CR-128577

(NASA-CR-128517) ADVANCED COMMUNICATION SYSTEM TIME DOMAIN MODELING TECHNIQUES ASYSTD SOFTWARE DESCRIPTION. VOLUME 2: PROGRAM SUPPORT DOCUMENTATION (Systems Associates, Inc.) Aug. 1972 70 p CSCL 17B G3/07

N72-30149

Unclas
39625

SYSTEMS ASSOCIATES

SYSTEMS ASSOCIATES, INC.
444 West Ocean Boulevard
Long Beach, California 90802

72

N72-30149

R72-001
Contract No. NAS9-11743

ADVANCED COMMUNICATION SYSTEM TIME DOMAIN MODELING TECHNIQUES ASYSTD SOFTWARE DESCRIPTION

VOLUME II

PROGRAM SUPPORT DOCUMENTATION

August 1972

Prepared for:

Space Electronics System Division
National Aeronautics Space Administration
Manned Spacecraft Center
Houston, Texas 77058

SYSTEMS ASSOCIATES

444 West Ocean Boulevard
Long Beach, California 90802
Telephone 213/435-8282

TABLE OF CONTENTS

Preceding page blank

TABLE OF CONTENTS (CONT)

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 1.0

INTRODUCTION

1.1 ASYSTD SIMULATION METHODOLOGY

Time domain simulation of systems has classically employed analog computation, mainly in the area of control systems analysis. With the advent of second generation digital computers (IBM 7094) for example, time domain simulation using digital computers advanced rapidly with the development of programs such as MIDAS, MIMIC, DSL/90, CSMP, ECAP, SCEPTRE, etc. These programs, however, were designed with control systems or circuit analysis in mind. Thus, the simulation programs which have been available in the past are not optimized for analysis of telecommunications systems, systems which possess characteristics not encountered in control systems or circuits.

It is only with the availability of large scale digital machines, such as the UNIVAC 1108, that economic considerations favor digital computer simulation as opposed to analog simulation. It will be recognized that the costs of analog versus digital simulation cannot be weighed on a one-to-one basis. For instance, analog simulation requires significant set-up and check-out time for problem initialization, with additional time for modifications to the original situation, but features extremely low unit run costs even for wide-bandwidth systems. In contrast, a digital simulation, using a well designed program, requires minimum set-up time, very limited initial checking, and negligible additional time penalties for parametric or topological variations but at higher hourly costs. Further, degradation of the electronic elements of the analog computer may create large solution error, further limiting inherent analog equipment fidelity. However, for certain situations use of an analog or hybrid computer simulation may be appropriate.

In general, it would appear that use of digital computer simulation techniques offers the most attractive cost-performance characteristics.

Simple acceptance of the worth of the digital computer for system simulation does not lead directly to adoption of an optimum digital simulation technique. Section 1.2 will touch on some of the several digital simulation techniques which are available to the user.

In particular, one unique characteristic of telecommunications systems analysis impacts a time domain simulation badly. This characteristic is the generally large ratio between the RF carrier and the baseband frequencies; i.e., one must compute the system's response for a large number of carrier cycles while allowing propagation of much slower baseband excitation. Appendix A will illustrate the techniques employed in ASYSTD to minimize this difficulty.

1.2 COMPARATIVE ANALYSIS TECHNIQUES

There are several digital computer-aided techniques which can be implemented for telecommunications systems analysis [1] [2] [8] [9] [82] [84] [85] [97]. These include signal power/noise spectral density frequency domain and autocorrelation techniques as well as the time domain approach. The use of the spectral density approach is often of interest for certain limited applications, where the phase characteristic of the telecommunications link elements is not considered, i.e., any spectral density approach must necessarily be insensitive to phase transfer functions. This is a major weakness in systems analysis of high fidelity . . . . The phase properties of real telecommunications links do contribute to performance degradation, and must be considered. The time domain approach does not have this limitation, and in addition, will inherently treat non-linear system element responses in an accurate, straightforward fashion. The common time domain simulation failing of excessive computer run time is offset by the use of a high speed digital computer and the particular transform and carrier-to-baseband frequency translation approach which is employed in the ASYSTD program.

1.3 ANALYSIS

The theoretical basis for the ASYSTD program is discussed in detail in Appendix A. In addition, the extensive bibliography given in this document will illustrate some of the extensive work accomplished in the area of time domain simulation. The techniques employed in the ASYSTD program are those of its predecessor, SYSTID, developed under Contract No. NAS9-10831. The additions have been in the areas of modeling and language program enhancements, in addition to study work in the area of orthogonal transform modeling, error analysis, general filter models, BER measurements, etc. Several models have been developed which utilize the COMSAT generated orthogonal transform algorithms. These models are described in Volume I, Appendix B; however, no attempt to describe the COMSAT algorithms has been made. The analysis results are presented in a series of reports under separate cover. It will be recognized that the ASYSTD program represents a significant improvement over its predecessor, SYSTID, in both the language and model libraries; however, no fundamental change in methodology has been incorporated.

APPENDIX A

THEORETICAL BASIS FOR ASYSTD

A.1 THEORETICAL INTRODUCTION

Direct representation of systems on the digital computer by sample data simulation is a powerful systems analysis technique. Such simulation requires transformation by the computer of continuous system input functions in a manner which characterizes system behavior. The digital computation/process by which this transformation is accomplished is known as a digital filter. This is an algorithm by which sample values of a continuous input function are transformed into sample values of the continuous output function which would result from operating on the input with a given continuous transfer characteristic. The central problem in sample data simulation is obtaining the digital filter algorithm which effects this transformation in the most accurate and efficient manner.

Digital filters may be classified into two major categories as recursive or non-recursive. Non-recursive digital filter outputs depend only on present and previous input samples; recursive filter outputs depend on previous output values as well. The design methods for these two filter types are distinctly different as are their properties. The non-recursive filter has finite memory and excellent phase response characteristics but may require a large number of terms to obtain sharp cutoff properties. The recursive filter has infinite memory but rather poor characteristics. Recursive filters have fewer terms and lend themselves more efficiently to applications requiring sharp cutoff properties. The recursive filter is the digital counterpart of the linear lumped parameter continuous filter. For these reasons, recursive filters are of greater interest in systems analysis by sample data simulation and will be summarized briefly.

If it is assumed that the linear system for which a digital approximation is sought has a transfer characteristic of the form

$$H(s) = \frac{\sum_{m=0}^{M} c_m s^m}{\sum_{n=0}^{N} d_n s^n} \qquad \text{(A-1)}$$

where $s = j\omega$, then the corresponding digital transfer characteristic has the form

$$H^*(z) = \frac{\sum_{j=0}^{N-1} a_j z^{-j}}{1 + \sum_{j=1}^{N} b_j z^{-j}} \qquad \text{(A-2)}$$

where z^{-1} is the unit delay operator. It is assumed that the continuous function H(s) is known or can be determined by established design procedures. The digital filter design problem is thus reduced to determining the coefficients a_j and b_j in $H^*(z)$ such that the continuous filter characteristic H(s) is best approximated for a given number of terms.

One digital filter design technique is based on the standard z-transform, defined so that the impulse response of the digital filter is identical to the sampled impulse response of the corresponding continuous filter. The standard z-transform of H(s) is given by

$$H^*(s) = \sum_{m=-\infty}^{\infty} H(s + jm\,\omega s) \qquad \text{(A-3)}$$

or in terms of the filter impulse response h(t)

$$H^*(z) = T \sum_{\ell=o}^{\infty} h(\ell T) z^{-1} \tag{A-4}$$

where

$s \quad = \sigma + j\omega$

$H(s) \quad =$ Laplace transform of h(t)

$\omega_s \quad = \frac{2\pi}{T}$, radian sampling frequency

$H^*(s) \quad =$ Laplace transform of sampled filter impulse response

$z^{-1} \quad = e^{-st}$, unit delay operator

$H(z) \quad = H^*(s) \big| s = \ln(z)/T$, z-transform of h(t)

For s greater than some critical frequency ω_c, H(s) is assumed to have the form

$$H(s)|_{s>j\omega_c} = K/s^n \tag{A-5}$$

where $n > 0$ and K is a constant.

Equations A-3 and A-4 are the digital filter transfer functions which approximate that of the continuous filter.

The disagreement between the digital filter characteristics provided by the standard z-transform and the continuous filter characteristic in the baseband ($-\omega s/2 \le \omega \le \omega s/2$) is known as frequency aliasing error and results from terms of the form $H(s + jm\omega_s)$, $m \neq 0$. This disagreement is present whenever the continuous filter characteristic is not bandlimited to the baseband. Unfortunately this is the case for most lumped parameter systems, for which H(s) is a rational function of s. Thus, for physical systems of interest, the standard z-transform yields $H^*(s) \neq H(s)$ in the baseband and aliasing error is present to some degree.

For higher order continuous filter transfer functions (n in Equation A-5 is large) having a critical frequency ω_c much less than the sample frequency ω_s, aliasing error is sufficiently small that the standard z-transform yields useful results. In many practical situations, however, neither of these conditions are met. In these cases, the standard z-transform results in prohibitive aliasing errors in the digital filter frequency characteristic.

Frequency aliasing error may be avoided if digital filters are designed by means of an artifice known as the bi-linear z-transform. The bi-linear z-transform maps the entire complex s plane into an s_1 plane bounded by the lines $s_1 = j\omega_s/2$ and $s_1 = -j\omega_s/2$. The bi-linear z-transform is defined by

$$s = \frac{2}{T} \tanh \frac{s_1 T}{2} \qquad \text{(A-6)}$$

where $s_1 = j\omega_s/2$ and T is the sample interval. This becomes upon substitution of the unit delay operator $z^{-1} = e^{-s_1 T}$,

$$s = \frac{2}{T}\left(\frac{1 - z^{-1}}{1 + z^{-1}}\right) \qquad \text{(A-7)}$$

The digital filter transfer function, $H^*(z)$ is determined by substituting the bi-linear z-transform into the continuous filter transfer function H(s),

$$H^*(z) = H(s)\Big|_{s = \frac{2}{T}\left(\frac{1 - z^{-1}}{1 + z^{-1}}\right)} \qquad \text{(A-8)}$$

One aspect of the digital filter so obtained is that a non-linear warping is imparted to its frequency scale in accord with the transformation

$$\frac{\omega T}{2} = \tan \frac{\omega_1 T}{2} \tag{A-9}$$

This transformation is depicted in Figure A-1 which plots normalized warped frequency ω, versus normalized unwarped frequency ω. Frequency warping is not a significant constraint on the versatility of the bi-linear z-transform. The warping may be arbitrarily reduced by making the sample frequency ω_s high compared to the critical frequency ω_c of the continuous filter. Furthermore, frequency warping may be compensated for by prewarping the critical frequencies of the continuous filter so that transformed frequencies will be shifted back to the desired ones. Because it obviates inaccuracies due to aliasing error, the bi-linear z-transform is

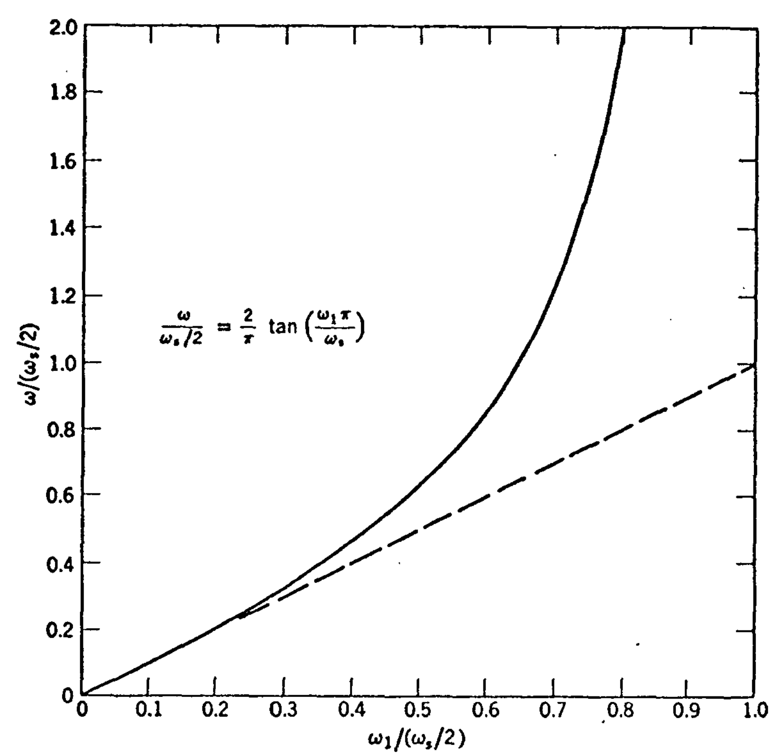


Figure A-1. Non-Linear Warping of the Frequency Scale in the Bi-Linear z-Transformation a most appealing digital filter design technique. It is applicable to low-pass, band-pass, band-stop, and other continuous filters whose magnitude characteristics are essentially constant within successive pass and stop bands.

A.2 APPLICATIONS OF SAMPLED DATA TECHNIQUES IN THE ASYSTD PROGRAM

A great diversity of systems are amenable to analysis by sample data simulation. Telemetry links may be modeled with any combination of amplitude or angle modulation, phase-locked loops may be simulated either separately or as part of a more complex system, transient response of filters to various input waveforms can be accurately determined, or recorded data may be processed by digital filtering.

The existing extensive SAI library of sample data simulation programs[1] is designed for maximum flexibility. As an example of sample data simulation, consider the following analysis of a radio frequency communications link.

A sample data simulation of an RF link can be implemented by a sampled carrier and any of several modulation and demodulation schemes. In this case, however, the sample frequency ω_s must be high enough that frequency warping due to the bi-linear z-transform remains within acceptable limits. Since the carrier frequency is typically many orders of magnitude higher than the baseband frequency, a correspondingly higher sample rate is indicated by Figure A-1, resulting in excessive computer run time.

A technique has been devised whereby carrier frequency can be eliminated from the system representation. This permits greatly reduced sample rate and correspondingly shorter computer run time without degrading the carrier frequency band-pass filter characteristics.

---

[1]The SAMDAT program library, for instance.

Consider the following system:

$$x(t) \longrightarrow \boxed{h(t)} \longrightarrow y(t)$$

where

$$x(t) \;=\; A(t)e^{j\left[\omega_c t + \phi(t)\right]} + n(t)$$

$$y(t) \;=\; A_0(t)e^{j\left[\omega_c t + \phi_0(t)\right]}$$

$n(t) \;=\;$ Additive noise

$h(t) \;=\;$ Impulse response of the filter

The input is a general modulated signal with additive noise. The output is a complex signal represented by an amplitude and a phase.

The actual signal inputs and outputs are the real parts of those given. In accordance with Reference 1, for amplitudes and phases to be determined by the magnitudes and phases of the complex signals, all inputs must be analytic signals, i. e., the imaginary parts must be the Hilbert transform of the real parts. This condition is satisfied on the signal term in $x(t)$ if the frequency spectrum of $A(t)e^{j\phi(t)}$ is essentially zero at the frequence ω_c; however, the noise term $n(t)$ must also be an analytic signal.

Assume $Re\left[n(t)\right]$, the actual noise term, is Gaussian with a frequency spectrum which is symmetric about ω_c. (This does not imply that the band-pass filter has symmetric response about ω_c.) The real part of the noise term can be written as

$$Re\left[n(T)\right] \;=\; n_1(t)\cos\omega_c t - n_2(t)\sin\omega_c t \qquad \text{(A-10)}$$

The two quantities $n_1(t)$ and $n_2(t)$ will be independent and Gaussian, will have identical frequency spectra equal to the original spectrum translated to dc, and will each have variance equal to that of $\mathrm{Re}[n(t)]$. The imaginary part of n(t) must be the Hilbert transform of this quantity. As long as the frequency spectrum of $n_1(t)$ or $n_2(t)$ is essentially zero at ω_c, n(t) is given by

$$n(t) = \left[n_1(t) + jn_2(t)\right]e^{j\omega_c t} \tag{A-11}$$

The system now is represented by the following convolution equation.

$$A_0(t)e^{j\phi_0(t)}e^{j\omega_c t} = \int_0^\infty h(\mu)\left[A(t-\mu)e^{j\phi(t-\mu)} + n_1(t-\mu) + jn_2(t-\mu)\right]e^{j\omega_c(t-\mu)}d\mu \tag{A-12}$$

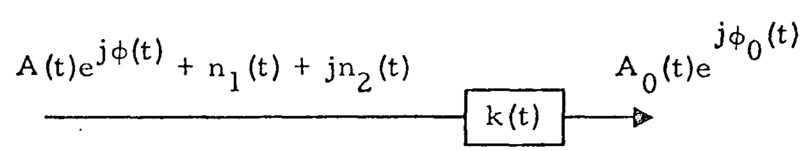
After rearranging factors and removing the carrier term from the integral, the above equation becomes

$$A_0(t)e^{j\phi_0(t)} = \int_0^\infty k(\mu)\left[A(t-\mu)e^{j\phi(t-\mu)} + n_1(t-\mu) + jn_2(t-\mu)\,d\mu\right] \tag{A-13}$$

where

$$k(t) \equiv h(t)e^{-j\omega_c t}$$

Equation A-13 implies that our system can be simulated by the following one having the same output and input except that the carrier frequency has been eliminated



In order to simulate the new system it will be convenient to determine $K_r(s)$ and $K_i(s)$ where these are the Laplace transforms of $k_r(t)$ and $k_i(t)$, the real and imaginary parts of $k(t)$. Also, because the subroutine used to set up the simulation first removes the center frequency phase from $H(s)$, it will be convenient to define

$$k(t) \equiv h(t)e^{-j(\omega_c t + \psi)} \tag{A-14}$$

where ψ is a constant phase.

The two systems will now be the same except for the constant phase ψ.

It may be seen that

$$k_r(t) = h(t)\cos(\omega_c t + \psi)$$

$$k_i(t) = -h(t)\sin(\omega_c t + \psi) \tag{A-15}$$

Taking Laplace transforms and applying Euler's formulas to the sin and cos yields:

$$K_r(s) = \frac{1}{2}\int_0^\infty h(t)\left[e^{j(\omega_c t+\psi)} + e^{-j(\omega_c t+\psi)}\right]e^{-st}\,dt$$

$$K_i(s) = \frac{-1}{2j}\int_0^\infty h(t)\left[e^{j(\omega_c t+\psi)} - e^{-j(\omega_c t+\psi)}\right]e^{-st}\,dt \tag{A-16}$$

Equation A-15 becomes

$$K_r(s) = \frac{1}{2}\left[e^{-j\psi}H(s + j\omega_c) + e^{j\psi}H(s - j\omega_c)\right]$$

$$K_i(s) = \frac{1}{2j}\left[e^{-j\psi}H(s + j\omega_c) - e^{j\psi}H(s - j\omega_c)\right] \qquad \text{(A-17)}$$

H(s) is a rational fraction in s. If the real or imaginary parts of e^{-j} $H(s + j\omega_c)$ are taken with respect to the coefficients of s, not s itself, Equation A-17 can be rewritten as

$$K_r(s) = \mathrm{Re}\left[e^{-j\psi}H(s + j\omega_c)\right]$$

$$K_i(s) = \mathrm{Im}\left[e^{-j\psi}H(s + j\omega_c)\right] \qquad \text{(A-18)}$$

or the entire function

$$K(s) = e^{-j\psi}H(s + j\omega_c) \qquad \text{(A-18A)}$$

If H(s) represents a band-pass filter, K(s) will have band-pass regions about the origin and about the frequency $-2\omega_c$. The response at $-2\omega_c$ is not important since the input signals should not have spectral components there. As indicated in the discussion of digital filter designs, aliasing error is eliminated by the bi-linear z-transform. Thus, the sample frequency can be selected in accordance with the bandwidth of the filter, not its center frequency.

If $A(t)e^{j\phi(t)}$ has frequency components greater than one-half the sample frequency, aliasing error in the signal will occur although the filter will be represented correctly. Also if $A(t)e^{j\phi(t)}$ has a non-zero frequency spectrum at $\omega_1 = 2/T \tan \omega_c T/2$, which corresponds to the center frequency shifted by the bi-linear z-transform, a ripple in the output amplitude and phase at frequencies about $2\omega_1$ may occur. This is because the assumption that $A(t)e^{j\phi(t)}\ e^{j\omega_c t}$ is an analytic will not be true. Some ripple for instance may be detected if A(t) is a step which has a frequency response which rolls off slowly at higher frequencies.

A sample data demodulation can now be implemented quite simply, following the band-pass filter. Denote the output of the second system as

$$A_0(t)e^{j\phi_o(t)} = a(t) + jb(t) \tag{A-19}$$

Then

$$A_o^2(t) = a^2(t) + b^2(t) \tag{A-20}$$

and the output phase is given by

$$\phi_o(t) = \tan^{-1} \frac{b(t)}{a(t)} \tag{A-21}$$

In an FM system, $\mu_o(t)$, the instantaneous frequency, is of interest, rather than the phase. Whenever $A_o(t)$ does not pass through zero, $\mu_o(t)$ is given by

$$\mu_o(t) = \frac{d\phi_o(t)}{dt} = \frac{a(t)\frac{db(t)}{dt} - b(t)\frac{da(t)}{dt}}{A_o^2(t)} \tag{A-22}$$

$A_o(t)$ by definition is always positive. The sign information which would normally be associated with $A_o(t)$ is implied by the phase. This means that a sign reversal in $A_o(t)$ would be manifested by $A_o(t)$ going to zero between two successive sample points and the phase undergoing a step change of π radians. This sign reversal can be detected by sensing when both a(t) and b(t) change signs from one sample point to the next. The step change in phase can be implemented by causing $\mu_o(t)$ to contain an impulse function of magnitude $\pm\pi$, which in a sample data system is simply one point of magnitude $\pm\pi/T$. The sign of the impulse is selected alternately as plus and minus. This is correct because if $A_o(t)$ had passed from positive to negative at some point in time it must next pass from negative to positive.

Equation A-11 suggests that Gaussian noise may be inserted, less the carrier component, at the input to the RF frequency filter in the following form:

$$n(t) = n_1(t) + jn_2(t) \tag{A-23}$$

The terms $n_1(t)$ and $n_2(t)$ were assumed to have identical power spectra and were assumed to be statistically independent. They were also assumed to have a power spectrum which is essentially zero at the carrier frequency ω_c.

The last assumption requires that n(t) be bandlimited by a filter prior to its insertion into the link. The possibility of relaxing this requirement for certain simulations avoids the use of an extra filter.

Let $n_1(t)$ and $n_2(t)$ be generated as pseudo-random numbers, independent, and with Gaussian statistics by a computer random number subroutine. Let these numbers be of zero mean and standard deviation σ. Because they are independent they can be assumed to be samples of a noise process whose spectrum is uniformly distributed between $-f_s/2$ to $f_s/2$ so that the two-sided power spectral density η is given by

$$\eta = \frac{\sigma^2}{f_s} \tag{A-24}$$

where f_s = sample frequency. Now assume this signal is passed through a sample data bandlimiting filter so that its power spectrum in the interval $-\omega_s/2 \le \omega \le \omega_s/2$ becomes

$$S_1(\omega) = \frac{\sigma^2}{f_s} \left| H_1\left(j\frac{2}{T} \tan \frac{\omega T}{2}\right)\right|^2 \tag{A-25}$$

The spectrum $S_1(\omega)$ is now passed through the band-pass filter transfer function $H(j\omega)$ translated to zero and distorted in frequency response to produce the output spectrum

$$S(\omega) = \frac{\sigma^2}{f_s} \left| H_1\left(j\frac{2}{T} \tan \frac{\omega T}{2}\right) H\left(j\frac{2}{T} \tan\frac{\omega T}{2} + j\,\omega_c\right)\right|^2 \tag{A-26}$$

Without bandlimiting by $H_1(j\omega)$, $S(\omega)$ will have the same shape as $\left| H(j2/T \tan \omega T/2 + j\omega_c)\right|^2$, i.e., it will have the desired band-pass shape about dc and in addition will have a more narrow band-pass shape about the frequency $-\omega_{c2} \equiv -2/T \tan^{-1} \omega_c T$. The analytic signal assumption, because of the frequency warping, is equivalent to setting $S(\omega) = 0$ for $\omega < -\omega_{c1} \equiv -2/T \tan^{-1} \omega_c T/2$ so that any noise power below $-\omega_{c1}$ represents an error in the simulation.

Let N denote the total noise power above $-\omega_{c1}$ and N_e denote that below. N will then be the noise power which should normally be used in computing carrier to noise ratio. N_e, the extra noise added by the simulation, will normally affect the outputs at frequencies which are not of interest and may later be filtered out. However, it is important to realize that N_e will affect threshold in the system and must be kept small whenever the simulation is to function near or below threshold.

The quantities N and N_e can be derived by integrating Equation A-26 over the proper regions to obtain

$$N = \sigma^2 \frac{f_b}{f_s}$$

$$N_e = \sigma^2 \frac{f_{be}}{f_s} \qquad \text{(A-27)}$$

where

$$f_b = \frac{1}{2\pi} \int_{-\frac{2}{T} \tan^{-1} \frac{\omega_c T}{2}}^{\omega_s/2} \left| H_1\left(j\frac{2}{T} \tan \frac{\omega T}{2}\right) H\left(j\frac{2}{T} \tan \frac{\omega T}{2} + j\,\omega_c\right) \right|^2 d\omega \qquad \text{(A-28)}$$

$$f_{be} = \frac{1}{2\pi} \int_{-\omega_s/2}^{-\frac{2}{T} \tan^{-1} \frac{\omega_c T}{2}} \left| H_1\left(j\frac{2}{T} \tan \frac{\omega T}{2}\right) H\left(j\frac{2}{T} \tan \frac{\omega T}{2} + j\omega_c\right) \right|^2 d\omega \qquad \text{(A-29)}$$

Equations A-28 and A-29 can be modified by a change in variables to the following equations.

$$f_b = \frac{1}{2\pi} \int_0^{\infty} \left| H_1 (j\omega - j\omega_c)\, H(j\omega) \right|^2 \frac{d\omega}{1 + \left(\frac{(\omega - \omega_c)T}{2}\right)^2} \qquad \text{(A-30)}$$

$$f_{be} = \frac{1}{2\pi} \int_{-\infty}^{0} \left| H_1 (j\omega - j\omega_c)\, H(j\omega) \right|^2 \frac{d\omega}{1 + \left(\frac{(\omega - \omega_c)T}{2}\right)^2} \qquad \text{(A-31)}$$

From Equation A-30 it can be seen that if $H_1(j\omega) \approx 1$ over a range equal to the passband of $H(j\omega)$ and if $\omega - \omega_c \ll \omega_s/2$, f_b is the equivalent noise bandwidth of the modulation link filter. An estimate of the ratio of N_e to N can be obtained by assuming $H(j\omega)$ is narrowband. This estimate is given below.

$$\frac{N_e}{N} \approx \left| \frac{H_1(j2\omega_c)}{H_1(j0)} \right|^2 \frac{1}{1 + \left(2\pi \frac{\omega_c}{\omega_s}\right)^2} \qquad \text{(A-32)}$$

If the system requires that the ratio of N_e/N be small this may be achieved if the carrier frequency is much higher than the sample frequency, or it can be assured by bandlimiting the noise by a low-pass filter $H_1(j\omega)$.

APPENDIX B

PROGRAM DESCRIPTION

B. 1 ASYSTD SIMULATION METHODOLOGY

B. 1. 1 Terminology

The ASYSTD program is a two-phase processor consisting of a language processor (translator) and a library. When coupled with the UNIVAC EXEC II system, ASYSTD becomes an easily used system simulation program entirely user-oriented.

The ASYSTD processor accepts input decks written in the ASYSTD language representative of systems or models to be used in systems. ASYSTD then generates a symbolic FORTRAN V program (for systems) or subroutine (for models) which is then compiled by the FORTRAN V compiler. If the input describes a model, a temporary entry is made in the library dictionary for subsequent use. In either case, the symbolic routines are written into the Program Complex File (PCF) for subsequent use and interfacing with EXEC II. The ASYSTD simulation program functional flow diagram, illustrating these interactions, is depicted in Figure B-1.

A SYSTEM is a complete program which is to be executed to simulate a specific telecommunications link; i. e., is a model prefixed with system parameter definitions, input/output specifications, and post-processing declarations.

A MODEL is a subroutine which simulates the properties of a device in a telecommunications link. A MODEL is characterized by defining its topology and components.

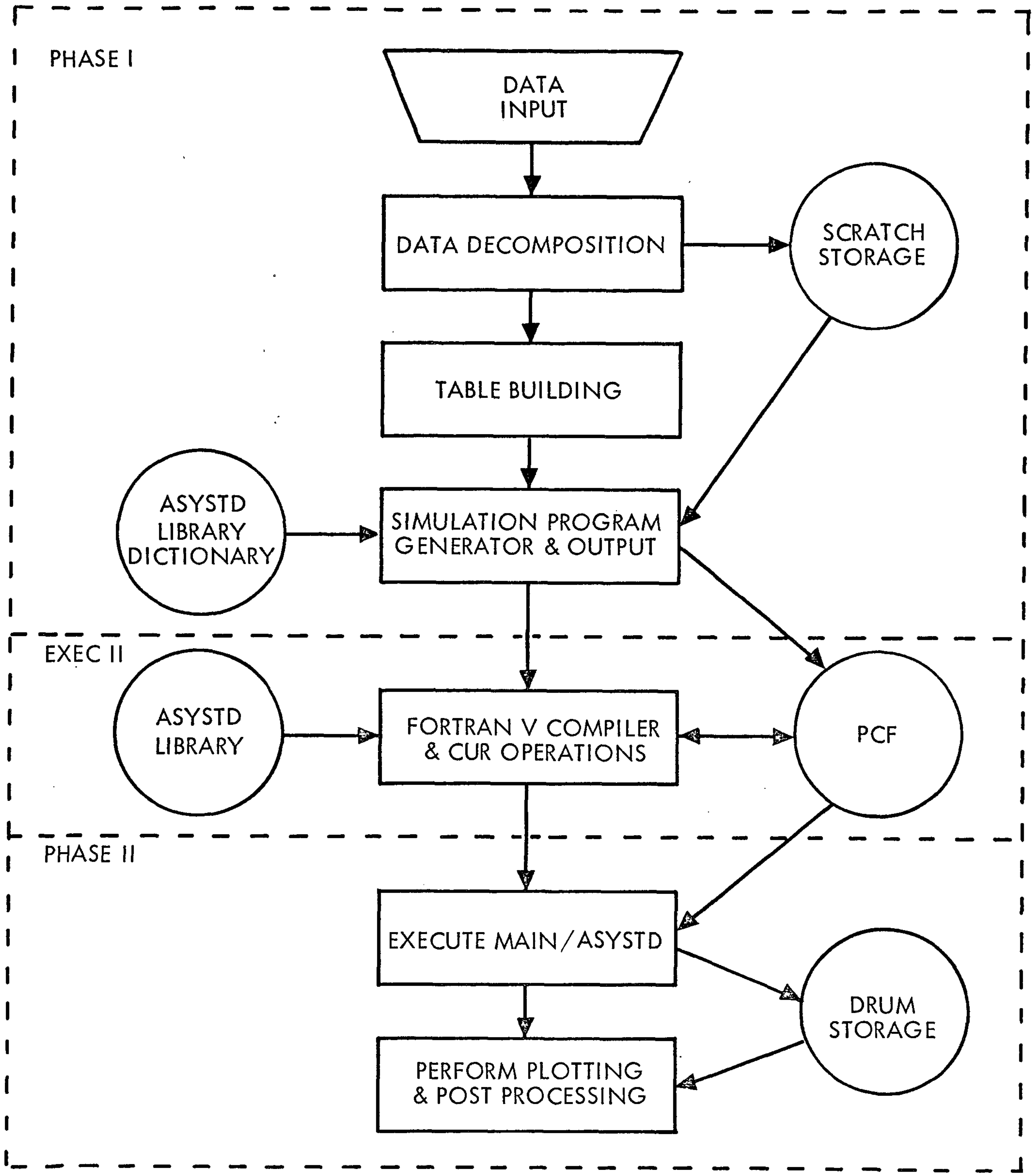


Figure B-1. ASYSTD Functional Flow

The following will describe both the language processor and the sample-data techniques as applied to telecommunications analysis and simulation.

B.1.2 ASYSTD Language Processor

The ASYSTD Language Processor translates topological descriptions into a procedural method of solution based upon a fixed algorithm. This algorithm is best depicted by Figure B-2, the logical structure of input decomposition.

The input data for a model or system is decomposed into a set of linked tables which define the topological characteristic of the input. Once the tables are constructed, they are systematically scanned, starting at the left node name table entry "INPUT", until all expressions are solved up to and including the right node table entry "OUTPUT". The signal progress convention is from left node to right node; left being the input. Taps, depending on context, can be either inputs or outputs. The output of the device is passed to the next device through their common node.

The linked tables of Figure B-2 are built and searched by the routine depicted in Figure B-3. Whenever a tap is referenced by an expression, the "tap table" provides the necessary information for the reference. All addressing is relative to the input node of the model or system, at all levels. That is, all models generated by the ASYSTD language processor become re-entrant, and each reference is independent of any other reference to the same model. With models referencing other models several times, this factor is of prime importance. The relative addressing is performed by a floating index which is set to the next available location by each model routine as it is entered so as to provide for its own storage requirements.

B.1.3 Sampled Data Modeling

The technique utilized in the ASYSTD library for simulating continuous systems is the bi-linear z-transformation. The major advantage over the standard z-transform is that aliasing errors are eliminated,

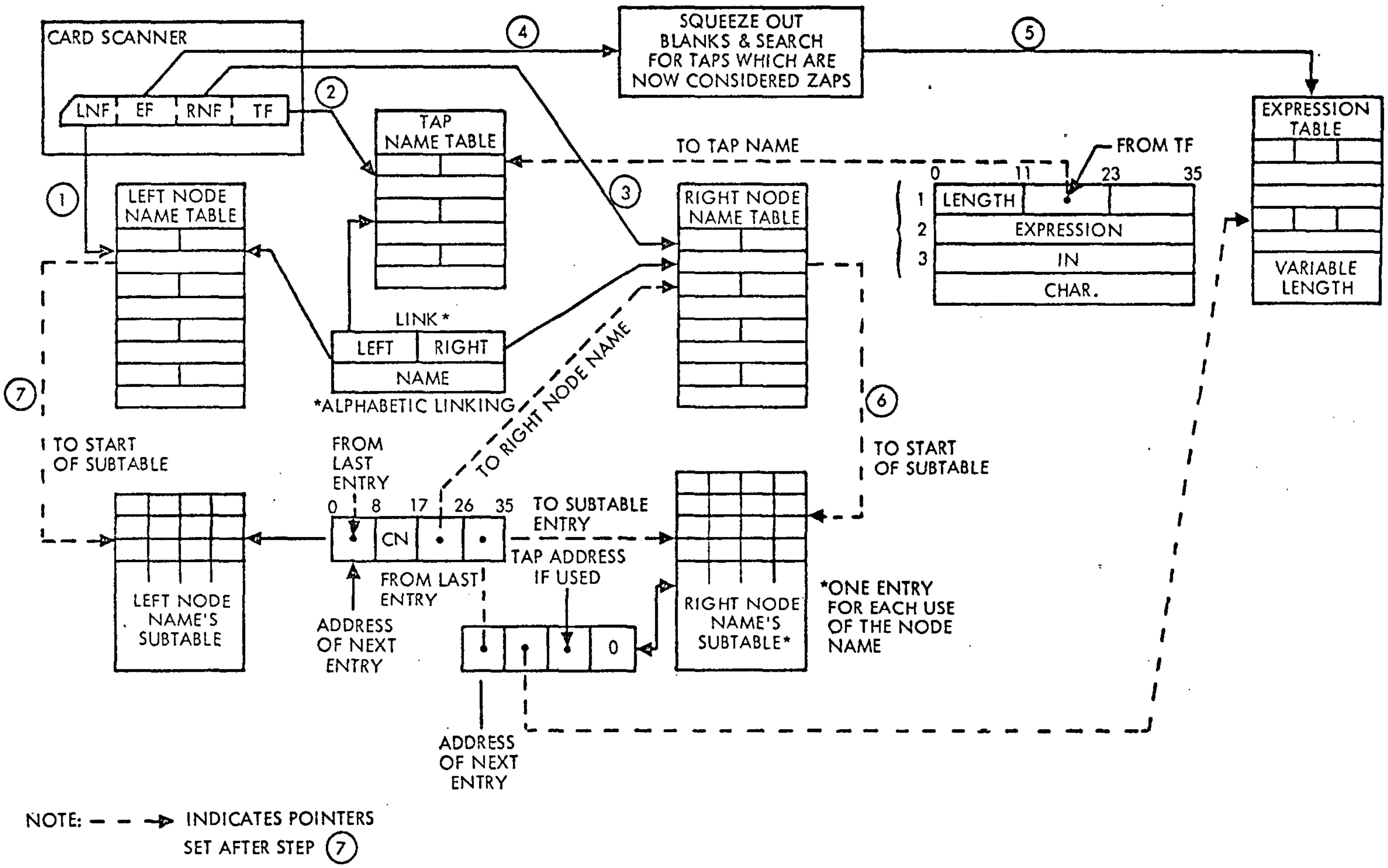


Figure B-2. ASYSTD Logical Structure of Input Decomposition

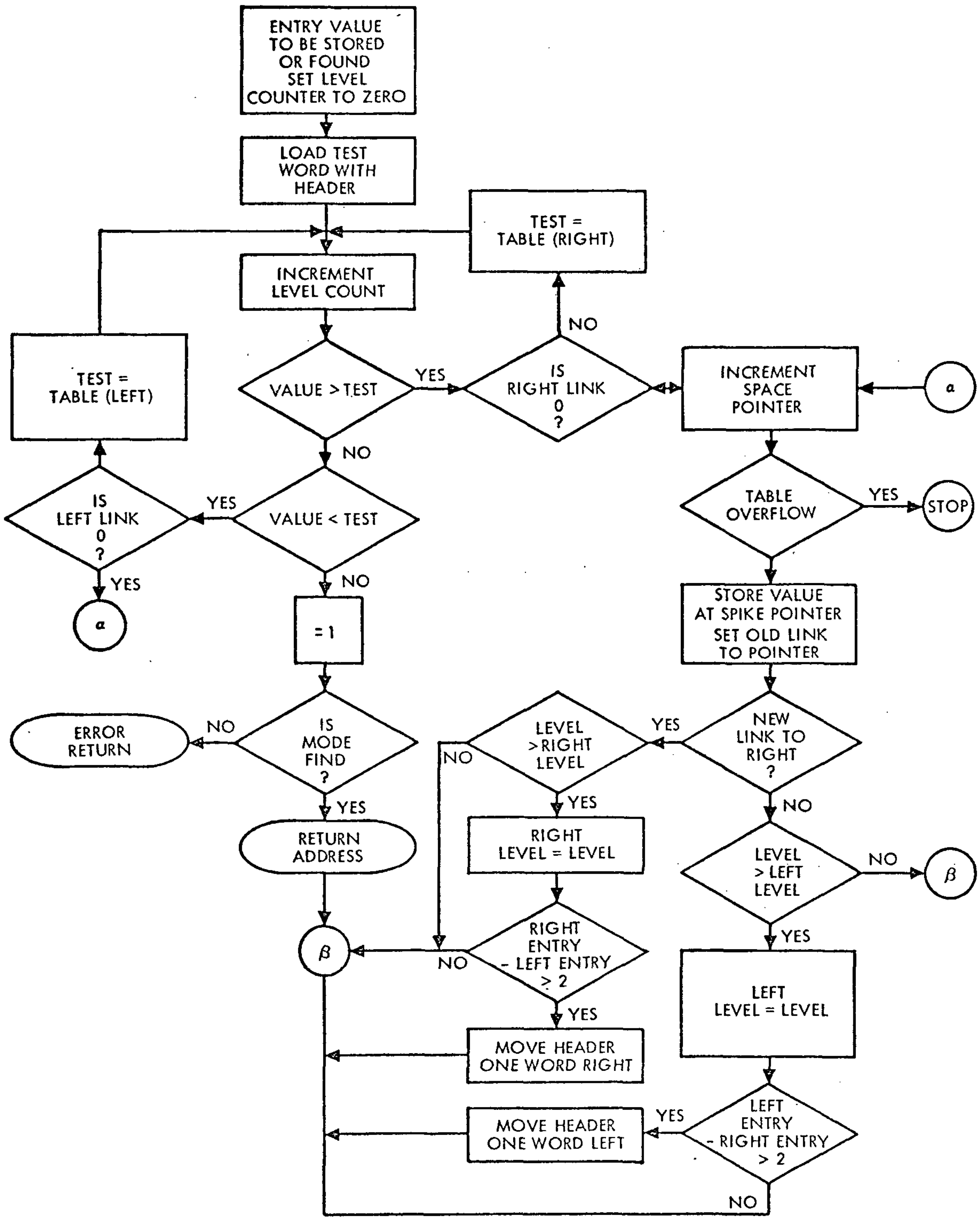


Figure B-3. ASYSTD Linked Table Search Routines making possible the realization of commonly encountered functions whose response does not approach zero for high frequencies (e.g., high-pass, band-stop) and allowing the reduction of required sampling frequency. Note that aliasing of the signals, however, is possible if the sampling rate is too low.

When representing an RF link, the ability to model in the baseband region is significant when considering computer run times. Use of the sampled-data technique coupled with a translation process for all RF components provides a reduction in computation time grossly given by the ratio of baseband frequency to RF frequency. The complication, of course, is that the translation results in complex representations of all signals and components; the key system component being continuous functions or filters. Appendix A presents the mathematical development of such a process. When given an RF system, the ASYSTD library will translate and maintain the integrity of any continuous transfer function. When translating from a carrier frequency (ω_c) to baseband, a band-pass function will have band-pass regions about the frequency origin and $-2\omega_c$. The response at $-2\omega_c$ should not be of importance since the baseband signal should be analytic (i.e., the imaginary part must equal the Hilbert transform of the real part). This means the spectrum of the baseband signal should be zero at $\pm\omega_c$.

It is noted that aliasing of the baseband signal will occur if the signal has frequency components greater than one-half the sample (Nyquist) frequency. Also, if the signal is not analytic, a ripple in the output amplitude and phase at $2\omega_1$ $\left(\omega_1 = 2/T\left\{\tan\left[\omega_c T/2\right]\right\}\right)$ will occur.

The techniques discussed here have been utilized in the past with great success, resulting in the SAI SAMDAT library.

B.2 SUBROUTINE DESCRIPTION

The following is a list of all subroutines and procedures utilized by ASYSTD. A brief synopsis of each subroutine's function is also given.

B.2.1 ABORT

This routine terminates PHASE I if there is an I/O error while reading in cards.

B.2.2 ASGTAP

This routine keeps track of the tap subscripts used in the model library.

B.2.3 ASSIGN

This routine assigns the V-array subscripts to nodes. If the node was previously assigned a subscript, ASSIGN merely returns the value. If the node is in both right-hand and left-hand node tables, the subscript increment is stored in both tables (in H2 of the first subword under a node name).

B.2.4 COMPIL

This routine controls generation of the body of the model in terms of its topology and appropriate generation of describing code of the proper sequence.

B.2.5 CREATE

This routine creates the model and main elements in the program complex file (PCF), and also the preamble code for models and systems. (The ASYSTD processor presently can create one FORTRAN file at a time . . . future updates may include multiple file capability.)

B.2.6 DATIN

This routine gets the date and the time of day.

B.2.7 EDIT

This routine edits the input expression and replaces the $ characteristics with the appropriate V-array subscripted variable. It also edits the delimiter set into a FORTRAN-acceptable set.

B.2.8 EDIT1

This routine scans the expression to find model, function, and variable names. When a name is found the various tables are examined to determine the type of expression.

B.2.9 EINES

This routine maintains a linked subtable under each entry in the left node table, by means of linear linking.

B.2.10 EQUNIT

This routine establishes correspondence between a FORTRAN unit and an opened element in the PCF.

B.2.11 ERECTS

This routine generates the various FORTRAN statements necessary to reference a model or function.

B.2.12 ERRMSG

This routine returns the error message and its length in words, given an error number.

B.2.13 GREBE

This routine maintains a non-linked table of the expressions found between node names on the input deck.

B.2.14 GWIN

This routine processes the ASYSTD input language. It constructs the tables and lists necessary for the generation of the output FORTRAN program.

B.2.15 INCLUD

This routine generates "include main list" to include the canned programs for use in the main program.

B.2.16 LIB 003

This routine reads in the library element and makes its entries into the library search program.

B.2.17 LIB 004

This routine is the library, consisting of a linked table of model names and associated descriptors. It is initially loaded with entries in the "library" element.

B.2.18 LISTIT

This routine lists the input deck and diagnostic edited into proper position before the error line.

B.2.19 LIT

This routine converts an integer to its BCD form.

B.2.20 LUCHT

This routine maintains the tree of the left node names.

B.2.21 MOGUE

This routine generates a data statement for the "DEFAUL" values.

B.2.22 NADINE

This routine reads in the input deck and copies it on storage drum.

B.2.23 NAMLST

This routine generates a namelist statement to read in the variables in the data statement.

B.2.24 NEXECS

This routine generates non-executable statements associated with printing and post-processing.

B. 2. 25 OUTPUT

This routine manages the saving of values specified in PRINT, PLOT, PPLOT, and POST statements. OUTPUT also generates the code to print and save these values on drum as well as the calls to the post-processing routines.

B. 2. 26 PCF/ISD

This routine performs all PCF manipulations on the SAI 1108 system.

B. 2. 27 PCF/MSC

This routine performs all PCF manipulations on the MSC 1108 system.

B. 2. 28 PREAMB

This routine generates the preamble or subroutine entry point for the "Model" programs.

B. 2. 29 QUEUE

This routine maintains a list of "Active" nodes, i. e., nodes which have appeared in the right node field, but which have not been processed yet.

B. 2. 30 RECTUS

This routine maintains the tree of the right node names.

B. 2. 31 SETUP

This routine is called initially and after the completion of processing on each model or system, it initializes all the necessary pointers and zeroes the appropriate tables, and loads the work area with the next batch of cards from the input buffer.

B. 2. 32 SPORU

This routine maintains the linked TAP tables.

B. 2. 33 <u>ASYSTD</u>

ASYSTD is the main program.

B. 2. 34 <u>TAPONE</u>

This routine assigns each tap a number, beginning with 1 and increasing sequentially.

B. 2. 35 <u>TAPTWO</u>

This routine assigns output tap V-array subscript locations and generates the necessary FORTRAN statements.

B. 2. 36 <u>TAP 3</u>

This routine generates the V-array subscripted variable literals for taps in expression.

B. 2. 37 <u>THOR</u>

This routine tests for legitimate FORTRAN name, and if not, returns an error message.

B. 2. 38 <u>TITLE</u>

This routine loads the SYSTEM comment into an array for use by the output routines.

B. 2. 39 <u>USRELT</u>

This routine is an SAI 1108 system routine used for the creation of PCF element by users. It is used by the SAI version of ASYSTD.

B. 2. 40 <u>V</u>

This routine generates the literal V(z + nn) where z is the input bias.

B. 2. 41 <u>ZWEI</u>

This routine maintains a linked subtable below each mode entry in the right node table (RECTUS), with linear linking.

B. 2. 42 DRUM

This procedure defines all parameters associated with saving variables on drum.

B. 2. 43 NTAB$

Used only on SAI 1108 system.

B. 2. 44 PARAM

This procedure defines the parameters used in ASYSTD for fixing program size.

B. 2. 45 PROCO5

This procedure defines several procedures utilized throughout the ASYSTD processor for listing processing and linked table manipulation.

B. 2. 46 QUARTR

This procedure defines the quarter-word functions used in ASYSTD.

B. 2. 47 SUB 1

This procedure defines some functions necessary in the linking routines.

APPENDIX C

THEORETICAL BASIS FOR FILTER MODELS

C. 1 TRANSFER FUNCTION RESPONSE FROM POLE-ZERO LOCATION

A general transfer function may be expressed in the following form:

$$H(s) = \frac{A \prod_{i=1}^{M} (s - Z_i)}{\prod_{i=1}^{N} (s - P_i)} \tag{C-1}$$

where A is some constant multiplier

$s = j\omega =$ complex frequency

$P_i =$ Complex Pole $(s + j\omega_i)$

$Z_i =$ Complex Zero $(s + j\omega_i)$

$N =$ Order of the filter (number of poles)

$M =$ Number of zeros

The poles and zeros are always either complex conjugate pairs or single real values. The roots of most of the functions of interest consist entirely of conjugate pairs, if even, and have one additional real root, if odd.

The computation takes advantage of the computer's ability to do complex arithmetic. The response at a particular frequency (ω_a) is obtained by substituting into Equation C-1 which yields

$$H\,(j\omega_a) = \frac{A \prod_{i=1}^{M} (j\omega_a - Z_i)}{\prod_{i=1}^{N} (j\omega_a - P_i)} = \alpha + j\beta \tag{C-2}$$

The complex result has a magnitude and phase response given by the following expressions

$$\begin{aligned} |H(j\omega_a)| &= \sqrt{\alpha^2 + \beta^2} = \text{Magnitude Response} \\ \text{Tan}^{-1}\,[\beta/\alpha] &= \text{Phase Response} \end{aligned} \tag{C-2A}$$

The multiplicative constant, A, is used most often in the program as a normalizing factor. In general, it is desired to make the response be unity at DC.[1] This is accomplished if A is computed as

$$A = \frac{\prod_{i=1}^{N} |P_i|}{\prod_{i=1}^{M} |Z_i|} \tag{C-3}$$

All that is needed then is to determine the poles (and zeros for the elliptic function) of the different types of filters. Since each filter is derived differently, the roots of each are found differently and a section is devoted to each type. The poles found are for the normalized low-pass (1 rad/sec bandwidth). The response for high-pass, etc., will be discussed in the section on transformations.

---

[1]For low-pass functions, otherwise at center frequency for band-pass functions.

C.2 GENERATION OF FILTER PROTOTYPE TRANSFER FUNCTIONS

C.2.1 Ideal Filter

An ideal (or zonal) filter is defined as one which has unity gain and linear phase over some bandwidth f_B. This filter has not been included in the Filter Subroutine. Its characteristics are given in Figure C-1.

C.2.2 Butterworth Filters

A Butterworth or maximally flat amplitude filter has a magnitude response given by

$$|H(j\omega)|^2 = \frac{1}{1 + \omega^{2N}} \qquad \text{(C-4)}$$

where N is the order of the filter.

This is an approximation to an ideal low-pass as shown in Figure C-2. The higher the order, N, the nearer the filter response approaches the ideal.

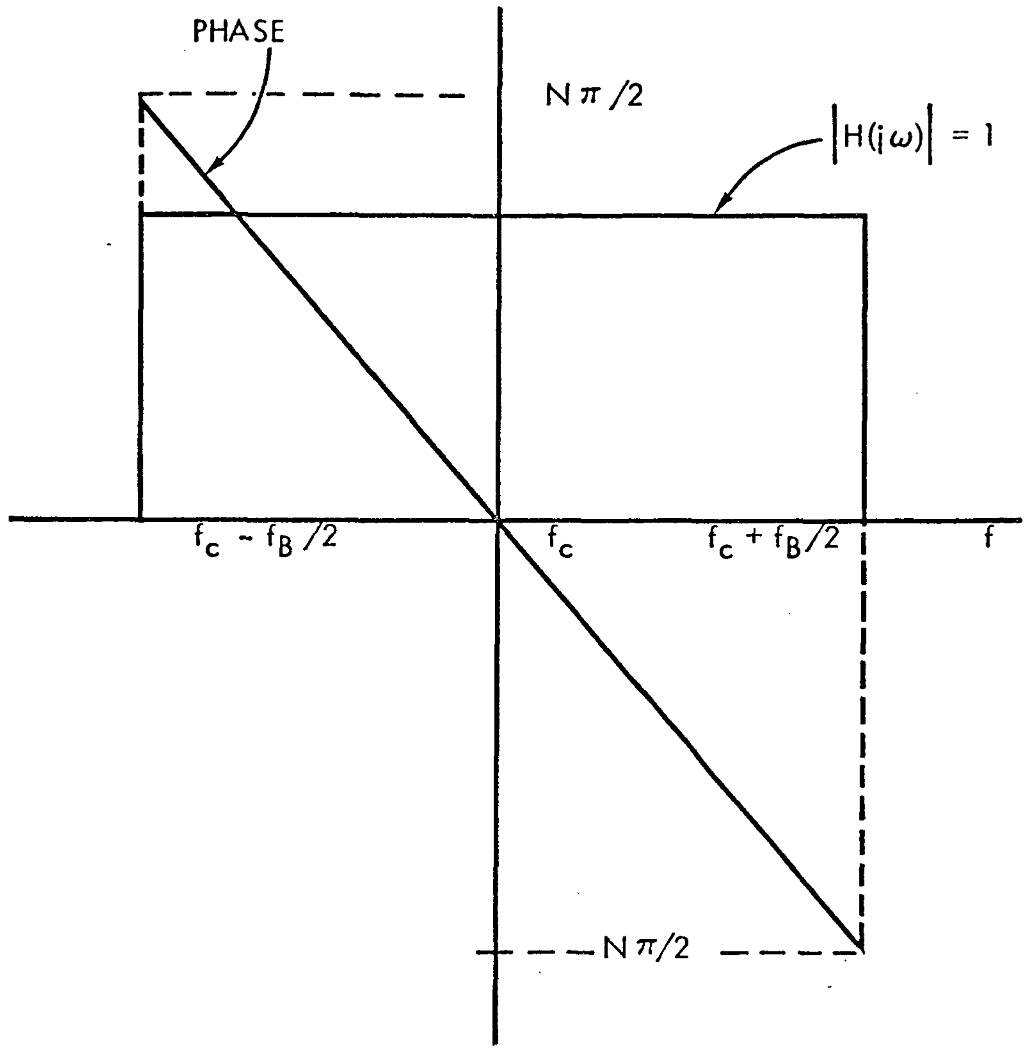
The transfer function may be obtained by substituting $s^2 = -\omega^2$.

$$|H(s)|^2 = H(s)\,H(-s) = \frac{1}{1 + s^{2N}(-1)^N} \qquad \text{(C-5)}$$

The roots of Equation C-5 are given by the expression in Equation C-6

$$s_i = \exp\left[j\pi\left(\frac{N + 1 + 2k}{2N}\right)\right], \; k = 0, \; 2N-1 \qquad \text{(C-6)}$$

and determine the poles of the filter function. The poles of $H(s)^2$ will be equally spaced on a unit circle in the complex frequency plane. Those belonging to H(s) will be only in the left-half plane. The pole locations are



NOTE THAT THE TIME DELAY IS $N/2f_B$

Figure C-1. Amplitude and Phase Response for Ideal (Zonal) Filter

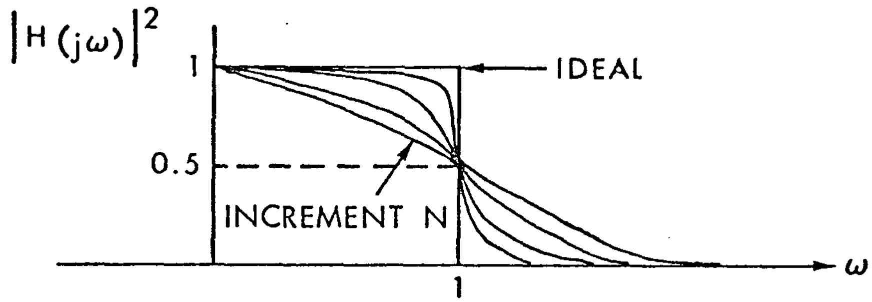


Figure C-2. Butterworth Response Approximation to an Ideal Response

illustrated for odd and even order Butterworth functions in Figures C-3(a) and C-3(b), respectively. The equations for the poles for both odd and even order are given by Equations C-7 and C-8.

Odd:

$$\left.\begin{matrix} P_{2i-1} \\ P_{2i} \end{matrix}\right\} = -\cos\left(\frac{i\pi}{N}\right) \pm j \sin\left(\frac{i\pi}{N}\right) \quad i = 1, 2, \ldots, \frac{N-1}{2} \qquad \text{(C-7)}$$

$$P_N = -1 + j0$$

Even:

$$\left.\begin{matrix} P_{2i-1} \\ P_{2i} \end{matrix}\right\} = -\cos\left[\frac{(i-0.5)\pi}{N}\right] \pm j \sin\left[\frac{(i-0.5)}{N}\pi\right] \quad i = 1, 2, \ldots, \frac{N}{2} \qquad \text{(C-8)}$$

C.2.3 Chebyshev Filters

Chebyshev filters are characterized by an equiripple passband as shown in Figures C-4(a) and C-4(b) for even and odd orders, and a monotonic passband response. An alternate Chebyshev polynomial approximation provides an inverse response, i.e., monotone passband and ripply stopband response.

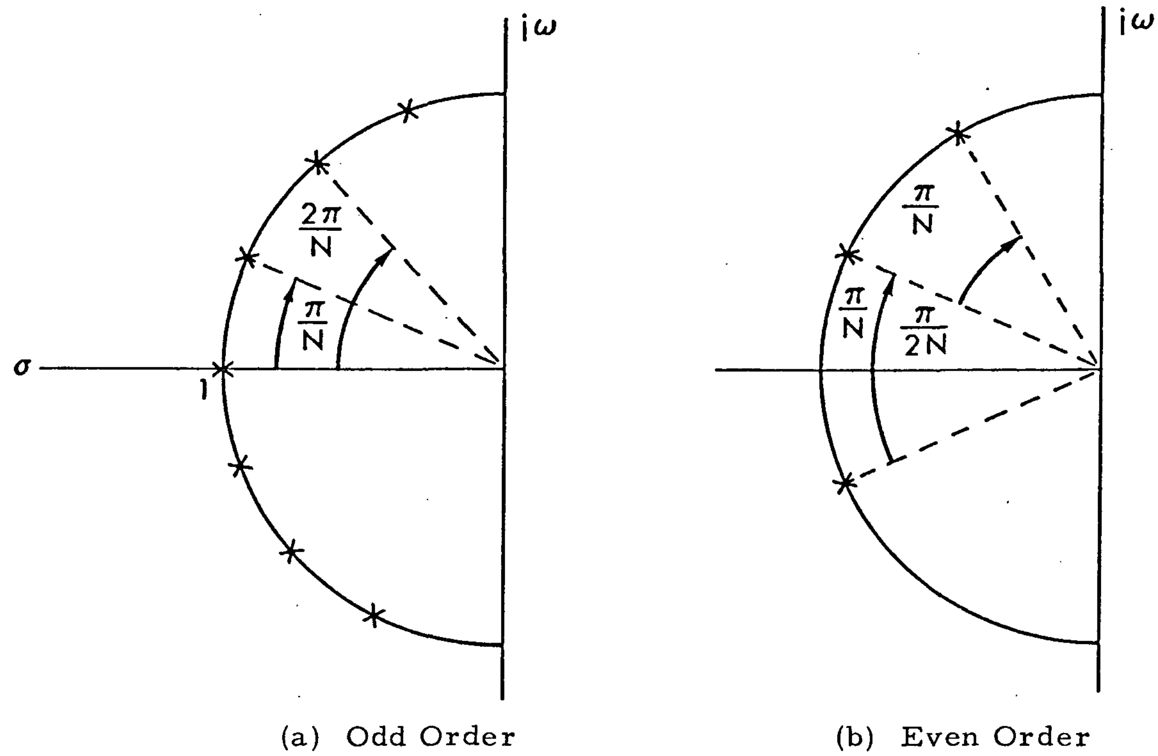


Figure C-3. Butterworth Pole Locations

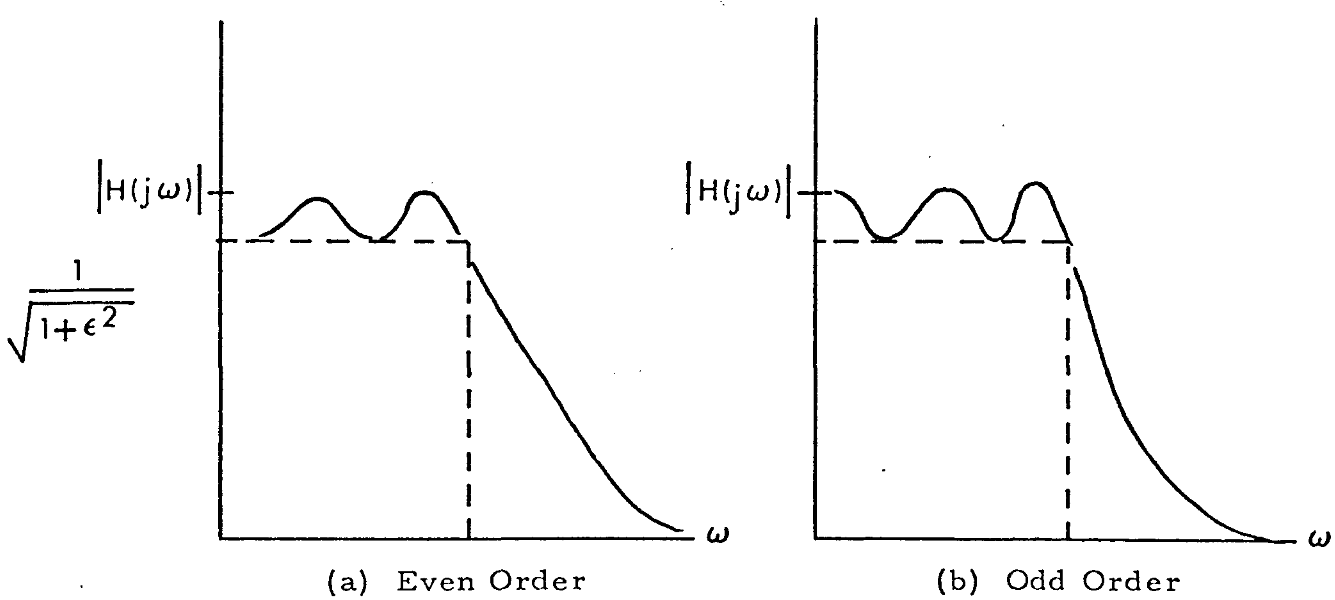


Figure C-4. Chebyshev Amplitude Response

The equiripple approximation has been shown to provide the sharpest cut-off filters thus the Chebyshev is the sharpest possible all pole filter function.

The magnitude squared of the transfer function for the Chebyshev filter response function is given in Equation C-9.

$$|H(j\omega)|^2 = \frac{1}{1 + \epsilon^2 T_N^{\ 2}(\omega)} \qquad \text{(C-9)}$$

where

$$\begin{aligned} T_N(\omega) &= \cos\left[N \cos^{-1}(\omega)\right] \\ &= \cos\left[N \cos^{-1}(\omega)\right] \quad 0 \le \omega \le 1 \\ &= \cosh\left[N \cosh^{-1}(\omega)\right] \quad \omega > 1 \end{aligned} \qquad \text{(C-9A)}$$

T_N can be put in polynomial form, yielding the Chebyshev polynomials of order N.

The roots can be found by solving the denominator of Equation C-9 for s after substituting $\omega = s/j$ and selecting the poles in the left-half plane.

This expression has been solved in Reference 54 and the Chebyshev poles have been shown to be on an ellipse in the s plane. It has also been shown that the Chebyshev poles are simply related to the Butterworth poles. This relationship is given by defining an intermediate variable, ϕ, to be

$$\phi = \frac{1}{N} \sinh^{-1}\left(\frac{1}{\epsilon}\right) \qquad \text{(C-10)}$$

or

$$\phi = \frac{1}{N}\left[\ln\left(\frac{1}{\epsilon} + \sqrt{1 + \frac{1}{\epsilon^2}}\right)\right] \qquad \text{(C-10A)}$$

The Chebyshev poles are then computed to be

$$P_i = \alpha_{B_i} \cosh(\phi) + j\,\beta_{B_i} \sinh(\phi) \qquad \text{(C-11)}$$

where α_{B_i} and β_{B_i} are the real and imaginary parts respectively of the Butterworth pole positions as defined in Equations C-7 and C-8. The value of ϵ may be computed in terms of the ripple amplitude (A_R). The ripple in dB is given by

$$A_R = -20 \log_{10}\left[\frac{1}{\sqrt{1+\epsilon^2}}\right] = 10 \log_{10}\left[1+\epsilon^2\right] \qquad \text{(C-12)}$$

which yields

$$\epsilon = \sqrt{10^{A_R/10} - 1} \qquad \text{(C-12A)}$$

C.2.4 Bessel Filters*

The Bessel filter is characterized by its maximally flat time delay (i.e., linear phase) characteristic. The linear phase characteristic is obtained without regard for the amplitude response and the result is a non-selective amplitude characteristic.

The transfer function for an ideal time delay (= 1 sec) is given by

$$H(s) = e^{-s} = \frac{1}{\cosh(s) + \sinh(s)} \qquad \text{(C-13)}$$

This expression cannot be expanded directly and truncated at N terms because it is not Hurwitz (all poles in left-half plane) for $N > 4$. The problem is to find a Hurwitz denominator. A Hurwitz polynomial is the sum of even and odd parts of some reactance functions $m(s)/n(s)$ (even/odd).

*Bessel filters, named for the Bessel polynomial, used in their realization were derived by W. E. Thomson, and are sometimes referred to as Thomson filters.

Note that sinh is an odd function and cosh an even function. If we expand the following in a series and then in a continued fraction representation the result is

$$\frac{\cosh s}{\sin h\ s} = \frac{1 + \frac{s^2}{2!} \text{---}}{s + \frac{s^3}{3!} \text{---}} = \frac{1}{s} + \cfrac{1}{\frac{3}{s} + \cfrac{1}{\frac{5}{s} + \cfrac{1}{\frac{7}{s} \text{---}}}} \tag{C-14}$$

This is the form of a reactance function (all coefficients are positive) and may be truncated at the Nth step to form:

$$H(s) = \frac{K}{m(s) + n(s)} = \frac{K}{B_N(s)}$$

$$H(s) = \frac{b_o}{b_o + b_1 s \text{ --- } b_N s^N} \tag{C-15}$$

It has been shown that m(s) + n(s) is a Bessel polynomial which is defined by the following recursion relationship

$$B_N = (2N-1)B_{N-1} + s^2 B_{N-2} \tag{C-16}$$

where

$$B_o = 1$$

$$B_1 = s + 1$$

The Bessel coefficients are of the following form

$$b_k = \frac{(2N - k)!}{2^{N-k} \, (N-k)! \; k!} \qquad k = 0, N \tag{C-17}$$

The poles of this function can be found by digital computer and are published in Reference 58. Note that Equation C-15 has been derived for 1 sec time delay, but it would be desirable to work with a function normalized to a given bandwidth. The half power bandwidths for orders N = 1 to 12 were computed and are given in Table C-1.

Using these bandwidths we may then normalize the 1 sec time delay poles to a 1 rad/sec bandwidth by using Equation C-18.

$$P_i \text{ (1 rad)} = P_i \text{ (1 sec)}/BW(N) \tag{C-18}$$

A table of the 1 radian/sec poles is given in Table C-2.

Table C-1. Half Power Bandwidths
for Bessel Filters of Order N

Order N	BW(N)
1	1.0000000000
2	1.3616541384
3	1.7556723593
4	2.1139176843
5	2.4274106977
6	2.7033950803
7	2.9517221448
8	3.1796172138
9	3.3916931150
10	3.5909806248
11	3.7796073910
12	3.9591507907

Table C-2. Bessel Poles

BESSEL POLES NORMALIZED TO 1 RAD/SEC BANDWIDTH

ORDER	COMPLEX POLES	
1	-1.00000000	
2	-1.10160132	J 0.63600982
3	-1.04740916	J 0.99926443
	-1.32267579	
4	-1.37006782	J 0.41024971
	-0.99520876	J 1.25710572
5	-1.38087733	J 0.71790959
	-0.95767655	J 1.47112432
	-1.50231627	
6	-1.57149039	J 0.32089637
	-1.38185808	J 0.97147188
	-0.93065652	J 1.66186325
7	-1.61203876	J 0.58924451
	-1.37890321	J 1.19156677
	-0.90986778	J 1.83645135
	-1.68436818	
8	-1.75740841	J 0.27286757
	-1.63693944	J 0.82279563
	-1.37384123	J 1.38835658
	-0.89286973	J 1.99832587
9	-1.80717054	J 0.51238373
	-1.65239649	J 1.03138956
	-1.36758831	J 1.56773372
	-0.87839927	J 2.14980054
	-1.85660051	
10	-1.84219623	J 0.72725759
	-1.92761967	J 0.24162347
	-1.66181022	J 1.22110020
	-1.36069226	J 1.73350573
	-0.86575689	J 2.29260480
11	-1.98016065	J 0.45959875
	-1.86736125	J 0.92311559
	-1.66719365	J 1.39596291
	-1.35348669	J 1.88829686
	-0.85451259	J 2.42805946
	-2.01670149	

C.2.5 Butterworth-Thomson (Bessel) Filters

In Sections C.2.2 and C.2.4 we discussed filters which had a maximally flat amplitude and a maximally flat time delay. As one might suspect, some tradeoff between the two characteristics can be obtained. Another comparison between the two types can be made in terms of the rise time and overshoot of the two filters. The Butterworth has faster rise time but more overshoot while the Bessel has slower rise time and very little overshoot. A detailed comparison of the two plus a discussion of the transitional Butterworth-Thomson function is given in a paper by Peless and Murakami (Reference 58).

The transition from the Butterworth to the Bessel function is obtained by moving the poles in a smooth locus from one limit point to another. The movement of a typical pole is shown in Figure C-5.

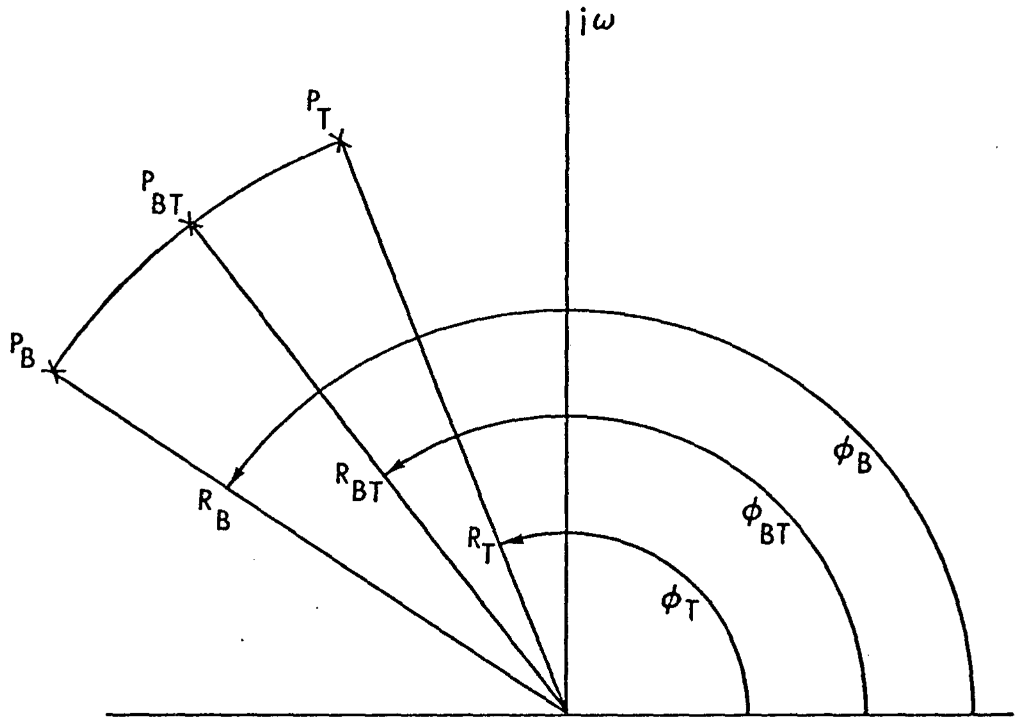
A parameter M determines the degree of movement from Butterworth ($M = 0$) to Bessel ($M = 1$). The Butterworth-Thomson poles then are given by Equation C-19.

$$P_{BT_i} = \left|P_{T_i}\right|^M e^{j\left[(1 - M)\phi_{Bi} + M\phi_{T_i}\right]} \qquad \text{(C-19)}$$

Reference 113, Volume I, uses the 1 sec time delay Bessel poles normalized so that

$$\prod_{k=1}^{N} |P_k| = 1$$

for the analysis. This produces a set of filters for which the bandwidth is a function of both the order and the parameter M. The bandwidth varies monotonially from 1 for $M = 0$ (Butterworth) to a number greater than 1 for the $M = 1$ (Bessel). Note that any normalization of poles (to change the bandwidth) only changes R and has no effect on ϕ. From this we can see that the response of the resulting BT filter is independent (except for



P_T = BESSEL POLES (1 RAD/SEC)

P_B = BUTTERWORTH POLES

$R_{BT} = R_T{}^M$

$_{BT} = (1 - M)\ _B + M\ _T$

Figure C-5. Butterworth and Bessel Pole Transition Loci

bandwidth scaling) of the bandwidth (and correspondingly the Rs) of the of the Bessel poles. Noting the monotonic behavior of the bandwidth as a function of M, it is logical to choose Bessel poles with a 1 rad/sec bandwidth, for then the bandwidth of the BT filter will be close to 1 rad/sec for all values of M. Care must be used in pairing the poles from Bessel and Butterworth. A rule that may be used is: choose a pole from each Bessel and Butterworth with the largest real part, then choose the next pair with the next largest real part, etc.

C.2.6 Elliptic Function Filter*

The elliptic function filter has been shown to be the optimum filter (sharpest cutoff for a given complexity) when both poles and zeros are permitted. The magnitude response is given by:

$$|H(j\omega)|^2 = \frac{1}{1 + \epsilon^2 R_n^2} \tag{C-20}$$

where

$$R_n = \frac{K_1 \omega \left(\omega_2^2 - \omega^2\right) \cdots \left(\omega^2_{n-1}\, \omega^2\right)}{\left(1 - \left(\frac{\omega_2}{\omega_s}\right)^2 \omega^2\right) \cdots \left(1 - \left(\frac{\omega_{n-1}}{\omega_s}\right)^2 \omega^2\right)}$$

for n = odd, and

$$R_n = \frac{K_2 \left(\omega_1^2 - \omega^2\right) \cdots \left(\omega^2_{n-1}\, \omega^2\right)}{\left(1 - \left(\frac{\omega_1}{\omega_s}\right)^2 \omega^2\right) \cdots \left(1 - \left(\frac{\omega_{n-1}}{\omega_s}\right)^2 \omega^2\right)}$$

for n = even.

There are two common ways of normalizing the elliptic filter function. We have chosen to normalize to the end of the passband ($\omega_n = 1$). The bandwidth specified then will be the "ripple bandwidth". With this normalization the zeros of H(s) are inversely proportional (with constant ω_s) to the maxima in the passband. The other method normalizes to the geometric

*Also called "Cauer parameter" filters, and "rational Chebyshev" filter.

mean of the end of the passband (ω_n) and the beginning of the stopband (ω_s) so that $\sqrt{\omega_n \omega_s} = 1$ and $\omega_n = 1/\omega_S$. With the latter normalization the zeros of H(s) are reciprocals of the maxima of the passband and the critical frequencies are

$$\omega_i = \sqrt{k_2}\, SN^*\left[\frac{iK\,(k_2)}{N}\right] \tag{C-21}$$

where

$$K(k) = \text{Complete elliptical integral}$$

$$k_2 = \frac{\omega_N}{\omega_s};\ i = 1,\ N$$

The even ordered filters are referred to as "hypothetical filters" since they cannot by synthesized without transformers. It has been shown (Reference 114, Volume I), however, that by applying a transformation a realizable filter function can be obtained while retaining the equiripple property. This transformation moves the lowest zero of $R_N(\omega_1)$ to zero and the highest pole (ω_s/ω_1) to infinity. The resulting R_N is given by Equation C-22.

$$R_N = \frac{K_3\ \omega^2\left(\omega_3'^2 - \omega^2\right) -- \left(\omega_n'^2 - 1 - \omega^2\right)}{\left(1 - \left(\frac{\omega_3'}{\omega_s}\right)^2 \omega^2\right) --- \left(1 - \left(\frac{\omega_{n-1}'}{\omega_s}\right)^2 \omega^2\right)} \tag{C-22}$$

*SN = Jacobi sine function.

where

$$\omega_i' = \frac{1}{\omega_n''}\sqrt{\frac{\omega_i^2 - \omega_1^2}{1 - \left(\frac{\omega_1}{\omega_s}\right)^2 \omega_i^2}}$$

$$\omega_n'' = \sqrt{\frac{1 - \omega_1^2}{1 - \left(\frac{\omega_1}{\omega_s}\right)^2}}$$

The factor ω_n'' is necessary so that $\omega_n' = 1$.

The three types of filter magnitude responses are shown in Figure C-6.

The location of the ω_i was first found by Cauer from elliptic function theory to be

$$\omega_i = SN\left(\frac{iK\ (k_2)}{N}\right) \quad i = 1, N \tag{C-23}$$

where

$$k_2 = \frac{1}{\omega_S}$$

Note that $\omega_N = 1$, since $SN[K(k)] = 1$.

The parameters of the filters are:

α_P = Minimum passband gain

α_S = Maximum stopband gain

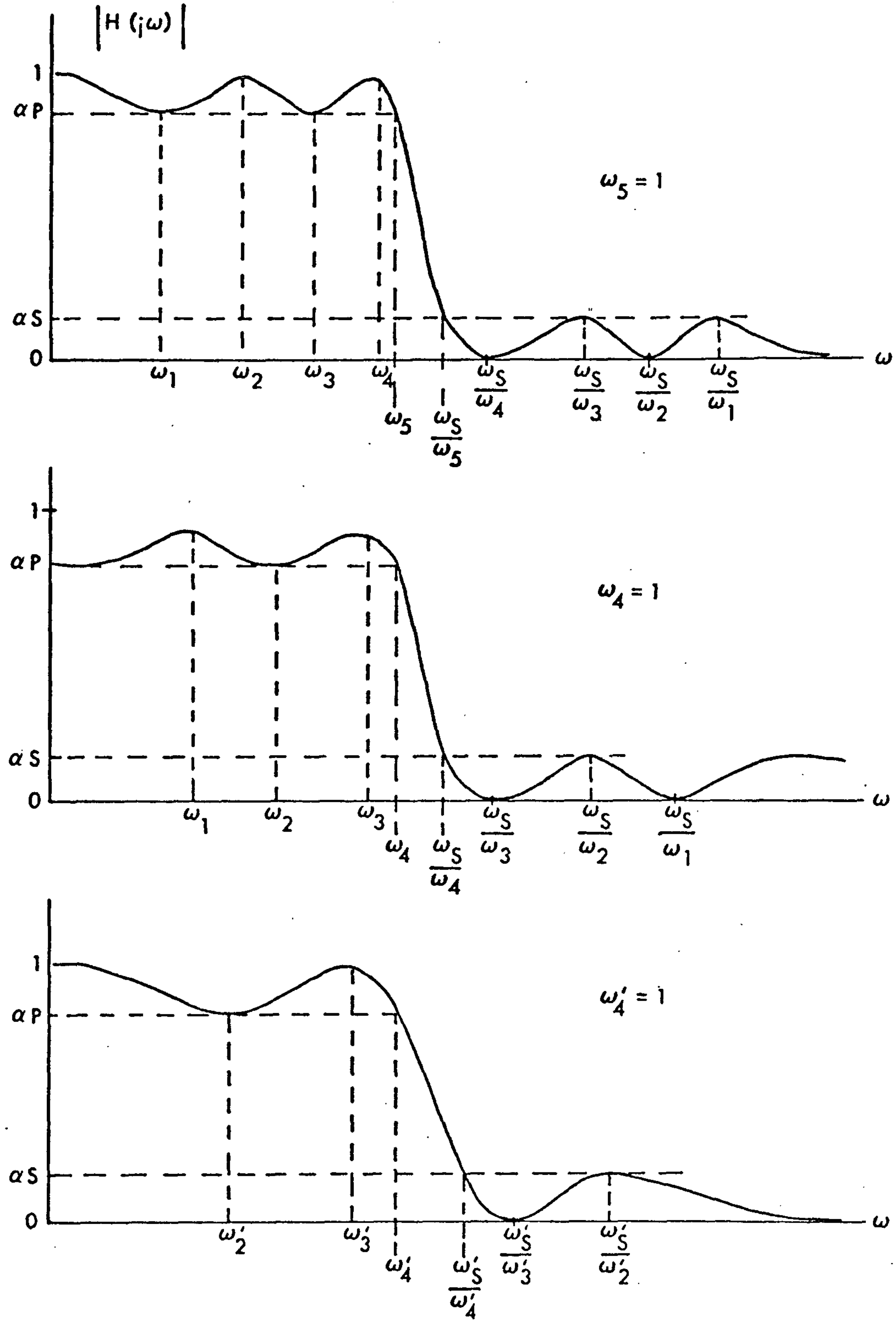


Figure C-6. Normalized and Hypothetical Filter Magnitude Responses

ω_s^{-1} = Transition bandwidth

N = Order (complexity) of the filter:

$$H(s) = \frac{\left(K\, s^M + s^{M-1} + \ldots\, a_o\right)}{s^N + s^{N-1} + \ldots\, b_o} \qquad \text{(C-24)}$$

where

$M = N - 1$ for odd N

$M = N$ for hypothetical even N

$M = N - 2$ for transformed even N

Only three of the four parameters are needed to specify the response since

$$k_1 = \sqrt{\frac{1/\alpha_P{}^2 - 1}{1/\alpha_S{}^2 - 1}} \quad k_2 = 1/\omega_s \qquad \text{(C-25)}$$

$$q(k_1) = \left[q(k_2)\right]^N \text{ or } \frac{K(k_1')}{K(k_1)} = N\frac{K(k_2')}{K(k_2)} \qquad \text{(C-26)}$$

If the order (N), α_p, and k_2 are given k_1 may be computed from Equation C-26 and α_S from Equation C-25. α_P and α_S are usually expressed in dB.

$$A_P = 20 \log_{10} \alpha_P \tag{C-27}$$

$$A_S = 20 \log_{10} \alpha_S \tag{C-28}$$

$$A_S = 10 \log_{10} \left[1 + \frac{1}{k_1^2}\left(\frac{1}{\alpha_P^2} - 1\right)\right] \tag{C-28A}$$

An alternate set of specifications used by the Telefunken Design Tables (Reference 115, Volume I) provide a smoother range of parameters which are related to the ones given above, They are:

$$\rho = \sqrt{1 - \alpha_P^2} = \text{Reflection Coefficient} \tag{C-29}$$

$$\theta = \sin^{-1} (k_2) = \text{Modular Angle} \tag{C-30}$$

A typical design might be made by specifying

1) ρ or A_P

2) θ or ω_s

3) A_S

The order needed may then be obtained from the tables in Reference 115, Volume I, or various nomographs available.

The poles and zeros of elliptic function filters have been found both by algebraic means and by use of conformal mapping (Reference 116, Volume I) through the Jacobi elliptic functions. The mapping is

$$s = j\,\mathrm{SN}\left[-j\,(U + j\,V),\ k_2\right] \tag{C-31}$$

$$k_2 = 1/\omega_s$$

The poles and zeros lie equally spaced on parallel lines in the W plane (W = U + jV) with the following coordinates:

	Zeros	Poles	
U	$-K(k_2')$	$\dfrac{K(k_2)F\left[\sin^{-1}(\alpha_P),\ k_1'\right]}{N\,K(k_1)}$	
V_i	$\pm\left(1 - \dfrac{2i-1}{N}\right)K(k_2)$	$\pm\left(1 - \dfrac{2i-1}{N}\right)K(k_2)$ 0 (Real pole for N odd)	$i = 1, \dfrac{N-1}{2}$

Using an identity for complex arguments of the SN ( ), we can write down the poles and zeros of H(s) in the s plane

$$Z_i = j\frac{1}{k_2 SN(V_i, k_2)} \tag{C-32}$$

$$P_i = \frac{-CN(V_i, k_2)\, DN(V_i, k_2)\, SN(U, k_2')\, CN(U, k_2') + j\, SN(V_i, k_2)\, DN(U, k_2')}{1 - SN^2(U, k_2')\, DN^2(V_i, k_2)} \tag{C-33}$$

Note that the real pole for N odd is

$$P_o = \frac{SN(U, k_2')}{CN(U, k_2')} \tag{C-34}$$

The poles and zeros given for N even are for the hypothetical filter and must be transformed to the realizable filter by following

$$\left[s_i' = \frac{1}{\omega_N''} \sqrt{\frac{s_i^2 + \omega_1^2}{s_i^2 + \left(\frac{\omega_s}{\omega_1}\right)^2}} \qquad \begin{aligned} s_i &= \text{complex pole or zero} \\ \omega_1 &= SN\left[\frac{K(k_2)}{N}\right] \end{aligned} \right. \tag{C-35}$$

This transformation is applied to both poles and zeros. The complex pair of zeros at (ω_s/ω_1) is deleted.

This transformation does not alter the pass or stopband ripple but it does increase the transition interval. The new beginning of the stopband is:

$$\omega_s' = \omega_s \left[\frac{1 - (\omega_1/\omega_s)^2}{1 - \omega_1^2}\right] \tag{C-36}$$

C.2.7 Other Transfer Functions

L-Filters

"Optimum filters with monotonic response"*, or L-filters, are a class of filters optimum with respect to the properties: (1) monotonic response, with (2) sharpest possible cutoff.

Thus, if the L-filter amplitude function is

$$A(\omega) = \frac{A_o}{\sqrt{1 + L_n(\omega^2)}} \tag{C-37}$$

*"On the Approximation Problem in Filter Design", A. Papoulis, IRE National Convention Record, Volume 5, Part 2, pp. 175-185, 1957.

then the nth order L-filter is defined by the nth order polynomial L_n which satisfies these three properties:

1) For all ω, $\dfrac{dL_n(\omega^2)}{d\omega} \geq 0$

2) $L_n(1) = 1$ (arbitrary normalization)

3) $\left.\dfrac{dL_n(\omega^2)}{d\omega}\right|_{\omega = 1}$ is maximum

C.3 FREQUENCY TRANSFORMATIONS

Frequency transformations are used in filter synthesis so that one basic (normalized low-pass) filter may be synthesized and then other types may be derived from it. The transformations provide for both scaling of the frequency scale and for obtaining a different kind of response (e.g., band-pass) through mapping s_λ. Once the desired response is obtained by choosing the correct transformations and transformation parameters (ω_o, ω_b), the basic network can be altered appropriately without deriving the actual transfer functions. The details of the transformation are discussed in Weinberg (Reference 58).

In the following discussion, Table C-3 will be useful to refer to the four commonly used transformations which are implemented in the FILTER program.

Poles and Zeros

The general form of a filter transfer function can be represented in the form of Equation C-38. The relationships

$$H(\lambda) = \frac{A\prod_{i=1}^{M} (\lambda - ZN_i)}{\prod_{i=1}^{N} (\lambda - PN_i)} \qquad \text{(C-38)}$$

Table C-3. Frequency Transformations

Filter	Transformations*	
	λ	Ω
Low Pass	s/ω_b	ω/ω_b
High Pass	ω_b/s	ω_b/ω
Band Pass	$\frac{s^2 + \omega_o^2}{\omega_b s}$	$\frac{\omega^2 - \omega_o^2}{\omega\omega_b}$
Band Stop	$\frac{\omega_b s}{s^2 + \omega_o^2}$	$\frac{\omega_b \omega}{\omega^2 - \omega_o^2}$

*$s = j\omega$

$\lambda = \Phi + j\Omega$

ω_o = Center frequency

ω_b = Bandwidth

where

A = real constant $\qquad A = \frac{\Pi PN_i}{\Pi ZN_i}$

λ = complex frequency (normalized)

ZN_i = complex zero (normalized)

PN_i = complex pole (normalized)

M, N = number of zeros and number of poles, respectively.

between λ and the complex frequency $s(s = j\omega)$ are given in Table C-4. The objective is to utilize the transform relationships and the normalized

Table C-4. Transformation Relationships

Filter Type	Transform
Low Pass	$\lambda = s/\omega_b$
High Pass	$\lambda = \omega_b/s$
Band Pass	$\lambda = \dfrac{s^2 + \omega_o^2}{s\,\omega_o}$
Band Stop	$\lambda = \dfrac{s\,\omega_b}{s^2 + \omega_o^2}$

low-pass filter prototypes in order to obtain Equation C-39 in the form of Equation C-38. This objective is

$$H(s) = \frac{B \prod_{i=1}^{NZ} (s - Z_i)}{\prod_{i=1}^{NP} (s - P_i)} \qquad \text{(C-39)}$$

achieved by determining the relationship between the normalized poles and zeros (PN_i and ZN_i) and the poles and zeros in Equation C-35 (P_i and Z_i). Tables C-4 and C-5 present a summary of these results which defines the transform expression for each of the four filter types. These expressions are implemented in the FILTER program.

Group Delay

Group delay (t_g) for a filter is defined as shown in Equation C-40.

$$t_g = \left. \frac{-d\,\phi(\omega)}{d\omega} \right|_{s = j\omega} \qquad \text{(C-40)}$$

C.V

Table C-5. Poles and Zeros of Transformed Filters

Type	Trans-formation	Transformed Multiplication Constant, B	No. of Poles Generated	Poles	No. of Zeros Generated	Zeroes
Low Pass	$\lambda = s/\omega_b$	$A \cdot \omega_b^{N-M}$	N	$\omega_b PN_i$	M	$\omega_b ZN_i$
High Pass	$\lambda = \omega_b/s$	1	N	$\omega_b PN_i$	M	ω_b / ZN_i
					M	$0 + j0$
Band Pass	$\lambda = \dfrac{s^2 + \omega_o^2}{\omega_b s}$	$A \cdot \omega_b^{N-M}$	2N	$1/2\left[\omega_b PN_i \pm \sqrt{\omega_b^2 PN_i^2 - 4\omega_o^2}\right]$	2M	$1/2\left[\omega_b AN_i \pm \sqrt{\omega_b^2 ZN_i - 4\omega_o^2}\right]$
					N-M	$0 + j0$
Band Stop	$\lambda = \dfrac{\omega_o s}{s^2 + \omega_o^2}$	1	2N	$1/2\left[\dfrac{\omega_b}{PN_i} \pm \sqrt{\left(\dfrac{\omega_b}{PN_i}\right)^2 - 4\omega_o^2}\right]$	2N	$1/2\left[\left(\dfrac{\omega_b}{ZN_i}\right)^2 \pm \sqrt{\left(\dfrac{\omega_b}{ZN_i}\right)^2 - 4\omega_o^2}\right]$
					2(N-M)	$\pm j\omega_o$

ZN_i = Normalized Zero

PN_i = Normalized Pole

ω_o = Center Frequency

ω_b = Bandwidth

M = Number of Normalized Zeros

N = Number of Normalized Poles

where

ω = Radian frequency

$\phi(\omega)$ = Steady state phase response of filter

The complex steady state phase response for the normalized filter is found by substituting $\lambda = j\Omega$ in Equation C-38. This expression is given in Equation C-41.

$$H(j\Omega) = A \frac{\prod_{i=1}^{M} (j\Omega - ZN_i)}{\prod_{i=1}^{N} (j\Omega - PN_i)} \tag{C-41}$$

where

ZN_i = Normalized Zero = $\alpha_i + j\beta_i$

PN_i = Normalized Pole = $\alpha_i + j\epsilon_i$

The phase response of this function is readily computed below in Equation C-42.

$$\phi(\Omega) = \sum_{i=1}^{N} \tan^{-1}\left(\frac{\Omega - \beta_i}{-\alpha_i}\right) - \sum_{i=1}^{M} \tan^{-1}\left(\frac{\Omega - \epsilon_i}{-\gamma_i}\right) \tag{C-42}$$

In order to find the phase response of the transformed filter the equality in Equation C-43 is used, where $T(\omega)$ is the appropriate transformation taken from Table C-6. Using Equation C-42 and

$$\phi(\Omega) = \phi\left(T(\omega)\right) \tag{C-43}$$

Table C-6. Formulas for Group Delay of Transformed Filters

Filter	$T(\omega)$	$\frac{dT(\omega)}{d\omega}$	$t_g(\omega)$
Low Pass	$\Omega = \omega/\omega_b$	$1/\omega_b$	$\omega_b \left\{ \sum_{i=1}^{M} \frac{\alpha_i}{\alpha_i^2 \omega_b^2 + (\beta_i \omega_b - \omega)^2} - \sum_{i=1}^{N} \frac{\gamma_i}{\gamma_i^2 \omega_b^2 + (\epsilon_i \omega_b - \omega)^2} \right\}$
High Pass	$\Omega = -\omega_b/\omega$	$\frac{\omega_o}{\omega^2}$	$\omega_b \left\{ \sum_{i=1}^{M} \frac{\alpha_i}{\alpha_i^2 \omega_b^2 + (\beta_i \omega_b + \omega)^2} - \sum_{i=1}^{N} \frac{\gamma_i}{\gamma_i^2 \omega_b^2 + (\epsilon_i \omega_b + \omega)^2} \right\}$
Band Pass	$\Omega = \frac{\omega^2 - \omega_o^2}{\omega\omega_b}$	$\frac{\omega^2 + \omega_o^2}{\omega^2 \omega_b}$	$\omega_b(\omega^2 + \omega_o^2) \left\{ \sum_{i=1}^{M} \frac{\alpha_i}{\alpha_i^2 \omega^2 \omega_b^2 + (\beta_i \omega\omega_b + \omega_o^2 - \omega^2)^2} - \sum_{i=1}^{N} \frac{\gamma_i}{\gamma_i^2 \omega^2 \omega_b^2 + (\epsilon_i \omega\omega_b + \omega_o^2 - \omega^2)^2} \right\}$
Band Stop	$\Omega = \frac{\omega\omega_b}{\omega_o^2 - \omega^2}$	$\frac{\omega_b(\omega^2 + \omega_o)^2}{(\omega^2 - \omega_o^2)^2}$	$\omega_b(\omega^2 + \omega_o^2) \left\{ \sum_{i=1}^{M} \frac{\alpha_i}{\alpha_i^2(\omega^2 - \omega_o^2) + [\beta(\omega^2 - \omega_o^2) + \omega\omega_b]^2} - \sum_{i=1}^{N} \frac{\gamma_i}{\theta_i^2(\omega^2 - \omega_o^2)^2 + [\epsilon(\omega^2 - \omega_o^2) + \omega\omega_b]^2} \right\}$

where $PN_i = \gamma_i + j\,\epsilon_i$ = Normalized Poles

$ZN_i = \alpha_i + j\,\beta_i$ = Normalized Zeroes

ω_o = Geometric Center Frequency

ω_b = Bandwidth

M = Number of Normalized Zeros

N = Number of Normalized Poles

and

$$t_g = -\frac{d\phi(\omega)}{d(\omega)} = -\frac{d\,\phi\,(\Omega)}{d\Omega}\,\frac{d\Omega}{d\omega} \tag{C-44}$$

the group delay of a normalized low-pass filter can be obtained as given in Equation C-45.

$$t_{gn}(\Omega) = \sum_{i=1}^{N} \frac{a_i}{a_i^{\,2} + (\beta_i - \Omega)^2} + \sum_{i=1}^{M} \frac{\gamma_i}{\gamma_i^{\,2} + (\epsilon_i - \Omega)^2} \tag{C-45}$$

and

$$t_g(\omega) = t_{gn}\Big(T(\omega)\Big)\,\frac{d\Omega}{d\omega} \tag{C-46}$$

The group delay for the transformed filters is derived from these relationships and the multiplier $d\Omega/d\omega$ which is a function of the type of transformation being made. Equation C-46 shows the expression for group delay as a function of the normalized low-pass expression and Table C-6 lists the group delay functions for the four transformations being considered. The group delay at zero frequency and at center frequency are also of interest. These functions are tabulated in Table C-7. Also, there are several interesting observations to be made about the group delay, and they are as follows:

- Always continuous and bounded and is zero only at $\omega = \infty$.
- Poles and zeros with a zero real part contribute nothing to group delay.
- If $\left|P_i\right| = 1$ group delay for low-pass is the same as for high-pass and band-pass is the same as band-stop (i. e., Butterworth filters case).

Table C-7. Group Delay at Zero and Center Frequency

Filter	$t_g(0)$	$t_g(\omega_o)$
Low Pass	$-\frac{1}{\omega_b}\sum_{i=1}^{N}\frac{P_i}{\|P_i\|^2}$	- - - - - -
High Pass	$-\frac{1}{\omega_b}\sum_{i=1}^{N} P_i$	- - - - - -
Band Pass	$-\frac{\omega_b}{\omega_o^2}\sum_{i=1}^{N} P_i$	$-\frac{2}{\omega_b}\sum_{i=1}^{NP}\frac{P_i}{\|P_i\|^2}$
Band Stop	$-\frac{\omega_b}{\omega_o^2}\sum_{i=1}^{N}\frac{P_i}{\|P_i\|^2}$	$-\frac{2}{\omega_b}\sum_{i=1}^{NP} P_i$

Notes: 1) Zeros are assumed to have zero real parts.

2) Poles are real or conjugate pairs.

C.4 EQUIVALENT NOISE BANDWIDTH

There are numerous definitions of the equivalent noise bandwidth (ENB) of a filter. The one which is implemented in this FILTER program is given in Equation C-47.

$$ENB = \frac{\frac{1}{2\Pi}\int_0^{\infty} |H(j\omega)|^2 \, d\omega}{|H(j\omega)|^2_{max}} \tag{C-47}$$

This is the bandwidth of a hypothetical filter having unity gain in the passband and zero gain in the stopband followed by an amplifier with gain = H_{max}. The noise power passed by such a filter and amplifier would be

$$P_n = N_o \, ENB \, H_{max}^2 \tag{C-48}$$

where N_o is the one-sided power spectral density of the (flat) noise in watts/Hz and where $H(j\omega)$ is the filter transfer function with $s = j\omega$. The integral can be evaluated in the following way:

$$H(j\omega)^2 = H(j\omega)\, H^*(j\omega) = H(s)\, H(-s) \tag{C-49}$$

Substituting $s = j\omega$ in the integral in Equation C-47 yields

$$ENB = \frac{\left(\frac{1}{2\Pi j}\right)\left(\frac{1}{2}\right)\int_{-j\infty}^{j\infty} H(s)\, H(-s)\, ds}{H_{max}^2} = \frac{1}{2\Pi j}\,\frac{1}{2H_{max}^2}\int_C H(s)\, H(-s)\, ds \tag{C-50}$$

where C is a closed contour in the complex s-plane. Since the right-half plane poles are the mirror image of the left-half plane, the contour of integration need only contain those poles in the left-half plane. The integral in Equation C-50 can be evaluated by finding its residues. The integrand can be represented as follows

$$H(s)\, H(-s) = -A^2 \frac{\prod_{i=1}^{NZ} (s - Z_i)(-s - Z_i)}{\prod_{i=1}^{NP} (s - P_i)(-s - P_i)} \tag{C-51}$$

The definition of the residues of Equation C-51 is given as

$$R_m(P_j) = \lim_{S \to P_j} \left[H(s)\, H(-s)\, (s - P_j) \right] \qquad \text{(C-52)}$$

which reduces to

$$R_m(P_j) = -A^2 \frac{\prod_{i=1}^{NZ} \left(Z_i^{\,2} - P_j^{\,2} \right)}{2P_j \prod_{i=1}^{NP} \left(P_i^{\,2} - P_j^{\,2} \right)} \quad i \neq j \qquad \text{(C-53)}$$

substituting this result into Equation C-50 yields the appropriate expression (Equation C-54) for the ENB in terms of the residues of the poles in the left-half of the s-plane.

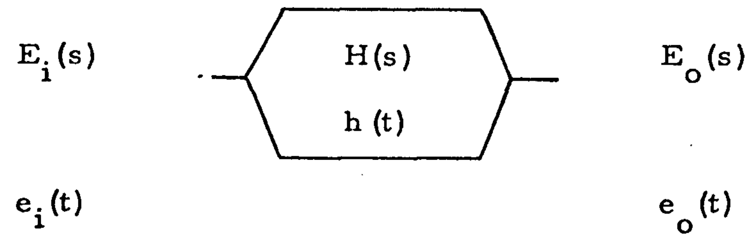
$$ENB = \frac{1}{2H_{max}^2} \sum_{j=1}^{NP} R_m\,(P_j) \qquad \text{(C-54)}$$

It is important to note the following:

1) H(s) is the transformed filter transfer function.
2) H(s) can have no poles on the $j\omega$ axis.
3) Multiple poles are not allowed.
4) The number of poles must be greater than the number of zeros.

C.5 TRANSIENT RESPONSE

The time response of a filter may be obtained by evaluating the inverse Laplace transforms of the input signal and transfer function.



where

$e_i(t) \Longleftrightarrow E_i(s)$	input signal
$h(t) \Longleftrightarrow H(s)$	impulse response
$e_o(t) \Longleftrightarrow E_o(s)$	output signal

The Laplace transform pairs are given in Equations C-55 and C-56.

$$G(s) = \int_0^{\infty} g(t)e^{-st}\,dt \qquad \text{(C-55)}$$

$$g(t) = \int_{\epsilon-j\infty}^{\epsilon+j\infty} G(s)e^{st}\,ds \qquad \text{(C-56)}$$

We write

$$G(s) = L\big(g(t)\big) \qquad \text{(C-57)}$$

$$g(t) = L^{-1}\big(G(s)\big) \qquad \text{(C-58)}$$

The transfer function $H(s)$ is known (computed by the program) and the Laplace transforms for various input signals have been tabulated. The Laplace transform of the output signal is:

$$E_o(s) = E_i(s)\,H(s) \tag{C-59}$$

The problem is to evaluate the inverse Laplace transform $L^{-1}(E_o(s)) = e_o(t)$ from the inversion formula

$$e_o(t) = \frac{1}{2\Pi j}\int_{\epsilon-j\infty}^{\epsilon+j\infty} E_o(s)e^{st}\,ds = \frac{1}{2\Pi j}\int_C E_o(s)e^{st}\,ds \tag{C-60}$$

In useful cases $E_o(s)$ has poles only in the left-half plane ($Re(s) < 0$) or on the $j\omega$ axis and has more poles than zeros. The integrand then vanishes at $s\ -\infty$ for $t > 0$, and the integral can be replaced by a contour integral around the left-half plane shown in Figure C-7.

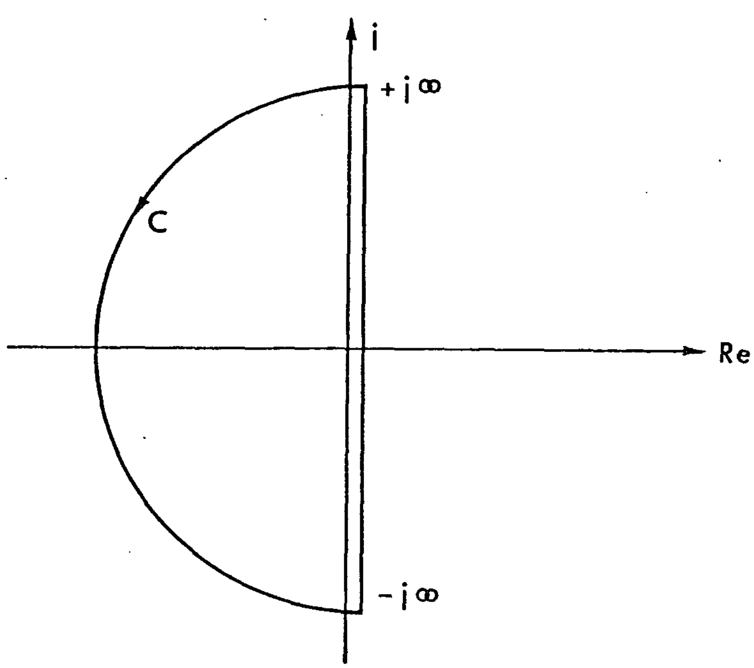


Figure C-7. Contour of Integration

Then $e_o(t) = \Sigma$ residues of $E_o(s)e^{st}$ at poles of $E_o(s)$

$$E_o(s) = \frac{B \prod_{i=1}^{NZ'} (s - Z_i)}{\prod_{i=1}^{NP'} (s - P_i)} \qquad NP' > NZ' \tag{C-61}$$

where NP' and NZ' are the number of poles and zeros of $E_o(s)$.

For simple poles and a general function G(s)

$$R(P_h) = (s - P_h)\, G(s) \tag{C-62}$$

$$\lim s \rightarrow P_h$$

for poles of multiplicity m

$$R(P_h) = \frac{1}{(m-1)!} \frac{d^{m-1}}{ds^{m-1}} \left[(s-P_h)^m \, G(s)\right] \tag{C-63}$$

$$\lim s \rightarrow P_h$$

If there are no multiple poles the modified residue $R'(P_i)$ may be evaluated:

$$R'(P_h) \frac{A \prod_{i=1}^{NZ'} (P_h - Z_i)\, e^{+P_h t}}{\prod_{i=1}^{NP'} (P_h - P_i)} = R(P_h) e^{P_h t} \quad , \quad i \neq h \tag{C-64}$$

Note that $R(P_h)$ is the residue at P_h of $E_o(s)$.

The output is thus:

$$e_o(t) = \sum_{h=1}^{NP'} R(P_h)\, e^{P_h t} \tag{C-65}$$

Note that while this equation contains complex quantities, the sum is real.

If there is a double pole at zero (a ramp input), the residue can be found from Equation C-63.

$$E_o(s) = \frac{1}{s^2} H(s) \tag{C-66}$$

$$R(o) = \frac{d}{ds}\left[H(s)e^{st}\right] \tag{C-67A}$$

$$= \left[t\, H(s)e^{st} + e^{st}\frac{dH(s)}{ds}\right]_{s=o} \tag{C-67B}$$

$$= tH(o) + \left.\frac{dH(s)}{ds}\right|_{s=o} \tag{C-67C}$$

$$H(s) = P(s)/Q(s) \qquad \begin{aligned} P(s) &= b_n s^n \ldots b_1 s \\ &= b_o \\ Q(s) &= a_n s^n \ldots a_1 s \\ &= a_o \end{aligned} \tag{C-68}$$

$$\frac{dH}{ds} = \left.\frac{QP' - PQ'}{Q^2}\right|_{s=o} = \frac{a_o b_1 - b_o a_1}{a_o^2} \tag{C-69}$$

Note that

$$a_o = \Pi e\, P_i \qquad\qquad b_o = \Pi Z_i$$

$$a_1 = \Sigma P_i \qquad\qquad b_i = \Sigma Z_i$$

$$H'(o) = \frac{\left[\sum Z_i\right]\left[\Pi P_i\right] - \left[\sum P_j\right]\left[\Pi Z_i\right]}{\left[\Pi P_i\right]^2} \tag{C-70}$$

Therefore for $e_i(t) = t$

$$e_o(t) = tH(o) + H'(o) + \sum_{i=1}^{NP} R(P_h) e^{P_h t} \tag{C-71}$$

While it is possible to compute the response for multiple poles other than the one described, these cases have not been implemented in the program described later. If a multiple pole (other than zero) is encountered, an overflow will result.

APPENDIX D

DISTRIBUTION LIST

NASA/MSC Facility and Laboratory Support Branch Houston, Texas 77058 Attention: Mr. Jerry Carlson Mail Code BB321 (77)	1 Copy
NASA/MSC Technical Library Branch Houston, Texas 77058 Attention: Ms. Retha Shirkey Mail Code JM6	4 Copies
NASA/MSC Management Services Division Houston, Texas 77058 Attention: Mr. John T. Wheeler Mail Code JM7	1 Copy
NASA/MSC Space Electronics System Division Houston, Texas 77058 Attention: Mr. C. T. Dawson Technical Monitor, EE8	4 Copies