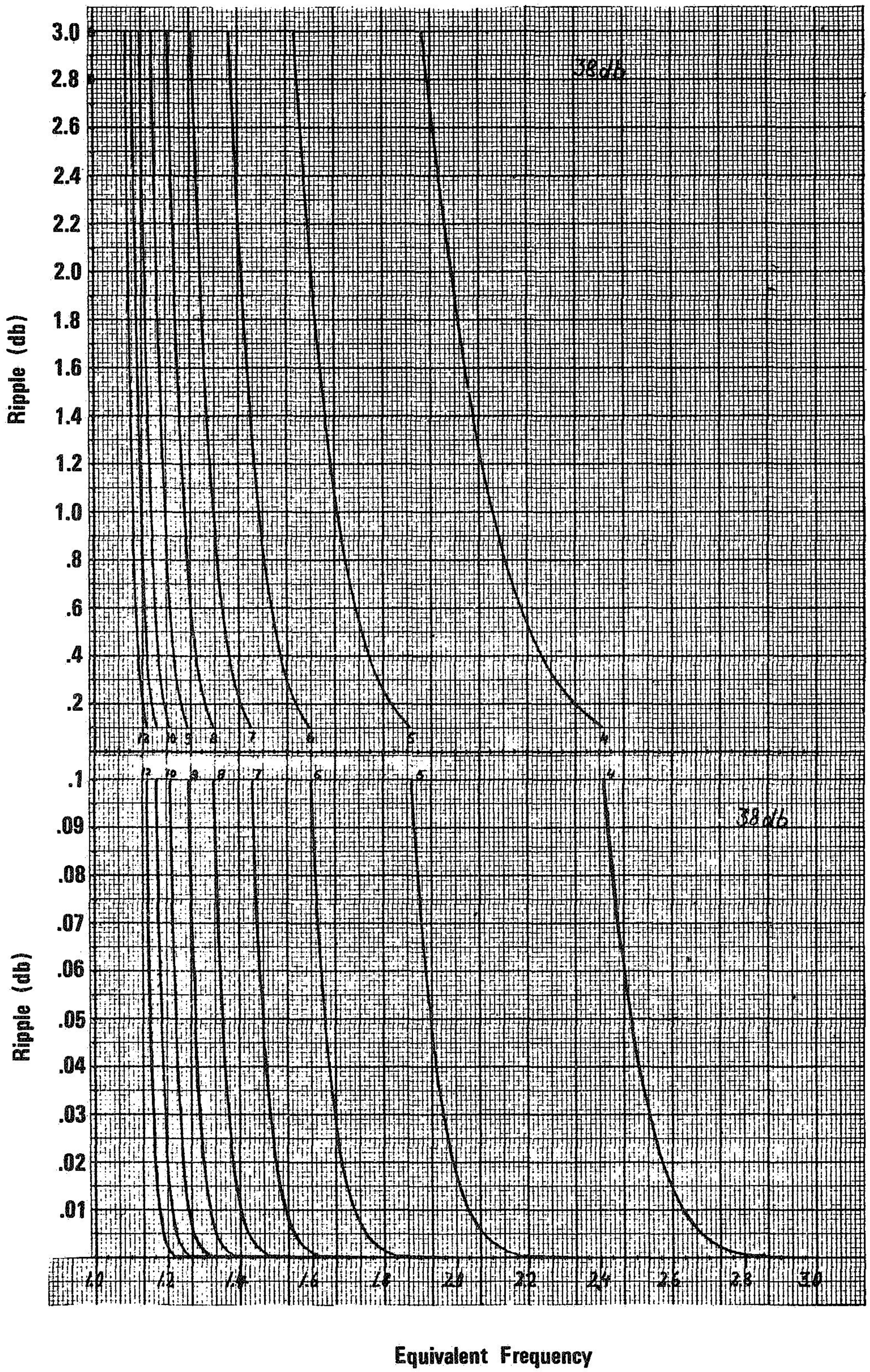



FIGURE 7–18.—Filter selection chart, 38 db attenuation

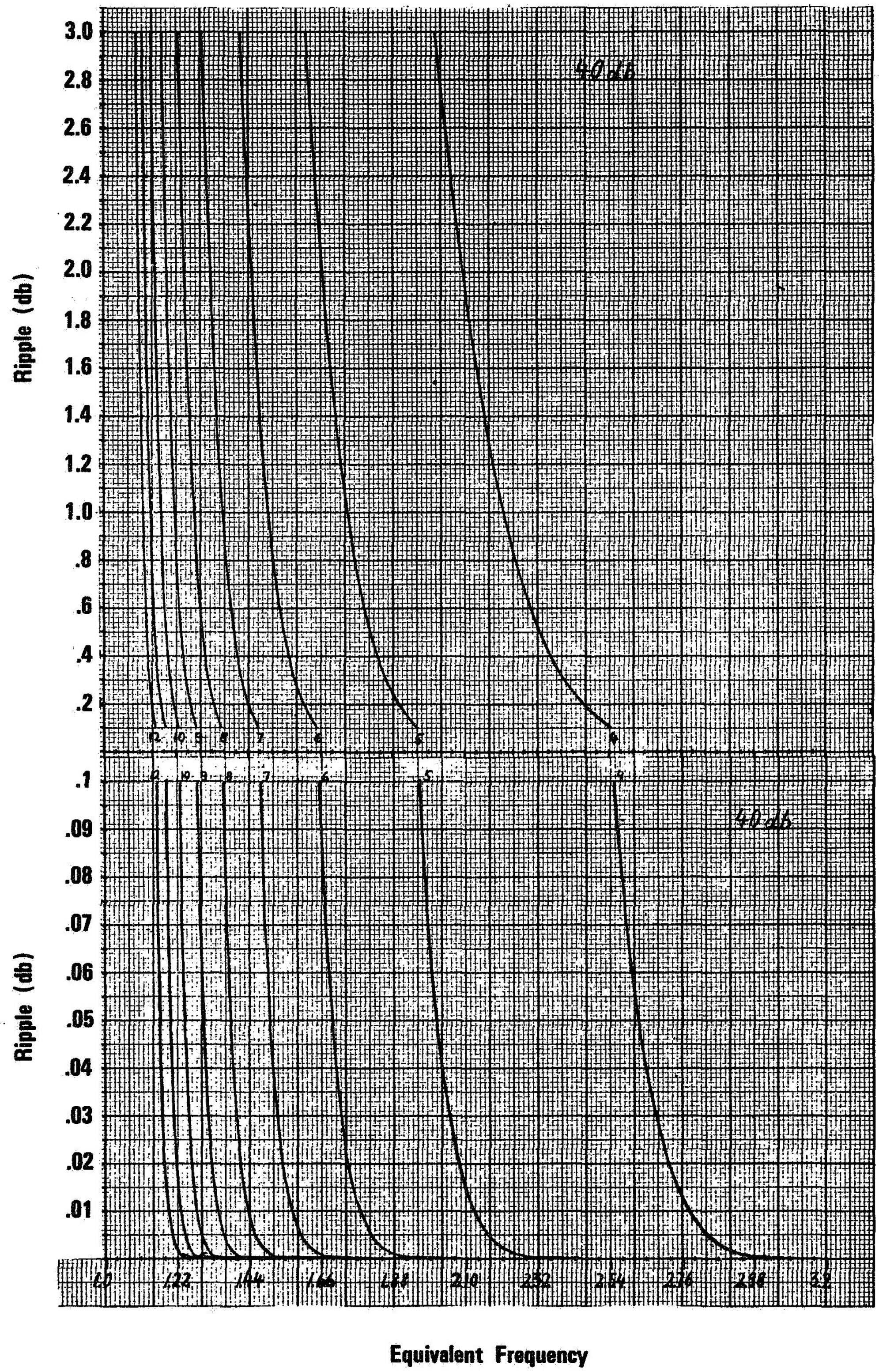



FIGURE 7–19.—Filter selection chart, 40 db attenuation

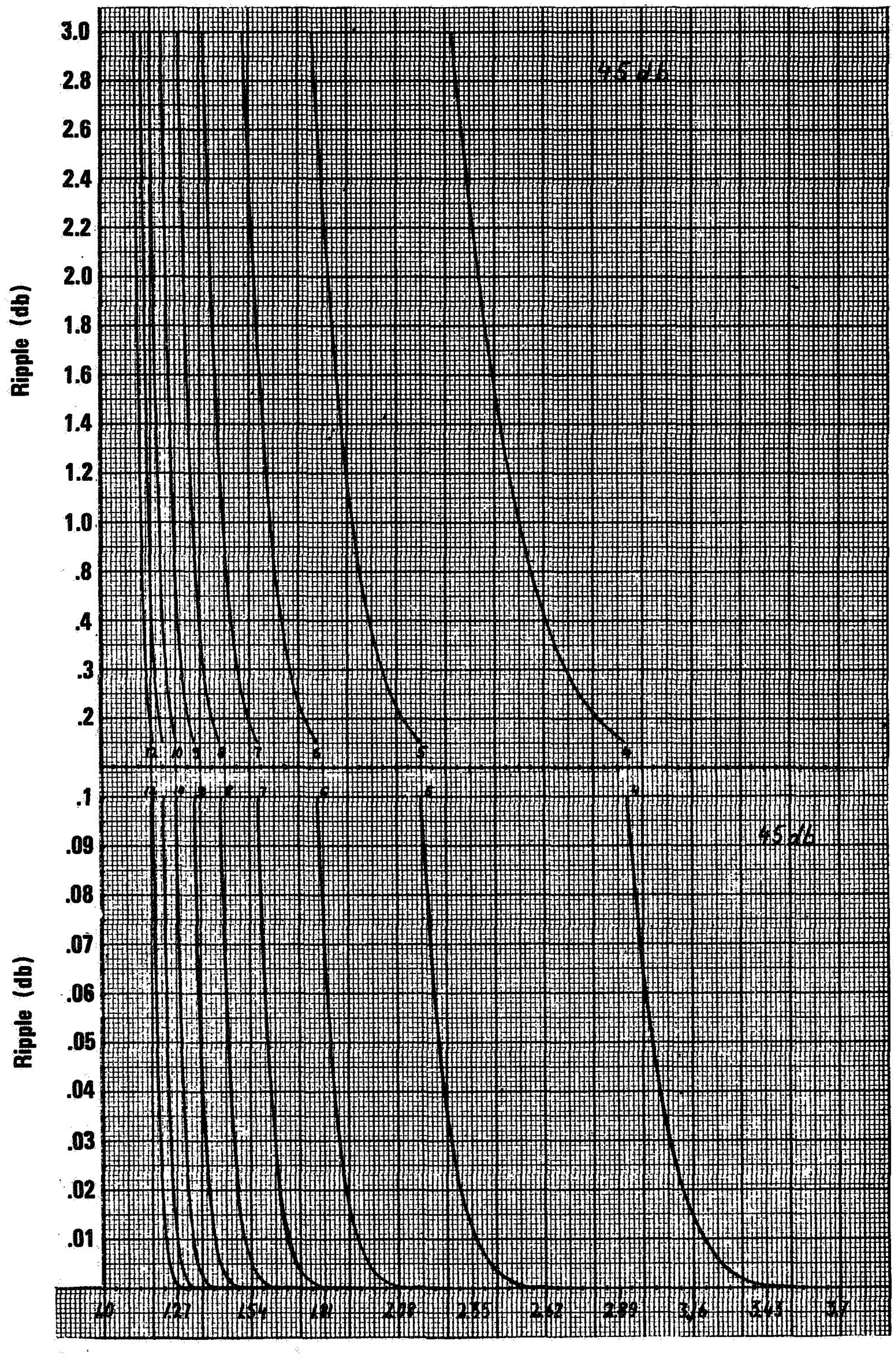



FIGURE 7–20.—Filter selection chart, 45 db attenuation

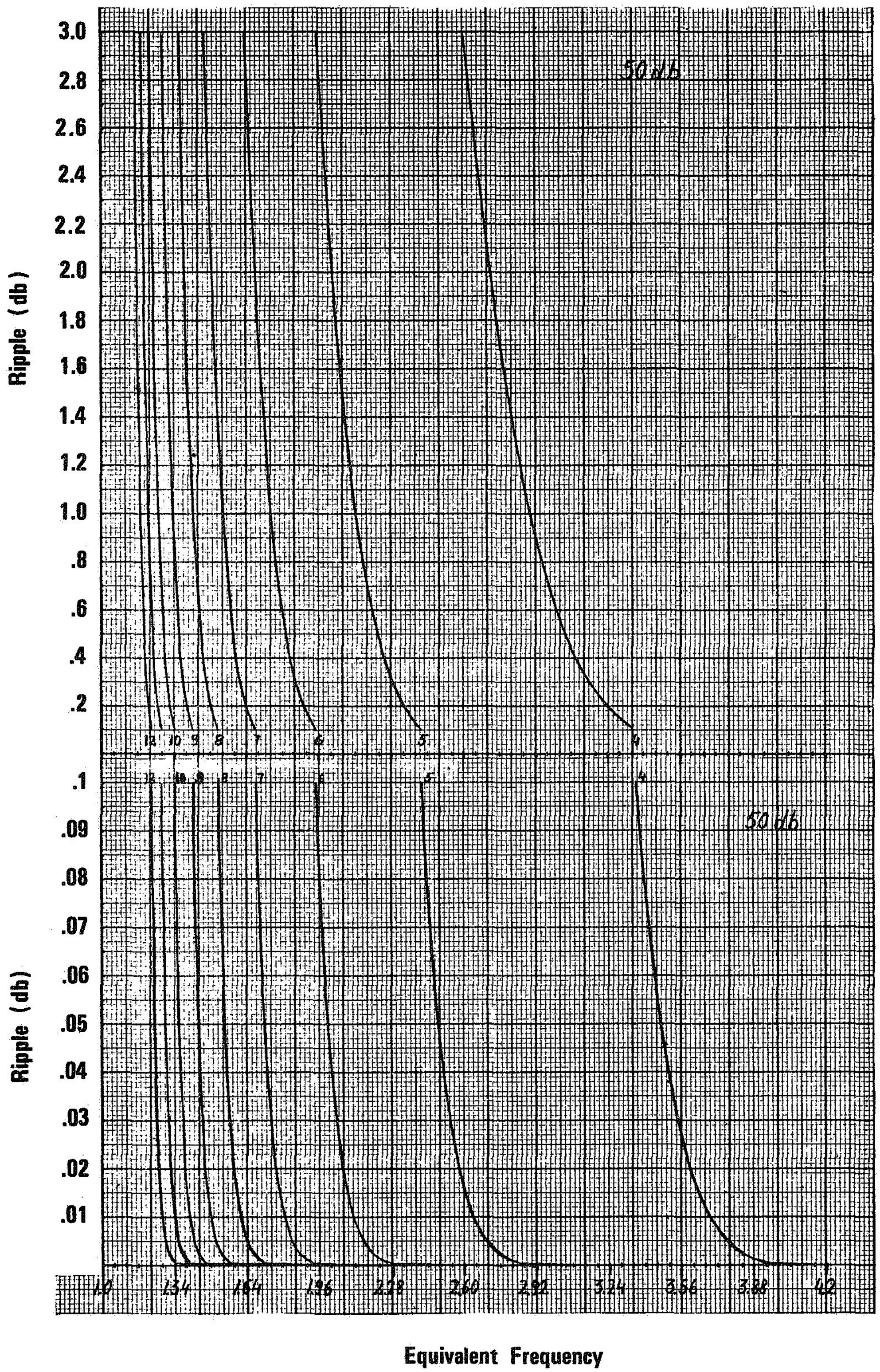



FIGURE 7–21.—Filter selection chart, 50 db attenuation

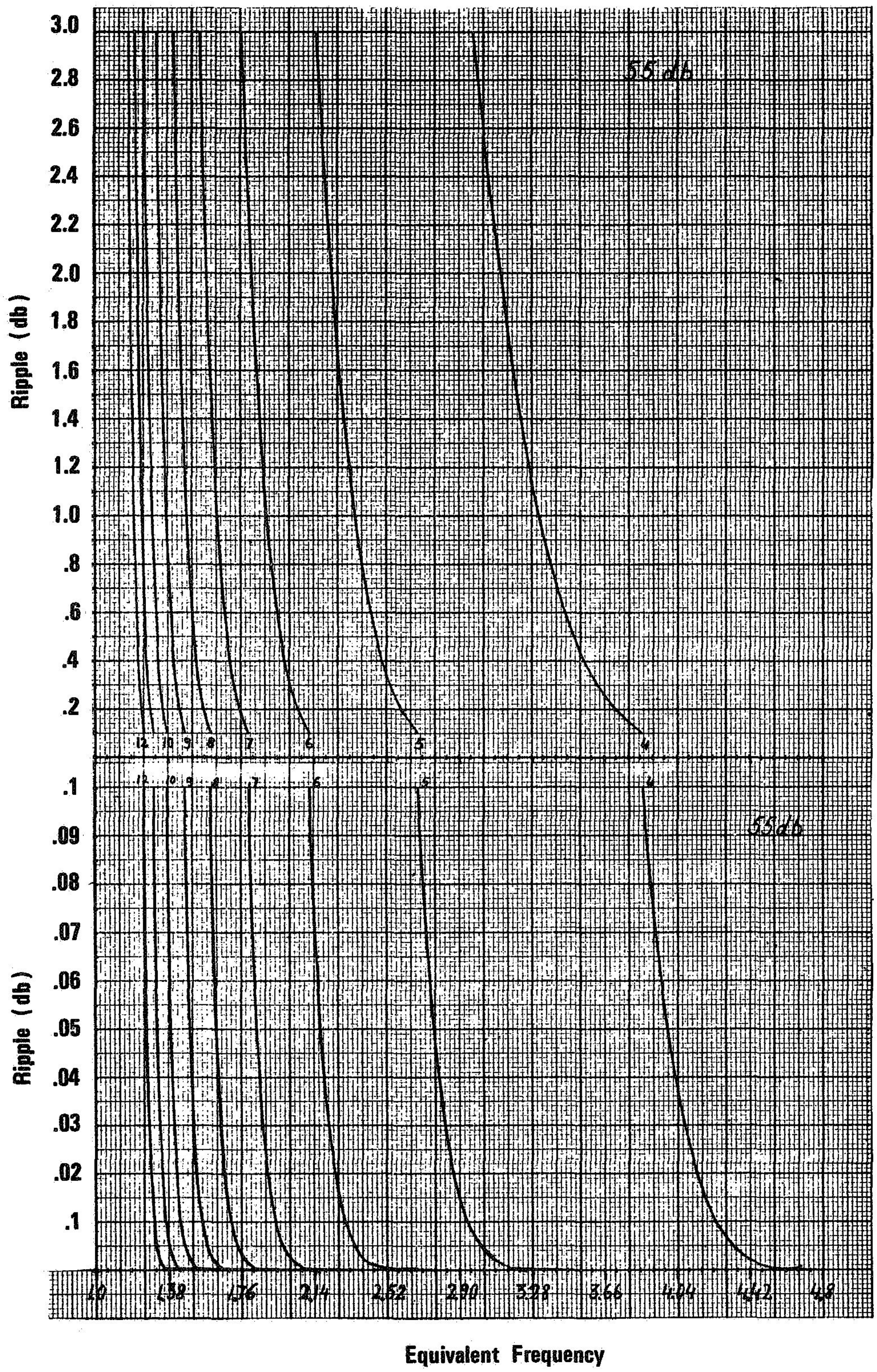



FIGURE 7–22.—Filter selection chart, 55 db attenuation

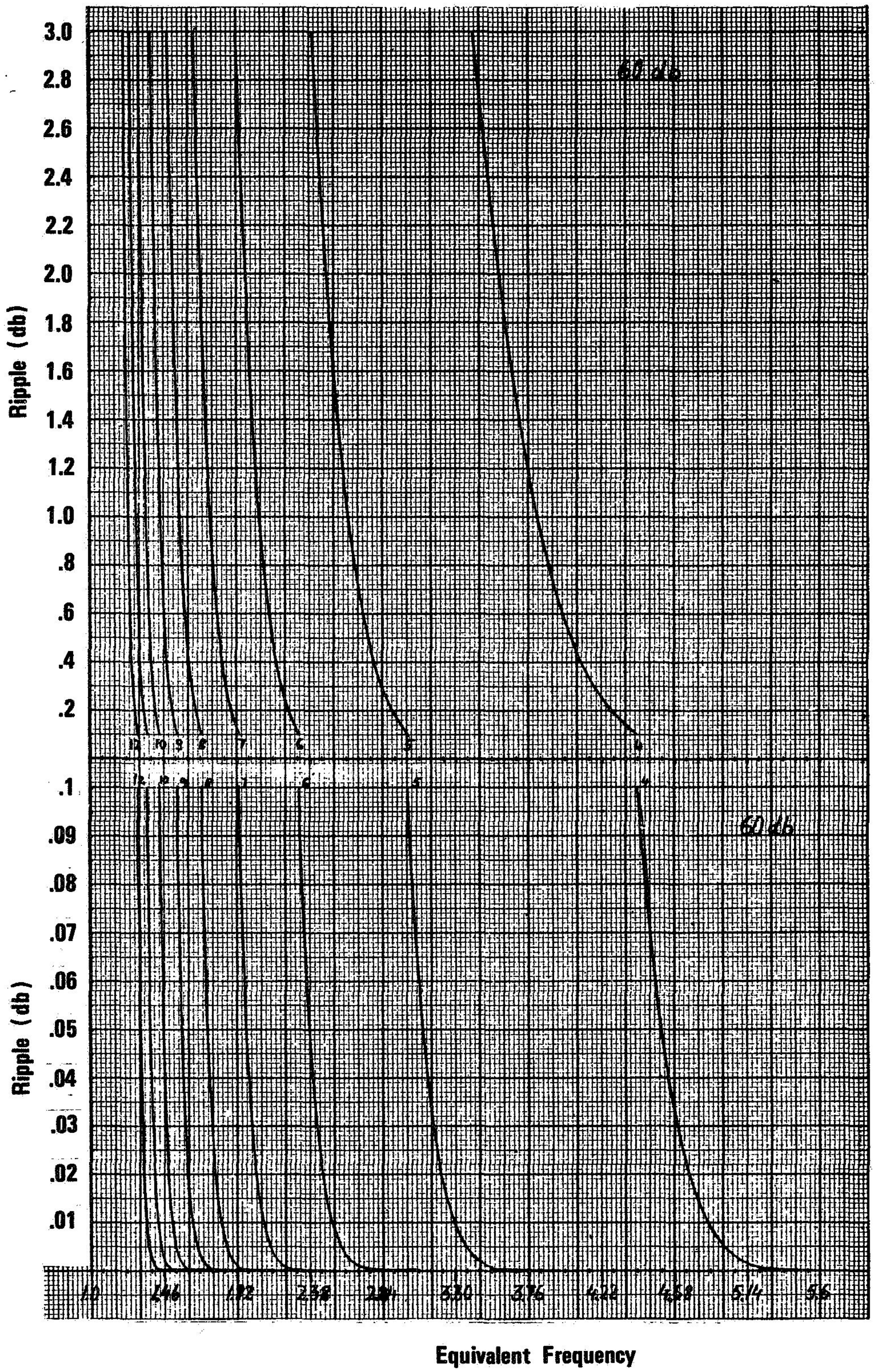



FIGURE 7–23.—Filter selection chart, 60 db attenuation

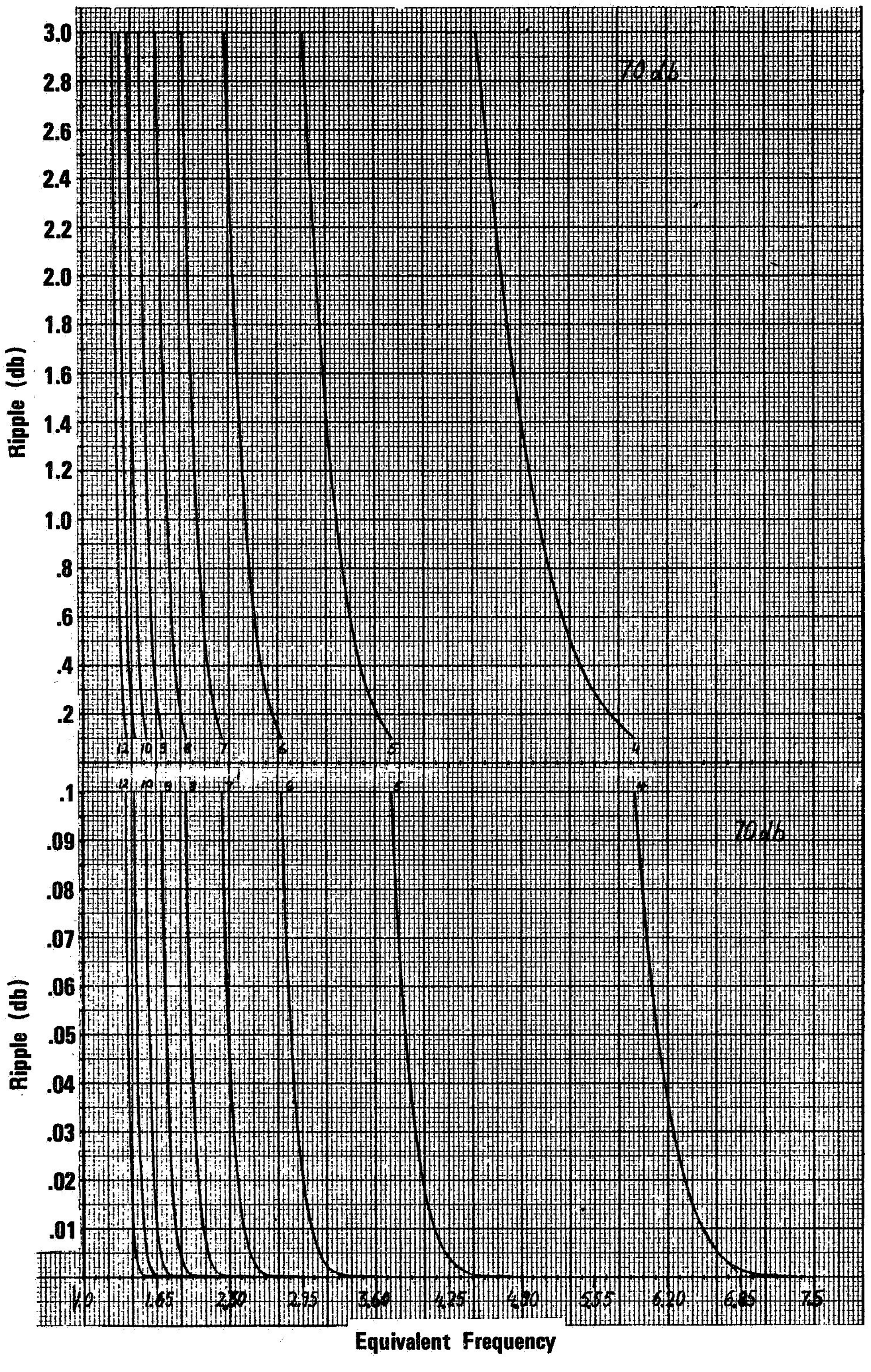



FIGURE 7-24.—Filter selection chart, 70 db attenuation

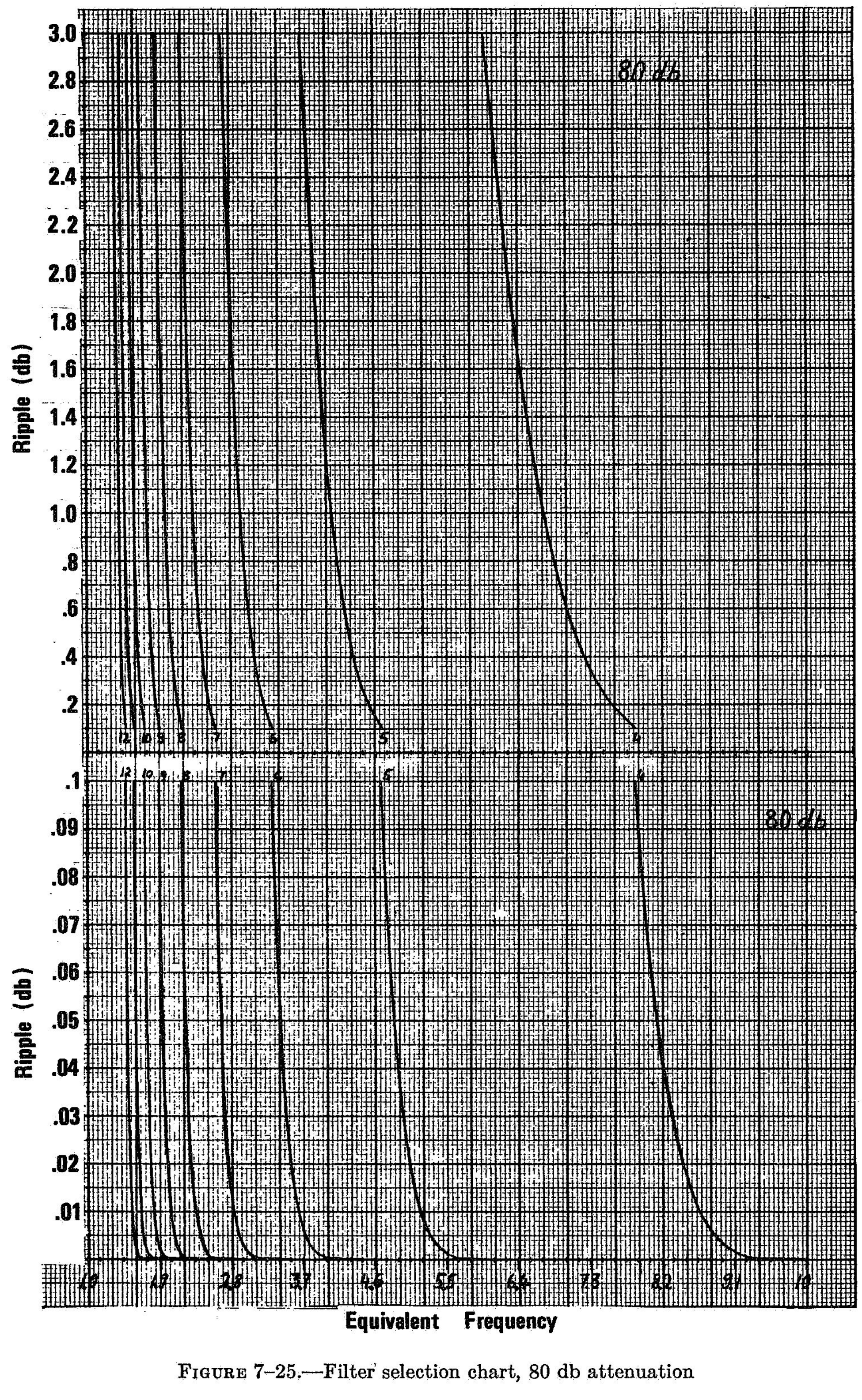



FIGURE 7–25.—Filter selection chart, 80 db attenuation

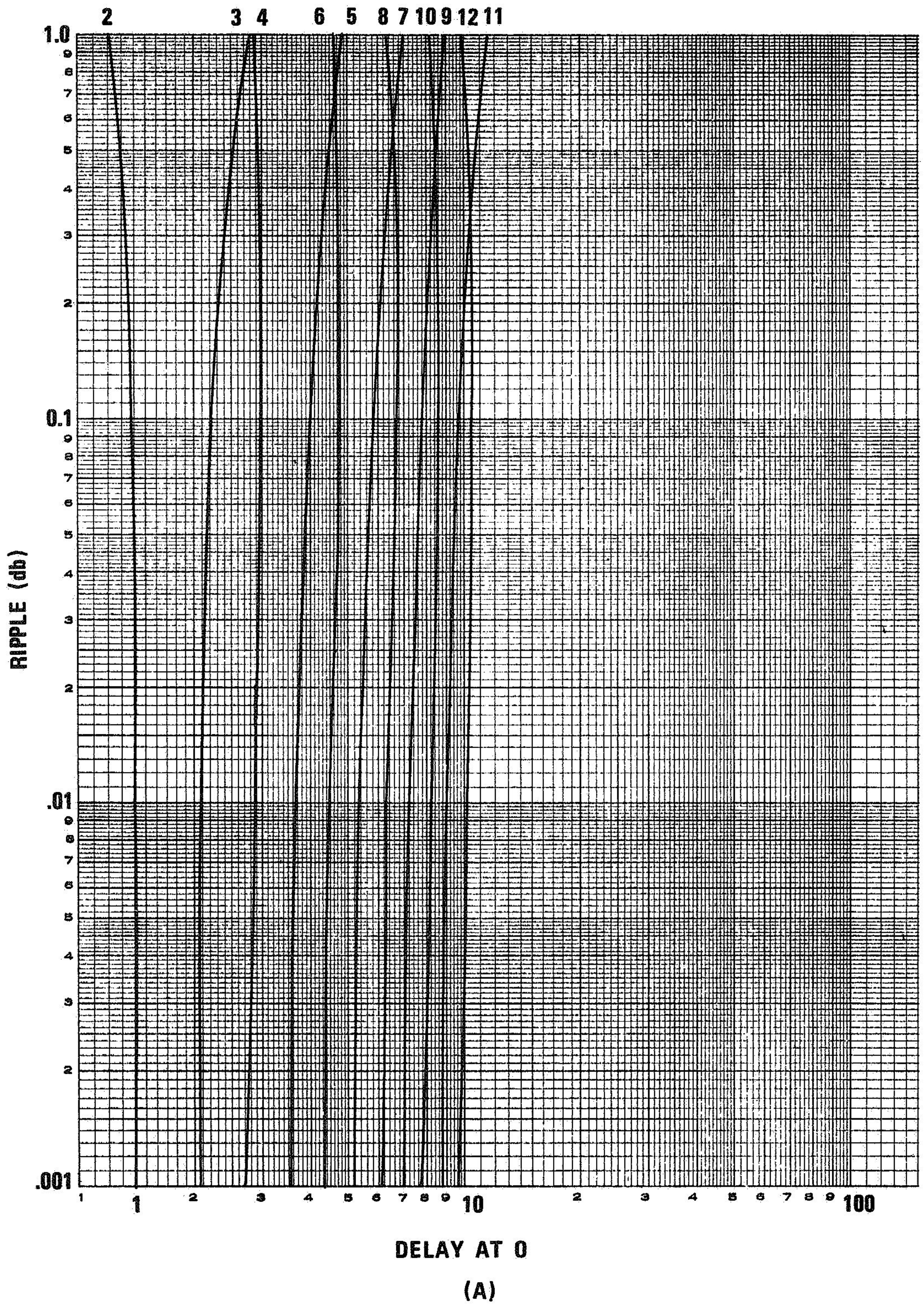



FIGURE 7–26.—(A) and (B) delay at equivalent frequency of 0 for various order filters

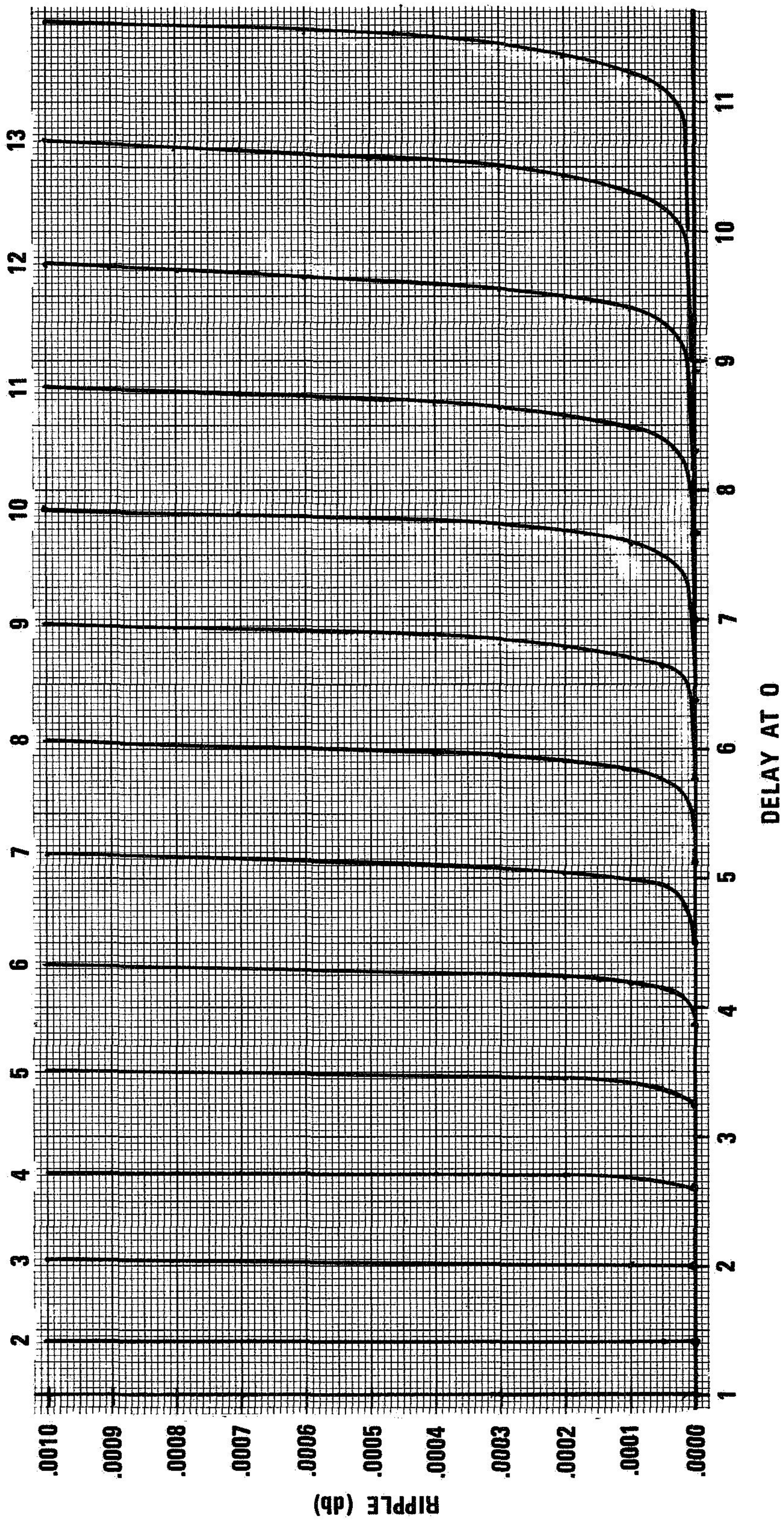



FIGURE 7-26.—Concluded

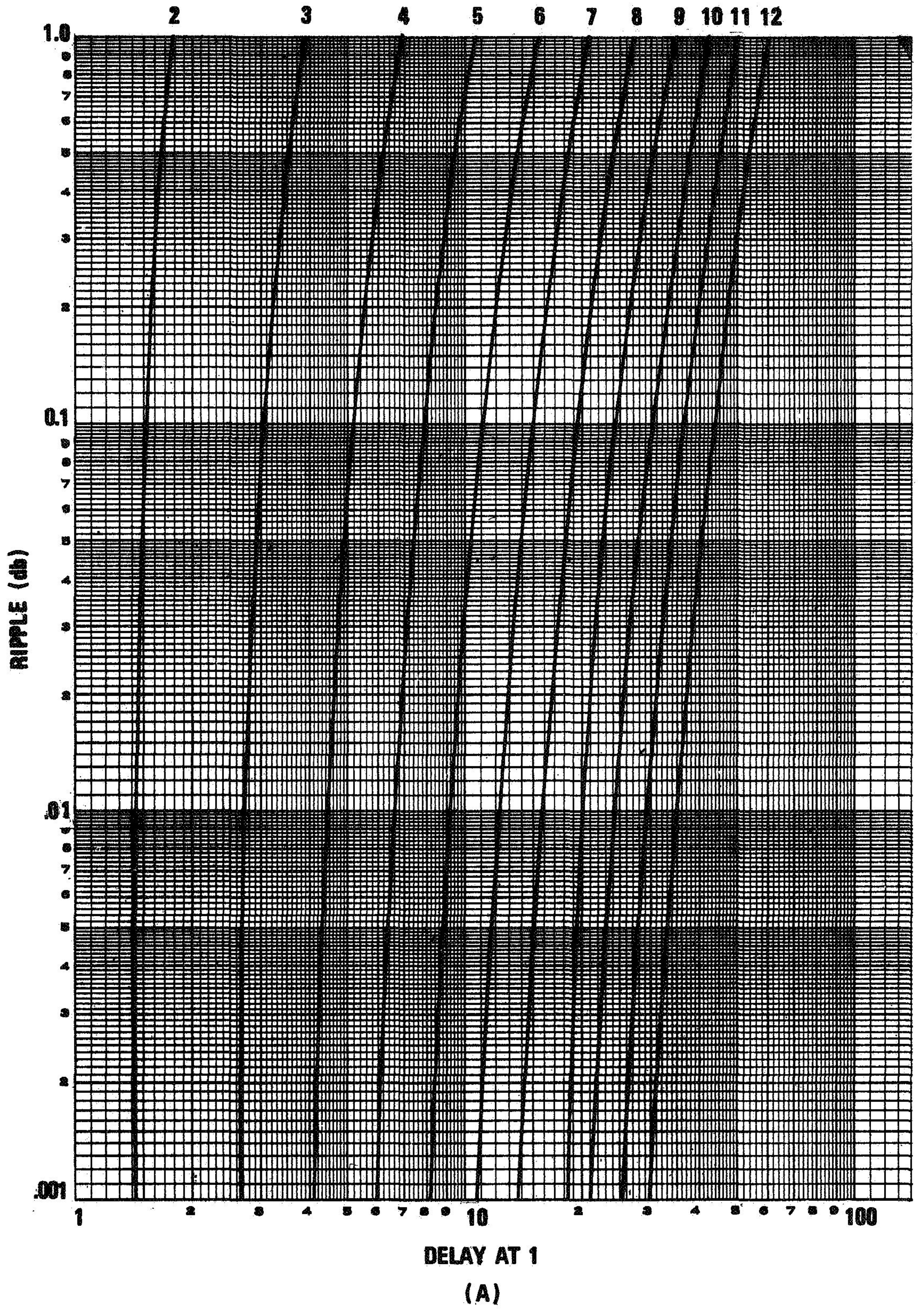



FIGURE 7-27.—(A) and (B) delay at equivalent frequency of 1 for various order filters

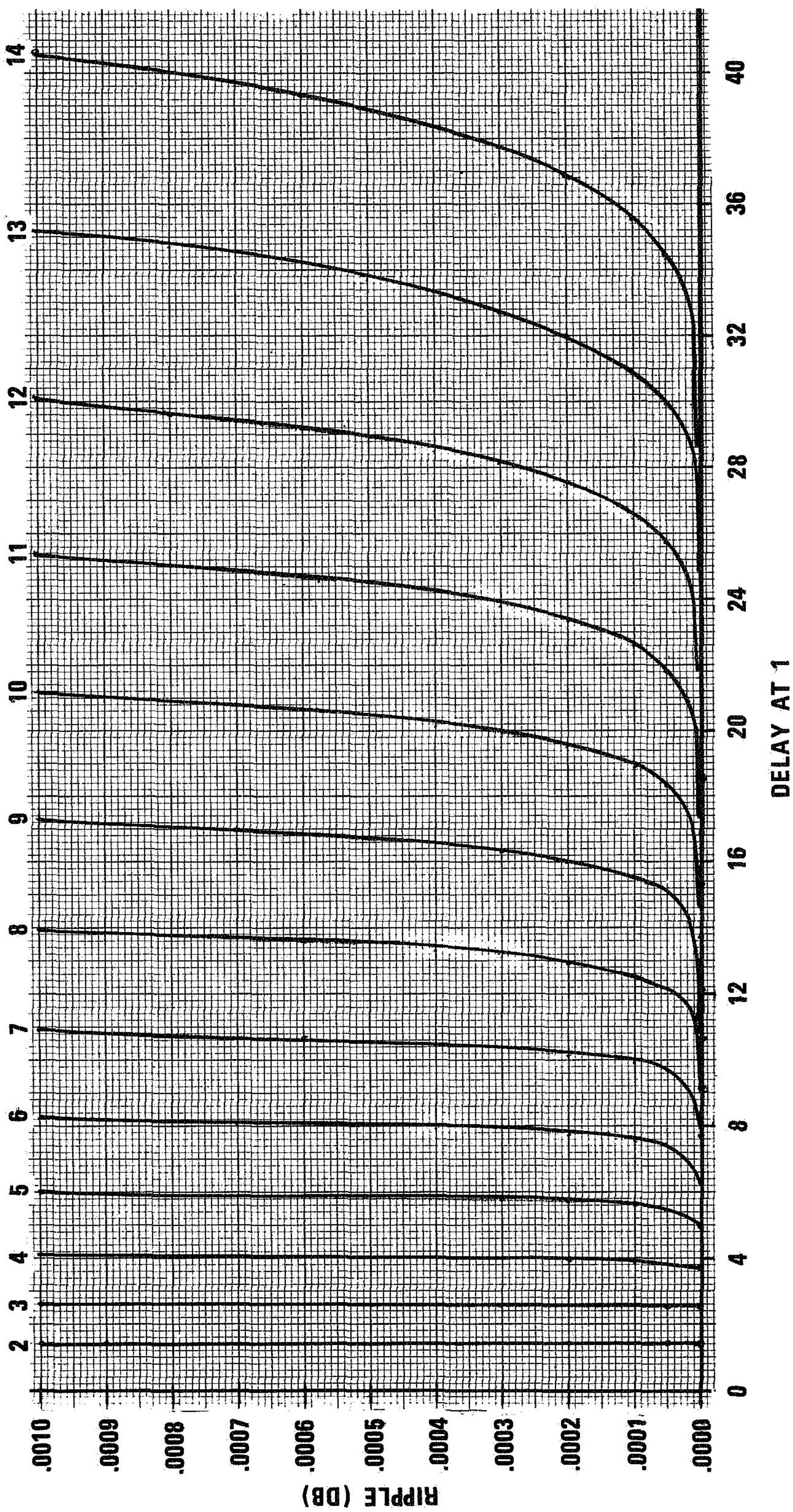



FIGURE 7-27.—Concluded

ORDER

| ATTENUATIONS (DB) | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 1.109 | 1.071 | 1.012 | 1.042 | 1.035 | 1.030 | 1.026 | 1.023 | 1.021 | 1.019 | 1.017 |
| 6 | 1.314 | 1.200 | 1.146 | 1.115 | 1.095 | 1.081 | 10.71 | 1.062 | 1.056 | 1.051 | 1.046 |
| 8 | 1.518 | 1.321 | 1.232 | 1.182 | 1.149 | 1.127 | 1.110 | 1.097 | 1.087 | 1.079 | 1.072 |
| 10 | 1.732 | 1.442 | 1.316 | 1.246 | 1.201 | 1.170 | 1.147 | 1.130 | 1.116 | 1.105 | 1.096 |
| 12 | 1.963 | 1.568 | 1.401 | 1.310 | 1.252 | 1.212 | 1.184 | 1.162 | 1.144 | 1.130 | 1.119 |
| 14 | 2.216 | 1.700 | 1.490 | 1.375 | 1.304 | 1.255 | 1.220 | 1.193 | 1.172 | 1.156 | 1.142 |
| 16 | 2.496 | 1.840 | 1.580 | 1.442 | 1.356 | 1.299 | 1.257 | 1.225 | 1.201 | 1.181 | 1.165 |
| 18 | 2.807 | 1.990 | 1.675 | 1.511 | 1.411 | 1.343 | 1.294 | 1.258 | 1.229 | 1.206 | 1.188 |
| 20 | 3.154 | 2.151 | 1.776 | 1.583 | 1.466 | 1.388 | 1.333 | 1.291 | 1.258 | 1.232 | 1.211 |
| 22 | 3.542 | 2.324 | 1.882 | 1.658 | 1.524 | 1.435 | 1.372 | 1.324 | 1.288 | 1.258 | 1.235 |
| 24 | 3.977 | 2.510 | 1.994 | 1.737 | 1.584 | 1.483 | 1.412 | 1.359 | 1.318 | 1.285 | 1.259 |
| 26 | 4.464 | 2.711 | 2.113 | 1.819 | 1.647 | 1.533 | 1.454 | 1.394 | 1.349 | 1.313 | 1.283 |
| 28 | 5.010 | 2.928 | 2.238 | 1.905 | 1.711 | 1.585 | 1.496 | 1.431 | 1.380 | 1.340 | 1.308 |
| 30 | 5.622 | 3.162 | 2.371 | 1.995 | 1.778 | 1.638 | 1.540 | 1.468 | 1.412 | 1.369 | 1.333 |
| 32 | 6.309 | 3.414 | 2.512 | 2.089 | 1.848 | 1.693 | 1.585 | 1.506 | 1.445 | 1.398 | 1.359 |
| 34 | 7.079 | 3.687 | 2.661 | 2.188 | 1.920 | 1.749 | 1.631 | 1.545 | 1.479 | 1.427 | 1.386 |
| 36 | 7.943 | 3.981 | 2.818 | 2.291 | 1.995 | 1.808 | 1.679 | 1.585 | 1.514 | 1.457 | 1.413 |
| 38 | 8.912 | 4.299 | 2.986 | 2.399 | 2.073 | 1.868 | 1.728 | 1.626 | 1.549 | 1.488 | 1.440 |
| 40 | 10.00 | 4.642 | 3.162 | 2.512 | 2.154 | 1.931 | 1.778 | 1.668 | 1.585 | 1.520 | 1.468 |
| 45 | 13.33 | 5.623 | 3.652 | 2.818 | 2.371 | 2.096 | 1.911 | 1.778 | 1.679 | 1.602 | 1.540 |
| 50 | 17.78 | 6.813 | 4.217 | 3.162 | 2.610 | 2.276 | 2.054 | 1.896 | 1.779 | 1.688 | 1.616 |
| 55 | 23.71 | 8.254 | 4.870 | 3.548 | 2.873 | 2.471 | 2.207 | 2.021 | 1.884 | 1.778 | 1.695 |
| 60 | 31.6 | 10.00 | 5.623 | 3.981 | 3.162 | 2.683 | 2.371 | 2.154 | 1.995 | 1.874 | 1.778 |
| 70 | 56.23 | 14.68 | 7.500 | 5.012 | 3.831 | 3.162 | 2.738 | 2.448 | 2.239 | 2.081 | 1.957 |
| 80 | 100.0 | 21.54 | 10.00 | 6.310 | 4.642 | 3.728 | 3.162 | 2.783 | 2.512 | 2.310 | 2.215 |
| | | | | | | | | | | | |

(A)

FIGURE 7–28.—(A) Zero ripple equivalent frequencies for various attenuations and (B) Zero ripple delay at equivalent frequencies of 0 and 1 for various order filters

| ORDER | DELAY AT 0 | DELAY AT 1 |
|---|---|---|
| 2 | 1.41 | 1.41 |
| 3 | 2.00 | 2.50 |
| 4 | 2.61 | 3.70 |
| 5 | 3.24 | 4.97 |
| 6 | 3.86 | 6.31 |
| 7 | 4.49 | 7.71 |
| 8 | 5.13 | 9.15 |
| 9 | 5.76 | 10.6 |
| 10 | 6.39 | 12.1 |
| 11 | 7.03 | 13.7 |
| 12 | 7.66 | 15.3 |
| 13 | 8.30 | 16.9 |
| 14 | 8.93 | 18.5 |
| 15 | 9.57 | 20.1 |
| 16 | 10.2 | 21.8 |
| 17 | 10.8 | 23.5 |
| 18 | 11.5 | 25.2 |
| 19 | 12.1 | 26.9 |
| 20 | 12.7 | 28.7 |

(B)

FIGURE 7-28.—Concluded

SECTION 8

# Design Examples

### 8.1 FILTER SELECTION EXAMPLE 1

As an example of the filter selection procedure, consider the problem of a bandpass filter specified as follows:

Bandwidth = 20 kHz
Center frequency = 70 kHz
Bandpass ripple $\leq$ 1 db
Required attenuation = 30 db at 100 kHz and 40 kHz
Terminal impedance = 1000 ohms

Good design procedure requires that the foregoing data be carefully inspected to assure that the requirements are reasonable and that all the required information is given. The foregoing specifications meet both these requirements.

The filter selection charts must be used to isolate the optimum filter design to meet the given specifications. The optimum design is the lowest-order filter that can meet the specifications and is the most economical selection. Use of the filter selection charts requires evaluation of the normalized lowpass equivalent frequency. Program ML4 is provided for this purpose. The instructions for using program ML4 are given in section 3.4. The list of data that must be provided to the program for this example is as follows:

Frequency of interest = 40 kHz and 100 kHz
Data factor = $-1$
$Q$ Factor = 25
Configuration factor = 3
Center frequency = 70 kHz
Bandwidth = 20 kHz

Two values are given for the frequency of interest, and the equivalent frequency for each of these values must be found. The program must be run twice—for 40 kHz and for 100 kHz. The data factor is $-1$ because Format 1 Data has been chosen. For the $Q$ factor, assume a value of 25. This is a reasonable estimate for circuit components in this frequency range. The configuration factor is 3 because the filter is to be a bandpass type. The center frequency (70 kHz) and bandwidth (20 kHz) are determined from the filter specifications. Figure 8–1 (A) and (B) give the results of the two runs of program ML4.

The smaller absolute value of the two equivalent frequency values is 2.55. This means that the attenuation requirement at 100 kHz is the most stringent. The equivalent frequency value 2.55 must be used to select a filter design from the filter selection charts. If the filter meets the 100 kHz–30 db attenuation requirement, it will more than meet the 40 kHz–30 db attenuation requirement.

By referring to the 30-db attenuation filter selection chart and using the equivalent frequency value of 2.55 and maximum ripple of 1 db, a rectangle can be formed as shown in figure 8–2. This rectangle is called the region of acceptable configuration, and any-order filter lying inside this rectangle will meet the design specifications. Observe that the lowest order that falls within this rectangle is a third-order filter.

The selection of the filter can now be made. For this example, a third-order filter with 1-db passband ripple is chosen. Now that the filter has been selected, program ML2B, ML3 or ML6 can be used for the actual network synthesis.

### 8.2 FILTER SELECTION EXAMPLE 2

As a second example of filter selection, con-



353–813 O—69——6

```
BAND PASS FILTER, CENTER FREQUENCY 0.700000E 05          BANDWIDTH 0.200000E 05

EQUIVALENT FREQUENCY -0.412500E 01          MULTIPLIER  0.127627E-02

Q FACTOR =     0.285714E 01

FREQUENCY OF INTEREST          0.400000E 05
```

(A)

```
BAND PASS FILTER, CENTER FREQUENCY 0.700000E 05          BANDWIDTH 0.200000E 05

EQUIVALENT FREQUENCY   0.254999E 01          MULTIPLIER  0.468096E-03

Q FACTOR =     0.285714E 01

FREQUENCY OF INTEREST          0.100000E 06
```

(B)

FIGURE 8–1.—(A) and (B) Results of program ML4 for selection example 1.

sider the problem of a low pass filter with a cutoff of 100 kHz, no passband ripple, and at least 20 db of attenuation at 170 kHz. Here the normalized lowpass equivalent frequency is 1.70. The zero-ripple design chart of figure 7–28 shows that a 5-pole filter is the optimum selection for this design specification. Entering the chart at 20-db attenuation, the first lowest number below 1.70 is 1.583, which corresponds to a fifth-order filter.

### 8.3 PASSIVE SYNTHESIS EXAMPLE 1

The filter selected in section 8.1, which was a third-order bandpass filter with 1-db passband ripple, is now realized with the passive synthesis programs. Assuming that a singly terminated network is required, program ML3 is used for the network synthesis. Program ML2B is used if the filter is to be doubly terminated, and ML3 is used if the filter is to be singly terminated. The instructions for using program ML3 are given in section 5.3. The list of data that must be provided to the program and the values for this example are as follows:

Filter order=3
Ripple factor=1 db
$Q$ factor=25
Configuration factor=3
Center frequency=70 kHz
Bandwidth=20 kHz
Terminal impedance=1000 ohms

The $Q$ factor of 25 was selected from a knowledge of the characteristics of the available circuit components. The configuration factor is three because the program is for a bandpass filter. The output of program ML3 for the foregoing data is shown in figure 8–3. Observe that the flat loss for this filter will be 1.06 db.

Figure 5–5 relates the Cauer circuit elements from the program to actual circuit components. Figure 8–4 (A) and (B) show the final filter design with and without tapped coils. Although the use of tapped coils is optional, the tapped-coil configuration is usually more convenient than the nontapped-coil case. Note that the capacitors are all the same size in the tapped-coil configuration.

### 8.4 PASSIVE SYNTHESIS EXAMPLE 2

The fifth-order lowpass Butterworth (0 db ripple) filter selected in section 8.2 is realized in passive form. Assuming a singly terminated network is specified, Cauer synthesis program ML3 should be used. The following data is read with the program:

Ripple = 0
Order = 5
$Q$ factor = assume 10.5
Configuration factor = 1 (lowpass)
Cutoff frequency = 100 kHz
Bandwidth = blank
Terminal impedance = 5 kilohms

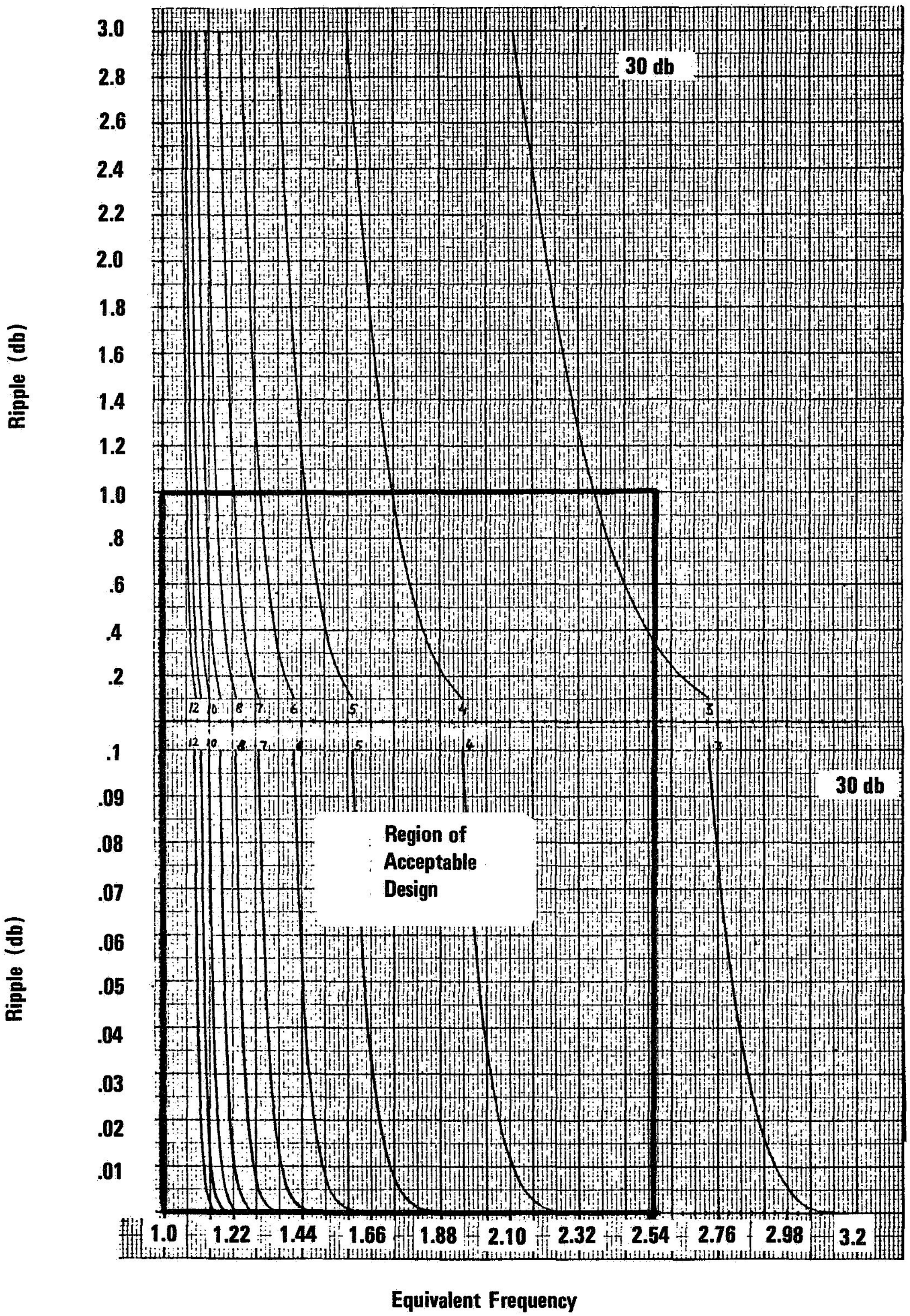



Figure 8-2.—Filter selection chart for selection example 1.

```
    RIPPLE =    0.100000E 01

THEORETICAL POLES

    1        -0.225671E 00          0.882290E 00
       -0.451343E 00
N ODD

BRANCH BANDWIDTH        0.280000E 04

FILTER BANDWIDTH    0.200000E 05

CENTER FREQUENCY         0.700000E 05



COIL         TAP 2 RATIO           TAP 3 RATIO

   1             0.491622E 01                      0.607178E 01


 COIL          FC. DEF. 2                          FC. DEF. 3

   1             0.139605E 05                      0.111835E 05

FLAT LOSS =      0.105774E 01

CAUER CIRCUIT ELEMENTS

I= 1        F=     0.164863E-01
I= 2        F=     0.115598E-07
I= 3        F=     0.108082E-01
TERMINAL RESISTANCE =     0.100000E 04

N =    3   Q =       25.000003   E =    0.508847E 00
```

FIGURE 8–3.—Results of program ML3 for synthesis example 1

The filter schematic specified by program ML3 is shown in figure 8–5. The theoretical response of a 5-pole lowpass filter, obtained from program ML1A, is shown in figure 8–6.

## 8.5 ACTIVE SYNTHESIS EXAMPLE 1

For active synthesis, the same lowpass filter characteristics used in the passive synthesis of section 8.4 were read into program ML6 with the exception that the $Q$ factor was assumed to be infinite. The output data of program ML6, shown in the sample printout of the program in figure 6–16, were used as a guide in synthesizing the filter with program ML7. The input data provided to program ML6 for synthesis of this example are as follows:

Filter order=5
Passband ripple=0 db
$Q$ factor=0
Configuration factor=1
Cutoff frequency=100 kHz
Bandwidth=blank

The bandwidth requirement is left blank since it is only utilized for the bandpass case. A $Q$ factor of zero is taken by the computer to be an infinitely high value. A fifth-order, lowpass Butterworth filter having a 3-db cutoff frequency at 100 kHz was constructed according to the circuit diagram shown in figure 8–7. The components values were obtained directly from the sample computer printout shown for program ML7 in figure 6–18. The input data to program ML7 for this example are listed at the end of section 6.3.

Without any tuning being performed, the measured amplitude response was found to coincide very closely with the theoretical response. The normalized frequency response curves for both the measured and theoretical responses are shown in figure 8–8. The theoretical phase response and group delay response for this filter are shown in figures 8–9 and 8–10, respectively. The data for these theoretical response curves were obtained from program ML1A. These curves show the typical phase and group delay responses for a lowpass filter.

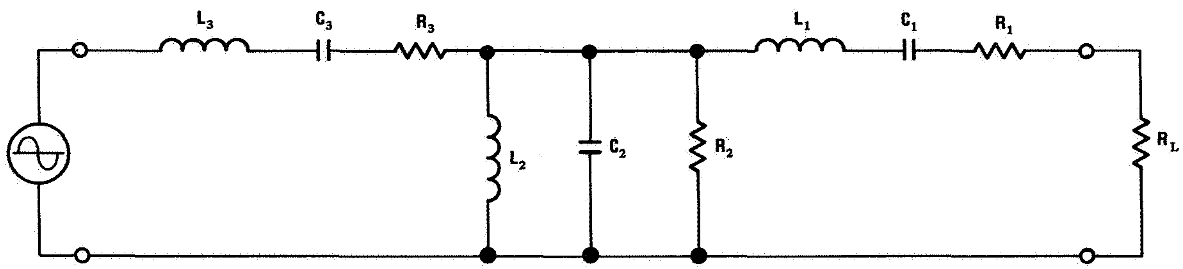
## 8.6 ACTIVE SYNTHESIS EXAMPLE 2

Another active filter circuit was constructed from the data obtained from the second sample computer printout shown for program ML7 in figure 6–19. The filter is a second order, bandpass Butterworth having a bandwidth of 40 kHz at a center frequency of 45.826 kHz. The 3-db cutoff frequencies are at 30 kHz and 70 kHz. The schematic diagram of the filter is shown in figure 8–11. The normalized frequency response curves for the measured and theoretical responses are shown in figure 8–12. Without tuning, the measured response curve was almost identical with the theoretical response.

The theoretical phase response and group delay response for this filter are shown in figures 8–13 and 8–14, respectively. It can be seen that the group delay deviation is more severe for the second-order bandpass filter than for the fifth-order lowpass filter. The degree of phase distortion for these filters is proportional to their group delay variation.

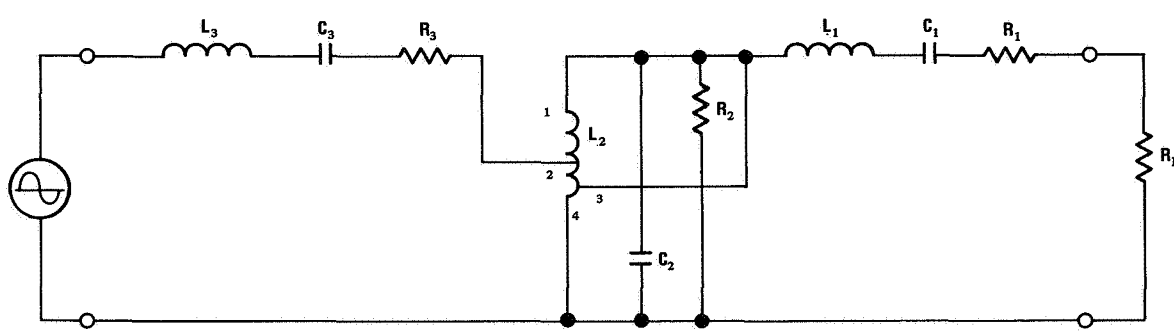
## 8.7 ACTIVE SYNTHESIS EXAMPLE 3

The realistic network models utilized by the active filter design programs allow filter re-




| | |
|---|---|
| $L_1$ | 16.5 mh |
| $C_1$ | 313 pf |
| $L_2$ | .45 mh |
| $C_2$ | .0116 μf |
| $L_3$ | 10.8 mh |
| $C_3$ | 478 pf |
| $R_L$ | 1000 Ω |
| $R_1 R_2 R_3$ | Selected to give a branch bandwidth of 2.8 kHz |

(A)




| | |
|---|---|
| $L_1, L_2, L_3$ | 16.5 mh. |
| $C_1, C_2, C_3$ | 313 pf |
| $R_1, R_2, R_3$ | Selected to give a branch bandwidth of 2.8 kHz |
| Tap 2 Ratio | 4.92 |
| Tap 3 Ratio | 6.07 |
| $R_L$ | 1000 Ω |

(B)

FIGURE 8-4.—Cauer synthesis configurations (A) nontapped filter and (B) tapped filter

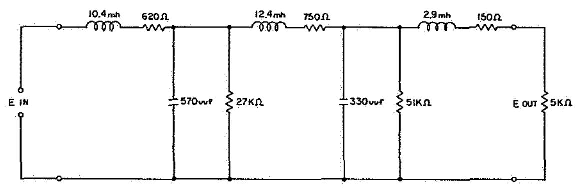



FIGURE 8-5.—Schematic of fifth-order lowpass Butterworth (passive)

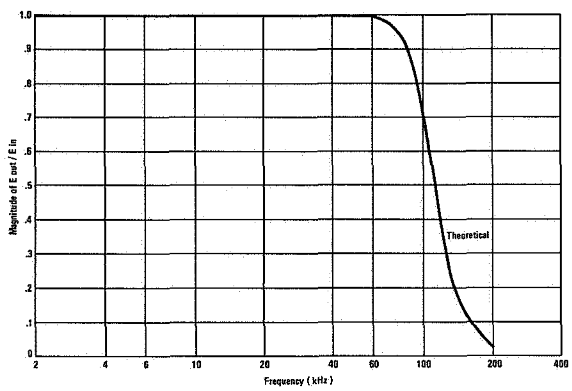



FIGURE 8-6.—5 pole lowpass Butterworth response

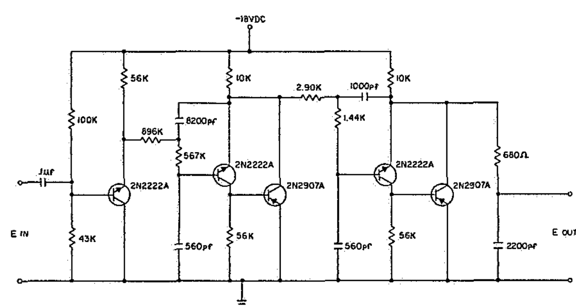



FIGURE 8-7.—Schematic of fifth-order lowpass Butterworth (active)

sponse curves to be realized with much greater accuracy than is possible with conventional network models. To demonstrate the improvement in accuracy obtained with the realistic models, a bandpass filter was designed and constructed from both the conventional and realistic models. Identical circuit configurations and amplifier elements were used for both filters so that the relative accuracy of the filters could be compared. The bandpass filter selected was one specified for use in a

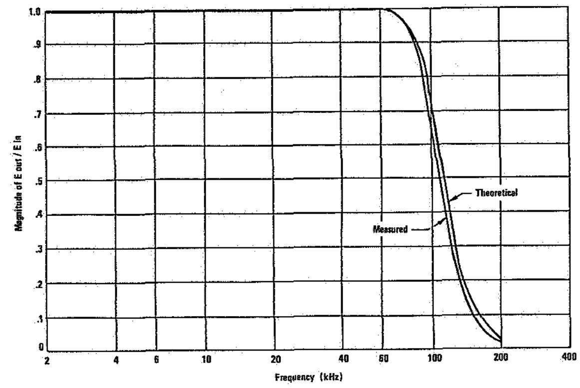



FIGURE 8-8.—Magnitude response of fifth-order lowpass Butterworth (active)

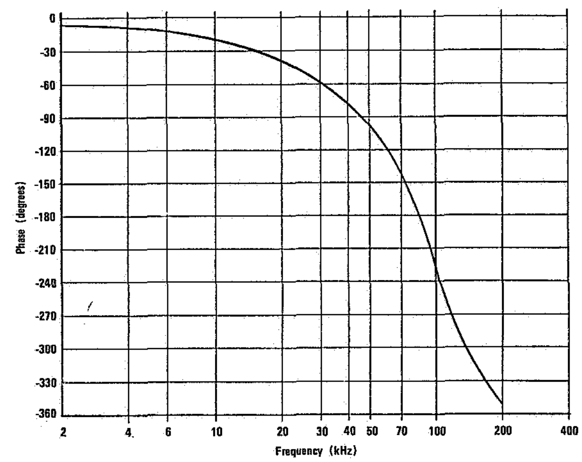



FIGURE 8-9.—Phase response of fifth-order lowpass Butterworth

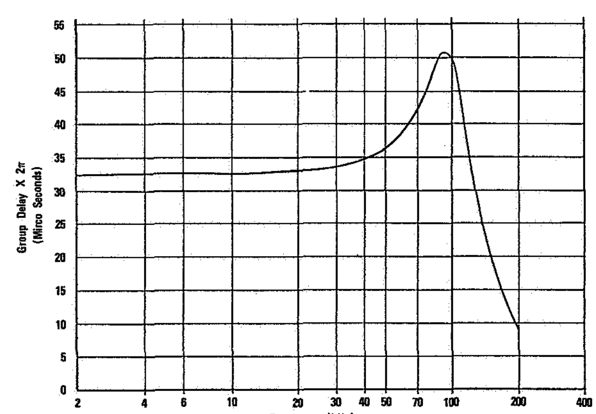



FIGURE 8-10.—Group delay response of fifth-order lowpass Butterworth

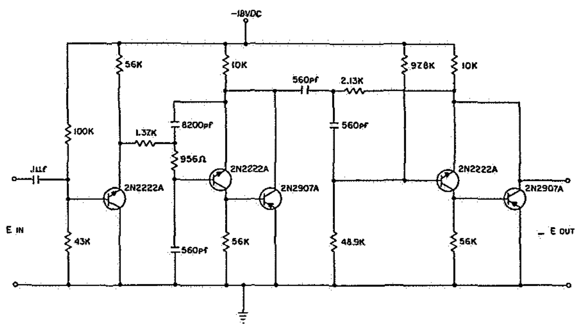



FIGURE 8–11.—Schematic of second-order bandpass Butterworth (active)

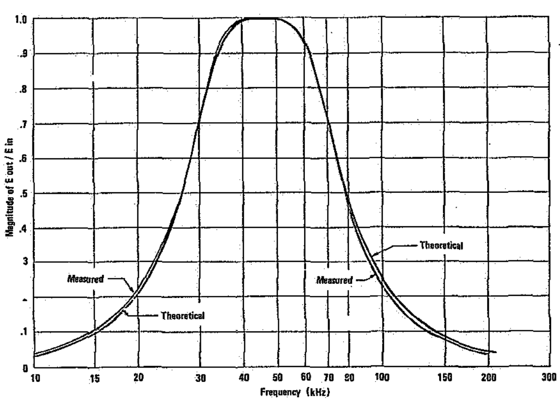



FIGURE 8–12.—Magnitude response of second-order bandpass Butterworth (active)

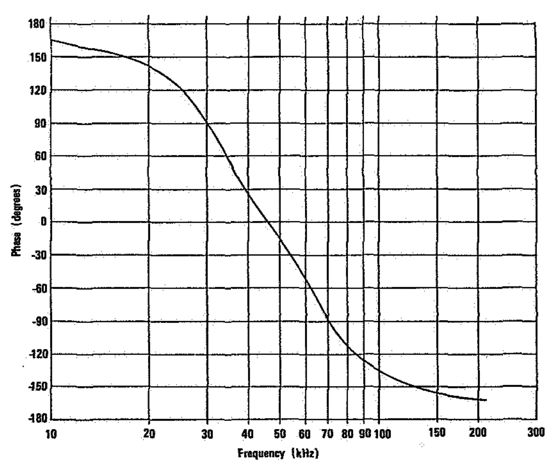



FIGURE 8–13.—Phase response of second-order bandpass Butterworth

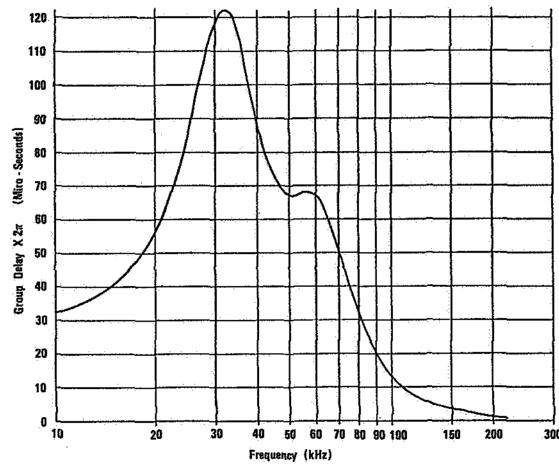



FIGURE 8–14.—Group delay response of second-order bandpass Butterworth

constant bandwidth frequency translator for FM systems. Because of phase distortion requirements, the filter consisted of an eighth-order, lowpass Butterworth in series with a sixth-order highpass Butterworth.

The schematic diagrams for the filter designed from the conventional models and from the realistic models are given in figures 8–15 and 8–16, respectively. Although the bandpass filter is constructed by cascading alternate lowpass and highpass sections, the response curves of the individual lowpass and highpass filters are shown separately. The 3-db attenuation cutoff frequencies occur at 192 kHz for the lowpass and at 128 kHz for the highpass filter.

The measured and theoretical magnitude response curves for the filter designed with the conventional networks are compared in figure 8–17. The measured response deviates significantly from the theoretical response. The same filter was then designed with the realistic networks. The measured and theoretical response curves for this filter are compared in figure 8–18. Here, the measured response coincides very closely with the theoretical response. In neither case were any of the networks tuned by adjustment of component values after the filters were constructed.

Of the seven amplifiers used in each bandpass filter, five have an output resistance of 24 ohms, and two have an output resistance of 7 ohms.

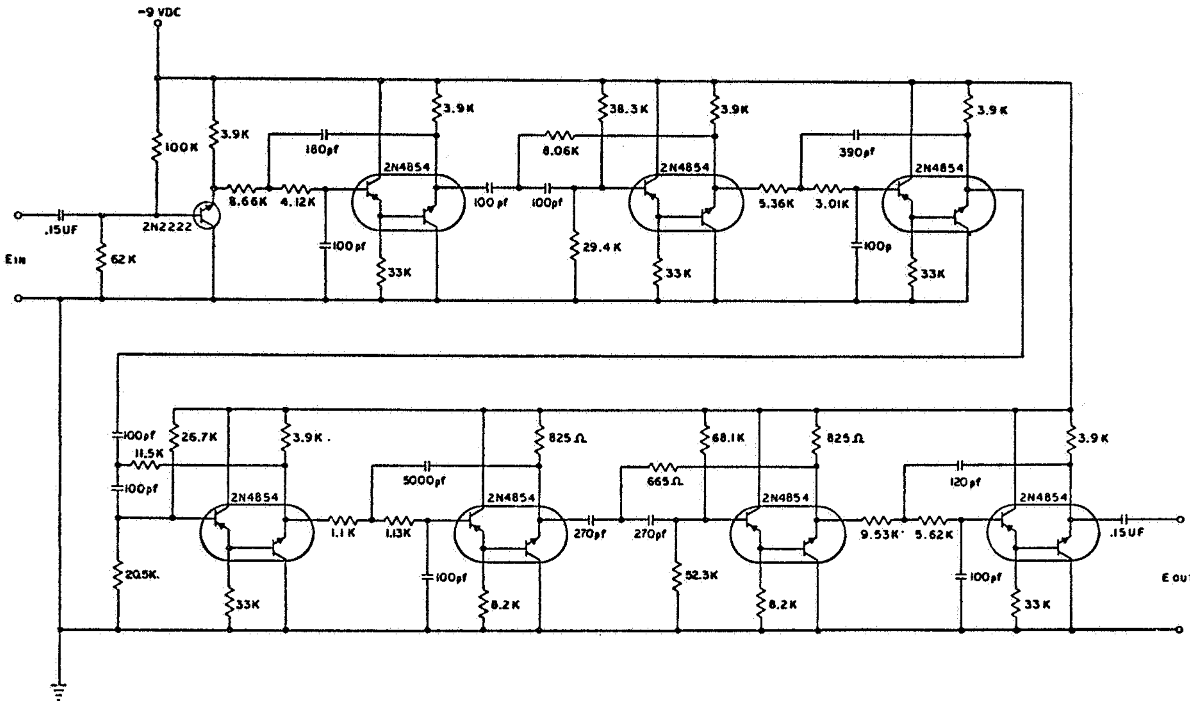



FIGURE 8-15.—Schematic of bandpass filter using conventional quadratic networks

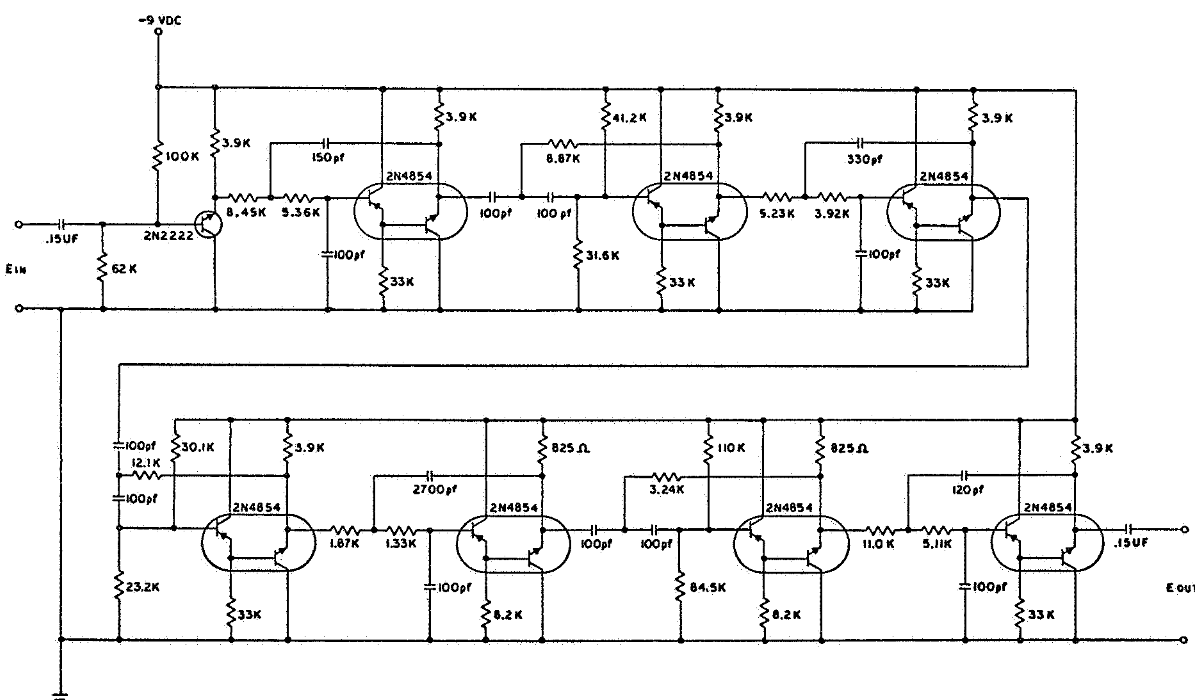



FIGURE 8-16.—Schematic of bandpass filter using realistic quadratic networks

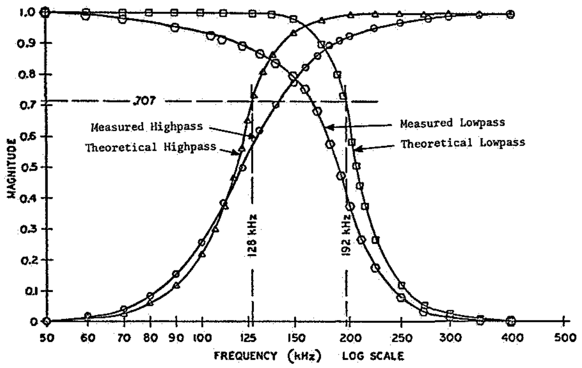



FIGURE 8–17.—Magnitude response of filter using conventional quadratic networks

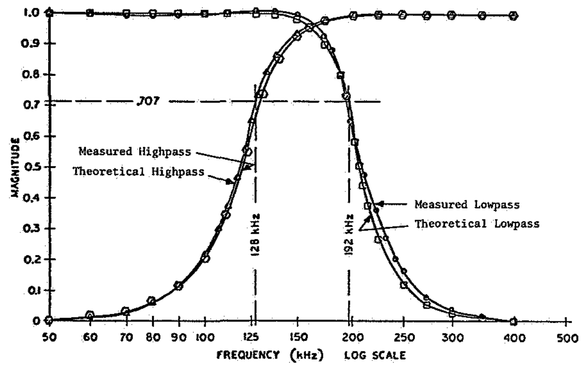



FIGURE 8–18.—Magnitude response of filter using realistic quadratic networks

All have an input capacitance of 7 picofarads. If the output resistance or input capacitance of the amplifier elements were higher, the response curve of the conventional networks would deviate further from the theoretical response. This example shows that the improved design equations used in the computer programs can be used to realize a desired filter response more accurately than is possible with the conventional design equations.

# APPENDIX A

# Computer Programs

Figures A–1 through A–10 contain the computer main line programs used in the filter design procedure. The programs, which are written in FORTRAN IV, are listed below.

ML1A—Frequency Analysis
ML1B—Transient Analysis
ML2A—Darlington Synthesis (even order)
ML2B—Darlington Synthesis (odd order)
ML3—Cauer Synthesis
ML4—Transformation Calculations
ML5—Chart Plotter
ML6—Active Filter Design
ML7—Active Filter Synthesis
ML8—Temperature-Dependent Response

```
// JOB
// FOR
*IOCS(CARD, 1132 PRINTER, TYPEWRITER, DISK)
*ONE WORD INTEGERS
* LIST SOURCE PROGRAM
*NAME ML1A
C      MAIN LINE ONE
       DIMENSION APHA(10), BETA(10)
       READ(2,4)N,R,Q
       READ(2,94)YY,FO,BB
       YYY=YY
       READ(2,94)FMAX,FMIN,DF
C
C      YY=1  LOW PASS
C      YY=2  HIGH PASS
C      YY=3  BAND PASS
C      YY=4  BAND STOP
       CALL CARIP(R,E)
       XN=N
       XN=XN/2.
       XI=1.
    10 T=XN-XI
       IF(T)1,2,3
     3 XI=XI+1.
       GO TO 10
C
C      N EVEN
     2 K=N/2
       WRITE(3,31)
    31 FORMAT(1X,'N EVEN')
       L=0
       M=0
       NN=0
       CALL CAPOE(E,N,APHA,BETA)
       CALL CARES(YYY,FO,BB,FMAX,FMIN,DF,K,L,M,NN,
      1APHA,BETA,APHA,BETA,0.,0.)
       GO TO 500
C
C      N ODD
     1 K=(N-1)/2
       WRITE(3,32)
    32 FORMAT(1X,'N ODD')
       L=0
       M=1
       NN=0
       CALL CAPOO(E,N,APHA,BETA,SPOL)
       CALL CARES(YYY,FO,BB,FMAX,FMIN,DF,K,L,M,NN,
      1APHA,BETA,APHA,BETA,SPOL,SPOL)
       GO TO 500
     4 FORMAT(I5,F15.0,F10.0)
    94 FORMAT(3F10.0)
   500 STOP
       END
```

FIGURE A–1.—ML1 A program

```
// JOB
// FOR
*IOCS(CARD, 1132 PRINTER, TYPEWRITER, DISK)
*NAME MTS
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
C
C      TEST OF ATRAN
       READ(2,1)N
       READ(2,2)WO,R
       READ(2,3)W,TMAX,TMIN,DT
       WRITE(3,4)N,WO,R,W
       CALL CARIP(R,E)
       CALL ATRAN(N,W,WO,E,TMAX,TMIN,DT)
     1 FORMAT(I5)
     2 FORMAT(2F10.0)
     3 FORMAT(4F10.0)
     4 FORMAT(5X,I2,2X,11HPOLE FILTER,//,5X,4HWO =,E16.5,
    112X,3HR =,E14.6,//,5X,3HW =,E14.6,///)
       STOP
       END
```

FIGURE A–2.—ML1B program

```
// JOB
// FOR
*IOCS(CARD, 1132 PRINTER, TYPEWRITER, DISK)
*NAME ML2
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      DIMENSION AP(10),AZ(10)
C     N EVEN
C     MASTER PROGRAM
  500 READ(2,4)N,R,Q
      CALL CARIP(R,E)
      READ(2,94)YYY,FO,BB,Z
      A7N=N
      A8N=A7N/2.
      A8I=1.
    5 A9=A8N-A8I
      IF(A9)1,2,3
    3 A8I=A8I+1.
      GO TO 5
    1 CONTINUE
      GO TO  15
    2 CALL ANEV(N,E,YYY,FO,BB,FMAX,FMIN,DF,Q,AP,AZ)
   15 CALL AFIN(AP,AZ,N,Q,Z,FO,YYY,BB)
      WRITE(3,38)N,Q,E
    4 FORMAT(I5,F15.7,F10.0)
   94 FORMAT(4F10.0)
   38 FORMAT(1X,//,5X,3HN =,I5,3X,3HQ =,F15.6,3X,3HE =,E15.6,//)
      GO TO 500
  501 STOP
      END
```

FIGURE A–3.—ML2A program

```
// JOB
// FOR
*IOCS(CARD, 1132 PRINTER, TYPEWRITER, DISK)
*NAME ML2
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      DIMENSION AP(10),AZ(10)
C     N ODD
C     MASTER PROGRAM
  500 READ(2,4)N,R,Q
      CALL CARIP(R,E)
      READ(2,94)YYY,FO,BB,Z
      A7N=N
      A8N=A7N/2.
      A8I=1.
    5 A9=A8N-A8I
      IF(A9)1,2,3
    3 A8I=A8I+1.
      GO TO 5
    1 CALL ANOD(N,E,YYY,FO,BB,FMAX,FMIN,DF,Q,AP,AZ)
      GO TO 15
    2 CONTINUE
   15 CALL AFIN(AP,AZ,N,Q,Z,FO,YYY,BB)
      WRITE(3,38)N,Q,E
    4 FORMAT(I5,F15.7,F10.0)
   94 FORMAT(4F10.0)
   38 FORMAT(1X,//,5X,3HN =,I5,3X,3HQ =,F15.6,3X,3HE =,E15.6,//)
      GO TO 500
  501 STOP
      END
```

FIGURE A-4.—ML2B program

```
// JOB
// FOR
*IOCS(CARD, 1132 PRINTER, TYPEWRITER, DISK)
*NAME ML3
*ONE WORD INTEGERS
* LIST SOURCE PROGRAM
      DIMENSION APHA(10), BETA(10),AD(10), F(10)
C     MAIN LINE THREE    CAUER SYNTHESIS
  500 READ(2,4)N,R,Q
      CALL CARIP(R,E)
      WRITE(3,999)R
  999 FORMAT(10X,'RIPPLE = ',E15.6)
      READ(2,94)YY,FO,BB,Z
      QC=FO/BB
      YYY=YY-2.5
      IF(YYY)100,100,101
  101 QP=Q/QC
      GO TO 102
  100 QP=Q
  102 WRITE(3,45)
   45 FORMAT(5X,//,5X,'THEORETICAL POLES',/)
      A7N=N
      A8N=A7N/2.
      A8I=1.
    5 A9=A8N-A8I
      IF(A9)1,2,3
    3 A8I=A8I+1.
      GO TO 5
C     N ODD
    1 CALL CAPOO(E,N,APHA,BETA,SPOL)
      ML77=(N-1)/2
      DO 28 I=1,ML77
      WRITE(3,26)I,APHA(I),BETA(I)
   28 CONTINUE
      WRITE(3,30)SPOL
   30 FORMAT(10X,E15.6)
      CALL PRDSO(N,QP,APHA,BETA,SPOL)
      M=(N-1)/2
      K6=7
      K7=0
      K8=1
      K9=0
      CALL XOPOL(APHA,BETA,N,SPOL,AD)
      GO TO 18
C     N EVEN
    2 CALL CAPOE(E,N,APHA,BETA)
      ML88=N/2
      DO 25 I=1,ML88
      WRITE(3,26)I,APHA(I),BETA(I)
   26 FORMAT(5X,I5,5X,E15.6,5X,E15.6)
   25 CONTINUE
      CALL PRDSE(N,QP,APHA,BETA)
      M=N/2
      K6=M
      K7=0
      K8=0
      K9=0
      CALL XEPOL(APHA,BETA,N,AD)
   18 CALL SYCAS(AD,N,F)
      TES=YY-3.
      IF(TES)888,889,888
  889 CALL FITUN(FO,Q,BB,N,F)
  888 WRITE(3,40)
   40 FORMAT(5X,//,5X,'CAUER CIRCUIT ELEMENTS',/)
      CALL FISCA(Z,FO,YYY,BB,F,N)
      WRITE(3,38)N,Q,E
    4 FORMAT(I5,F15.7,F10.0)
   38 FORMAT(1X,//,5X,3HN =,I5,3X,3HQ =,F15.6,3X,3HE =,E15.6,//)
   94 FORMAT(4F10.0)
      GO TO 500
  277 STOP
      END
```

FIGURE A-5.—ML3 program

```
// JOB
// FOR
*IOCS(CARD, 1132 PRINTER, TYPEWRITER, DISK)
* LIST SOURCE PROGRAM
*ONE WORD INTEGERS
C       ML4
C       MAIN LIME 4   TRANSFORMATION CALCULATION
        READ(2,100)F
        READ(2,100)Y
        READ(2,100)Q
        IF(Y)1,1,2
      1 READ(2,101)X,FO,B
        GO TO 3
      2 READ(2,101)X,FH,FL
        FO=SQRT(FH*FL)
        B=FH-FL
      3 WO=FO/(2.*3.14159)
        BW=B/(2.*3.14159)
        W=F/(2.*3.14159)
C       TEST X
        X=X-2.5
        IF(X)13,13,14
     13 X=X+1.
        IF(X)15,15,16
     14 X=X-1.
        IF(X)17,17,18
C
     15 WEQ=W/WO
        XPL=1./WO
        WRITE(3,102)FO
        GO TO 20
C
     16 WEQ=-WO/W
        XPL=WO/W**2
        WRITE(3,103)FO
        GO TO 20
C
     17 WEQ=WO/BW*(W/WO-WO/W)
        XPL=WO/BW*(1./WO+WO/W**2)
        Q=Q*B/FO
        WRITE(3,104)FO,B
        GO TO 20
C
     18 WEQ=-1./(WO/B*(W/WO-WO/W))
        XPL=(1./WO+WO/W**2)/(WO/BW*(W/WO-WO/W)**2)
        Q=Q*B/FO
        WRITE(3,105)FO,B
        GO TO 20
C
     20 WRITE(3,106)WEQ,XPL
        WRITE(3,108)Q
        WRITE(3,107)F
C
    100 FORMAT(F10.0)
    101 FORMAT(3F10.0)
    102 FORMAT(5X,30HLOW PASS FILTER WITH CUTOFF AT,E13.6)
    103 FORMAT(5X,26HHIGH PASS FILTER, CUTON AT,E13.6)
    104 FORMAT(5X,'BAND PASS FILTER, CENTER FREQUENCY',E13.6,3X,'
       1BANDWIDTH'E13.6)
    105 FORMAT(5X,'BAND STOP FILTER, CENTER FREQUENCY ',E13.6,5X,'
       1BANDWIDTH ',E13.6)
    106 FORMAT(5X,//,5X,21HEQUIVALENT FREQUENCY ,E13.6,
       110X,11HMULTIPLIER ,E13.6)
    107 FORMAT(5X,'FREQUENCY OF INTEREST ',5X,E13.6)
    108 FORMAT(2X,//,5X,'Q FACTOR = ',E15.6,/)
        STOP
        END
```

FIGURE A-6.—ML4 program

```
// JOB
// FOR
*IOCS(CARD,TYPEWRITER,1132 PRINTER,DISK)
* LIST SOURCE PROGRAM
*ONE WORD INTEGERS
C       PROGRAM TWO ,  DESIGN CHARTS
C       G     ATTENUATION
C       R     RIPPLE
      1 READ(2,947)G,XMAX
    947 FORMAT(2F10.0)
        XO=1.
        XS=15./(XMAX-XO)
        YS=1./.01
        YO=0.
        XXX=(XMAX-XO)/60.
        CALL SCALF(XS,YS,XO,YO)
        CALL FGRID(1,XO,0.,.005,20)
        CALL FGRID(0,XO,0.,XXX,60)
        EE=2.7183
        G=-G
        X=G
        G=10.**(G/20.)
        DO 20 J=4,12
        N=J
        XN=N
    997 R=0.
        W=(1./G**2-1.)**(1./(2.*XN))
        CALL FPLOT(-2,W,R)
        R=.0005
        DO 651 I=1,10
        CALL CARIP(R,E)
        YY=2./E
        YLN=ALOG(YY)
        AO=(YLN-E**2/4.-3.*E**4/32.-15.*E**6/288.-105.*E**8/3072.)/XN
        CAO=(EE**AO+EE**(-AO))/2.
        E=E*SQRT(G**2/(1.-G**2))
        YY=2./E
        YLN=ALOG(YY)
        AO=(YLN-E**2/4.-3.*E**4/32.-15.*E**6/288.-105.*E**8/3072.)/XN
        CA1=(EE**AO+EE**(-AO))/2.
        W=CA1/CAO
        CALL FPLOT(-2,W,R)
        R=R+.0005
    651 CONTINUE
      2 R=.005
        DO 3 I=1,20
        CALL CARIP(R,E)
        YY=2./E
        YLN=ALOG(YY)
        AO=(YLN-E**2/4.-3.*E**4/32.-15.*E**6/288.-105.*E**8/3072.)/XN
        CAO=(EE**AO+EE**(-AO))/2.
        E=E*SQRT(G**2/(1.-G**2))
        YY=2./E
        YLN=ALOG(YY)
        AO=(YLN-E**2/4.-3.*E**4/32.-15.*E**6/288.-105.*E**8/3072.)/XN
        CA1=(EE**AO+EE**(-AO))/2.
        W=CA1/CAO
        CALL FPLOT(-2,W,R)
        R=R+.005
      3 CONTINUE
        CALL FPLOT(3,XO,0.)
     20 CONTINUE
        GO TO 1
    500 STOP
        END
```

FIGURE A-7.—ML5 program

```
// JOB
// FOR
*IOCS(CARD, 1132 PRINTER, TYPEWRITER, DISK)
* LIST SOURCE PROGRAM
*NAMEACTFS ML 6
*ONE WORD INTEGERS
      DIMENSION APHA(20), BETA(20),A(20),B(20),      G(20),D(20),
     * E(20),W(20),CL(20),BL(20)
    8 READ(2,10) N,R,Q
   10 FORMAT(I5,F15.0,1E10.0)
      IF(N) 600,600,9
C     N = ORDER OF FILTER
C     R = RIPPLE IN DB
C     YY = 1  LOW-PASS  FILTER
C     YY = 2  HIGH-PASS FILTER
C     YY = 3  BAND-PASS FILTER
C     YY = 4  BAND-STOP FILTER
    9 READ(2,11) YY,FO,BW
   11 FORMAT(3E10.0)
      READ(2,12) FMAX,FMIN,DF,SCAL
   12 FORMAT(4E10.0)
      WRITE (3,110)
      WRITE (3,111) N,R,Q,YY,FO,BW,FMAX,FMIN,DF,SCAL
      IF (Q) 14,14,15
   14 Q = 1.0E25
   15 CALL CARIP(R,EN)
      PI = 6.283185
C     TEST FOR N EVEN OR ODD
      IYY = YY
      XN = N
      YN = XN/2.
   20 YN = YN - 1.
      IF (YN) 21,22,20
C     PROGRAM FOR EVEN ORDER FILTERS
   22 CALL CAPOE (EN,N,APHA,BETA)
      CALL PDSEV (N,Q,APHA)
      M = N/2
      MM = M
      DO 30 I=1,M
      BL(I) = -2.*APHA(I)
   30 CL(I) = APHA(I)**2 + BETA(I)**2
      GO TO (40,50,60,70), IYY
C     LOW-PASS FOR N EVEN
   40 DO 42 I=1,M
      B(I) = BL(I)/SQRT(CL(I))
      W(I) = PI*FO*SQRT(CL(I))
      CALL CMPRL(I,B(I),W(I),A(I),G(I),D(I),E(I),Q)
   42 CONTINUE
      IF(YN) 82,90,90
C     HIGH-PASS FOR N EVEN
   50 DO 52 I=1,M
      B(I) = BL(I)/SQRT(CL(I))
      W(I) = PI*FO/SQRT(CL(I))
      CALL CMPRH(I,B(I),W(I),A(I),G(I),D(I),E(I),Q)
   52 CONTINUE
      IF(YN) 82,90,90
C     BAND-PASS FOR N EVEN
   60 WRITE(3,101)
      MM = 2*M
      I = 0
   62 I = I + 1
      CALL BPBCW(BL(I),CL(I),FO,BW,B1,B2,W1,W2)
      B(I) = B1
      W(I) = W1
      CALL CMPRL(I,B(I),W(I),A(I),G(I),D(I),E(I),Q)
      IF(I-M) 62,63,63
   63 I = 0
   64 I = I + 1
      CALL BPBCW(BL(I),CL(I),FO,BW,B1,B2,W1,W2)
      I = I + M
      B(I) = B2
      W(I) = W2
      CALL CMPRH(I,B(I),W(I),A(I),G(I),D(I),E(I),Q)
      I = I - M
      IF(I-M) 64,65,65
   65 IF(DF) 67,67,66
   66 CALL CARSP(MM,A,B,G,D,E,W,FMIN,FMAX,DF,SCAL)
   67 WRITE(3,102)
      I = 0
   68 I = I + 1
      CALL BPBCW(BL(I),CL(I),FO,BW,B1,B2,W1,W2)
      B(I) = B2
      W(I) = W2
      CALL CMPRL(I,B(I),W(I),A(I),G(I),D(I),E(I),Q)
      IF(I-M) 68,69,69
   69 I = 0
   71 I = I + 1
      CALL BPBCW(BL(I),CL(I),FO,BW,B1,B2,W1,W2)
      I  =I + M
      B(I) = B1
      W(I) = W1
      CALL CMPRH(I,B(I),W(I),A(I),G(I),D(I),E(I),Q)
      I = I - M
      IF(I-M) 71,90,90
C     BAND-STOP FOR N EVEN AND ODD
   70 WRITE(3,103)
      GO TO 500
C     PROGRAM FOR ODD ORDER FILTERS
   21 CALL CAPOO(EN,N,APHA,BETA,SPOL)
      CALL PDSOD(N,Q,APHA,SPOL)
      M = (N-1)/2
      IF (M) 82,82,81
   81 DO 80 I=1,M
      BL(I) = -2.*APHA(I)
   80 CL(I) = APHA(I)**2 + BETA(I)**2
      GO TO (40,50,160,70), IYY
   82 I = M + 1
      B(I) = -1./SPOL
      W(I) = PI*FO
      MM = M + 1
      GO TO (140,150,160,70),IYY
C     LOW-PASS FOR N ODD
  140 CALL CPFRL(B(I),W(I),A(I),G(I),D(I),E(I),Q,I)
      GO TO 90
C     HIGH-PASS FOR N ODD
  150 CALL CPFRH(B(I),W(I),A(I),G(I),D(I),E(I),Q,I)
      GO TO 90
C     BAND-PASS FOR N ODD
  160 WRITE (3,104)
      GO TO 500
   90 IF(DF) 500,500,92
   92 CALL CARSP(MM,A,B,G,D,E,W,FMIN,FMAX,DF,SCAL)
  101 FORMAT (1H0,19X,'BAND-PASS FILTER NETWORKS ( HIGH GAIN )'/)
  102 FORMAT(1H0,//1H0,19X,'BAND-PASS FILTER NETWORKS ( LOW GAIN )'/)
  103 FORMAT(1H0,3X,'BAND-STOP FILTER PROGRAM NOT YET AVAILABLE')
  104 FORMAT(1H0,3X,'BAND-PASS FILTER PROGRAM NOT YET AVAILABLE FOR ODD
     1ORDER FILTERS'/4X,'CHANGE TO EVEN ORDER FILTER (N=EVEN NO.) AND
     2RELOAD PROGRAM')
  110 FORMAT(1H1,33X,'ACTIVE FILTER DESIGN PROGRAM'////14X,'INPUT DATA')
  111 FORMAT (1H0,9X,'N =',I4,13X,'R(DB) =',E12.5,9X,'Q =',E12.5
     1  /9X,'YY =',E12.5,8X,'FO =',E12.5,8X,'BW =',E12.5
     2  /7X,'FMAX =',E12.5,6X,'FMIN =',E12.5,8X,'DF =',E12.5,
     3  6X,'SCAL =',E12.5///)
  500 GO TO 8
  600 STOP
      END

// XEQ          2
*LOCAL,CARIP,CAPOE,CAPOO,CARSP,CMPRL,CMPRH,CPFRL,CPFRH,BPBCW,
*LOCALPDSEV,PDSOD
```

FIGURE A-8.—ML6 program

```
// JOB
// FOR
*IOCS(CARD,1132 PRINTER,DISK,PLOTTER)
*LIST SOURCE PROGRAM
*NAMEANWKS ML 7
      DIMENSION A(20),B(20),G(20),D(20),E(20),W(20),      X(8),TITLE(12)
      READ(2,80) (TITLE(I),I=1,20)
   80 FORMAT(20A4)
      WRITE(3,90) (TITLE(I),I=1,20)
   90 FORMAT(1H1,33X,'ACTIVE FILTER SYNTHESIS PROGRAM'///10X,20A4//)
    7 I = 0
    9 READ(2,81) M,FMAX,FMIN,DF,SCAL
   81 FORMAT(I2,4E10.0)
      IF (M) 99,99,8
    8 M = M - 10
      GO TO (10,20,30,40,50),M
   10 CALL CAFIL(I,X,A,B,G,D,E,W)
      GO TO 9
   20 CALL CASEL(I,X,A,B,G,D,E,W)
      GO TO 9
   30 CALL CAFIH(I,X,A,B,G,D,E,W)
      GO TO 9
   40 CALL CASEH(I,X,A,B,G,D,E,W)
      GO TO 9
   50 CALL CARSP(I,A,B,G,D,E,W,FMIN,FMAX,DF,SCAL)
      GO TO 7
   99 STOP
      END


// XEQ          1
*LOCAL,CASEL,CASEH,CAFIL,CAFIH
```

FIGURE A-9.—ML7 program

```
ML8
// JOB
// FOR
*IOCS(CARD,TYPEWRITER,1132 PRINTER,DISK)
*ONE WORD INTEGERS
* LIST SOURCE PROGRAM
      DIMENSION A(20),B(20),G(20),D(20),E(20),W(20)
      CALL TEMP (MM,A,B,G,D,E,W)
      READ (2,10) FMAX,FMIN,DF,SCAL
   10 FORMAT (4E10.0)
      CALL CARSP (MM,A,B,G,D,E,W,FMIN,FMAX,DF,SCAL)
      STOP
      END
```

FIGURE A-10.—ML8 program

# APPENDIX B

# Computer Subroutines

Figures B–1 through B–39 are the subroutines called by the main line programs given in Appendix A. The subroutines, which are written in FORTAN IV, are listed below.

ANOD—Subdivision of ML2B
FISDA—Filter Scale Darlington
FITUN—Filter Tuning
ATRAN—Transient
SYCAS—Synthetic Division Cauer Synthesis
CARES—Calculate Response
FISCA—Filter Scale
AATAN—Arc Tangent
CAPOE—Calculated Poles Even
PRDSO—Predistort Odd
ADPOL—Add Polynominal
PRDSE—Predistort Even
SOPOL—Subdivision of ATRAN
CAKI—Subdivision of ATRAN
CAKO—Subdivision of ATRAN
TAKE—Subdivision ATRAN
CAZEE—Calculate Zeros Even
CAPOO—Calculate Poles Odd
ANEV—Subdivision of ML2A
CARIP—Calculate Ripple
CAZEO—Calculate Zeros Odd
SEPOL—Subdivision of ATRAN
AFIN—Subdivision of ML2
XEPOL—Multiply Even Polynomial
SYDIV—Synthetic Division
XOPOL—Multiply Odd Polynomial
CMPRL—Compute R Second-Order Lowpass
CMPRH—Compute R Second-Order Highpass
CPFRL—Compute R First-Order Lowpass
CPFRH—Compute R First-Order Highpass
BPBCW—Compute bandpass B, C, and $\omega$
CARSP—Calculate Frequency Response
PDSEV—Predistort Even
PDSOD—Predistort Odd
CASEL—Calculate Second-Order Lowpass
CASEH—Calculate Second-Order Highpass
CAFIL—Calculate First-Order Lowpass
CAFIH—Calculate First-Order Highpass
TEMP—Calculate Temperature Response

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE ANOD(N,E,YYY,FO,BB,FMAX,FMIN,DF,Q,AP,AZ)
      DIMENSION APHA(10),BETA(10),AHA(10),BTA(10),AP(10),AZ(10)
      CALL CAPOO(E,N,APHA,BETA,SPOL)
      WRITE(3,90)
   90 FORMAT(5X,5HN ODD,//)
      M1=(N-1)/2
      WRITE(3,6)
    6 FORMAT(1X,//,5X,17HTHEORETICAL POLES)
      WRITE(3,8)SPOL
    8 FORMAT(3X,6HSPOL =,E15.6)
      DO 100 I=1,M1
      WRITE(3,7)I,APHA(I),BETA(I)
    7 FORMAT(3X,3HI =,I3,3X,6HAPHA =,E15.6,3X,6HBETA =,E15.6)
  100 CONTINUE
      CALL CAZEO(N,AHA,BTA,SOL,E)
      WRITE(3,10)
   10 FORMAT(1X,//,5X,18HTHEORETICAL ZEROES)
      WRITE(3,9)SOL
    9 FORMAT(3X,5HSOL =,E15.6)
      DO 200 I=1,M1
      WRITE(3,11)I,AHA(I),BTA(I)
   11 FORMAT(3X,3HI =,I3,3X,5HAHA =,E15.6,3X,5HBTA =,E15.6)
  200 CONTINUE
      CALL PRDSO(N,Q,APHA,BETA,SPOL)
      WRITE(3,12)
   12 FORMAT(3X,//,5X,18HPREDISTORTED POLES)
      WRITE(3,8)SPOL
      DO 201 I=1,M1
      WRITE(3,7)I,APHA(I),BETA(I)
  201 CONTINUE
      CALL PRDSO(N,Q,AHA,BTA,SOL)
      WRITE(3,16)
   16 FORMAT(1X,//,5X,19HPREDISTORTED ZEROES)
      WRITE(3,9)SOL
      DO 300 I=1,M1
      WRITE(3,11)I,AHA(I),BTA(I)
  300 CONTINUE
      CALL XOPOL(APHA,BETA,N,SPOL,AP)
      WRITE(3,20)
   20 FORMAT(3X,//,5X,22HDENOMINATOR POLYNOMIAL)
      DO 400 I=1,N
      WRITE(3,21)I,AP(I)
   21 FORMAT(3X,3HAP(,I2,3H) =,3X,E15.6)
  400 CONTINUE
      CALL XOPOL(AHA,BTA,N,SOL,AZ)
      WRITE(3,22)
   22 FORMAT(3X,//,5X,20HNUMERATOR POLYNOMIAL)
      DO 501 I=1,N
      WRITE(3,23)I,AZ(I)
   23 FORMAT(3X,3HAZ(,I2,3H) =,3X,E15.6)
  501 CONTINUE
      RETURN
      END
```

FIGURE B–1.—Subroutine ANOD

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE FISDA(Z,FO,YYY,BB,F,N)
      DIMENSION F(10)
      BBB=BB*2.*3.14159
      WO=FO*2.*3.14159
      Y=YYY
      Y=Y-2.5
      IF(Y)3,3,4
    3 Y=Y+1.
      IF(Y)5,5,6
    4 Y=Y-1.
      IF(Y)7,7,8
    5 GO TO 7
    6 GO TO 8
    7 ZS=Z
      GO TO 10
    8 ZS=1./Z
   10 A7N=N
      A8N=A7N/2.
      A8I=1.
   11 A9=A8N-A8I
      IF(A9)21,22,23
   23 A8I=A8I+1.
      GO TO 11
C
C     N EVEN
   22 M1=N-1
      DO 31 I=1,M1,2
      F(I)=F(I)*ZS/BBB
   31 CONTINUE
      DO 32 I=2,N,2
      F(I)=F(I)/(ZS*BBB)
   32 CONTINUE
      F(N+1)=F(N+1)*Z
      GO TO 100
C
C     N ODD
   21 DO 41 I=1,N,2
      F(I)=F(I)*ZS/BBB
   41 CONTINUE
      M1=N-1
      DO 42 I=2,M1,2
      F(I)=F(I)/(ZS*BBB)
   42 CONTINUE
      F(N+1)=F(N+1)*Z
  100 M3=N+1
      DO 101 I=1,M3
      WRITE(3,150)I,F(I)
  101 CONTINUE
      WRITE(3,151)Z
  150 FORMAT(5X,2HI=,I2,8X,2HF=,2X,E15.6)
  151 FORMAT(5X,2HZ=,2X,E15.6)
      RETURN
      END
```

FIGURE B-2.—Subroutine FISDA

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE FITUN(FO,Q,BB,N,F)
      DIMENSION F(10), TR3(10), TR2(10), DF2(10), DF3(10)
      BW1=FO/Q
      WRITE(3,100)BW1
  100 FORMAT(5X,/,5X,'BRANCH BANDWIDTH',5X,E14.6,/)
      WRITE(3,101)BB,FO
  101 FORMAT(5X,17HFILTER BANDWIDTH ,E14.6,//,5X,
     117HCENTER FREQUENCY ,5X,E14.6,//)
      M=N-1
C     TEST N EVEN OR ODD
      XN=N
      XN2=XN/2.
    5 IF(XN2)1,2,3
    3 XN2=XN2-1.
      GO TO 5
C     N EVEN
    2 M3=N/2
      TR3(1)=1.
      TR2(1)=SQRT(F(1)*F(2))*FO/BB
      XC=TR3(1)/TR2(1)
      DO 300 I=2,M3
      J=2*I-1
      TR3(I)=SQRT(F(J)/F(1))/XC
      TR2(I)=SQRT(F(J+1)*F(1))*XC*FO/BB*TR3(I)
      XC=XC*TR3(I)/TR2(I)
  300 CONTINUE
      GO TO 407
C
C     N ODD
    1 M3=(N-1)/2
      XC=1.
      DO 302 I=1,M3
      J=2*I
      TR3(I)=SQRT(F(1)*F(J))*XC*FO/BB
      TR2(I)=SQRT(F(J+1)/F(1))*TR3(I)/XC
      XC=XC*TR2(I)/TR3(I)
  302 CONTINUE
  407 WRITE(3,60)
   60 FORMAT(5X,//,5X,'COIL         TAP 2 RATIO          TAP 3 RATIO',/)
  400 DO 306 I=1,M3
      WRITE(3,301)I,TR2(I),TR3(I)
  306 CONTINUE
      WRITE(3,61)
   61 FORMAT(1X,///,6X,4HCOIL,8X,11H FC. DEF. 2,20X,11H FC. DEF. 3,/)
      DO 308 I=1,M3
      DF2(I)=SQRT(FO**2/TR2(I)**2-BW1**2)
      DF3(I)=SQRT(FO**2/TR3(I)**2-BW1**2)
      WRITE(3,301)I,DF2(I),DF3(I)
  308 CONTINUE
      FL=1.
      DO 947 I=1,N
      X=1.
      DO 948 J=1,I
      M17=N+1-J
  948 X=X*F(M17)/Q
  947 FL=FL+X
      WRITE(3,310)FL
  310 FORMAT(1X,//,5X,11HFLAT LOSS =,2X,E15.6)
  301 FORMAT(7X,I2,10X,E15.6,12X,E15.6)
      RETURN
      END
```

FIGURE B-3.—Subroutine FITUN

```
// JOB
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE ATRAN(N,W,WO,EE,TMAX,TMIN,DT)
      DIMENSION APHA(10),BETA(10),A(10),AK(10),PK(10)
      E=2.7183
      XN=N
      XI=I
    4 TX=(XN-XI)
      IF(TX)1,2,3
    3 XI=XI+1.
      GO TO 4
C
C     N EVEN
    2 CALL CAPOE(EE,N,APHA,BETA)
      CALL XEPOL(APHA,BETA,N,A)
      CALL CAKO(N,A,AKO,PKO,WO,W)
      M=N/2
      DO 10 J=1,M
      CALL SEPOL(APHA,BETA,N,J,A)
      CALL CAKI(APHA(J),BETA(J),N,W,WO,AK(J),PK(J),A)
   10 CONTINUE
      SPOL=0.
      FAC=0.
      AK(M+1)=0.
      GO TO 100
C     N ODD
    1 CALL CAPOO(EE,N,APHA,BETA,SPOL)
      CALL XOPOL(APHA,BETA,N,SPOL,A)
      CALL CAKO(N,A,AKO,PKO,WO,W)
      M=(N-1)/2
      DO 20 J=1,M
      CALL SOPOL(APHA,BETA,SPOL,M,J,A)
      CALL CAKI(APHA(J),BETA(J),N,W,WO,AK(J),PK(J),A)
   20 CONTINUE
C
      CALL SOPOL(APHA,BETA,SPOL,N,M,A)
      N3=N-2
      AT=0.
      RT=0.
      DO 50 I=1,N3
      K=N-1-I
      CALL TAKE(SPOL,0.,K,RO,AO)
      RT=RT+RO*A(I)
      AT=AT+AO*A(I)
   50 CONTINUE
      CALL TAKE(SPOL,0,N-1,RO,AO)
      RT=RT+RO
      AT=AT+AO
      RT=RT+A(N-1)
      ZT1=SQRT(AT**2+RT**2)
      CALL AATAN(AT,RT,PT1)
      RAD=W**2 +SPOL**2
      AMAD=0.
      ZAD=SQRT(RAD**2+AMAD**2)
      CALL AATAN(AMAD,RAD,PAD)
      ZT=ZAD*ZT1
      PT=PT1+PAD
      M7=(N+1)/2
      AK(M7)=WO*W/ZT
      PK(M7)=PT
      FAC=1.
      M=(N-1)/2
      GO TO 100
C
  100 T=TMIN*WO/(2.*3.14159)
  101 FT=FAC*AK(M+1)*E**(SPOL*T)
      FT=AKO*(WO/W)*SIN(W/WO*T+PKO)+FT
      DO 200 I=1,M
      FT=FT+AK(I)/BETA(I)*SIN(BETA(I)*T+PK(I))*E**(APHA(I)*T)
  200 CONTINUE
      TT=T*2.*3.14159/WO
      WRITE(3,205)TT,FT
      T=T+DT*WO/(2.*3.14159)
      TTT=TMAX-TT
      IF(TTT)102,101,101
  205 FORMAT(5X,2HT=,2X,E15.6,5X,3HFT=,2X,E15.6)
  102 RETURN
      END
```

FIGURE B-4.—Subroutine ATRAN

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE SYCAS(AD,N,F)
      DIMENSION AD(8),A(8),B(8),R(8),F(8)
C     PROGRAM FOR SYNTHETIC DIVISION
C     CAUER SYNTHESIS
    1 DO 2 I=1,8
      A(I)=0.
      B(I)=0.
    2 CONTINUE
      A7N=N
      A8N=A7N/2.
      A8I=1.
    6 A9=A8N-A8I
      IF(A9)3,4,5
    5 A8I=A8I+1.
      GO TO 6
C     N ODD
    3 J=1
      WRITE(3,100)
  100 FORMAT(5X,5HN ODD)
      M1=(N+1)/2
      DO 7 I=1,M1
      B(I)=AD(J)
      J=J+2
    7 CONTINUE
      J=2
      A(1)=1.
      DO 8 I=2,M1
      A(I)=AD(J)
      J=J+2
    8 CONTINUE
C
      J=1
C
      M2=(N-1)/2+1-J
      M3=(N+1)/2+1-J
C
    9 F(J)=A(1)/B(1)
      DO 10 I=1,M2
      R(I)=A(I+1)-B(I+1)*A(1)/B(1)
   10 CONTINUE
C
C     RESET
      DO 11 I=1,M3
      A(I)=B(I)
   11 CONTINUE
      DO 12 I=1,M2
      B(I)=R(I)
   12 CONTINUE
      GO TO 13
C
C     N EVEN
    4 J=1
      WRITE(3,101)
  101 FORMAT(5X,6HN EVEN)
      M5=N/2
      DO 14 I=1,M5
      B(I)=AD(J)
      J=J+2
   14 CONTINUE
      J=2
      A(1)=1.
      M6=N/2+1
      DO 15 I=2,M6
      A(I)=AD(J)
      J=J+2
   15 CONTINUE
C
      J=1
C
      M7=N/2-J
      M8=N/2+1-J
      M9=N/2+2-J
C
   16 F(J)=A(1)/B(1)
      DO 17 I=1,M7
      R(I)=A(I+1)-B(I+1)*A(1)/B(1)
   17 CONTINUE
      R(M8)=A(M9)
C     RESET
      DO 18 I=1,M8
      A(I)=B(I)
      B(I)=R(I)
   18 CONTINUE
C
      J=J+1
      M3=M8
      M2=M3-1
      GO TO 9
C
   13 M10=N-J
      IF(M10)19,19,20
C
   20 J=J+1
      M8=M2
      M9=M3
      M7=M8-1
      GO TO 16
   19 RETURN
      END
```

FIGURE B-5.—Subroutine SYCAS

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE  CARES(YYY,FO,BB,FMAX,FMIN,DF,K,L,M,N,
     1ALPH1,BETA1,ALPH2,BETA2,SPOL1,SPOL2)
      DIMENSION ALPHA(10),BETA(10),B(10),C(10),T(10),X(10),Y(10)
     1,P(10),D(10),PW(10),ALPH1(10),BETA1(10),ALPH2(10),BETA2(10),
     2ALPH3(10),ALPH4(10)
      ALPH3(1)=SPOL1
      ALPH4(1)=SPOL2
C     X=1. LOW PASS           FO  SCALE FREQUENCY
C     X=2.  HIGH PASS               B  BANDWIDTH
C     X=3.  BAND PASS
C     X=4.  BAND STOP
C     K= NUMBER OF CONJUGATE POLE PAIRS
C     L= NUMBER OF CONJUGATE ZERO PAIRS
C     M= NUMBER OF SINGLE POLES
C     N= NUMBER OF SINGLE ZEROES
C
   80 YY=YYY
      FC=FMIN
      F=FC
   18 TT=1.
      PT=0.
      PWT=0.
      DT=0.
      YY=YY-2.5
      IF(YY)3,3,4
    3 YY=YY+1.
      IF(YY)5,5,6
    4 YY=YY-1.
      IF(YY)7,7,8
    5 FC=F/FO
      DD=1./FO
      GO TO 9
    6 FC=-FO/F
      DD=FO/(FC**2)
      GO TO 9
    7 FC=FO/BB*(F/FO-FO/F)
      DD=FO/BB*(1./FO+FO/(F**2))
      GO TO 9
    8 FC=FO/(FO/BB*(F/FO-FO/F))
      DD=-BB*(1./FO+FO/(F**2))/(F/FO-FO/F)
      GO TO 9
    9 W=FC
      IF(K)1050,1050,1051
 1051 DO 101 I=1,K
      ALPHA(I)=ALPH1(I)
      BETA(I)=BETA1(I)
      B(I)=-2.*ALPHA(I)
      C(I)=ALPHA(I)**2+BETA(I)**2
      T(I)=1./SQRT((C(I)-W**2)**2+(W*B(I))**2)
      X(I)=W*B(I)
      Y(I)=C(I)-W**2
      CALL AATAN(X(I),Y(I),Z)
      P(I)=-Z
      D(I)=1./(1.+(W*B(I)/(C(I)-W**2))**2)*((C(I)-W**2)*B(I)+
     1W**2*B(I)*2.)/((C(I)-W**2)**2)
      D(I)=D(I)*DD
      PW(I)=P(I)/W
      TT=TT*T(I)
      PT=PT+P(I)
      PTW=PT/W
      DT=DT+D(I)
  101 CONTINUE
 1050 IF(L)1060,1060,1061
 1061 DO 102 I=1,L
      ALPHA(I)=ALPH2(I)
      BETA(I)=BETA2(I)
      T(I)=SQRT((C(I)-W**2)**2+(W*B(I))**2)
      X(I)=W*B(I)
      Y(I)=C(I)-W**2
      CALL AATAN(X(I),Y(I),Z)
      P(I)=Z
      D(I)=1./(1.+(W*B(I)/(C(I)-W**2))**2)*((C(I)-W**2)*B(I)+
     1W**2*B(I)*2.)/((C(I)-W**2)**2)
      D(I)=D(I)*DD
      D(I)=-D(I)
      PW(I)=P(I)/W
      TT=TT*T(I)
      PT=PT+P(I)
      PTW=PT/W
      DT=DT+D(I)
  102 CONTINUE
 1060 IF(M)1070,1070,1071
 1071 DO 103 I=1,M
      ALPHA(I)=ALPH3(I)
      T(I)=1./SQRT(W**2+ALPHA(I)**2)
      XX=W
      Y(I)=-ALPHA(I)
      CALL AATAN(XX,Y(I),Z)
      P(I)=-Z
      D(I)=-1./(ALPHA(I))*1./(1.+(W/ALPHA(I))**2)
      D(I)=D(I)*DD
      PW(I)=P(I)/W
      TT=TT*T(I)
      PT=PT+P(I)
      DT=DT+D(I)
  103 CONTINUE
 1070 IF(N)1080,1080,1081
 1081 DO 104 I=1,N
      ALPHA(I)=ALPH4(I)
      T(I)=SQRT(W**2+ALPHA(I)**2)
      XX=W
      Y(I)=-ALPHA(I)
      CALL AATAN(XX,Y(I),Z)
      P(I)=Z
      D(I)=1./(ALPHA(I))*1./(1.+(W/ALPHA(I))**2)
      D(I)=D(I)*DD
      PW(I)=P(I)/W
      TT=TT*T(I)
      PT=PT+P(I)
      DT=DT+D(I)
  104 CONTINUE
 1080 PT=PT*180./3.14159
      PWT=PT/W
      DT=DT/(2.*3.14159)
      WRITE(3,16)F,TT,PT,PWT,DT
   16 FORMAT(2X,4HFC =,E12.5,2X,5HAMP =,E12.5,2X,7HPHASE =,E12.5,
     12X,9HPHASE/W =,E12.5,2X,7HDELAY =,E12.5)
      F=F+DF
      AXY=FMAX-F
      YY=YYY
      IF(AXY)17,17,18
   17 RETURN
      END
```

FIGURE B–6.—Subroutine CARES

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE FISCA(Z,FO,YYY,BB,F,N)
      DIMENSION F(10)
      BBB=BB*2.*3.14159
      WO=FO*(2.*3.14159)
      Y=YYY
      Y=Y-2.5
      IF(Y)3,3,4
    3 Y=Y+1.
      IF(Y)5,5,6
    4 Y=Y-1.
      IF(Y)7,7,8
    5 GO TO 7
    6 GO TO 8
    7 ZS=Z
      GO TO 10
    8 ZS=1./Z
   10 A7N=N
      A8N=A7N/2.
      A8I=1.
   11 A9=A8N-A8I
      IF(A9)21,22,23
   23 A8I=A8I+1.
      GO TO 11
C
C     N EVEN
   22 M1=N-1
      DO 31 I=1,M1,2
      F(I)=F(I)/(ZS*BBB)
   31 CONTINUE
      DO 32 I=2,N,2
      F(I)=F(I)*ZS/BBB
   32 CONTINUE
      GO TO 100
C
C     N ODD
   21 DO 41 I=1,N,2
      F(I)=F(I)*ZS/BBB
   41 CONTINUE
      M1=N-1
      DO 42 I=2,M1,2
      F(I)=F(I)/(ZS*BBB)
   42 CONTINUE
  100 DO 101 I=1,N
      WRITE(3,150)I,F(I)
  101 CONTINUE
      WRITE(3,151)Z
  150 FORMAT(5X,2HI=,I2,8X,2HF=,2X,E15.6)
  151 FORMAT(5X,'TERMINAL RESISTANCE = ',E15.6)
      RETURN
      END
```

FIGURE B–7.—Subroutine FISCA

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE AATAN(X,Y,Z)
      PHASE=0.
      P=X/Y
      IF(X)1,2,2
    1 IF(Y)3,9,4
    2 IF(Y)5,10,6
    3 P=P
      PHASE=3.14159+ATAN(P)
      GO TO 7
    4 P=-P
      PHASE=-ATAN(P)
      GO TO 7
    5 P=-P
      PHASE=3.14159-ATAN(P)
      GO TO 7
    6 P=P
      PHASE=ATAN(P)
    7 Z=PHASE
      GO TO 11
    9 Z=-3.14159/2.
      GO TO 11
   10 Z=3.14159/2.
   11 RETURN
      END
```

FIGURE B–8.—Subroutine AATAN

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE  CAPOE(E,N,APHA,BETA)
      DIMENSION AN(20),APHA(20),BETA(20)
C     CALCULATE POLES,N EVEM
    1 XN=N
      IF(E)2,3,2
    2 YY=2./E
      YLN=ALOG(YY)
      A=(YLN+E**2/4.-3.*E**4/32.+15.*E**6/288.-105.*E**8/3072.)/XN
      AA=(YLN-E**2/4.-3.*E**4/32.-15.*E**6/288.-105.*E**8/3072.)/XN
      EE=2.7183
      SA1=(EE**A-EE**(-A))/2.
      CA1=(EE**A+EE**(-A))/2.
      CA2=(EE**AA+EE**(-AA))/2.
      GO TO 4
    3 SA1=1.
      CA1=1.
      CA2=1.
    4 M=N/2
      DO 10 I=1,M
      A13I=I
      A13N=N
      AN(I)=(2.*A13I-1.)/A13N
      APHA(I)=-SA1*SIN(AN(I)*3.14159/2.)/CA2
      BETA(I)=CA1*COS(AN(I)*3.14159/2.)/CA2
   10 CONTINUE
      RETURN
      END
```

FIGURE B–9.—Subroutine CAPOE

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE PRDSO(N,Q,ALPH,BETA,SPOL)
      DIMENSION ALPH(20),BETA(20)
C
C
    1 M=(N-1)/2
      DO 10 I=1,M
      ALPH(I)=ALPH(I)+1./Q
      BETA(I)=BETA(I)
   10 CONTINUE
      SPOL=SPOL+1./Q
      RETURN
      END
```

FIGURE B–10.—Subroutine PRDSO

```
// JOB
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE ADPOL(AP,AZ,N,AN,AD)
      DIMENSION AP(20),AZ(20),AN(20),AD(20)
      DO 10 I=1,N
      AD(I)=AP(I)-AZ(I)
      AN(I)=AP(I)+AZ(I)
   10 CONTINUE
      RETURN
      END
```

FIGURE B–11.—Subroutine ADPOL

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE PRDSE(N,Q,ALPA,BETA)
      DIMENSION ALPA(20),BETA(20)
C     SUBROUTINE TO COMPENSATE FOR DISSIPATION
C                   BY    PREDISTORTION    N EVEN
    1 M=N/2
      DO 10 I=1,M
      ALPA(I)=ALPA(I)+1./Q
      BETA(I)=BETA(I)
   10 CONTINUE
      RETURN
      END
```

FIGURE B–12.—Subroutine PRDSE

```
// JOB
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE SOPOL(APHA,BETA,SPOL,N,J,A)
      DIMENSION APHA(10),BETA(10),A(10)
      M=(N-1)/2
      IT=J-M
      IF(IT)10,10,11
   10 MM=M-1
      DO 20 I=1,MM
      IF(I-J)22,22,21
   21 APHA(I)=APHA(I)
      BETA(I)=BETA(I)
      GO TO 20
   22 APHA(I)=APHA(I+1)
      BETA(I)=BETA(I+1)
   20 CONTINUE
      NN=N-2
      CALL XOPOL(APHA,BETA,NN,SPOL,A)
      GO TO 30
   11 NNN=N-1
      CALL XEPOL(APHA,BETA,NNN,A)
   30 RETURN
      END
```

Figure B-13.—Subroutine SOPOL

```
// JOB
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE CAKI(AP,BP,N,W,WO,AK,PK,A)
      DIMENSION A(10)
      M=N-2-1
      RT=0.
      AT=0.
      DO 20 I=1,M
      K=N-2-I
      CALL TAKE(AP,BP,K,RO,AO)
      RT=RT+RO*A(I)
      AT=AT+AO*A(I)
   20 CONTINUE
      M3=N-2
      CALL TAKE(AP,BP,M3,RO,AO)
      RT=RT+RO
      AT=AT+AO
      RT=RT+A(N-2)
      ZT1=SQRT(AT**2+RT**2)
      CALL AATAN(AT,RT,PT1)
      RAD=(W/WO)**2+AP**2+BP**2
      AMAD=2.*AP*BP
      ZAD=SQRT(RAD**2+AMAD**2)
      CALL AATAN(AMAD,RAD,PAD)
      ZT=ZAD*ZT1
      PT=PAD+PT1
      AK=(W/WO)*1./ZT
      PK=-PT
      RETURN
      END
```

Figure B-14.—Subroutine CAKI

```
// JOB
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE CAKO(N,A,AKO,PKO,WO,W)
      DIMENSION A(10)
      Z=W/WO
      NN=N-1
      RT=0.
      AT=0.
      DO 10 I=1,NN
      K=N-I
      CALL TAKE(0.,Z,K,RO,AO)
      RT=RT+RO*A(I)
      AT=AT+AO*A(I)
   10 CONTINUE
      CALL TAKE(0.,Z,N,RO,AO)
      RT=RT+RO
      AT=AT+AO
      RT=RT+A(N)
      ZT=SQRT(AT**2+RT**2)
      CALL AATAN(AT,RT,PKO)
      AKO=W/(WO*ZT)
      PKO=-PKO
      RETURN
      END
```

Figure B-15.—Subroutine CAKO

```
// JOB
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE TAKE(RE,AM,N,RO,AO)
      IF(RE)1,2,1
    2 Z=SQRT(AM**2)
      AN=3.14159/2.
      GO TO 3
    1 Z=SQRT(RE**2+AM**2)
      CALL AATAN(AM,RE,AN)
    3 ZN=Z**N
      XN=N
      ANN=XN*AN
      RO=ZN*COS(ANN)
      AO=ZN*SIN(ANN)
      RETURN
      END
```

Figure B-16.—Subroutine TAKE

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE CAZEE(N,ALPA,BETA,E)
      DIMENSION ALPA(20),BETA(20),AN(20)
C     CALCULATE ZEROS  N EVEN
      XN=N
      IF(E)2,3,2
    2 YY=2./E
      YLN=ALOG(YY)
      AA=(YLN-E**2/4.-3.*E**4/32.-15.*E**6/288.-105.*E**8/3072.)/XN
      EE=2.7183
      CA2=(EE**AA+EE**(-AA))/2.
      GO TO 4
    3 CA2=1.0E+15
    4 M=N/2
      DO 10 I=1,M
      A13I=I
      A13N=N
      AN(I)=(2.*A13I-1.)/A13N
      ALPA(I)=0.
      BETA(I)=(COS(AN(I)*3.14159/2.))/CA2
   10 CONTINUE
      RETURN
      END
```

Figure B-17.—Subroutine CAZEE

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE  CAPOO(E,N,APHA,BETA,SPOL)
      DIMENSION AN(20),APHA(20),BETA(20)
C     CALCULATE POLES N ODD
    1 XN=N
      IF(E)2,3,2
    2 YY=2./E
      YLN=ALOG(YY)
      A=(YLN+E**2/4.-3.*E**4/32.+15.*E**6/288.-105.*E**8/3072.)/XN
      AA=(YLN-E**2/4.-3.*E**4/32.-15.*E**6/288.-105.*E**8/3072.)/XN
      EE=2.7183
      SA1=(EE**A-EE**(-A))/2.
      CA1=(EE**A+EE**(-A))/2.
      CA2=(EE**AA+EE**(-AA))/2.
      GO TO 4
    3 SA1=1.
      CA1=1.
      CA2=1.
    4 M=(N-1)/2
      DO 10 I=1,M
      A13I=I
      A13N=N
      AN(I)=(2.*A13I-1.)/A13N
      APHA(I)=-SA1*SIN(AN(I)*3.14159/2.)/CA2
      BETA(I)=CA1*COS(AN(I)*3.14159/2.)/CA2
   10 CONTINUE
      SPOL=-SA1/CA2
      RETURN
      END
```

Figure B-18.—Subroutine CAPOO

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE ANEV(N,E,YYY,FO,BB,FMAX,FMIN,DF,Q,AP,AZ)
      DIMENSION APHA(10),BETA(10),AHA(10),BTA(10),AP(10),AZ(10)
      CALL CAPOE(E,N,APHA,BETA)
      M4=N/2
      WRITE(3,91)
   91 FORMAT(5X,6HN EVEN,//)
      WRITE(3,6)
      DO 700 I = I, M4
      WRITE(3,7)I,APHA(I),BETA(I)
  700 CONTINUE
      CALL CAZEE(N,AHA,BTA,E)
      WRITE(3,10)
      DO 800 I=1,M4
      WRITE(3,11)I,AHA(I),BTA(I)
  800 CONTINUE
      CALL PRDSE(N,Q,APHA,BETA)
      WRITE(3,12)
      DO 900 I=1,M4
      WRITE(3,7)I,APHA(I),BETA(I)
  900 CONTINUE
      CALL PRDSE(N,Q,AHA,BTA)
      WRITE(3,16)
      DO 950 I=1,M4
      WRITE(3,11)I,AHA(I),BTA(I)
  950 CONTINUE
      CALL XEPOL(APHA,BETA,N,AP)
      WRITE(3,20)
      DO 961 I=1,N
      WRITE(3,21)I,AP(I)
  961 CONTINUE
      CALL XEPOL(AHA,BTA,N,AZ)
      WRITE(3,22)
      DO 962 I=1,N
      WRITE(3,23)I,AZ(I)
  962 CONTINUE
    6 FORMAT(1X,//,5X,17HTHEORETICAL POLES)
    7 FORMAT(3X,3HI =,I3,3X,6HAPHA =,E15.6,3X,6HBETA =,E15.6)
   10 FORMAT(1X,//,5X,18HTHEORETICAL ZEROES)
   11 FORMAT(3X,3HI =,I3,3X,5HAHA =,E15.6,3X,5HBTA =,E15.6)
   12 FORMAT(3X,//,5X,18HPREDISTORTED POLES)
   16 FORMAT(1X,//,5X,19HPREDISTORTED ZEROES)
   20 FORMAT(3X,//,5X,22HDENOMINATOR POLYNOMIAL)
   21 FORMAT(3X,3HAP(,I2,3H) =,3X,E15.6)
   22 FORMAT(3X,//,5X,20HNUMERATOR POLYNOMIAL)
   23 FORMAT(3X,3HAZ(,I2,3H) =,3X,E15.6)
      RETURN
      END
```

FIGURE B-19.—Subroutine ANEV

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE CARIP(R,E)
      RR=R/10.
      E=SQRT(10.**RR-1.)
      RETURN
      END
```

FIGURE B-20.—Subroutine CARIP

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE CAZEO(N,ALPA,BETA,SPOL,E)
      DIMENSION AN(20),ALPA(20),BETA(20)
      XN=N
C     CALCULATE ZEROES    N ODD
      IF(E)2,3,2
    2 YY=2./E
      YLN=ALOG(YY)
      AA=(YLN-E**2/4.-3.*E**4/32.-15.*E**6/288.-105.*E**8/3072.)/XN
      EE=2.7183
      CA2=(EE**AA+EE**(-AA))/2.
      GO TO 4
    3 CA2=1.0E+15
    4 M=(N-1)/2
      DO 10 I=1,M
      A13I=I
      A13N=N
      AN(I)=(2.*A13I-1.)/A13N
      ALPA(I)=0.
      BETA(I)=(COS(AN(I)*3.14159/2.))/CA2
   10 CONTINUE
      SPOL=0.
      RETURN
      END
```

FIGURE B-21.—Subroutine CAZEO

```
// JOB
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE SEPOL(APHA,BETA,N,J,A)
      DIMENSION APHA(10),BETA(10),A(10)
      M=N/2
      MM=M-1
      DO 100 I=1,MM
      IF(I-J)2,2,1
    1 APHA(I)=APHA(I)
      BETA(I)=BETA(I)
      GO TO 100
    2 APHA(I)=APHA(I+1)
      BETA(I)=BETA(I+1)
  100 CONTINUE
      NN=N-2
      CALL XEPOL(APHA,BETA,NN,A)
      RETURN
      END
```

FIGURE B-22.—Subroutine SEPOL

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE AFIN(AP,AZ,N,Q,Z,FO,YYY,BB)
      DIMENSION AP(10),AZ(10),AN(10),AD(10),F(10)
      CALL ADPOL(AP,AZ,N,AN,AD)
      WRITE(3,30)
   30 FORMAT(3X,//,5X,34HDENOMINATOR POLYNOMIAL AFTER ADPOL)
      DO 500 I=1,N
      WRITE(3,31)I,AD(I)
   31 FORMAT(3X,3HAD(,I2,3H) =,E15.6)
  500 CONTINUE
      WRITE(3,32)
   32 FORMAT(3X,//,5X,32HNUMERATOR POLYNOMIAL AFTER ADPOL)
      M2=N
      DO 600 I=1,M2
      WRITE(3,33)I,AN(I)
   33 FORMAT(3X,3HAN(,I2,3H) =,E15.6)
  600 CONTINUE
      CALL SYDIV(AN,AD,N,F)
      CALL FISDA(Z,FO,YYY,BB,F,N)
      M7=N+1
      DO 25 I=1,M7
      WRITE(3,39)I,F(I)
   25 CONTINUE
   39 FORMAT(3X,2HF(,I2,4H) = ,E15.7)
      RETURN
      END
```

FIGURE B-23.—Subroutine AFIN

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE XEPOL(APHA,BETA,N,AA)
      DIMENSION B(20),APHA(20),BETA(20),C(20),A(20,20),AA(20)
C     PROGRAM TO TAKE PRODUCT OF POLYNOMIAL TERMS    N  EVEN
C
C     MULTIPLY CONJUGATE TERMS
    1 M=N/2
      DO 10 I=1,M
      B(2*I)=-2.*APHA(I)
      C(2*I)=APHA(I)**2+BETA(I)**2
   10 CONTINUE
C     SET UP FIRST POLYNOMIAL
      A(1,2)=B(2)
      A(2,2)=C(2)
C     CALCULATE PRODUCT TERMS
      MM=N-2
      DO 20 J=2,MM,2
      A(1,J+2)=A(1,J)+B(J+2)
      A(2,J+2)=A(2,J)+A(1,J)*B(J+2)+C(J+2)
      A(J+1,J+2)=A(J-1,J)*C(J+2)+A(J,J)*B(J+2)
      A(J+2,J+2)=A(J,J)*C(J+2)
      IF(J-2)20,20,11
   11 DO 20 I=3,J
      A(I,J+2)=A(I,J)+A(I-1,J)*B(J+2)+A(I-2,J)*C(J+2)
   20 CONTINUE
   21 DO 30 I=1,N
      AA(I)=A(I,N)
   30 CONTINUE
      RETURN
      END
```

FIGURE B-24.—Subroutine XEPOL

```
// JOB
// FOR
*ONE WORD INTEGERS
* LIST SOURCE PROGRAM
      SUBROUTINE SYDIV(AN,AD,N,F)
      DIMENSION AN(20),AD(20),A(30),B(30),R(20),F(20)
C     PROGRAM FOR SYNTHETIC DIVISION
C     DARLINGTON SYNTHESIS
      DO 10 I=1,30
      A(I)=0.
      B(I)=0.
   10 CONTINUE
      A(1)=2.
      DO 20 I=1,N
      A(I+1)=AN(I)
   20 B(I)=AD(I)
      M=N-1
      DO 30 J=1,M
      F(J)=A(1)/B(1)
      M1=N-J-1-1
      IF(M1)50,55,55
   55 MM=N-J-1
      DO 21 I=1,MM
      R(I)=A(I+2)-B(I+2)*A(1)/B(1)
   21 CONTINUE
   50 M7=N-J+2
      M8=N-J
      R(M8)=A(M7)
      M5=N+1-J
      DO 22I=1,M5
   22 A(I)=B(I)
      MMM=N-J
      DO 23 I=1,MMM
   23 B(I)=R(I)
   30 CONTINUE
      M4=N+1
      DO 40 J=N,M4
      F(J)=A(1)/B(1)
      R(1)=A(2)
      A(1)=B(1)
      B(1)=R(1)
   40 CONTINUE
      RETURN
      END
```

FIGURE B-25.—Subroutine SYDIV

```
// JOB
// FOR
*LIST SOURCE PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE XOPOL(APHA,BETA,N,SPOL,AA)
      DIMENSION APHA(20),BETA(20),A(20,20),AA(20)
C     PROGRAM TO TAKE PRODUCT OF POLYNOMIALS  TERMS  N ODD
C
      M=N-1
      CALL XEPOL(APHA,BETA,M,AA)
      A(1,N)=AA(1)-SPOL
      A(N,N)=-SPOL*AA(M)
      MM=N-2
      DO 10 I=1,MM
      I2=I+1
      A(I2,N)=AA(I2)-SPOL*AA(I)
   10 CONTINUE
      DO 30 I=1,N
      AA(I)=A(I,N)
   30 CONTINUE
      RETURN
      END
```

FIGURE B-26.—Subroutine XOPOL

```
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE CMPRL (I,B,W,A,G,D,E,Q)
      REAL K, KMIN, KMAX
      C2 = 1.00E-10
      CA = 1.00E-11
      C2 = C2 + CA
      IF (B-1.00) 11,11,12
   12 KMIN = 1.0 - B**2/10.0
      K = .90
      GO TO 13
   11 KMIN = 1.0 - B**2/18.0
      K = KMIN
   13 KMAX = 1.0 + B**2/3.0
      E   = K
      ROMIN= (B-2.*SQRT(1.-KMIN))*(B**2- 8.*(1.-KMIN))/(16.*C2*W*KMIN)
      IF (100.0-ROMIN) 14,15,15
   14 RO = 100.0
      GO TO 16
   15 RO = ROMIN
   16 CX1 = B**2 + 4.*RO*C2*W*B - 4.*(1.-K)
      CD1 = 4.*RO*W*(2.*RO*W*C2+B*K)
      C1 = (CX1 - SQRT(CX1**2-8.*C2*CD1))/CD1
      X1 =(B-RO*C1*W*(2.-K))/(2.*(C2+C1*(1.-K))*W)
      Y1 = (1.+RO*C1*W*(RO*C1*W-B))/(C1*(C2+C1*(1.-K))*W**2)
      R1 = X1 + SQRT(X1**2-Y1)
      X2 = (B-RO*K*C1*W)/(2.*C2*W)
      Y2 = (C2/C1 + (1.-K) + RO*C2*W*(RO*C1*W-B))/(C2*W)**2
      R2 = X2 - SQRT(X2**2-Y2)
      X1S = X1**2
      X2S = X2**2
      Z1 = SQRT(X1S -Y1)
      Z2 = SQRT(X2S -Y2)
      RG = 0.0
      A  =   (R1*R2+RO*(R1+R2))*C1*C2*W**2
      G  =   RO*R2*C1*C2*W**2
      D  =   RO*C1*W
      BA = (R1*C1*(1.-K) + C2*(R1+R2) + RO*C1)*W
      RCW = RO*C1*W
      C2M = C1*((B-RCW*K)**2 - 4.*(1.-K))/(4.*(1.-RCW*(B-RCW))) -CA
      WRITE (3,41) I
      IF (X1**2-Y1) 29,20,20
   20 IF (X2**2-Y2) 29,21,21
   21 IF (R2) 29,29,22
   22 IF (.001-ABS(B-BA)) 29,29,23
   23 IF (.001-ABS(A  - 1.00)) 29,29,24
   29 WRITE (3,42)
   24 CONTINUE
      C2 = C2 - CA
      R1 = R1 - RG
      F = W/6.283185
      FLIM = F*SQRT(E/G)
      ZO = RO*(1.+W*C1*(R1*K-RO)/B)
      TERM = ((R2+RO)*C1*C2*W**2)**2 + (C2+C1*(1.-K))**2*W**2
      ZIN = B/SQRT(TERM)
      WRITE(3,43) B,F,Q,FLIM,K,RO,RG,CA,R1,R2,C1,C2
      WRITE(3,44) KMIN,KMAX,ROMIN,C2M,BA,A,ZO,ZIN
      WRITE(3,45) X1,Z1,X1S,Y1
      WRITE(3,46) X2,Z2,X2S,Y2
   41 FORMAT(1H0,7X,'I =',I2,7X,'LOW-PASS  SECOND ORDER')
   42 FORMAT(1H+,'ERROR')
   43 FORMAT (1H0,13X,'B =',E12.5,9X,'F =',E12.5,9X,'Q =',E12.5,
     1  6X,'FLIM =',E12.5/14X,'K =',E12.5,8X,'RO =',E12.5,8X,
     2  'RG =',E12.5,8X,'CA =',E12.5/13X,'R1 =',E12.5,8X,'R2 =',
     3  E12.5,8X,'C1 =',E12.5,8X,'C2 =',E12.5)
   44 FORMAT (1H0,10X,'KMIN =',E12.5,6X,'KMAX =',E12.5,5X,'ROMAX =',
     1 E12.5,7X, 'C2M =',E12.5/11X,'BCAL =',E12.5,9X,'A =',E12.5,
     2  8X,'ZO =',E12.5,7X,'ZIN =',E12.5)
   45 FORMAT (1H0,12X,'R1 =  X1 + Z1  =  X1 + SQRT( X1**2 - Y1 )'/13X,
     1 'X1 =',E12.5,8X,'Z1 =',E12.5,5X,'X1**2 =',E12.5,8X,'Y1 =',E12.5)
   46 FORMAT (1H0,12X,'R2 =  X2 - Z2  =  X2 - SQRT( X2**2 - Y2 )'/13X,
     1 'X2 =',E12.5,8X,'Z2 =',E12.5,5X,'X2**2 =',E12.5,8X,'Y2 =',E12.5,
     2  //)
      RETURN
      END
```

FIGURE B-27.—Subroutine CMPRH

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE CMPRH (I,B,W,A,G,D,E,Q)
      REAL K, KMIN, KMAX
      CA = 1.00E-11
      E = 0.0
      IF (B-1.00) 11,11,12
   12 KMIN = 1.0 - B**2/10.0
      K = .90
      GO TO 13
   11 KMIN = 1.0 - B**2/18.0
      K = KMIN
   13 KMAX = 1.0 + B**2/3.0
      CMIN = 2.0*CA*(5.-B**2)/(B**2-8.*(1.-K))
      IF (CMIN-1.0E-10) 15,14,14
   15 C = 1.0E-10
      GO TO 16
   14 C = CMIN
   16 ROMIN = B/(18.*W*C)
      IF (100.0-ROMIN) 18,19,19
   18 RO = 100.0
      GO TO 17
   19 RO = ROMIN
   17 RG = RO
      X1 = B/(4.*C*W) -(4.*RO*(C+CA)+RG*(C*(2.-K)+CA))/(4.*(C+2.*CA))
      TERM = (RO+RG*(1.-K))*(B-C*W*(2.*RO+RG))
      Y1 = (CA + C*(1.-K) - C**2*W*TERM)/(2.*C**2*W**2*(C+2.*CA))
      R1 = X1 + SQRT(X1**2-Y1)
      DR = C*(R1*(C+2.*CA) +RO*C + RG*C*(1.-K))
      X2 =1./(W**2*DR)
      Z2 = RG*R1*C**2/DR
      R2 = X2 - Z2
      X1S = X1**2
      Z1 = SQRT(X1S -Y1)
       A    =  R2*(C**2*(RG*(1.-K)+RO+R1)+2.*R1*CA*C)*W**2
     *          + RG*R1*(C*W)**2
      BA =R2*(CA+C*(1.-K))*W + 2.*C*W*(RO+R1) + RG*C*W
      G     = R2*(R1*K +RO*(1.+CA/C))*(C*W)**2
      D = RO*C*W
      WRITE(3,41) I
      IF (X1**2-Y1) 29,20,20
   20 IF (X1) 29,29,21
   21 IF (R2) 29,29,22
   22 IF (.001-ABS(B-BA)) 29,29,23
   23 IF (.001-ABS(A  - 1.00)) 29,29,24
   29 WRITE(3,42)
   24 CONTINUE
      F = W/6.283185
      FLIM = F*D/G
      ZO = RO*(1. +W*C*(R2*K-2.*RO)/B)
      TERM1 = (C*W*(B-C*(R1+RO)))**2
      TERM2 = (W*(C-CA*(1.-2.*R1*R2*CA*C*W**2)))**2
      ZIN = B/SQRT(TERM1 + TERM2)
      C1 = C
      C2 = C
      WRITE(3,43) B,F,Q,FLIM,K,RO,RG,CA,R1,R2,C1,C2
      WRITE(3,44) KMIN,KMAX,ROMIN,CMIN,BA,A,ZO,ZIN
      WRITE(3,45) X1,Z1,X1S,Y1
      WRITE (3,46) X2,Z2
   41 FORMAT(1H0,7X,'I =',I2,7X,'HIGH-PASS  SECOND ORDER')
   42 FORMAT(1H+,'ERROR')
   43 FORMAT (1H0,13X,'B =',E12.5,9X,'F =',E12.5,9X,'Q =',E12.5,
     1  6X,'FLIM =',E12.5/14X,'K =',E12.5,8X,'RO =',E12.5,8X,
     2  'RG =',E12.5,8X,'CA =',E12.5/13X,'R1 =',E12.5,8X,'R2 =',
     3  E12.5,8X,'C1 =',E12.5,8X,'C2 =',E12.5)
   44 FORMAT (1H0,10X,'KMIN =',E12.5,6X,'KMAX =',E12.5,5X,'ROMAX =',
     1 E12.5,7X,'CMIN =',E12.5/11X,'BCAL =',E12.5,9X,'A =',E12.5,
     2  8X,'ZO =',E12.5,7X,'ZIN =',E12.5)
   45 FORMAT (1H0,12X,'R1 =  X1 + Z1  =  X1 + SQRT( X1**2 - Y1 )'/13X,
     1 'X1 =',E12.5,8X,'Z1 =',E12.5,5X,'X1**2 =',E12.5,8X,'Y1 =',E12.5)
   46 FORMAT (1H0,12X,'R2 =  X2 - Z2'/13X,'X2 =',E12.5,8X,'Z2 =',E12.5
     1  //)
      RETURN
      END
```

FIGURE B–28.—Subroutine CMPRH

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE CPFRL(B,W,A,G,D,E,Q,I)
      C1MAX = B/(200.*W)
      IF (C1MAX-1.0E-09) 11,11,10
   10 C1 = 1.0E-09
      GO TO 12
   11 C1 = C1MAX
   12 RLMIN = 2.*B/(C1*W)
      IF (RLMIN-1.0E+05) 13,14,14
   13 RL = 1.0E05
      GO TO 15
   14 RL = RLMIN
   15 RGMAX = .5/(C1MAX*W/B - 1./1.0E05)
      IF (100.0-RGMAX) 16,17,17
   16 RG = 100.0
      GO TO 18
   17 RG = RGMAX
   18 R1 = 1./(W*C1/B - 1./RL) - RG
      A = 0.0
      G = 0.0
      D = 0.0
      E = RL/(R1+RG+RL)
      WRITE (3,40) I
      IF (R1) 21,21,22
   21 WRITE (3,41)
   22 F = W/6.283185
      WRITE (3,42) B,F,W,Q,R1,C1,RG,RL
      WRITE (3,43) C1MAX,RGMAX,RLMIN
   40 FORMAT (1H0,7X,'I =',I2,7X,'LOW-PASS  FIRST ORDER')
   41 FORMAT(1H+,'ERROR')
   42 FORMAT(1H0,13X,'B =',E12.5,9X,'F =',E12.5,9X,'W =',E12.5,9X,'Q =',
     1E12.5/13X,'R1 =',E12.5,8X,'C1 =',E12.5,8X,'RG =',E12.5,8X,'RL =',
     2E12.5)
   43 FORMAT(1H0,9X,'C1MAX =',E12.5,5X,'RGMAX =',E12.5,5X,'RLMIN =',
     1E12.5//)
      RETURN
      END
```

FIGURE B–29.—Subroutine CPFLR

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE CPFRH(B,W,A,G,D,E,Q,I)
      C1 = 1.0E-09
      RGMAX = B/(1.2*C1*W)
      IF (100.0-RGMAX) 11,12,12
   11 RG = 100.0
      GO TO 14
   12 RG = RGMAX
   14 RLMIN = B/(5.9*C1*W)
      IF (RLMIN-1.0E05) 15,16,16
   15 RL = 1.0E05
      GO TO 18
   16 RL = RLMIN
   18 R1 = RL*(B-RG*C1*W)/(C1*W*(RG+RL)-B)
      A = 0.0
      G = 0.0
      E = 0.0
      D = R1*RL*C1*W/(R1+RL)
      WRITE (3,40) I
      IF (R1) 21,21,22
   21 WRITE (3,41)
   22 F = W/6.283185
      WRITE (3,42) B,F,W,Q,R1,C1,RG,RL
      WRITE (3,43) C1,RGMAX,RLMIN
   40 FORMAT (1H0,7X,'I =',I2,7X,'HIGH-PASS  FIRST ORDER')
   41 FORMAT(1H+,'ERROR')
   42 FORMAT(1H0,13X,'B =',E12.5,9X,'F =',E12.5,9X,'W =',E12.5,9X,'Q =',
     1E12.5/13X,'R1 =',E12.5,8X,'C1 =',E12.5,8X,'RG =',E12.5,8X,'RL =',
     2E12.5)
   43 FORMAT(1H0, 8X,'FOR C1 =',E12.5,5X,'RGMAX =',E12.5,5X,'RLMIN =',
     1E12.5//)
      RETURN
      END
```

FIGURE B–30.—Subroutine CPFRH

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE BPBCW (BL,CL,FO,BW,B1,B2,W1,W2)
      BWR = BW/FO
      BN = BL*BWR
      CN = CL*BWR**2
      E = SQRT(CN-(BN/2.)**2)
      RS = BN**2/2. - CN - 4.
      QS = BN*E
      R = SQRT((RS+SQRT(RS**2+QS**2))/2.)
      Q = SQRT(R**2-RS)
      IF (QS) 41,40,40
   41 R = -R
   40 B1 = BN/2. + R
      C1 = ((R+BN/2.)/2.)**2 + ((Q+E)/2.)**2
      B2 = BN/2. - R
      C2 =((R-BN/2.)/2.)**2 + ((Q-E)/2.)**2
      CS1 = SQRT(C1)
      CS2 = SQRT(C2)
      B1 = B1/CS1
      B2 = B2/CS2
      PI = 6.283185
      W1 = PI*FO*CS1
      W2 = PI*FO*CS2
      RETURN
      END
```

FIGURE B-31.—Subroutine BPBCW

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE CARSP(MM,A,B,G,D,E,W,FMIN,FMAX,DF,SCAL)
      DIMENSION A(20),B(20),C(20),G(20),D(20),E(20),W(20)
      WRITE(3,90)
      WRITE(3,91)
      PI = 3.14159
      DO 10 I=1,MM
      F = W(I)/(2.*PI)
      C(I) = 1.0
   10 WRITE(3,92) I,A(I),B(I),C(I),G(I),D(I),E(I),F
      FM = FMIN
      IF (DF)  40,40,80
   80 WF = 2.*PI*FM
      TT = 1.0
      PDT = 0.0
      PWT = 0.0
      DLT = 0.0
      WRITE(3,93) FM
      DO 20 I=1,MM
      WN = WF/W(I)
      C(I) = 1.0
      XN = E(I) - G(I)*WN**2
      YN = D(I)*WN
      XD = C(I) - A(I)*WN**2
      YD = B(I)*WN
      T = SQRT((XN**2 + YN**2)/(XD**2 + YD**2))
      IF (XN) 21,19,21
   19 PN = PI/2.0
      GO TO 25
   21 PN = ATAN(YN/XN)
      IF (YN) 22,23,23
   22 PN = PN + PI
   23 IF (YN/XN) 24,25,25
   24 PN = PN + PI
   25 PD = ATAN(YD/XD)
      IF (YD) 26,27,27
   26 PD = PD + PI
   27 IF (YD/XD) 28,29,29
   28 PD = PD + PI
   29 PHD = (PN - PD)*180./PI
      PHW = (PN - PD)/WF
      DLY = -(D(I)*XN/W(I) +2.*G(I)*YN*WF/W(I)**2)/(XN**2 +YN**2)
     1      +(B(I)*XD/W(I) +2.*A(I)*YD*WF/W(I)**2)/(XD**2 +YD**2)
      TT = TT*T
      PDT = PDT + PHD
      PWT = PWT ● PHW
      DLT = DLT + DLY
      WRITE(3,94) I,T,PHD,PHW,DLY
   20 CONTINUE
      IF(SCAL) 31,31,32
   31 SCAL = 1.0
   32 TT = TT/SCAL
      WRITE(3,95) TT,PDT,PWT,DLT
      FM = FM + DF
      IF(FMAX - FM) 40,80,80
   90 FORMAT(1H0,7X,'FILTER RESPONSE CURVES',//)
   91 FORMAT(1H0,8X,'I',9X,'A',14X,'B',14X,'C',14X,'G',14X,'D',14X,'E',
     *14X,'F'/)
   92 FORMAT(8X,I2,7E15.5)
   93 FORMAT(1H0,/3X,'FREQ =',E14.6)
   94 FORMAT (6X,'I =',I2,
     1             4X,'A=',E13.5,4X,'P=',E13.5,4X,'P/W=',E13.5,4X,'D=',
     2   E13.5)
   95 FORMAT(14X,'AT=',E13.5,3X,'PT=',E13.5,3X,'PT/W=',E13.5,3X,'DT=',
     *E13.5)
   40 RETURN
      END
```

FIGURE B-32.—Subroutine CARSP

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE PDSEV(N,Q,APHA)
      DIMENSION APHA(20)
      M = N/2
      IF (APHA(1)+1./Q)12,11,11
   11 Q =-.1/(APHA(1)+.001)
   12 DO 13 I=1,M
   13 APHA(I) = APHA(I) + 1./Q
      RETURN
      END
```

FIGURE B-33.—Subroutine PDSEV

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE PDSOD(N,Q,APHA,SPOL)
      DIMENSION APHA(20)
      M = (N-1)/2
      IF (N-2) 7,7,8
    7 IF (SPOL + 1./Q)10,9,9
    9 Q =-.1/(SPOL+.001)
   10 GO TO 14
    8 IF (APHA(1)+1./Q)12,11,11
   11 Q =-.1/(APHA(1)+.001)
   12 DO 13 I=1,M
   13 APHA(I) = APHA(I) + 1./Q
   14 SPOL = SPOL + 1./Q
      RETURN
      END
```

FIGURE B-34.—Subroutine PDSOD

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE CASEL(I,XN,A,BN,G,D,E,WN)
      REAL K, KMIN, KMAX
      DIMENSION A(20), BN(20), G(20), D(20), E(20), WN(20), X(20),XN(8)
      EQUIVALENCE (B,X(1)),  (F,X(2)), (C2,X(3)),(RG,X(4)), (K,X(5)),
     *(RO,X(6)), (CA,X(7)), (C1,X(8))
      WRITE(3,90)
      DO 9 J = 1,8
    9 X(J) = XN(J)
      C1 = 0.0
   10 READ(2,80) IX,XX
      IF (IX-9) 20,99,20
   20 IF (IX) 30,30,22
   22 X(IX) = XX
      XN(IX) = XX
      GO TO 10
   30 I = I + 1
      W = 6.283185*F
      C2 = C2 + CA
      XC1 = 0.0
      ZC1 = 0.0
      XC1S = 0.0
      YC1 = 0.0
      IF (C1) 32,32,33
   32 CX1 = B**2 + 4.*RO*C2*W*B - 4.*(1.-K)
      CD1 = 4.*RO*W*(2.*RO*W*C2+B*K)
      C1 = (CX1 - SQRT(CX1**2-8.*C2*CD1))/CD1
   33 RCW = RO*C1*W
      X1 =(B-RO*C1*W*(2.-K))/(2.*(C2+C1*(1.-K))*W)
      Y1 = (1.+RO*C1*W*(RO*C1*W-B))/(C1*(C2+C1*(1.-K))*W**2)
      X1S = X1**2
      Z1 = SQRT(X1S-Y1)
      R1 = X1 + Z1
      X2 = (B-RO*K*C1*W)/(2.*C2*W)
      Y2 = (C2/C1 + 1. - K + RO*C2*W*(RCW-B))/(C2*W)**2
      X2S = X2**2
      Z2 = SQRT(X2S-Y2)
      R2 = X2 - Z2
      BN(I) = B
      A(I) = (R1*R2 + RO*(R1+R2))*C1*C2*W**2
      BA = (R1*C1*(1.-K) + C2*(R1+R2) + RO*C1)*W
      G(I) = RO*R2*C1*C2*W**2
      D(I) = RO*C1*W
      E(I) = K
      WN(I) = W
      FLIM = F*SQRT(E(I)/G(I))
      IF (B-1.00) 36,36,35
   35 KMIN = 1.0 - B**2/10.0
      GO TO 37
   36 KMIN = 1.0 - B**2/18.0
   37 KMAX = 1.0 + B**2/3.0
      ROMIN= (B-2.*SQRT(1.-KMIN))*(B**2-8.*(1.-KMIN))/(16.*C2*W*KMIN)
      ZO = RO*(1.+W*C1*(R1*K-RO)/B)
      TERM = ((R2+RO)*C1*C2*W**2)**2 + (C2+C1*(1.-K))**2*W**2
      ZIN = B/SQRT(TERM)
      C2M = C1*((B-RCW*K)**2 - 4.*(1.-K))/(4.*(1.-RCW*(B-RCW))) -CA
      R1 = R1 - RG
      C2 = C2 - CA
      WRITE(3,91) I
      IF (X1S-Y1) 69,60,60
   60 IF (X2S-Y2) 69,61,61
   61 IF (X2-Z2) 69,69,62
   62 IF (.0001-ABS(A(I)-1.0)) 69,69,63
   63 IF (.0001-ABS(B-BA))69,69,64
   64 IF (R1) 69,69,65
   65 IF (X1) 69,69,66
   66 IF (C2M-C2) 69,67,67
   69 WRITE(3,92)
   67 WRITE(3,93) B,F,W,FLIM,K,RO,RG,CA,R1,R2,C1,C2
      WRITE(3,94) KMIN,KMAX,ROMIN,C2M,BA,A(I),ZO,ZIN
      WRITE(3,95) X1,Z1,X1S,Y1
      WRITE(3,96) X2,Z2,X2S,Y2
      GO TO 10
   80 FORMAT(I1,E11.0)
   90 FORMAT (1H0,//18X,'LOW-PASS SECOND ORDER FILTER NETWORKS'/)
   91 FORMAT(1H0,7X,'I =',I2)
   92 FORMAT(1H+,'ERROR')
   93 FORMAT (1H0,13X,'B =',E12.5,9X,'F =',E12.5,9X,'W =',E12.5,
     1  6X,'FLIM =',E12.5/14X,'K =',E12.5,8X,'RO =',E12.5,8X,
     2  'RG =',E12.5,8X,'CA =',E12.5/13X,'R1 =',E12.5,8X,'R2 =',
     3  E12.5,8X,'C1 =',E12.5,8X,'C2 =',E12.5)
   94 FORMAT (1H0,10X,'KMIN =',E12.5,6X,'KMAX =',E12.5,5X,'ROMAX =',
     1 E12.5,7X, 'C2M =',E12.5/11X,'BCAL =',E12.5,9X,'A =',E12.5,
     2  8X,'ZO =',E12.5,7X,'ZIN =',E12.5)
   95 FORMAT (1H0,12X,'R1 =  X1 + Z1  =  X1 + SQRT( X1**2 - Y1 )'/13X,
     1 'X1 =',E12.5,8X,'Z1 =',E12.5,5X,'X1**2 =',E12.5,8X,'Y1 =',E12.5)
   96 FORMAT (1H0,12X,'R2 =  X2 - Z2  =  X2 - SQRT( X2**2 - Y2 )'/13X,
     1 'X2 =',E12.5,8X,'Z2 =',E12.5,5X,'X2**2 =',E12.5,8X,'Y2 =',E12.5,
     2  //)
   99 RETURN
      END
```

FIGURE B-35.—Subroutine CASEL

```
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE  CASEH(I,XN,A,BN,G,D,E,WN)
      REAL K, KMIN, KMAX
      DIMENSION A(20),BN(20),G(20),D(20),E(20),WN(20),X(20),XN(8)
      EQUIVALENCE (B,X(1)),(F,X(2)),(C,X(3)),(RG,X(4)),(K,X(5)),
     *  (RO,X(6)),(CA,X(7))
      WRITE(3,90)
      DO 9 J = 1,8
    9 X(J) = XN(J)
   10 READ(2,80) IX,XX
      IF (IX-9) 20,99,20
   20 IF (IX) 30,30,22
   22 X(IX) = XX
      XN(IX) = XX
      GO TO 10
   30 I = I + 1
      W = 6.283185*F
      X1 = B/(4.*C*W) -(4.*RO*(C+CA)+RG*(C*(2.-K)+CA))/(4.*(C+2.*CA))
      TERM = (RO+RG*(1.-K))*(B-C*W*(2.*RO+RG))
      Y1 = (CA + C*(1.-K) - C**2*W*TERM)/(2.*C**2*W**2*(C+2.*CA))
      X1S = X1**2
      R1 = X1 + SQRT(X1S - Y1)
      Z1 = SQRT(X1S - Y1)
      DR = C*(R1*(C+2.*CA) +RO*C + RG*C*(1.-K))
      X2 =1./(W**2*DR)
      Z2 = RG*R1*C**2/DR
      R2 = X2 - Z2
      BN(I) = B
      WN(I) = W
      A(I) =  R2*(C**2*(RG*(1.-K)+RO+R1)+2.*R1*CA*C)*W**2
     *          + RG*R1*(C*W)**2
      BA =R2*(CA+C*(1.-K))*W + 2.*C*W*(RO+R1) + RG*C*W
      G(I) = R2*(R1*K +RO*(1.+CA/C))*(C*W)**2
      D(I) = RO*C*W
      E(I) = 0.0
      FLIM = F*D(I)/G(I)
      IF(B-1.00) 36,36,35
   35 KMIN = 1.0 -B**2/10.0
      GO TO 37
   36 KMIN = 1.0 -B**2/18.0
   37 KMAX = 1.0 +B**2/3.0
      CMIN = 2.0*CA*(5.-B**2)/(B**2-8.*(1.-K))
      ROMIN = B/(18.*W*C)
      ZO = RO*(1. +W*C*(R2*K-2.*RO)/B)
      TERM1 = (C*W*(B-C*(R1+RO)))**2
      TERM2 = (W*(C-CA*(1.-2.*R1*R2*CA*C*W**2)))**2
      ZIN = B/SQRT(TERM1 + TERM2)
      C1 = C
      C2 = C
      WRITE (3,91) I
      IF(X1S -Y1) 69,61,61
   61 IF(X2 - Z2) 69,69,62
   62 IF(.0001 -ABS(A(I)-1.0)) 69,69,63
   63 IF(.0001 -ABS(B - BA)) 69,69,64
   64 IF(X1) 69,69,65
   69 WRITE (3,92)
   65 WRITE(3,93) B,F,W,FLIM,K,RO,RG,CA,R1,R2,C1,C2
      WRITE(3,94) KMIN,KMAX,ROMIN,CMIN,BA,A(I),ZO,ZIN
      WRITE(3,95) X1,Z1,X1S,Y1
      WRITE (3,96) X2,Z2
      GO TO 10
   80 FORMAT(I1,E11.0)
   90 FORMAT (1H0,//18X,'HIGH-PASS SECOND ORDER FILTER NETWORKS'/)
   91 FORMAT(1H0,7X,'I =',I2)
   92 FORMAT(1H+,'ERROR')
   93 FORMAT (1H0,13X,'B =',E12.5,9X,'F =',E12.5,9X,'W =',E12.5,
     1  6X,'FLIM =',E12.5/14X,'K =',E12.5,8X,'RO =',E12.5,8X,
     2  'RG =',E12.5,8X,'CA =',E12.5/13X,'R1 =',E12.5,8X,'R2 =',
     3  E12.5,8X,'C1 =',E12.5,8X,'C2 =',E12.5)
   94 FORMAT (1H0,10X,'KMIN =',E12.5,6X,'KMAX =',E12.5,5X,'ROMAX =',
     1 E12.5,7X,'CMIN =',E12.5/11X,'BCAL =',E12.5,9X,'A =',E12.5,
     2  8X,'ZO =',E12.5,7X,'ZIN =',E12.5)
   95 FORMAT (1H0,12X,'R1 =  X1 + Z1  =  X1 + SQRT( X1**2 - Y1 )'/13X,
     1 'X1 =',E12.5,8X,'Z1 =',E12.5,5X,'X1**2 =',E12.5,8X,'Y1 =',E12.5)
   96 FORMAT (1H0,12X,'R2 =  X2 - Z2'/13X,'X2 =',E12.5,8X,'Z2 =',E12.5,
     2   //)
   99 RETURN
      END
```

FIGURE B-36.—Subroutine CASEH

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE CAFIL (I,XN,A,BN,G,D,E,WN)
      DIMENSION A(20),BN(20),G(20),D(20),E(20),WN(20),X(8),XN(8)
      EQUIVALENCE (B,X(1)), (F,X(2)), (C1,X(3)), (RG,X(4)), (RL,X(5))
      WRITE(3,90)
      DO 9 J = 1,8
    9 X(J) = XN(J)
      RL = 1.0E25
   10 READ(2,80) IX,XX
      IF (IX-9) 20,99,20
   20 IF (IX) 30,30,22
   22 X(IX) = XX
      XN(IX) = XX
      GO TO 10
   30 I = I + 1
      W = 6.283185*F
      R1 = 1.0/(W*C1/B - 1./RL) - RG
      C1MAX = B/(200.*W)
      RLMIN = 2.*B/(C1*W)
      RGMAX = .5/(C1MAX*W/B - 1./RL)
      A(I) = 0.0
      G(I) = 0.0
      D(I) = 0.0
      E(I) = RL/(R1+RG+RL)
      BA = E(I)*C1*(R1+RG)*W
      BN(I) = B
      WN(I) = W
      WRITE(3,91) I
      IF (R1) 39,39,32
   32 IF (.001-ABS(B-BA)) 39,39,43
   39 WRITE(3,92)
   43 WRITE(3,93) B,F,W,BA,R1,C1,RG,RL
      WRITE(3,94) C1MAX,RGMAX,RLMIN,E(I)
      GO TO 10
   80 FORMAT(I1,E11.0)
   90 FORMAT (1H0,//18X,'LOW-PASS FIRST ORDER FILTER NETWORKS'/)
   91 FORMAT(1H0,7X,'I =',I2)
   92 FORMAT(1H+,'ERROR')
   93 FORMAT(1H0,13X,'B =',E12.5,9X,'F =',E12.5,9X,'W =',
     * E12.5,6X,'BCAL =',E12.5/13X,'R1 =',E12.5,8X,'C1 =',
     * E12.5,8X,'RG =',E12.5,8X,'RL =',E12.5)
   94 FORMAT(1H0,9X,'C1MAX =',E12.5,5X,'RGMAX =',E12.5,5X,'RLMIN =',
     *E12.5,9X,'E =',E12.5//)
   99 RETURN
      END
```

FIGURE B–37.—Subroutine CAFIL

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE CAFIH(I,XN,A,BN,G,D,E,WN)
      DIMENSION A(20), BN(20), G(20), D(20),E(20), WN(20), X(20),XN(8)
      EQUIVALENCE (B,X(1)), (F,X(2)), (C1,X(3)), (RG,X(4)),
     *(RL,X(5))
      WRITE(3,90)
      DO 9 J = 1,8
    9 X(J) = XN(J)
      RL = 1.0E25
   10 READ(2,80) IX,XX
      IF (IX-9) 20,99,20
   20 IF (IX) 30,30,22
   22 X(IX) = XX
      XN(IX) = XX
      GO TO 10
   30 I = I + 1
      W = 6.283185*F
      R1 = RL*(B-RG*C1*W)/(C1*W*(RG+RL)-B)
      RGMAX = B/(1.2*C1*W)
      RLMIN = B/(5.9*C1*W)
      A(I) = 0.0
      D(I) = R1*RL*C1*W/(R1+RL)
      G(I) = 0.0
      E(I) = 0.0
      BA = (RG + R1*RL/(R1+RL))*C1*W
      BN(I) = B
      WN(I) = W
      WRITE(3,91) I
      IF (R1) 39,39,32
   32 IF (.001-ABS(B-BA)) 39,39,43
   39 WRITE(3,92)
   43 WRITE(3,93) B,F,W,BA,R1,C1,RG,RL
      WRITE(3,94) C1,RGMAX,RLMIN,D(I)
      GO TO 10
   80 FORMAT(I1,E11.0)
   90 FORMAT (1H0,//18X,'HIGH-PASS FIRST ORDER FILTER NETWORKS'/)
   91 FORMAT(1H0,7X,'I =',I2)
   92 FORMAT(1H+,'ERROR')
   93 FORMAT(1H0,13X,'B =',E12.5,9X,'F =',E12.5,9X,'W =',
     * E12.5,6X,'BCAL =',E12.5/13X,'R1 =',E12.5,8X,'C1 =',
     * E12.5,8X,'RG =',E12.5,8X,'RL =',E12.5)
   94 FORMAT(1H0,8X,'FOR C1 =',E12.5,5X,'RGMAX =',E12.5,5X,
     * 'RLMIN =',E12.5,9X,'D =',E12.5//)
   99 RETURN
      END
```

FIGURE B–38.—Subroutine CAFIH

```
// JOB
// FOR
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
      SUBROUTINE TEMP (N,TG,TE,TA,TB,TC,K)
      DIMENSION R1(20),R2(20),C1(20),C2(20),DR1(20),DR2(20),DC1(20),
     1 DC2(20),RG(20),RO(20),CA(20),GA(20),TA(20),TB(20),TC(20),TG(20),
     2 TE(20),       A(20),B(20),       G(20),E(20),F(20),DA(20),DB(20),
     3           DG(20),DE(20),W(20)
      READ(2,10)J,K,L,M,T
      N=J+K+L+M
      WRITE (3,79) J,K,L,M,T
      DO 8 I = 1,N
      READ(2,12) RG(I),RO(I),GA(I),CA(I),F(I)
      READ(2,11) R1(I),R2(I),C1(I),C2(I),PR1,PR2,PC1,PC2
      WRITE (3,81) RG(I),RO(I),GA(I),CA(I),F(I)
      WRITE (3,82) R1(I),R2(I),C1(I),C2(I),PR1,PR2,PC1,PC2
      DR1(I) = R1(I)*PR1/100.0
      DR2(I) = R2(I)*PR2/100.0
      DC1(I) = C1(I)*PC1/100.0
      DC2(I) = C2(I)*PC2/100.0
    8 W(I) = 6.283185*F(I)
      I=1
  200 IF(I-J)20,20,30
   20 TA(I) = 0.0
      TB(I) = 0.0
      TC(I)= 1.0
      TG(I) = 0.0
      E(I)=(RG(I)+R1(I))*C1(I)
      DE(I)=C1(I)*DR1(I)+(RG(I)+R1(I))*DC1(I)
      DE(I) = DE(I)*(T -25.0)
      TE(I)=E(I)+DE(I)
      I=I+1
      GOTO200
   30 P=J+K
   32 IF(I-P)31,31,40
   31 A(I)=RO(I)*R2(I)*C1(I)*(C2(I)+CA(I))
      DA(I)=DR2(I)*RO(I)*C1(I)*(C2(I)+CA(I))+DC1(I)*RO(I)*R2(I)*(C2(I)+
     4 CA(I))+DC2(I)*RO(I)*R2(I)*C1(I)
      DA(I) = DA(I)*(T -25.0)
      TA(I)=A(I)+DA(I)
      B(I)=RO(I)*C1(I)
      DB(I)=RO(I)*DC1(I)
      DB(I) = DB(I)*(T -25.0)
      TB(I)=B(I)+DB(I)
      TC(I) = GA(I)
      E(I)=(R1(I)+RG(I))*C1(I)*(1.-GA(I))+(C2(I)+CA(I))*(R1(I)+RG(I)+
     9 R2(I))+RO(I)*C1(I)
      DE(I)=(C1(I)*(1.-GA(I))+C2(I)+CA(I))*DR1(I)+(C2(I)+CA(I))*DR2(I)+
     1 (R1(I)*(1.-GA(I))+RG(I)*(1.-GA(I))+CA(I)+RO(I))*DC1(I)+(R1(I)+
     2 RG(I)+R2(I))*DC2(I)
      DE(I) = DE(I)*(T -25.0)
      TE(I)=E(I)+DE(I)
      R1(I) = R1(I) + DR1(I)*(T -25.0)
      R2(I) = R2(I) + DR2(I)*(T -25.0)
      C1(I) = C1(I) + DC1(I)*(T -25.0)
      C2(I) = C2(I) + DC2(I)*(T -25.0)
      G(I)=(R1(I)+RG(I))*R2(I)*C1(I)*(C2(I)+CA(I))*(1.+RO(I)*(R1(I)+
     6RG(I)+R2(I))/((R1(I)+RG(I))*R2(I)))
      TG(I)=G(I)
      I = I + 1
      GO TO 32
   40 Q=J+K+L
   42 IF(I-Q)50,50,60
   50 TA(I) = 0.0
      B(I)=R1(I)*C1(I)
      DB(I)=C1(I)*DR1(I)+R1(I)*DC1(I)
      DB(I) = DB(I)*(T -25.0)
      TB(I)=B(I)+DB(I)
      TC(I) = 0.0
      TG(I) = 0.0
      E(I)=R1(I)*C1(I)+RG(I)*C1(I)
      DE(I)=C1(I)*DR1(I)+(R1(I)+RG(I))*DC1(I)
      DE(I) = DE(I)*(T -25.0)
      TE(I)=E(I)+DE(I)
      I = I + 1
      GO TO 42
   60 U=J+K+L+M
   62 IF(I-U)70,70,80
   70 A(I)=R1(I)*R2(I)*C1(I)*C2(I)*(GA(I)+(RO(I)/R1(I)))
      DA(I)=R2(I)*C1(I)*C2(I)*DR1(I)+(R1(I)*GA(I) +RO(I))*(DR2(I)*C1(I)*
     1  C2(I) +DC1(I)*R2(I)*C2(I)+DC2(I)*R2(I)*C1(I))
      DA(I) = DA(I)*(T -25.0)
      TA(I)=A(I)+DA(I)
      B(I)=RO(I)*C1(I)
      DB(I)=RO(I)*DC1(I)
      DB(I) = DB(I)*(T -25.0)
      TB(I)=B(I)+DB(I)
      TC(I) = 0.0
      G(I)=C1(I)*C2(I)*(R1(I)*R2(I)+R2(I)*RO(I)+R1(I)*RG(I))+R1(I)*R2(I)
     1 *CA(I)*(C1(I)+C2(I))
      DG(I)=DR1(I)*(C1(I)*C2(I)*(R2(I)+RG(I))+R2(I)*CA(I)*(C1(I)+C2(I)))
     1 +DR2(I)*(C1(I)*C2(I)*(R1(I)+RO(I))+R1(I)*CA(I)*(C1(I)+C2(I)))
     2 +DC1(I)*(R2(I)*C2(I)*(R1(I)+RO(I))+R1(I)*(RG(I)*C2(I)+R2(I)*
     3 CA(I)))+DC2(I)*(R2(I)*C1(I)*(R1(I)+RO(I))+R1(I)*(RG(I)*C1(I)+
     4 R2(I)*CA(I)))
      DG(I) = DG(I)*(T -25.0)
      TG(I)=G(I)+DG(I)
      E(I)=(R1(I)+RO(I))*(C1(I)+C2(I))+R2(I)*C2(I)*(1.-GA(I))+R2(I)*
     1 CA(I)+RG(I)*C1(I)
      DE(I)=(C1(I)+C2(I))*DR1(I)+(C2(I)-C2(I)*GA(I)+CA(I))*DR2(I)+(R1(I)
     1 +RO(I)+RG(I))*DC1(I)+(R1(I)+RO(I)+R2(I)-R2(I)*GA(I))*DC2(I)
      DE(I) = DE(I)*(T -25.0)
      TE(I)=E(I)+DE(I)
      I = I + 1
      GO TO 62
   80 DO 90 I = 1,N
      TA(I) = TA(I)*W(I)**2
      TB(I) = TB(I)*W(I)
      TG(I) = TG(I)*W(I)**2
   90 TE(I) = TE(I)*W(I)
   10 FORMAT (4I5/1E10.0)
   11 FORMAT (2E10.0,2E15.0/4E10.0)
   12 FORMAT(3F10.0,2E15.0)
   79 FORMAT (1H1,25X,'TEMPERATURE DEPENDENT FILTER RESPONSE PROGRAM'
     1 ////8X,'INPUT DATA'//
     2 8X,'LP 1ST ORDER =',I2,7X,'LP 2ND ORDER =',I2/
     3 8X,'HP 1ST ORDER =',I2,7X,'HP 2ND ORDER =',I2,
     4 15X,'TEMP =',E12.5/)
   81 FORMAT (1H0,7X,'RG =',E12.5,7X,'RO =',E12.5,8X,'K =',E12.5,
     1 7X,'CA =',E12.5,8X,'F =',E12.5)
   82 FORMAT (8X,'R1 =',E12.5,7X,'R2 =',E12.5,7X,'C1 =',E12.5,7X,
     1 'C2 =',E12.5/7X,'PR1 =',E12.5,6X,'PR2 =',E12.5,6X,'PC1 =',
     2 E12.5,6X,'PC2 =',E12.5/)
      RETURN
      END
```

FIGURE B–39.—Subroutine TEMP

# Bibliography


CALAHAN, DONALD A.: Modern Network Synthesis. Hayden Book Co., New York, 1964.

CHEN, WAYNE H.: Linear Network Design and Synthesis. McGraw-Hill Book Co., New York, 1964.

FOSTER, EDWARD J.: Active Lowpass Filter Design. IEEE Transactions on Audio, vol. AU-1?, no. 5, September 1965, pp. 104-111.

GEFFE, PHILLIPS R.: Simplified Modern Filter Design. J. R. Rider, 1963.

GUSSOW, SIDNEY S.: Analysis and Synthesis of Active Quadratic Filter Networks for Telemetry Applications. A Thesis, University of Alabama in Huntsville, 1969.

HERRERO, JOSE L., AND WILLONER, GIDEON: Synthesis of Filters. Prentice-Hall, Englewood Cliff, N.J., 1966.

KUO, FRANKLIN T.: Network Analysis and Synthesis. John Wiley & Sons, New York, 1962.

SALLEN, R. P., AND KEY, E. L.: A Practical Method of Designing RC Active Filters. IRE Transactions on Circuit Theory, March 1955, pp. 74-85.

SESHU, SUNDARAM; AND BALABANIAN, NORMAN: Linear Network Analysis. John Wiley & Sons, New York, 1963.

SKWIRZYNSKI, J. K.: Design Theory and Data for Electrical Filters. D. Van Nostrand Co., New York, 1965.

VAN VALKENBURG: Introduction to Modern Network Synthesis. John Wiley & Sons, New York, 1960, pp. 373-385.

WEINBERG, LOUIS: Network Analysis and Synthesis. McGraw-Hill Book Co., New York, 1962, pp. 485-553.




U.S. GOVERNMENT PRINTING OFFICE: 1969 O-353-813

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
WASHINGTON, D. C. 20546

OFFICIAL BUSINESS

FIRST CLASS MAIL

POSTAGE AND FEES PAID
NATIONAL AERONAUTICS AND
SPACE ADMINISTRATION

POSTMASTER: If Undeliverable (Section 158 Postal Manual) Do Not Return

---

*"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 Notes, and Technology Surveys.

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

**SCIENTIFIC AND TECHNICAL INFORMATION DIVISION**

**NATIONAL AERONAUTICS AND SPACE ADMINISTRATION**

**Washington, D.C. 20546**