

NPS ARCHIVE
1969
TAYLOR, A.

COMPUTER-AIDED FILTER DESIGN

by

Alec Patton Taylor

United States Naval Postgraduate School



THESIS

COMPUTER-AIDED FILTER DESIGN

by

Alec Patton Taylor, Jr.

December 1969

This document has been approved for public release and sale; its distribution is unlimited.

1133282

Computer-Aided Filter Design

by

Alec Patton Taylor, Jr.
Lieutenant Commander, United States Navy
B.S., Lincoln Memorial University, 1958

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1969

NPS ARCHIVE ~~Phase I/II/III~~ 61
1969
TAYLOR, A.

ABSTRACT

Several small-scale, special-purpose computer programs that were written to provide design data for a pulse-slimming filter over a wide range of parameter variations are contained herein. A large-scale computer program is also used to provide additional data. Some synthesis techniques are discussed, and some of the computer program outputs are directly applied in illustrating synthesis techniques for a specific pulse-slimming transfer function.

TABLE OF CONTENTS

I.	INTRODUCTION -----	5
	A. THE SIERRA FILTER -----	6
II.	COMPUTER SIMULATION -----	12
	A. PROGRAM NUMBER ONE -----	12
	B. PROGRAM NUMBER TWO -----	13
	C. PROGRAM NUMBER THREE -----	13
	1. Modification for N=2 -----	13
	2. Modification for N=3 -----	14
	3. Modification for N=4 -----	14
	4. Modification for N=5 -----	14
	5. Modification for N=6 -----	15
	6. Modification for N=7 -----	15
	7. Modification for N=9 -----	15
	D. PROGRAM NUMBER FOUR -----	16
	E. PROGRAM NUMBER FIVE -----	17
	1. Modifications Required for Denominator Coefficients -----	18
	2. Other Required Modifications -----	18
	F. PROGRAM NUMBER SIX -----	19
III.	SOME ASPECTS OF LATTICE SYNTHESIS -----	20
	A. THE UNLOADED LATTICE -----	21
	B. THE CONSTANT-RESISTANCE LATTICE -----	22
IV.	SYNTHESIS OF THE SIERRA TRANSFER FUNCTION FOR N=8 AND K=2 -----	26
	A. SYNTHESIZING THE IMPEDANCE, $Z_a(s)$ -----	26
V.	CONCLUSION -----	30

COMPUTER PROGRAMS -----	31
LIST OF REFERENCES -----	41
INITIAL DISTRIBUTION LIST -----	42
FORM DD 1473 -----	43

I. INTRODUCTION

Network analysis and synthesis, in general, continues to become less time-consuming and complex as a result of the continued upgrading of large-scale network-analysis computer programs such as those listed in Ref. 1 and the increased interest shown in computer-aided design by engineers at all levels.

A major disadvantage of the large-scale network analysis programs, however, is the vast amount of core storage usually required by them. If a large computer is not available to the design engineer, many of the more sophisticated programs will not be at his disposal. Additionally, owing to the fact that network synthesis, particularly filter design, is approached from many aspects depending on the specific application, the available large-scale programs must frequently be supplemented by a host of smaller, special-purpose programs in order to provide the required design information. Thus, it is essential that the present-day design engineer be well versed in network analysis programs and computer capabilities in order to successfully compete with his contemporaries.

Computer-aided design procedures are exemplified herein by a comprehensive analysis of the pulse-slimming filter designed by H. M. Sierra [Ref. 2]. The use of small, special-purpose programs coupled with a large-scale program provides the required design parameters.

Some aspects of lattice synthesis are discussed, and with the aid of the computer programs some of these synthesis techniques are applied to the Sierra transfer functions.

A. THE SIERRA FILTER

Reference 2 contains a detailed description of the Sierra Filter; however, for completeness, a brief description of this filter will be included.

Consider an input pulse, $f_i(t)$, approximated by the normal curve

$$\begin{aligned} f_i(t) &= \exp(-\xi^2 t^2) & -\alpha \leq t \leq \alpha \\ &= 0 & \text{otherwise.} \end{aligned} \quad (1.1)$$

The error in this approximation at $t = \alpha$ is simply the value of the Gaussian function at $t = \alpha$ or

$$\epsilon = \exp(-\xi^2 \alpha^2). \quad (1.2)$$

The problem as outlined by Sierra was to find a network composed of passive elements such that the output, $f_o(t)$, was approximated by the normal curve

$$\begin{aligned} f_o(t) &= \exp(-\eta^2 t^2) & -\beta \leq t \leq \beta \\ &= 0 & \text{otherwise.} \end{aligned} \quad (1.3)$$

Similarly, the error in this approximation at $t = \beta$ is

$$\epsilon = \exp(-\eta^2 \beta^2). \quad (1.4)$$

The compression ratio, K , was defined as

$$K = 2\alpha/2\beta. \quad (1.5)$$

Sierra assumed the same tolerable degree of error at both input and output pulse edges and equated (1.2) and (1.4) obtaining

$$\xi\alpha = \eta\beta \quad (1.6)$$

or $\xi/\eta = 1/K$.

The network transfer function was expressed as

$$H(s) = F_0(s)/F_i(s). \quad (1.7)$$

The two-sided Laplace transform of the input pulse was

$$\begin{aligned} F_i(s) &= \int_{-\infty}^{\infty} \exp(-\xi^2 t^2) \exp(-st) dt \\ &= (\sqrt{\pi}/\xi) \exp(s^2/4\xi^2). \end{aligned} \quad (1.8)$$

For the output pulse,

$$F_0(s) = (\sqrt{\pi}/\eta) \exp(s^2/4\eta^2). \quad (1.9)$$

After substituting (1.8) and (1.9) into (1.7), the transfer function becomes

$$H(s) = (\xi/\eta) \exp[(1/\eta^2) - (1/\xi^2)] s^2/4. \quad (1.10)$$

When (1.6) is substituted into (1.10), $H(s)$ becomes

$$H(s) = (1/K) \exp[(1/K^2) - 1] s^2/4\xi^2. \quad (1.11)$$

Since equation (1.11) still contains the parameter ξ , Sierra chose to normalize the input pulse by making $\xi = 1$, thus yielding

$$H(s) = (1/K) \exp[(1/K^2) - 1] s^2/4. \quad (1.12)$$

At this point, Sierra introduced the parameter ϕ , defining it as follows:

$$\phi^2 = (K^2 - 1)/4K^2 \quad (1.13)$$

or
$$\phi = \sqrt{K^2 - 1}/2K. \quad (1.14)$$

After substituting (1.13) into (1.12), $H(s)$ reduces to

$$H(s) = (1/K) \exp(-\phi^2 s^2). \quad (1.15)$$

As Sierra pointed out, the transfer function given by equation (1.15) is not realizable. Therefore, instead of the transfer function, another function was approximated. In this particular case, Sierra chose to approximate the magnitude function, noting the possibility of making this approximation and still obtaining accurate results in the time domain.

Substituting $s=j\omega$ into (1.15) and taking the magnitude,

$$|H(j\omega)| = (1/K) \exp(\phi^2 \omega^2) , \quad (1.16)$$

and $\text{ARG } H(j\omega) = 0^\circ . \quad (1.17)$

The magnitude functions of the input and output pulses are

$$|F_i(j\omega)| = \sqrt{\pi} \exp(-\omega^2/4) \quad (1.18)$$

$$|F_o(j\omega)| = (\sqrt{\pi}/K) \exp(-\omega^2/4K^2) . \quad (1.19)$$

To obtain the frequency, ω_e , at which $|F_i(j\omega)| = |F_o(j\omega)|$, Sierra equated (1.18) and (1.19), yielding the following:

$$\sqrt{\pi} \exp(-\omega_e^2/4) = (\sqrt{\pi}/K) \exp(-\omega_e^2/4K^2)$$

or $\omega_e = 2K[\ln K/(K^2 - 1)]^{1/2} . \quad (1.20)$

At this juncture, Sierra observed that the approximations to $|H(j\omega)|$ must be bandwidth limited in order to obtain realizable transfer functions; that is, every approximation is valid only up to a frequency ω_d . Sierra also pointed out that, since the network is to be realized with passive elements only, and with the same impedance level for $\omega_d > \omega_e$, the necessity for introducing some attenuation factor, A , exists.

Using equation (1.16), the magnitude-squared function can be obtained. It is

$$|H(j\omega)|^2 = (1/K^2) \exp(2\phi^2 \omega^2) \quad (1.21)$$

To approximate (1.21) by a ratio of polynomials in ω , Sierra chose to use the continued-fraction expansion because of the very fast convergence it offers.

$$\text{Substituting } z = 2\phi^2 \omega^2 \quad (1.22)$$

into (1.21) yields

$$|H(z)|^2 = (1/K^2) \exp(z) \quad (1.23)$$

Expressing $\exp(z)$ as a continued fraction, the following is obtained:

$$\exp(z) = \frac{1}{1-z} \cfrac{1+z}{2} \cfrac{2}{1-z} \cfrac{6}{1+z} \cfrac{6}{1-z} \cfrac{10}{1+z} \cfrac{10}{1-z} \cfrac{14}{1+} \dots \quad (1.24)$$

If N = number of terms taken in the expansion, the following approximations to $\exp(z)$ are obtained:

$$\text{For } N = 1: \frac{1}{1}$$

$$\text{For } N = 2: \frac{1}{1-z}$$

$$\text{For } N = 3: \frac{2+z}{2-z}, \text{ etc.}$$

Thus, substituting for z yields the following approximations:

$$\text{For } N = 1: |H(j\omega)|^2 \approx (1/K^2) \frac{1}{1}$$

$$\text{For } N = 2: |H(j\omega)|^2 \approx (1/K^2) \frac{1}{1 - 2\phi^2\omega^2}$$

$$\text{For } N = 3: |H(j\omega)|^2 \approx (1/K^2) \frac{1 + \phi^2\omega^2}{1 - \phi^2\omega^2}, \text{ etc.}$$

Approximations to $H(s)H(-s)$ were obtained by substituting $s = j\omega$.

$$\text{For } N = 1: H(s)H(-s) \approx (1/K^2) \frac{1}{1}$$

$$\text{For } N = 2: H(s)H(-s) \approx (1/K^2) \frac{1}{1 + 2\phi^2s^2}$$

$$\text{For } N = 3: H(s)H(-s) \approx (1/K^2) \frac{1 - \phi^2s^2}{1 + \phi^2s^2}, \text{ etc.}$$

The poles and zeros for the approximations to $H(s)H(-s)$ for $N = 2$ through $N = 9$ are contained in Table I. The numerical values of these poles and zeros, that is, the values prior to being multiplied by $1/\phi$ will hereafter be referred to as the pseudo poles and zeros.

N	ZEROS	POLES
2		$\frac{1}{\phi}(0 \pm j0.70710678)$
3	$\frac{1}{\phi}(\pm 1 + j0)$	$\frac{1}{\phi}(0 \pm j1)$
4	$\frac{1}{\phi}(\pm 1.22474487 + j0)$	$\frac{1}{\phi}(\pm 0.3352200067 \pm j1.054690705)$
5	$\frac{1}{\phi}(\pm 1.271229882 \pm j0.340625032)$	$\frac{1}{\phi}(\pm 0.340625032 \pm j1.271229882)$
6	$\frac{1}{\phi}(\pm 1.45534667 \pm j0.343560805)$	$\frac{1}{\phi}(0 \pm j1.34867233)$
		$\frac{1}{\phi}(\pm 0.587391538 \pm j1.29829514)$
	$\frac{1}{\phi}(\pm 1.52387182 + j0)$	$\frac{1}{\phi}(0 \pm j1.52387182)$
7	$\frac{1}{\phi}(\pm 1.47994076 \pm j0.5927200616)$	$\frac{1}{\phi}(\pm 0.5927200616 \pm j1.47994076)$
	$\frac{1}{\phi}(\pm 1.680548372 + j0)$	$\frac{1}{\phi}(\pm 0.79700116 \pm j1.497202089)$
8	$\frac{1}{\phi}(\pm 1.64112335 \pm j0.59616025)$	$\frac{1}{\phi}(\pm 0.250044024 \pm j1.56720106)$
	$\frac{1}{\phi}(\pm 1.6572801 \pm j0.801741003)$	$\frac{1}{\phi}(\pm 0.801741003 \pm j1.6572801)$
9	$\frac{1}{\phi}(\pm 1.72038868 \pm j0.252045949)$	$\frac{1}{\phi}(\pm 0.252045949 \pm j1.72038868)$

TABLE I. Poles and Zeros of $H(s)H(-s)$ for $N = 2$ through 9.

II. COMPUTER SIMULATION

In order to effectively analyze the Sierra Filter, several small-scale, special-purpose computer programs were required. Most of these programs were written to provide an output over a range of compression ratios for a given input. In addition to providing an optimum compression ratio for a given transfer function, this procedure permitted the rapid evaluation of design parameters for a specified compression ratio. Computer programs were written to provide design information for all the Sierra transfer functions from $N = 2$ through $N = 9$ using both right-hand-plane (RHP) zeros and left-hand-plane (LHP) poles and LHP zeros and LHP poles. In some instances, therefore, minor modifications to the basic program were required. These modifications, of course, depended upon the specific transfer function being considered. A brief description of the pertinent computer programs with the required modifications will now be delineated.

A. PROGRAM NUMBER ONE

This program generated sample values of the normalized input Gaussian pulse for various values of standard deviation. The mean was chosen to be zero for convenience.

The output of this program provided the time, standard deviation, and value of the normalized pulse. A plot of the input pulse was also provided.

The primary purpose of this program was to provide the standard deviation for a specified pulse width and truncation error. This program, however, also provided sample values of the input pulse which could be used in spectrum analysis or convolution routines as required.

B. PROGRAM NUMBER TWO

This program generated the poles and zeros for all the Sierra transfer functions listed in Table I for values of compression ratio varying from 1.1 through 10.0 in steps of 0.1. In effect, this program performed the iterative multiplication suggested in columns 2 and 3 of Table I. The output, of course, provided the required poles and zeros.

C. PROGRAM NUMBER THREE

This program computed and plotted the magnitude and phase of the Sierra transfer functions for values of compression ratio ranging from 1.2 through 3.0 in steps of 0.1.

The inputs to this program were the pseudo poles and zeros listed in Table I and the desired compression ratios. The required iterations were performed by the program.

The output of this program provided the compression ratio, the corresponding value of omega in radians and in degrees, and the corresponding magnitude and phase. A rapid means of comparing magnitude and phase for specified compression ratios was made possible by this program output.

Minor modifications to this program were required for each transfer function considered. These modifications related to the manner in which the transfer function was generated and will be included herein for convenience.

1. Modification for N = 2

```
DO 70 I=1,100
PSI=2.*TK/SQRT(TK**2-1.)
S1=(0.0,0.70710678+W/PSI)
S2=(0.0,-0.70710678+W/PSI)
P1=PSI*S1
P2=PSI*S2
W1=(1.0,0.0)
W2=P1*P2
T=(1./TK)*(W1/W2)
```

2. Modification for N = 3

```
DO 70 I=1,100
PSI=2.*TK/SQRT(TK**2-1.)
R1=(-1.0,W/PSI)
S1=(0.0,1.0+W/PSI)
S2=(0.0,-1.0+W/PSI)
P1=PSI*S1
P2=PSI*S2
W1=Z1
W2=P1*P2
T=(1./TK)*(W1/W2)
```

3. Modification for N = 4

```
DO 70 I=1,100
PSI=2.*TK/SQRT(TK**2-1.)
R1=(-1.22474487,W/PSI)
S1=(0.33522001,1.05469071+W/PSI)
S2=(0.33522001,-1.05469071+W/PSI)
P1=PSI*S1
P2=PSI*S2
P3=(1.0,0.0)
P4=(1.0,0.0)
Z1=PSI*R1
Z2=(1.0,0.0)
Z3=(1.0,0.0)
Z4=(1.0,0.0)
W1=Z1*Z2*Z3*Z4
W2=P1*P2*P3*P4
T=(1./TK)*(W1/W2)
```

4. Modification for N = 5

```
DO 70 I=1,100
PSI=2.*TK/SQRT(TK**2-1.)
R1=(-1.27122988,0.34062503+W/PSI)
R2=(-1.27122988,-0.34062503+W/PSI)
S1=(0.34062503,1.27122988+W/PSI)
S2=(0.34062503,-1.27122988+W/PSI)
P1=PSI*S1
P2=PSI*S2
P3=(1.0,0.0)
P4=(1.0,0.0)
Z1=PSI*R1
Z2=PSI*R2
Z3=(1.0,0.0)
Z4=(1.0,0.0)
W1=Z1*Z2*Z3*Z4
W2=P1*P2*P3*P4
T=(1./TK)*(W1/W2)
```

5. Modification for N = 6

```
DO 70 I=1,100
PSI=2*TK/SQRT(TK**2-1.)
R1=(-1.45534667,0.34356080+W/PSI)
R2=(-1.45534667,-0.34356080+W/PSI)
S1=(0.0,1.34867233+W/PSI)
S2=(0.0,-1.34867233+W/PSI)
S3=(0.58739154,1.2982951+W/PSI)
S4=(0.58739154,-1.2982951+W/PSI)
P1=PSI*S1
P2=PSI*S2
P3=PSI*S3
P4=PSI*S4
Z1=PSI*R1
Z2=PSI*R2
Z3=(1.0,0.0)
Z4=(1.0,0.0)
W1=Z1*Z2*Z3*Z4
W2=P1*P2*P3*P4
T=(1./TK)*(W1/W2)
```

6. Modification for N = 7

```
DO 70 I=1,100
PSI=2.*TK/SQRT(TK**2-1.)
R1=(-1.52387182,0.0+W/PSI)
R2=(-1.47994076,0.59272006+W/PSI)
R3=(-1.47994076,-0.59272006+W/PSI)
S1=(0.0,1.52387182+W/PSI)
S2=(0.0,-1.52387182+W/PSI)
S3=(0.59272006,1.47994076+W/PSI)
S4=(0.59272006,-1.47994076+W/PSI)
P1=PSI*S1
P2=PSI*S2
P3=PSI*S3
P4=PSI*S4
Z1=PSI*R1
Z2=PSI*R2
Z3=PSI*R3
Z4=(1.0,0.0)
W1=Z1*Z2*Z3*Z4
W2=P1*P2*P3*P4
T=(1./TK)*(W1/W2)
```

7. Modification for N = 9

```
DO 70 I=1,100
PSI=2.*TK/SQRT(TK**2-1.)
R1=(-1.6572801,0.80174100+W/PSI)
R2=(-1.6572801,-0.80174100+W/PSI)
R3=(-1.72038868,0.25204595+W/PSI)
R4=(-1.72038868,-0.25204595+W/PSI)
```

```

S1=(0.80174100,1.6572801+W/PSI)
S2=(0.80174100,-1.6572801+W/PSI)
S3=(0.25204595,1.72038868+W/PSI)
S4=(0.25204595,-1.72038868+W/PSI)
P1=PSI*S1
P2=PSI*S2
P3=PSI*S3
P4=PSI*S4
Z1=PSI*R1
Z2=PSI*R2
Z3=PSI*R3
Z4=PSI*R4
W1=Z1*Z2*Z3*Z4
W2=P1*P2*P3*P4
T=(1./TK)*(W1/W2)

```

D. PROGRAM NUMBER FOUR

This program computed the maximum magnitudes of the Sierra transfer functions and the corresponding values of omega to five decimal places for values of compression ratio ranging from 1.2 through 3.0.

The inputs to this program were the pseudo poles and zeros listed in Table I, the desired compression ratio and the values of delta, the step size for omega. This program computed the maximum magnitude as follows: Initially, the step size for omega was 0.1. When the maximum was passed, i.e., omega (I) less than omega (I-1), the computer went to omega (I-2), changed the step size to 0.01 and continued the process. Any desired degree of accuracy could thus be obtained employing this technique.

In many instances, it is necessary to multiply the original transfer function, $H(s)$, by some constant in order to realize it. Specifically, in order to realize the constant-resistance lattice network, the requirement that $|H(j\omega)|$ be less than or equal to unity must be met. This program, therefore, provides an accurate value of the maximum magnitude so that these requirements can be fulfilled.

Since this program generated the transfer functions in the same manner as Program Number Three, the same basic modifications are applicable. These modifications will not be reiterated.

E. PROGRAM NUMBER FIVE

This program computes the numerator/denominator coefficients of the Sierra transfer functions for values of compression ratio ranging from 1.1 through 10.0 by performing the iterative multiplication of the factored terms in the numerator and denominator.

The input to this program was the pseudo poles and zeros. The output provided the required coefficients. This program was written to provide the coefficients using either LHP zeros or RHP zeros. Thus, the sign of the pseudo poles and zeros and the indicator value determined which coefficients were computed for a specified transfer function.

Program Number Two generated the poles and zeros for the Sierra transfer functions and provided the transfer functions in the form

$$H(s) = (1/K) \frac{(s+z_1)(s+z_2) \dots (s+z_n)}{(s+p_1)(s+p_2) \dots (s+p_m)} \quad (2.1)$$

Program Number Five, however, provided the transfer functions in the form

$$H(s) = (1/K) \frac{a_n s^n + a_{n-1} s^{n-1} + \dots + a_0}{b_m s^m + b_{m-1} s^{m-1} + \dots + b_0} \quad (2.2)$$

Program Number Five was written, therefore, to provide additional design flexibility, since some network analysis programs can only accept transfer functions appearing in the form of (2.1) while others can only accept them in the form of (2.2). Also, analysis and synthesis frequently

demand that both the poles and zeros and the coefficients of the transfer functions be available.

Minor modifications to this program were required in order to calculate the numerator and denominator coefficients for all the transfer functions. Since the same program was used to generate both the numerator and denominator coefficients, the FORMAT statements and the manner in which the coefficients were generated required modification.

1. Modifications Required for Denominator Coefficients

In order to label the output properly, the two FORMAT statements utilizing the Hollerith field must be changed when generating the denominator coefficients.

2. Other Required Modifications

By inspecting the poles and zeros contained in Table I, it is evident that the numerator and denominator coefficients for $N = 9$ and the denominator coefficients for $N = 8$ are generated in the same manner. These coefficients were generated as follows:

```
70 PSI=2.*TK/SQRT(TK**2-1.)
   A1=((A**2)+(B**2))
   A2=((C**2)+(D**2))
   S0=(PSI**4)*A1*A2
   S1=(2.0*(PSI**3))*(C*A1+A*A2)
   S2=(PSI**2)*(A1+A2+4.0*A*C)
   S3=2.0*PSI*(A+C)
```

The numerator coefficients for $N = 8$ and $N = 7$ were generated in the same manner as shown in Program Number Five.

The denominator coefficients for $N = 7$ and $N = 6$ were generated using the following modification:

```
70 PSI=2.*TK/SQRT(TK**2-1.)
   A1=((C**2)+(D**2))
   S0=(PSI**4)*((B**2)*A1)
   S1=(PSI**3)*2.0*(B**2)*C
   S2=(PSI**2)*((B**2)+A1)
   S3=PSI*2.0*C
```

The numerator coefficients for $N = 6$, the numerator and denominator coefficients for $N = 5$, and the denominator coefficients for $N = 4$ were generated as follows:

```
70 PSI=2.*TK/SQRT(TK**2-1.)
   A1=((A**2)+(B**2))
   S0=(PSI**2)*A1
   S1=PSI*2.0*A
   S2=1.0
   S3=0.0
```

The appropriate modification to this program, therefore, yielded the desired coefficients for the Sierra transfer functions.

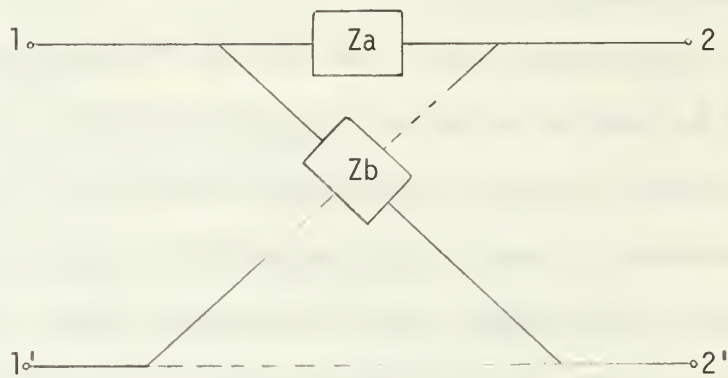
F. PROGRAM NUMBER SIX

This program makes use of the DSL/360 (Digital Simulation Language) [Ref. 3] to provide the output for a given Gaussian input pulse and a given transfer function. The standard deviation of the input pulse can be varied several times in the same program, thus permitting a rapid comparison of the outputs versus input pulse width. Other parameters such as compression ratio and mean can also be varied as desired.

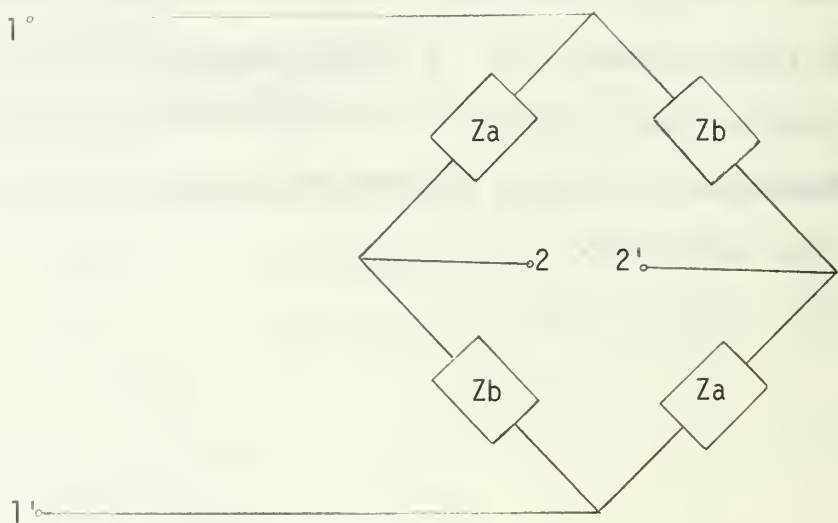
The inputs to this program were the numerator and denominator coefficients of the transfer functions. These coefficients were generated by Program Number Five. A FORTRAN subprogram was written to generate the Gaussian input pulse with a truncation error of one percent. Program Number One, of course, provided the required standard deviation for a given pulse width.

III. SOME ASPECTS OF LATTICE SYNTHESIS

In analyzing the Sierra Filter, synthesis was restricted to the symmetrical lattice. Similar synthesis procedures, however, could be applied to ladder development. In fact, in many cases, the symmetrical lattice can be decomposed into an equivalent ladder network [Ref. 4]. Figure 1 below shows the standard form of the symmetrical lattice and the bridge form, respectively.



(a). Standard Form



(b). Bridge Form

Figure 1. The Symmetrical Lattice

A. THE UNLOADED LATTICE

Referring to Figure 1a, it is evident that

$$V_{11'} = Z_{11}I_1 + Z_{12}I_2 \quad (3.1)$$

$$\text{and } V_{22'} = Z_{21}I_1 + Z_{22}I_2 \quad (3.2)$$

$$\text{Thus, } Z_{11} = \left. \frac{V_{11'}}{I_1} \right|_{I_2=0} \quad (3.3)$$

$$\text{and } Z_{12} = \left. \frac{V_{11'}}{I_2} \right|_{I_1=0} \quad (3.4)$$

Since the symmetrical lattice is being considered, $Z_{11}=Z_{22}$ and $Z_{12}=Z_{21}$. By applying elementary network analysis, it can be shown that

$$Z_{11} = Z_{22} = 1/2 (Z_a + Z_b) \quad (3.5)$$

$$\text{and } Z_{12} = Z_{21} = 1/2 (Z_b - Z_a) \quad (3.6)$$

Consider now the transfer function, $H(s)$, defined as follows:

$$H(s) = \left. \frac{V_{22'}(s)}{V_{11'}(s)} \right|_{I_2=0} \quad (3.7)$$

or, from (3.1) and (3.2)

$$H(s) = \frac{Z_{21}}{Z_{22}} \quad (3.8)$$

Substituting (3.5) and (3.6) into (3.8) yields

$$H(s) = \frac{1/2 (Z_b - Z_a)}{1/2 (Z_a + Z_b)} = \frac{(Z_b - Z_a)}{(Z_b + Z_a)} \quad (3.9)$$

Dividing numerator and denominator of (3.9) by Z_b , the transfer function can be written as

$$H(s) = \frac{1 - Z_a/Z_b}{1 + Z_a/Z_b} \quad (3.10)$$

Solving for Z_a/Z_b

$$Z_a/Z_b = \frac{1 - H(s)}{1 + H(s)} \quad (3.11)$$

$$\text{Thus, } Z_a = 1 - H(s) \quad (3.12)$$

$$\text{and } Z_b = 1 + H(s) \quad (3.13)$$

are synthesis possibilities for the unloaded symmetrical lattice.

B. THE CONSTANT-RESISTANCE LATTICE

The symmetrical lattice terminated in a load of constant resistance R and possessing a driving point impedance of R at the input terminals will now be considered. Figure 2 below depicts this situation.

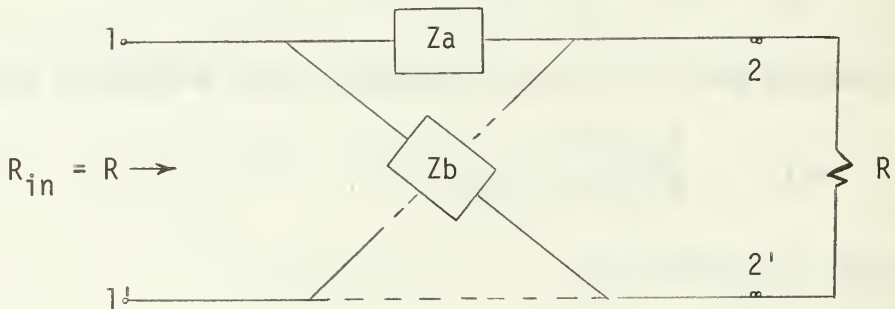


Figure 2. Symmetrical lattice terminated in a resistance R with an input impedance of R .

Using Eqs. (3.1) and (3.2) and noting the fact that $V_{22'} = -I_2 R$, the following equations are obtained:

$$V_{11'} = Z_{11} I_1 + Z_{12} I_2 \quad (3.14)$$

$$\text{and } -I_2 R = Z_{21} I_1 + Z_{22} I_2 \quad (3.15)$$

Solving for I_2 in (3.15) and substituting into (3.14) yields

$$V_{11'} = Z_{11}I_1 + Z_{12} \left(\frac{-Z_{21}I_1}{Z_{22} + R} \right) . \quad (3.16)$$

Solving (3.16) for $\frac{V_{11'}}{I_1}$, the driving-point impedance is obtained.

$$Z_{dp}(s) = \frac{V_{11'}}{I_1} = Z_{11} - \frac{Z_{12}Z_{21}}{Z_{22} + R} . \quad (3.17)$$

Since $Z_{11} = Z_{22}$ and $Z_{12} = Z_{21}$, Eq. (3.17) can be written as follows:

$$Z_{dp}(s) = \frac{Z_{11}^2 - Z_{12}^2 + Z_{11}R}{Z_{11} + R} . \quad (3.18)$$

Thus, for the driving-point impedance to equal R ohms, it is required that the right-hand portion of Eq. (3.18) equal R . In other words

$$Z_{11}^2 - Z_{12}^2 + Z_{11}R = Z_{11}R + R^2 \quad (3.19)$$

$$\text{or } Z_{11}^2 = Z_{12}^2 + R^2 . \quad (3.20)$$

Substituting (3.5) and (3.6) into (3.20) yielded

$$1/4 (Z_a^2 + 2Z_aZ_b + Z_b^2) = R^2 + 1/4 (Z_b^2 - 2Z_aZ_b + Z_a^2) \quad (3.21)$$

$$\text{or } Z_aZ_b = R^2 .$$

If R is normalized to 1 ohm, then Eq. (3.22) implies that, in order for the driving-point impedance to equal the constant value 1, Z_a and Z_b must be reciprocal impedances.

Using the normalized value for R and noting the fact that $V_{22'} = -I_2$, the following relations were obtained from Eqs. (3.1) and (3.2):

$$V_{11}' = Z_{11}I_1 - Z_{12}V_{22}' \quad (3.23)$$

$$\text{and } V_{22}' = Z_{21}I_1 - Z_{22}V_{22}' \quad (3.24)$$

Again, using the relations $Z_{11} = Z_{22}$ and $Z_{12} = Z_{21}$, the voltage transfer function defined by

$$H(s) = \frac{V_{22}'}{V_{11}'} \quad (3.25)$$

was found. Multiplying (3.23) by Z_{12} and (3.24) by Z_{11} and subtracting yielded

$$(Z_{11}^2 - Z_{12}^2 + Z_{11}) V_{22}' = Z_{12}V_{11}' \quad (3.26)$$

$$\text{or } H(s) = \frac{V_{22}'}{V_{11}'} = \frac{Z_{12}}{(Z_{11}^2 - Z_{12}^2 + Z_{11})} \quad (3.27)$$

Substituting (3.5) and (3.6) into (3.27) gave

$$H(s) = \frac{Z_b - Z_a}{Z_a + 2Z_aZ_b + Z_b} \quad (3.28)$$

Since R is normalized to 1 ohm, $Z_b = 1/Z_a$ was substituted into Eq. (3.28) obtaining

$$H(s) = \frac{1 - Z_a}{1 + Z_a} \quad (3.29)$$

$$\text{or } Z_a = \frac{1 - H(s)}{1 + H(s)} \quad (3.30)$$

$$\text{and } Z_b = 1/Z_a \quad (3.31)$$

thus yielding a synthesis possibility for the constant-resistance symmetrical lattice.

The restrictions that must be placed on $H(s)$ in order for Z_a to be realizable will now be examined.

First of all, $H(s)$ can have no poles in the RHP. This restriction, of course, applies to all output-to-input transfer functions. Secondly, $|H(j\omega)|$ must be bounded and less than or equal to 1. A cogent discussion of the second requirement is given in Ref. 5. Thus, for Z_a and Z_b to be positive real and, therefore, realizable, the following requirements must be met:

- (1) $H(j\omega)$ has no poles in the RHP or on the $j\omega$ axis.
- (2) $|H(j\omega)| \leq 1$.

Baycura [Ref. 6] outlines in detail a method for realizing the Sierra transfer functions as constant-resistance symmetrical lattice networks using the relations given in Eqs. (3.30) and (3.31). In fact, Ref. 6 contains the realization of the Sierra transfer function for $N = 9$ and $K = 2$. Similar techniques will now be applied in realizing the Sierra transfer function for $N = 8$ and $K = 2$ using the relations given in Eqs. (3.12) and (3.13).

IV. SYNTHESIS OF THE SIERRA TRANSFER FUNCTION FOR N=8 AND K=2

Employing Eqs. (3.12) and (3.13) as a synthesis possibility for the unloaded symmetrical lattice, synthesis techniques for the Sierra transfer function for $N = 8$ and $K = 2$ were demonstrated. The outputs of the computer programs discussed in Chapter II were used extensively and greatly simplified the synthesis task. It is noteworthy that the synthesis techniques outlined herein can be readily applied to any of the Sierra transfer functions, the poles of which are located in the left-hand plane.

A. SYNTHESIZING THE IMPEDANCE $Z_a(s)$

The transfer function for $N = 8$ and $K = 2$ was divided by $H(s)_{MAX}$ to obtain

$$H^*(s) = \frac{H(s)}{H(s)_{MAX}} = \frac{a_3s^3 + a_2s^2 + a_1s + a_0}{(s^2 + X1s + X2)(s^2 + X3s + X4)} \quad (4.1)$$

The technique of resistance padding discussed by Guillemin [Ref. 7] could have been used, thus alleviating the necessity of dividing $H(s)$ by $H(s)_{MAX}$. This technique, however, can also be applied after the transfer function is normalized.

$Z_a(s)$, as given in Eq. (3.12) can now be written as

$$Z_a(s) = 1 - H^*(s) = 1 - \frac{N(s)}{D(s)} \quad (4.2)$$

or

$$Z_a(s) = \frac{D(s) - N(s)}{D(s)} \quad (4.3)$$

Writing $Z_a(s)$ as a ratio of polynomials

$$Z_a(s) = \frac{A_4s^4 + A_3s^3 + A_2s^2 + A_1s + A_0}{B_4s^4 + B_3s^3 + B_2s^2 + B_1s + B_0} \quad (4.4)$$

Program Number Two provided the poles and zeros for the original transfer function, $H(s)$. Since the poles of $Z_a(s)$ given in Eq. (4.4) are the same as the poles of the original transfer function, (4.4) was expanded in partial fractions as follows:

$$Z_a(s) = \frac{As^2 + B}{s^2 + X_1s + X_2} + \frac{Cs^2 + D}{s^2 + X_3s + X_4} + E \quad (4.5)$$

Using the output of Program Number Two for $N = 8$ and $K = 2$, the factorization required by (4.5) was performed and the values of X_1 , X_2 , X_3 , and X_4 were thus obtained.

Equations (4.4) and (4.5) are identities and must, therefore, be equal for all values of s . After equating (4.4) and (4.5) and clearing fractions, five equations in five unknowns were obtained. Program Number Five provided the denominator coefficients of Eq. (4.4), thereby simplifying the task of clearing fractions.

An existing subroutine was used to solve the above equations. The results were:

$$\begin{aligned} A &= 0.47590151 \approx 0.48 \\ B &= 13.81737083 \approx 13.82 \\ C &= -1.18232859 \approx -1.18 \\ D &= -20.97217061 \approx -20.97 \\ E &= 1.70642708 \approx 1.71 \end{aligned}$$

Substituting these values into Eq. (4.5) yielded

$$Z_a(s) = \frac{0.48s^2 + 13.82}{s^2 + 3.68s + 15.34} + \frac{-1.18s^2 - 20.97}{s^2 + 1.15s + 13.43} + 1.71 \quad (4.6)$$

$$\text{or } Z_a(s) = Z_1(s) + Z_2(s) + Z_3(s) \quad (4.7)$$

The equivalent impedance of a series LC circuit shunted by a resistance R was found to be

$$Z(s) = \frac{Rs^2 + R/LC}{s^2 + R/Ls + 1/LC} \quad (4.8)$$

Thus, the first term in (4.6), $Z_1(s)$, can be realized as a series LC circuit shunted by a resistance R.

The second and third terms of (4.6) were written as follows:

$$Z_2(s) + Z_3(s) = \frac{-1.18s^2 - 20.97 + 1.71(s^2 + 1.15s + 13.43)}{s^2 + 1.15s + 13.43} \quad (4.9)$$

Simplifying yielded

$$Z_2(s) + Z_3(s) = \frac{0.53s^2 + 1.975 + 2.03}{s^2 + 1.15s + 13.43} \quad (4.10)$$

Equation (4.10) was written as follows:

$$Z_2(s) + Z_3(s) = \frac{0.53s^2 + 2.03}{s^2 + 1.15s + 13.43} + \frac{1.97s}{s^2 + 1.15s + 13.43} \quad (4.11)$$

$$\text{or } Z_2(s) = \frac{0.53s^2 + 2.03}{s^2 + 1.15s + 13.43} \quad (4.12)$$

$$\text{and } Z_3(s) = \frac{1.97s}{s^2 + 1.15s + 13.43} \quad (4.13)$$

$Z_2(s)$ is now in the form of Eq. (4.8) and can, therefore, be realized as a series LC circuit shunted by a resistance.

The equivalent impedance of a parallel RLC circuit was found to be

$$Z(s) = \frac{(1/C) s}{s^2 + (1/RC)s + (1/LC)} \quad (4.14)$$

Equation (4.13) has the same form as (4.14) and can, therefore, be realized as a parallel RLC circuit. Thus, $Z_a(s)$ can be realized as three RLC circuits in series. Similar synthesis techniques can be applied to obtain a realization for $Z_b(s)$.

V. CONCLUSIONS

The Sierra Filters, like the linear phase filters discussed by Kuo [Ref. 8], produce increasingly complex transfer functions as the number of terms in the continued fraction expansion is increased.

Several computer programs were written to provide a host of useful design parameters for virtually all the Sierra transfer functions ranging from $N = 2$ through $N = 9$ with the compression ratio and frequency varying over a significant range. Although these programs do not provide all the design parameters that may be required, they frequently simplify the task of obtaining others.

The use of small-scale programs that are designed to provide an output over a wide range of parameter variations can sometimes be more effective than the large-scale programs, some of which seem to handle parameter variations awkwardly and provide redundant information.

Some synthesis techniques were delineated, primarily to demonstrate the value of these small-scale programs in synthesizing the transfer functions and, secondly, to demonstrate a synthesis technique that is relatively simple and can be applied to all the Sierra transfer functions that are realizable using passive elements.

```

C
C   PROGRAM NUMBER ONE
C
C   ---TAYLOR---THESIS
C
C   THIS PROGRAM GENERATES SAMPLE VALUES OF INPUT
C   GAUSSIAN PULSE FOR VARIOUS VALUES OF STANDARD
C   DEVIATION (SIGMA). THE MEAN IS EQUAL TO ZERO FOR
C   CONVENIENCE.
C
C   DIMENSION A(61),B(61),D(61,2),JXY(2)
C   Z=1.00000000
10  WRITE (6,20)
20  FORMAT (1H1,4HTIME,7X,5HSIGMA,8X,8HFUNCTION,/)
C   T=-3.00000000
C
C   CALCULATE THE INPUT PULSE VALUE AS A FUNCTION OF
C   TIME AND ITERATE THE VALUE OF SIGMA.
C   DO 40 I=1,61
C   X=-(Z*T)**2
C   F=EXP(X)
C   A(I)=F
C   B(I)=T
C   SIGMA=1./(SQRT(2.)*Z)
C
C   PRINT OUT TIME, STANDARD DEVIATION AND VALUE OF
C   INPUT PULSE.
C
C   WRITE (6,30) T,SIGMA,F
30  FORMAT (1H ,F4.1,2F14.8)
C   T=T+0.100000000
40  CONTINUE
C
C   STORE VALUES IN A MANNER ACCEPTABLE TO THE PLOT
C   ROUTINE.
C
C   DO 50 I=1,61
C   D(I,1)=B(I)
50  D(I,2)=A(I)
C   JXY(1)=1
C   JXY(2)=2
C
C   PLOT THE INPUT PULSE VERSUS TIME FOR VARIOUS VALUES
C   OF STANDARD DEVIATION.
C
C   CALL VPLOT (D,JXY,61,61,1,0,XMIN,XMAX,YMIN,YMAX)
C   Z=Z+0.100000000
C   IF(Z.EQ.4.0) GO TO 70
60  GO TO 10
70  STOP
C   END

```

```
FUNCTION F(X)  
F=EXP(X)  
RETURN  
END
```

```

C
C   PROGRAM NUMBER TWO
C
C   ---TAYLOR---THESIS
C
C   THIS PROGRAM GENERATES THE POLES AND ZEROS FOR THE
C   SIERRA TRANSFER FUNCTIONS FOR VALUES OF COMPRESSION
C   RATIO (TK) VARYING FROM 1.1 THROUGH 10.0 IN
C   INCREMENTS OF 0.1.
C
C   N=8
C
C   COMPLEX P1,P2,Z1,Z2,S1,S2,R1,R2
C   WRITE (6,10)
10  FORMAT (1H1,1X,2HTK,6X,3HPHI,14X,5HPOLES,19X,
15HPOLES,19X,5HZEROS,19X,5HZEROS,/)
C
C   READ IN THE PSEUDO POLES AND ZEROS FOR THE
C   TRANSFER FUNCTION.
C
C   R1=(1.68054837,0.00000000)
C   R2=(1.64112335,0.59616025)
C   S1=(0.79700116,1.49720209)
C   S2=(0.25004402,1.56720106)
C   TK=1.1
C
C   CALCULATE THE POLES AND ZEROS.
C
C   DO 30 I=1,90
C   PHI=2.*TK/SQRT(TK**2-1.)
C   P1=PHI*S1
C   P2=PHI*S2
C   Z1=PHI*R1
C   Z2=PHI*R2
C
C   PRINT OUT THE POLES AND ZEROS AND THE CORRESPONDING
C   COMPRESSION RATIO.
C
C   WRITE (6,20) TK,PHI,P1,P2,Z1,Z2
20  FORMAT (1H ,F3.1,9F12.8)
C   TK=TK+0.1
30  CONTINUE
C   STOP
C   END

```

```

C
C   PROGRAM NUMBER THREE
C
C   ---TAYLOR---THESIS
C
C   THIS PROGRAM COMPUTES AND PLOTS THE MAGNITUDE AND
C   PHASE OF THE SIERRA TRANSFER FUNCTIONS FOR VALUES
C   OF COMPRESSION RATIO VARYING FROM 1.2 THROUGH 3.0
C   AND OMEGA VARYING FROM ZERO TO TEN RADIAN PER SEC.
C
      COMPLEX T,R1,R2,R3,R4,S1,S2,S3,P1,P2,P3,P4,Z1,S4,
      1Z2,Z3,Z4,W1,W2
      DIMENSION A(100),B(100),C(100),D(100,2),E(100,2),
      1JXY(2)
10  READ(5,20) TK
20  FORMAT (F10.9)
      N=8
      WRITE (6,30) N
30  FORMAT (1H1,29HMAG AND PHASE VS OMEGA FOR N=,I1,//1X,
      12HTK,6X,10HOMEGA (RAD),5X,10HOMEGA (DEG),7X,
      19HMAGNITUDE,9X,5HPHASE,/)
      W=0.0000000

C
C   GENERATE THE SIERRA TRANSFER FUNCTION IN A MANNER
C   THAT PERMITS ITERATION OF OMEGA AND COMPRESSION RATIO.
C
      DO 70 I=1,100
      PSI=2.*TK/SQRT(TK**2-1.)
      R1=CMPLX(-1.68054837,0.00000000+W/PSI)
      R2=CMPLX(-1.64112335,0.59616025+W/PSI)
      R3=CMPLX(-1.64112335,-0.59616025+W/PSI)
      S1=CMPLX(0.79700116,1.49720209+W/PSI)
      S2=CMPLX(0.79700116,-1.49720209+W/PSI)
      S3=CMPLX(0.25004402,1.56720106+W/PSI)
      S4=CMPLX(0.25004402,-1.56720106+W/PSI)
      P1=PSI*S1
      P2=PSI*S2
      P3=PSI*S3
      P4=PSI*S4
      Z1=PSI*R1
      Z2=PSI*R2
      Z3=PSI*R3
      Z4=CMPLX(1.0,0.0)
      W1=Z1*Z2*Z3*Z4
      W2=P1*P2*P3*P4
      T=(1./TK)*(W1/W2)

C
C   COMPUTE THE MAGNITUDE OF THE TRANSFER FUNCTION.
C
      TMAG=CABS(T)
      A(I)=TMAG

```

```

C
C   COMPUTE THE PHASE OF THE TRANSFER FUNCTION.
C
      PHASE=57.29578*(ATAN2(AIMAG(W1),REAL(W1))
3-ATAN2(AIMAG(W2),REAL(W2)))
C
C   APPLY SCALE FACTOR TO PHASE.
50  C(I)=PHASE/10.0
C
      OMEGA=57.29578*W
      B(I)=OMEGA
C
C   PRINT OUT COMPRESSION RATIO, OMEGA IN RADIANS AND
C   IN DEGREES, MAGNITUDE AND PHASE.
C
      WRITE(6,60) TK,W,OMEGA,TMAG,PHASE
60  FORMAT(1H ,F3.1,4F16.8)
      W=W+0.1
70  CONTINUE
      J=25
      NDIM=100
C
C   PLOT THE MAGNITUDE AND PHASE VERSUS OMEGA.
C
      DO 80 I=1,J
      D(I,1)=B(4*I-1)
      D(I,2)=A(4*I-1)
80  CONTINUE
      JXY(1)=1
      JXY(2)=2
      CALL VPLOT(D,JXY,J,NDIM,1,0,XMIN,XMAX,YMIN,YMAX)
      DO 90 I=1,J
      E(I,1)=D(I,1)
      E(I,2)=C(4*I-1)
90  CONTINUE
      CALL VPLOT(E,JXY,J,NDIM,1,0,XMIN,XMAX,YMIN,YMAX)
      IF(TK.EQ.3.0) GO TO 100
      GO TO 10
100 STOP
      END

```



```

C
C   PROGRAM NUMBER FOUR
C
C   ---TAYLOR---THESIS
C
C   THIS PROGRAM COMPUTES THE MAXIMUM MAGNITUDE OF THE
C   SIERRA TRANSFER FUNCTIONS FOR VALUES OF
C   COMPRESSION RATIO RANGING FROM 1.2 THROUGH 3.0.
C
C   COMPLEX T,R1,R2,R3,R4,S1,S2,S3,S4,P1,P2,P3,P4,Z1,
1  Z2,Z3,Z4,W1,W2
C   DIMENSION A(500),B(500),C(500)
C   N=8
C   WRITE (6,40) N
40  FORMAT (1H1,27HHMAX VS OMEGA AND TK FOR N=,I1, //2X,
1  2HTK,7X,10HOMEGA(RAD),6X,10HOMEGA(DEG),13X,
1  18HMAG(MAX),/)
C
C   READ IN DELTA, TK, AND J. DELTA IS THE STEP SIZE
C   FOR OMEGA, TK IS THE COMPRESSION RATIO AND J IS AN
C   INDICATOR.
C
10  READ(5,20) DELTA,TK,J
20  FORMAT(2F10.9,I1)
30  W=0.00000000
C   B(1)=0.00000000
50  I=2
C
C   GENERATE THE TRANSFER FUNCTIONS IN A MANNER THAT
C   PERMITS ITERATION OF COMPRESSION RATIO AND OMEGA.
C
60  PSI=(2.*TK)/SQRT((TK**2)-1.)
R1=CMPLX(-1.68054837,0.00000000+W/PSI)
R2=CMPLX(-1.64112335,0.59616025+W/PSI)
R3=CMPLX(-1.64112335,-0.59616025+W/PSI)
S1=CMPLX(0.79700116,1.49720209+W/PSI)
S2=CMPLX(0.79700116,-1.49720209+W/PSI)
S3=CMPLX(0.25004402,1.56720106+W/PSI)
S4=CMPLX(0.25004402,-1.56720106+W/PSI)
P1=PSI*S1
P2=PSI*S2
P3=PSI*S3
P4=PSI*S4
Z1=PSI*R1
Z2=PSI*R2
Z3=PSI*R3
Z4=CMPLX(1.0,0.0)
W1=Z1*Z2*Z3*Z4
W2=P1*P2*P3*P4
T=(1./TK)*(W1/W2)

```



```

C      COMPUTE THE MAGNITUDE OF THE TRANSFER FUNCTION.
C
      TMAG=CABS(T)
      A(I)=W
      C(I)=57.29587*A(I)
      B(I)=TMAG
C
C      COMPUTE THE MAXIMUM MAGNITUDE BY CONTINUALLY
C      PASSING THROUGH IT AND REDUCING THE STEP SIZE
C      FOR OMEGA.
C
      IF(B(I)-B(I-1)) 80,70,70
70     W=W+DELTA
      I=I+1
      GO TO 60
80     W=A(I-2)
      READ(5,90) DELTA, TK,J
90     FORMAT(2F10.9,I1)
      IF(J) 130,120,100
C
C      PRINT OUT THE MAXIMUM MAGNITUDE, OMEGA IN DEGREES
C      AND IN RADIANS AND THE COMPRESSION RATIO.
C
100    WRITE(6,110) TK,A(I-1),C(I-1),B(I-1)
110    FORMAT(1H ,F4.1,2F16.8,F22.8)
      GO TO 10
120    I=I+1
      GO TO 60
130    STOP
      END

```

```

C
C      PROGRAM NUMBER FIVE
C
C      ---TAYLOR---THESIS
C
C      THIS PROGRAM COMPUTES THE NUMERATOR/DENOMINATOR
C      COEFFICIENTS OF THE SIERRA TRANSFER FUNCTIONS
C      FOR VALUES OF COMPRESSION RATIO (TK) RANGING
C      FROM 1.1 THROUGH 10.0 IN STEPS OF 0.1.
C
C      READ IN THE PSEUDO POLES AND ZEROS, THE
C      COEFFICIENT OF THE FOURTH POWER OF S AND THE
C      INDICATOR, I.
C
10  READ (5,20) A,B,C,S4,I
20  FORMAT(4F12.8,I1)
    TK=1.1
    N=8
C
C      IF I IS LESS THAN ZERO, RHP ZEROS ARE BEING
C      USED. IF I IS EQUAL TO ZERO, LHP ZEROS
C      ARE BEING USED.
C
C      IF(I-1) 30,50,100
30  WRITE(6,40) N
40  FORMAT(1H1,28HNUMERATOR COEFFICIENTS USING,/1X,
130HRHP ZEROS AND LHP POLES FOR N=,I1, //2X,2HTK,
19X,2HS0,15X,2HS1,15X,2HS2,15X,2HS3,15X,2HS4,/)
C
C      GENERATE THE COEFFICIENTS.
C
70  PSI=(2.*TK)/SQRT(TK**2-1.)
    A1=((B**2)+(C**2))
    S0=(PSI**3)*A*A1
    S1=(PSI**2)*(A1+2.0*A*B)
    S2=PSI*(A+2.0*B)
    S3=1.00000000
C
C      PRINT OUT THE COEFFICIENTS AND CORRESPONDING
C      VALUE OF COMPRESSION RATIO
C

```

```
      WRITE(6,80) TK,S0,S1,S2,S3,S4
80    FORMAT(1H ,F4.1,5F16.8)
      IF(TK.GT.9.9) GO TO 10
90    TK=TK+0.1
      GO TO 70
      STOP
      END
```

```

*          PROGRAM NUMBER SIX
*
*          ---TAYLOR---THESIS
*
*          THIS PROGRAM PROVIDES THE OUTPUT PULSE
*          FOR A SPECIFIED INPUT AND TRANSFER FCN.
*
STORAG  IC(4),NUM(4),DEN(5)
TABLE   IC(1-4)=4*0,NUM(1-4)=0.5,-5.7305,...
        22.8391,-31.5524,DEN(1-5)=1,4.8360,...
        33.02724,67.1682,206.0996
PARAM   TMU=3.0,SIGMA=0.9
        Y=GAUS(TMU,SIGMA,TIME)
        OUT=TRNFR(3,4,IC,NUM,DEN,Y)
CONTRL  FINTIM=8,DELT=0.1
PRINT   0.1,Y,OUT,DELT
PREPAR  0.1,OUT
GRAPH   TIME,OUT
PRPLOT  ONLY
END
PARAM   SIGMA=0.70710678
END
PARAM   SIGMA=0.58925565
END
PARAM   SIGMA=0.41594517
END
STOP
FORTRAN
        REAL FUNCTION GAUS(TMU,SIGMA,TIME)
        GAUS=EXP((-0.5)*((TIME-TMU)/SIGMA)**2)
        IF(ABS(GAUS).LE.0.01) GO TO 5
        RETURN
5      GAUS=0.0
        RETURN
        END

```

LIST OF REFERENCES

1. Sketoe, J. G., Computer Aided Design and Analysis, M.S. Thesis, Naval Postgraduate School, Monterey, 1968.
2. Sierra, H. M., "Increased Magnetic Recording Readback Resolution by Means of a Linear Passive Network," IBM Journal of Research and Development, vol. 7, pp. 22-23, January, 1963.
3. Syn, W. M., Turner, N. N. and Wyman, D. G., DSL/360 Digital Simulation Language User Manual, November, 1968.
4. Van Valkenburg, M. E., Introduction to Modern Network Synthesis, Wiley, 1960.
5. Guillemin, E. A., Synthesis of Passive Networks, Wiley, 1957.
6. Baycura, O. M., A Computer Solution for the Sierra Filter, unpublished.
7. Guillemin, E. A., Synthesis of Passive Networks, pp. 337-343, Wiley, 1957.
8. Kuo, F. F., Network Analysis and Synthesis, 2d ed., pp. 383-388, Wiley, 1962.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Documentation Center Cameron Station Alexandria, Virginia 22314	20
2. Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3. Naval Security Group Hdq. 3801 Nebraska Ave., N.W. Washington, D. C. 20360	1
4. Assoc. Professor O. M. Baycura Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	10
5. LCDR Alec Patton Taylor, Jr., USN c/o Naval Security Group Hdq. 3801 Nebraska Ave., N.W. Washington, D. C. 20360	1

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)		2a. REPORT SECURITY CLASSIFICATION	
Naval Postgraduate School Monterey, California 93940		Unclassified	
3. REPORT TITLE		2b. GROUP	
Computer-Aided Filter Design			
4. DESCRIPTIVE NOTES (Type of report and, inclusive dates)			
Master's Thesis; December 1969			
5. AUTHOR(S) (First name, middle initial, last name)			
Alec Patton Taylor, Jr.			
6. REPORT DATE		7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
December 1969		44	8
8a. CONTRACT OR GRANT NO.		9a. ORIGINATOR'S REPORT NUMBER(S)	
b. PROJECT NO			
c.		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
d.			
10. DISTRIBUTION STATEMENT			
This document has been approved for public release and sale; its distribution is unlimited.			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY	
		Naval Postgraduate School Monterey, California 93940	
13. ABSTRACT			
<p>Several small-scale, special-purpose computer programs that were written to provide design data for a pulse-slimming filter over a wide range of parameter variations are contained herein. A large-scale computer program is also used to provide additional data. Some synthesis techniques are discussed, and some of the computer program outputs are directly applied in illustrating synthesis techniques for a specific pulse-slimming transfer function.</p>			

14

KEY WORDS

LINK A

LINK B

LINK C

ROLE

WT

ROLE

WT

ROLE

WT

Computer-Aided Design

Sierra Filter

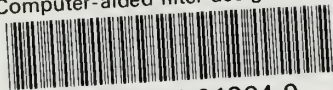
Pulse Compression Filter

Filter Design

Lattice Synthesis

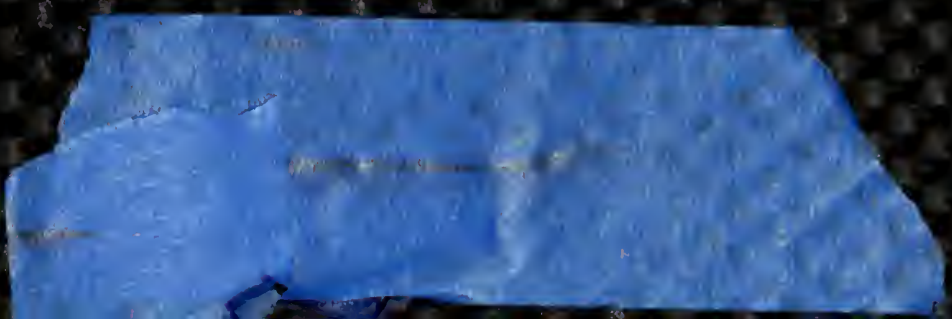
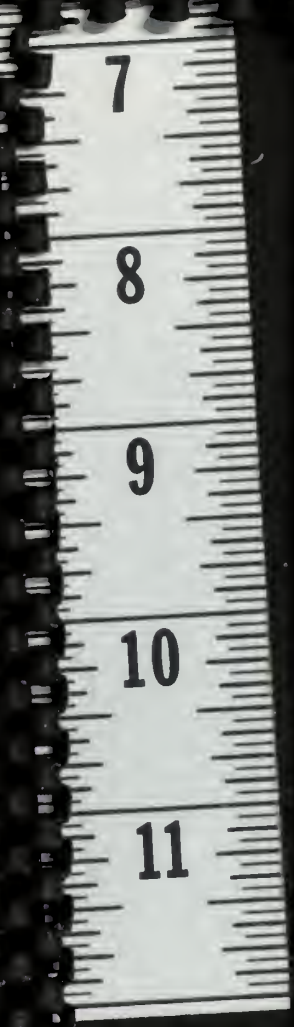
thesT193

Computer-aided filter design.

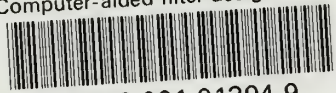


3 2768 001 01394 9

DUDLEY KNOX LIBRARY



thes 1 193
Computer-aided filter design.



3 2768 001 01394 9
DUDLEY KNOX LIBRARY