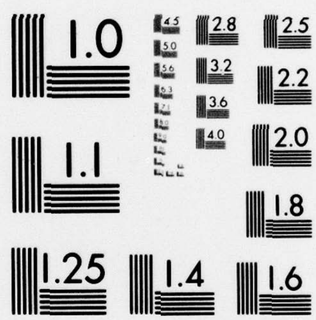
AD-A073 962 NAVAL POSTGRADUATE SCHOOL MONTEREY CA F/G 9/2
AN IN-DEPTH STUDY OF THE SENSITIVITY OF WAVE DIGITAL FILTERS,(U)
JUN 79 M SANAIE-FARD

UNCLASSIFIED NL

1 OF 3
AD
A073962


1.0
4.5
5.0
5.6
6.3
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

AN IN-DEPTH STUDY OF THE SENSITIVITY
OF WAVE DIGITAL FILTERS

by

Masoud Sanaie-Fard

June 1979

Thesis Advisor: Sydney R. Parker

Approved for public release; distribution unlimited.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) An In-depth Study of the Sensitivity of Wave Digital Filters.	(11)	5. TYPE OF REPORT & PERIOD COVERED Electrical Engineer; Jun[redacted]979
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) Masoud Sanaie-Fard		8. CONTRACT OR GRANT NUMBER(s) (12) 240p.
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE June 1979
		13. NUMBER OF PAGES 239
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A detailed analytical and experimental study of the sensitivity of Wave Digital Filters has been conducted. The results indicate that the wave digital filter tends to achieve the same low sensitivity characteristic as the analogue circuit from which it was derived. Other results indicate relatively higher sensitivity to terminating resistance values compared with internal element values, lower sensitivity for algorithms derived from simple rather than multiple elements, and higher sensitivity at the critical

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
(Page 1) S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

251 450 JOB

frequencies. Finally the rms error due to the quantization in the number of bits in the multiplier coefficients has been measured at approximately 3 db per bit for the many examples tested.

Approved for public release; distribution unlimited.

An In-depth Study of the Sensitivity of Wave Digital Filters

by

Masoud Sanaie-Fard
Commander, Iranian Navy
BSEE, Electrical Engineering, Royal Naval
Engineering College Manadon,
Plymouth, England, 1967
MSEE, Electrical Engineering, United States
Naval Postgraduate School, 1978

Submitted in partial fulfillment of the
requirements for the degree of

ELECTRICAL ENGINEER

from the

NAVAL POSTGRADUATE SCHOOL
June 1979

Author: M. Sanaie-Fard

Approved by: Sydney R. Parker, Thesis Advisor

Richard W. Hamming, Second Reader

D. E. Kirk, Chairman, Department of Electrical Engineering

William M. Tolles, Dean of Science and Engineering

ABSTRACT

A detailed analytical and experimental study of the sensitivity of Wave Digital Filters has been conducted. The results indicate that the wave digital filter tends to achieve the same low sensitivity characteristic as the analogue circuit from which it was derived. Other results indicate relatively higher sensitivity to terminating resistance values compared with internal element values, lower sensitivity for algorithms derived from simple rather than multiple elements, and higher sensitivity at the critical frequencies. Finally the rms error due to the quantization in the number of bits in the multiplier coefficients has been measured at approximately 3 db per bit for the many examples tested.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Dedicated to the

Memory of My Father

ACKNOWLEDGMENT

I would like to express gratefully and cordially my sincere gratitude to Professor Sydney R. Parker, for his constant guidance, encouragement, and generous devotion of the time, which made completion of this thesis possible. Working with him has been a great privilege and experience.

I am also very obliged to Professors Walter Ku, Cornell University, B. Peikari, Southern Methodist University, and R. E. Hamming for their generous help, and also the Iranian Navy for giving me this great opportunity.

Finally it would be inappropriate if I did not mention Ms. M. Marks who with great patience and sincerity did the typing and helped me in formatting this thesis.

I. INTRODUCTION AND SCOPE

A. INTRODUCTION

Signal processing is necessary in diverse areas of science and engineering, such as communication, social sciences, biomedical, control, radar, acoustics, telemetry, and intelligence and information gathering, etc. In general this process can be done with analogue (continuous time), or digital (discrete time, discrete amplitude) signals.

With the advent of LSI and VLSI, microprocessors, the need for efficient digital signal processing algorithms becomes more and more important. Much of the current literature is devoted to design of the linear algorithms, under the title of digital filters. Important factors in designing the digital algorithms are, time of the calculation, implementation, accuracy (error), etc.

Although there is evidence that direct digital filter design is possible [1], nearly all of the digital filter design algorithms use the analogue to digital transformation techniques. It is interesting to note that the rapid development of the digital signal processing is partly because of the existence of the well established theory on the analogue techniques and partly because of the abundance of the general purpose digital computers.

The digital computer and specially microprocessors, being physical objects from space and economical point of view, can only accommodate for finite precision in the size of the digital filter multiplier coefficients. Thus the need for digital filter algorithms with low sensitivity to digital filter multiplier coefficients arises.

Fettweis [2] in order to design a low sensitivity digital algorithm has proposed an alternative digital filter design method, namely wave digital filter, in which analogue LC circuit is transformed into discrete algorithm using wave or scattering matrix parameters. This approach in design is different from the conventional design techniques, because we have a new set of variables which are referred to as "incident and reflected wave parameters".

Wave digital filter design is believed to be difficult to understand, and is therefore generally avoided. But in reality this is not the case. In fact, to design a wave digital filter one need not go into the details of the algorithm development. One merely needs to know some basic facts and then can use the final wave digital filter equations and tables in order to design the required filter.

It is a well known fact that analogue LC networks have very low sensitivity to variation in LC component values. Upon this fact Fettweis and others have argued that, since the wave digital multiplier coefficients are derived from the LC parameters of the parent circuits, they should also have the same favorable low sensitivity characteristics to multiplier coefficients. Furthermore it is also known that the digital algorithms with low sensitivity to multiplier variations also exhibit minimum round-off noise due to quantization after multiplication of these multipliers with signals. As a result it is conjectured that the wave digital filter will have minimum round-off noise properties when compared with other digital filter algorithms. The purpose of this thesis is to analyze and check this conjecture.

B. SCOPE AND ORGANIZATION

In order to develop and study the wave digital filter, the research in this thesis is divided into seven chapters. In Chapter II a brief discussion of the general digital filter theory is given. The presentation contains only selected and necessary background required for wave digital filter theory development in this thesis later on. Also included in this chapter are A B C D matrix theory, and the concept of the delay free feedback which plays an essential role in the wave digital filter theory development. Necessary sensitivity theory background used in the sensitivity analysis of digital filter is discussed briefly in Chapter III. The development of wave digital filter theory is done in Chapter IV. This development is straightforward and general in a sense that only one algorithm is developed for both series and shunt element. The effect of sampling interval is also introduced for the first time into the wave digital filter algorithms. Four useful tables of wave digital iterative algorithms for simple L and C elements in both series and shunt configuration are also given. The delay free path which plays an essential role in wave digital filter theory is emphasized throughout this chapter. The studies of the sensitivity of the wave digital filter to quantization in the number of bits of the multiplier coefficients is done in Chapter V. To do these studies in a fairly general sense, three different wave digital filter algorithms developed in Chapter IV are used with two different conventional digital filters for the given filter. A total of nine different filter types with different terminating source resistances were examined and compared with each other to arrive at a general conclusion. Chapter VI presents a specialized in-depth study of the sensitivity of the simple wave digital filter algorithm. In this chapter sensitivity distribution along the filter structure was examined in order to understand the sensitivity behavior of the subsections of the wave digital filter on the overall sensitivity of the wave digital filter. Chapter VII summarizes the new results and proposes future research ideas related to the wave digital filter theory. There are three appendices. Appendix 1 includes an example to show the nearly exact equivalence between the wave digital filter and the conventional digital filter, both in the time domain and frequency domain when infinite precision arithmetic is used. The rest of the appendices include ten computer programs. The computer programs are used to derive the results of the main text. It is important to note that in order to facilitate the better understanding of the computer programs, explanatory remarks are made in the comment cards. These computer programs are in FORTRAN IV and can be used on any standard general purpose digital computer capable of dealing with FORTRAN IV scientific computer language.

Finally it is important to note that in this thesis for easy access, the references are given at the end of each chapter, rather than collectively at the end of the thesis.

References

1. A. G. Constantinides, Digital Notch Filters, Electronics Lett. Vol. 5, No. 9, pp. 198-199, May 1969.

2. Alfred Fettweis, Digital filter structures related to classical filter networks, AEÜ, Vol. 25, No. 2, pp. 79-89, February 1971, [text is in English].

II. GENERAL BACKGROUND

A. INTRODUCTION

The main intent of this chapter is to briefly review general digital filter theory in order to establish the background necessary for the main subject of this thesis, i.e. wave digital filters. The design of the conventional digital filter is a well established subject. Thus it will be dealt very briefly. Also discussed in this chapter is the A B C D parameter matrix theory, which will be used later on followed by the concept of the delay-free feedback path or delay-free loop in the digital two port network which are used in deriving the causal wave digital filter algorithms in Chapter IV. z domain transform theory is assumed as a background and is not discussed.

B. GENERAL REVIEW OF DIGITAL FILTER DESIGN

The design of electronic filters in the analogue domain is a well established subject, which not only includes very sophisticated techniques, but also has some very well established supporting computer programs as well. Much of the development of the digital filter has been directed towards the transfer of these results from the analogue domain to the digital domain.

In the most general sense a digital filter is a linear, shift invariant discrete time system which is realized using finite precision arithmetic. The design of the digital filters involves three basic steps:

(1) the specification of the desired properties of the system

(2) the approximation of these specifications using a causal discrete time system

(3) the realization of the system using finite precision arithmetic

Note that these three steps are not independent of each other. In this thesis we are mainly interested in step 2 and to some extent on step 3.

C. METHODS OF DESIGNING THE DIGITAL FILTER

An important class of techniques for designing infinite impulse response filters to be realized recursively is based on a transformation of a continuous time filter. This class comprises at least three techniques.

1. Impulse Invariance Method

The impulse invariance method, also called standard z transformation (or standard z), is a technique in which the impulse response of the derived digital filter is identical to the sampled impulse response of the continuous time filter. If the impulse response of the filter is h(t) then the sampled impulse response will be

$$h^{*}(t) = h(t)\cdot \delta_T(t) \tag{2.1}$$

where $\delta_T(t)$ is the sampling function and is defined by

$$\delta_T(t) = \begin{cases} 1 & t=nT \\ 0 & t\neq nT \end{cases} \quad \text{where } n=0,1,2,\ldots$$

It can be shown that the Laplace transform of $h^*(t)$ will be

$$H^*(s) = \frac{1}{T} \sum_{K=-\infty}^{K=\infty} h(s + \frac{jK2\pi}{T}) \qquad (2.2)$$

and the impulse invariance response of the filter will be

$$H(z) = H^*(s) \Big|_{z = e^{sT}} \qquad (2.3)$$

from the relationship $z = e^{sT}$ it is seen that the strips of width $\frac{2\pi}{T}$ in the s plane map into the entire z plane as depicted in Figure 2.1, the left half of s plane strip maps into the interior of the unit circle, and the imaginary axis of the s plane maps onto the unit circle in such a way that each segment of length $\frac{2\pi}{T}$ is mapped once around unit circle. Thus, the mapping is not a one to one mapping, and hence it can easily be shown that the impulse invariance response is only satisfactory if H(s) is band limited. And as in most cases if H(s) is not sufficiently band limited H(z) is an aliased version of H(s). Therefore the technique is used for narrow band filter design or else H(s) is broken into cascaded subsections of first order and second order sections with guard filters in between, which in some cases is a tedious job. Also it is clear from equation (2.2) that due to the $\frac{1}{T}$ multiplier, the digital filters derived using the impulse invariance method have a gain approximately $\frac{1}{T}$ to that of continuous time filter, which should be taken into account in the design.

2. Matched z Transform

This procedure is based on mapping the poles and zeros of the continuous time filter by the substitution of $(s-s_1) \rightarrow (1-e^{s_1 T} \cdot z^{-1})$.

This means that poles of H(z) will be identical to those obtained by impulse invariant method; however the zeros will not correspond.

3. Designs Based on Numerical Solution of Analogue Differential Equations

a. Conventional Digital Filter Design

In this method, the differential equation of the analogue filter is approximated by a recursive equation, which is the standard procedure in Numerical Analysis. There are three basic numerical integration techniques, namely,

(i) Euler forward integration

(ii) Euler backward integration

(iii) bilinear integration

All these techniques plus many others are described fully in the literature. And there is no need to go into details for all of these techniques, but because of the importance of bilinear transformation we describe it briefly.

The approach uses the algebraic transformation

$$s = \frac{2}{T}\left[\frac{1-z^{-1}}{1+z^{-1}}\right] \tag{2.4}$$

to derive the system transfer function of the digital filter such that

$$H(z) = H(s)\Big|_{s = \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)} \tag{2.5}$$

This transformation has the effect of mapping the entire left half s plane into the inside of the unit circle and entire right half of the s plane into the outside of the unit circle as shown in Figure 2.2. This results in a nonlinear warping of the frequency scale according to the relation

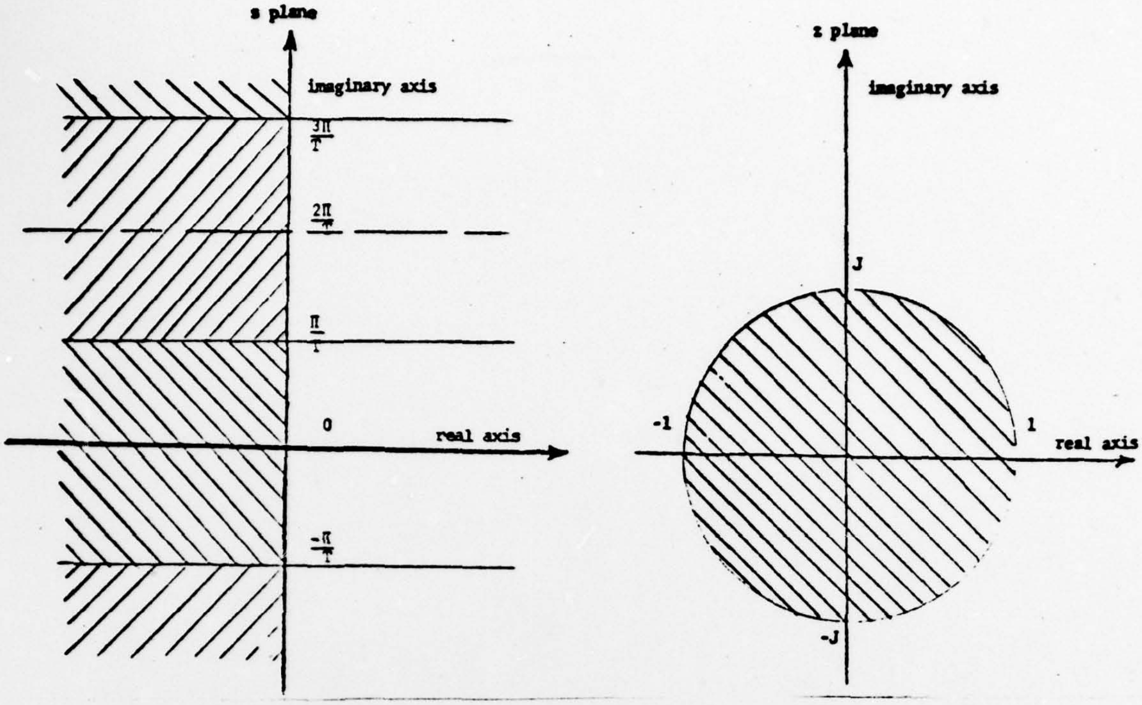


Fig. 2.1. Mapping of f(s) into f(z) as per relationship $z = e^{sT}$. Note that the area between the strips π/T and $-\pi/T$ map into the entire z plane in such a way that the area on the left half s plane strip maps into the inside of the unit circle and the area on the right half s plane strip maps outside the unit circle. The process is repeated infinitely many times thus mapping or transformation is not one to one.

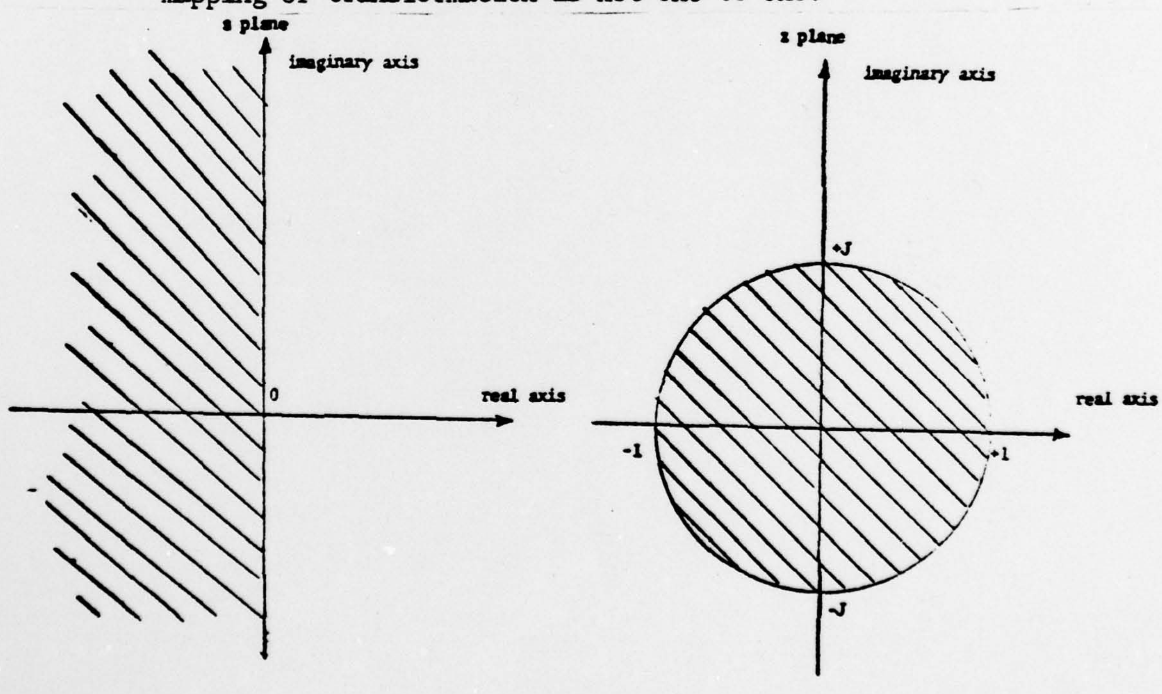


Fig. 2.2. Bilinear mapping of f(s) into f(z) as per relationship $s = 2(z-1)/T(z+1)$. Note that the transformation is a one to one mapping.

$$\frac{\omega_c T}{2} = \tan \frac{\Omega_{cd} T}{2} \qquad (2.6)$$

where ω_c is the critical frequency of analogue filter and Ω_{cd} is the critical frequency of the digital filter. Because of this warping of the frequency scale this design technique is most useful in obtaining digital design of filters whose frequency response can be divided up into a finite number of pass bands and stop bands in which the response is essentially constant. Figure 2.3 shows the frequency response of an analogue filter and its approximated digital frequency response using bilinear transform techniques.

From Figure 2.3 it is obvious that if we require the critical frequency of the digital filter to be say at Ω_{cd} then we have to frequency scale the critical frequency of the analogue filter by a factor

$$\text{factor} = \frac{T}{2 \tan \frac{\Omega_{cd}}{2}} \omega_c \; . \qquad (2.7)$$

Typical frequency selective analogue filters are Butterworth, Chebyshev, and Elliptic filters. Note that all these filters have closed form analogue design formulas and by the use of bilinear transformation we can easily get approximate closed form digital filter algorithms.

A Butterworth filter is a monotonic in the pass band and in the stop band.

A Chebyshev filter has an equiripple characteristic in the pass band and monotonic in the stop band, or vice versa.

An Elliptic filter is equiripple in both pass band and stop band. Clearly these properties will be preserved when the filter is mapped to a digital filter with the approximated bilinear transformation as shown in Figure 2.3.

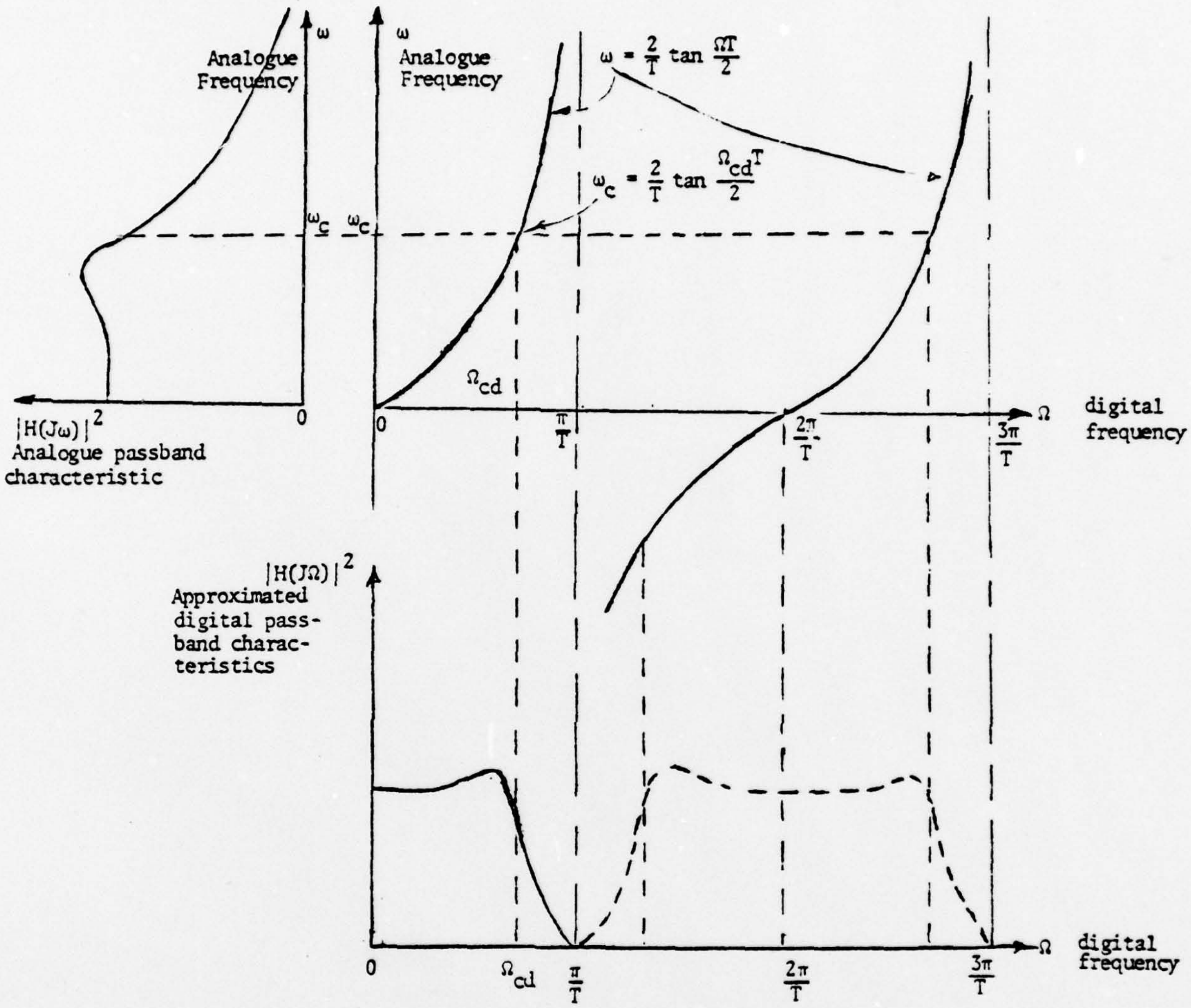


Figure 2.3. Transformation of analogue filter H(s) into digital filter using bilinear transform S = 2(z-1)/T(z+1).

b. Wave Digital Filter Design

This technique basically uses the bilinear transformation of the analogue to digital design exactly the same as part a iii, but the attempt is made only on LC filters having input and output terminating resistances of R_1 and R_2.

In this technique each reactive element in the analogue ladder structure is transformed into a two port signal flow structure using the bilinear transformation and wave flow techniques, and at the same time matching the port one impedance of the derived two port structure to the port two impedance of the previously derived element, thus eliminating mismatch between the succeeding elements. The details of this and its implementation are left for Chapter IV. Note that the idea is a very basic one and can be applied on many varieties of common filters.

D. CHAIN OR A B C D MATRIX THEORY

The analysis of any passive linear network with several inputs and outputs can be done in many ways. The most usual ones are signal flow graph, system matrix equations, input/output algorithms, transform matrix equations, etc. However, for systems of order higher than 2, most of the above analysis becomes tedious and prone to mistakes due to system complexity. A most useful and convenient method of dealing with a complex system is, whenever possible, to break the system into several subsystems and interrelate these subsystems by a chain matrix, thus allowing each subsystem to be analyzed and investigated separately one at a time, without even thinking about the rest of the system. To illustrate the point consider the network N of Figure 2.4 with wave inputs a_1 and a_2 and wave outputs b_1 and b_2. The relationship between port 2 and port 1 for this network can be written as

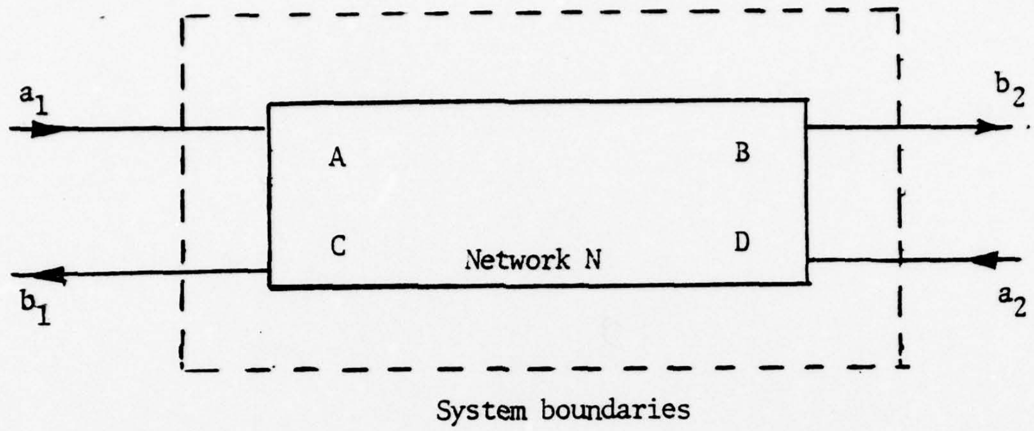


Fig. 2.4. A general two port structure with inputs a_1 and a_2 and outputs b_1 and b_2, into and out of the system boundaries respectively.

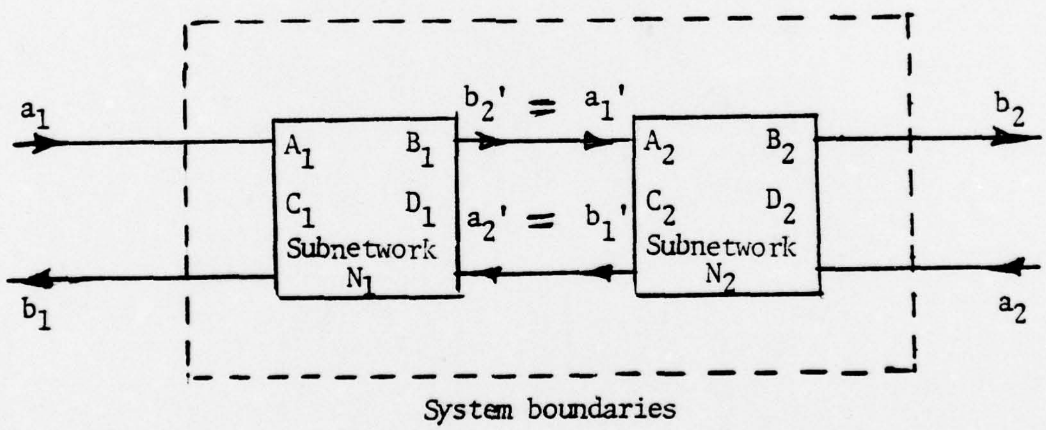


Fig. 2.5. A general two port structure with inputs a_1 and a_2 and outputs b_1 and b_2 into and out of the system boundaries respectively. Note that the system of Fig. 2.4 is equivalent to the system of Fig. 2.5 if the inputs (a_1 and a_2) and outputs (b_1 and b_2) are exactly the same.

$$\begin{bmatrix} b_2 \\ a_2 \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \tag{2.8}$$

Now if we can break network N into several subnetworks inside the dotted line without touching the input and output ports, the resulting network will be exactly the same as network N. Note that the networks N_1 and N_2 resulting from network N do not necessarily have to have equal subsections as shown in Fig. 2.5. Note also how the consistency in the wave-flow direction is maintained.

Thus from Fig. 2.4

$$\begin{bmatrix} b_2' \\ a_2' \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \tag{2.9}$$

$$\begin{bmatrix} b_2' \\ a_2' \end{bmatrix} = \begin{bmatrix} a_1' \\ b_1' \end{bmatrix} \tag{2.10}$$

and

$$\begin{bmatrix} b_2 \\ a_2 \end{bmatrix} = \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} a_1' \\ b_1' \end{bmatrix} \tag{2.11}$$

$$\begin{bmatrix} b_2 \\ a_2 \end{bmatrix} = \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \tag{2.12}$$

or

$$\begin{bmatrix} b_2 \\ a_2 \end{bmatrix} = \begin{bmatrix} A_1A_2+B_2C_1 & A_2B_1+B_2D_1 \\ A_1C_2+C_1D_2 & C_2B_1+D_1D_2 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \tag{2.13}$$

Note that equations (2.8), (2.13) are identical. This simple example clearly demonstrates how a system when made up of only two simple subsystems N_1 and N_2 which are easy to be analyzed each separately, when combined into the system N becomes a complex system, and very difficult to get analyzed.

Note that in the first case it is a very easy job to analyze subelements A_1, B_1, C_1 and D_1 of system N_1 or that of the system N_2. While clearly the analysis of the element $A = A_1A_2 + B_2C_1$ etc. of the system N will not be an easy job, and in most cases when the system is made up of more than 2 subsystems it is a tedious if not an impossible task. Thus equation (2.12) clearly demonstrates the fundamental property of the chain matrix analysis.

Whenever two or more than two pairs are connected in cascade the chain matrix of the composite network will be the product of the individual chain matrices. Since in general matrix multiplication is not commutative, it is important that the matrices be multiplied in the same order as the circuits are cascaded. It can easily be shown that if all the individual matrices are reciprocal the composite two port will also be reciprocal.

E. CONCEPT OF DELAY FREE FEEDBACK PATH, OR DELAY FREE LOOP IN A TWO PORT NETWORK

As mentioned in section II-D the relationship between port two and port one of a subsection of a general causal system such as that of Fig. 2.5 can best be described by the equation (2.8), i.e.:

$$\begin{bmatrix} b_2' \\ a_2' \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \qquad (2.14)$$

where b_1, a_1 are the output and input quantities at port one and b_2', a_2' are the output and input quantities at port two. This equation can be rewritten in terms of input quantities in the following form

$$\begin{bmatrix} b_1 \\ b_2' \end{bmatrix} = \begin{bmatrix} -\frac{C_1}{D_1} & \frac{1}{D_1} \\ \frac{A_1D_1-B_1C_1}{D_1} & \frac{B_1}{D_1} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2' \end{bmatrix} \tag{2.15}$$

or we can write equation (2.15) in the simpler form of

$$\begin{bmatrix} b_1 \\ b_2' \end{bmatrix} = \begin{bmatrix} f_1(z) & f_2(z) \\ f_3(z) & f_4(z) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2' \end{bmatrix} \tag{2.16}$$

Note that for a causal digital system the values $f_1(z)$, $f_2(z)$, $f_3(z)$ and $f_4(z)$ are of the form

$$f(z) = \frac{a_o + a_1z^{-1} + \text{---} + a_iz^{-i} + \text{---}\ a_nz^{-n}}{1 + b_1z^{-1} + \text{---} + b_iz^{-i} + \text{---}\ b_nz^{-n}}\ . \tag{2.17}$$

Note also that when written in terms of positive exponents of z, the order of z in the numerator must be equal to or less than the order of the denominator for causality. With this in mind the iterative equations derived from equation (2.15) are

$b_1(n) = \alpha_1a_1(n) + \beta_1a_2'(n) +$ weighted values of past inputs at port one and port two plus weighted values of past outputs at port one (2.18)

and

$b_2'(n) = \alpha_2a_1(n) + \beta_2a_2'(n) +$ weighted values of past inputs at port one and port two plus weighted values of past outputs at port two (2.19)

Note that α_1, α_2, β_1, β_2 are merely weighting constants, and equations (2.18) and (2.19) are both causal and realizable.

Now if the input $a_2'(n)$ at the port two is a function of $b_2'(n)$ of the port two, i.e.

$$a_2'(n) = Kb_2'(n) + \text{weighted values of past input and output values at port two} \quad (2.20)$$

which is the case for most cascaded two port systems such as the one shown in Fig. 2.5 the equations (2.18) and (2.19) further reduce to

$$b_1(n) = \alpha_1 a_1(n) + \beta_1 K b_2'(n) + \text{weighted values of past inputs and outputs at port one and port two} \quad (2.21)$$

and

$$b_2'(n) = \alpha_2 a_1(n) + \beta_2 K b_2'(n) + \text{weighted values of past inputs at port one and port two plus the weighted values of past outputs at port two} \quad (2.22)$$

Note that the iterative equation (2.22) is not causal since for the calculation of $b_2'(n)$ it requires $b_2'(n)$ which is not possible. Thus for the causality either K must be equal to zero or β_2 must be equal to zero which are two distinct cases.

Case 1

By considering Fig. 2.5 it can be seen that a_2', b_2' are merely the port one quantities of the second stage. Thus if we are going to consider the first stage only we cannot force K to be equal to zero. Thus the only variable left is β_2 and by making β_2 equal to zero we can make equation (2.22) a valid equation. This condition is referred to the case of no internal delay free path from a_2' to b_2' or no delay free path in port two. Thus to implement this condition the order of the numerator of the function $f_4(z) = \frac{B_1(z)}{D_1(z)}$ must be at least one order lower than the denominator.

Case 2

As it was noted in the Case 1 since a_2', b_2' are merely port one quantities of the second stage. If we can make $b_1'(n)$ of the second stage to be independent of the $a_1'(n)$ of the second stage and only dependent on the past inputs and outputs of the port one of the second stage, then we have actually managed to make $a_2'(n)$ of the first stage to be independent of $b_2'(n)$ of the first stage. This can be done by forcing $\alpha_1 = 0$ in the second stage. This condition is referred to the case of no internal delay free path from a_1 to b_1 or no delay free path in port one. Note that to implement this condition the order of the numerator of the function $f_3(z) = \frac{A_1(z)D_1(z) - B_1(z)C_1(z)}{D_1(z)}$ must be at least one order lower than that of the denominator.

F. SUMMARY

In this chapter we have reviewed briefly the ground work required for the matched two port wave digital filter theory and design.

The contents of this chapter are used throughout this thesis. No particular mention of any reference has been made since most of the subjects discussed are well established and details can be found in most digital filter design handbooks or papers. It is worthwhile to note that the concept of delay free feedback path, though important, in most papers reviewed were merely stated without any proof. Thus in this chapter an attempt was made to prove it in the most general sense.

III. GENERAL DISCUSSION ON SENSITIVITY THEORY RELATED TO WAVE DIGITAL FILTERS

A. INTRODUCTION

The main intent of this chapter is to start with the low sensitivity properties of the doubly terminated analogue LC structure and then extend this property to the wave digital filter. Later on in the chapter we explain briefly the development of wave digital filter theory. It is not the intention to go into details, but merely to give an overview of previous works for which full development is available in the references.

Another objective of this chapter is to briefly discuss the different wave digital filter structures and algorithms which are all called wave digital filters, each structure having its own characteristics and limitations. The newcomer to this field will be astonished by the many different structures and algorithms which are present in the literature; all of them are offered under the same name of wave digital filter.

Later in this chapter a natural development of the wave digital filter theory is traced from the initial conjecture of Fettweis up to the present day state of the art.

B. DEFINITION OF SENSITIVITY

The term sensitivity of a certain filter structure has a broad meaning, and the literature is full of different definitions for sensitivity functions meeting a specified requirement. But in general all of the definitions end up more or less to the following:

As the filter element values or coefficients are varied about their true value, the filter pole and zero locations are shifted correspondingly, causing a change in the magnitude and phase characteristic of the filter. A mean shift in the characteristic can be computed on a point by point basis and used as a figure of merit, but it may be meaningless when determining whether or not the original filter specifications are met. Therefore sensitivity must be interpreted in terms of several factors to include such parameters as bandwidth, cut-off frequency, ripple in the pass and stop band, etc. Several sensitivity functions are most commonly used. They are

(i) Q sensitivity and pole frequency sensitivity

(ii) Root sensitivity

(iii) Coefficient sensitivity

(iv) Frequency response sensitivity, etc.

Note that most of these sensitivity functions are meaningful and for most cases there is a relationship between most of these sensitivity functions with each other [12].

Since in this thesis we are mainly interested in the parameter or coefficient sensitivity, thus no discussion of the other sensitivity functions will be made.

C. DEFINITION OF PARAMETER SENSITIVITY

For the function $H(C_1, C_2,\ldots, C_i,\ldots, C_n)$, where H is a function of multiparameter $(C_1, C_2,\ldots, C_i,\ldots, C_n)$, the parameter sensitivity H of the system to any change of variables or elements C_i is defined in several ways

i) $S = \frac{\partial H}{\partial C_i} \cdot \frac{C_i}{H}$, logarithmic or factorial sensitivity (3.1)

ii) $S = \frac{\partial H}{\partial C_i}$, derivative sensitivity (3.2)

iii) $S = \frac{\partial H}{\partial C_i} \cdot \frac{1}{H}$ or $S = \frac{\partial H}{\partial C_i} C_i$, semi-logarithmic sensitivity

and all of these sensitivity functions are discussed fully in the literature [12], [13], [14], [15].

D. QUANTIZATION ERROR IN DIGITAL FILTERS AND ITS SENSITIVITY EFFECT

Although tolerance problems which exist in the physical world in the case of analogue filters, do not exist as such for digital filters, still there is an interest in obtaining structures with low sensitivity to parameter variations. There are several reasons for this. The primary reason is the fact that the structures with low sensitivity are less affected by coefficient truncation. Thus element values or coefficients with shorter word length are sufficient for meeting a given specification. The second reason stated and proved by Fettweis [10] is that there exists a relationship between sensitivity with respect to multiplier variation and the round off noise at the output of a digital filter. Thus any improvement in the sensitivity would result in a reduction in the corresponding round-off noise at the output.

In order to compare the coefficient sensitivity of different digital structures Ku and Ng [8] have used a root mean square error criteria in the frequency response given by

$$E_{rms} = [\frac{1}{N+1}] \sum_{i=0}^{N} [W(\omega_i)[\Delta H(\omega_i)]^2]^{\frac{1}{2}} \qquad (3.3)$$

Note that this criteria is in the frequency domain and is based on the deviation of the magnitude of the finite precision output from the ideal or infinite precision output.

Thus $\Delta H(\omega_i)$ is defined as

$$\Delta H(\omega_i) = |H(\omega_i) - H_o(\omega_i)| \tag{3.4}$$

where $H_o(\omega_i)$ is the magnitude of the ideal output at frequency ω_i, and $H(\omega_i)$ is the magnitude of the finite precision output at frequency ω_i, and $W(\omega_i)$ is the weight chosen to reflect the relative importance of error at various frequencies, and N is the number of sample points in the frequency domain.

Furthermore in order to equalize the effect of the various coefficients, floating point arithmetic is used rather than fixed point arithmetic, during the process of rounding off of the coefficient to the required number of the bits. Note that more details and implementation of the idea are left for Chapter V where we investigate case studies based on bit quantization error.

E. SENSITIVITY OF LC LADDER STRUCTURES

It is a well known fact that the doubly terminated analogue LC ladder structures are relatively insensitive to element value changes. To prove this fact most researchers have used the theory of the total conservation of input and output power quantities in a reactive lossless network, the discussion of which is interesting but it not in the scope of this thesis. A very good review is made in this respect by Renner and Gupta [7].

Thus, it is a reasonable assumption that the digital filter derived directly from the topology of a resistively terminated analogue lossless structure would have the same favorable sensitivity properties as its analogue counterpart.

To show the low sensitivity of wave digital filters by the conventional n port scattering matrix network theory, Fettweis [10] defines the instantaneous "pseudopower" transmitted through a wave port k with port input resistance R_k by the means of the equation

$$\mathcal{P}_k = \left(\frac{a_k^2 - b_k^2}{R_k}\right) \qquad (3\text{-}5)$$

where $\mathcal{P}_k$ is the total instantaneous pseudopower input through port k and a_k is the total input wave at port k and b_k is the total reflected output wave at port k. Similarly he defines steady state pseudopower by the relationship

$$P_k = \left(\frac{|A_k|^2 - |B_k|^2}{R_k}\right) \qquad (3\text{-}6)$$

where P_k and A_k and B_k are the steady state values of p_k, a_k, b_k, respectively at an arbitrary frequency "f". In this way he then shows that for all wave digital building blocks which are derived from the LC doubly terminated ladder structures, the total sum of instantaneous pseudopower absorbed by all ports is equal to zero, i.e.

$$\sum_{k=1}^{n} p_k = 0$$

where n is the total number of the ports of the wave digital subelement or building block and p_k is the instantaneous pseudopower at port k.

Similarly he also proves that the total steady state pseudopower absorbed by all building blocks of a terminated wave digital filter at an arbitrary complex frequency "f" is equal to zero, i.e.

$$\sum_{k=1}^{n} P_k = 0$$

where P_k is the steady state pseudopower at an arbitrary complex frequency f, and n is the total number of the ports of the wave digital building block. The low sensitivity property then follows from the zero pseudopower relationships. Although power has physical meaning in analogue circuits its meaning in digital filter algorithms is nebulous.

F. PREVIOUS WORKS ON THE DEVELOPMENT OF THE WAVE DIGITAL FILTERS

In the previous section we discussed the favorable low sensitivity to parameter variation properties of LC ladder structures. It was this fact that led Fettweis to propose the idea of wave digital filters in 1970 [3], derived from the doubly resistive terminated LC ladder structures. Since then numerous papers relating to the wave digital filters have been presented.

To understand the wave digital filters and the state of art, it is necessary to summarize the state of evolution of the wave digital filters.

Partial credit of the development of the wave digital filters can also be given to Richards [1], Kuroda, Ozaki and Ishii [2] and numerous other researchers who contributed to the development and synthesis of the strip line filters and also resistor-transmission line circuits.

The basic idea behind wave digital filter development from the beginning was to keep the desired low sensitivity of lumped LC resistively terminated filters in the already developed analogue domain and transfer it to digital domain with very minor modifications.

Ideal strip line filter and transmission line circuits having inherent LC structure is an ideal starting point. To avoid the loading effect of one element of the LC strip line upon the succeeding and preceding elements, Ozaki and Ishii [2] introduced the idea of inserting unity element, or the lossless transmission line acting as an ideal transformer (which was already developed by Kuroda and known as Kuroda's identity), between the stages as a lossless matching element.

In a more detailed paper Fettweis [4], in order to arrive at realizable signal flow diagram, used wave quantities instead of the usual current and voltage input quantities to the conventional filter. Actually in using the wave quantities as input and output signal, Fettweis made use of the already developed scattering parameter matrix theory of the n port networks. In order to avoid the loading effect of one section upon the other, and thus to eliminate the unwanted mismatch reflections between sections, he employed unit element or impedance transformer, known as Kuroda's identity [2].

Although with the introduction of the unity element a desirable result was achieved in realizing the LC filter structures, this introduction further increased the number of required operations (mainly multiplications) over the conventional digital filter.

In order to avoid the unity element several attempts were made by different researchers, Sedlmeyer and Fettweis [5], Van Haften and Chirlain [6], and others, to eliminate the need for unity elements or impedance transformers between the cascading sections. One of the attempts was made by Fettweis [5] and later on the method was renamed by Ku and Ng [8] as the "new Fettweis method." The new Fettweis method forced one of the scattering multiplier coefficients in the three port network to be equal to 1, thus a two port element was made from the three port element. By starting at one port and progressing forward towards the other port, and by suitable choice of input impedance of the next section (i.e. matching the output impedance of one section to the succeeding section), the need for unity element or impedance transformer matching was eliminated.

Van Haften and Chirlain [6] next came up with the idea of cascading the wave digital three port section directly without the unity element but taking into account the approximate attenuation of one section upon the next section, thus eliminating the need for unity element and extra multiplications associated with it. At the same time this introduced errors associated with the calculation of the approximate attenuation of one section upon the other.

S. Erfani and B. Peikari [9] and N. S. Swamy and K. S. Thyagarajan [11] at the same time came up with a new approach to eliminate the unity element. This technique is the basis of the further investigation in this thesis, and details are given in Chapter IV and other chapters.

References

1. Paul I. Richards, Resistor transmission line circuits, Proceedings of IRE, February 1948, pages 217-220.

2. H. Ozaki and J. Ishii, Synthesis of a class of strip line filters, IRE Transactions on Circuit Theory, Vol.CT-5,June 1958,pages 104-109.

3. Alfred Fettweis, Some principles of designing digital filters imitating classical filter structures, IEE Transactions on Circuit Theory, March 1971, pages 314-316.

4. Alfred Fettweis, Digital filter structures related to classical filter networks, Archiv fur Elektronik und Vbertrangungstechnik, Vol. 25, No. 2, February 1971, [text written in English].

5. Axel Sedlmeyer and Alfred Fettweis, Digital filters with true ladder configuration, IEE, Circuit Theory and Applications, Vol. 1, 5-10, 1973.

6. D. Van Haften, An analysis of errors in wave digital filters, Doctoral dissertation, 1977, Stevens University.

7. K. Renner and S. C. Gupta, On the design of wave digital filters with low sensitivity properties, IEEE Transactions on Circuit Theory, Vol. CT-20, No. 5, September 1973, pages 555-567.

8. W. H. Ku and S. M. Ng, Floating point coefficient sensitivity and round-off noise of recursive digital filters realized in ladder structures, IEE Transactions of Circuits and Systems, Vol. CAS-22, No. 12, December 1975.

9. S. Erfani, Design of fixed and variable digital filters using generalized delay units, Ph.D. dissertation, 1976, Southern Methodist University, Dallas, Texas.

10. Alfred Fettweis, Pseudopassivity, sensitivity, and stability of wave digital filters, IEEE Transactions on Circuit Theory, Vol. CT-19, No. 6, November 1972.

11. K. S. Thyagarajan, One and two dimensional wave digital filters with low coefficient sensitivities, 1977, Doctoral dissertation, Concordia University, Montreal, Canada.

12. G. C. Temes and S. K. Mitra, Modern filter theory and design, Chapter 8, 1973, Published by John Wiley and Sons.

13. S. R. Parker, Sensitivity analysis and models of nonlinear circuits, IEEE Transactions on Circuit Theory, Vol. CT-16, No. 4, November 1969.

14. R. Tomovic and M. Vukobratovic, General sensitivity theory, 1972, American Elsevier Publishing Company, Inc., New York.

15. S. R. Parker, Sensitivity old questions, some new answers, IEE Transactions on Circuit Theory, Vol. CT-18, No. 1, January 1971.

IV. GENERAL THEORY OF THE WAVE DIGITAL FILTER

A. INTRODUCTION

The intent of this chapter is to derive the generalized algorithm for a two port wave digital filter on a step by step basis using the principles of the circuit theory, matrix algebra, and scattering matrix wave theory.

The resulting algorithms are summarized in two tables in section E. Also, two illustrative examples are given, example one being a simple element, suitable for matched source; and example two being a complex element, suitable for a matched load.

It must be emphasized that the techniques used to derive the wave digital algorithms in this chapter are more general than the previous works, in the sense that only one algorithm is derived for both series and shunt element. Also the effect of sampling time is introduced into the algorithms.

B. DERIVATION OF THE GENERALIZED TRANSFER RATIO FOR A TWO PORT WAVE DIGITAL FILTER ALGORITHM

Consider the two port network N_1 of Figure 4.1b whose inputs are a_1 and a_2 and outputs are b_1 and b_2. In order to use the chain matrix theory developed earlier (Chapter II, Section B-D) we need to find port two input and output waves a_2, b_2 in terms of port one input and output waves a_1, b_1 or vice versa.

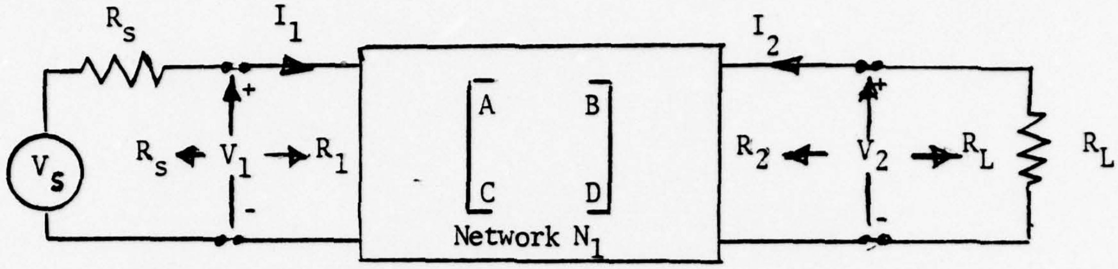


Fig. 4.1a.

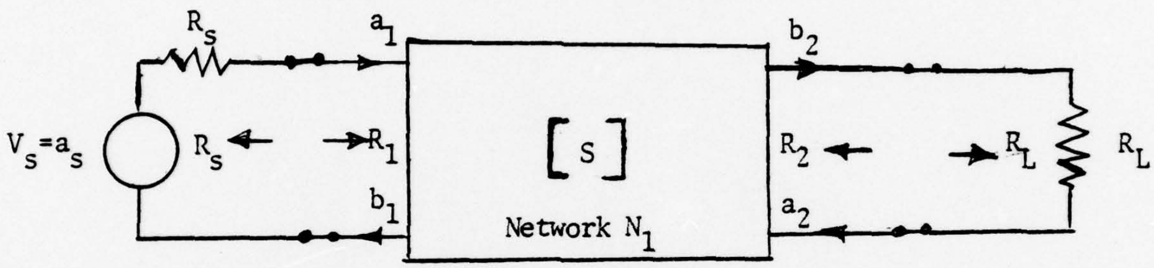


Fig. 4.1b.

Fig. 4.1. Two generalized representations of a two port network. Fig. 4.1a representation in terms of voltage and current quantities. Fig. 4.1b representation in terms of scattering matrix wave quantities. Note that R_1 is the input impedance of the network and R_2 is the output impedance of the network. R_s is the source impedance and R_L is the load impedance.

Thus considering the scattering matrix model of Figure 4.1b, we have

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} 1 & R_1 \\ 1 & -R_1 \end{bmatrix} \begin{bmatrix} V_1 \\ I_1 \end{bmatrix} \tag{4.1}$$

and

$$\begin{bmatrix} a_2 \\ b_2 \end{bmatrix} = \begin{bmatrix} 1 & R_2 \\ 1 & -R_2 \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix} \tag{4.2}$$

where R_1 and R_2 are port impedances of port one and port two respectively.

Note that a_1, b_1 and a_2, b_2 are voltage waves. Now in order to find a_2, b_2 in terms of a_1, b_1 we have the chain matrix of

$$\begin{bmatrix} V_1 \\ I_1 \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix} \tag{4.3}$$

Equations (4.1) and (4.3) lead to

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} 1 & R_1 \\ 1 & -R_1 \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix}$$

or

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} A + R_1C & B + R_1D \\ A - R_1C & B - R_1D \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix} \qquad (4.4)$$

But from equation (4.2)

$$\begin{bmatrix} V_2 \\ I_2 \end{bmatrix} = \begin{bmatrix} 1 & R_2 \\ 1 & -R_2 \end{bmatrix}^{-1} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix}$$

or

$$\begin{bmatrix} V_2 \\ I_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2}R_2 & -\frac{1}{2}R_2 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \qquad (4.5)$$

Thus equations (4.4) and (4.5) lead to

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} \frac{A+R_1C}{2} + \frac{1}{2R_2}(B+R_1D) & \frac{A+R_1C}{2} - \frac{1}{2R_2}(B+R_1D) \\ \frac{A-R_1C}{2} + \frac{1}{2R_2}(B-R_1D) & \frac{A-R_1C}{2} - \frac{1}{2R_2}(B-R_1D) \end{bmatrix} \begin{bmatrix} b_2 \\ a_2 \end{bmatrix} \qquad (4.6)$$

or

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} \mu & \lambda \\ \nu & \kappa \end{bmatrix} \begin{bmatrix} b_2 \\ a_2 \end{bmatrix} \qquad (4.7)$$

where

$$\mu = \frac{A+R_1C}{2} + \frac{1}{2R_2}(B+R_1D) \qquad (4.7a)$$

$$\lambda = \frac{A+R_1C}{2} - \frac{1}{2R_2}(B+R_1D) \qquad (4.7b)$$

$$\nu = \frac{A-R_1C}{2} + \frac{1}{2R_2}(B-R_1D) \qquad (4.7c)$$

$$\kappa = \frac{A-R_1C}{2} - \frac{1}{2R_2}(B-R_1D) \qquad (4.7d)$$

Thus from equation (4.7) we have

$$b_1 = \frac{\nu}{\mu} a_1 + (\kappa - \frac{\lambda\nu}{\mu}) a_2 \qquad (4.8)$$

$$b_2 = \frac{1}{\mu} a_1 - \frac{\lambda}{\mu} a_2 \qquad (4.9)$$

and the appropriate wave flow diagram is drawn in Figure 4.2.

Note that

$$V_1 = V_s - I_1R_s \qquad (4.10)$$

$$V_2 = - I_2R_L \qquad (4.11)$$

Thus equations (4.2) and (4.11) lead to

$$a_2 = V_2 + R_2I_2$$

$$b_2 = V_2 - R_2I_2$$

$$V_2 = - I_2R_L$$

so that

$$a_2 = b_2\phi \qquad (4.12)$$

$$b_2 = \frac{2V_2}{1+\phi} \qquad (4.13)$$

where

$$\phi = \frac{R_L - R_2}{R_L + R_2} \qquad (4.13a)$$

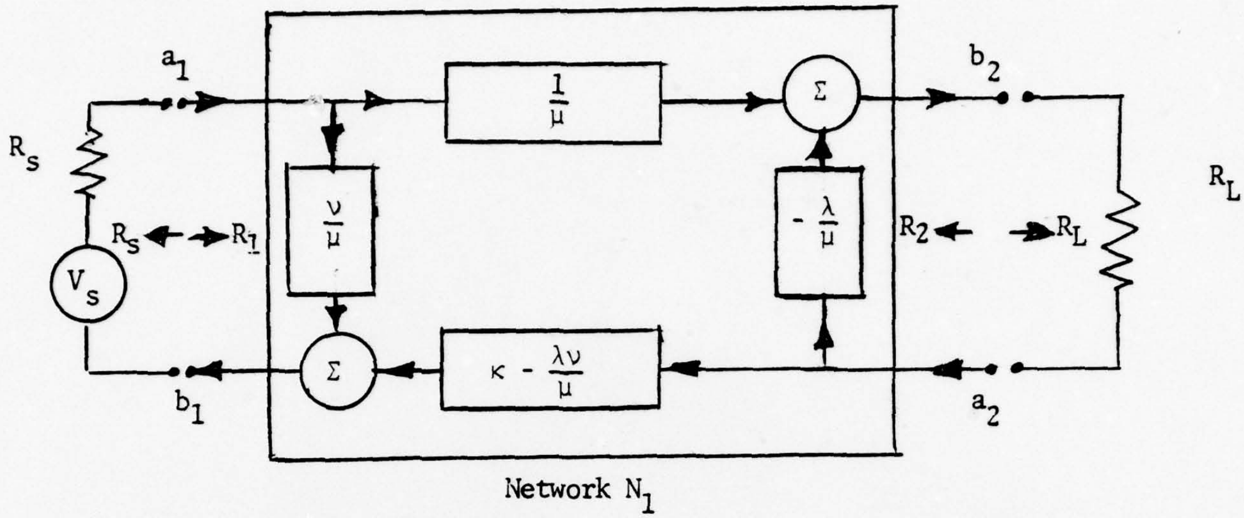


Fig. 4.2. Wave flow diagram representing network N_1 of Fig. 4.1.

and also equations (4.1) and (4.10) lead to

$$a_1 = V_1 + R_1 I_1$$

$$b_1 = V_1 - R_1 I_1$$

$$V_1 = V_s - I_1 R_s$$

so that
$$a_1 + \theta b_1 = (1 + \theta)\, V_s \tag{4.14}$$

where

$$\theta = \frac{R_1 - R_s}{R_1 + R_s} \tag{4.15}$$

Where V_s is the input voltage. Thus from equations (4.13) and (4.14) we have

$$H(z) = \frac{V_2}{V_s} = \frac{(1+\phi)(1+\theta)}{2} \cdot \frac{b_2(z)}{a_1(z)+\theta b_1(z)} \tag{4.16}$$

Note that for $\theta = 0$, from equation (4.15), $R_1 = R_s$.

$$V_s = a_s = a_1$$

and

$$H(z) = \frac{(1+\phi)}{2} \cdot \frac{b_2(z)}{a_s(z)} \tag{4.17}$$

For $\phi = 0$, from equation (4.13a), $R_2 = R_L$ and from equation (4.12) $a_2 = 0$. Thus from equation (4.16)

$$H(z) = \frac{(1+\theta)}{2} \cdot \frac{b_2(z)}{a_s(z)+\theta b_1(z)} \tag{4.18}$$

or

$$H(z) = \frac{1+\theta}{2(1 + \theta \frac{\nu}{\mu})} \cdot \frac{b_2(z)}{a_s(z)} \tag{4.19}$$

The general realization with the aid of equations (4.11), (4.14) and (4.17) is shown in Figure 4.3.

C. REALIZATION OF TWO PORT WAVE DIGITAL STRUCTURE

Theoretically LC passive structures such as ladder, lattice and most of the other symmetrical structures can be reduced to a form of Figure 4.4 which is the "T" form with positive or negative element values. Note that in A B C D matrix,the direction if I_2 is in the direction as shown in Figure 4.4 and Z_1, Z_2 or Z_3 can be positive or negative capacitors, inductors and/or tank circuits with positive or negative element values.

The A B C D matrix of Figure 4.4 as per equation (4.3) is found from the equations

$$V_1 = V_2 + I_2Z_3 + I_1Z_1$$

$$(I_1 - I_2)Z_2 = V_2 + I_2Z_3$$

and the A B C D matrix is

$$\begin{bmatrix} V_1 \\ I_1 \end{bmatrix} = \begin{bmatrix} 1 + \frac{Z_1}{Z_2} & Z_1 + Z_3 + \frac{Z_1Z_3}{Z_2} \\ \frac{1}{Z_2} & 1 + \frac{Z_3}{Z_2} \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix} \qquad (4.20)$$

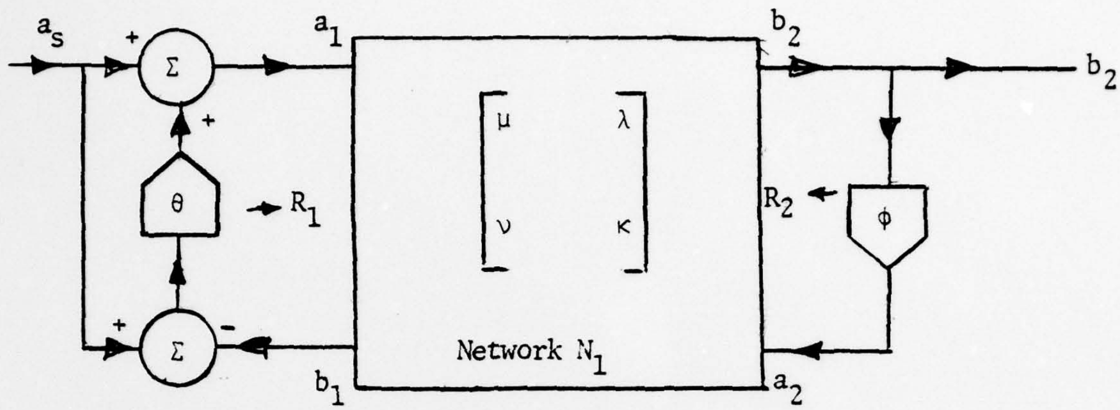
Thus

$$A = 1 + \frac{Z_1}{Z_2} \qquad (4.20a)$$

$$B = Z_1 + Z_3 + \frac{Z_1Z_3}{Z_2} \qquad (4.20b)$$

$$C = \frac{1}{Z_2} \qquad (4.20c)$$

$$D = 1 + \frac{Z_3}{Z_2} \qquad (4.20d)$$



$$\theta = \frac{R_1 - R_s}{R_1 + R_s} \qquad \phi = \frac{R_L - R_2}{R_L + R_2}$$

Fig. 4.3. General realization of a two port network N_1 with input and output waves. Note that a_s is the input voltage wave.

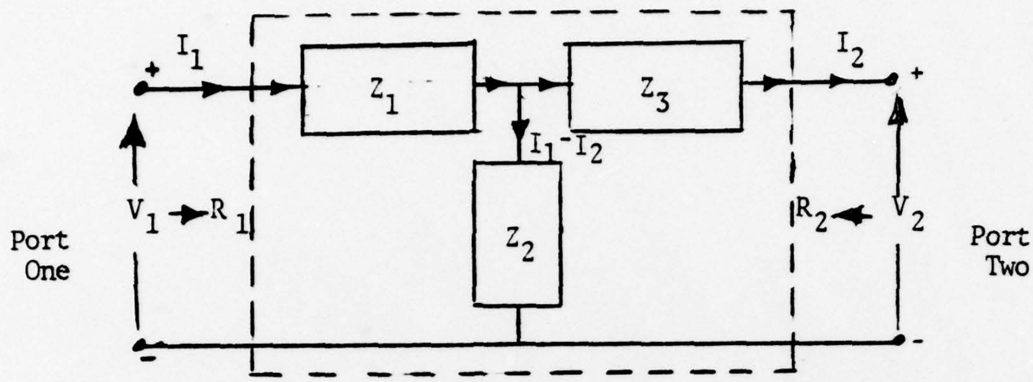


Fig. 4.4. A general two port "T" network with complex elements Z_1, Z_2, Z_3. Note that R_1 is the input impedance of Port 1, and R_2 is the input impedance of Port 2.

Note that for a single series element Z_1 as per Figure 4.5 and equation (4.20)

$$Z_3 = 0$$

$$Z_2 = \infty$$

$$\begin{bmatrix} V_1 \\ I_1 \end{bmatrix} = \begin{bmatrix} 1 & Z_1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix}$$

and for a single shunt element Z_2 as per Figure 4.6 and equation (4.20)

$$Z_1 = 0$$

$$Z_3 = 0$$

$$\begin{bmatrix} V_1 \\ I_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{1}{Z_2} & 1 \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix}$$

Now we can easily find the values of ν, λ, μ, κ of equation (4.7). Thus equations (4.7a) and (4.20) lead to

$$\mu = \frac{R_1 + R_2 + Z_1 + Z_3 + \dfrac{(R_1 + Z_1)(R_2 + Z_3)}{Z_2}}{2R_2} \qquad (4.21)$$

and equations (4.7b) and (4.20) lead to

$$\lambda = \frac{-R_1 + R_2 - Z_1 - Z_3 + \dfrac{(R_1 + Z_1)(R_2 - Z_3)}{Z_2}}{2R_2} \qquad (4.22)$$

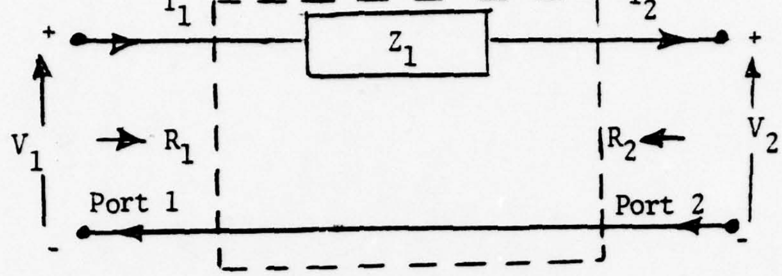


Fig. 4.5. A general two port with series element only. Note that R_1 and R_2 are input impedances of Port 1 and Port 2 respectively.

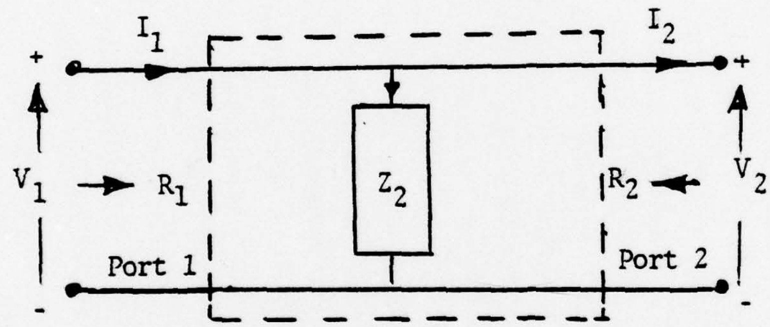


Fig. 4.6. A general two port with shunt element only. Note that R_1 and R_2 are input impedances of Port 1 and Port 2 respectively.

equations (4.7c) and (4.20) lead to

$$\nu = \frac{-R_1 + R_2 + Z_1 + Z_3 + \dfrac{(-R_1 + Z_1)(R_2 + Z_3)}{Z_2}}{2R_2} \tag{4.23}$$

equations (4.7d) and (4.20) lead to

$$\kappa = \frac{R_1 + R_2 - Z_1 + Z_3 + \dfrac{(Z_1 - R_1)(R_2 - Z_3)}{Z_2}}{2R_2} \tag{4.24}$$

Thus the elements of Figure 4.2 will be

$$\frac{1}{\mu} = \frac{2R_2}{R_1 + R_2 + Z_1 + Z_3 + \dfrac{(R_1 + Z_1)(R_2 + Z_3)}{Z_2}} \tag{4.25}$$

$$\frac{\nu}{\mu} = \frac{-R_1 + R_2 + Z_1 + Z_3 + \dfrac{(-R_1 + Z_1)(R_2 + Z_3)}{Z_2}}{R_1 + R_2 + Z_1 + Z_3 + \dfrac{(R_1 + Z_1)(R_2 + Z_3)}{Z_2}} \tag{4.26}$$

$$\kappa - \frac{\lambda\nu}{\mu} = \frac{2R_1}{R_1 + R_2 + Z_1 + Z_3 + \dfrac{(R_1 + Z_1)(R_2 - Z_3)}{Z_2}} \tag{4.27}$$

$$-\frac{\lambda}{\mu} = \frac{R_1 - R_2 + Z_1 + Z_3 - \dfrac{(R_1 + Z_1)(R_2 - Z_3)}{Z_2}}{R_1 + R_2 + Z_1 + Z_3 \quad \dfrac{(R_1 + Z_1)(R_2 - Z_3)}{Z_2}} \tag{4.28}$$

Note that for the single series element of Figure 4.5 the equations (4.25), (4.26), (4.27), (4.28) reduce to

$$\frac{1}{\mu} = \frac{2R_2}{R_1 + R_2 + Z_1} \tag{4.29}$$

$$\frac{\nu}{\mu} = \frac{-R_1 + R_2 + Z_1}{R_1 + R_2 + Z_1} \tag{4.30}$$

$$\kappa - \frac{\lambda\nu}{\mu} = \frac{2R_1}{R_1 + R_2 + Z_1} \tag{4.31}$$

$$-\frac{\lambda}{\mu} = \frac{R_1 - R_2 + Z_1}{R_1 + R_2 + Z_1} \tag{4.32}$$

and for a single shunt element of Figure 4.6 the aforesaid equations will be

$$\frac{1}{\mu} = \frac{2G_1}{G_1 + G_2 + Y_2} \tag{4.33}$$

$$\frac{\nu}{\mu} = \frac{-Y_2 - G_1 + G_2}{Y_2 + G_1 + G_2} \tag{4.34}$$

$$\kappa - \frac{\lambda\nu}{\mu} = \frac{2G_2}{Y_2 + G_1 + G_2} \tag{4.35}$$

$$-\frac{\lambda}{\mu} = \frac{-Y_2 + G_1 - G_2}{Y_2 + G_1 + G_2} \tag{4.36}$$

where

$$G_1 = \frac{1}{R_1}$$

and

$$G_2 = \frac{1}{R_2}$$

$$Y_2 = \frac{1}{Z_2}$$

D. DERIVATION OF GENERALIZED ALGORITHM WITH NO DELAY FREE PATH IN PORT ONE OR SIMILARLY WITH NO DELAY FREE PATH IN PORT TWO

The wave flow diagram of Figure 4.2 is not suitable to be used in the chain matrix equation,since most probably it has delay free paths between a_1 to b_1 and a_2 to b_2 which we have not analyzed as yet,and if so it is not desirable. Thus by a careful choice of either R_1 or R_2 we can force the wave flow from either a_1 to b_1 to have no delay free path, or from a_2 to b_2 to have no delay free path; but not both of them since we have only one variable to adjust, and that is either R_1 or R_2.

The delay free wave flow for the case of no delay free path from a_1 to b_1 for a three section filter is shown in Figure 4.7 and the delay free wave flow for the case of no delay free path from a_2 to b_2 for the same section filter is shown in Figure 4.8.

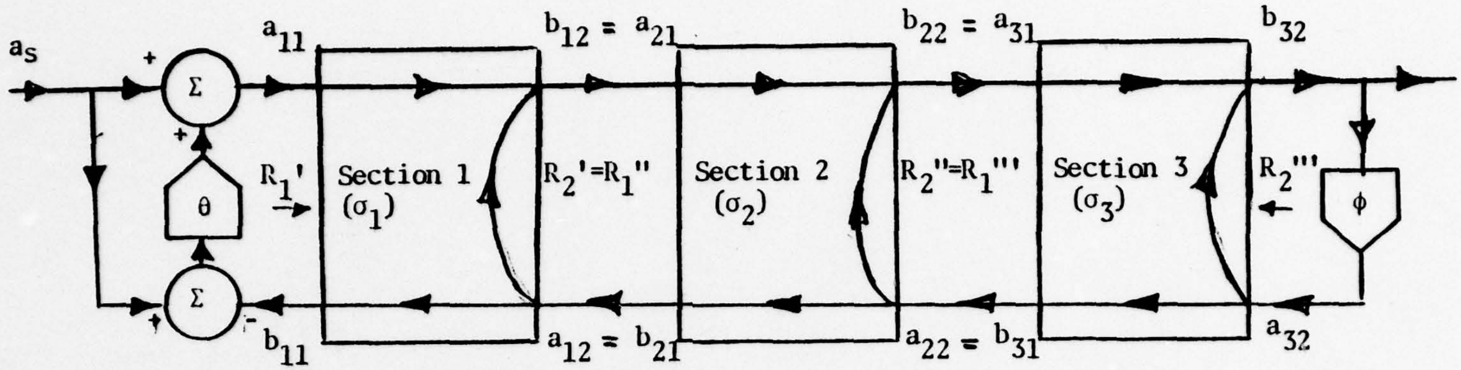
Note that a close examination of Figure 4.7 and Figure 4.8 still reveals a delay free feedback loop at the terminating points. Thus to make the sections of practical value, ϕ must be forced equal to zero for Figure 4.7 implying that

$$\phi = \frac{R_L - R_2}{R_L + R_2} = 0 \quad \text{or} \quad R_L = R_2 \tag{4.37}$$

and for the Figure 4.8, θ must be forced equal to zero implying that

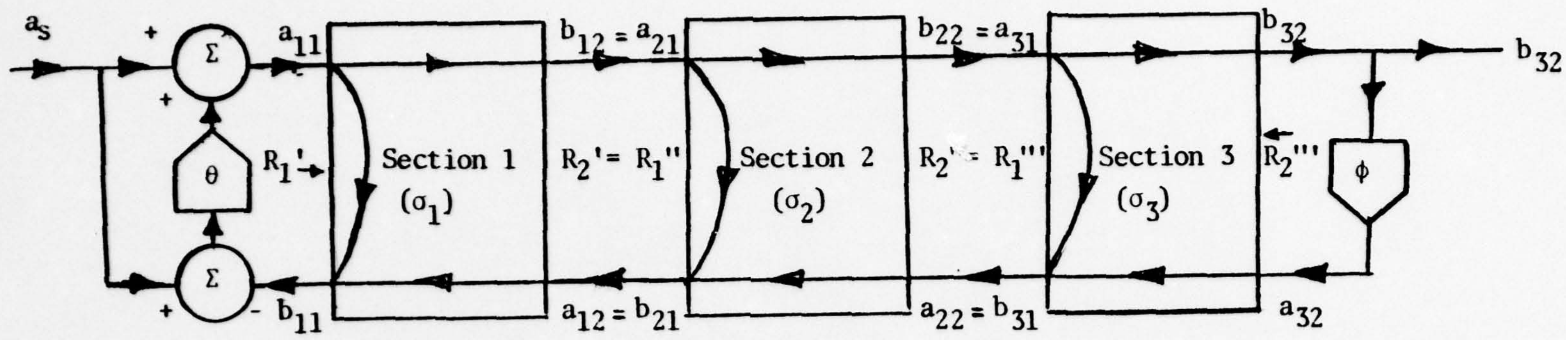
$$\theta = \frac{R_1''' - R_s}{R_1''' + R_s} = 0 \quad \text{or} \quad R_s = R_1''' \tag{4.38}$$

Thus the updated and useful delay free wave flow diagrams would be as per Figures 4.9 and 4.10. Note that in the first case we have to start at load end and work our way through to the source end, and in the second



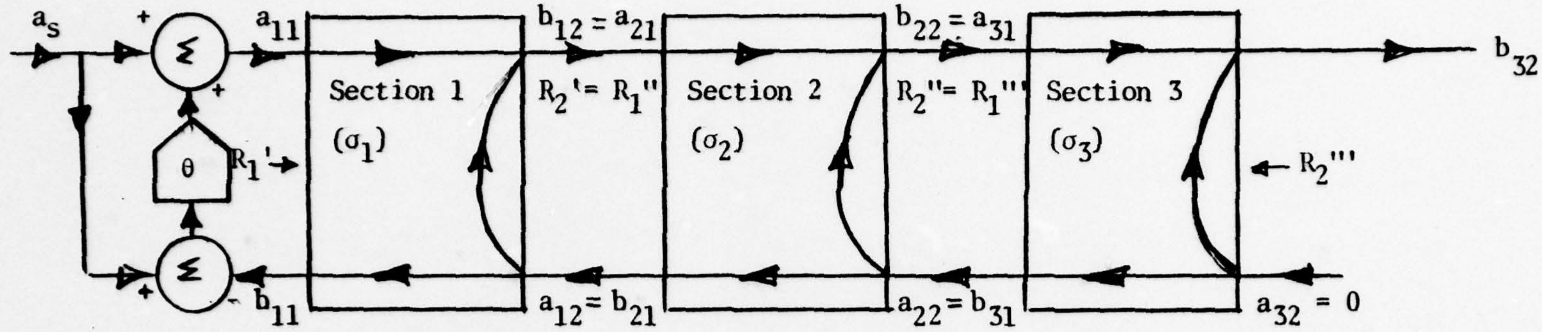
$$\theta = \frac{R_s - R_1'}{R_s + R_1'} \qquad \phi = \frac{R_L - R_2'''}{R_L + R_2'''}$$

Figure 4.7. Signal flow graph of the direct delay free paths for three cascaded two ports with no delay free path from a_1 to b_1, with multiplier coefficients σ_1, σ_2, and σ_3. Note that this is not a complete signal flow graph since delayed signal paths are not shown. Note also that R_1' is the input impedance of the port one of the section 1, and R_2''' is the input impedance of the port two of the section 3.



$$\theta = \frac{R_s - R_1'}{R_s + R_1'} \qquad \phi = \frac{R_L - R_2'''}{R_L + R_2'''}$$

Fig. 4.8. Signal flow graph of the direct delay free paths for a three cascaded two ports with no delay free path from a_2 to b_2. Note that this is not a complete signal flow graph since the delayed signal paths are not shown. Note also that R_1' is the input impedance of the port one of the section 1, and R_2''' is the input impedance of the port two of the section 3.



$$\theta = \frac{R_s - R_1'}{R_s + R_1'}$$

$$R_L = R_2'''$$

Figure 4.9. Signal flow graph of Fig. 4.7 modified to eliminate the existing delay free loop at the terminating load (no delay free path from a_1 to b_1).

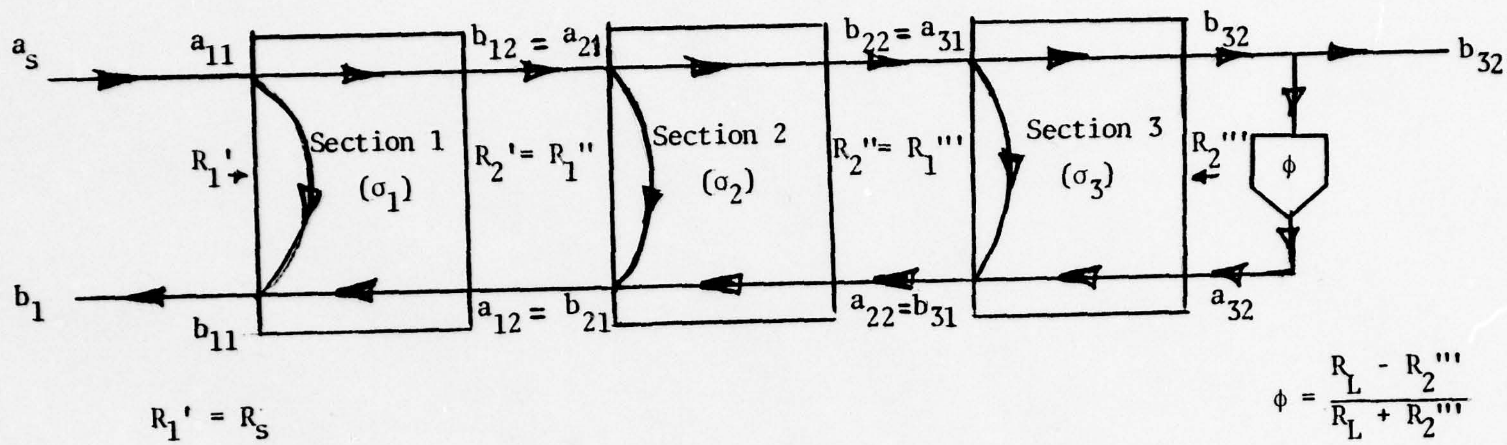


Figure 4.10. Signal flow graph of Fig. 4.8 modified to eliminate the existing delay free loop at the terminating source (no delay free path from a_2 to b_2).

case, i.e. Fig. 4.10, we have to start at the source and work through to the load end. Alternately we can start from both ends and progress towards each other. Where they meet we have to introduce a transformer or unit element such as Kuroda's identity, but this is not really recommended since it introduces further complications. This approach is not desirable and is to be avoided.

Now with the aid of formulas (4.25), (4.26), (4.27), (4.28), we can cascade as many elements in a section as complex as Figure 4.4 or as simple as Figure 4.5 or Figure 4.6.

Note that the element values Z_1, Z_2 and Z_3 must be in s domain, and for digitization we use the bilinear transformation

$$s = \frac{2}{T}\frac{(z-1)}{(z+1)} \tag{4.39}$$

where T is the sampling period of the digital filter.

Example 4-1

Series L, no delay free path from a_1 to b_1, the case of a simple element

With reference to Figure 4.2, ν/μ must have no delay free path, and the aim is to find R_1 given R_2, the load end terminating resistance. From equation (4.30)

$$\frac{\nu}{\mu} = \frac{Z_1 - R_1 + R_2}{Z_1 + R_1 + R_2}$$

and

$$Z_1 = sL = \frac{2}{T}\frac{(z-1)}{(z+1)} \cdot L$$

Thus

$$\frac{\nu}{\mu} = \frac{z\cdot(\frac{2L}{T} + R_2 - R_1) + (R_2 - R_1 - \frac{2L}{T})}{z\cdot(\frac{2L}{T} + R_1 + R_2) + (R_1 + R_2 - \frac{2L}{T})} \tag{4.40}$$

for $\frac{\nu}{\mu}$ to be with no delay free path from equation (4.40)

$$\frac{2L}{T} + R_2 - R_1 = 0$$

which leads to

$$R_1 = R_2 + \frac{2L}{T} \tag{4.41}$$

Thus from equations (4.29), (4.30), (4.31), (4.32), (4.38), (4.39), (4.41) we have

$$\frac{\nu}{\mu} = \frac{\sigma - 1}{z + \sigma} \quad \text{where} \quad \sigma = \frac{R_2}{R_1} \tag{4.42}$$

$$\frac{1}{\mu} = \frac{2R_2}{Z_1 + R_1 + R_2}$$

$$= \frac{\sigma(z+1)}{(z+\sigma)} \tag{4.43}$$

$$\kappa - \frac{\lambda\nu}{\mu} = \frac{2R_1}{R_1 + R_2 + Z_1} \tag{4.44}$$

$$= \frac{z+1}{z+\sigma}$$

$$-\frac{\lambda}{\nu} = \frac{R_1 - R_2 + Z_1}{R_1 + R_2 + Z_1} \tag{4.45}$$

$$= \frac{z(1-\sigma)}{z+\sigma}$$

Thus the output equations for series L with no delay free path from a_1 to b_1 are

$$b_1 = \frac{\sigma - 1}{z+\sigma} a_1 + \frac{z+1}{z+\sigma} a_2$$

$$b_2 = \frac{\sigma(z+1)}{z+\sigma} a_1 + \frac{z(1-\sigma)}{z+\sigma} a_2 \tag{4.46}$$

or

$$b_1 = \frac{(\sigma-1)z^{-1}}{1 + \sigma z^{-1}} a_1 + \frac{1+z^{-1}}{1+\sigma z^{-1}} a_2$$

$$b_2 = \frac{\sigma(1+z^{-1})}{1 + \sigma z^{-1}} a_1 + \frac{(1-\sigma)}{1+\sigma z^{-1}} a_2 \tag{4.47}$$

with $$\sigma = \frac{R_2}{R_1}$$

and the iterative equations, corresponding to (4.46), (4.47) are

$$b_1(n) = (\sigma-1)a_1(n-1) + a_2(n) + a_2(n-1) - \sigma b_1(n-1) \tag{4.48}$$

$$b_2(n) = \sigma a_1(n) + \sigma a_1(n-1) + (1-\sigma)a_2(n) - \sigma b_2(n-1) \tag{4.49}$$

With zero initial conditions, i.e.

$$x_1 = 0$$

$$x_2 = 0$$

$$x_3 = 0$$

$$x_4 = 0$$

The equations (4.48), (4.49) can be rewritten as

$$b_1 = \sigma(x_1 - x_3) - x_1 - x_2 + a_2 \tag{4.50}$$

$$b_2 = \sigma(a_1 - a_2 + x_1 - x_4) + a_2 \tag{4.51}$$

and updated equations

$$x_1 = a_1 \tag{4.52}$$

$$x_2 = b_1 \tag{4.53}$$

$$x_3 = a_2 \tag{4.54}$$

$$x_4 = b_2 \tag{4.55}$$

Note that there is only two multiplications for each iteration.

Example 4-2

An example of designing the composite series and shunt element with no delay free path on port two.

As an example of a composite series and shunt algorithm, let's consider the two element section of Fig. 4.11. The aim is to derive the wave flow algorithm with no delay free path from a_2 to b_2. From equation (4.28) we have

$$-\frac{\lambda}{\mu} = \frac{R_1 - R_2 + sL - (R_1 + sL)\, R_2\, sC}{R_1 + R_2 + sL + (R_1 + sL)\, R_2\, sC}$$

$$= \left. \frac{R_1 - R_2 + s(L - R_1R_2C) - s^2(R_2LC)}{R_1 + R_2 + s(L + R_1R_2C) + s^2(R_2LC)} \right|_{s = \frac{2}{T}\frac{(z-1)}{(z+1)}}$$

$$= \frac{A_2z^2 + B_2z + C_2}{A_1z^2 + B_1z + C_1}$$

where

$$A_2 = R_1 - R_2 + \frac{2L}{T} - \frac{2R_1R_2C}{T} - \frac{4R_2LC}{T^2}$$

$$B_2 = 2(R_1 - R_2 + \frac{4R_2LC}{T^2})$$

$$C_2 = R_1 - R_2 - \frac{2L}{T} + \frac{2R_1R_2C}{T} - \frac{4R_2LC}{T^2}$$

$$A_1 = R_1 + R_2 + \frac{2L}{T} + \frac{2R_1R_2C}{T} + \frac{4R_2LC}{T^2}$$

$$B_1 = 2(R_1 + R_2 - \frac{4R_2LC}{T^2})$$

$$C_1 = R_1 + R_2 + \frac{2L}{T} - \frac{2R_1R_2C}{T} + \frac{4R_2LC}{T^2}$$

with reference to Fig. 4.2 for no delay free path on port two, $-\frac{\lambda}{\mu}$ must have no delay free path, thus

$$A_2 = 0$$

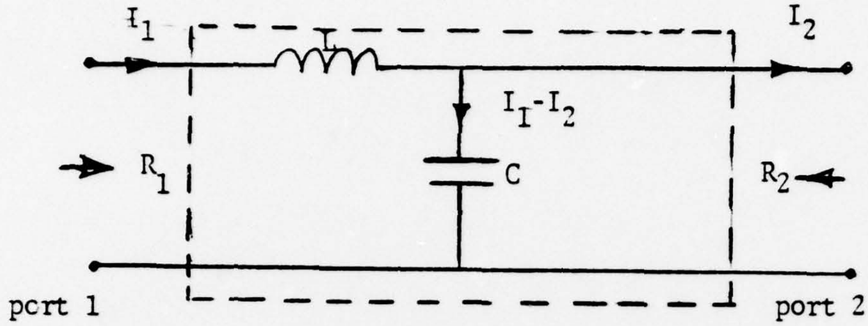


Figure 4.11. Composite series and shunt element of example 4.2.

This leads to

$$R_2 = \frac{R_1 + \frac{2L}{T}}{1 + \frac{2R_1C}{T} + \frac{4LC}{T^2}} \tag{4.57}$$

Thus from equations (4.57) and (4.28) we have

$$-\frac{\lambda}{\mu} = \frac{\beta_2 z^{-1} + \gamma_2 z^{-2}}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} \tag{4.58}$$

where

$$\beta_2 = \frac{\frac{2R_1^2C}{T} + \frac{8R_1LC}{T^2} - \frac{2L}{T} + \frac{8L^2C}{T^3}}{(R_1 + \frac{2L}{T})(1 + \frac{2R_1C}{T} + \frac{4LC}{T^2})}$$

$$\gamma_2 = \frac{\frac{2R_1^2C}{T} - \frac{2L}{T} - \frac{8L^2C}{T^3}}{(R_1 + \frac{2L}{T})(1 + \frac{2R_1C}{T} + \frac{4LC}{T^2})}$$

$$\beta_1 = \frac{2R_1 + \frac{2L}{T} + \frac{2R_1^2C}{T} - \frac{8L^2C}{T^3}}{(R_1 + \frac{2L}{T})(1 + \frac{2R_1C}{T} + \frac{4LC}{T^2})}$$

$$\gamma_1 = \frac{R_1}{(R_1 + \frac{2L}{T})(1 + \frac{2R_1C}{T} + \frac{4LC}{T^2})}$$

and from equations (4.57) and (4.25) we have

$$\frac{1}{\mu} = \frac{(1+z^{-1})^2\ \beta_3}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} \tag{4.59}$$

where

$$\beta_3 = \frac{1}{1 + \frac{2R_1C}{T} + \frac{4LC}{T^2}}$$

and from equations (4.57) and (4.27) we have

$$\kappa - \frac{\lambda\nu}{\mu} = \frac{\beta_4(1+z^{-1})^2}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} \tag{4.60}$$

where

$$\beta_4 = \frac{R_1}{R_1 + \frac{2L}{T}}$$

and from equations (4.57) and (4.26) we have

$$\frac{\nu}{\mu} = \frac{-\beta_2 z^{-1} - \gamma_2}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} \tag{4.61}$$

Thus the wave flow algorithms for the composite element with no delay free path from a_2 to b_2 will be

$$b_2 = \frac{\beta_3(1+z^{-1})^2}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} a_1 + \frac{\beta_2 z^{-1} + \gamma_2 z^{-2}}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} a_2 \tag{4.62}$$

$$b_1 = \frac{-\beta_2 z^{-1} - \gamma_2}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} a_1 + \frac{\beta_4(1+z^{-1})^2}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} a_2 \tag{4.63}$$

or

$$b_2(n) = \beta_3 a_1(n) + 2\beta_3 a_1(n-1) + \beta_3 a_1(n-2) + \beta_2 a_2(n-1) + \gamma_2 a_2(n-2) - \beta_1 b_2(n-1) - \gamma_1 b_2(n-2) \tag{4.64}$$

$$b_1(n) = -\beta_2 a_1(n-1) - \gamma_2 a_1(n) + \beta_4 a_2(n) + 2\beta_4 a_2(n-1) + \beta_4 a_2(n-2)$$

$$-\beta_1 b_1(n-1) - \gamma_1 b_1(n-2) \qquad (4.65)$$

and the iterative equation corresponding to equations (4.62), (4.63) with initial conditions set to zero, i.e.

$$x_i = 0 \qquad i=1,8$$

will lead to the iterative equations of

$$b_2 = \beta_3(a_1 + 2x_1 + x_2) + \beta_2 x_3 + \gamma_2 x_4 - \beta_1 x_7 - \gamma_1 x_8 \qquad (4.66)$$

$$b_1 = -\gamma_2 a_1 - \beta_2 x_1 + \beta_4(a_2 + 2x_3 + x_4) - \beta_1 x_5 - \gamma_1 x_6 \qquad (4.67)$$

with updated equations of

$$\begin{aligned} x_1 &= a_1 \\ x_2 &= x_1 \\ x_3 &= a_2 \\ x_4 &= x_3 \\ x_5 &= b_1 \\ x_6 &= x_5 \\ x_7 &= b_2 \\ x_8 &= x_7 \end{aligned} \qquad (4.68)$$

Note that there are 12 multiplications with six coefficients, i.e. α's, β's, γ's. It is interesting to note that since in this example we originally had two elements, i.e. Series L, and shunt C with input impedance R_1. Thus we expect the α's, β's, and γ's to be functions of R_1, L, and C only. This can be easily shown by identifying two new variables α and β such that

$$\alpha = \frac{\frac{2L}{T}}{R_1 + \frac{2L}{T}}$$

$$\beta = \frac{1}{1 + \frac{2R_1C}{T} + \frac{4LC}{T^2}}$$

Then

$$R_2^{*} = \frac{L\beta}{\alpha} = R_1 \cdot \frac{(\beta)}{1-\alpha}$$

$$\beta_1 = 1 - 2\alpha + \beta(\alpha+1)$$

$$\gamma_1 = \beta(1-\alpha)$$

$$\beta_2 = 1 - \beta(1+\alpha)$$

$$\gamma_2 = 1 - 2\alpha + \beta(\alpha-1)$$

$$\beta_3 = \beta$$

$$\beta_4 = 1-\alpha$$

---

* $R_2 = \frac{L\beta}{\alpha}$ is more general even for the case $R_1=R_s=0$ which makes $\alpha=1$ and $1-\alpha=0$ and if we use $R_2 = R_1 \frac{(\beta)}{1-\alpha}$, R_2 becomes $\frac{0}{0}$ which is indefinite, while $R_2 = \frac{L\beta}{\alpha}$ is not indefinite.

By inserting these new values into the iteration equations (4.66) and (4.67) the number of multiplications reduces from 12 to 9 which is significant. Thus the new iterative equations from (4.66), (4.67) will be

$$b_1 = a_2 - a_1 - x_1 + 2x_3 + x_4 - x_5 + \alpha(-a_2 - x_4 + 2(a_1 - x_3 + x_5)) + \beta(a_1 + x_1 - x_5 - x_6) + \alpha\beta(-a_1 + x_1 - x_5 + x_6) \tag{4.69}$$

$$b_2 = x_3 + x_4 - x_7 + 2\alpha(x_7 - x_4) + \beta(a_1 + 2x_1 + x_2 - x_3 - x_4 - x_7 - x_8) + \alpha\beta(-x_3 + x_4 - x_7 + x_8) \tag{4.70}$$

Note that the multiplication $\alpha\beta$ and 2α does not enter into the iteration and they can be premultiplied before the starting of the iteration process.

Notation

For latter analysis in Chapter V we call these latter set of equations i.e. equations (4.69) and (4.70) as "reduced parameter complex wave digital algorithms" and the original set, i.e. equations (4.66) and (4.67) as "complex wave digital algorithms".

Although theoretically it is always possible to find new variables equal to the number of original L's and C's, in practice it becomes a tedious job for more than two variables, and it is not recommended.

E. TABULATION OF SIMPLE L AND C ELEMENTS IN SERIES AND SHUNT

In a similar way as examples 1 and 2, the wave flow equations for the single L and C elements for both cases of series and shunt are tabulated in Tables 4.1a,b and 4.2a,b.

Note that Tables 4.1a, 4.1b are for the case of no delay free path from a_1 to b_1, and Tables 4.2a, 4.2b are for the case of no delay free path from a_2 to b_2.

Table 4.1a. Algorithms for simple L and C elements with no delay free path from a_1 to b_1 with sampling time T.

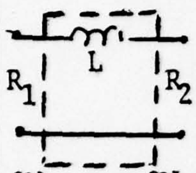
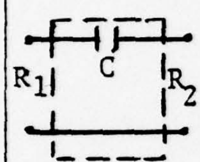
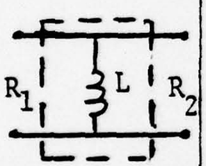
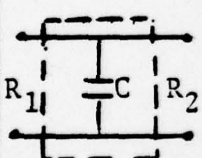
Element	Algorithm	Remarks
 (1) (2) Series L	$b_1 = \frac{\sigma-1}{z+\sigma} a_1 + \frac{z+1}{z+\sigma} a_2$ $b_2 = \frac{\sigma(z+1)}{z+\sigma} a_1 + \frac{z(1-\sigma)}{z+\sigma} a_2$	$R_1 = R_2 + \frac{2L}{T}$ $\sigma = \frac{R_2}{R_1}$
 (1) (2) Series C	$b_1 = \frac{1-\sigma}{z-\sigma} a_1 + \frac{z-1}{z-\sigma} a_2$ $b_2 = \frac{\sigma(z-1)}{z-\sigma} a_1 + \frac{z(1-\sigma)}{z-\sigma} a_2$	$R_1 = R_2 + \frac{T}{2C}$ $\sigma = \frac{R_2}{R_1}$
 (1) (2) Shunt L	$b_1 = \frac{\sigma-1}{z-\sigma} a_1 + \frac{\sigma(z-1)}{z-\sigma} a_2$ $b_2 = \frac{(z-1)}{z-\sigma} a_1 + \frac{z(\sigma-1)}{z-\sigma} a_2$	$G_1 = G_2 + \frac{T}{2L}$ $\sigma = \frac{R_1}{R_2} = \frac{G_2}{G_1}$
 (1) (2) Shunt C	$b_1 = \frac{1-\sigma}{z+\sigma} a_1 + \frac{\sigma(z+1)}{z+\sigma} a_2$ $b_2 = \frac{z+1}{z+\sigma} a_1 + \frac{z(\sigma-1)}{z+\sigma} a_2$	$G_1 = G_2 + \frac{2C}{T}$ $\sigma = \frac{R_1}{R_2} = \frac{G_2}{G_1}$

Table 4.1b. Iterative algorithms for simple L and C elements with no delay free path from a_1 to b_1 with sampling time T and with initial conditions i.e. x_1, x_2, x_3, x_4 set to zero

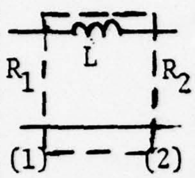
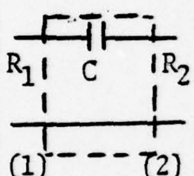
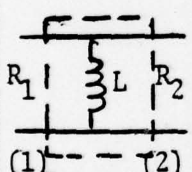
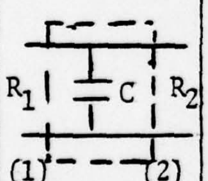
Element	Iterative Algorithm	Remarks	Updated Values
 Series L	$b_1 = a_2 - x_1 + x_2 + \sigma(x_1 - x_3)$ $b_2 = a_2 + \sigma(a_1 - a_2 + x_1 - x_4)$	$R_1 = R_2 + \frac{2L}{T}$ $\sigma = \frac{R_2}{R_1}$	$x_1 = a_1$ $x_2 = a_2$ $x_3 = b_1$ $x_4 = b_2$
 Series C	$b_1 = a_2 + x_1 - x_2 - \sigma(x_1 - x_3)$ $b_2 = a_2 + \sigma(a_1 - a_2 - x_1 + x_4)$	$R_1 = R_2 + \frac{T}{2C}$ $\sigma = \frac{R_2}{R_1}$	
 Shunt L	$b_1 = -x_1 + \sigma(a_2 + x_1 - x_2 + x_3)$ $b_2 = a_1 - a_2 - x_1 + \sigma(a_2 + x_4)$	$G_1 = G_2 + \frac{T}{2L}$ $\sigma = \frac{R_1}{R_2} = \frac{G_2}{G_1}$	
 Shunt C	$b_1 = x_1 + \sigma(a_2 - x_1 + x_2 - x_3)$ $b_2 = a_1 - a_2 + x_1 + \sigma(a_2 - x_4)$	$G_1 = G_2 + \frac{2C}{T}$ $\sigma = \frac{R_1}{R_2} = \frac{G_2}{G_1}$	

Table 4.2a. Algorithms for simple L and C elements with no delay free path from a_2 to b_2 with sampling time T

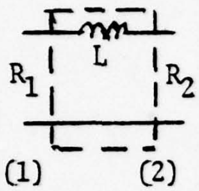
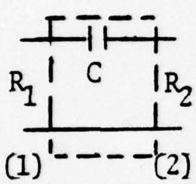
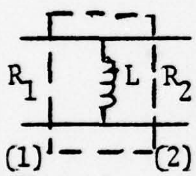
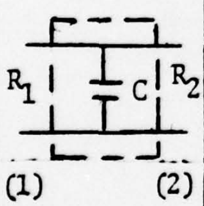
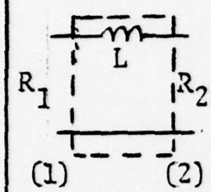
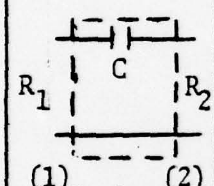
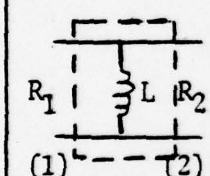
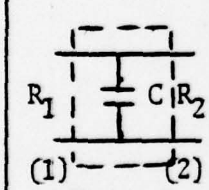
Element	Algorithm	Remarks
 Series L	$b_1 = \frac{z(1-\sigma)}{z+\sigma} a_1 + \frac{\sigma(z+1)}{z+\sigma} a_2$ $b_2 = \frac{z+1}{z+\sigma} a_1 + \frac{\sigma-1}{z+\sigma} a_2$	$R_2 = R_1 + \frac{2L}{T}$ $\sigma = \frac{R_1}{R_2}$
 Series C	$b_1 = \frac{z(1-\sigma)}{z-\sigma} a_1 + \frac{\sigma(z-1)}{z-\sigma} a_2$ $b_2 = \frac{z-1}{z-\sigma} a_1 + \frac{(1-\sigma)}{z-\sigma} a_2$	$R_2 = R_1 + \frac{T}{2C}$ $\sigma = \frac{R_1}{R_2}$
 Shunt L	$b_1 = \frac{z(\sigma-1)}{z-\sigma} a_1 + \frac{z-1}{z-\sigma} a_2$ $b_2 = \frac{\sigma(z-1)}{z-\sigma} a_1 + \frac{\sigma-1}{z-\sigma} a_2$	$G_2 = G_1 + \frac{T}{2L}$ $\sigma = \frac{G_1}{G_2} = \frac{R_1}{R_2}$
 Shunt C	$b_1 = \frac{z(\sigma-1)}{z+\sigma} a_1 + \frac{z+1}{z+\sigma} a_2$ $b_2 = \frac{\sigma(z+1)}{z+\sigma} a_1 + \frac{(1-\sigma)}{z+\sigma} a_2$	$G_2 = G_1 + \frac{2C}{T}$ $\sigma = \frac{G_1}{G_2} = \frac{R_1}{R_2}$

Table 4.2b. Iterative algorithms for simple L and C elements with no delay free path from a_2 to b_2 with sampling time T and with the initial conditions i.e. x_1, x_2, x_3, x_4 set to zero

Element	Iterative Algorithm	Remarks	Updated Values
 Series L	$b_1 = a_1+\sigma(a_2-a_1+x_2-x_3)$ $b_2 = a_1+x_1-x_2+\sigma(x_2-x_4)$	$R_2 = R_1 + \frac{2L}{T}$ $\sigma = \frac{R_1}{R_2}$	
 Series C	$b_1 = a_1+\sigma(a_2-a_1-x_2+x_3)$ $b_2 = a_1-x_1+x_2-\sigma(x_2-x_4)$	$R_2 = R_1 + \frac{T}{2C}$ $\sigma = \frac{R_1}{R_2}$	$x_1 = a_1$ $x_2 = a_2$ $x_3 = b_1$ $x_4 = b_2$
 Shunt L	$b_1 = a_2-a_1-x_2+\sigma(a_1+x_3)$ $b_2 =-x_2+\sigma(a_1-x_1+x_2+x_4)$	$G_2 = G_1 + \frac{T}{2L}$ $\sigma = \frac{G_1}{G_2} = \frac{R_1}{R_2}$	
 Shunt C	$b_1 = a_2-a_1+x_2+\sigma(a_1-x_3)$ $b_2 = x_2+\sigma(a_1+x_1-x_2-x_4)$	$G_2 = G_1 + \frac{2C}{T}$ $\sigma = \frac{G_1}{G_2} = \frac{R_1}{R_2}$	

F. FURTHER INVESTIGATION OF THE ALGORITHMS WITH NO DELAY FREE PATH IN PORT ONE, WITH THE ALGORITHMS WITH NO DELAY FREE PATH IN PORT TWO

A closer look to the two kinds of algorithms derived, i.e. algorithms with no delay free path from a_1 to b_1 and the algorithms with no delay free path from a_2 to b_2, will reveal that both algorithms would behave very much like their equivalent LC structure matched to load or matched to the source as per the reciprocity theorem. Note that this can also be proven by writing the chain matrix equations

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} \mu_1 & \lambda_1 \\ \nu_1 & \kappa_1 \end{bmatrix} \begin{bmatrix} b_2 \\ a_2 \end{bmatrix}$$ For no delay free path from a_1 to b_1

or

$$\begin{bmatrix} b_1 \\ a_1 \end{bmatrix} = \begin{bmatrix} \kappa_1 & \nu_1 \\ \lambda_1 & \mu_1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix}$$

or

$$\begin{bmatrix} a_2 \\ b_2 \end{bmatrix} = \begin{bmatrix} \kappa_1 & \nu_1 \\ \lambda_1 & \mu_1 \end{bmatrix}^{-1} \begin{bmatrix} b_1 \\ a_1 \end{bmatrix}$$ With no delay free from a_1 to b_1

So if we interchange <u>a_1 and a_2, and also b_1 and b_2</u> we would get

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} \kappa_1 & \nu_1 \\ \lambda_1 & \mu_1 \end{bmatrix}^{-1} \begin{bmatrix} b_2 \\ a_2 \end{bmatrix}$$ With no delay free from a_2 to b_2

Note that if we calculated the chain matrix for no delay free path from a_2 to b_2 i.e. the matrix

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} \mu_2 & \lambda_2 \\ \nu_2 & \kappa_2 \end{bmatrix} \begin{bmatrix} b_2 \\ a_2 \end{bmatrix}$$

Then the two matrices $\begin{bmatrix} \kappa_1 & \nu_1 \\ \lambda_1 & \mu_1 \end{bmatrix}^{-1}$ and $\begin{bmatrix} \mu_2 & \lambda_2 \\ \nu_2 & \kappa_2 \end{bmatrix}$

would be identical, i.e.

$$\begin{bmatrix} \dfrac{\mu_1}{\kappa_1\mu_1-\lambda_1\nu_1} & \dfrac{-\nu_1}{\kappa_1\mu_1-\lambda_1\nu_1} \\ \dfrac{-\lambda_1}{\kappa_1\mu_1-\lambda_1\nu_1} & \dfrac{\kappa_1}{\kappa_1\mu_1-\lambda_1\nu_1} \end{bmatrix} \equiv \begin{bmatrix} \mu_2 & \lambda_2 \\ \nu_2 & \kappa_2 \end{bmatrix}$$

Thus

$$\mu_2 = \frac{\mu_1}{\kappa_1\mu_1 - \lambda_1\nu_1}$$

$$\lambda_2 = \frac{\nu_1}{\lambda_1\nu_1 - \kappa_1\mu_1}$$

$$\nu_2 = \frac{\lambda_1}{\lambda_1\nu_1 - \kappa_1\mu_1}$$

$$\kappa_2 = \frac{\kappa_1}{\kappa_1\mu_1 - \lambda_1\nu_1}$$

G. DESIGN EXAMPLE

As an application of the illustration of the techniques developed, a seventh order low pass Chebechev filter with the given specification in

Appendix 1A was designed and implemented in both time domain and frequency domain. As a comparison the same filter was designed using the conventional digital techniques. The computer program and output results for both cases are also given in the same appendix. As can be seen, they are completely identical in both time domain and frequency domain, thus assuring the validity of the theory developed in this chapter.

Note that in this example the double precision arithmetic was used, thus ensuring infinite precision for multiplier coefficients, to come up with exactly the same answer.

H. SUMMARY

In this chapter we developed the generalized algorithm for a two port wave digital filter step by step from the first principles of circuit theory and background given in Chapters II and III. The generalized LC system considered was an LC "T" Section. For a π section the procedure would have followed in the same way with the exactly similar end results.

The delay free path does play an essential role in the algorithms. Thus emphasis was given on this matter and whenever possible explanatory delay free paths were illustrated in the diagrams. To make things as clear as possible two worked examples are given, one with a simple section with no delay free path at the terminating source port and second example with a simple complex section with no delay free path at the terminating load port. Then we investigated the relationship between matched source algorithm with matched load algorithm of a given system. Finally a practical design of a wave digital filter using the already derived algorithms were implemented and the complete results and computer program are given in the Appendix 1B,-E.

Note that no reference is given at the end of this chapter. The only relevant references would be references [9], [11] of Chapter III.

V. SENSITIVITY ANALYSIS OF THE WAVE DIGITAL FILTERS DUE TO TRUNCATION IN THE NUMBER OF BITS

A. INTRODUCTION

The intent of this chapter is to investigate the behavior of wave digital filters to quantization in the number of bits in the multiplier coefficients. The wave digital filter has been credited in most of the published literature as being minimum sensitive to multiplier coefficient variations, and thus conjectured to be optimal in requiring the fewest number of significant places to achieve a given specification.

In this chapter, for a given analogue resistively terminated LC filter, three different wave digital filter algorithms are computed, and their behavior with respect to truncation in the number of bits of the multiplier coefficients are investigated. The results are compared with conventional digital filter design. It is important to note that for all the algorithms under investigation the bilinear transformation is used to make the comparison meaningful.

In order to avoid any confusion in this chapter we use the term "multiplier coefficient" for the coefficients of wave digital filter algorithms and the term "polynomial coefficient" to denote the coefficients a_6, a_5,---,a_0 and b_6, b_5,---,b_0 of H(s) or H(z), of a given analogue or digital filter, respectively; where H(s) or H(z) are of the form

$$H(s) = \frac{1}{s^7+a_6s^6+a_5s^5+a_4s^4+a_3s^3+a_2s^2+a_1s+a_0}$$

$$H(z) = \frac{K(z+1)^7}{z^7+b_6z^6+b_5z^5+b_4z^4+b_3z^3+b_2z^2+b_1z+b_0}$$

It is also important at this stage to note that since wave digital filter multipliers are derived directly from the original L's and C's, they can be thought of as a modified or matched L or C elements. Since wave digital multipliers are not derived from the polynomial coefficients of the transfer function of filter but from L's and C's of the analogue circuit, it is meaningful to compare the effect of bit quantization error on wave digital filter multipliers with the effect of bit quantization error on L's or C's in a conventional digital filter algorithm, and not to compare with the effect of bit quantization error on the filter polynomial coefficients. Based upon this argument, the original filter L's and C's are quantized in the conventional digital filter algorithms and used for comparison purposes.

B. COMPARATIVE STUDY OF THE ERROR DUE TO QUANTIZATION IN THE NUMBER OF BITS IN VARIOUS DIGITAL FILTER ALGORITHMS

Coefficient errors introduce perturbations in the zeros and poles of the transfer functions, which in turn manifest themselves as errors in the frequency response. In Chapter III Section B we defined general sensitivity, and in Section C we discussed parameter sensitivity, and in Section D sensitivity due to quantization in the number of bits. The comparison criteria we have used in this chapter is the root mean square error criteria given in equation (3.3) and repeated here in greater detail, i.e.

$$E_{rms} = \frac{1}{N+1} \sum_{i=0}^{N} [W(\omega_i)[\Delta H(\omega_i)]^2]^{\frac{1}{2}}$$

where $W(\omega_i)$ is the frequency weighting function.

$\Delta H(\omega_i)$ is the difference in magnitude between infinite precision algorithm and the algorithm obtained in using a finite number of floating bits in the filter multiplier coefficients or component values.

N is the number of sampling points in the frequency domain.

ω_i is the frequency associated with the sampling point i

It should be pointed out that in floating point calculations, the mantissa of the floating number in the binary system is truncated to the specified length using sign magnitude representation. However the word length requirements of sign and exponent are not counted, since they do not enter into the calculations.

In total nine different normalized low pass seventh order filter algorithms were chosen for investigation from the handbook of filter synthesis [1]. These filters are

i) seventh order .5 db ripple Chebyshev low pass filter with R_S=1, R_L=1

ii) seventh order .1 db ripple Chebyshev low pass filter with R_S=1, R_L=1

iii) seventh order Butterworth low pass filter with R_S=1, R_L=1

iv) seventh order .5 db ripple Chebyshev low pass filter with R_S=0, R_L=1

v) seventh order .1 db ripple Chebyshev low pass filter with R_S=0, R_L=1

vi) seventh order Butterworth low pass filter with R_S=0, R_L=1

vii) seventh order .5 db ripple Chebyshev low pass filter with R_S=10, R_L=1

viii) seventh order .1 db ripple Chebyshev low pass filter with R_S=10, R_L=1

ix) seventh order Butterworth low pass filter with R_S=10, R_L=1

Each filter was designed using five different algorithms.

a) wave digital filter with simple sections, matched to the load (i.e. with no delay free path in port two)

b) wave digital filter with complex sections, matched to the load (i.e. with no delay free path in port two)

c) wave digital filter with complex sections but with reduced parameters and matched to the load (i.e. with no delay free path in port two

d) conventional direct digital filter of the form

$$H(z) = \frac{K(1+z^{-1})^7}{1+a_1z^{-1} + a_2z^{-2} + a_3z^{-3} + a_4z^{-4} + a_5z^{-5} + a_6z^{-6} + a_7z^{-7}}$$

(e) conventional cascaded digital filter of the form

$$H(z) = \frac{K(1+z^{-1})^7}{(1+\alpha_1z^{-1})(1+\alpha_2z^{-1} + \beta_2z^{-2})(1+\alpha_3z^{-1} + \beta_3z^{-2})(1+\alpha_4z^{-1} + \beta_4z^{-2})}$$

Note that all the filters are low pass and have zeros at $z = 1$. Thus there is no necessity for pole/zero pairing in order to optimize the response of the cascaded conventional digital filter. Thus all these examples are in a naturally optimum arrangement.

Double precision arithmetic used throughout in order to get practically the same result for all five algorithms with infinite precision arithmetic for the coefficient or component values.

The root mean square error in the frequency response for various filters under examination were found, and tabulated in Tables 5.1-9. The corresponding graphs are drawn in Figs. 5.1-9. Note that in order to avoid congestion the error graphs of conventional cascaded digital filter are not drawn.

A sample computer program for each of the five algorithms used is given in Appendix 2-A,F.

C. STUDY OF THE RESULTS OBTAINED

The study of the graphs for all different algorithms clearly indicate the following points:

No. of Bits \ Filter Algorithm	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.35652	.26039	.63499	.68992	.26039
2	.14008	.22310	.59928	.62552	.22310
3	.13521	.19962	.48117	.39121	.19962
4	$.58828\times10^{-1}$	$.95811\times10^{-1}$	.28575	.38759	$.95811\times10^{-1}$
5	$.29642\times10^{-1}$	$.61900\times10^{-1}$	.14949	.23245	$.61900\times10^{-1}$
6	$.17023\times10^{-1}$	$.3188\times10^{-1}$	$.61667\times10^{-1}$	$.78677\times10^{-1}$	$.31883\times10^{-1}$
7	$.62789\times10^{-2}$	$.12223\times10^{-1}$	$.39411\times10^{-1}$	$.46734\times10^{-1}$	$.12223\times10^{-1}$
8	$.43622\times10^{-2}$	$.43154\times10^{-2}$	$.19223\times10^{-1}$	$.24077\times10^{-1}$	$.43152\times10^{-2}$
9	$.22449\times10^{-2}$	$.24866\times10^{-2}$	$.10759\times10^{-1}$	$.12584\times10^{-1}$	$.24859\times10^{-2}$
10	$.10652\times10^{-2}$	$.17172\times10^{-2}$	$.49396\times10^{-2}$	$.69758\times10^{-2}$	$.17168\times10^{-2}$
11	$.59664\times10^{-3}$	$.82352\times10^{-3}$	$.21325\times10^{-2}$	$.11943\times10^{-2}$	$.82280\times10^{-3}$
12	$.35515\times10^{-3}$	$.21312\times10^{-3}$	$.91292\times10^{-3}$	$.11787\times10^{-2}$	$.21277\times10^{-3}$
13	$.12313\times10^{-3}$	$.14233\times10^{-3}$	$.51889\times10^{-3}$	$.67145\times10^{-3}$	$.14237\times10^{-3}$
14	$.37246\times10^{-4}$	$.84985\times10^{-4}$	$.16679\times10^{-3}$	$.30656\times10^{-3}$	$.85031\times10^{-4}$
15	$.28579\times10^{-4}$	$.39082\times10^{-4}$	$.10920\times10^{-3}$	$.15927\times10^{-3}$	$.39202\times10^{-4}$
16	$.15686\times10^{-4}$	$.28431\times10^{-4}$	$.85767\times10^{-4}$	$.79318\times10^{-4}$	$.28112\times10^{-4}$
17	$.96697\times10^{-5}$	$.13855\times10^{-4}$	$.29565\times10^{-4}$	$.65843\times10^{-4}$	$.13557\times10^{-4}$
18	$.56751\times10^{-5}$	$.61544\times10^{-5}$	$.15462\times10^{-4}$	$.19280\times10^{-4}$	$.61192\times10^{-5}$
19	$.26435\times10^{-5}$	$.12165\times10^{-5}$	$.68256\times10^{-5}$	$.82248\times10^{-5}$	$.12755\times10^{-5}$
20	$.12837\times10^{-5}$	$.42522\times10^{-6}$	$.34753\times10^{-5}$	$.26740\times10^{-5}$	$.65688\times10^{-6}$

Table 5.1. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .5 db ripple Chebyshev filter with $R_s=1.0$ (Case i). Note that all algorithms with infinite precision in the number of bits are identical.

Filter Algorithm / No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.14013	.24690	.65105	.70580	.24691
2	.12927	.12832	.56269	.64306	.12832
3	.91285x10^{-1}	.61380x10^{-1}	.47792	.44117	.61380x10^{-1}
4	.54932x10^{-1}	.29308x10^{-1}	.2885	.39337	.29308x10^{-1}
5	.34795x10^{-1}	.29308x10^{-1}	.12667	.20306	.29308x10^{-1}
6	.13549x10^{-1}	.2177 x10^{-1}	.64217x10^{-1}	.78850x10^{-1}	.47347
7	.51204x10^{-2}	.74590x10^{-2}	.26924x10^{-1}	.32705x10^{-1}	.74590x10^{-2}
8	.51204x10^{-2}	.24891x10^{-2}	.15956x10^{-1}	.14234x10^{-1}	.24892x10^{-2}
9	.23182x10^{-2}	.15261x10^{-2}	.87465x10^{-2}	.12626x10^{-1}	.15260x10^{-2}
10	.90195x10^{-3}	.10513x10^{-2}	.70939x10^{-2}	.85424x10^{-2}	.10513x10^{-2}
11	.35457x10^{-3}	.43836x10^{-3}	.24649x10^{-2}	.23713x10^{-2}	.43838x10^{-3}
12	.26058x10^{-3}	.32313x10^{-3}	.86570x10^{-3}	.11000x10^{-2}	.32304x10^{-3}
13	.12443x10^{-3}	.11146x10^{-3}	.24093x10^{-3}	.48572x10^{-3}	.11128x10^{-3}
14	.33816x10^{-4}	.49437x10^{-4}	.25903x10^{-3}	.55338x10^{-3}	.49478x10^{-4}
15	.22661x10^{-4}	.49437x10^{-4}	.16520x10^{-3}	.21469x10^{-3}	.49478x10^{-4}
16	.88987x10^{-5}	.21698x10^{-4}	.69406x10^{-4}	.11802x10^{-3}	.21610x10^{-4}
17	.58604x10^{-5}	.79930x10^{-5}	.20206x10^{-4}	.62724x10^{-4}	.79669x10^{-5}
18	.31205x10^{-5}	.40358x10^{-5}	.14222x10^{-4}	.27515x10^{-4}	.40597x10^{-5}
19	.69738x10^{-6}	.21025x10^{-5}	.62374x10^{-5}	.15394x10^{-4}	.23837x10^{-5}
20	.44562x10^{-6}	.90757x10^{-6}	.43746x10^{-5}	.92344x10^{-5}	.17713x10^{-5}

Table 5.2. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .1 db ripple Chebyshev filter with R_s=1.0 (Case ii). Note that all algorithms with infinite precision in the number of bits are identical.

Filter Algorithm / No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.15359	.37596	.62943	.70633	.55966
2	.98746x10^{-1}	.12358	.46779	.57919	.37285
3	.60823x10^{-1}	.68204x10^{-1}	.29004	.28707	.36805
4	.21522x10^{-1}	.24275x10^{-1}	.22520	.29007	.24275x10^{-1}
5	.14282x10^{-1}	.12006x10^{-1}	.13034	.18987	.12006x10^{-1}
6	.63848x10^{-2}	.12006x10^{-1}	.46460x10^{-1}	.80917x10^{-1}	.12006x10^{-1}
7	.63643x10^{-2}	.30730x10^{-2}	.23283x10^{-1}	.48606x10^{-1}	.30729x10^{-2}
8	.22526x10^{-2}	.58966x10^{-3}	.11585x10^{-1}	.17713x10^{-1}	.58954x10^{-3}
9	.75775x10^{-3}	.71222x10^{-3}	.49460x10^{-2}	.75595x10^{-2}	.71209x10^{-3}
10	.19093x10^{-3}	.35068x10^{-3}	.35341x10^{-2}	.39429x10^{-2}	.35063x10^{-3}
11	.14166x10^{-3}	.35068x10^{-3}	.18855x10^{-2}	.15097x10^{-2}	.35063x10^{-3}
12	.11492x10^{-3}	.16100x10^{-3}	.10525x10^{-2}	.13531x10^{-2}	.16087x10^{-3}
13	.30591x10^{-4}	.41379x10^{-4}	.42841x10^{-3}	.73249x10^{-3}	.41515x10^{-4}
14	.24121x10^{-4}	.13532x10^{-4}	.12317x10^{-3}	.38394x10^{-3}	.13593x10^{-4}
15	.69017x10^{-5}	.10017x10^{-4}	.11078x10^{-3}	.23523x10^{-3}	.99541x10^{-5}
16	.16907x10^{-4}	.53169x10^{-5}	.50313x10^{-4}	.99828x10^{-4}	.53993x10^{-5}
17	.12092x10^{-5}	.53169x10^{-5}	.22721x10^{-4}	.39670x10^{-4}	.53993x10^{-5}
18	.19942x10^{-5}	.12472x10^{-5}	.19869x10^{-4}	.25055x10^{-4}	.12047x10^{-5}
19	.12186x10^{-5}	.37570x10^{-6}	.90969x10^{-5}	.77472x10^{-5}	.68455x10^{-6}
20	.48111x10^{-6}	.43956x10^{-6}	.38317x10^{-5}	.74041x10^{-5}	.86858x10^{-6}

Table 5.3. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass Butterworth filter with $R_s = 1.0$ (Case iii). Note that all algorithms with infinite precision in the number of bits are identical.

No. of Bits \ Filter Algorithm	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.20668	.39184	.65288	.66916	.59279
2	.20668	.13616	.59238	.60740	.13616
3	.12086	$.92858x10^{-1}$	.47506	.40624	$.92859x10^{-1}$
4	$.96439x10^{-1}$	$.40249x10^{-1}$	.23035	.35495	$.40249x10^{-1}$
5	$.37448x10^{-1}$	$.18858x10^{-1}$	.10805	.20654	$.18858x10^{-1}$
6	$.13055x10^{-1}$	$.92725x10^{-2}$	$.66144x10^{-1}$	$.77421x10^{-1}$	$.92719x10^{-2}$
7	$.72820x10^{-2}$	$.92725x10^{-2}$	$.41680x10^{-1}$	$.28951x10^{-1}$	$.92719x10^{-2}$
8	$.46487x10^{-2}$	$.47486x10^{-2}$	$.22673x10^{-1}$	$.26828x10^{-1}$	$.47485x10^{-2}$
9	$.26696x10^{-2}$	$.21632x10^{-2}$	$.10407x10^{-1}$	$.13094x10^{-1}$	$.21632x10^{-2}$
10	$.14931x10^{-2}$	$.11369x10^{-2}$	$.53214x10^{-2}$	$.69720x10^{-2}$	$.11365x10^{-2}$
11	$.52650x10^{-3}$	$.40809x10^{-3}$	$.30710x10^{-2}$	$.11811x10^{-2}$	$.40696x10^{-3}$
12	$.22758x10^{-3}$	$.15602x10^{-3}$	$.18877x10^{-2}$	$.12389x10^{-2}$	$.15576x10^{-3}$
13	$.12816x10^{-3}$	$.87660x10^{-4}$	$.89403x10^{-3}$	$.41425x10^{-3}$	$.87011x10^{-4}$
14	$.66198x10^{-4}$	$.89247x10^{-4}$	$.30364x10^{-3}$	$.38149x10^{-3}$	$.88559x10^{-4}$
15	$.31131x10^{-4}$	$.37404x10^{-4}$	$.11601x10^{-3}$	$.19776x10^{-3}$	$.37172x10^{-4}$
16	$.22886x10^{-4}$	$.13096x10^{-4}$	$.43989x10^{-4}$	$.89407x10^{-4}$	$.13564x10^{-4}$
17	$.75737x10^{-5}$	$.94055x10^{-5}$	$.30817x10^{-4}$	$.40262x10^{-4}$	$.91978x10^{-5}$
18	$.31208x10^{-5}$	$.5592\ x10^{-5}$	$.23248x10^{-4}$	$.19890x10^{-4}$	$.55810x10^{-5}$
19	$.20293x10^{-5}$	$.19806x10^{-5}$	$.93558x10^{-5}$	$.60506x10^{-5}$	$.26475x10^{-5}$
20	$.12448x10^{-5}$	$.81813x10^{-6}$	$.49603x10^{-5}$	$.54628x10^{-5}$	$.11476x10^{-5}$

Table 5.4. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .5 db ripple Chebyshev filter with $R_S=0$ (Case iv). Note that all algorithms with infinite precision in the number of bits are identical.

Filter Algorithm / No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.25667	.38205	.67249	.68073	.58086
2	.18028	.31179	.58285	.61156	.31179
3	.11517	.15140	.46909	.39879	.23871
4	$.46371 \times 10^{-1}$	$.49949 \times 10^{-1}$	.22673	.36262	.37667
5	$.26852 \times 10^{-1}$	$.15935 \times 10^{-1}$	.15270	.19724	$.15935 \times 10^{-1}$
6	$.10936 \times 10^{-1}$	$.88682 \times 10^{-2}$	$.70773 \times 10^{-1}$	.11305	.44566
7	$.41706 \times 10^{-2}$	$.66298 \times 10^{-2}$	$.37405 \times 10^{-1}$	$.75338 \times 10^{-1}$	$.66297 \times 10^{-2}$
8	$.22884 \times 10^{-2}$	$.27692 \times 10^{-2}$	$.21136 \times 10^{-1}$	$.35243 \times 10^{-1}$	$.27690 \times 10^{-2}$
9	$.19746 \times 10^{-2}$	$.16209 \times 10^{-2}$	$.75015 \times 10^{-1}$	$.20651 \times 10^{-1}$	$.16208 \times 10^{-2}$
10	$.89626 \times 10^{-3}$	$.96046 \times 10^{-3}$	$.45627 \times 10^{-2}$	$.91342 \times 10^{-2}$	$.96083 \times 10^{-3}$
11	$.39825 \times 10^{-3}$	$.38064 \times 10^{-3}$	$.2725 \times 10^{-2}$	$.32694 \times 10^{-2}$	$.38038 \times 10^{-3}$
12	$.10908 \times 10^{-3}$	$.24080 \times 10^{-3}$	$.10593 \times 10^{-2}$	$.34804 \times 10^{-3}$	$.24081 \times 10^{-3}$
13	$.93393 \times 10^{-4}$	$.88964 \times 10^{-4}$	$.67468 \times 10^{-3}$	$.34342 \times 10^{-3}$	$.89049 \times 10^{-4}$
14	$.39722 \times 10^{-4}$	$.35992 \times 10^{-4}$	$.3080 \times 10^{-3}$	$.31101 \times 10^{-3}$	$.35926 \times 10^{-4}$
15	$.25247 \times 10^{-4}$	$.31273 \times 10^{-4}$	$.17009 \times 10^{-3}$	$.28543 \times 10^{-3}$	$.31234 \times 10^{-4}$
16	$.20389 \times 10^{-4}$	$.11650 \times 10^{-4}$	$.83274 \times 10^{-4}$	$.10079 \times 10^{-3}$	$.11775 \times 10^{-4}$
17	$.90891 \times 10^{-5}$	$.63657 \times 10^{-5}$	$.34725 \times 10^{-4}$	$.60767 \times 10^{-4}$	$.67364 \times 10^{-5}$
18	$.37639 \times 10^{-5}$	$.42258 \times 10^{-5}$	$.21278 \times 10^{-4}$	$.17197 \times 10^{-4}$	$.41910 \times 10^{-5}$
19	$.15389 \times 10^{-5}$	$.23810 \times 10^{-5}$	$.11086 \times 10^{-4}$	$.19679 \times 10^{-5}$	$.22792 \times 10^{-5}$
20	$.92275 \times 10^{-6}$	$.12793 \times 10^{-5}$	$.32788 \times 10^{-5}$	$.15601 \times 10^{-5}$	$.13736 \times 10^{-5}$

Table 5.5. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .1 db ripple Chebyshev filter with R_s=0 (Case v). Note that all algorithms with infinite precision in the number of bits are identical.

Filter Algorithm / No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.21385	.35023	.63007	.67050	.60960
2	.21684	.22495	.34968	.48535	.22496
3	.11543	.10316	.26403	.31619	.45125
4	$.37482x10^{-1}$	$.21924x10^{-1}$	.21129	.24584	.50695
5	$.15989x10^{-1}$	$.21324x10^{-1}$	.10929	$.70189x10^{-1}$	.13494
6	$.65223x10^{-2}$	$.11353x10^{-1}$	.57963x10-1	$.64994x10^{-1}$	$.11353x10^{-1}$
7	$.65223x10^{-2}$	$.27292x10^{-2}$	$.32092x10^{-1}$	$.24383x10^{-1}$	.41748
8	$.24615x10^{-2}$	$.25787x10^{-2}$	$.20078x10^{-1}$	$.24989x10^{-1}$	.33654
9	$.65170x10^{-3}$	$.16855x10^{-2}$	$.91341x10^{-2}$	$.10063x10^{-1}$	$.16854x10^{-2}$
10	$.41747x10^{-3}$	$.95453x10^{-3}$	$.56683x10^{-2}$	$.14316x10^{-2}$	$.95454x10^{-3}$
11	$.10700x10^{-3}$	$.46437x10^{-3}$	$.23622x10^{-2}$	$.90073x10^{-3}$	$.46429x10^{-3}$
12	$.21860x10^{-3}$	$.17166x10^{-3}$	$.76595x10^{-3}$	$.67584x10^{-3}$	$.17162x10^{-3}$
13	$.13791x10^{-3}$	$.59632x10^{-4}$	$.38607x10^{-3}$	$.67584x10^{-3}$	$.59672x10^{-4}$
14	$.73024x10^{-4}$	$.38676x10^{-4}$	$.13104x10^{-3}$	$.29072x10^{-3}$	$.38583x10^{-4}$
15	$.22530x10^{-4}$	$.24061x10^{-4}$	$.66822x10^{-4}$	$.10959x10^{-3}$	$.24105x10^{-4}$
16	$.12791x10^{-4}$	$.89135x10^{-5}$	$.37915x10^{-4}$	$.19217x10^{-4}$	$.88321x10^{-5}$
17	$.54724x10^{-5}$	$.31105x10^{-5}$	$.17687x10^{-4}$	$.15600x10^{-4}$	$.33782x10^{-5}$
18	$.10315x10^{-5}$	$.83639x10^{-6}$	$.13154x10^{-4}$	$.17382x10^{-4}$	$.10057x10^{-5}$
19	$.11843x10^{-5}$	$.94314x10^{-6}$	$.10485x10^{-4}$	$.57124x10^{-5}$	$.18559x10^{-5}$
20	$.56671x10^{-6}$	$.61413x10^{-6}$	$.42001x10^{-5}$	$.30040x10^{-5}$	$.64950x10^{-6}$

Table 5.6. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass Butterworth filter with R_s=0.0 (Case vi). Note that all algorithms with infinite precision in the number of bits are identical.

Filter Algorithm / No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	3.4045	.36760	.39479	.64376	.36760
2	2.5171	.25718	.25614	.44176	.25718
3	.75203	.12198	.20941	.17782	.12198
4	.10343	$.59672 \times 10^{-1}$	.20967	.32729	$.59673 \times 10^{-1}$
5	$.71300 \times 10^{-1}$	$.58777 \times 10^{-1}$	$.91164 \times 10^{-1}$	.16175	$.58778 \times 10^{-1}$
6	$.63749 \times 10^{-1}$	$.22581 \times 10^{-1}$	$.23961 \times 10^{-1}$	$.75455 \times 10^{-1}$	$.22582 \times 10^{-1}$
7	$.52302 \times 10^{-1}$	$.10077 \times 10^{-1}$	$.15663 \times 10^{-1}$	$.26816 \times 10^{-1}$	$.10078 \times 10^{-1}$
8	$.10162 \times 10^{-1}$	$.37212 \times 10^{-2}$	$.72677 \times 10^{-2}$	$.31405 \times 10^{-2}$	$.37218 \times 10^{-2}$
9	$.65050 \times 10^{-2}$	$.12500 \times 10^{-2}$	$.39261 \times 10^{-2}$	$.24761 \times 10^{-2}$	$.12502 \times 10^{-2}$
10	$.54846 \times 10^{-2}$	$.75861 \times 10^{-3}$	$.24435 \times 10^{-2}$	$.25190 \times 10^{-2}$	$.75847 \times 10^{-3}$
11	$.51118 \times 10^{-2}$	$.50342 \times 10^{-3}$	$.32102 \times 10^{-2}$	$.35433 \times 10^{-2}$	$.50396 \times 10^{-3}$
12	$.23803 \times 10^{-2}$	$.43317 \times 10^{-3}$	$.16176 \times 10^{-2}$	$.1225 \times 10^{-2}$	$.43352 \times 10^{-3}$
13	$.77886 \times 10^{-3}$	$.18262 \times 10^{-3}$	$.38714 \times 10^{-3}$	$.39487 \times 10^{-3}$	$.18315 \times 10^{-3}$
14	$.72844 \times 10^{-4}$	$.12202 \times 10^{-3}$	$.27273 \times 10^{-3}$	$.43587 \times 10^{-3}$	$.12213 \times 10^{-3}$
15	$.95398 \times 10^{-4}$	$.52849 \times 10^{-4}$	$.90359 \times 10^{-4}$	$.80415 \times 10^{-4}$	$.53353 \times 10^{-4}$
16	$.68363 \times 10^{-4}$	$.33381 \times 10^{-4}$	$.39191 \times 10^{-4}$	$.72033 \times 10^{-4}$	$.33771 \times 10^{-4}$
17	$.80049 \times 10^{-4}$	$.13892 \times 10^{-4}$	$.51176 \times 10^{-4}$	$.36564 \times 10^{-4}$	$.13793 \times 10^{-4}$
18	$.34872 \times 10^{-4}$	$.52833 \times 10^{-5}$	$.18834 \times 10^{-4}$	$.19394 \times 10^{-4}$	$.54028 \times 10^{-5}$
19	$.11083 \times 10^{-4}$	$.16061 \times 10^{-5}$	$.28221 \times 10^{-5}$	$.93569 \times 10^{-5}$	$.16240 \times 10^{-5}$
20	$.93252 \times 10^{-5}$	$.11946 \times 10^{-5}$	$.47440 \times 10^{-5}$	$.54222 \times 10^{-5}$	$.12857 \times 10^{-5}$

Table 5.7. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .5 db ripple Chebyshev filter with R_s=10 (Case vii). Note that all algorithms with infinite precision in the number of bits are identical.

Filter Algorithm / No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	4.9330	.34028	.22253	.60537	.34028
2	2.3234	.25423	.25134	.39941	.25423
3	.60260	.11162	.24523	.19258	.11162
4	$.28045 \times 10^{-1}$	.11162	.25260	.34407	.11162
5	$.19225 \times 10^{-1}$	$.51472 \times 10^{-1}$	$.81241 \times 10^{-1}$	.14301	$.51472 \times 10^{-1}$
6	$.16671 \times 10^{-1}$	$.17350 \times 10^{-1}$	$.41152 \times 10^{-1}$	$.52584 \times 10^{-1}$	$.1735 \times 10^{-1}$
7	$.14588 \times 10^{-1}$	$.90435 \times 10^{-2}$	$.53493 \times 10^{-2}$	$.41214 \times 10^{-1}$	$.90437 \times 10^{-2}$
8	$.14292 \times 10^{-1}$	$.50883 \times 10^{-2}$	$.46919 \times 10^{-2}$	$.12899 \times 10^{-1}$	$.50883 \times 10^{-2}$
9	$.14518 \times 10^{-1}$	$.18920 \times 10^{-2}$	$.99608 \times 10^{-2}$	$.82507 \times 10^{-2}$	$.18920 \times 10^{-2}$
10	$.34443 \times 10^{-2}$	$.50810 \times 10^{-3}$	$.86433 \times 10^{-3}$	$.36649 \times 10^{-2}$	$.50805 \times 10^{-3}$
11	$.32681 \times 10^{-2}$	$.48189 \times 10^{-3}$	$.15036 \times 10^{-2}$	$.23079 \times 10^{-2}$	$.48178 \times 10^{-3}$
12	$.51894 \times 10^{-3}$	$.27238 \times 10^{-3}$	$.12471 \times 10^{-2}$	$.21315 \times 10^{-3}$	$.27251 \times 10^{-3}$
13	$.65150 \times 10^{-3}$	$.14401 \times 10^{-3}$	$.28402 \times 10^{-3}$	$.81029 \times 10^{-3}$	$.14397 \times 10^{-3}$
14	$.69939 \times 10^{-3}$	$.86010 \times 10^{-4}$	$.46751 \times 10^{-3}$	$.32312 \times 10^{-3}$	$.86135 \times 10^{-4}$
15	$.33369 \times 10^{-3}$	$.20719 \times 10^{-4}$	$.21478 \times 10^{-3}$	$.15872 \times 10^{-3}$	$.20736 \times 10^{-4}$
16	$.15398 \times 10^{-3}$	$.14006 \times 10^{-4}$	$.78033 \times 10^{-4}$	$.79171 \times 10^{-4}$	$.14442 \times 10^{-2}$
17	$.72318 \times 10^{-4}$	$.80534 \times 10^{-5}$	$.38563 \times 10^{-4}$	$.46419 \times 10^{-4}$	$.84066 \times 10^{-5}$
18	$.35157 \times 10^{-4}$	$.23806 \times 10^{-5}$	$.23438 \times 10^{-4}$	$.17891 \times 10^{-4}$	$.24410 \times 10^{-5}$
19	$.14323 \times 10^{-4}$	$.13523 \times 10^{-5}$	$.87165 \times 10^{-5}$	$.91258 \times 10^{-5}$	$.15303 \times 10^{-5}$
20	$.39038 \times 10^{-5}$	$.63022 \times 10^{-6}$	$.11030 \times 10^{-5}$	$.50819 \times 10^{-5}$	$.10907 \times 10^{-5}$

Table 5.8. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .1 db ripple Chebyshev filter with R_s=10 (Case viii). Note that all algorithms with infinite precision in the number of bits are identical.

Filter Algorithm / No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	4.9388	.31595	.43082	.42707	.31595
2	1.7731	.19920	.21663	.20286	.19920
3	.73610	$.53069\times10^{-1}$	.21097	.27391	.33561
4	$.19259\times10^{-1}$	$.40313\times10^{-1}$	.18841	.20363	.32817
5	$.19821\times10^{-1}$	$.19975\times10^{-1}$	$.91724\times10^{-1}$	.13022	.51223
6	$.17707\times10^{-1}$	$.12376\times10^{-1}$	$.26432\times10^{-1}$	$.58937\times10^{-1}$	.29823
7	$.96715\times10^{-2}$	$.30099\times10^{-2}$	$.13196\times10^{-1}$	$.18244\times10^{-1}$	.32534
8	$.12042\times10^{-1}$	$.29839\times10^{-2}$	$.27875\times10^{-2}$	$.13429\times10^{-1}$	$.29839\times10^{-2}$
9	$.11451\times10^{-1}$	$.12108\times10^{-2}$	$.57712\times10^{-2}$	$.60203\times10^{-2}$	$.12108\times10^{-2}$
10	$.19437\times10^{-2}$	$.45825\times10^{-3}$	$.17493\times10^{-2}$	$.40077\times10^{-2}$	$.45820\times10^{-3}$
11	$.12416\times10^{-2}$	$.45719\times10^{-3}$	$.52506\times10^{-3}$	$.17260\times10^{-2}$	$.45717\times10^{-3}$
12	$.14031\times10^{-2}$	$.18987\times10^{-3}$	$.72945\times10^{-3}$	$.75322\times10^{-3}$	$.18979\times10^{-3}$
13	$.22524\times10^{-3}$	$.72993\times10^{-4}$	$.20080\times10^{-3}$	$.19607\times10^{-3}$	$.72942\times10^{-4}$
14	$.16257\times10^{-3}$	$.36324\times10^{-4}$	$.87813\times10^{-4}$	$.96061\times10^{-4}$	$.36521\times10^{-4}$
15	$.18233\times10^{-3}$	$.15476\times10^{-4}$	$.10840\times10^{-3}$	$.86385\times10^{-4}$	$.15527\times10^{-4}$
16	$.11824\times10^{-4}$	$.10164\times10^{-4}$	$.29437\times10^{-4}$	$.53998\times10^{-4}$	$.10109\times10^{-4}$
17	$.11995\times10^{-4}$	$.59171\times10^{-5}$	$.11695\times10^{-4}$	$.33243\times10^{-4}$	$.61673\times10^{-5}$
18	$.12213\times10^{-4}$	$.36245\times10^{-5}$	$.55384\times10^{-5}$	$.13669\times10^{-4}$	$.37382\times10^{-5}$
19	$.14193\times10^{-4}$	$.14071\times10^{-5}$	$.92956\times10^{-5}$	$.72502\times10^{-5}$	$.15625\times10^{-5}$
20	$.32546\times10^{-5}$	$.38308\times10^{-6}$	$.12126\times10^{-5}$	$.32347\times10^{-5}$	$.39579\times10^{-6}$

Table 5.9. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass Butterworth filter with R_s=10 (Case ix). Note that all algorithms with infinite precision in the number of bits are identical.

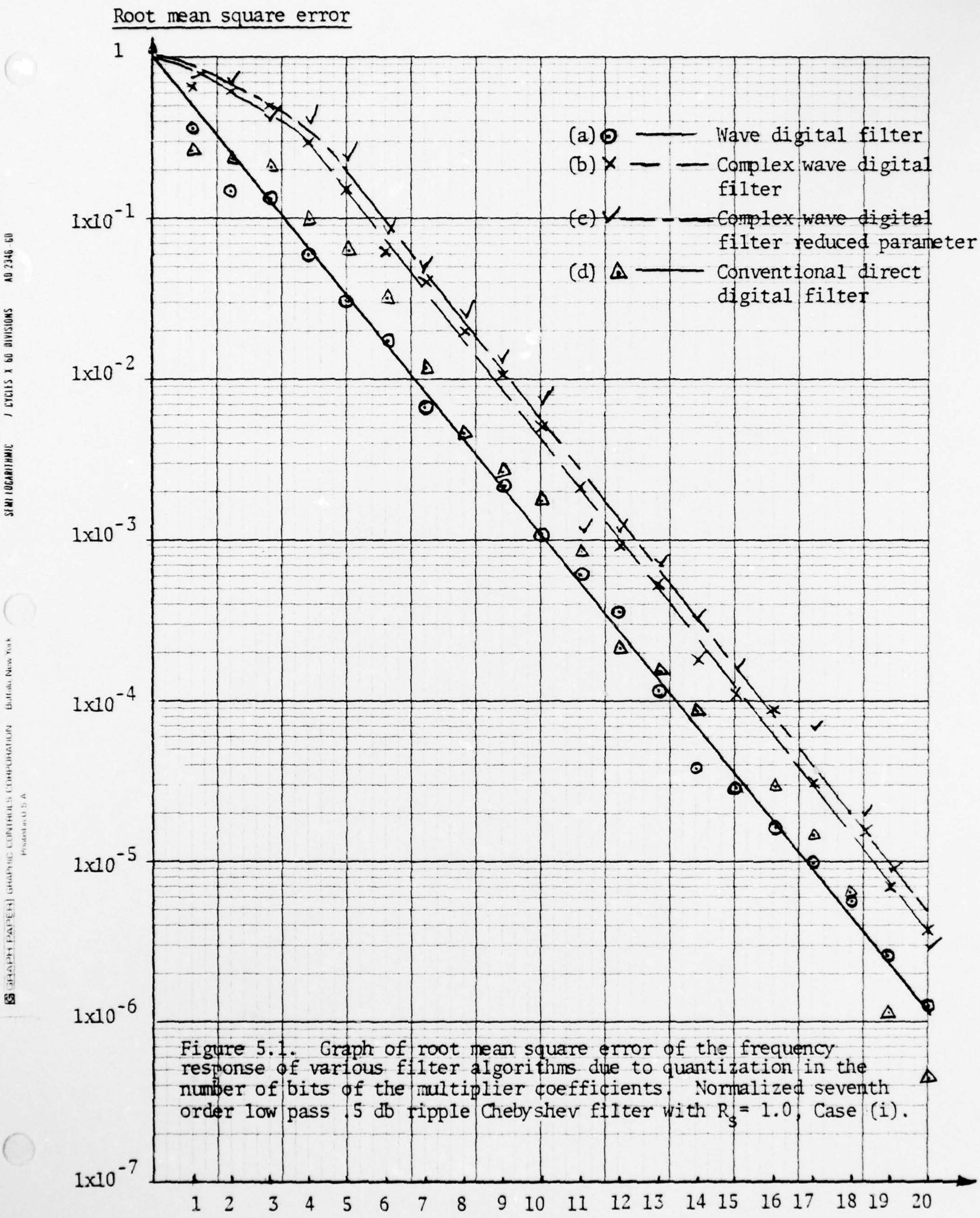


Figure 5.1. Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the multiplier coefficients. Normalized seventh order low pass .5 db ripple Chebyshev filter with $R_s = 1.0$, Case (i).

AD-A073 962 NAVAL POSTGRADUATE SCHOOL MONTEREY CA F/G 9/2

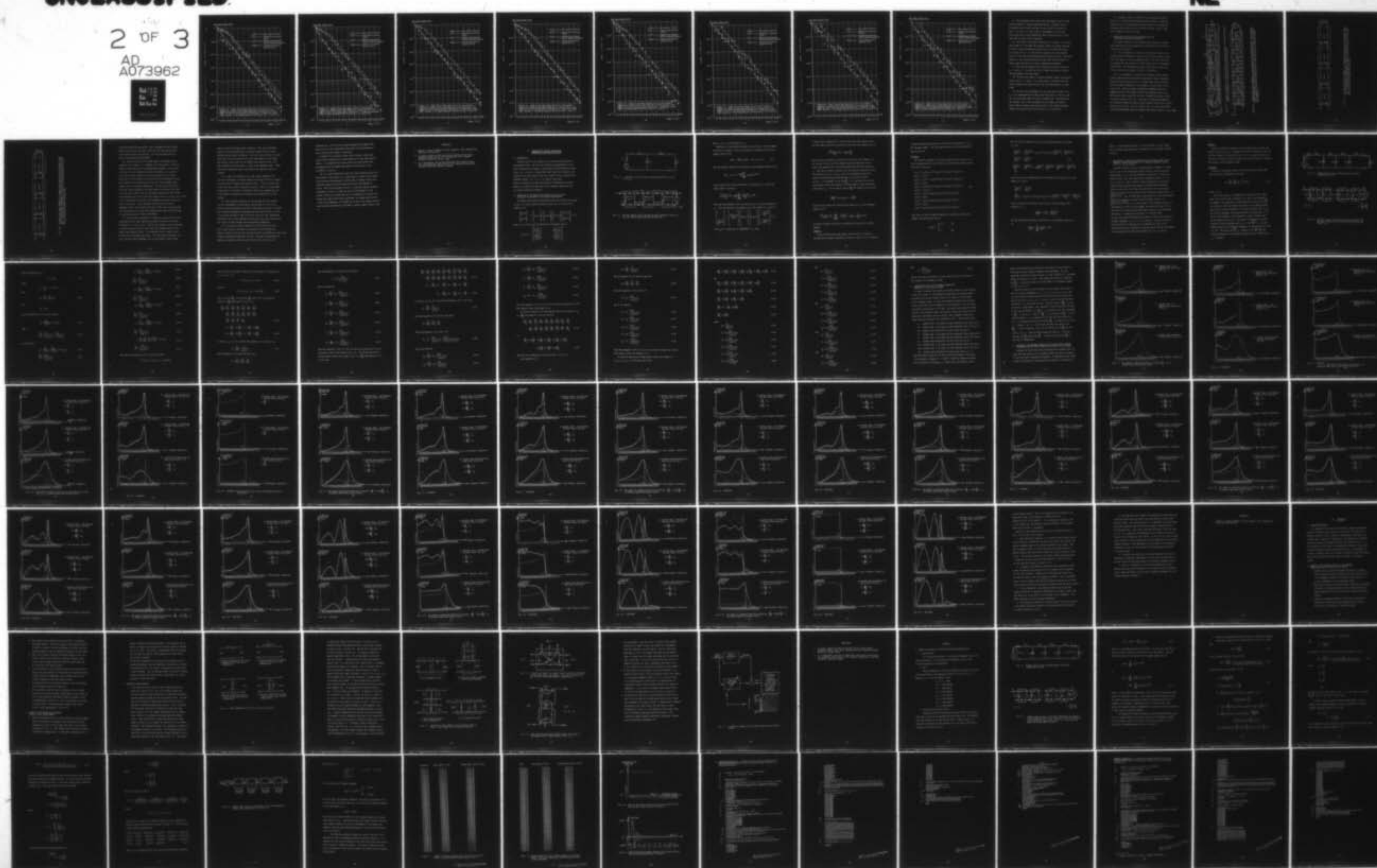
AN IN-DEPTH STUDY OF THE SENSITIVITY OF WAVE DIGITAL FILTERS,(U)

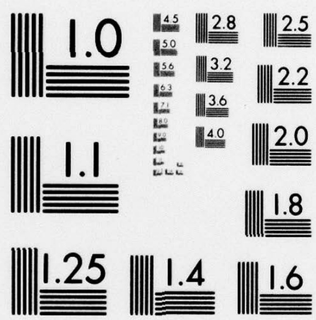
JUN 79 M SANAIE-FARD

UNCLASSIFIED NL

2 OF 3

AD
A073962





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

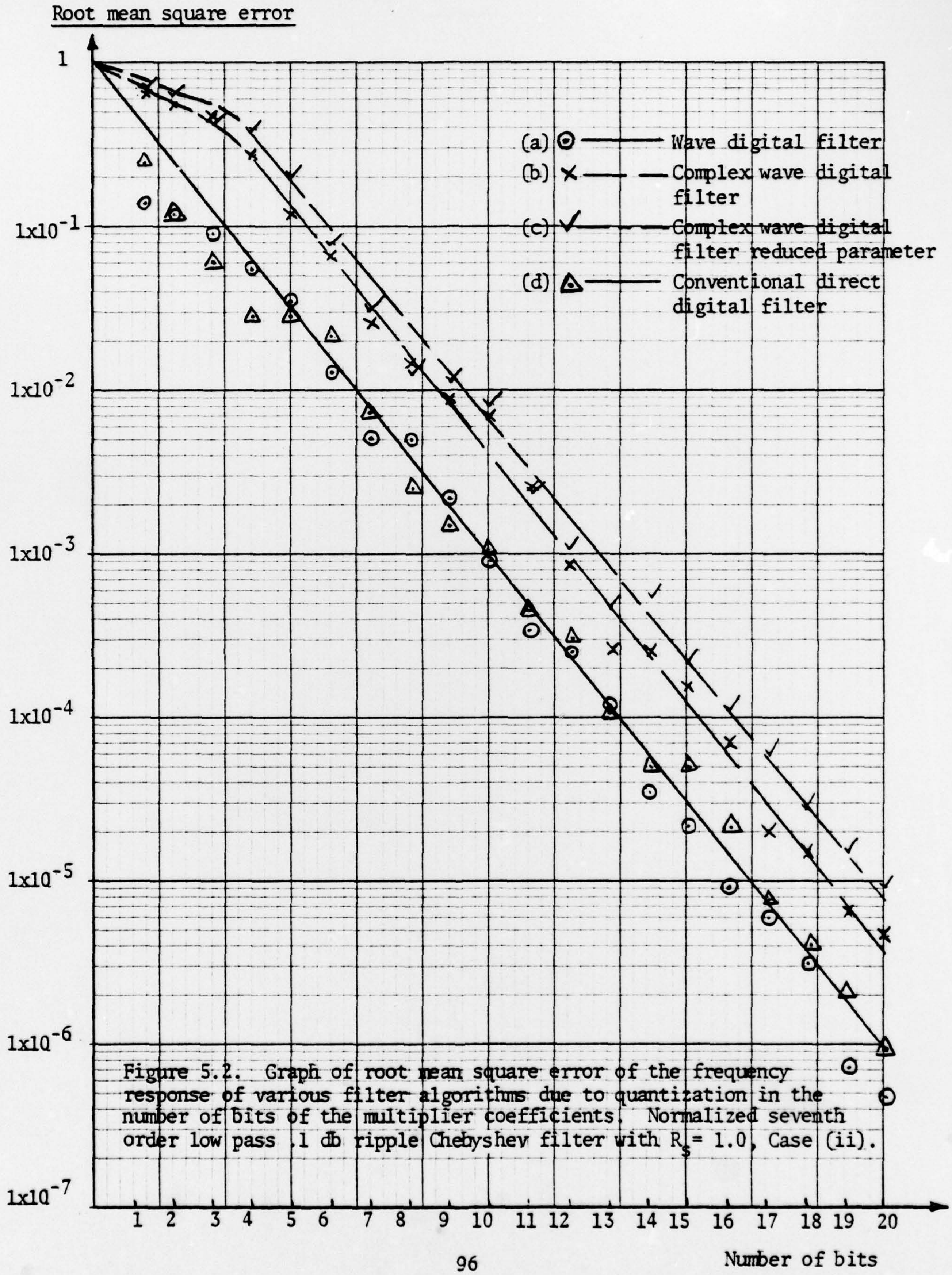


Figure 5.2. Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the multiplier coefficients. Normalized seventh order low pass .1 db ripple Chebyshev filter with $R_s = 1.0$, Case (ii).

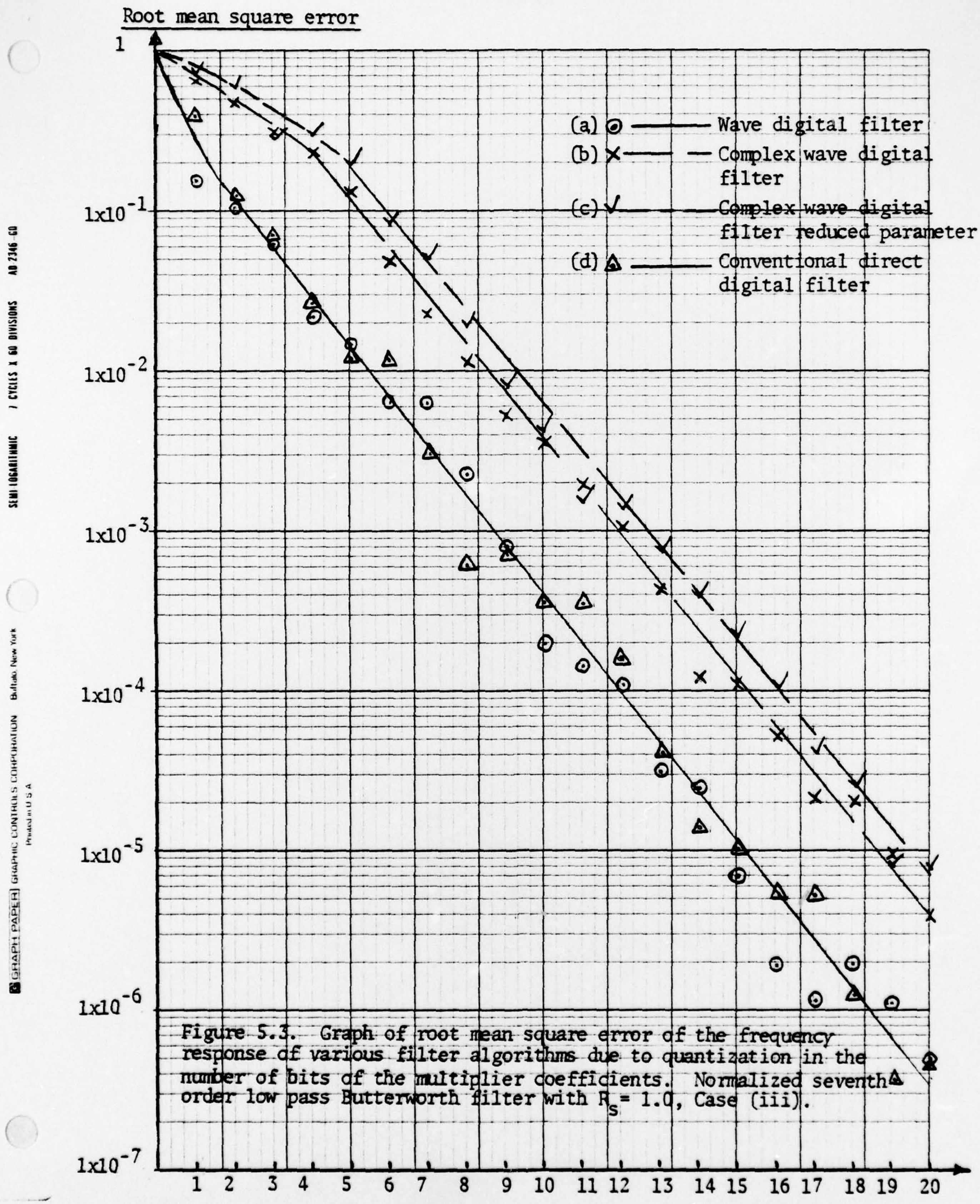


Figure 5.3. Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the multiplier coefficients. Normalized seventh order low pass Butterworth filter with $R_s = 1.0$, Case (iii).

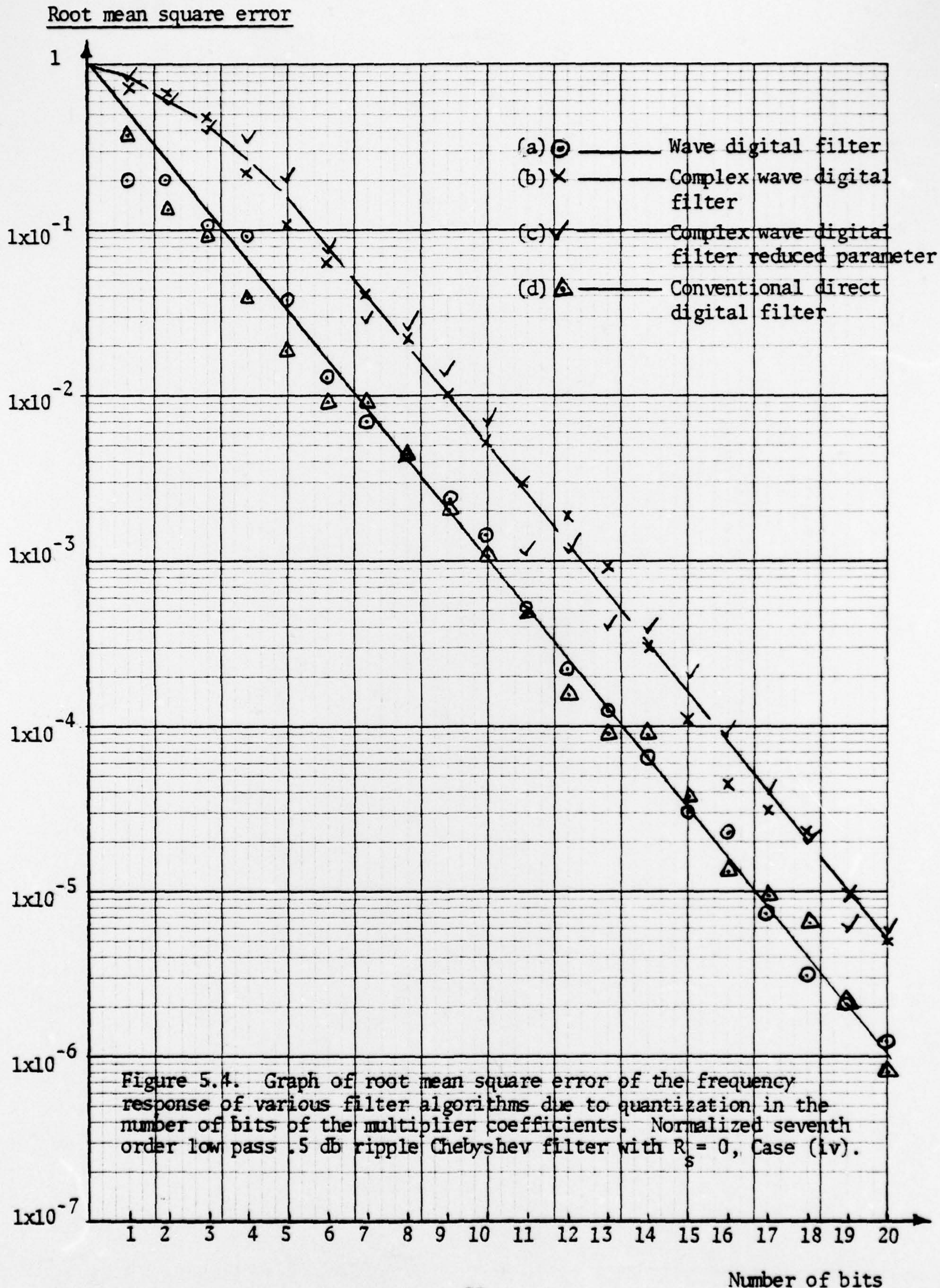


Figure 5.4. Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the multiplier coefficients. Normalized seventh order low pass .5 db ripple Chebyshev filter with $R_s = 0$, Case (iv).

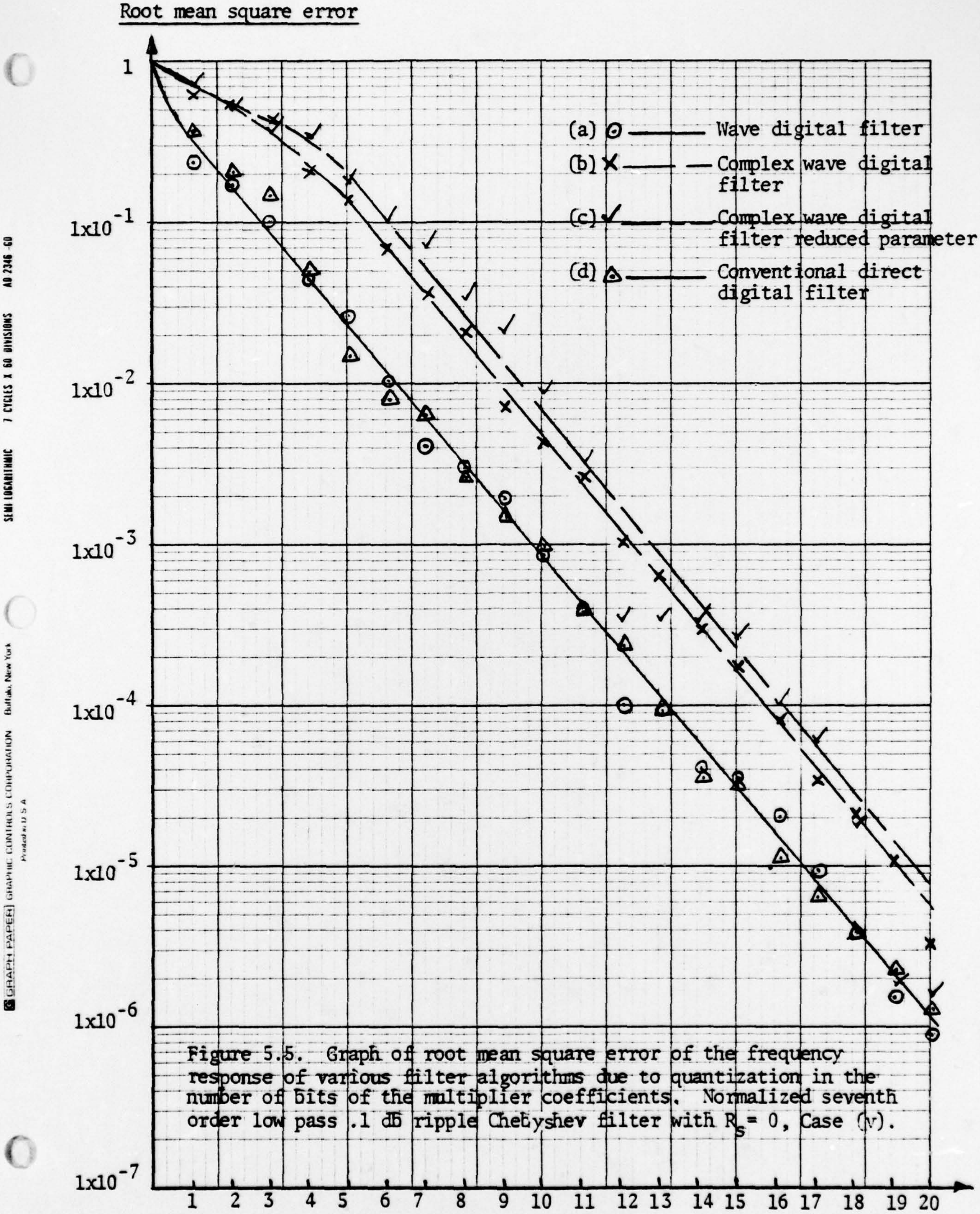


Figure 5.5. Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the multiplier coefficients. Normalized seventh order low pass .1 dB ripple Chebyshev filter with $R_S = 0$, Case (v).

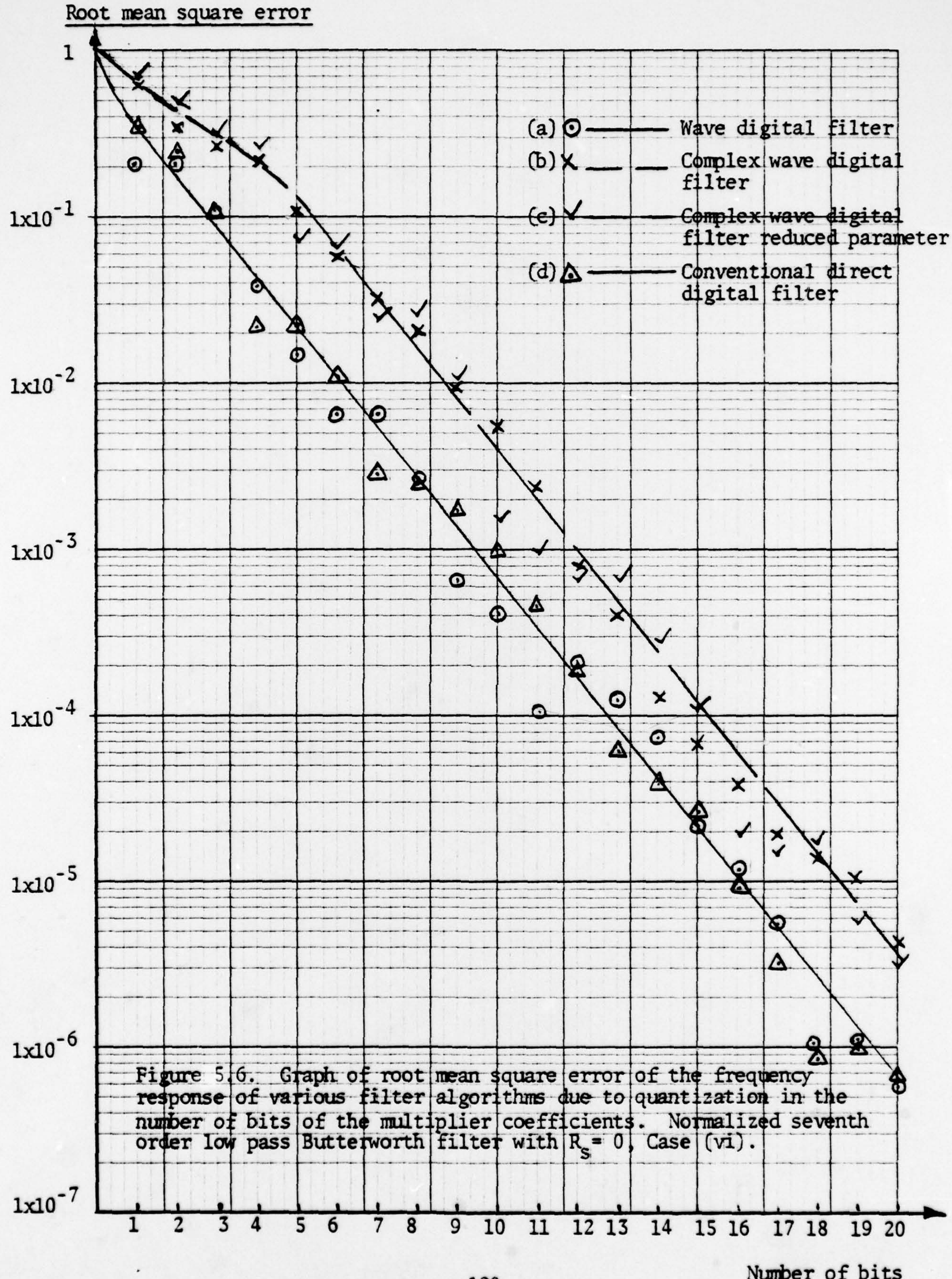


Figure 5.6. Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the multiplier coefficients. Normalized seventh order low pass Butterworth filter with $R_s = 0$, Case (vi).

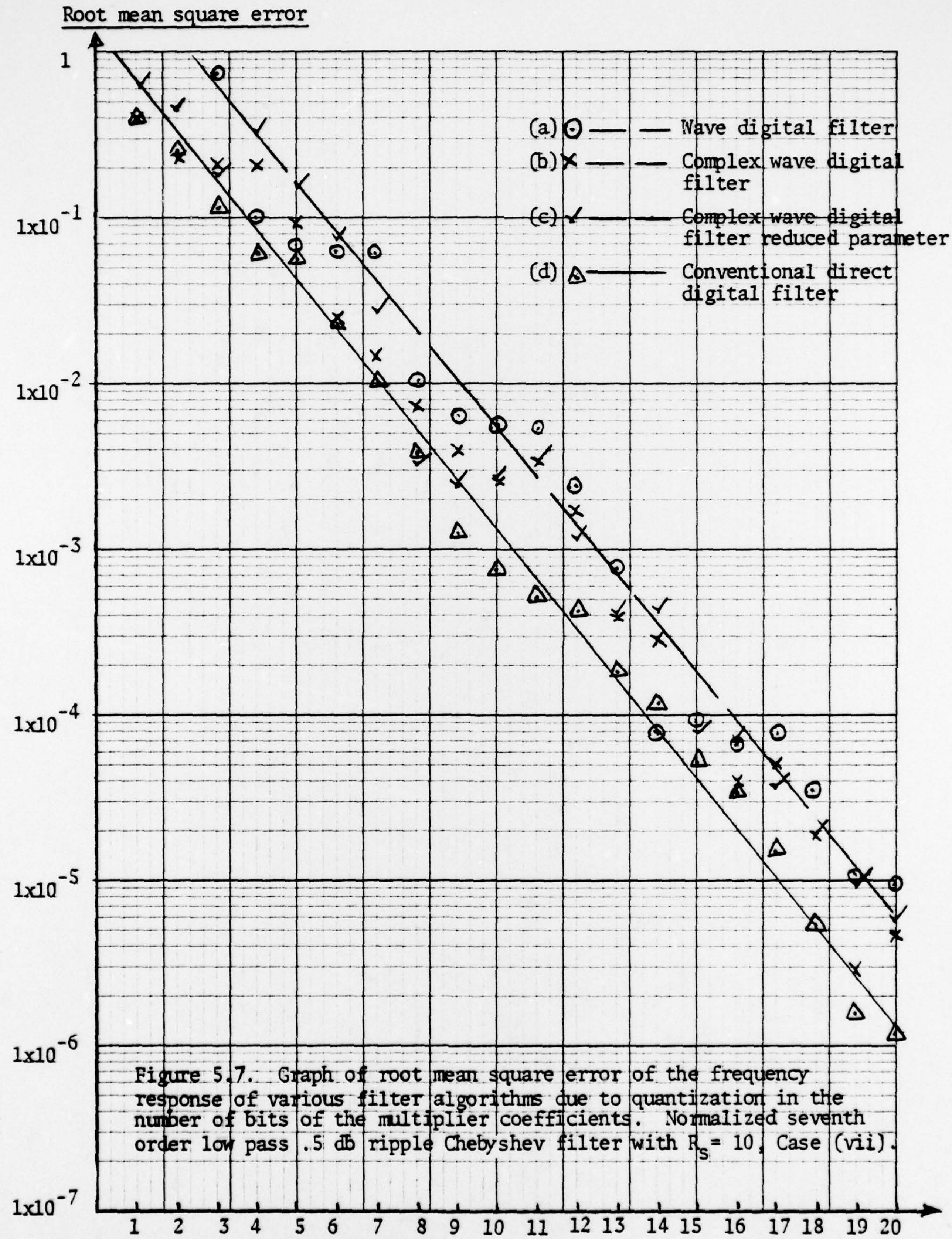


Figure 5.7. Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the multiplier coefficients. Normalized seventh order low pass .5 db ripple Chebyshev filter with $R_s = 10$, Case (vii).

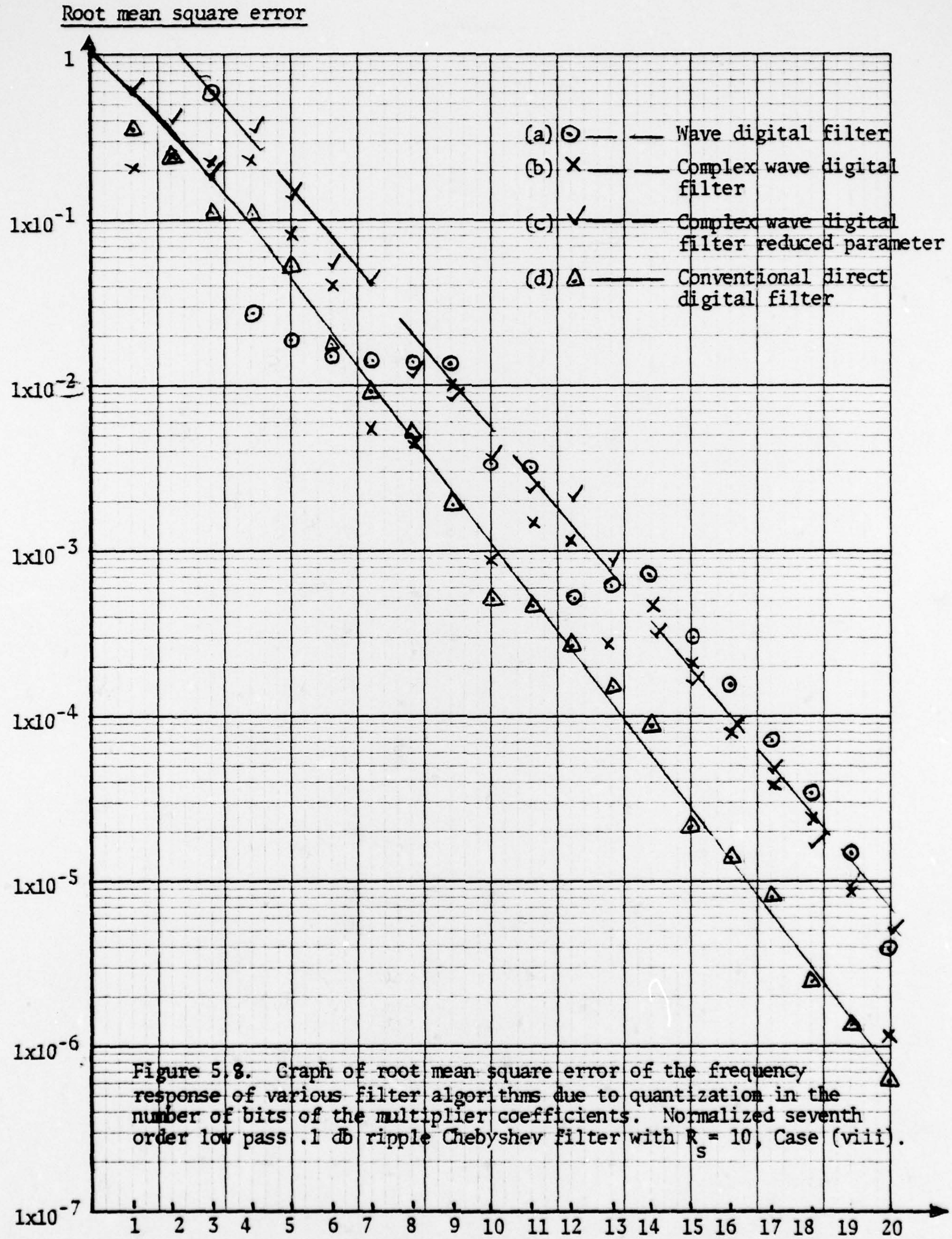


Figure 5.8. Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the multiplier coefficients. Normalized seventh order low pass .1 db ripple Chebyshev filter with $R_s = 10$, Case (viii).

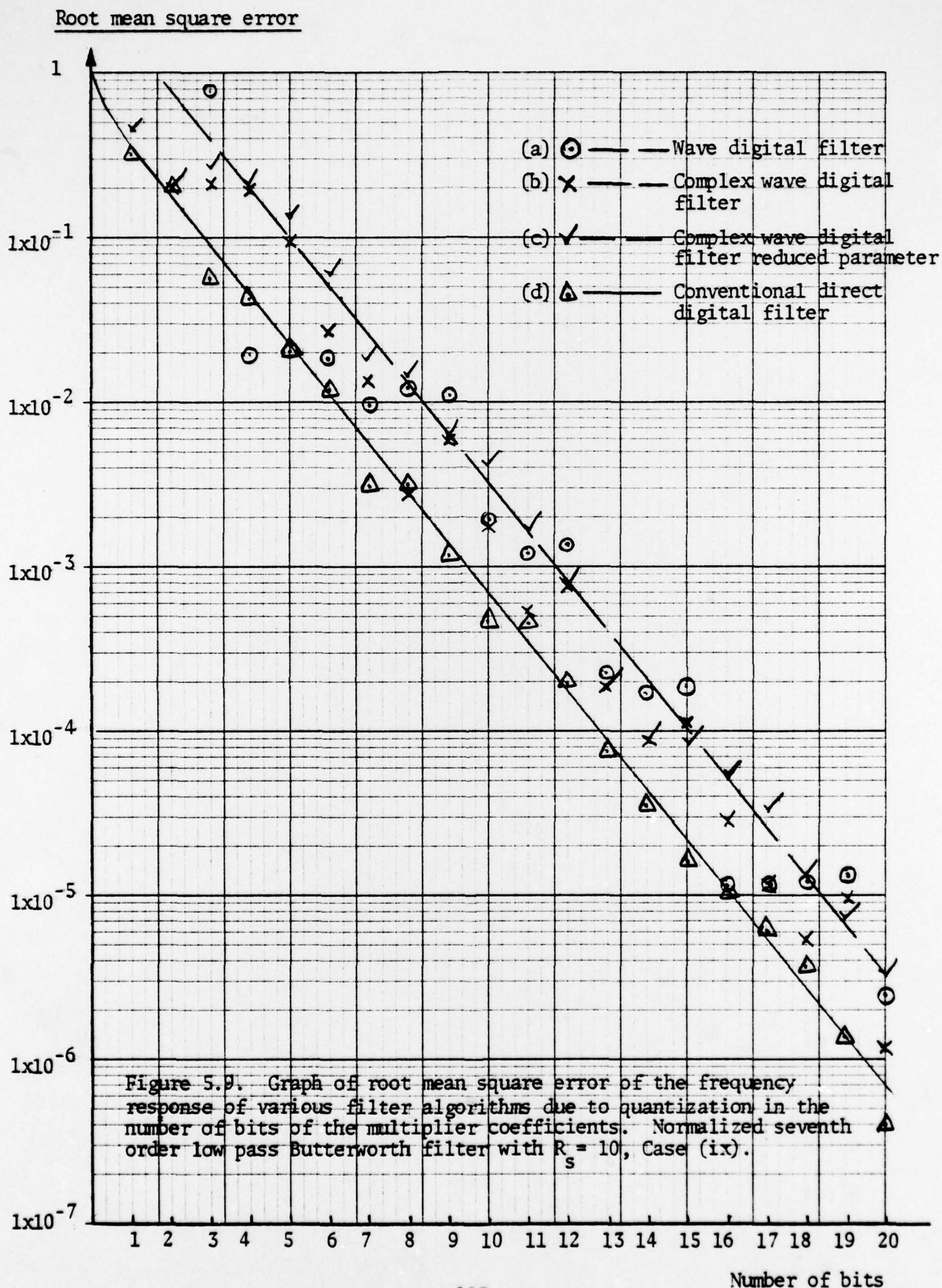


Figure 5.9. Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the multiplier coefficients. Normalized seventh order low pass Butterworth filter with $R_s = 10$, Case (ix).

(1) The logarithm of rms error versus the number of bits for the direct conventional digital algorithms follows a straight line for all seventh order filters and they all pass close to the point of error = 1 at bits = 0 with a slope of approximately 3 db per bit. Also the rms error of the conventional direct digital filter is never worse than the other filter algorithms.

(2) For filters with source resistance $R_s = 1$ and $R_s = 0$, the rms error graphs for the simple wave digital filters are exactly the same as that of direct conventional digital filter, with the same slope. For filters with $R_s = 10$, the rms error graph for the wave digital filter is worse than the conventional direct digital filter. This error behavior is very interesting and shows that wave digital filters do have a higher sensitivity to filter terminating resistances.

(3) The rms error graph for the complex wave digital filter for $R_s = 1$ and $R_s = 0$ is worse than that of simple wave digital filters, with approximately the same slope.

(4) The rms error graph for reduced parameter complex wave digital filters with $R_s = 1$ and $R_s = 0$ in some cases is slightly worse than that of complex wave digital filter, but with approximately the same slope.

(5) The rms error performance for all wave digital filters with $R_s = 10$ is approximately the same, after first four or five bits, and the rms error graph tends to follow a straight line. Departure from the straight line is more noticeable for the simple wave digital filter than the complex wave digital filter or the reduced parameter complex wave digital filter for the first six or seven bits.

(6) Although, ideally we expect the two conventional digital filters (i.e. the direct and cascaded digital filters) to have the same rms error, the factorization process used for the denominator introduces some slight error in the pole locations, which in some cases changes the rms error value.

D. COMPARISON OF RESULTS WITH EACH OTHER AND WITH THOSE PUBLISHED IN THE LITERATURE

The results obtained in this chapter and in Section C indicate the following points when compared with each other and with results published in literature:

(1) For all purposes the digital filter derived directly from analogue doubly terminated LC structure is the least sensitive structure with respect to filter component values. This fact should not be confused with the sensitivity of the digital filter to polynomial coefficients. The foregoing also apply to band pass or band stop filters derived from analogue low pass LC realization.

(2) It is worthwhile to consider the structure of wave digital filters compared with the structure of analogue LC doubly terminated filters for cases under study (i.e. seventh order low-pass filters). The total number resonances in the analogue LC filter shown in Fig. 5.10,is ten. These are evenly distributed resonances and can be compared with the total number of delay free feedback paths in the wave digital filter, which for the case of simple wave digital filter is eight as shown in Fig. 5.11. Note that the delay free feedbacks are localized towards the source for wave digital filters with no delay free path in port two. Also, as shown in Fig. 5.12, the total number of delay free feedback paths for complex wave digital filters is five, again

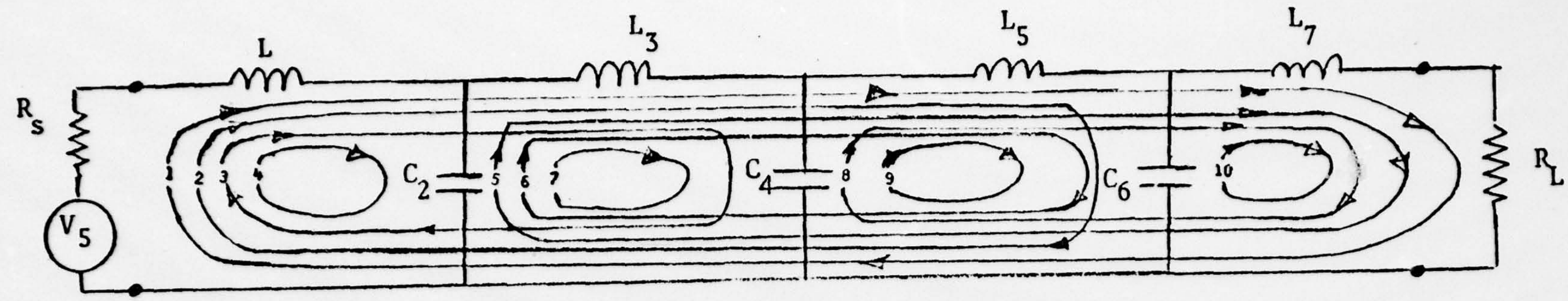


Fig. 5.10. Internal resonances in a 7th order doubly terminated LC filter. Note that the total number of resonances is ten and they are evenly distributed.

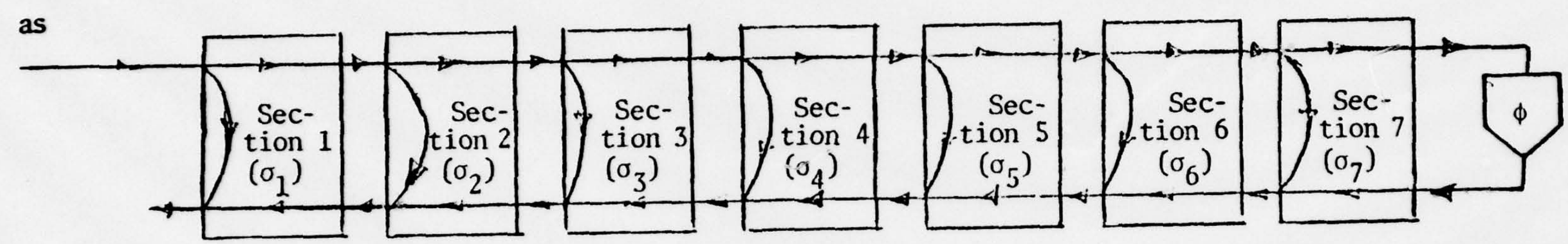


Fig. 5.11. Internal delay free feedback in a wave digital filter patterned after the 7th order doubly terminated LC structure (with no delay free path on port two). Note that the total number of delay free feedback paths is eight and they are localized towards the source end.

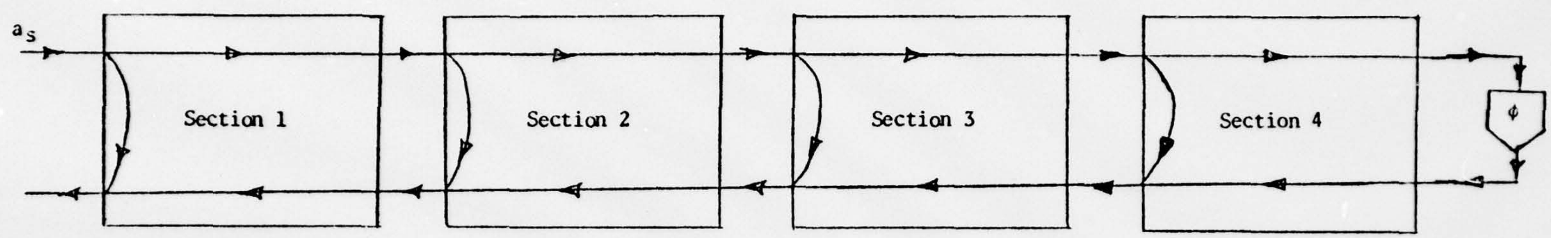


Fig. 5.12. Internal delay free feedback in a complex wave digital filter patterned after the 7th order doubly terminated LC structure (with no delay free path in port two). Note that the total number of delay free feedback paths are five and they are localized towards the source end.

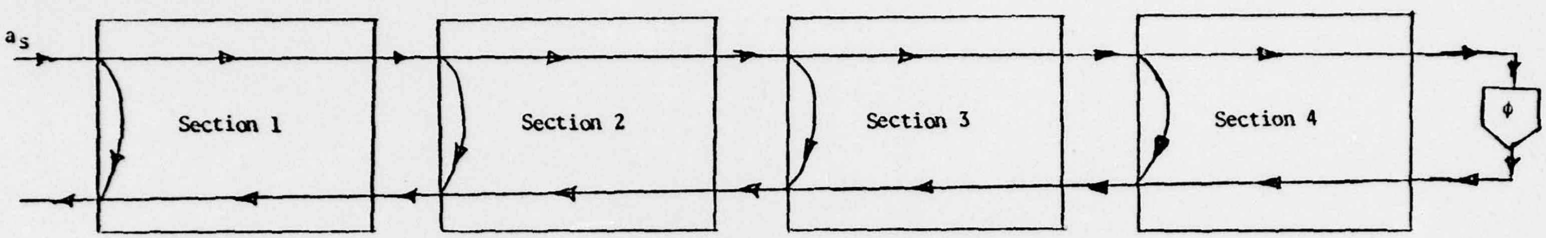


Fig. 5.12. Internal delay free feedback in a complex wave digital filter patterned after the 7th order doubly terminated LC structure (with no delay free path in port two). Note that the total number of delay free feedback paths are five and they are localized towards the source end.

localized towards the source end. Had we designed the wave digital filter with no delay free path in port one this localization would have occurred towards the load end. Now with the above points in mind we can establish the following:

(a) From the results of the large number of different filters studied in this chapter, it is apparent that wave digital filters do exhibit a high sensitivity to terminating resistances. For the case of filters with low terminating source resistances (i.e. R_s=0 and R_s=1) the rms error behavior of the wave digital filter is almost identical to direct digital filter derived from the analogue LC structure, because of the localization of the delay free feedback paths towards the low resistance termination. For the case of the filters with high source resistance, the localization of the delay free feedback path occurs towards the high resistance termination; thus the rms error, compared with that of the direct digital filter, is worse; this is due to the localization of the delay free feedback in the high sensitivity port of the filter. Had we designed the high resistive source impedance filters with no delay free path in port one, (thus directing the localization of the delay free feedback paths towards the load end) we would have obtained a better performance.

(b) The rms error performance of simple wave digital filters were considerably better than complex wave digital filters for low source resistances (i.e. R_s=0, R_s=1). Thus rms error behavior can be explained as follows because there are fewer delay free feedback paths for the complex wave digital filters. This feedback occurs in fewer sections and the error performance gets worse. For high source resistance, i.e. R_s=10, the rms error performance for all wave digital filters after fifth or sixth bit becomes almost identical. This can be explained because of the high sensitivity of the wave digital filters to the terminating high source resistance. Thus the localized delay free feedbacks near the source increases the error performance for the simple wave digital filter while for the complex wave digital filter, with fewer feedbacks, the sensitivity to high source impedance drops. Therefore the two algorithms tend to give nearly the same rms error performance.

(c) In some cases especially for lower source impedances the rms error performance of reduced complex wave digital filter is slightly worse than that of complex wave digital filter. This is probably due to the fact that, when the truncation process is done on a large number of multipliers, the effect of truncation tends to be compensating, but if the number of multipliers are reduced, this equalization is less evident.

(3) From the above discussion we can conclude that the ultimate rms error performance of wave digital filters tends to that of the directly digitized analogue resistively terminated LC filters. To achieve this ultimate performance in designing the wave digital filter, we have to use simple sections as much as possible, and also since terminating resistances are the most important we must direct the delay free port of wave digital filters towards the low resistance termination.

(4) If we wanted to make a band pass or band stop filter, which has LC tank circuits, the complex wave digital filter derived from these LC sections would not give the same rms error performance as that of the original conventional digitized LC filter. Moreover the algorithms for complex wave digital filters derived from LC tank circuits in the

literature [2], [3] are of the reduced parameter type complex wave digital filter, which furthermore have poorer error performance as can be seen from the graphs in some cases.

In Chapter VII we discuss briefly how to design a simple wave digital filter algorithm for a given band pass or band stop specification. It is expected that these filters will have better error performance than the complex wave digital filters given in the literature [2] and [3].

(5) it is also interesting to note that, after familiarization with wave digital filter theory, it is much easier to design a wave digital filter derived from analogue LC filters than to design a direct conventional digital filter both derived from the same analogue LC filter. Thus the wave digital filter is a practical design technique. Furthermore in order to reduce the rms error performance of wave digital filters we can even make use of a combination of complex wave digital and simple wave digital algorithms. For example when we have a high source impedance, we can make use of one or two complex sections near the high resistance end, followed by several simple sections later on.

References

1. Anatol I. Zverev, Handbook of Filter Synthesis, 1967, published by John Wiley and Sons, Inc.

2. S. Erfani, Design of Fixed and Variable Digital Filters Using Generalized Delay Units, Ph.D. Dissertation, 1976, Southern Methodist University, Dallas, Texas.

3. K. S. Thyagarajan, One and Two-dimensional Wave Digital Filters with Low Coefficient Sensitivities, 1977, Doctoral Dissertation, Concordia University, Montreal, Canada.

VI. DERIVATION OF PARTIAL SENSITIVITY FUNCTIONS OF WAVE DIGITAL FILTERS

A. INTRODUCTION

The main intent of this chapter is to derive the sensitivity of wave digital filter with the aid of partial differentiation of the filter algorithm with respect to wave digital filter multiplier coefficients, i.e. σ's and ϕ. Having found these sensitivity functions the sensitivity of wave digital filter with respect to the original filter component values, i.e. L's, C's, R_S, and R_L were found.
Finally the behavior of these two sensitivity functions are investigated and compared with each other in the frequency domain and with the results obtained in Chapter V.

B. DERIVATION OF THE SENSITIVITY FUNCTION OF WAVE DIGITAL FILTER DUE TO VARIATION IN MULTIPLIER COEFFICIENTS

The general wave digital algorithm for the third order filter given in Fig. 6.2, with no delay free path in port two, is of the form of equation (6.1)

$$\begin{bmatrix} a_{11}(z) \\ b_{11}(z) \end{bmatrix} = \begin{bmatrix} f_1(\sigma_1, z) \end{bmatrix} \begin{bmatrix} f_2(\sigma_2, z) \end{bmatrix} \begin{bmatrix} f_3(\sigma_3, z) \end{bmatrix} \begin{bmatrix} b_{32}(z) \\ \phi b_{32}(z) \end{bmatrix} \tag{6.1}$$

where f_1, f_2, and f_3 are 2x2 matrices of the general form of

$$f_1(\sigma_1, z) = \begin{bmatrix} \dfrac{\alpha_1+\beta_1 z^{-1}}{1+\gamma_1 z^{-1}} & \dfrac{\alpha_2+\beta_2 z^{-1}}{1+\gamma_2 z^{-1}} \\ \dfrac{\alpha_3+\beta_3 z^{-1}}{1+\gamma_3 z^{-1}} & \dfrac{\alpha_4+\beta_4 z^{-1}}{1+\gamma_4 z^{-1}} \end{bmatrix}$$

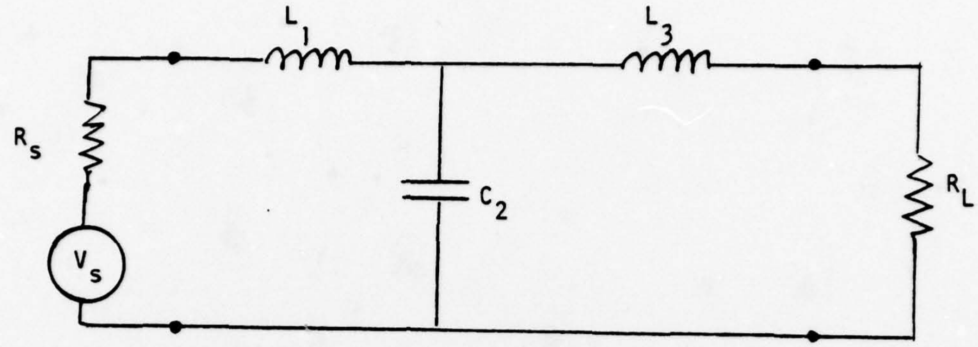


Fig. 6.1. A general third order doubly terminated analogue low pass LC filter.

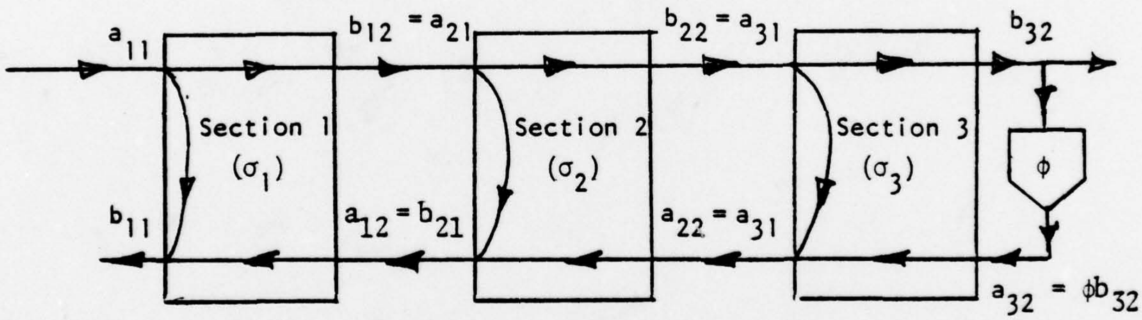


Fig. 6.2. The wave digital filter derived from the analogue LC filter of Fig. 6.1 with no delay free path on port two.

where α_i, β_i, γ_i are functions of σ_1.

From Fig. 6.2 the transfer function of the filter in the time domain as depicted in Chapter IV and also explained in Appendix 1A with unity impulse input will be

$$h(nT) = \left(\frac{1+\phi}{2}\right) b_{32}(nT) = h(\sigma_1, \sigma_2, ---\phi) \qquad (6.2)$$

Thus the transfer function of the filter in the frequency domain will be

$$H(\sigma_1, \sigma_2, ---\phi) = \frac{1+\phi}{2} \sum_{n=0}^{N} b_{32}(nT) e^{-j\omega nT}$$

and the sensitivity of the filter due to variations in σ_1 in the frequency domain is given by

$$\frac{\partial H(\sigma_1, \sigma_2 ---\phi)}{\partial \sigma_1} = \frac{1+\phi}{2} \sum_{n=0}^{N} \frac{\partial b_{32}(nT)}{\partial \sigma_1} e^{-j\omega nT}$$

Note that from equation (6.1), the matrices f_2, f_3 are not functions of σ_1. Thus the derivative of equation (6.1) with respect to σ_1 will be

$$\begin{bmatrix} 0 \\ \\ \frac{\partial b_{11}(z)}{\partial \sigma_1} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1(\sigma_1, z)}{\partial \sigma_1} \end{bmatrix} \begin{bmatrix} f_2(\sigma_2, z) \end{bmatrix} \begin{bmatrix} f_3(\sigma_3, z) \end{bmatrix} \begin{bmatrix} \frac{\partial b_{32}(z)}{\partial \sigma_1} \\ \\ \phi \cdot \frac{\partial b_{32}(z)}{\partial \sigma_1} \end{bmatrix} \qquad (6.3)$$

Note $a_{11}(z)$ is input and is independent of σ_1 also.

A closer look at equation (6.2) reveals that with unity impulse input, the time domain sensitivity function of the filter with respect to σ_1 is

$$\frac{\partial h(\sigma_1,\sigma_2,---\phi)}{\partial\sigma_1} = \left(\frac{1+\phi}{2}\right) \frac{\partial b_{32}}{\partial\sigma_1}$$

Thus to find the sensitivity function of the filter with respect to σ_1 all we have to do is write the complete iterative equations of the filter from equation (6.1) and then differentiate them with respect to σ_1. Note that in doing so actually all the iterative equations derived from matrices $f_2(\sigma_2,z)$, $f_3(\sigma_3,z)$ will not be effected and the only iterative equations being differentiated will be that of matrix $f_1(\sigma_1,z)$. So in general we can find $\frac{\partial H}{\partial\sigma_n}$, i.e. the sensitivity of the wave digital filter with respect to all wave digital filter multiplier coefficients σ_n. For the special case of $\frac{\partial H}{\partial\phi}$ from equation (6.2) we have

$$\frac{\partial h(nT)}{\partial\phi} = \frac{1}{2}\left(b_{32}(nT) + \phi\,\frac{\partial b_{32}}{\partial\phi}\right)$$

and the filter sensitivity function with respect to ϕ in the frequency domain will be

$$\frac{\partial H(\sigma_1,\sigma_2---\phi)}{\partial\phi} = \sum_{n=0}^{N}\left[\frac{b_{32}(nT)}{2} + \frac{(1+\phi)}{2}\cdot\frac{\partial b_{32}}{\partial\phi}\right]e^{-j\omega nT}$$

The above procedure can best be illustrated with the aid of a simple example.

Example 1

Given the third order wave digital filter of Fig. 6.2 which is derived from the doubly terminated LC filter of Fig. 6.1; it is required to find the sensitivity function of the filter with respect to σ_1 in the frequency domain. The wave digital filter is with no delay free path in port two.

Procedure

The iterative equations of the wave digital filter of Fig. 6.2 in the time domain with the aid of Table 4.2b and with all initial conditions set to zero are:

$$
\begin{aligned}
b_{12}(n) &= a_{11}(n)+a_{11}(n-1)-a_{12}(n-1)+\sigma_1(a_{12}(n-1)-b_{12}(n-1)) \\
a_{21}(n) &= b_{12}(n) \\
b_{22}(n) &= a_{22}(n-1)+\sigma_2(a_{21}(n)+a_{21}(n-1)-a_{22}(n-1)-b_{22}(n-1)) \\
a_{31}(n) &= b_{22}(n) \\
b_{32}(n) &= a_{31}(n)+a_{31}(n-1)-a_{32}(n-1)+\sigma_3(a_{32}(n-1)-b_{32}(n-1)) \\
a_{32}(n) &= b_{32}(n)\cdot\phi \\
b_{31}(n) &= a_{31}(n)+\sigma_3(a_{32}(n)-a_{31}(n)+a_{32}(n-1)-b_{31}(n-1)) \\
a_{22}(n) &= b_{31}(n) \\
b_{21}(n) &= a_{22}(n)-a_{21}(n)+a_{22}(n-1)+\sigma_2(a_{21}(n)-b_{21}(n-1)) \\
a_{12}(n) &= b_{21}(n) \\
b_{11}(n) &= a_{11}(n)+\sigma_1(a_{12}(n)-a_{11}(n)+a_{12}(n-1)-b_{11}(n-1))
\end{aligned}
\tag{6.4}
$$

Note that to find the transfer function of the filter in the time domain, the input $a_{11}(n)$ will be

$$
a_{11}(n) = \begin{cases} 1.0 & n=0 \\ 0 & n\neq 0 \end{cases}
$$

and the iterative equations after partial differentiation with respect to σ_1 are

$$\frac{\partial b_{12}(n)}{\partial \sigma_1} = -\frac{\partial a_{12}(n-1)}{\partial \sigma_1} + a_{12}(n-1) - b_{12}(n-1) + \sigma_1\left(\frac{\partial a_{12}(n-1)}{\partial \sigma_1} - \frac{\partial b_{12}(n-1)}{\partial \sigma_1}\right)$$

$$\frac{\partial a_{21}(n)}{\partial \sigma_1} = \frac{\partial b_{12}(n)}{\partial \sigma_1} \tag{6.5}$$

$$\frac{\partial b_{22}(n)}{\partial \sigma_1} = \frac{\partial a_{22}(n-1)}{\partial \sigma_1} + \sigma_2\left(\frac{\partial a_{21}(n)}{\partial \sigma_1} + \frac{\partial a_{21}(n-1)}{\partial \sigma_1} - \frac{\partial a_{22}(n-1)}{\partial \sigma_1} - \frac{\partial b_{22}(n-1)}{\partial \sigma_1}\right)$$

etc.
⋮

and in this way, we differentiate all the equations in equation (6.4) with respect to σ_1 up to

$$\frac{\partial a_{12}(n)}{\partial \sigma_1} = \frac{\partial b_{21}(n)}{\partial \sigma_1}$$

$$\frac{\partial b_{11}}{\partial \sigma_1} = a_{12}(n) - a_{11}(n) + a_{12}(n-1) - b_{11}(n-1) + \sigma_1\left(\frac{\partial a_{12}(n)}{\partial \sigma_1} + \frac{\partial a_{12}(n-1)}{\partial \sigma_1} - \frac{\partial b_{11}(n-1)}{\partial \sigma_1}\right)$$

Thus the sensitivity function of the filter in the time domain from equation 6.2 is

$$\frac{\partial h(nT)}{\partial \sigma_1} = \left(\frac{1+\phi}{2}\right) \cdot \frac{\partial b_{32}(nT)}{\partial \sigma_1}$$

and the sensitivity function of the filter in the frequency domain will be

$$\frac{\partial H(\omega)}{\partial \sigma_1} = \sum_{n=0}^{N} \frac{\partial h(nT)}{\partial \sigma_1} e^{-j\omega nT}$$

where T is the sampling interval, ω is the frequency of input signal, and N is chosen large enough for transients due to the impulse response to decay.

C. DERIVATION OF SENSITIVITY FUNCTION OF WAVE DIGITAL FILTER IN THE FREQUENCY DOMAIN WITH RESPECT TO ITS ORIGINAL COMPONENT VALUES, I.E. L's, C's, R_S, AND R_L

As noted in Table 4.1 and 4.2 and mentioned in Appendix 1-A, there exists a one to one relationship between the wave digital multiplier coefficients, i.e. σ's and the reflection coefficient ϕ of the wave digital filter, with the original filter component values, i.e. L's, C's, R_S, and R_L. Thus in order to find the sensitivity of the wave digital filter algorithm with respect to original filter component values, the best way is to find the sensitivity of the wave digital filter with respect to wave digital filter multiplier coefficients, i.e. $\frac{\partial H(\sigma_1,\sigma_2---\phi)}{\partial\sigma_1}$, etc. and from these, the sensitivity of the wave digital filter with respect to original filter component values, i.e. $\frac{\partial H(L_1,C_2,L_3---,R_S,R_L)}{\partial L_1}$, etc. can be found. This procedure can best be illustrated with the aid of an example, the results of which will be used later on to find the semi-logarithmic sensitivity (or percentage sensitivity; or normalized sensitivity) of a seventh order low pass resistively terminated wave digital LC filter with respect to original components (i.e. L's, C's, R_S, and R_L). Later on these normalized sensitivity functions will be compared with that of the normalized sensitivity functions of the wave digital filter with respect to wave digital filter multiplier coefficients (i.e. σ's and ϕ).

Example 2

Given the sensitivity functions of the seventh order low pass wave digital filter of Fig. 6.4, designed with no delay free path in port two from the original LC structure of Fig. 6.3; it is required to find the sensitivity function of the wave digital filter with respect to original wave digital filter component values.

Procedure

Since the wave digital filter is of the type with no delay free path in port two, from Table 4.2 we have

$$\sigma_1 = \frac{R_1}{R_2} = \frac{R_1}{R_1 + L_1} = f(R_s, L_1) \qquad (6.6)$$

where $R_1 = R_s$.

Also at this stage it is important to note that L and C components of the filter used in equation (6.6) and in the remainder of the example are predivided or premultiplied by the factor $\tan \frac{\Omega_{cd}T}{2} / \omega_c$ in order to take into account the effect of the sampling period and also the critical frequency of the digital filter, where Ω_{cd} is the critical frequency of the digital filter in rad/sec and ω_c is the critical frequency of the analogue LC filter in rad/sec and T is the sampling time in sec. This factor is obtained by the multiplication of the prewarping factor, i.e. $2 \tan \frac{\Omega_{cd}T}{2} / T\, \omega_c$ by the sampling factor of the reactive components of the wave digital filter, i.e. T/2. The factor $\tan \frac{\Omega_{cd}T}{2} / \omega_c$ reduces to $\tan \frac{\Omega_{cd}T}{2}$ when the critical frequency of the original analogue filter is normalized at $\omega_c = 1$ rad/sec.

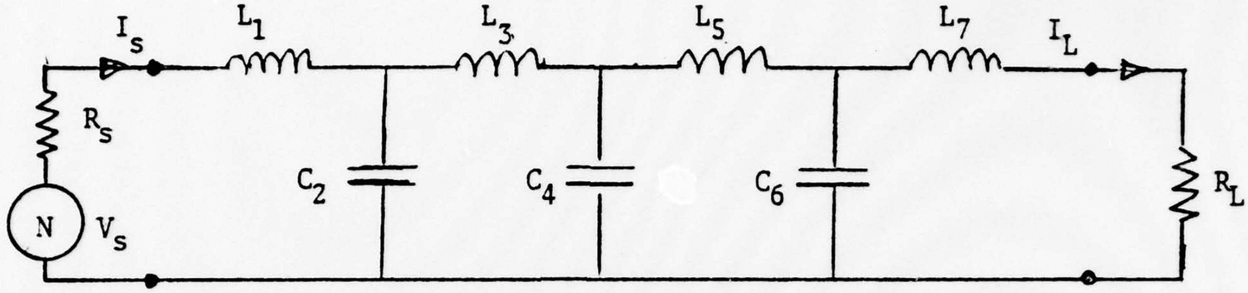


Fig. 6.3. Seventh order low pass analogue double resistively terminated LC filter.

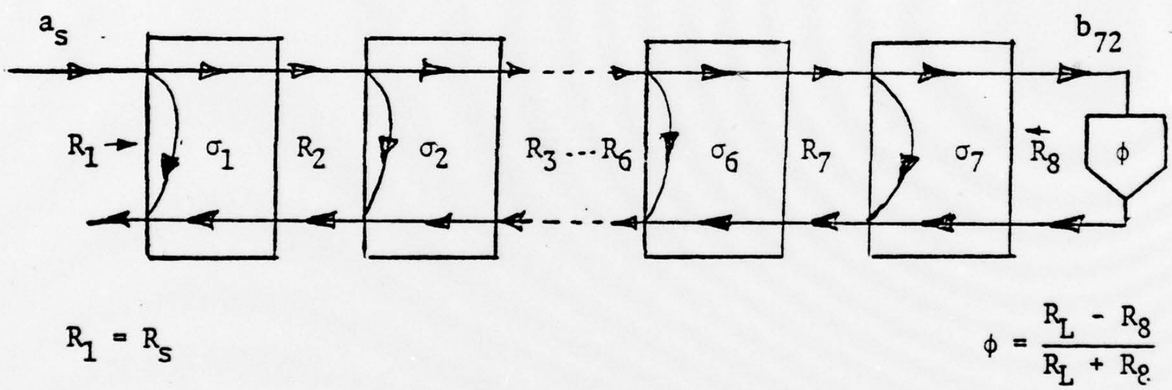


Fig. 6.4. The wave digital filter derived from the seventh order analogue LC filter of Fig. 6.3 with no delay free path on port two.

Thus from equation (6.6)

$$G_2 = \sigma_1 G_1 \tag{6.7}$$

where

$$G_i = \frac{1}{R_i}\ , \ i = 1,2,3\ldots$$

Also

$$b_2 = \frac{G_2}{G_3} = \frac{G_2}{G_2 + C_2} \tag{6.8}$$

and

$$R_3 = \sigma_2 R_2$$

Also from equation (6.7) and (6.8) we have

$$\sigma_2 = \frac{\sigma_1 G_1}{\sigma_1 G_1 + C_2} = f(\sigma_1, C_2) \tag{6.9}$$

thus

$$\frac{\partial\sigma_2}{\partial\sigma_1} = \frac{G_1 C_2}{(\sigma_1 G_1 + C_2)^2} = \frac{C_2 R_1}{(\sigma_1 + R_1 C_2)^2} \tag{6.10}$$

In the same way

$$\sigma_3 = \frac{R_3}{R_3 + L_3} = \frac{\sigma_2 R_2}{\sigma_2 R_2 + L_3} = f(\sigma_2, L_3) \tag{6.11}$$

$$\frac{\partial\sigma_3}{\partial\sigma_2} = \frac{R_2 L_3}{(\sigma_2 R_2 + L_3)^2} \tag{6.12}$$

$$\sigma_4 = \frac{G_4}{G_4+C_4} = \frac{\sigma_3 G_3}{\sigma_3 G_3+C_4} = f(\sigma_3, C_4) \tag{6.13}$$

$$\frac{\partial\sigma_4}{\partial\sigma_3} = \frac{G_3 C_4}{(\sigma_3 G_3+C_4)^2} \tag{6.14}$$

$$\sigma_5 = \frac{R_5}{R_4+L_5} = \frac{\sigma_4 R_4}{\sigma_4 R_4+L_5} = f(\sigma_4, L_5) \tag{6.15}$$

$$\frac{\partial\sigma_5}{\partial\sigma_4} = \frac{R_4 L_5}{(\sigma_4 R_4+L_5)^2} \tag{6.16}$$

$$\sigma_6 = \frac{G_6}{G_6+C_6} = \frac{\sigma_5 G_5}{\sigma_5 G_5+C_6} = f(\sigma_5, C_6) \tag{6.17}$$

$$\frac{\partial\sigma_6}{\partial\sigma_5} = \frac{G_5 C_6}{(\sigma_5 G_5+C_6)^2} \tag{6.18}$$

$$\sigma_7 = \frac{R_7}{R_7+L_7} = \frac{\sigma_6 R_6}{\sigma_6 R_6+L_7} = f(\sigma_6, L_7) \tag{6.19}$$

$$\frac{\partial\sigma_7}{\partial\sigma_6} = \frac{R_6 L_7}{(\sigma_6 R_6+L_7)^2} \tag{6.20}$$

$$\phi = \frac{R_L - R_8}{R_L + R_8} = \frac{R_L - G_7\sigma_7}{R_L + G_7\sigma_7} = f(\sigma_7, R_L) \tag{6.21}$$

$$\frac{\partial\phi}{\partial\sigma_7} = \frac{-2G_7 R_L}{(R_L+G_7\sigma_7)^2} \tag{6.22}$$

Note that from equation (6.21) it can be seen that

$$\phi = f(L_1,\ C_2,\ L_3,\ C_4, \text{----},\ L_7, R_s, R_L)$$

Note also that the transfer function H of the filter is a function of σ_1, σ_2, etc., i.e.

$$H = f(\sigma_1, \sigma_2, ---, \sigma_7, \phi) \tag{6.23}$$

and also

$$H = f(L_1, C_2, L_3, ---, L_7, R_s, R_L) \tag{6.24}$$

Thus if we have $\frac{\partial H}{\partial \sigma_1}$, we can derive $\frac{\partial H}{\partial R_s}$ from (6.23) by writing all partials of $\frac{\partial H}{\partial R_s}$ from equation (6.23), i.e.

$$\begin{aligned}\frac{\partial H}{\partial R_s} = {} & \frac{\partial H}{\partial \sigma_1} \cdot \frac{\partial \sigma_1}{\partial R_s} + \frac{\partial H}{\partial \sigma_2} \cdot \frac{\partial \sigma_2}{\partial R_s} + \frac{\partial H}{\partial \sigma_3} \cdot \frac{\partial \sigma_3}{\partial R_s} + \\ & \frac{\partial H}{\partial \sigma_4} \cdot \frac{\partial \sigma_4}{\partial R_s} + \frac{\partial H}{\partial \sigma_5} \cdot \frac{\partial \sigma_5}{\partial R_s} + \frac{\partial H}{\partial \sigma_6} \cdot \frac{\partial \sigma_6}{\partial R_s} + \\ & \frac{\partial H}{\partial \sigma_7} \cdot \frac{\partial \sigma_7}{\partial R_s} + \frac{\partial H}{\partial \phi} \cdot \frac{\partial \phi}{\partial R_s}\end{aligned} \tag{6.25}$$

$$\begin{aligned} = {} & A_1 \cdot \frac{\partial H}{\partial \sigma_1} + A_2 \frac{\partial H}{\partial \sigma_2} + A_3 \cdot \frac{\partial H}{\partial \sigma_3} + A_4 \cdot \frac{\partial H}{\partial \sigma_4} \\ & + A_5 \cdot \frac{\partial H}{\partial \sigma_5} + A_6 \cdot \frac{\partial H}{\partial \sigma_6} + A_7 \cdot \frac{\partial H}{\partial \sigma_7} + A_8 \cdot \frac{\partial H}{\partial \sigma_8}\end{aligned} \tag{6.26}$$

to find A_1, A_2, A_3, etc. we have from equation (6.6) with $R_1 = R_s$.

$$A_1 = \frac{\partial \sigma_1}{\partial R_s} = \frac{L_1}{(R_1 + L_1)^2} \tag{6.27}$$

and from equation (6.25) and the chain rule

$$A_2 = \frac{\partial \sigma_2}{\partial R_s} = \frac{\partial \sigma_2}{\partial \sigma_1} \cdot \frac{\partial \sigma_1}{\partial R_s}$$

Thus from equations (6.10) and (6.27) we have

$$A_2 = A_1 \cdot \frac{C_2 R_1}{(\sigma_1 + C_2 R_1)^2} \tag{6.28}$$

and in the same way

$$A_3 = \frac{\partial \sigma_3}{\partial R_s} = A_2 \cdot \frac{R_2 L_3}{(\sigma_2 R_2 + L_3)^2} \tag{6.29}$$

$$A_4 = \frac{\partial \sigma_4}{\partial R_s} = A_3 \cdot \frac{G_3 C_4}{(\sigma_3 G_3 + C_4)^2} \tag{6.30}$$

$$A_5 = \frac{\partial \sigma_5}{\partial R_s} = A_4 \cdot \frac{R_4 L_5}{(\sigma_4 R_4 + L_5)^2} \tag{6.31}$$

$$A_6 = \frac{\partial \sigma_6}{\partial R_s} = A_5 \cdot \frac{G_5 C_6}{(\sigma_5 G_5 + C_6)^2} \tag{6.32}$$

$$A_7 = \frac{\partial \sigma_7}{\partial R_s} = A_6 \cdot \frac{R_6 L_7}{(\sigma_6 R_6 + L_7)^2} \tag{6.33}$$

$$A_8 = \frac{\partial \phi}{\partial R_s} = A_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7 + R_L)^2} \tag{6.34}$$

Thus from equations (6.26) to (6.34) we can find the sensitivity of the wave digital filter with respect to $R_s = R_1$. To find the sensitivity of wave digital filter with respect to L_1, i.e. $\frac{\partial H}{\partial L_1}$ from equation (6.24) we can write

$$\frac{\partial H}{\partial L_1} = \frac{\partial H}{\partial \sigma_1} \cdot \frac{\partial \sigma_1}{\partial L_1} + \frac{\partial H}{\partial \sigma_2} \cdot \frac{\partial \sigma_2}{\partial L_1} + \frac{\partial H}{\partial \sigma_3} \cdot \frac{\partial \sigma_3}{\partial L_1} + \frac{\partial H}{\partial \sigma_4} \cdot \frac{\partial \sigma_4}{\partial L_1}$$

$$+ \frac{\partial H}{\partial \sigma_5} \cdot \frac{\partial \sigma_5}{\partial L_1} + \frac{\partial H}{\partial \sigma_6} \cdot \frac{\partial \sigma_6}{\partial L_1} + \frac{\partial H}{\partial \sigma_7} \cdot \frac{\partial \sigma_7}{\partial L_1} + \frac{\partial H}{\partial \phi} \cdot \frac{\partial \phi}{\partial L_1} \qquad (6.35)$$

$$= B_1 \cdot \frac{\partial H}{\partial \sigma_1} + B_2 \cdot \frac{\partial H}{\partial \sigma_2} + B_3 \cdot \frac{\partial H}{\partial \sigma_3} + B_4 \cdot \frac{\partial H}{\partial \sigma_4}$$

$$+ B_5 \cdot \frac{\partial H}{\partial \sigma_5} + B_7 \cdot \frac{\partial H}{\partial \sigma_7} + B_8 \cdot \frac{\partial H}{\partial \phi} \qquad (6.36)$$

To find B_1, B_2, B_3, etc. we have from equations (6.6), and (6.36)

$$B_1 = \frac{\partial \sigma_1}{\partial L_1} = \frac{-R_1}{(R_1+L_1)^2} \qquad (6.37)$$

and from equation (6.35) and the Chain Rule

$$B_2 = \frac{\partial \sigma_2}{\partial L_1} = \frac{\partial \sigma_2}{\partial \sigma_1} \cdot \frac{\partial \sigma_1}{\partial L_1}$$

Thus from equations (6.10) and (6.37)

$$B_2 = B_1 \cdot \frac{G_1 C_2}{(\sigma_1 G_1 + C_2)^2} = \frac{-C_2 R_1^{\,2}}{(R_1+L_1)^2 (\sigma_1 + R_1 C_2)^2} \qquad (6.38)$$

and in the same way

$$B_3 = \frac{\partial \sigma_3}{\partial L_1} = B_2 \cdot \frac{R_2 L_3}{(\sigma_2 R_2 + L_3)^2} \qquad (6.39)$$

$$B_4 = \frac{\partial \sigma_4}{\partial L_1} = B_3 \cdot \frac{G_3 C_4}{(\sigma_3 G_3 + C_4)^2} \qquad (6.40)$$

$$B_5 = \frac{\partial\sigma_5}{\partial L_1} = B_4 \cdot \frac{R_4 L_5}{(\sigma_4 R_4 + L_5)^2} \tag{6.41}$$

$$B_6 = \frac{\partial\sigma_6}{\partial L_1} = B_5 \cdot \frac{G_5 C_6}{(\sigma_5 G_5 + C_6)^2} \tag{6.42}$$

$$B_7 = \frac{\partial\sigma_7}{\partial L_1} = B_6 \cdot \frac{R_6 L_7}{(\sigma_6 R_6 + L_7)^2} \tag{6.43}$$

$$B_8 = \frac{\partial\phi}{} = B_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7 + R_L)^2} \tag{6.44}$$

Thus from equations (6.36) to (6.44) we can find the sensitivity of the wave digital filter with respect to L_1.

To find the sensitivity of wave digital filter with respect to C_2, i.e. $\frac{\partial H}{\partial C_2}$ from equation (6.24) we can write

$$\frac{\partial H}{\partial C_2} = \frac{\partial H}{\partial\sigma_1} \cdot \frac{\partial\sigma_1}{\partial C_2} + \frac{\partial H}{\partial\sigma_2} \cdot \frac{\partial\sigma_2}{\partial C_2} + \frac{\partial H}{\partial\sigma_3} \cdot \frac{\partial\sigma_3}{\partial C_2} + \frac{\partial H}{\partial\sigma_4} \cdot \frac{\partial\sigma_4}{\partial C_2}$$

$$+ \frac{\partial H}{\partial\sigma_5} \cdot \frac{\partial\sigma_5}{\partial C_2} + \frac{\partial H}{\partial\sigma_6} \cdot \frac{\partial\sigma_6}{\partial C_2} + \frac{\partial H}{\partial\sigma_7} \cdot \frac{\partial\sigma_7}{\partial C_2} + \frac{\partial H}{\partial\sigma} \cdot \frac{\partial\phi}{\partial C_2} \tag{6.45}$$

or

$$\frac{\partial H}{\partial C_2} = D_1 \cdot \frac{\partial H}{\partial\sigma_1} + D_2 \cdot \frac{\partial H}{\partial\sigma_2} + D_3 \cdot \frac{\partial H}{\partial\sigma_3} + D_4 \cdot \frac{\partial H}{\partial\sigma_4} + D_5 \cdot \frac{\partial H}{\partial\sigma_5}$$

$$+ D_6 \cdot \frac{\partial H}{\partial\sigma_6} + D_7 \cdot \frac{\partial H}{\partial\sigma_7} + D_8 \cdot \frac{\partial H}{\partial\sigma_8} \tag{6.46}$$

Note that the coefficient of the first term, i.e. $D_1 = 0$.

From equation (6.9)

$$D_2 = \frac{\partial\sigma_2}{\partial C_2} = \frac{-G_2}{(G_2+C_2)^2} \qquad (6.47)$$

and from equation (6.45) and the Chain Rule

$$D_3 = \frac{\partial\sigma_3}{\partial C_2} = \frac{\partial\sigma_3}{\partial\sigma_2} \cdot \frac{\partial\sigma_2}{\partial C_2} \qquad (6.48)$$

Thus from equations (6.12) and (6.47)

$$D_3 = D_2 \cdot \frac{R_2L_3}{(\sigma_2R_2+L_3)^2} \qquad (6.49)$$

and in the same way

$$D_4 = D_3 \cdot \frac{G_3C_4}{(\sigma_3G_3+C_4)^2} \qquad (6.50)$$

$$D_5 = D_4 \cdot \frac{R_4L_5}{(\sigma_4R_4+L_5)^2} \qquad (6.51)$$

$$D_6 = D_5 \cdot \frac{G_5C_6}{(\sigma_5G_5+C_6)^2} \qquad (6.52)$$

$$D_7 = D_6 \cdot \frac{R_6L_7}{(\sigma_6R_6+L_7)^2} \qquad (6.53)$$

$$D_8 = D_7 \cdot \frac{-2G_7R_L}{(\sigma_7G_7+R_L)^2} \qquad (6.54)$$

Thus from equations (6.46) to (6.54) we can find the sensitivity of the wave digital filter with respect to C_2.

To find the sensitivity of wave digital filter with respect to L_3, C_4, L_5, C_6, L_7 in the same way we have

$$\frac{\partial H}{\partial L_3} = E_3 \cdot \frac{\partial H}{\partial \sigma_3} + E_4 \cdot \frac{\partial H}{\partial \sigma_4} + E_5 \cdot \frac{\partial H}{\partial \sigma_5} + E_6 \cdot \frac{\partial H}{\partial \sigma_6} + E_7 \cdot \frac{\partial H}{\partial \sigma_7} + E_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.55)$$

and

$$\frac{\partial H}{\partial C_4} = P_4 \cdot \frac{\partial H}{\partial \sigma_4} + P_5 \cdot \frac{\partial H}{\partial \sigma_5} + P_6 \cdot \frac{\partial H}{\partial \sigma_6} + P_7 \cdot \frac{\partial H}{\partial \sigma_7} + P_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.56)$$

$$\frac{\partial H}{\partial L_5} = Q_5 \cdot \frac{\partial H}{\partial \sigma_5} + Q_6 \cdot \frac{\partial H}{\partial \sigma_6} + Q_7 \cdot \frac{\partial H}{\partial \sigma_7} + Q_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.57)$$

$$\frac{\partial H}{\partial C_6} = S_6 \cdot \frac{\partial H}{\partial \sigma_6} + S_7 \cdot \frac{\partial H}{\partial \sigma_7} + S_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.58)$$

$$\frac{\partial H}{\partial L_7} = U_7 \cdot \frac{\partial H}{\partial \sigma_7} + U_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.59)$$

$$\frac{\partial H}{\partial R_L} = V_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.60)$$

where

$$E_3 = \frac{-R_3}{(R_3+L_3)^2} \quad (6.61)$$

$$E_4 = E_3 \cdot \frac{G_3C_4}{(\sigma_3G_3+C_4)^2} \quad (6.62)$$

$$E_5 = E_4 \cdot \frac{R_4L_5}{(\sigma_4R_4+L_5)^2} \quad (6.63)$$

$$E_6 = E_5 \cdot \frac{G_5C_6}{(\sigma_5G_5+C_6)^2} \quad (6.64)$$

$$E_7 = E_6 \cdot \frac{R_6L_7}{(\sigma_6R_6+L_7)^2} \quad (6.65)$$

$$E_8 = E_7 \cdot \frac{-2G_7R_L}{(\sigma_7G_7+R_L)^2} \quad (6.66)$$

and

$$P_4 = \frac{-G_4}{(G_5+C_4)^2} \tag{6.67}$$

$$P_5 = P_4 \cdot \frac{R_4 L_5}{(\sigma_4 R_4+L_5)^2} \tag{6.68}$$

$$P_6 = P_5 \cdot \frac{G_5 C_6}{(\sigma_5 G_5+C_6)^2} \tag{6.69}$$

$$P_7 = P_6 \cdot \frac{R_6 L_7}{(\sigma_6 R_6+L_7)^2} \tag{6.70}$$

$$P_8 = P_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7+R_L)^2} \tag{6.71}$$

and

$$Q_5 = \frac{-R_5}{(R_5+L_5)^2} \tag{6.72}$$

$$Q_6 = Q_5 \cdot \frac{G_5 C_6}{(\sigma_5 G_5+C_6)^2} \tag{6.73}$$

$$Q_7 = Q_6 \cdot \frac{R_6 L_7}{(\sigma_5 R_6+L_7)^2} \tag{6.74}$$

$$Q_8 = Q_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7+R_L)^2} \tag{6.75}$$

and

$$S_6 = \frac{-G_6}{(G_6+C_6)^2} \tag{6.76}$$

$$S_7 = S_6 \cdot \frac{R_6 L_7}{(\sigma_6 R_6+L_7)^2} \tag{6.77}$$

$$S_8 = S_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7+R_L)^2} \tag{6.78}$$

and

$$U_7 = \frac{-R_7}{(R_7+L_7)^2} \tag{6.79}$$

$$U_8 = U_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7+R_L)^2} \tag{6.80}$$

and

$$V_8 = \frac{2G_7\sigma_7}{(\sigma_7 G_7+R_L)^2} \tag{6.81}$$

Thus we can find the sensitivity of wave digital filter with respect to the original filter component values.

D. EXPERIMENTAL STUDY ON THE INTERNAL SENSITIVITY BEHAVIOR OF WAVE DIGITAL FILTERS

In order to analyze the internal sensitivity behavior of wave digital filters with respect to the multiplier coefficients and compare them to the sensitivity of wave digital filters with respect to the original filter component values, in total nine different seventh order low pass filters were taken from the Handbook of Filter Synthesis [1] with source resistances varying from 0 to 10 ohms. The sensitivity behavior of these filters in the frequency domain with respect to both wave digital filter multiplier coefficients and the original filter component values were found using the procedures set in examples 1 and 2. These filters were

i) seventh order .5 db ripple low pass Chebyshev filter with R_s=1.0

ii) seventh order .1 db ripple low pass Chebyshev filter with R_s=1.0

iii) seventh order Butterworth low pass filter with R_s=1.0

iv) seventh order .5 db ripple low pass Chebyshev filter with R_s=10.0

v) seventh order .1 db ripple low pass Chebyshev filter with R_s=10.0

vi) seventh order Butterworth low pass filter with R_s=10.0

vii) seventh order .5 db ripple low pass Chebyshev filter with R_s=0.0

viii) seventh order .1 db ripple low pass Chebyshev filter with R_s=0.0

ix) seventh order Butterworth low pass filter with R_s = 0.0

Note that these filters are the same filters investigated in the sensitivity analysis of Chapter V. In order to make the comparison between the two sensitivity functions, i.e. sensitivity with respect to wave digital filter multiplier coefficients and sensitivity with respect to wave digital filter original components more meaningful, the semi-logarithmic sensitivity function which is in fact normalized or percentage change in sensitivity is used. These normalized sensitivity functions are $\frac{\partial H}{\partial \sigma_n} \cdot \sigma_n$ and are plotted on the same graph in the frequency domain in Figs. 6.3 to 6.11.

Note that there are nine variables for the seventh order filter in terms of original components(i.e. L_1, C_2, L_3, C_4, L_5, C_6, L_7, R_s, and R_L) while there are eight variables for the filter with no delay free path in port two (i.e. σ_1, σ_2,---σ_7 and ϕ), since the source reflection coefficient, θ is made equal to zero. Thus the curves of $\frac{\partial H}{\partial R_s} \cdot R_s$ are plotted on a single graph. The remainder of eight variables are plotted on the same coordinates, i.e. $\frac{\partial H}{\partial L_1} \cdot L_1$ is plotted on the same coordinate as $\frac{\partial H}{\partial \sigma_1} \cdot \sigma_1$ and so on. Finally $\frac{\partial H}{\partial R_L} \cdot R_L$ is plotted with $\frac{\partial H}{\partial \phi} \cdot \phi$ on the same coordinates. It is important to note that for the filters with $R_s = 0$, the normalized partial sensitivity $\frac{\partial H}{\partial \sigma_1} \cdot \sigma_1$ equals to zero since $\sigma_1 = 0$ when $R_1 = R_s = 0$. Also we note that for this special case $\frac{\partial H}{\partial L_1} = 0$ since in equation (6.26) all the coefficients of $\frac{\partial H}{\partial L_1}$ are zero with $R_1 = R_s = 0$. Thus for these reasons for when $R_s = 0$ we have plotted only $\frac{\partial H}{\partial \sigma_1}$ rather than $\frac{\partial H}{\partial \sigma_1} \cdot \sigma_1$ and we also note that $\frac{\partial H}{\partial L_1}$ is zero for all frequencies.

E. ANALYSIS OF THE INTERNAL STRUCTURE OF THE WAVE DIGITAL FILTERS IN FREQUENCY DOMAIN USING FILTER PARTIAL SENSITIVITY FUNCTIONS

Although all the filters under investigation are seventh order low pass, they were chosen to be as different from each other as possible, in type and source termination resistances. As mentioned earlier these filters are identical to those analyzed in Chapter V for sensitivity

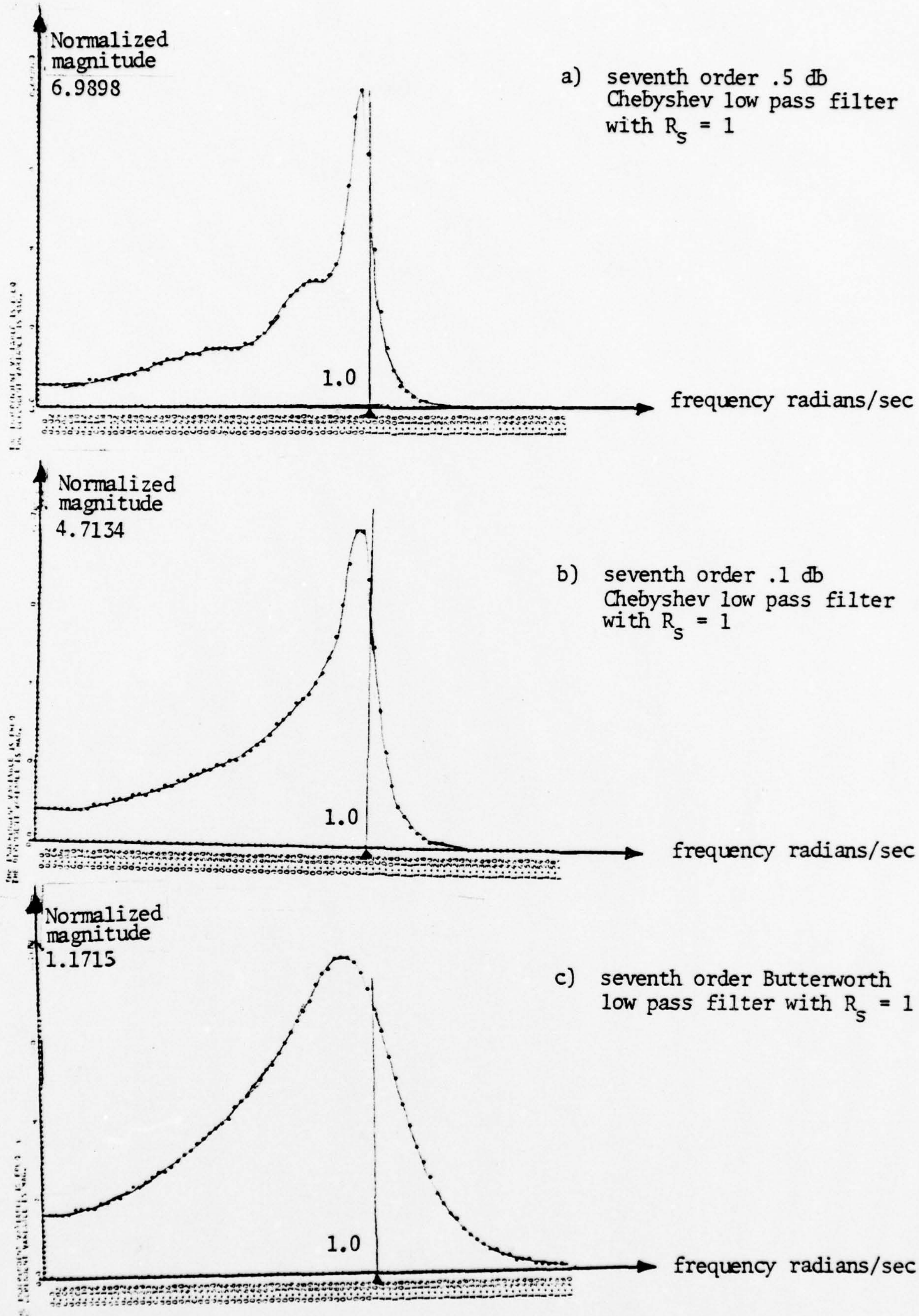
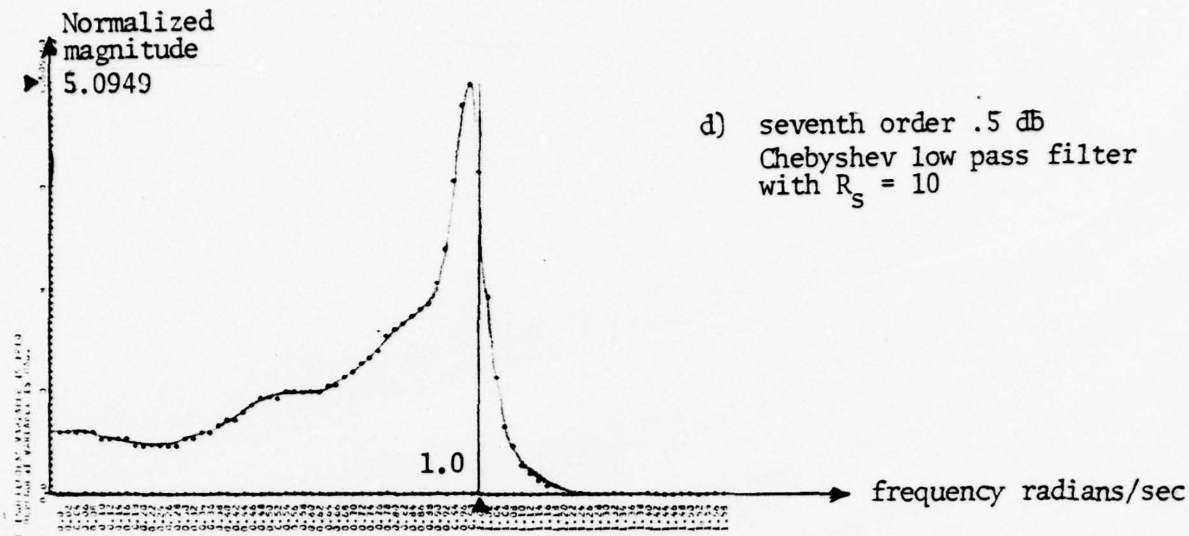
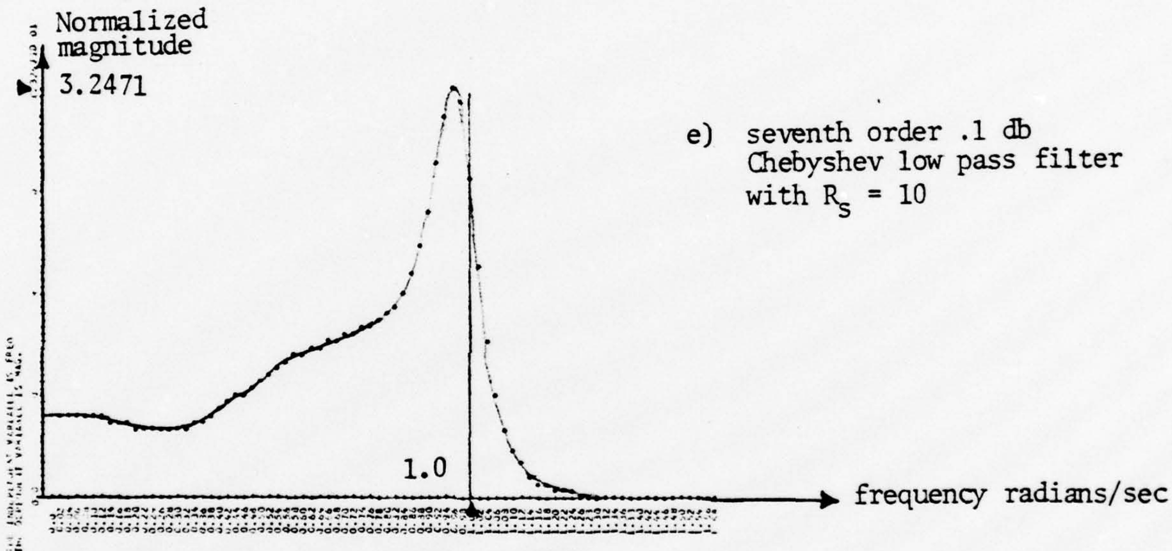


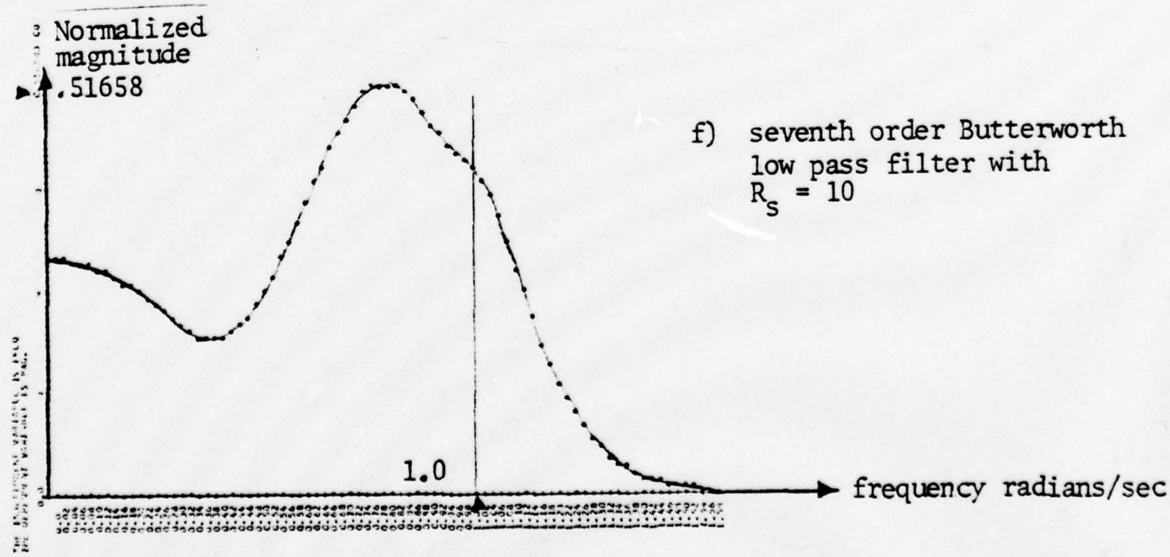
Fig. 6.5. The graphs of normalized sensitivity function $\frac{\partial H}{\partial R_s} \cdot R_s$ of various simple wave digital filters.



d) seventh order .5 db Chebyshev low pass filter with R_s = 10

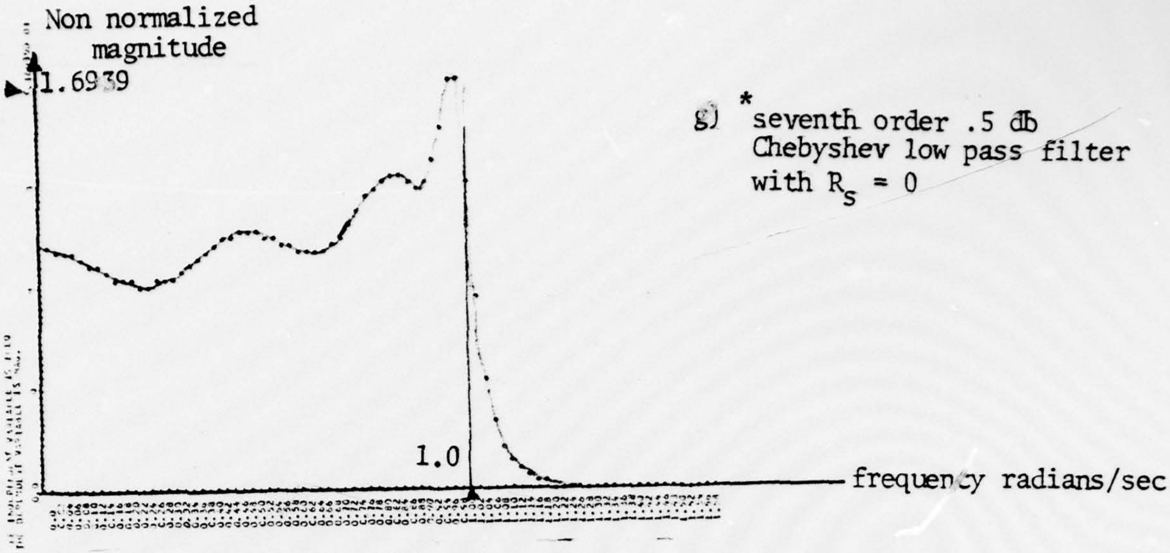


e) seventh order .1 db Chebyshev low pass filter with R_s = 10

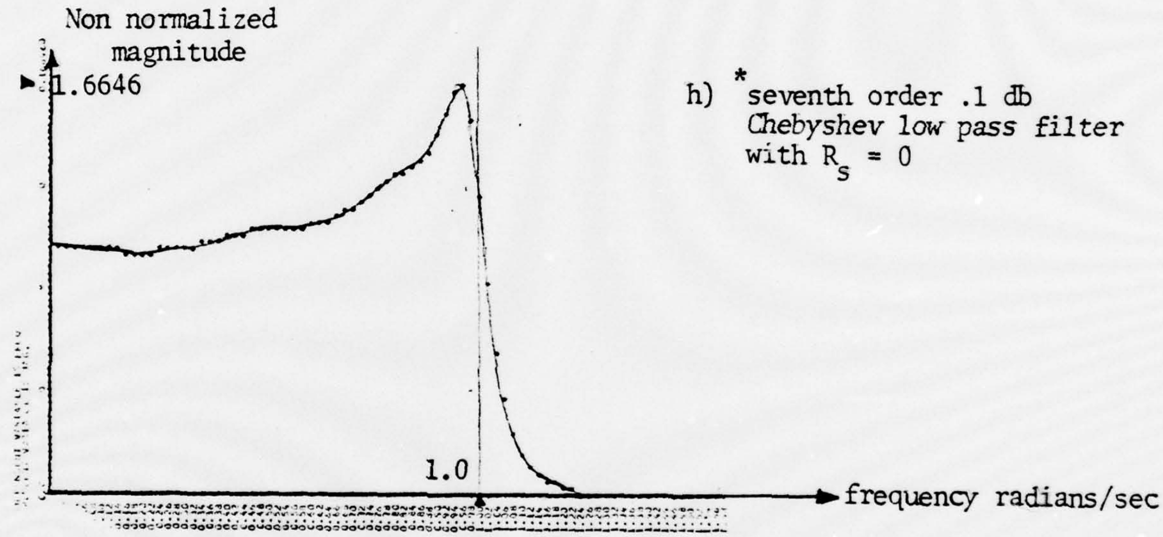


f) seventh order Butterworth low pass filter with R_s = 10

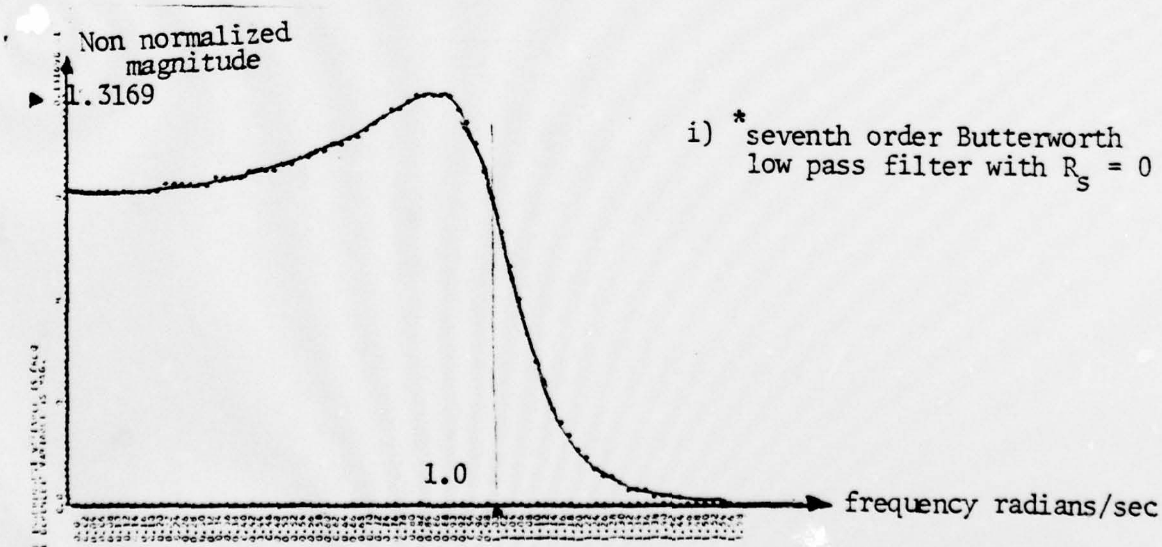
Fig. 6.5. Continued



g) *seventh order .5 db Chebyshev low pass filter with $R_s = 0$



h) *seventh order .1 db *Chebyshev low pass filter* with $R_s = 0$



i) *seventh order Butterworth low pass filter with $R_s = 0$

Fig. 6.5. Continued

*Note that for the special case of R_s=0, non normalized sensitivity function $\frac{\partial H}{\partial R_s}$ are plotted.

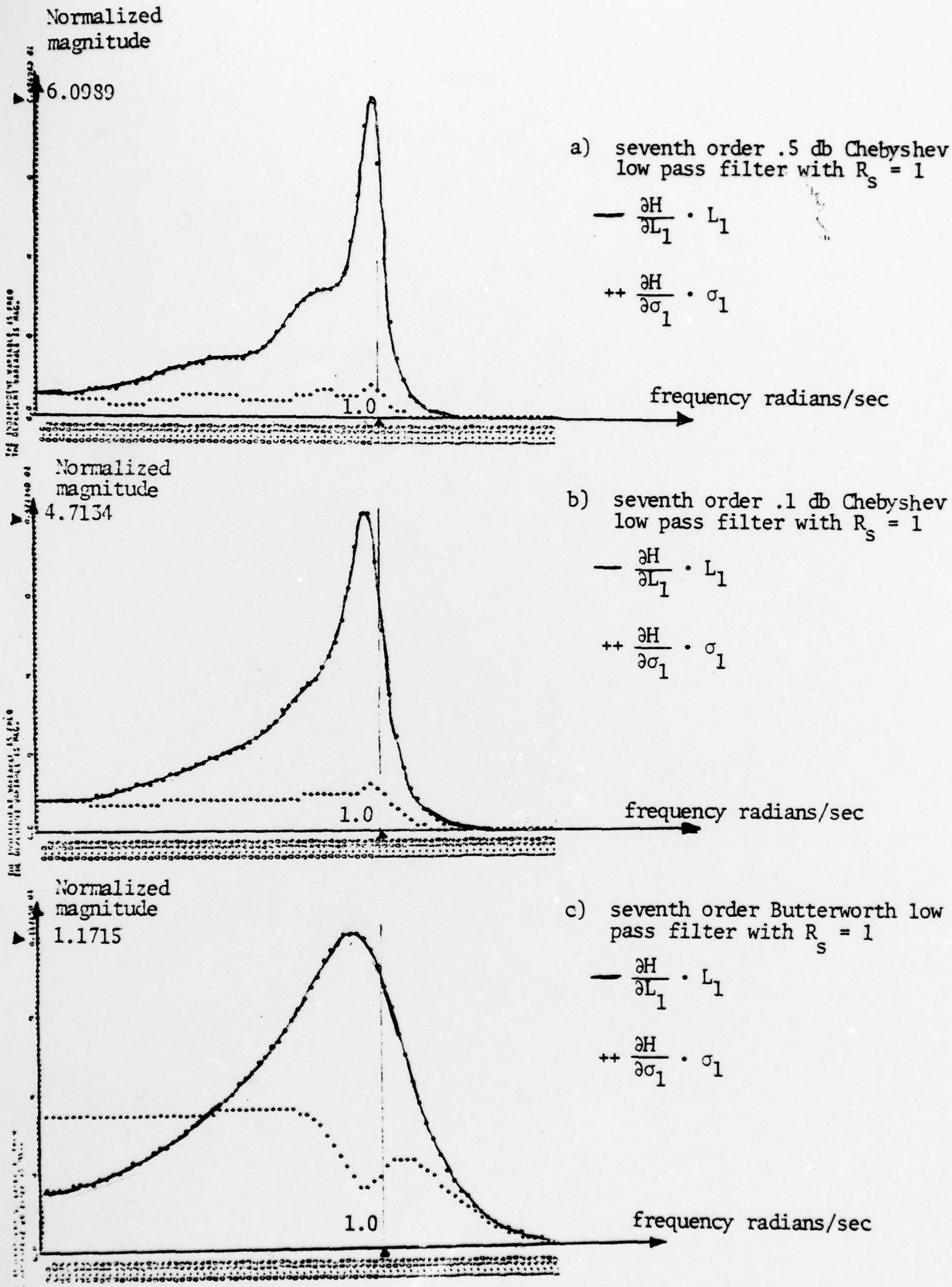
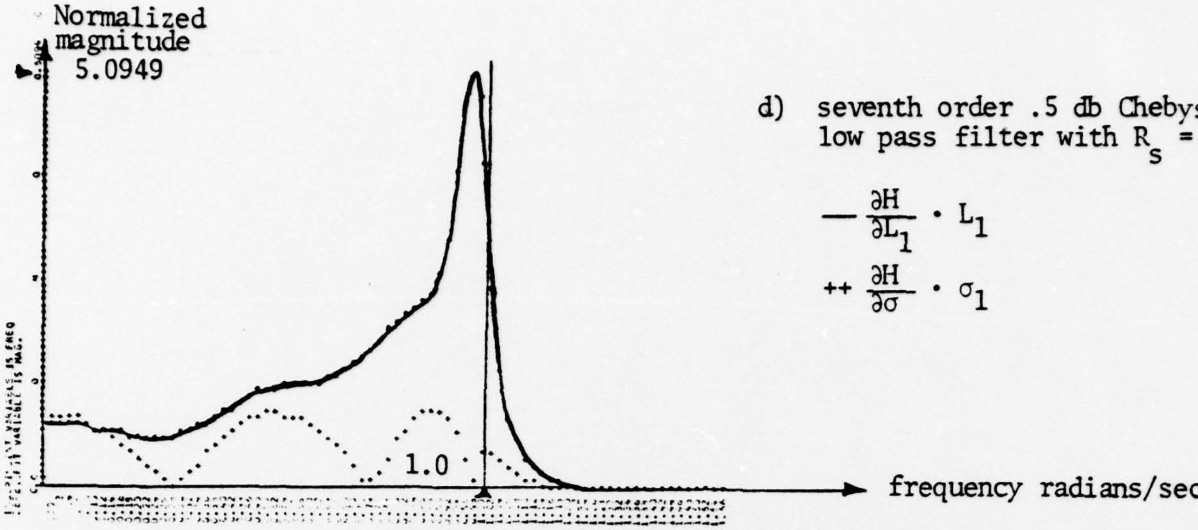


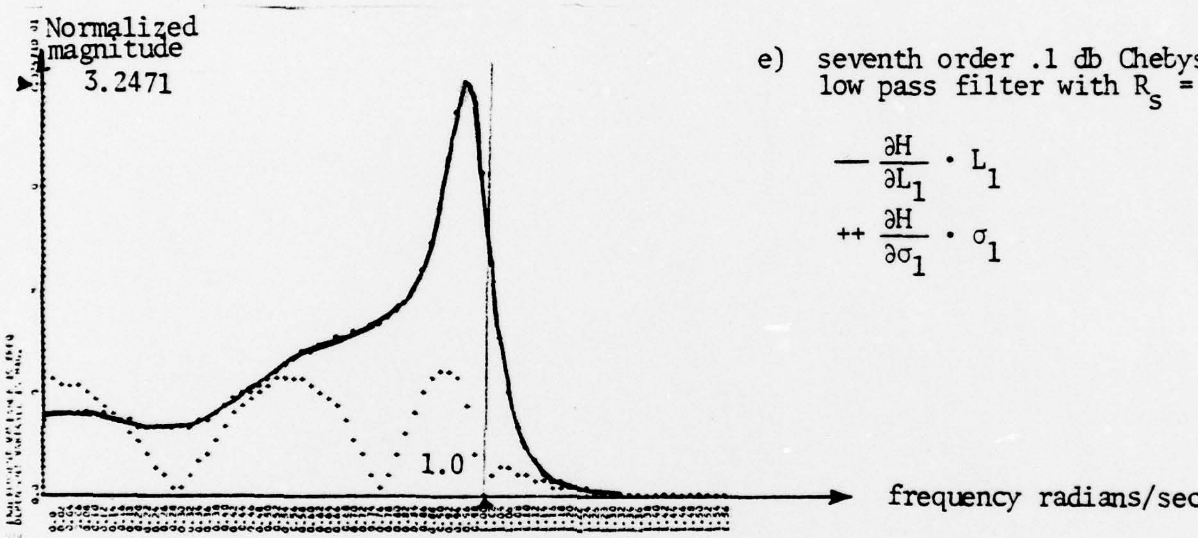
Fig. 6.6. The graphs of normalized sensitivity functions $\partial H/\partial L_1 \cdot L_1$ and $\partial H/\partial \sigma_1 \cdot \sigma_1$ of various simple wave digital filters.



d) seventh order .5 db Chebyshev low pass filter with R_s = 10

— $\frac{\partial H}{\partial L_1} \cdot L_1$

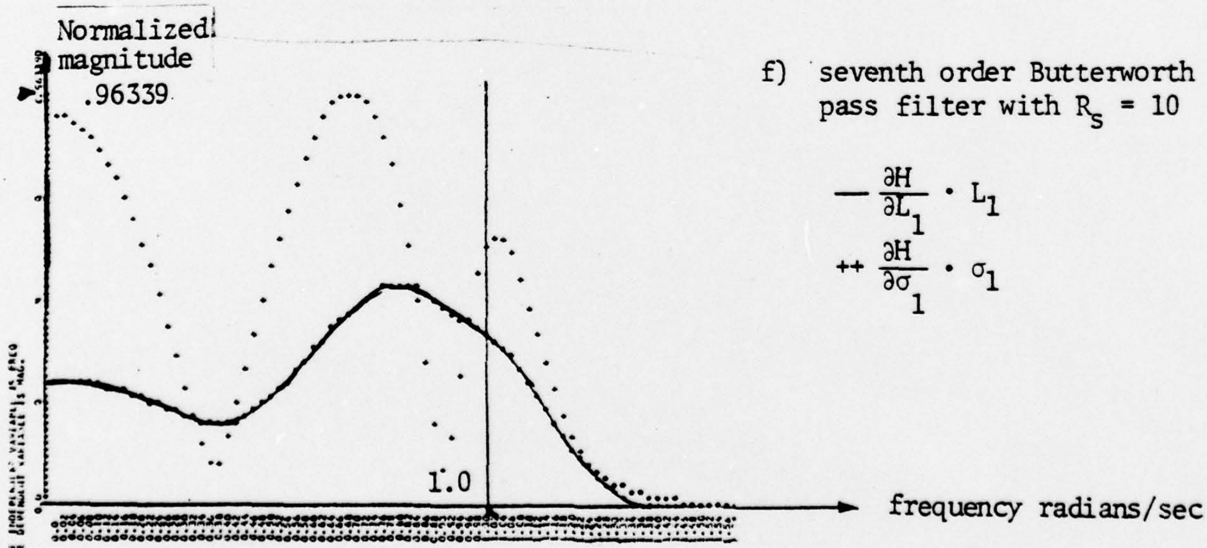
++ $\frac{\partial H}{\partial \sigma} \cdot \sigma_1$



e) seventh order .1 db Chebyshev low pass filter with R_s = 10

— $\frac{\partial H}{\partial L_1} \cdot L_1$

++ $\frac{\partial H}{\partial \sigma_1} \cdot \sigma_1$

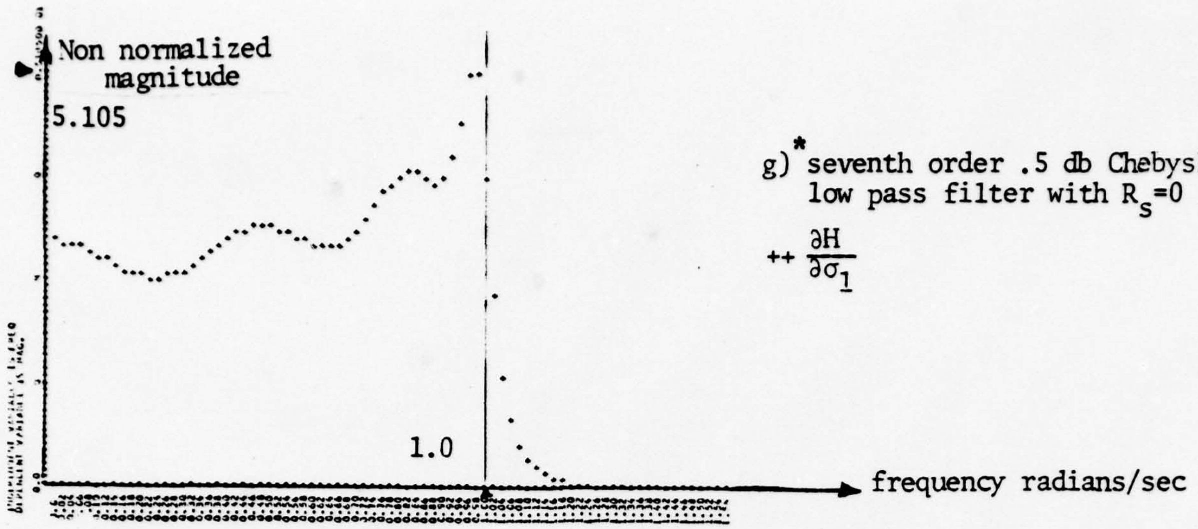


f) seventh order Butterworth low pass filter with R_s = 10

— $\frac{\partial H}{\partial L_1} \cdot L_1$

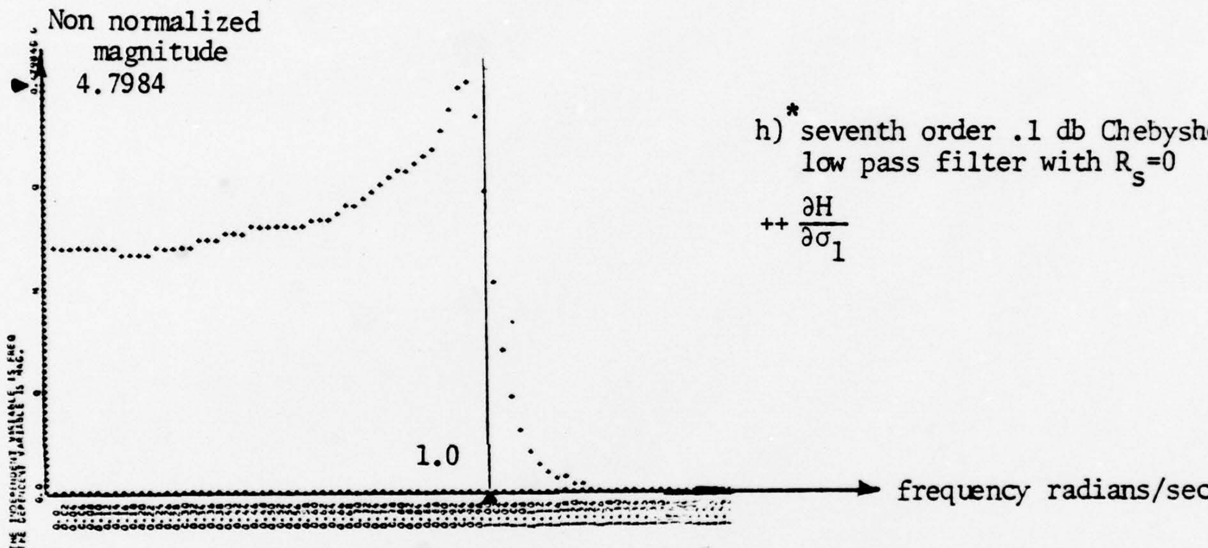
++ $\frac{\partial H}{\partial \sigma_1} \cdot \sigma_1$

Fig. 6.6. Continued



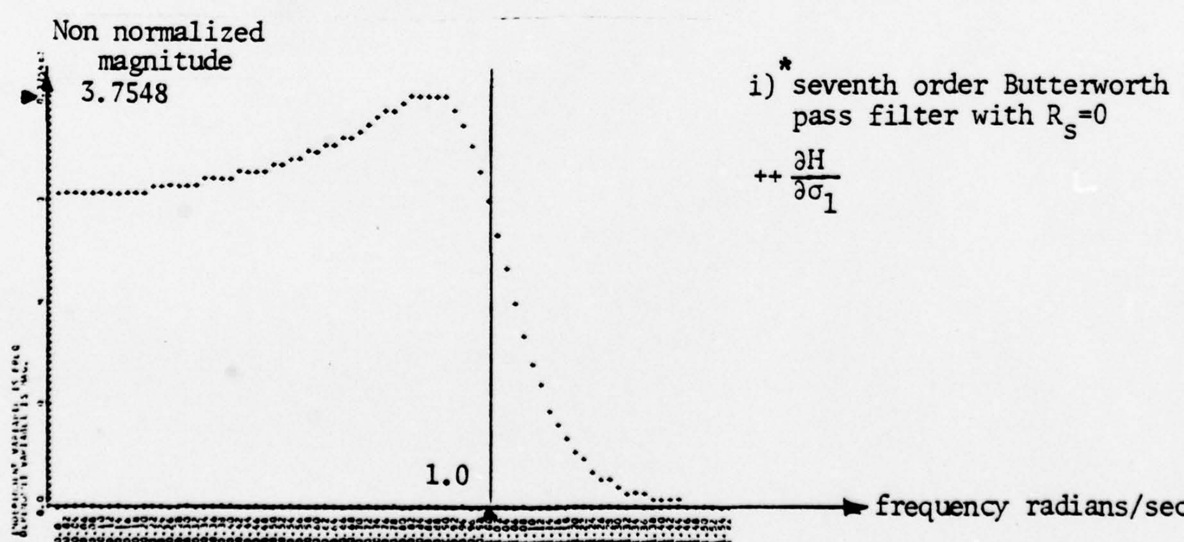
g)* seventh order .5 db Chebyshev low pass filter with $R_s=0$

++ $\frac{\partial H}{\partial \sigma_1}$



h)* seventh order .1 db Chebyshev low pass filter with $R_s=0$

++ $\frac{\partial H}{\partial \sigma_1}$



i)* seventh order Butterworth low pass filter with $R_s=0$

++ $\frac{\partial H}{\partial \sigma_1}$

Fig. 6.6. Continued *For the special case of $R_s=0$ non normalized curves are plotted.

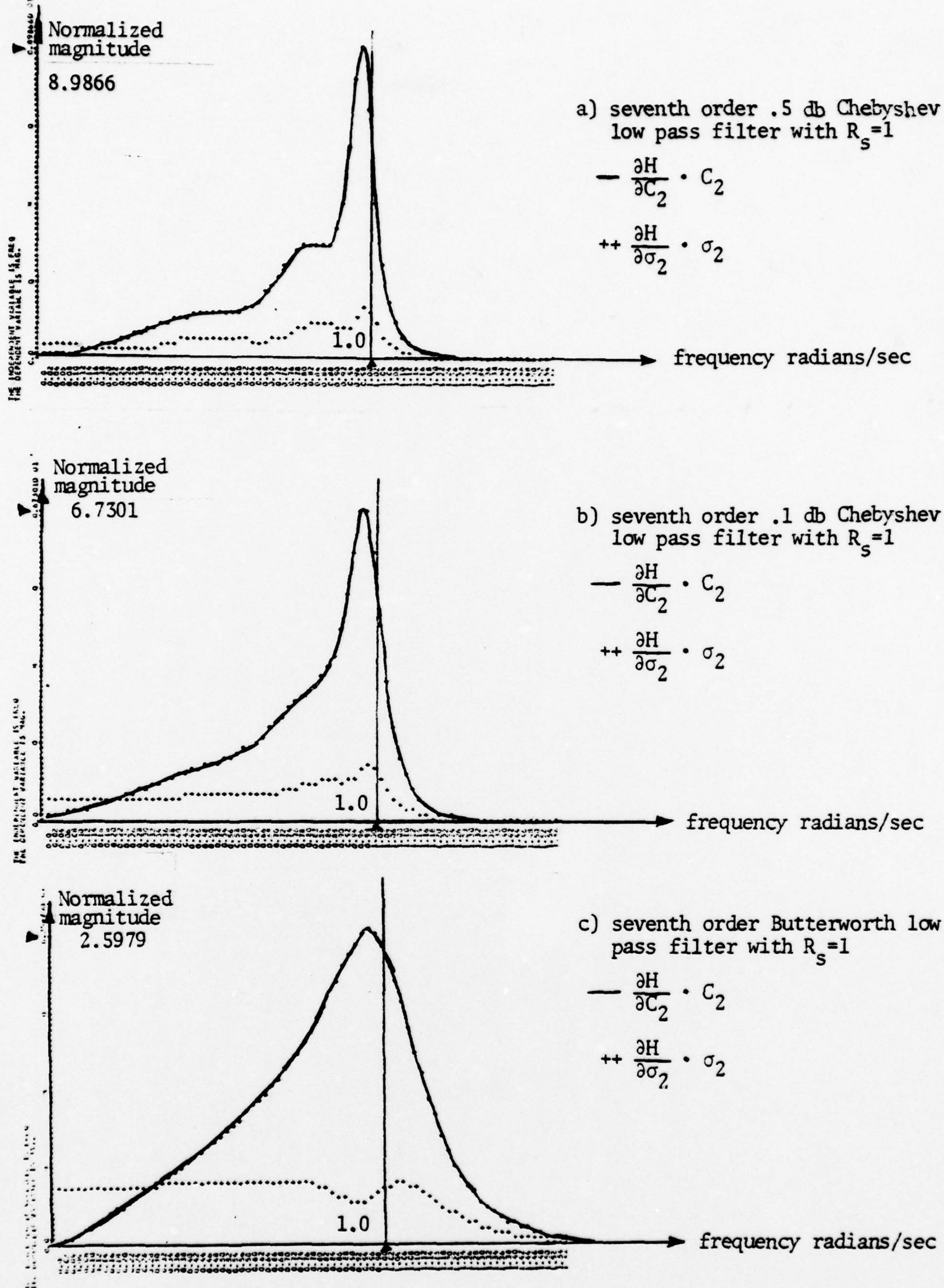
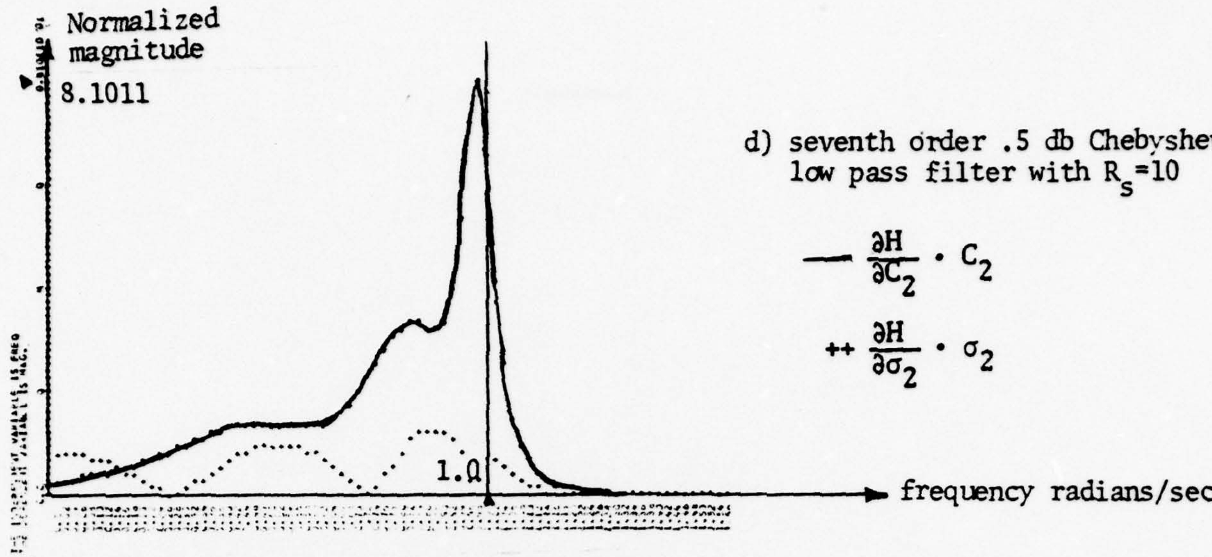


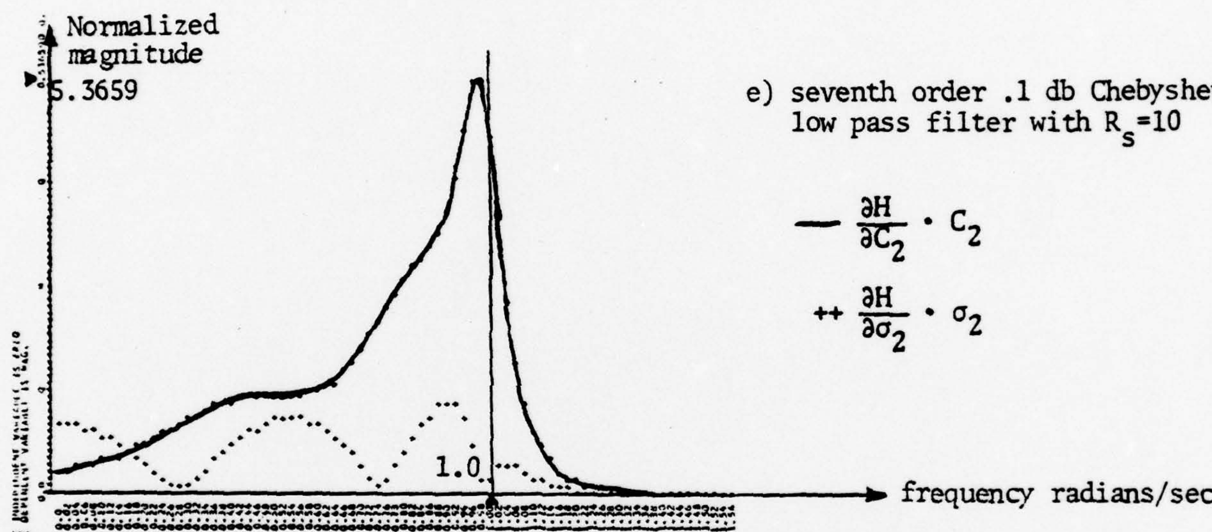
Fig. 6.7. The graphs of normalized sensitivity function, $\frac{\partial H}{\partial C_2} \cdot C_2$ and $\frac{\partial H}{\partial \sigma_2} \cdot \sigma_2$ of various simple wave digital filters.



d) seventh order .5 db Chebyshev low pass filter with R_s=10

$$— \frac{\partial H}{\partial C_2} \cdot C_2$$

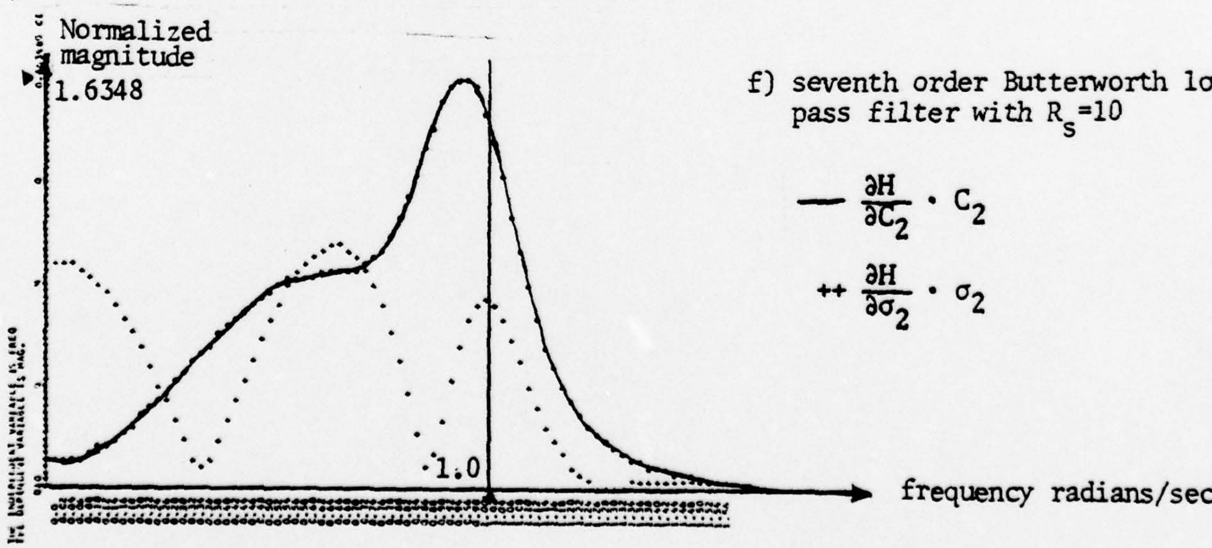
$$++ \frac{\partial H}{\partial \sigma_2} \cdot \sigma_2$$



e) seventh order .1 db Chebyshev low pass filter with R_s=10

$$— \frac{\partial H}{\partial C_2} \cdot C_2$$

$$++ \frac{\partial H}{\partial \sigma_2} \cdot \sigma_2$$

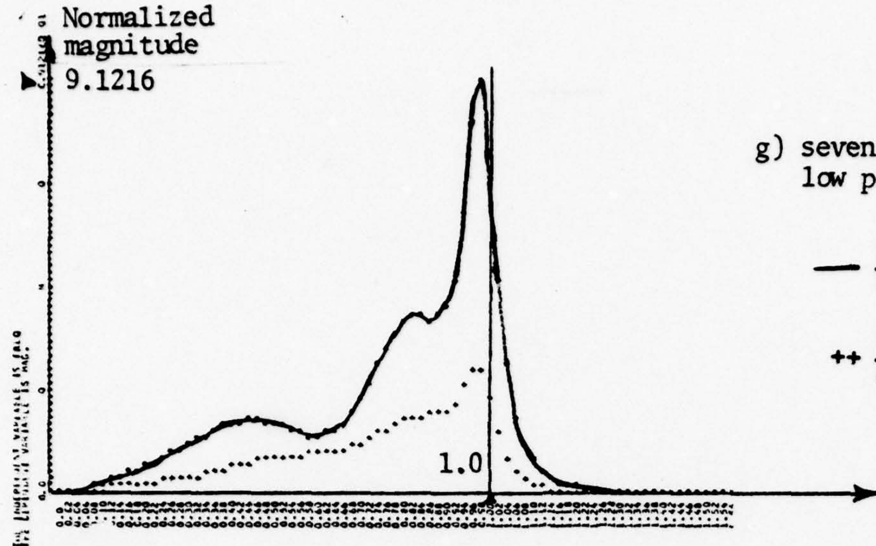


f) seventh order Butterworth low pass filter with R_s=10

$$— \frac{\partial H}{\partial C_2} \cdot C_2$$

$$++ \frac{\partial H}{\partial \sigma_2} \cdot \sigma_2$$

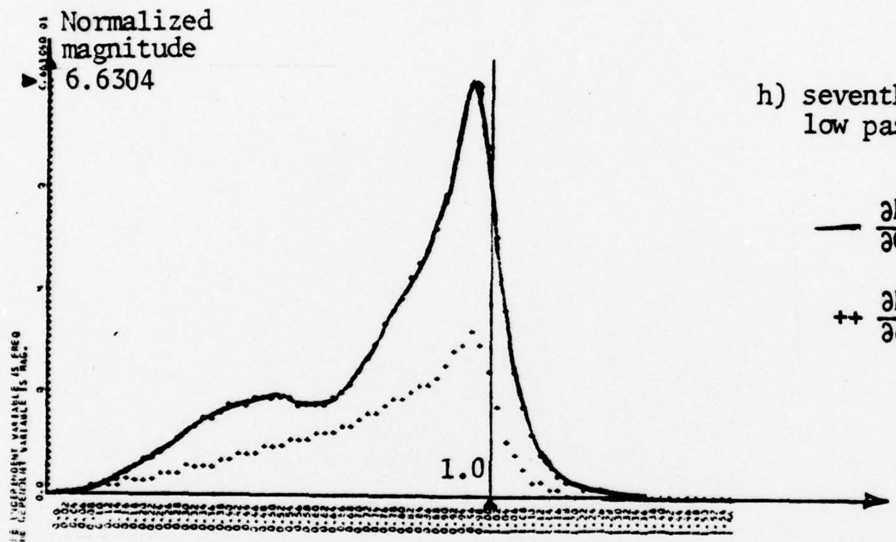
Fig. 6.7. Continued



g) seventh order .5 db Chebyshev low pass filter with $R_s=0$

— $\frac{\partial H}{\partial C_2} \cdot C_2$

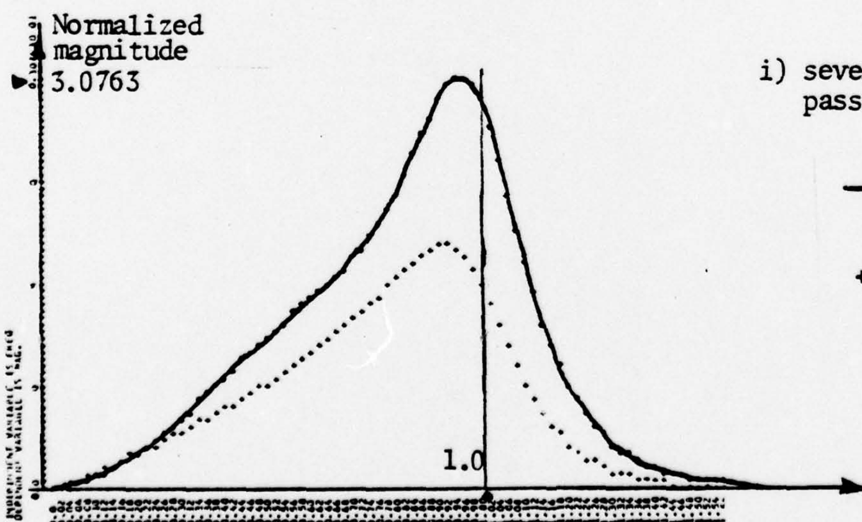
++ $\frac{\partial H}{\partial \sigma_2} \cdot \sigma_2$



h) seventh order .1 db Chebyshev low pass filter with $R_s=0$

— $\frac{\partial H}{\partial C_2} \cdot C_2$

++ $\frac{\partial H}{\partial \sigma_2} \cdot \sigma_2$



i) seventh order Butterworth low pass filter with $R_s=0$

— $\frac{\partial H}{\partial C_2} \cdot C_2$

++ $\frac{\partial H}{\partial \sigma_2} \cdot \sigma_2$

Fig. 6.7. Continued

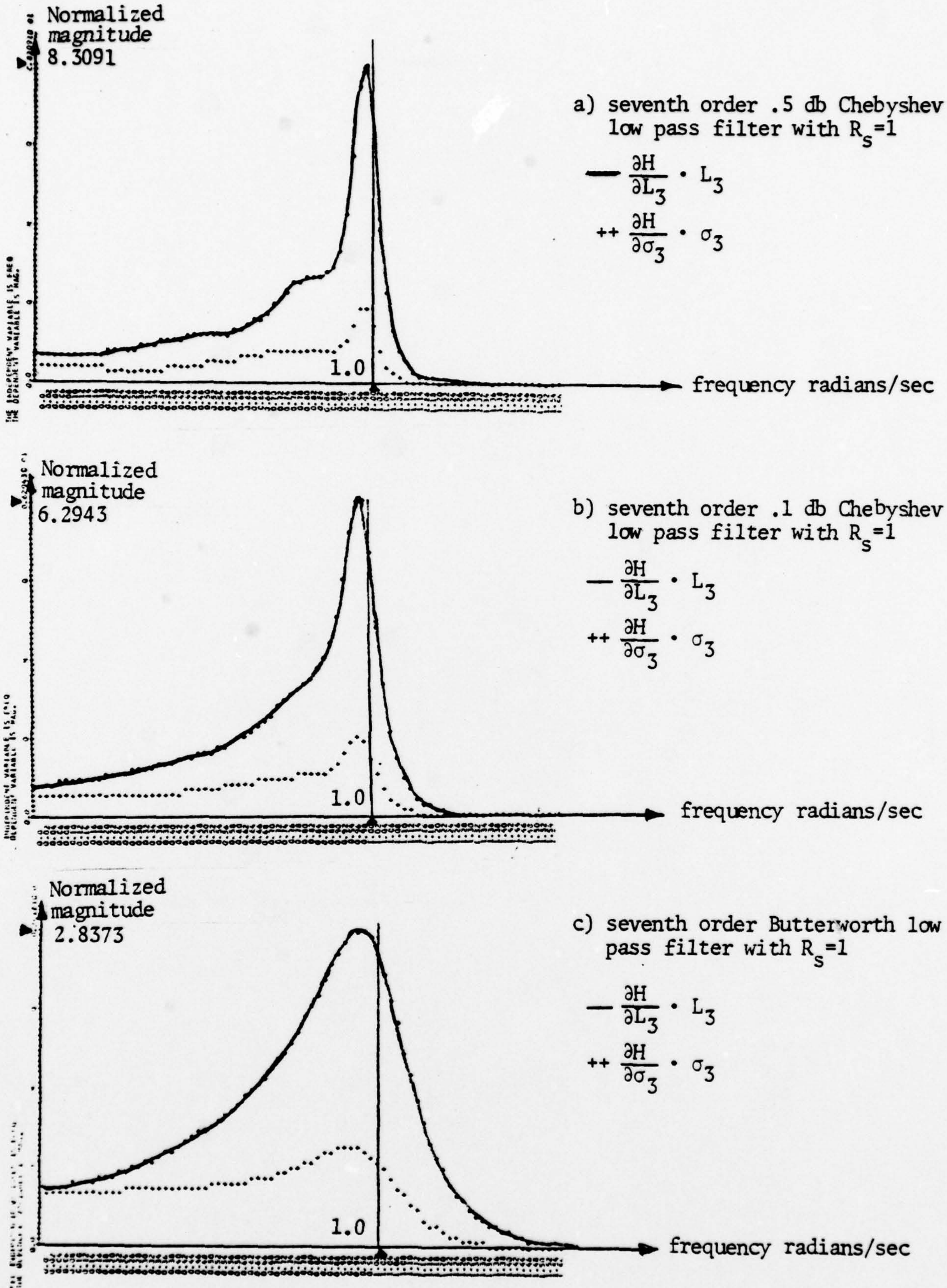
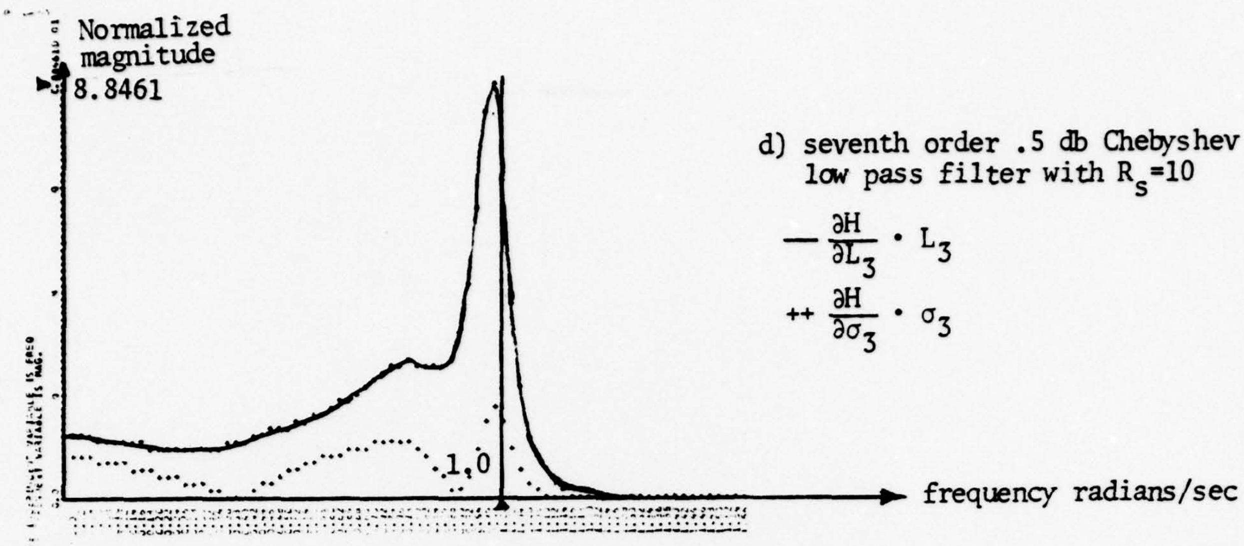
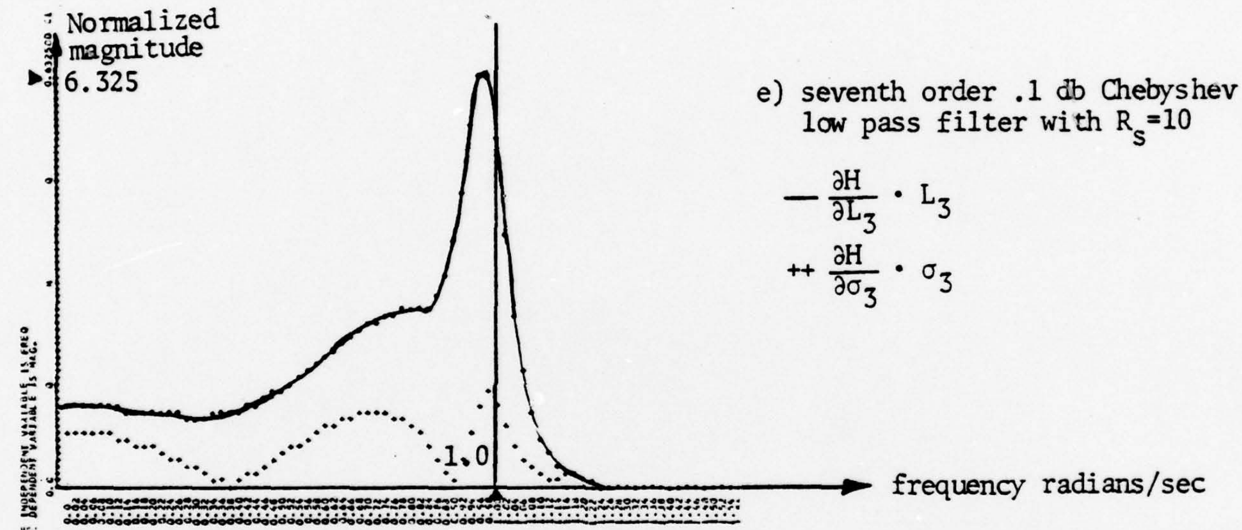


Fig. 6.8. The graphs of normalized sensitivity function, $\frac{\partial H}{\partial L_3} \cdot L_3$ and $\frac{\partial H}{\partial \sigma_3} \cdot \sigma_3$ of various simple wave digital filters.





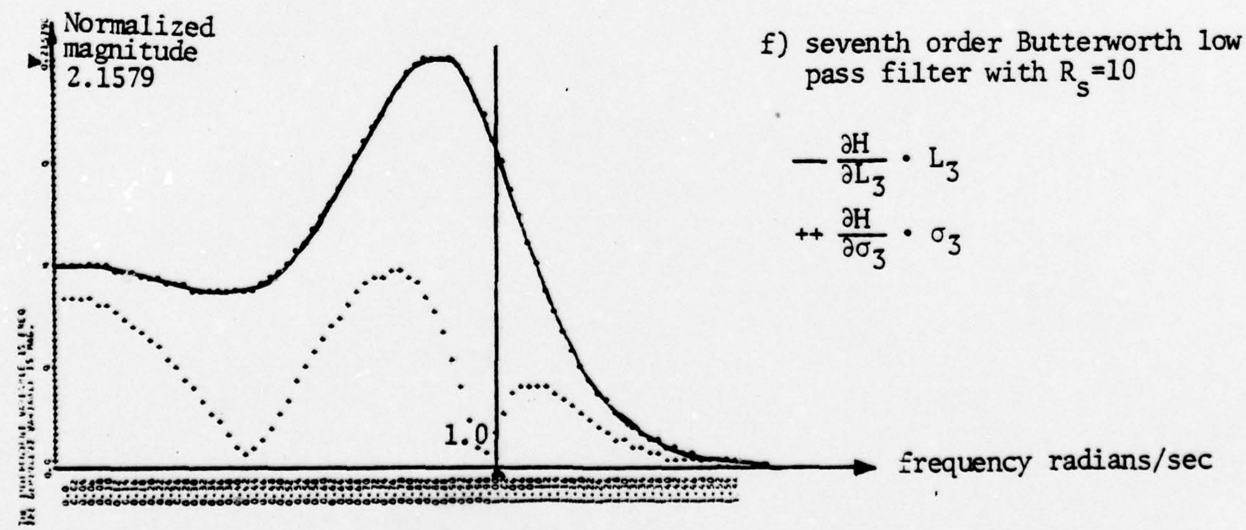
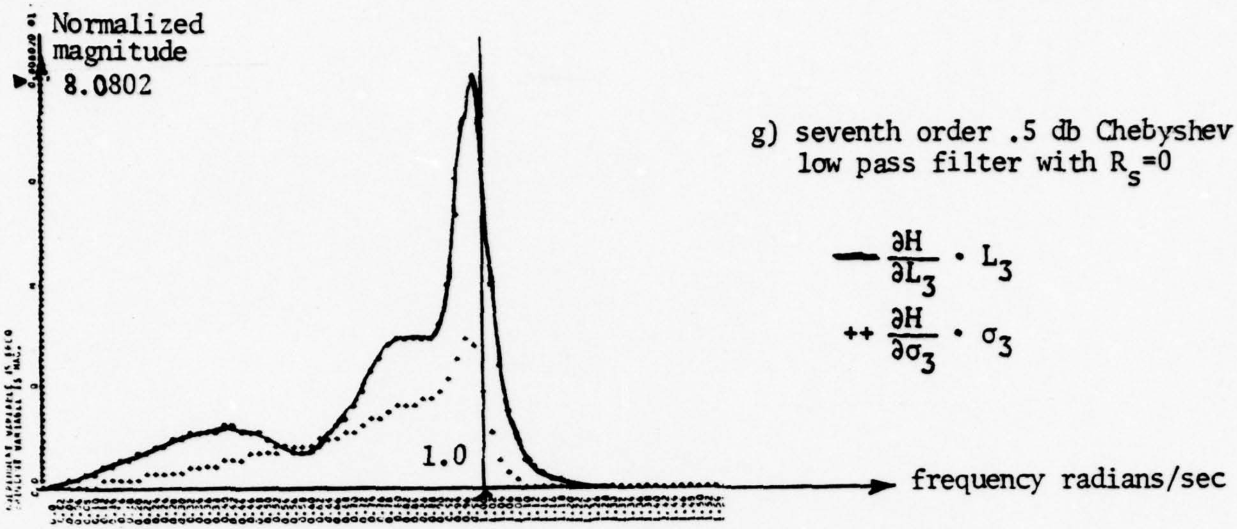


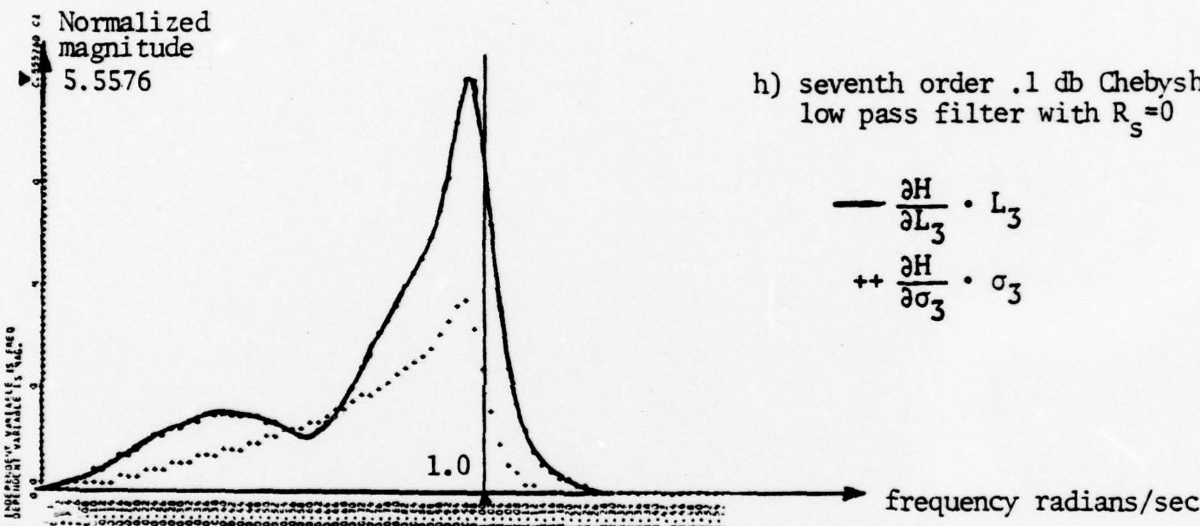
Fig. 6.8. Continued



g) seventh order .5 db Chebyshev low pass filter with $R_s=0$

— $\frac{\partial H}{\partial L_3} \cdot L_3$

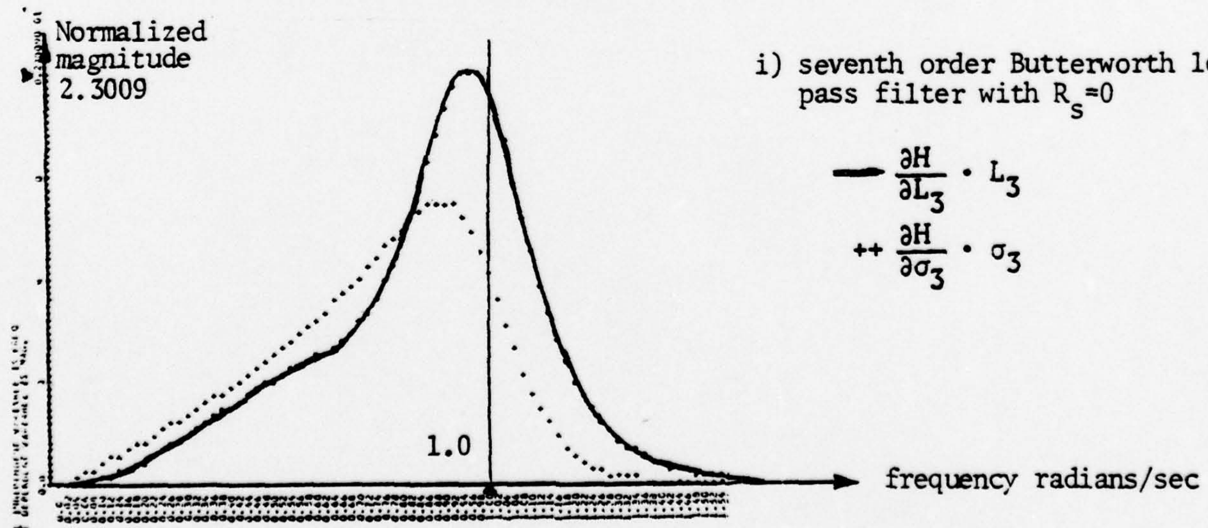
++ $\frac{\partial H}{\partial \sigma_3} \cdot \sigma_3$



h) seventh order .1 db Chebyshev low pass filter with $R_s=0$

— $\frac{\partial H}{\partial L_3} \cdot L_3$

++ $\frac{\partial H}{\partial \sigma_3} \cdot \sigma_3$



i) seventh order Butterworth low pass filter with $R_s=0$

— $\frac{\partial H}{\partial L_3} \cdot L_3$

++ $\frac{\partial H}{\partial \sigma_3} \cdot \sigma_3$

Fig. 6.8. Continued.

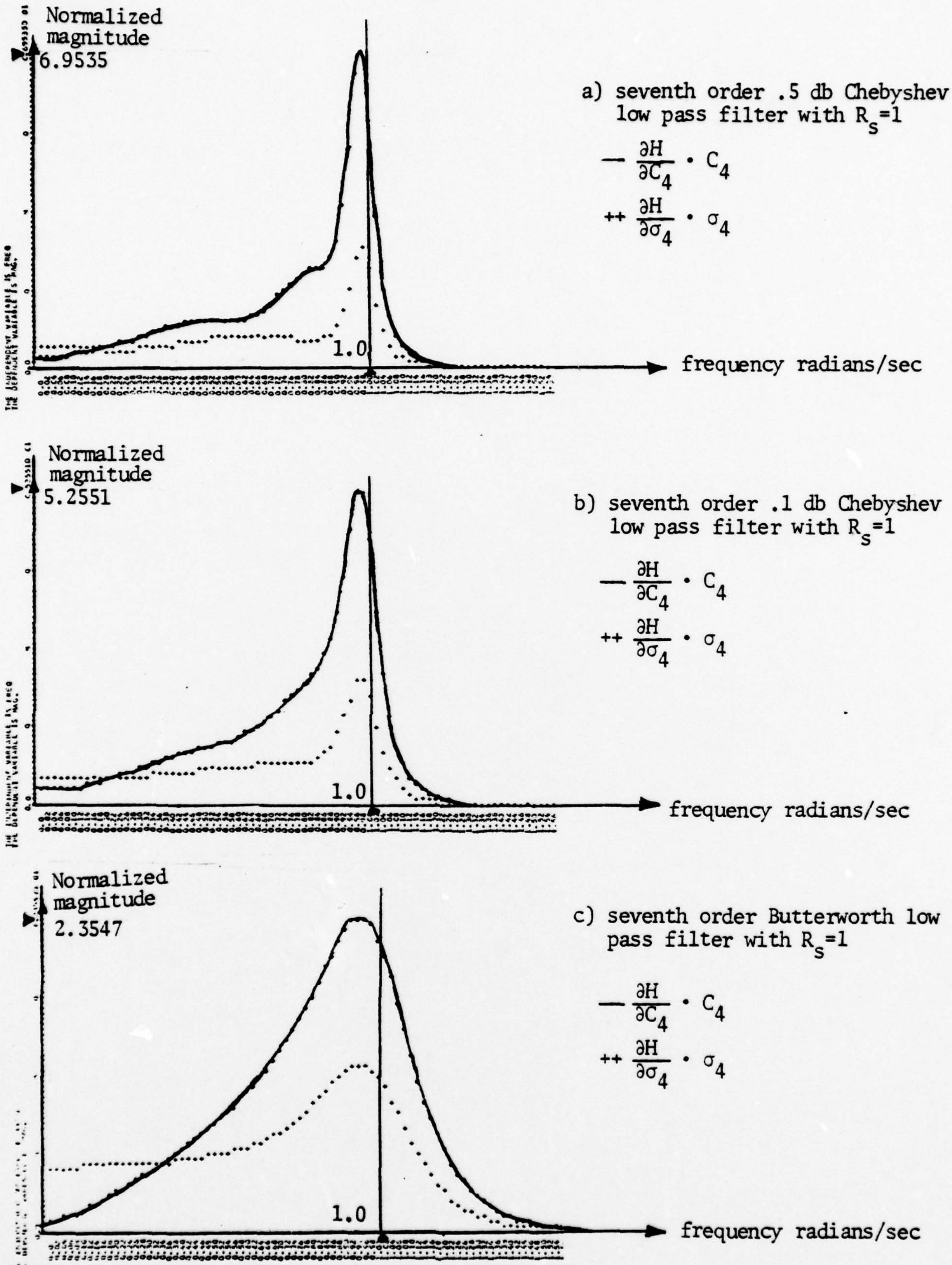


Fig. 6.9. The graphs of normalized sensitivity function, $\frac{\partial H}{\partial C_4} \cdot C_4$ and $\frac{\partial H}{\partial \sigma_4} \cdot \sigma_4$ of various simple wave digital filters.

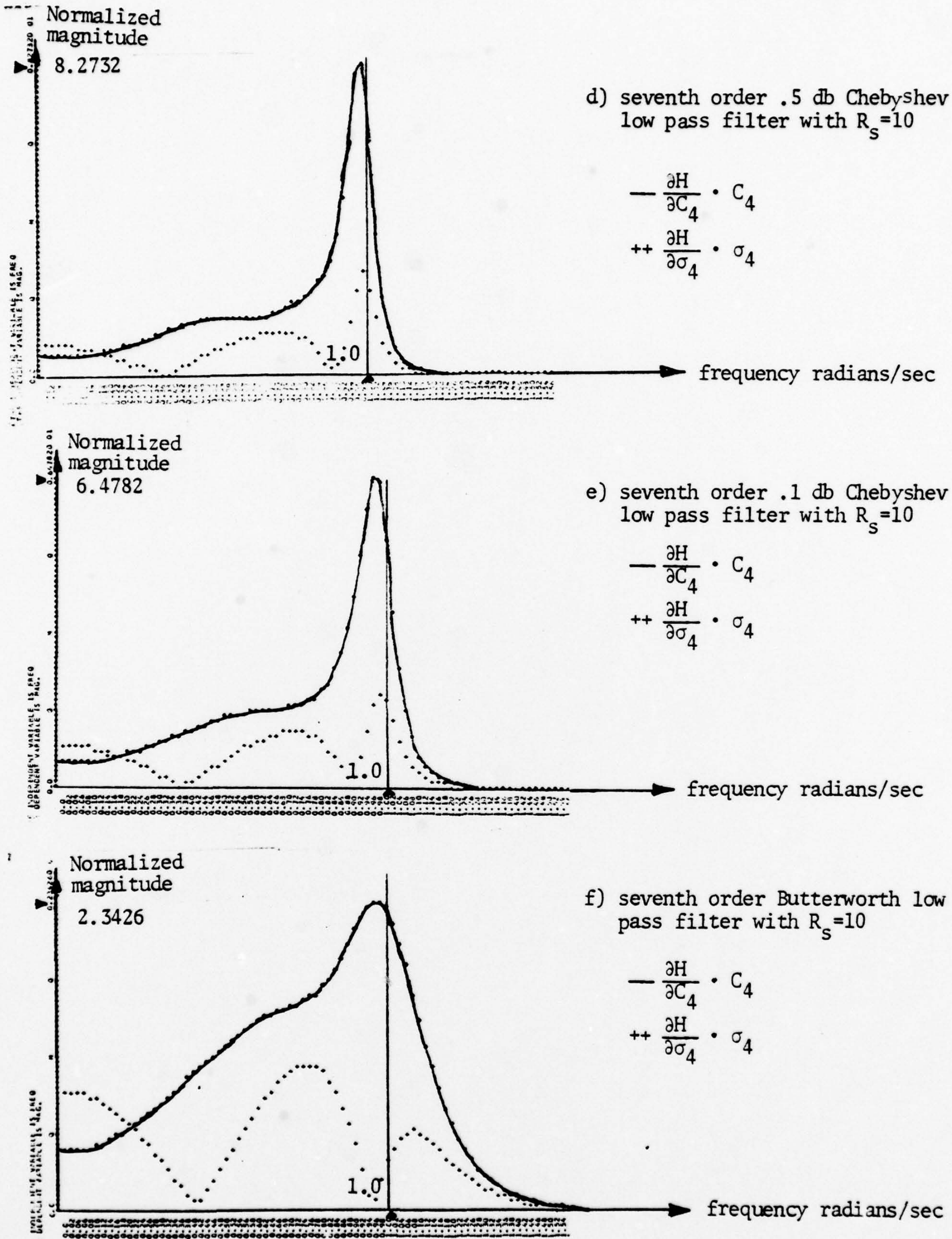


Fig. 6.9. Continued.

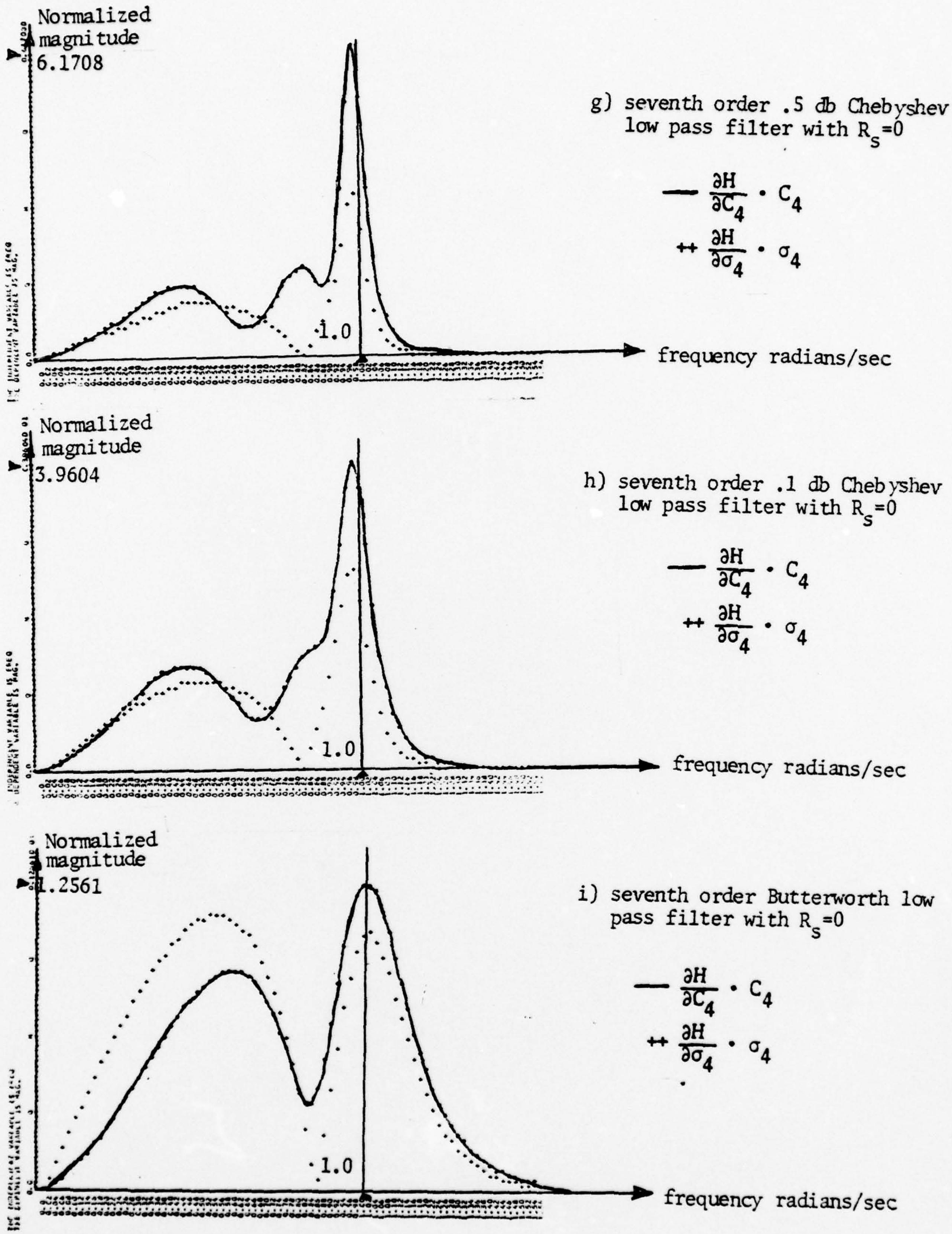


Fig. 6.9. Continued.

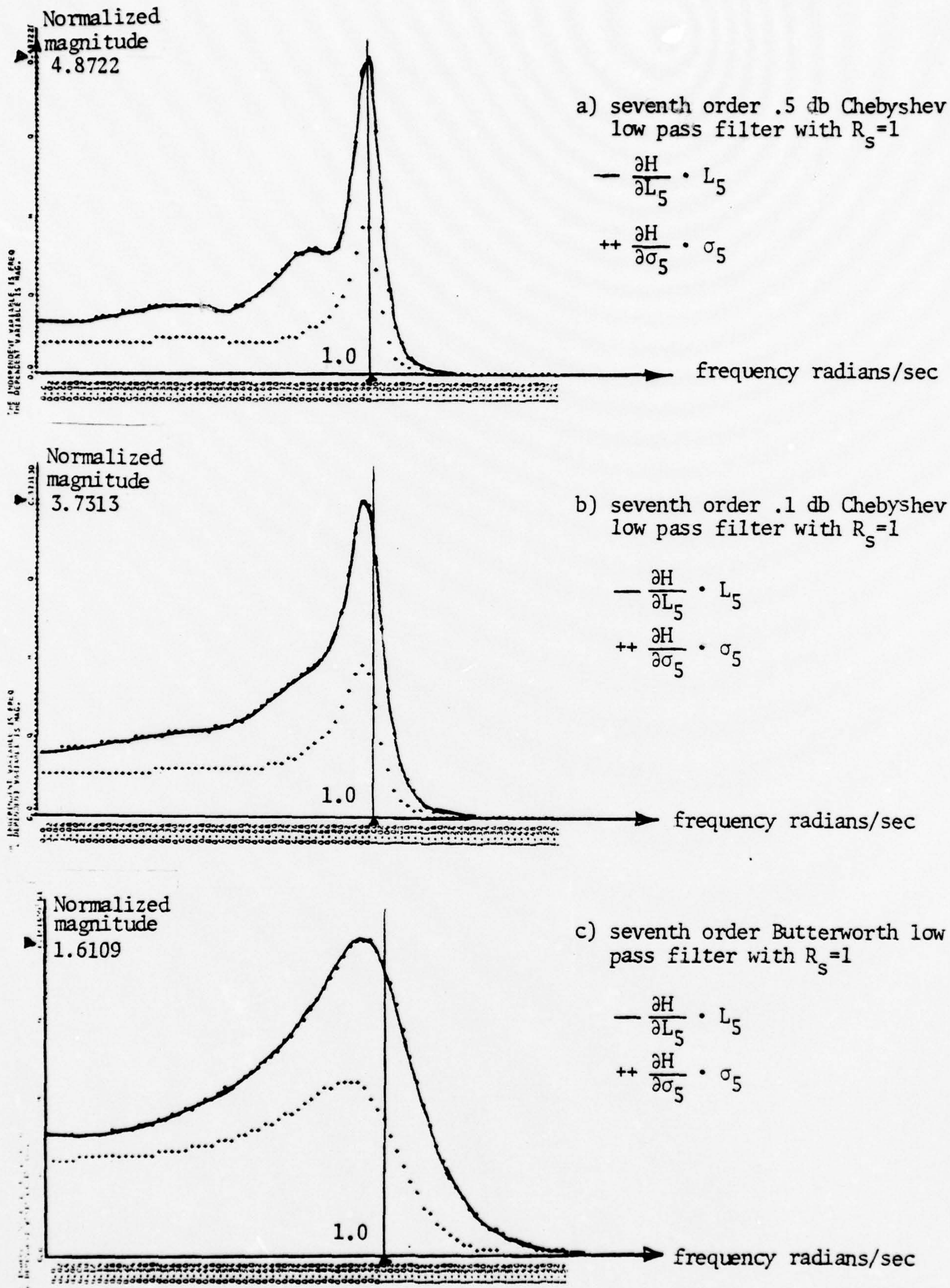


Fig. 6.10. The graphs of normalized sensitivity function, $\frac{\partial H}{\partial L_5} \cdot L_5$ and $\frac{\partial H}{\partial \sigma_5} \cdot \sigma_5$ of various simple wave digital filters.

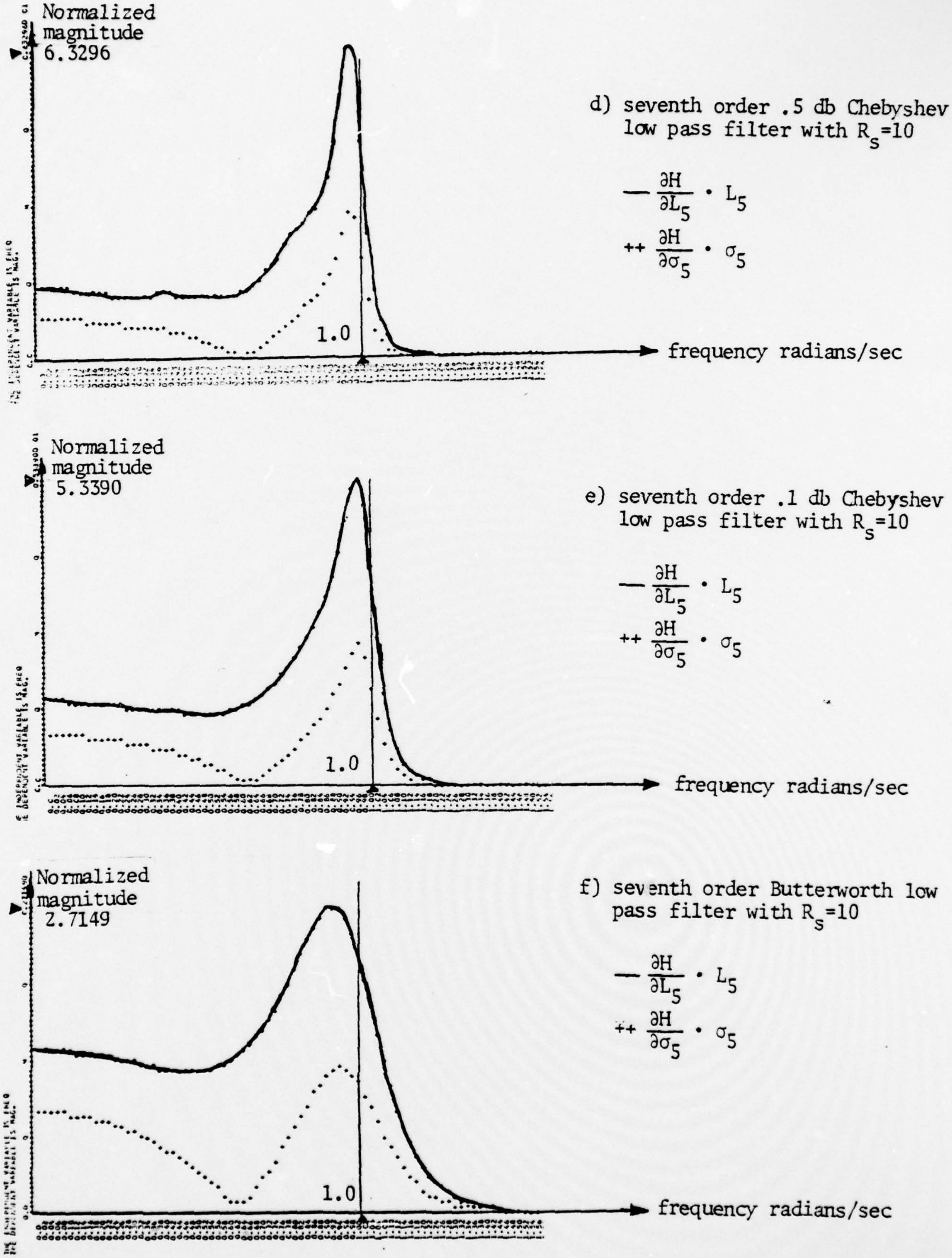
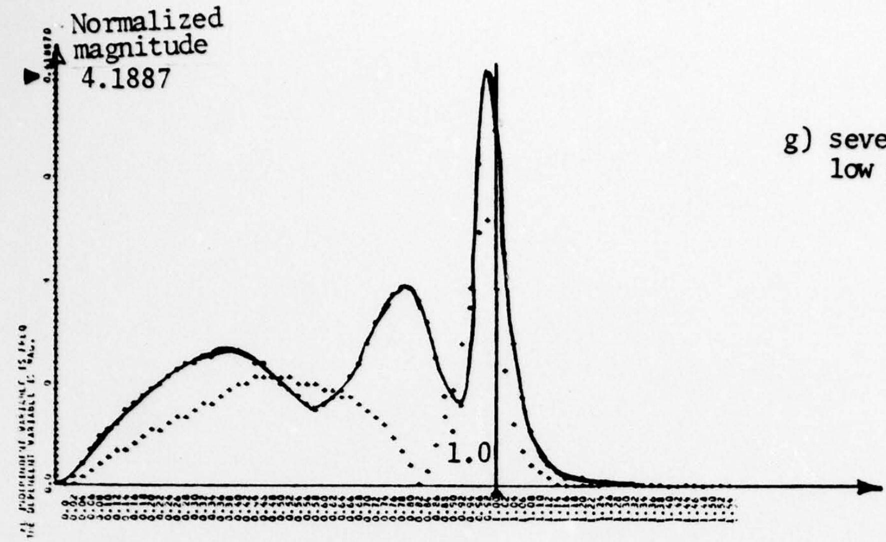


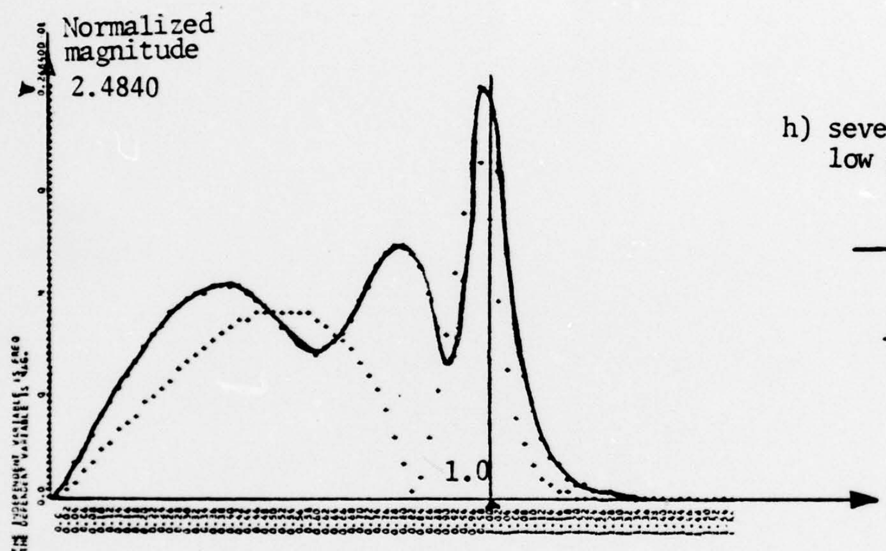
Fig. 6.10. Continued



g) seventh order .5 db Chebyshev low pass filter with $R_s=0$

— $\frac{\partial H}{\partial L_5} \cdot L_5$

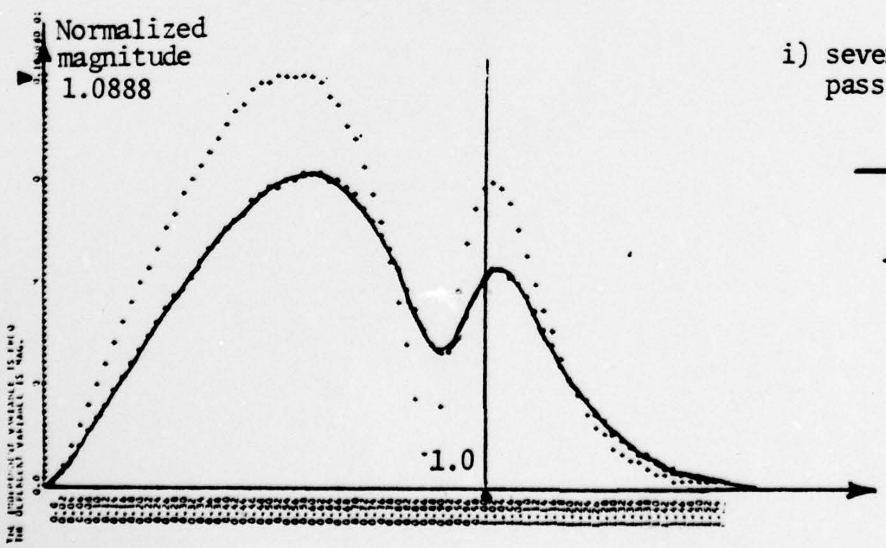
++ $\frac{\partial H}{\partial \sigma_5} \cdot \sigma_5$



h) seventh order .1 db Chebyshev low pass filter with $R_s=0$

— $\frac{\partial H}{\partial L_5} \cdot L_5$

++ $\frac{\partial H}{\partial \sigma_5} \cdot \sigma_5$



i) seventh order Butterworth low pass filter with $R_s=0$

— $\frac{\partial H}{\partial L_5} \cdot L_5$

++ $\frac{\partial H}{\partial \sigma_5} \cdot \sigma_5$

Fig. 6.10. Continued

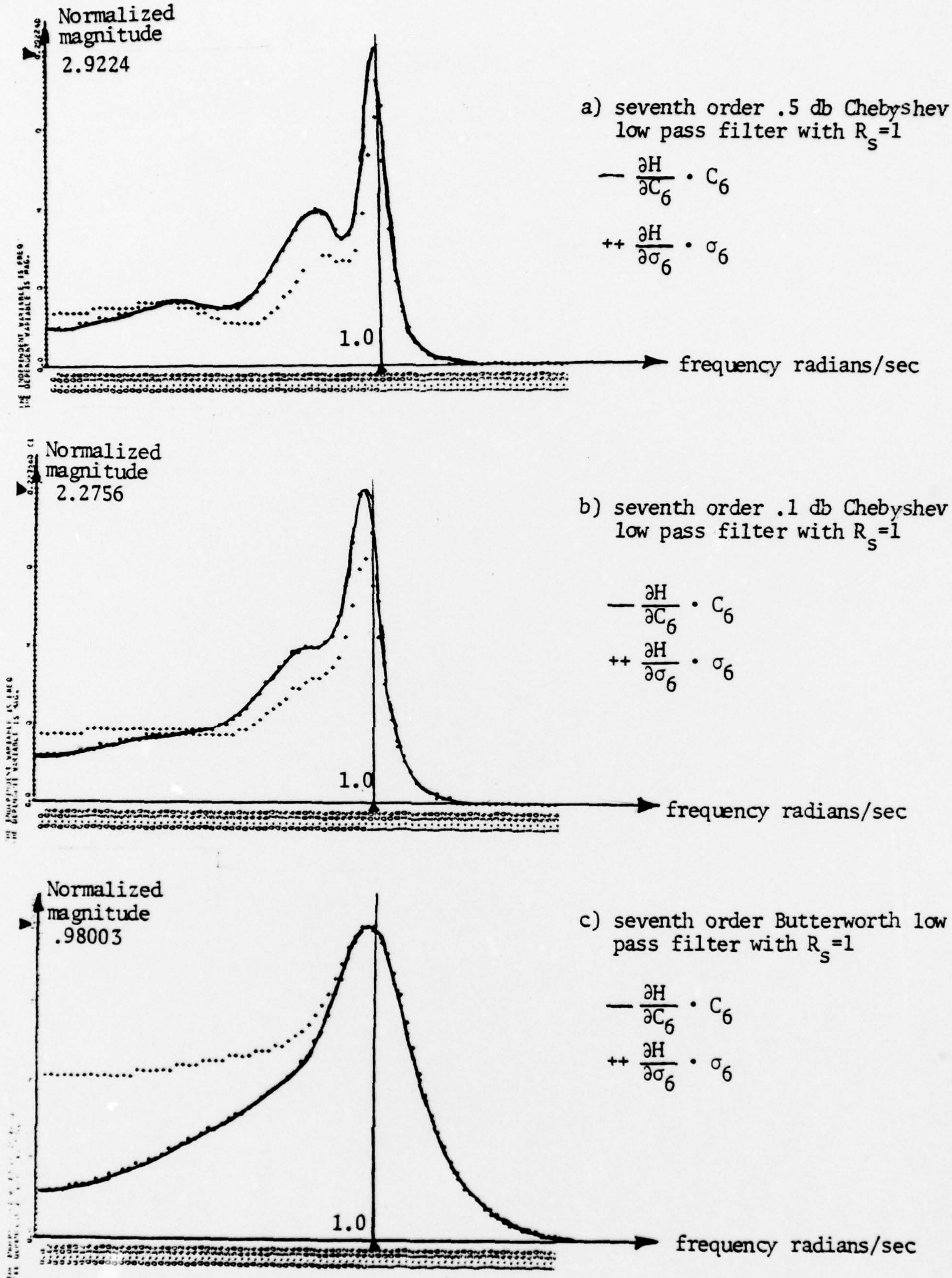


Fig. 6.11. The graphs of normalized sensitivity function, $\frac{\partial H}{\partial C_6} \cdot C_6$ and $\frac{\partial H}{\partial \sigma_6} \cdot \sigma_6$ of various simple wave digital filters.

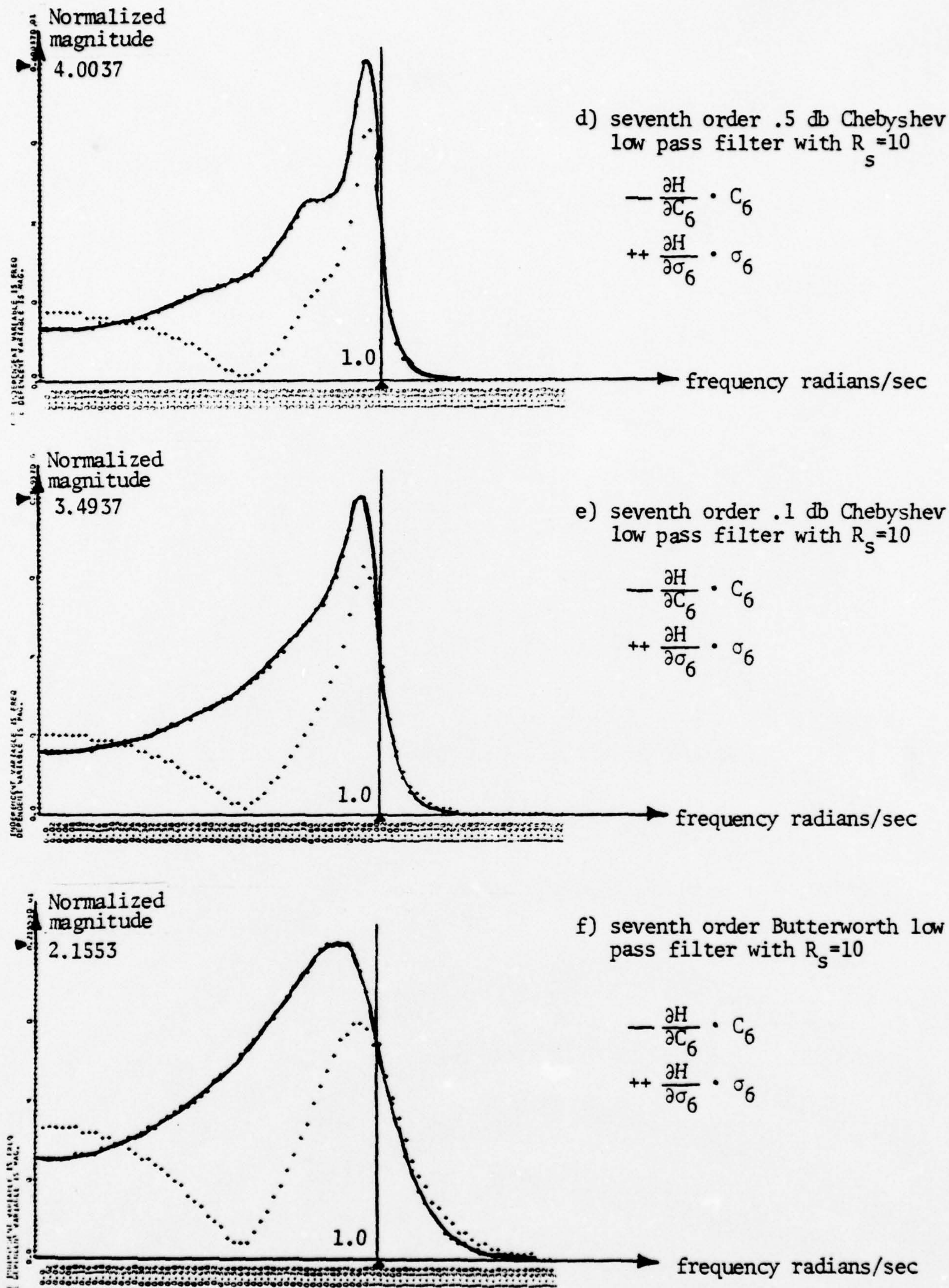
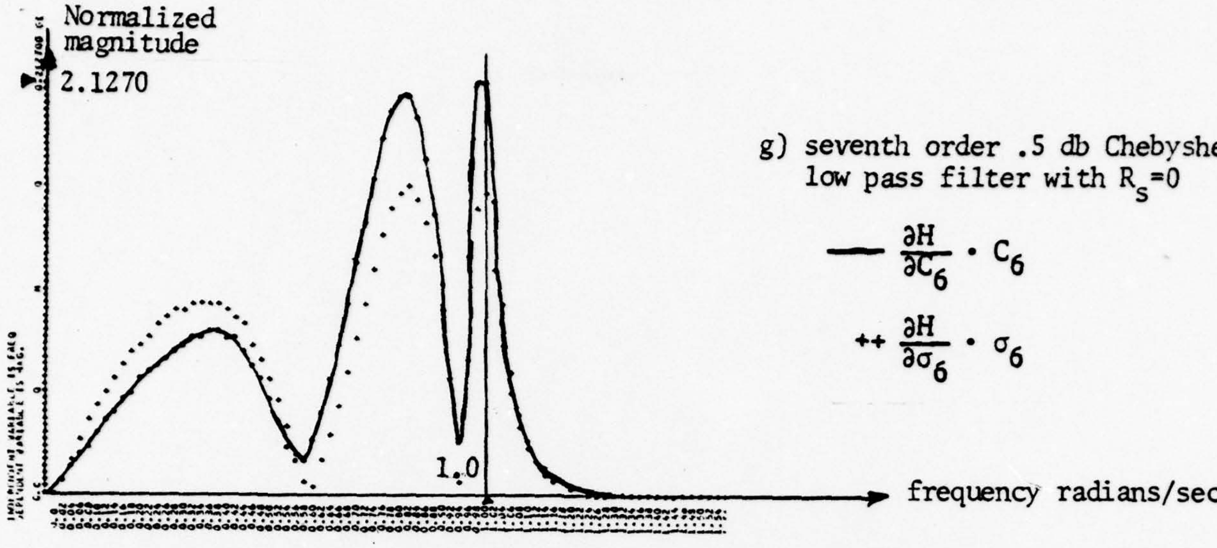


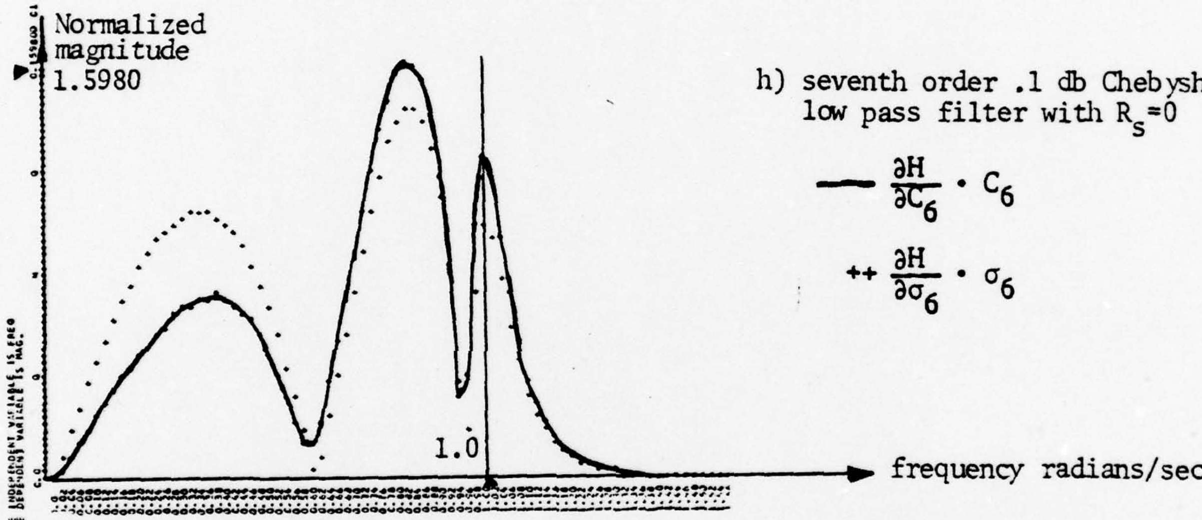
Fig. 6.11. Continued



g) seventh order .5 db Chebyshev low pass filter with R_s=0

— $\frac{\partial H}{\partial C_6} \cdot C_6$

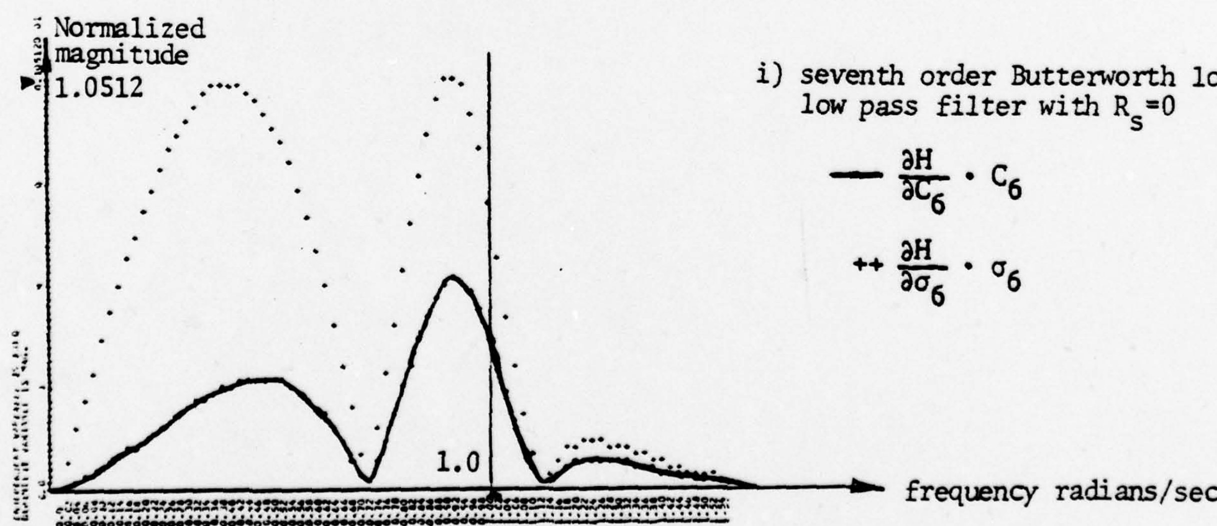
++ $\frac{\partial H}{\partial \sigma_6} \cdot \sigma_6$



h) seventh order .1 db Chebyshev low pass filter with R_s=0

— $\frac{\partial H}{\partial C_6} \cdot C_6$

++ $\frac{\partial H}{\partial \sigma_6} \cdot \sigma_6$

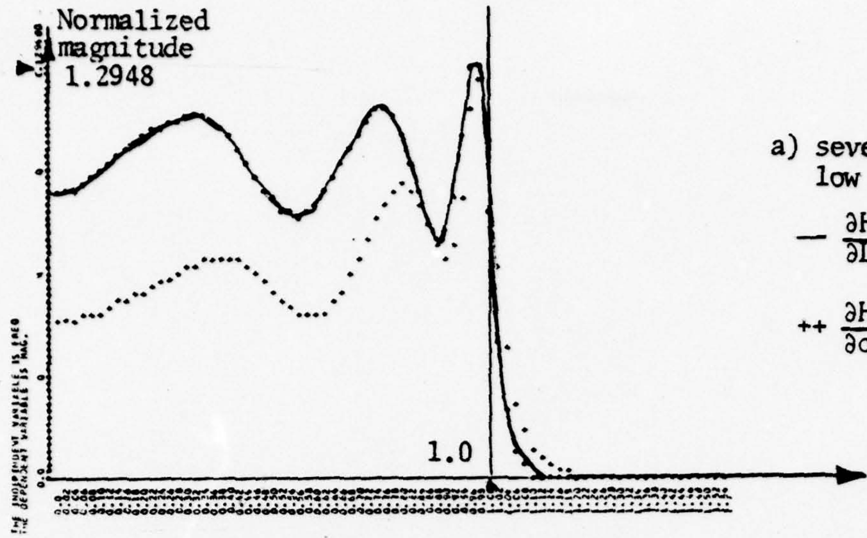


i) seventh order Butterworth low low pass filter with R_s=0

— $\frac{\partial H}{\partial C_6} \cdot C_6$

++ $\frac{\partial H}{\partial \sigma_6} \cdot \sigma_6$

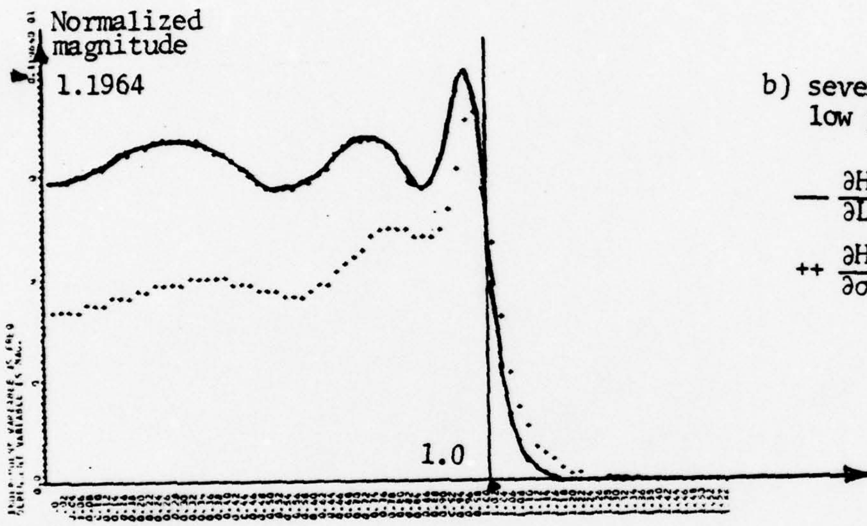
Fig. 6.11. Continued



a) seventh order .5 db Chebyshev low pass filter with R_s=1

— $\frac{\partial H}{\partial L_7} \cdot L_7$

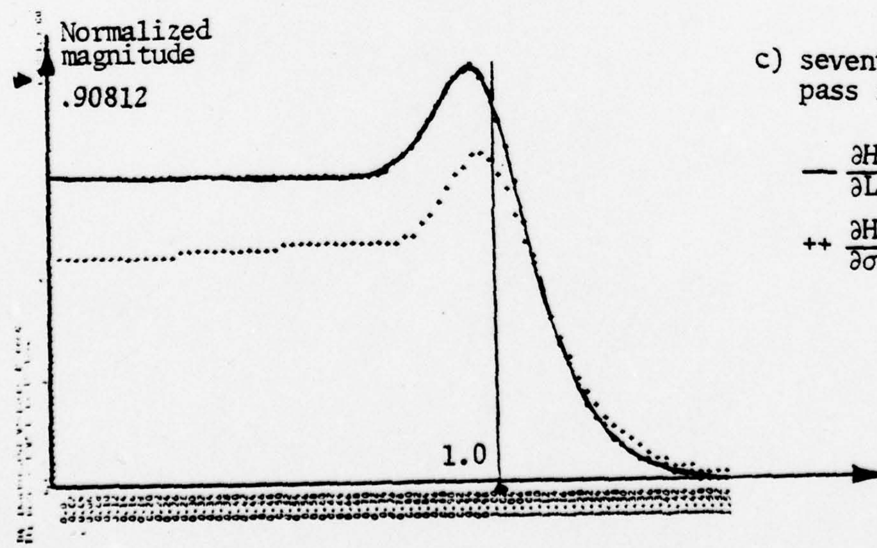
++ $\frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$



b) seventh order .1 db Chebyshev low pass filter with R_s=1

— $\frac{\partial H}{\partial L_7} \cdot L_7$

++ $\frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$

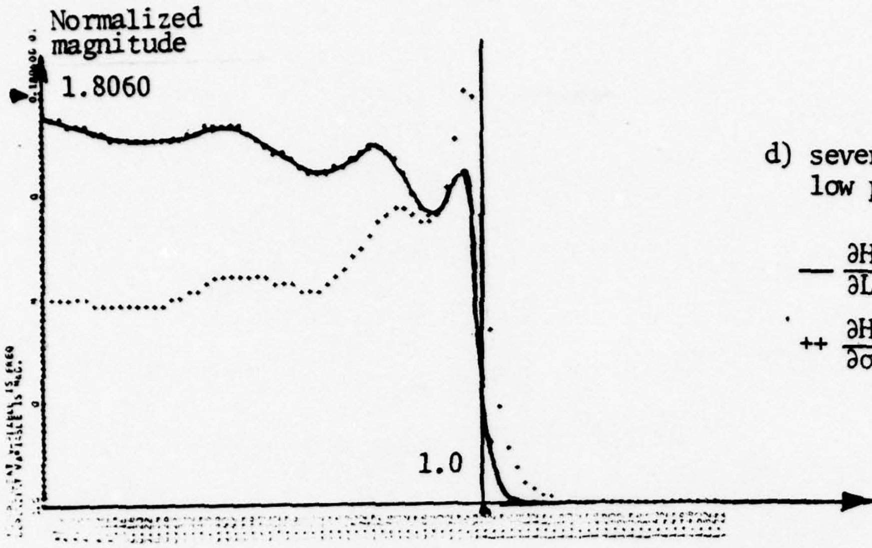


c) seventh order Butterworth low pass filter with R_s=1

— $\frac{\partial H}{\partial L_7} \cdot L_7$

++ $\frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$

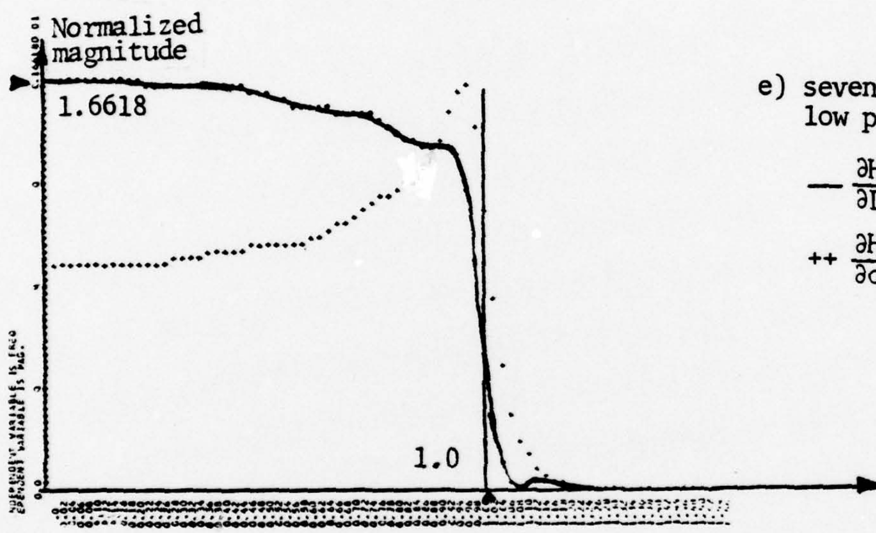
Fig. 6.12. The graphs of normalized sensitivity function, $\frac{\partial H}{\partial L_7} \cdot L_7$ and $\frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$ of various simple wave digital filters.



d) seventh order .5 db Chebyshev low pass filter with R_s=10

— $\frac{\partial H}{\partial L_7} \cdot L_7$

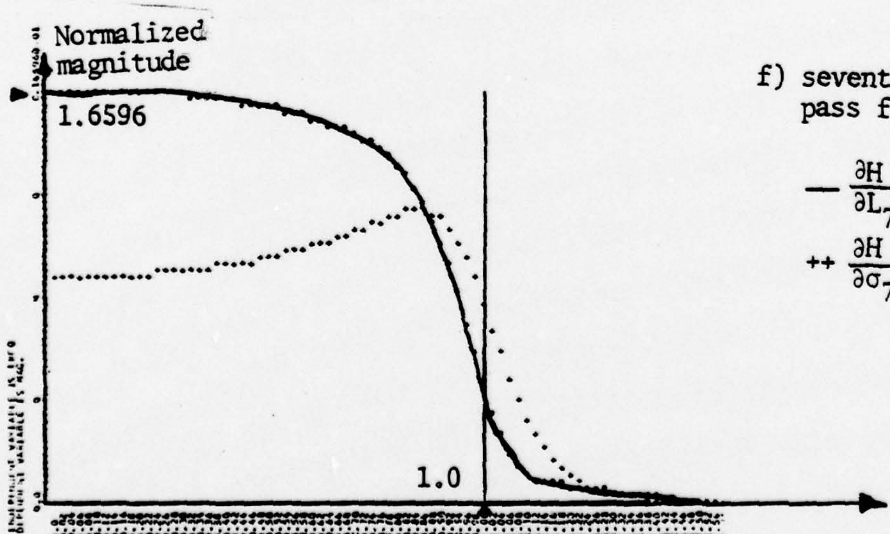
++ $\frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$



e) seventh order .1 db Chebyshev low pass filter with R_s=10

— $\frac{\partial H}{\partial L_7} \cdot L_7$

++ $\frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$

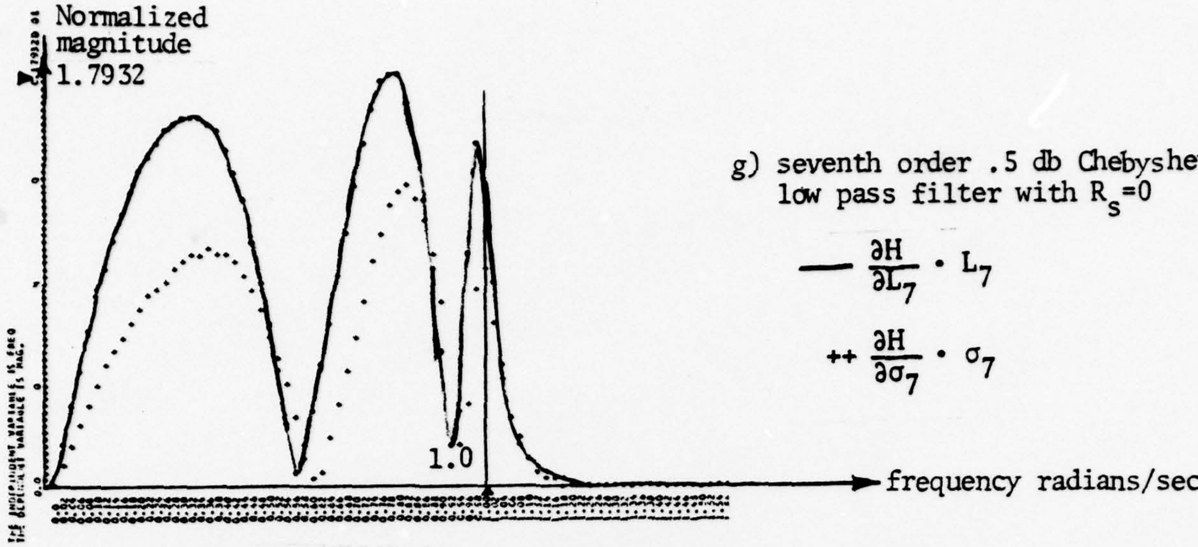


f) seventh order Butterworth low pass filter with R_s=10

— $\frac{\partial H}{\partial L_7} \cdot L_7$

++ $\frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$

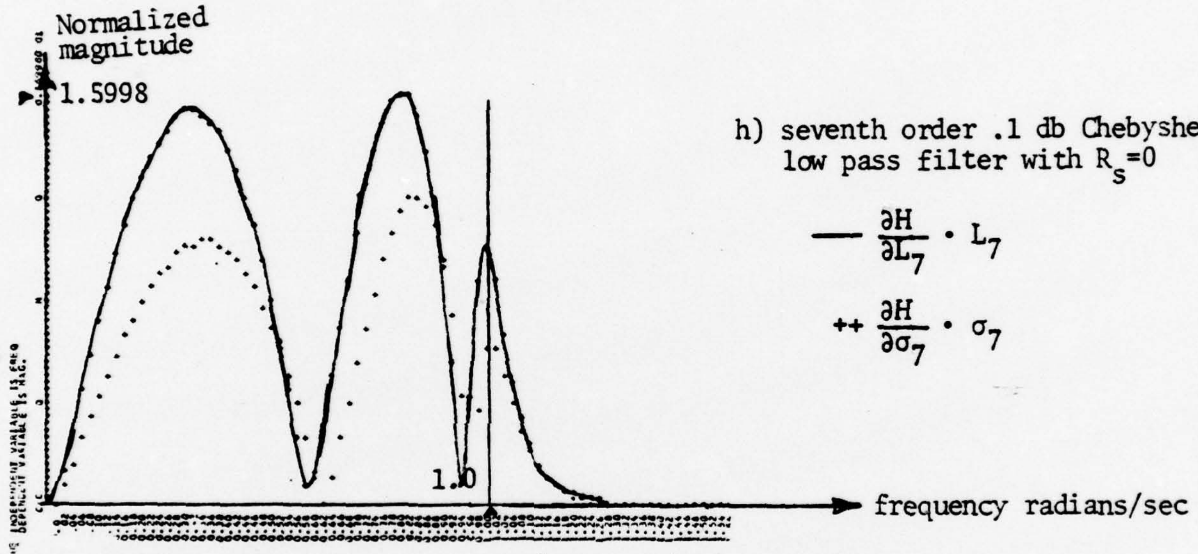
Fig. 6.12. Continued



g) seventh order .5 db Chebyshev low pass filter with $R_s=0$

— $\frac{\partial H}{\partial L_7} \cdot L_7$

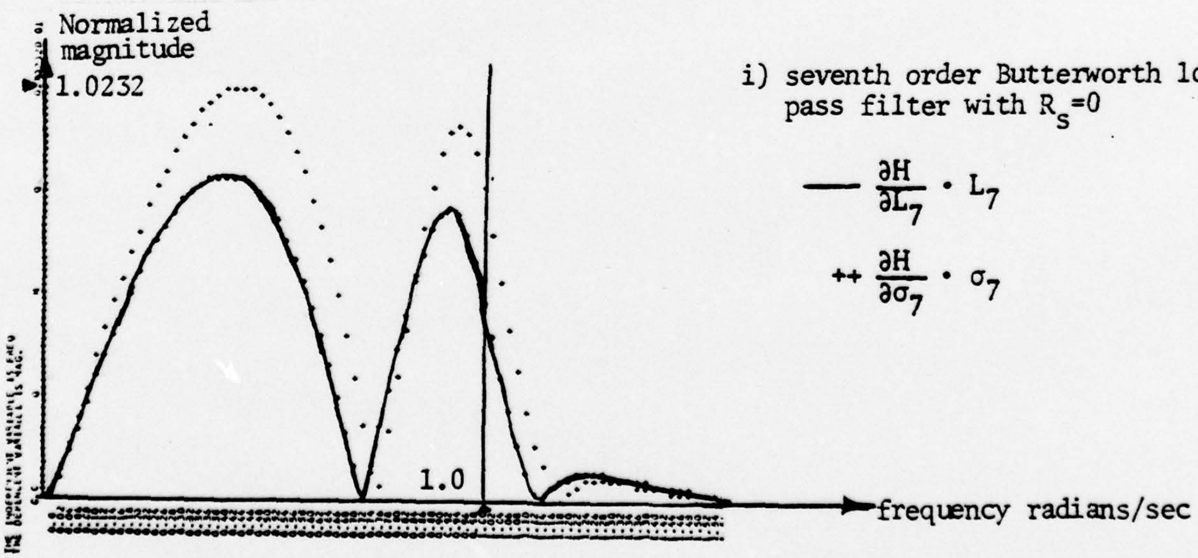
++ $\frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$



h) seventh order .1 db Chebyshev low pass filter with $R_s=0$

— $\frac{\partial H}{\partial L_7} \cdot L_7$

++ $\frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$



i) seventh order Butterworth low pass filter with $R_s=0$

— $\frac{\partial H}{\partial L_7} \cdot L_7$

++ $\frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$

Fig. 6.12. Continued

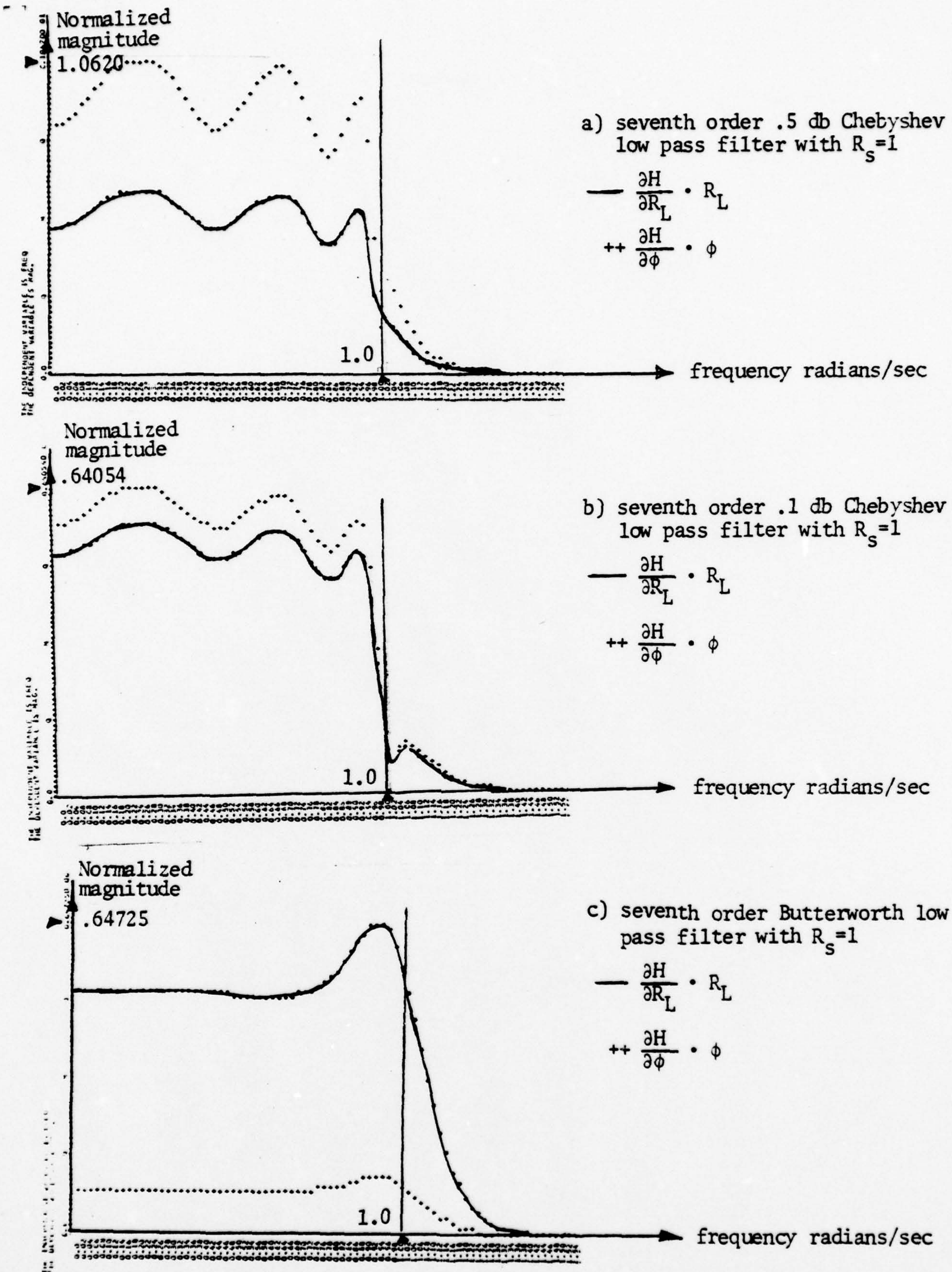
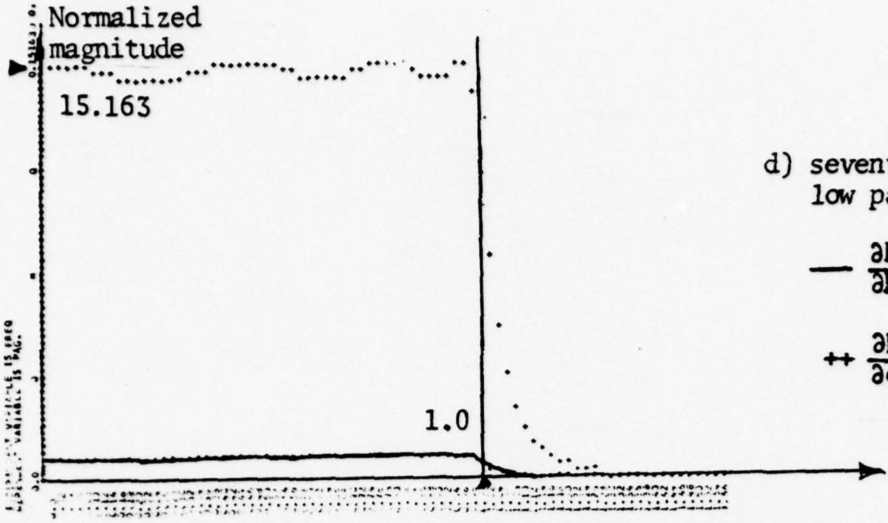


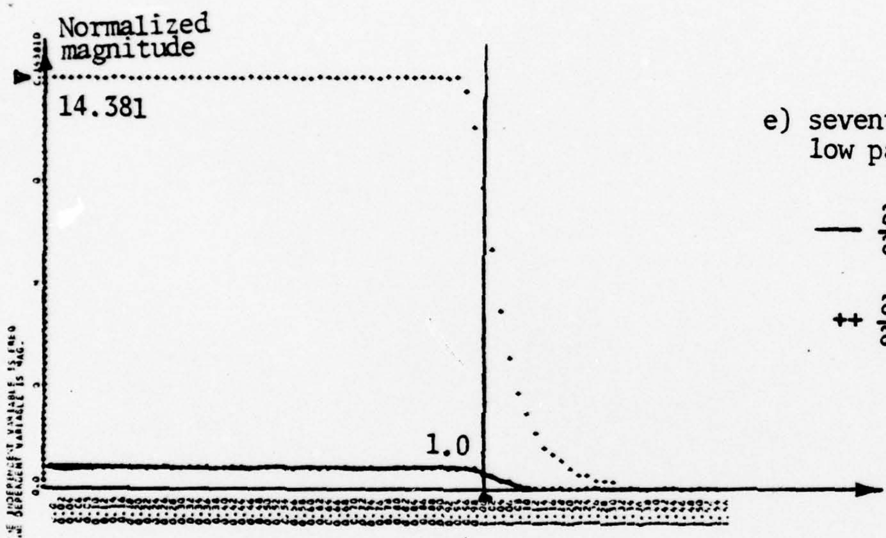
Fig. 6.13. The graphs of normalized sensitivity function, $\frac{\partial H}{\partial R_L} \cdot R_L$ and $\frac{\partial H}{\partial \phi} \cdot \phi$ of various simple wave digital filters.



d) seventh order .5 db Chebyshev low pass filter with $R_s=10$

— $\frac{\partial H}{\partial R_L} \cdot R_L$

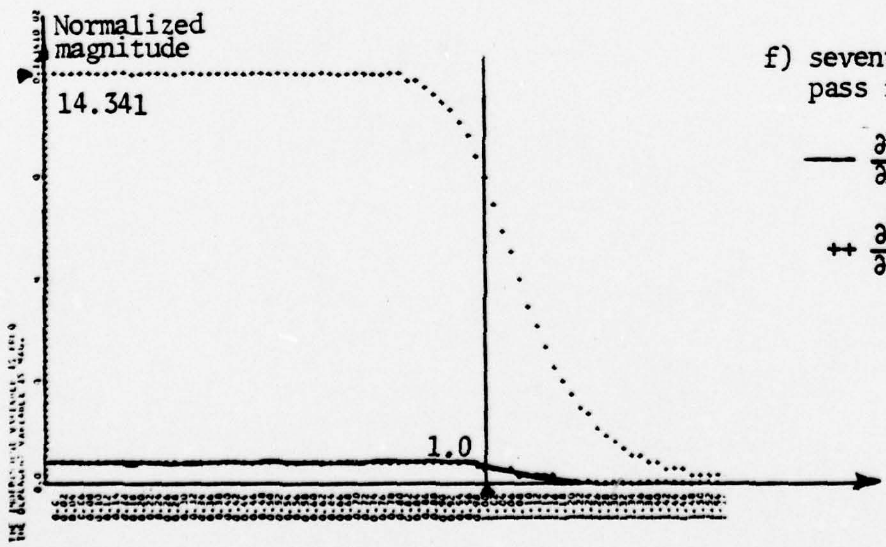
++ $\frac{\partial H}{\partial \phi} \cdot \phi$



e) seventh order .1 db Chebyshev low pass filter with $R_s=10$

— $\frac{\partial H}{\partial R_L} \cdot R_L$

++ $\frac{\partial H}{\partial \phi} \cdot \phi$

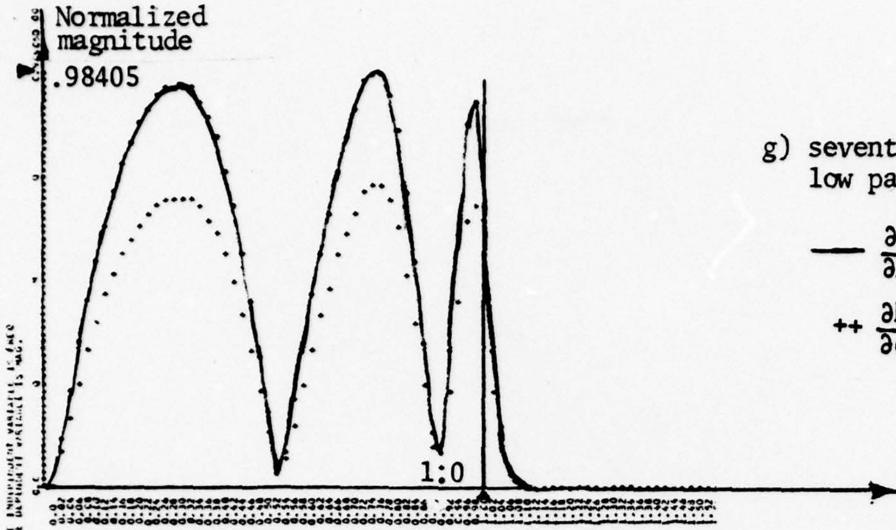


f) seventh order Butterworth low pass filter with $R_s=10$

— $\frac{\partial H}{\partial R_L} \cdot R_L$

++ $\frac{\partial H}{\partial \phi} \cdot \phi$

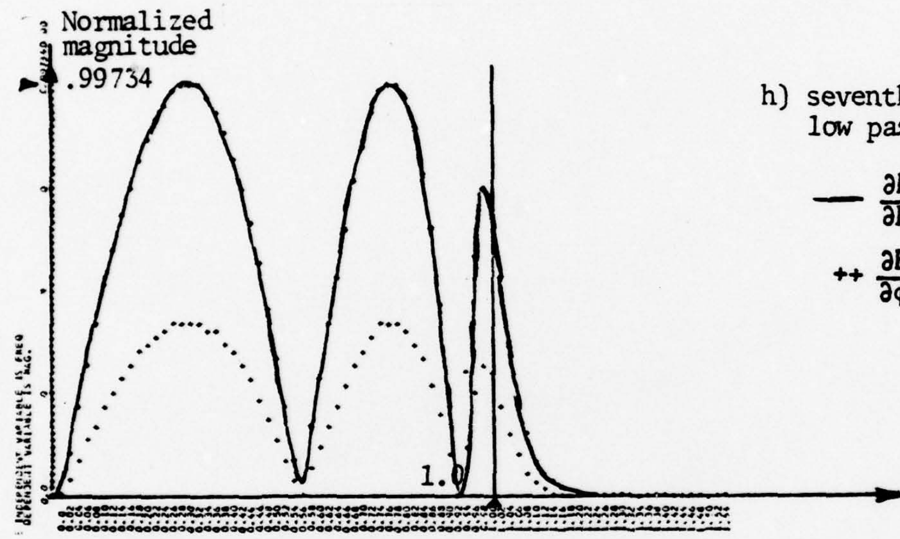
Fig. 6.13. Continued



g) seventh order .5 db Chebyshev low pass filter with $R_s=0$

—— $\frac{\partial H}{\partial R_L} \cdot R_L$

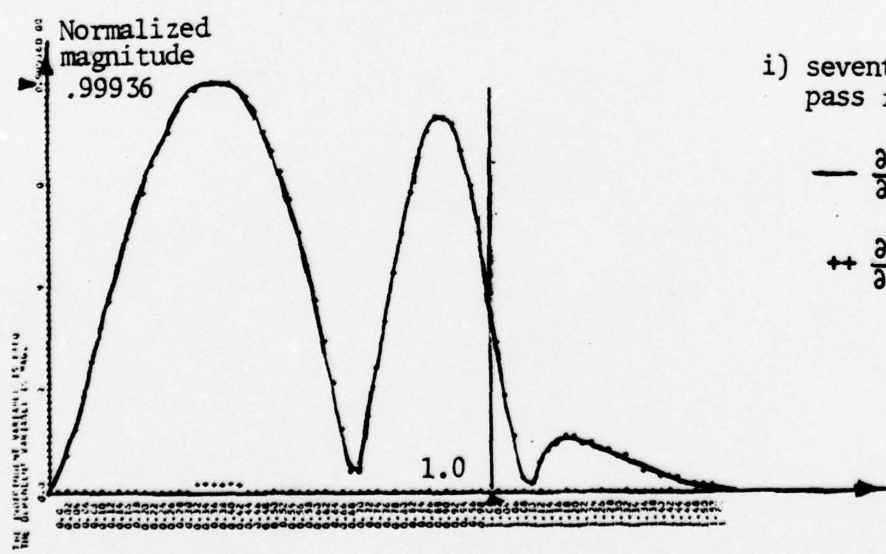
++ $\frac{\partial H}{\partial \phi} \cdot \phi$



h) seventh order .1 db Chebyshev low pass filter with $R_s=0$

—— $\frac{\partial H}{\partial R_L} \cdot R_L$

++ $\frac{\partial H}{\partial \phi} \cdot \phi$



i) seventh order Butterworth low pass filter with $R_s=0$

—— $\frac{\partial H}{\partial R_L} \cdot R_L$

++ $\frac{\partial H}{\partial \phi} \cdot \phi$

Fig. 6.13. Continued

in the frequency domain. Thus it is expected that the product of the sensitivity study of this chapter to conform to the bit truncation study of the Chapter V. It is important to emphasize that in this chapter only the internal sensitivity behavior of the simple wave digital filters is considered.

Analysis of the normalized partial sensitivity function curves of Figs. 6.5, 13 reveal the following.

1) Both sensitivity functions, i.e. sensitivity due to wave digital filter multiplier coefficients and sensitivity due to the original wave digital components peak at the critical frequency in all cases. This can be expected, since it is well known that the digital filters in general exhibit high sensitivities around the critical frequency of the filter. The critical frequency of the filters under investigation was 1 radian/sec and this point has been marked by a vertical line on all the sensitivity graphs in Figs. 6.5, 13.

2) In general the normalized sensitivity due to multiplier coefficients are smaller than sensitivity due to original filter components for the first sections. Gradually this difference gets smaller and for some few cases in the last section it reverses. This is reasonable since it can be expected that earlier multiplier coefficients will have lower sensitivity than the coefficients associated with the later sections.

3) At low frequencies, the normalized sensitivity of the wave digital filters due to multiplier coefficients are slightly larger than the sensitivity of the filter to the original filter components. This effect reverses in the mid and high frequencies.

4) The sensitivity function of the wave digital filter with respect to the source resistance increases with increasing source resistance, as evident from Figs. 6.5, 13.

5) The sensitivity with respect to terminating load resistance and reflection coefficient have exactly the same pattern as evident from equation (6.60). Apart from this fact it is important to note that both sensitivity functions tend to follow the frequency characteristic of the original filter. This can be seen from the graph 6.13 for the case $R_s = 10$. As R_s decreases the fluctuations in the sensitivity curve increases while following the same pattern. In fact for $R_s=0$ these fluctuations increase to such an extent that at some frequencies the sensitivity becomes negligible. This phenomenon is also evident in the multiplier coefficients of the last few sections of the wave digital filter but to a lesser extent.

6) In general the sensitivity of the wave digital filter due to variations in multiplier coefficients is lower than the sensitivity of the wave digital filter with respect to the original component values.

7) Also it is worthwhile to note that the high sensitivities observed for high source resistances in wave digital filters algorithms with no delay free path in port two are in complete agreement with the results obtained in Chapter V.

References

1. Anatol I. Zverev, Handbook of Filter Synthesis, 1967, published by John Wiley and Sons, Inc.

VII. CONCLUSION

A. INTRODUCTORY REMARKS

It is important to note that in this thesis, whenever appropriate, detailed conclusions and discussions of the results are made in the chapters concerned. Thus it would be repetitious to state these conclusions again. However for completeness, the highlights of important conclusions based on experimental results are summarized here. It is also important to note that in the experimental studies of this thesis, in order to achieve reliable and accurate results, a large number of filters of different types with different termination source resistances were studied, and the conclusions made on the basis of collective results.

B. SUMMARY OF THE IMPORTANT RESULTS IN THE FREQUENCY DOMAIN BEHAVIOR OF THE WAVE DIGITAL FILTER

1 - Digital filters, derived from doubly terminated LC analogue filters using the bilinear transform, have the lowest sensitivity of frequency response for variation of the original L, C, R_S and R_L parameters of the algorithms tested.

2 - Wave digital filters, derived from doubly terminated LC analogue structures, if designed properly tend to achieve exactly the same low sensitivity as that of the above conventional digital filters.

3 - Design of conventional digital filters from LC structures is relatively a tedious job, while designing the wave digital filters from LC structures is relatively simple.

4 - Wave digital filters exhibit high sensitivity to termination resistance values. Thus in the design of wave digital filters, in order to achieve a desired performance, the delay free loop should be chosen at the low impedance termination of the filter.

5 - Due to the internal structure of the wave digital filter composed of sections with multiple LC resonant elements, these filters exhibit higher sensitivity than the same filter made up of sections with simple elements.

6 - The sensitivities of the internal sections of the wave digital filter, both with respect to filter multiplier coefficients and original filter RLC components, tend to peak sharply at the critical frequency of the wave digital filter.

7 - The sensitivities of the internal sections of the wave digital filter increases towards the load end.

8 - For the seventh order low pass wave digital filters studied, the slope of the rms error due to quantization in the number of bits of the multiplier coefficients versus the number of bits is approximately 3 db per bit. Also, interestingly, the slope of the rms error of the conventional seventh order digital filter is also approximately 3 db per bit.

C. SUMMARY OF NEW THEORETICAL EXTENSIONS TO WAVE DIGITAL FILTER THEORY

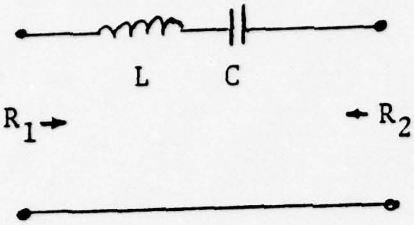
1 - Earlier researchers [1] and [2] have stated as a typical example that for the port two resistance of the wave digital section designed for series L with no delay free path in port two, one should use $R_2 = R_1 + L$. This formula does not take into account the effect of sampling time. In the theory developed in this

thesis, following a derivation parallel to the one given in [1] and [2], the port two resistance for the said section is derived as $R_2 = R_1 + \frac{2L}{T}$. This result is more general than the previous one, for which an inherent fixed sampling time of one second must always be assumed.

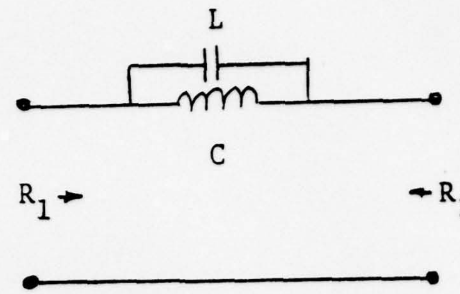
2 - In the theory developed in this thesis for the design of wave digital subsections, only one algorithm is derived for a multiple LC resonant section. This result is applicable for both series or shunt elements. This new approach makes it possible to design several alternate wave digital filter algorithms, for a given analogue LC filter structure.

D. SUGGESTED FUTURE RESEARCH

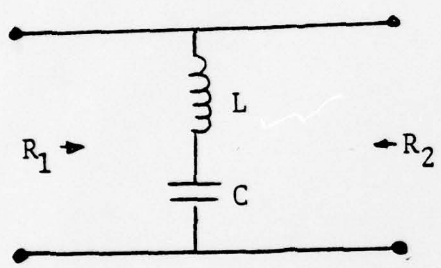
1 - In the analysis of this thesis we have established that the simple wave digital filter, due to its inherent ladder-like structure, exhibits lower sensitivities to multiplier truncation than the complex cascaded section wave digital filter. Thus the need for the design of simple wave digital filters, even for band pass or band stop applications, arises. In the literature all the present algorithms for band pass or band stop wave digital filters are of the complex cascaded multiple LC element type. However it is possible to design simple wave digital filter, even for band pass or band stop applications using simple sections, so as to reduce the sensitivity of the overall structure. The technique basically is as developed in Chapter IV. The suggested outline is as follows. The analogue band pass or band stop LC filter derived from the low pass analogue LC filter would have sections of the type shown in Fig. 7.1. The design



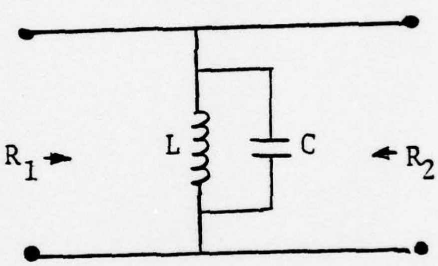
a) two port network with two L and C elements in the total series configuration



b) two port network with parallel L and C in the total series configuration



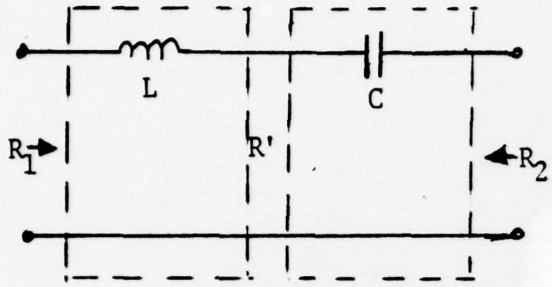
c) two port network with series L and C in the total parallel configuration



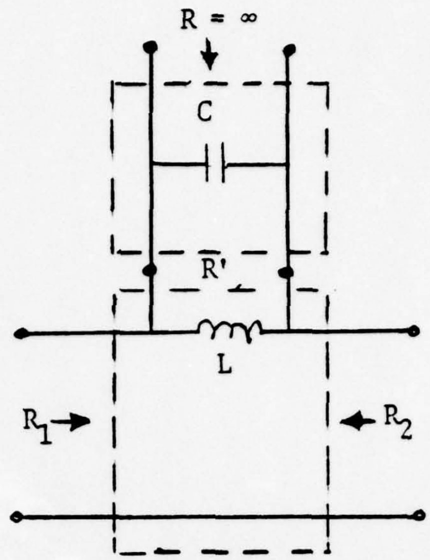
d) two port network with parallel L and C in the total parallel configuration

Fig. 7.1. Total combination of L and C in a two port network.

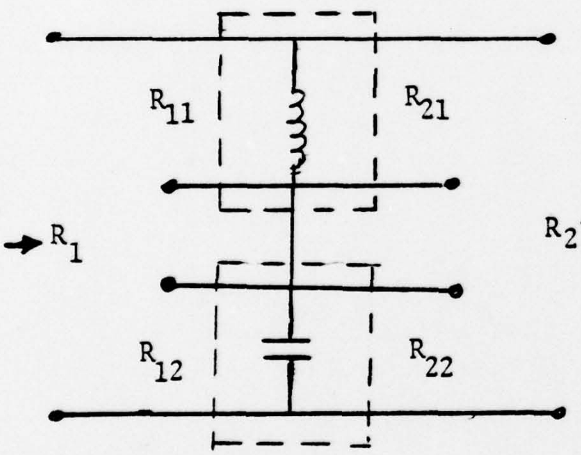
of simple wave digital filter for Figs. 7.1a and 7.1d is no problem and they can be separated into two port simple sections as shown in Figs. 7.2a and 7.2d. Thus we have to discuss only the cases of Figs. 7.1b and 7.1c. To design a simple wave digital algorithm for Fig. 7.1b we must employ a three port wave flow network. A general three port wave flow network is shown in Fig. 7.3. Note that in this figure there is a feedback path from all inputs to all outputs. Fig.7.2b reveals that in order to design two simple wave digital sections we have to employ a three port signal flow graph. Either L or C of Fig. 7.2b can be adapted into a three port structure. Consider element L as the three port element. It can be shown that in order to have a causal network, two of the three ports must have no delay-free feedback path. We can make port-two and port-three of Fig. 7.3 with no delay free feedback. In doing so we can find R_2 in terms of R_1, R_3 and L. Also we can find R_3 in terms of R_1, R_2 and L. Thus we have two equations and two unknowns and we can solve for R_2 and R_3 in terms of R_1 and L. Now we can cascade a two port network designed for the component C into port three, and also cascade the succeeding sections into port two. It must be emphasized that the two port network designed for element C has no termination resistance on its second port; thus it is open circuited and its reflection coefficient in port two will be equal to 1. To design a simple section wave digital filter for Fig. 7.1c, we have to use a somewhat different approach. With the concept of delay free feedback in mind and with reference to Fig. 7.4, we can design a two port network



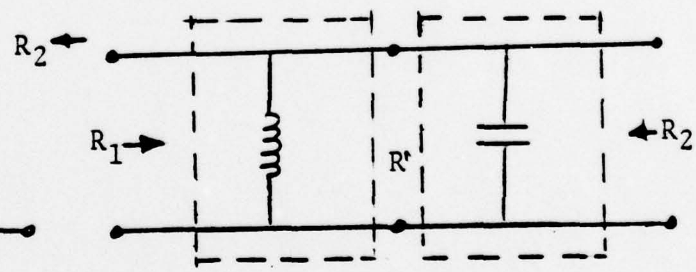
a) two cascaded two ports



b) three port element connected to the two port element



c) two series two ports in shunt configuration



d) two cascaded two ports

Fig. 7.2. Separation of the complex two port networks shown in Fig. 7.1 into simple two or three port networks.

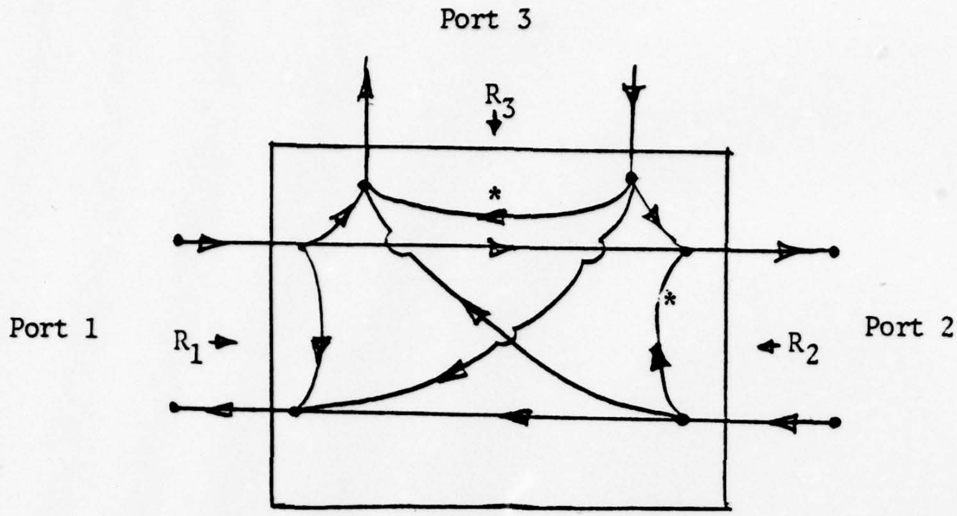


Fig. 7.3. A three port signal flow graph, with all possible combinations of delay free inputs and outputs. Note that the delay free path in port two and port three are marked with *.

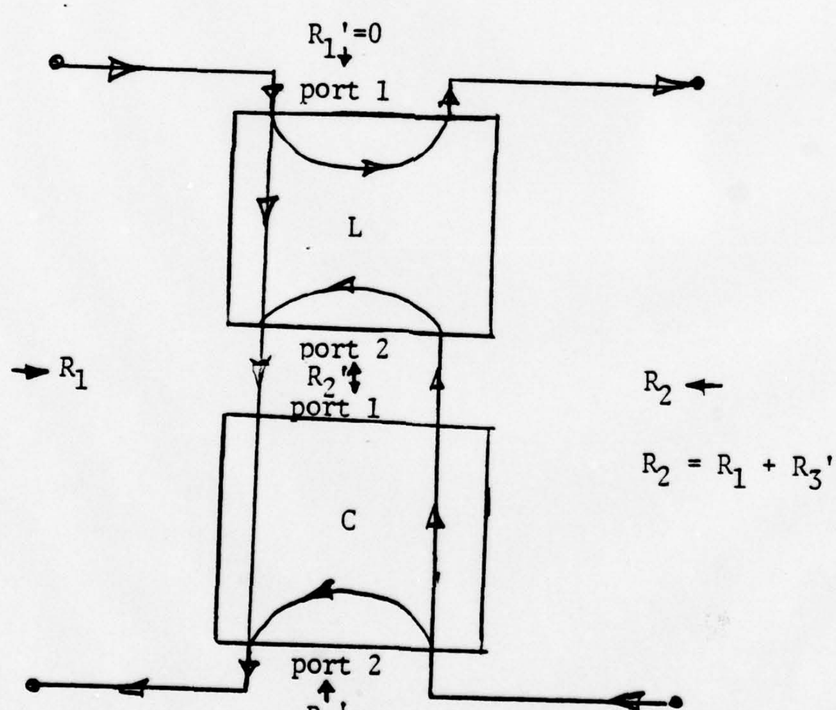


Fig. 7.4. Simple section wave digital filter signal flow graph of the complex series, shunt network of Fig. 7.1c.

for the element L such that there is no delay free feedback from port two (input) to port one (output) for subelement L. From this constraint, we can find R_2', with R_1' arbitrarily made equal to zero. The signal flow graph for the element C then is a normal two port signal flow graph with no delay free path in port one. It is fairly obvious that port two of the composite section will have a termination resistance of $R_2 = R_1 + R_3'$ where R_3' is the port two resistance of the element C.

2 - In the analysis of the internal structure of Chapter VI, we found the sensitivities of the wave digital filter with respect to the original component values, i.e. L's, C's, R_S, and R_L. It is interesting, for a given filter, to differentiate the conventional digital filter transfer function with respect to components L, C, R_S, and R_L and compare these sensitivities to those of the wave digital filter obtained in Chapter VI.

3 - The sensitivity functions of the wave digital filter with respect to filter multiplier coefficients can be used to implement an adaptive wave digital filter, by feeding back a weighted percentage of the sensitivities into the input as shown schematically in Fig. 7.5. The adaptivity can be done using any of the known optimization methods available in the literature, namely gradient optimization techniques, Fletcher-Powell optimization techniques, etc.

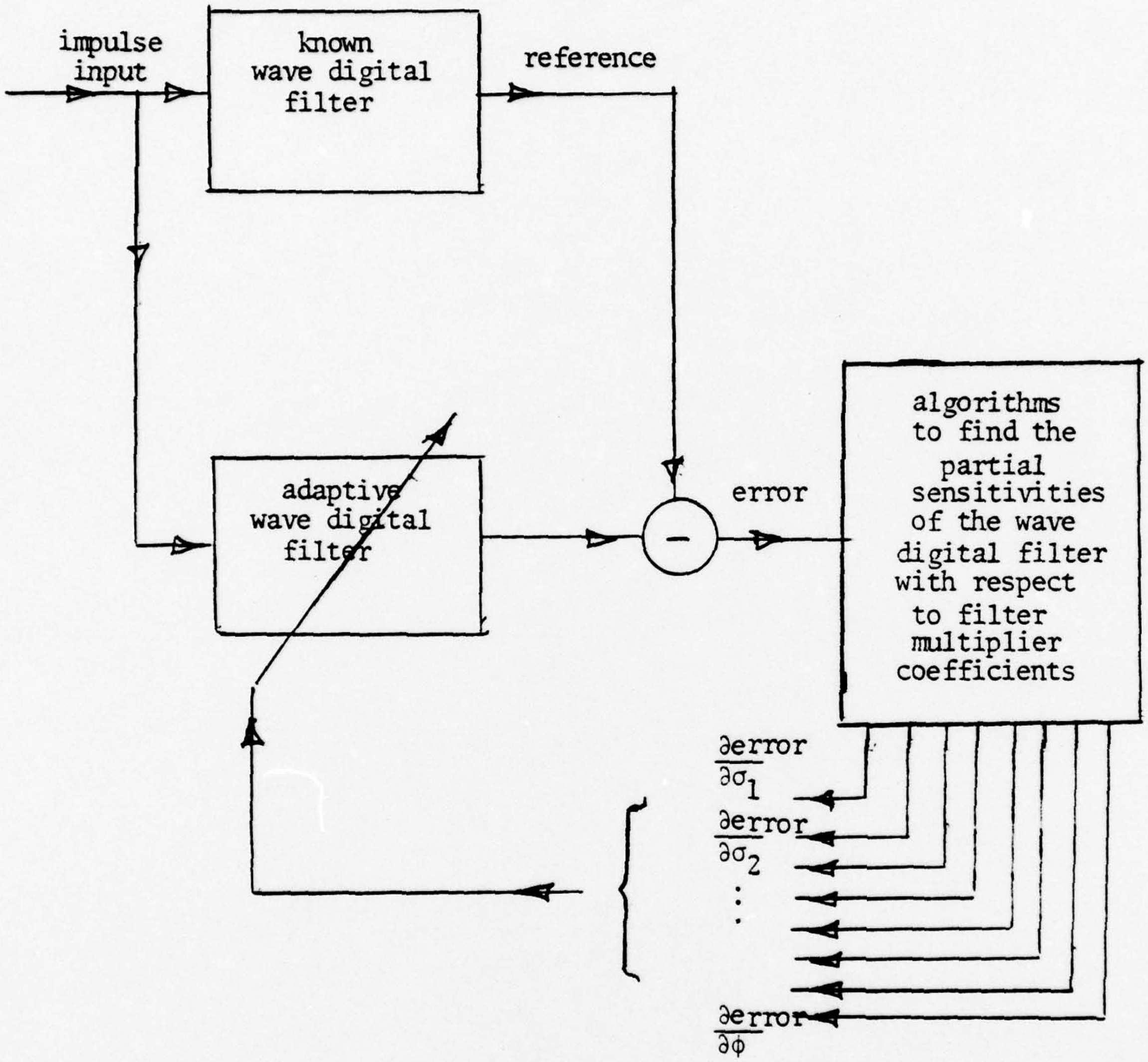


Fig. 7.5. A schematic diagram for the proposed adaptive wave digital filter.

References

1. S. Erfani, Design of fixed and variable digital filters using generalized delay units, Ph.D. dissertation, 1976, Southern Methodist University, Dallas, Texas.

2. K. S. Thyagarajan, One and two dimensional wave digital filters with low coefficient sensitivities, 1977, Doctoral dissertation, Concordia University, Montreal, Canada.

APPENDIX 1

A. DESIGN OF LOW PASS LC FILTER WITH THE GIVEN SPECIFICATION

1. Specification

It is required to design a .5 db low pass Chebechev filter with load resistance $R_L = 50\ \Omega$ and source resistance $R_s = 100\ \Omega$, with critical frequency of 100 radian/sec.

2. Data

From the Handbook of the Filter Synthesis by Zverev [1] for the given specification, the normalized values of L and C for $R_L = 1\ \Omega$, making $R_s = 2.0\ \Omega$ as per Figure A.1 are

$R_s = 2.0$ ohms

$L_1 = .4799$ henries

$C_2 = .3536$ farads

$L_3 = 2.2726$ henries

$C_4 = .7512$ farads

$L_5 = 3.5532$ henries

$C_6 = .9513$ farads

$L_7 = 3.0640$ henries

3. Design of Wave Digital Filter

Using this data the required wave digital filter with no delay free path in port two was designed using the table 4.2b. The schematic wave flow diagram of the wave digital filter is shown in Fig. A.2. Note that from Figure A.2 and equation (4.17) the unity input impulse response of the filter will be

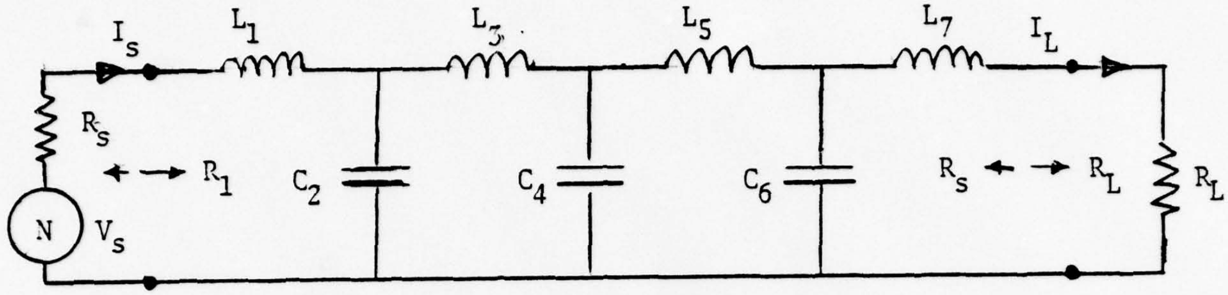


Fig. A.1. Seventh order low pass analogue double resistively terminated LC filter.

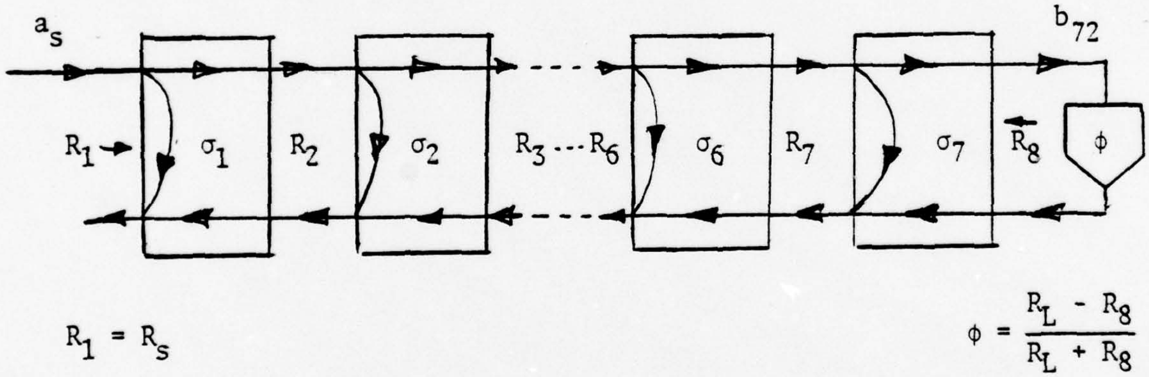


Fig. A.2. Seventh order low pass .5 db ripple Chebyschev wave digital filter designed with no delay free path on port two after the seventh order low pass filter of Fig. A.1. Note that only delay free signal paths are shown.

$$c(nT) = h(nT) = \left(\frac{1+\phi}{2}\right) b_{72}(nT) \qquad \text{(A.1)}$$

where T is the sampling period of the filter. We also note that h(nT) is also the transfer function of the filter in the time domain. Thus the transfer function of the filter in the frequency domain will be

$$H(J\omega) = \sum_{n=0}^{N} h(nT)\, e^{-j\omega nT}$$

or

$$H(J\omega) = \sum_{n=0}^{N} \left(\frac{1+\phi}{2}\right) b_{72}(nT)\cdot e^{-j\omega nT}$$

$$= \frac{1+\phi}{2} \sum_{n=0}^{N} b_{72}(nT)\cdot e^{-j\omega nT} \qquad \text{(A.2)}$$

where ϕ is the reflection coefficient of the filter and N was chosen large enough for transients to decay. Using these results the frequency transfer function and also the impulse response of required filter was programmed in the computer. Sampling period of .01 sec was used, thus making the sampling frequency approximately six times the critical frequency, or three times the Nyquist frequency.

The appropriate computer programs for filter transfer function, and also its impulse response are given at the end of this appendix together with the computer output results which are listed in Tables A.1 and A.2 with their corresponding graphs in Figs. A.4 and A.5.

4. Design of the Conventional Digital Filter for Comparison Purposes

The transfer function of the given filter of Fig. A.1 is of the form

$$H(s) = \frac{V_2(s)}{V_1(s)} \tag{A.3}$$

It can be shown that H(s) is of the form

$$H(s) = \frac{V_2(s)}{V_1(s)} = \frac{1}{b_7s^7+b_6s^6+b_5s^5+b_4s^4+b_3s^3+b_2s^2+b_1sb_0} \tag{A.4}$$

where from the circuit analysis we find

$$b_7 = \frac{L_1C_2L_3C_4L_5C_6L_7}{R_L}$$

$$b_6 = C_2L_3C_4L_5C_6\ (L_1 + L_7 \frac{R_S}{R_L}) \tag{A.5}$$

$$b_5 = \frac{L_1}{R_L}\ [C_2L_3(C_4L_5 + C_4L_7 + C_6L_7) + L_5C_6L_7(C_2 + C_4)]$$

$$+ L_3C_4L_5C_6(C_2R_S + \frac{L_7}{R_L})$$

$$b_4 = (L_1+L_7 \frac{R_S}{R_L})\ [L_5C_6(C_2+C_4)+C_2L_3(C_4+C_6)]+L_3C_4L_5(C_2 \frac{R_S}{R_L} + C_6)$$

$$b_3 = \frac{C_4}{R_L}(L_1+L_3)(L_5+L_7)+(L_3+L_5)\ [C_2C_6R_S + \frac{L_1C_2+C_6L_7}{R_L})$$

$$+ C_4R_S(C_2L_3+L_5C_6)+ \frac{L_1L_7}{R_L}\ (C_2+C_6)$$

$$b_2 = (C_4+C_6)(L_1+L_3+L_7 \frac{R_S}{R_L})+C_2[L_1+ \frac{R_S}{R_L}(L_3+L_5+L_7)]+L_5(C_4 \frac{R_S}{R_L} + C_6)$$

$$b_1 = \frac{1}{R_L}(L_1+L_3+L_5+L_7) + R_S(C_2+C_4+C_6)$$

and

$$b_o = \frac{R_S + R_L}{R_L}$$

The equation (A.4) can be rewritten in a more familiar form of

$$H(s) = \frac{V_2(s)}{V_1(s)} = \frac{K_1}{s^7+a_6s^6+a_5s^5+a_4s^4+a_3s^3+a_2s^2+a_1s+a_0} \qquad \text{(A.6)}$$

where $K_1 = \frac{1}{b_7}$

$$a_6 = \frac{b_6}{b_7}$$

$$a_5 = \frac{b_5}{b_7}$$

etc.

To find the filter scale factor we let $s \to 0$. This leads to the value $\frac{R_L + R_S}{R_L}$ which is the filter's scale factor.

To find T(z) for the direct digital filter design, i.e. the digital filter transfer function, we can use equation (A.6) and the bilinear transform equation (2.4) to get

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

or for simplicity we can factor H(s) into one first order and three second order sections as shown in equation (A.8)

$$H(s) = \frac{K}{(s^2+A_1s+B_1)(s^2+A_2s+B_2)(s^2+A_3s+B_3)(s+A_4)} \tag{A.8}$$

Note that in effect we are going to have one first order filter cascaded with three second order cascaded filters. It is much easier to bilinear transform the subsections one at a time rather than bilinear transform the whole H(s). Thus each second order section becomes

$$\left.\frac{1}{s^2+A_1s+B_1}\right|_{s = \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)}$$

$$= K_2 \frac{1 + 2z^{-1} + z^{-2}}{1 + \alpha_1 z^{-1} + \beta_1 z^{-2}}$$

where

$$K_2 = \frac{1}{\frac{4}{T^2} + \frac{2A_1}{T} + B_1}$$

$$\alpha_1 = -2\,\frac{\frac{4}{T^2} - B_1}{\frac{4}{T^2} + \frac{2A_1}{T} + B_1}$$

$$\beta_1 = \frac{\frac{4}{T^2} - \frac{2A_1}{T} + B_1}{\frac{4}{T^2} + \frac{2A_1}{T} + B_1}$$

and the first order section transforms into

$$\left.\frac{1}{s + A_4}\right|_{s = \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)}$$

$$= K_5 \frac{1 + z^{-1}}{1+\alpha_4 z^{-1}}$$

where

$$K_5 = \frac{1}{\frac{2}{T} + A_4}$$

$$\alpha_4 = -\frac{\frac{2}{T} - A_4}{\frac{2}{T} + A_4}$$

Thus (A.7) and (A.8) lead to

$$T(z) = K \frac{(1+z^{-1})^2}{(1+\alpha_1 z^{-1}+\beta_1 z^{-2})} \cdot \frac{(1+z^{-1})^2}{(1+\alpha_2 z^{-1}+\beta_2 z^{-2})} \cdot \frac{(1+z^{-1})^2}{(1+\alpha_3 z^{-1}+\beta_3 z^{-2})} \cdot \frac{1+z^{-1}}{1+\alpha_4 z^{-1}}$$

where

$$K = K_1 \cdot K_2 \cdot K_3 \cdot K_4 \cdot K_5$$

Note that T(z) is merely the transfer function of four cascaded first order and second order sections as shown in Figure A.3. Thus the appropriate iterative equations are

$$\begin{aligned}
V_1(nT) &= KV_{in}(nT) + 2KV_{in}(nT-T) + KV_{in}(nT-2T) - \alpha_1 V_1(nT-T) - \beta_1 V_1(nT-2T) \\
V_2(nT) &= V_1(nT) + 2V_1(nT-T) + V_1(nT-2T) - \alpha_2 V_2(nT-T) - \beta_2 V_2(nT-2T) \\
V_3(nT) &= V_2(nT) + 2V_2(nT-T) + V_2(nT-2T) - \alpha_3 V_3(nT-T) - \beta_3 V_3(nT-2T) \\
V_4(nT) &= V_3(nT) + V_3(nT-T) - \alpha_4 V_0(nT-T) \qquad \text{(A.9)}
\end{aligned}$$

where T is the sampling time of the filter and with initial conditions

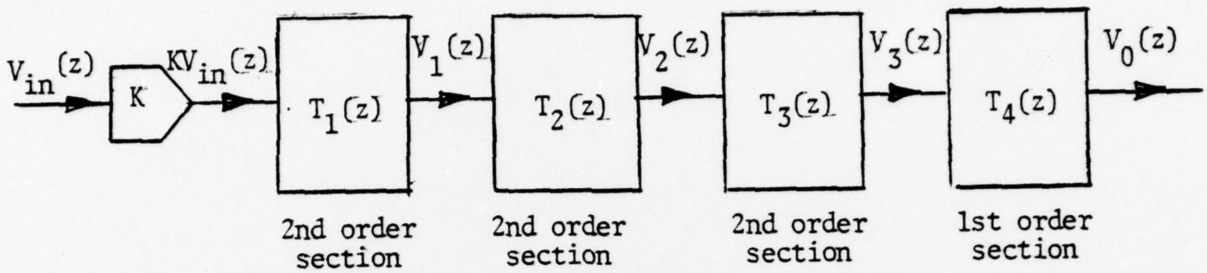


Fig. A.3. Seventh order digitized and cascaded filter corresponding to seventh order analogue filter of Fig. A.1.

set to zero, i.e.

$$V_{in}(I) = 0 \qquad I = -1,-2, \quad J = 1,2,3$$

$$V_J(I) = 0$$

$$V_o(-1) = 0$$

Also we note that for impulse response

$$V_{in}(I) = V_{in}(nT) = \begin{cases} 1 & \text{for } n=0 \\ 0 & \text{for } n\neq 0 \end{cases}$$

Thus the unity input impulse response of the filter from equation (A.9) will be $V_o(nT)$ and we note that this is also the filter transfer function in the time domain, i.e.

$$V_o(nT) = h(nT)$$

Also the filter transfer function in the frequency domain can be found from equation (A.2). Using these results the transfer function and also unity impulse response of filter was programmed in the computer and computed using the same sampling frequency as that of the wave digital filter of Section 3.

The simulation computer programs are given at the end of this appendix and their corresponding outputs are given in Tables A.1, A.2 together with the results obtained for the same filter using wave digital filter design for comparison purposes. Note that no graphs are given for this simulation, since they were exactly the same as the wave digital filter graphs.

Frequency	Wave Digital Filter	Conventional Digital Filter
0.0	0.100000D 01	0.100000D 01
0.200000D 01	[illegible]	[illegible]
0.400000D 01	0.995870D 00	0.995870D 00
0.600000D 01	0.991030D 00	0.991030D 00
0.800000D 01	0.984830D 00	0.984830D 00
0.100000D 02	0.977750D 00	0.977750D 00
0.120000D 02	0.970310D 00	0.970320D 00
0.140000D 02	0.963060D 00	0.963060D 00
0.160000D 02	0.956480D 00	0.956480D 00
0.180000D 02	0.950970D 00	0.950990D 00
0.200000D 02	0.946940D 00	0.946950D 00
0.220000D 02	0.944600D 00	0.944610D 00
0.240000D 02	0.944130D 00	0.944130D 00
0.260000D 02	0.945590D 00	0.945590D 00
0.280000D 02	0.948940D 00	0.948940D 00
0.300000D 02	0.954020D 00	0.954020D 00
0.320000D 02	0.960550D 00	0.960550D 00
0.340000D 02	0.968120D 00	0.968120D 00
0.360000D 02	0.976170D 00	0.976170D 00
0.380000D 02	0.984060D 00	0.984060D 00
0.400000D 02	0.991040D 00	0.991050D 00
0.420000D 02	0.996390D 00	0.996400D 00
0.440000D 02	0.999460D 00	0.999460D 00
0.460000D 02	0.999790D 00	0.999790D 00
0.480000D 02	0.997220D 00	0.997220D 00
0.500000D 02	0.991930D 00	0.991930D 00
0.520000D 02	0.984430D 00	0.984430D 00
0.540000D 02	0.975520D 00	0.975520D 00
0.560000D 02	0.966170D 00	0.966170D 00
0.580000D 02	0.957420D 00	0.957420D 00
0.600000D 02	0.950270D 00	0.950270D 00
0.620000D 02	0.945590D 00	0.945590D 00
0.640000D 02	0.944080D 00	0.944080D 00
0.660000D 02	0.946160D 00	0.946160D 00
0.680000D 02	0.951920D 00	0.951920D 00
0.700000D 02	0.960960D 00	0.960960D 00
0.720000D 02	0.972300D 00	0.972310D 00
0.740000D 02	0.984220D 00	0.984220D 00
0.760000D 02	0.994260D 00	0.994260D 00
0.780000D 02	0.999670D 00	0.999670D 00
0.800000D 02	0.998150D 00	0.998150D 00
0.820000D 02	0.989060D 00	0.989070D 00
0.840000D 02	0.974230D 00	0.974230D 00
0.860000D 02	0.957970D 00	0.957970D 00
0.880000D 02	0.946230D 00	0.946240D 00
0.900000D 02	0.945400D 00	0.945400D 00
0.920000D 02	0.960260D 00	0.960260D 00
0.940000D 02	0.987710D 00	0.987710D 00
0.960000D 02	0.996860D 00	0.996860D 00
0.980000D 02	0.911060D 00	0.911060D 00
0.100000D 03	0.706920D 00	0.706920D 00
0.102000D 03	0.492110D 00	0.492110D 00
0.104000D 03	0.334990D 00	0.334990D 00
0.106000D 03	0.231800D 00	0.231800D 00
0.108000D 03	0.164450D 00	0.164450D 00
0.110000D 03	0.119500D 00	0.119500D 00
0.112000D 03	0.886630D-01	0.886620D-01
0.114000D 03	0.669500D-01	0.669500D-01
0.116000D 03	0.513130D-01	0.513130D-01
0.118000D 03	0.398290D-01	0.398290D-01
0.120000D 03	0.312540D-01	0.312540D-01
0.122000D 03	0.247570D-01	0.247570D-01
0.124000D 03	0.197730D-01	0.197730D-01
0.126000D 03	0.159070D-01	0.159070D-01
0.128000D 03	0.128790D-01	0.128790D-01
0.130000D 03	0.104880D-01	0.104880D-01
0.132000D 03	0.858400D-02	0.858400D-02
0.134000D 03	0.705820D-02	0.705820D-02
0.136000D 03	0.582770D-02	0.582770D-02
0.138000D 03	0.482970D-02	0.482970D-02
0.140000D 03	0.401620D-02	0.401620D-02
0.142000D 03	0.335000D-02	0.335000D-02
0.144000D 03	0.280210D-02	0.280210D-02
0.146000D 03	0.234970D-02	0.234970D-02
0.148000D 03	0.197490D-02	0.197490D-02
0.150000D 03	0.166340D-02	0.166340D-02
0.152000D 03	0.140360D-02	0.140360D-02
0.154000D 03	0.118640D-02	0.118640D-02
0.156000D 03	0.100440D-02	0.100440D-02
0.158000D 03	0.851450D-03	0.851450D-03
0.160000D 03	0.722690D-03	0.722680D-03
0.162000D 03	0.614060D-03	0.614060D-03
0.164000D 03	0.522260D-03	0.522260D-03
0.166000D 03	0.444550D-03	[illegible]
0.168000D 03	0.378670D-03	0.378670D-03
0.170000D 03	0.322740D-03	0.322740D-03
0.172000D 03	0.275210D-03	0.275210D-03
0.174000D 03	0.234760D-03	0.234760D-03
0.176000D 03	0.200310D-03	[illegible]
0.178000D 03	0.170940D-03	[illegible]
0.180000D 03	0.145890D-03	[illegible]
0.182000D 03	0.124510D-03	[illegible]
0.184000D 03	0.106240D-03	[illegible]
0.186000D 03	0.906350D-04	[illegible]
0.188000D 03	0.772930D-04	[illegible]
0.190000D 03	0.658860D-04	[illegible]
0.192000D 03	0.561370D-04	[illegible]
0.194000D 03	0.477990D-04	[illegible]

Table A.1. Computer frequency response output for both wave digital filter and conventional cascaded digital filter.

Time	Wave Digital Filter	Conventional Digital Filter
0.0	0.192850D-03	0.192850D-03
0.100000D-01	0.220300D-02	0.220300D-02
0.200000D-01	0.119600D-01	0.119600D-01
0.300000D-01	0.411100D-01	0.411100D-01
0.400000D-01	0.100330D 00	0.100330D 00
0.500000D-01	0.183980D 00	0.183980D 00
0.600000D-01	0.259580D 00	0.259580D 00
0.700000D-01	0.279570D 00	0.279570D 00
0.800000D-01	0.213400D 00	0.213400D 00
0.900000D-01	0.795460D-01	0.795460D-01
0.100000D 00	-0.583440D-01	-0.583440D-01
0.110000D 00	-0.129390D 00	-0.129390D 00
0.120000D 00	-0.104460D 00	-0.104480D 00
0.130000D 00	-0.165250D-01	-0.165260D-01
0.140000D 00	0.647120D-01	0.647120D-01
0.150000D 00	0.841570D-01	0.841570D-01
0.160000D 00	0.389290D-01	0.389290D-01
0.170000D 00	-0.259790D-01	-0.259790D-01
0.180000D 00	-0.574950D-01	-0.574950D-01
0.190000D 00	-0.359430D-01	-0.359430D-01
0.200000D 00	0.141420D-01	0.141420D-01
0.210000D 00	0.490260D-01	0.490260D-01
0.220000D 00	0.428390D-01	0.428390D-01
0.230000D 00	0.580320D-02	0.580340D-02
0.240000D 00	-0.287070D-01	-0.287060D-01
0.250000D 00	-0.340830D-01	-0.340830D-01
0.260000D 00	-0.110470D-01	-0.110470D-01
0.270000D 00	0.167550D-01	0.167550D-01
0.280000D 00	0.253870D-01	0.253870D-01
0.290000D 00	0.105230D-01	0.105230D-01
0.300000D 00	-0.115880D-01	-0.115880D-01
0.310000D 00	-0.206040D-01	-0.206040D-01
0.320000D 00	-0.100290D-01	-0.100290D-01
0.330000D 00	0.876790D-02	0.876780D-02
0.340000D 00	0.183400D-01	0.183400D-01
0.350000D 00	0.109640D-01	0.109640D-01
0.360000D 00	-0.553760D-02	-0.553750D-02
0.370000D 00	-0.159940D-01	-0.159940D-01
0.380000D 00	-0.117160D-01	-0.117160D-01
0.390000D 00	0.249000D-02	0.248990D-02
0.400000D 00	0.135720D-01	0.135720D-01
0.410000D 00	0.121680D-01	0.121680D-01
0.420000D 00	0.387620D-03	0.387740D-03
0.430000D 00	-0.108980D-01	-0.108980D-01
0.440000D 00	-0.120720D-01	-0.120720D-01
0.450000D 00	-0.289770D-02	-0.289760D-02
0.460000D 00	0.802160D-02	0.802150D-02
0.470000D 00	0.112880D-01	0.112880D-01
0.480000D 00	0.476440D-02	0.476450D-02
0.490000D 00	-0.521590D-02	-0.521580D-02
0.500000D 00	-0.993110D-02	-0.993110D-02
0.510000D 00	-0.586310D-02	-0.586320D-02
0.520000D 00	0.277640D-02	0.277630D-02
0.530000D 00	0.828750D-02	0.828740D-02
0.540000D 00	0.630320D-02	0.630320D-02
0.550000D 00	-0.815630D-03	-0.815540D-03
0.560000D 00	-0.658350D-02	-0.658340D-02
0.570000D 00	-0.625030D-02	-0.625030D-02
0.580000D 00	-0.662720D-03	-0.662800D-03
0.590000D 00	0.496390D-02	0.496380D-02
0.600000D 00	0.586150D-02	0.586150D-02
0.610000D 00	0.170760D-02	0.170770D-02
0.620000D 00	-0.350900D-02	-0.350900D-02
0.630000D 00	-0.526530D-02	-0.526530D-02
0.640000D 00	-0.238890D-02	-0.238900D-02
0.650000D 00	0.225320D-02	0.225310D-02
0.660000D 00	0.455680D-02	0.455680D-02
0.670000D 00	0.277770D-02	0.277770D-02
0.680000D 00	-0.120190D-02	-0.120180D-02
0.690000D 00	-0.380040D-02	-0.380040D-02
0.700000D 00	-0.293380D-02	-0.293380D-02
0.710000D 00	0.349560D-03	0.349510D-03
0.720000D 00	0.304170D-02	0.304160D-02
0.730000D 00	0.290720D-02	0.290720D-02
0.740000D 00	0.313550D-03	0.313600D-03
0.750000D 00	-0.231510D-02	-0.231510D-02
0.760000D 00	-0.274190D-02	-0.274190D-02
0.770000D 00	-0.800810D-03	-0.800850D-03
0.780000D 00	0.164740D-02	0.164740D-02
0.790000D 00	0.247770D-02	0.247770D-02
0.800000D 00	0.112940D-02	0.112950D-02
0.810000D 00	-0.105760D-02	-0.105760D-02
0.820000D 00	-0.215100D-02	-0.215100D-02
0.830000D 00	-0.132010D-02	-0.132010D-02
0.840000D 00	0.557200D-03	0.557170D-03
0.850000D 00	0.179320D-02	0.179320D-02
0.860000D 00	0.139600D-02	0.139600D-02
0.870000D 00	-0.150720D-03	-0.150690D-03
0.880000D 00	-0.143070D-02	-0.143060D-02
0.890000D 00	-0.138110D-02	-0.138110D-02
0.900000D 00	-0.163500D-03	-0.163520D-03
0.910000D 00	0.108350D-02	0.108340D-02
0.920000D 00	0.129850D-02	0.129850D-02
0.930000D 00	0.391790D-03	0.391810D-03
0.940000D 00	-0.766020D-03	-0.766010D-03
0.950000D 00	-0.116940D-02	-0.116930D-02
0.960000D 00	-0.543570D-03	-0.543530D-03
0.970000D 00	0.487460D-03	0.487450D-03

Table A.2. Computer output for unity impulse response of both wave digital filter and conventional cascaded digital filter in the time domain.

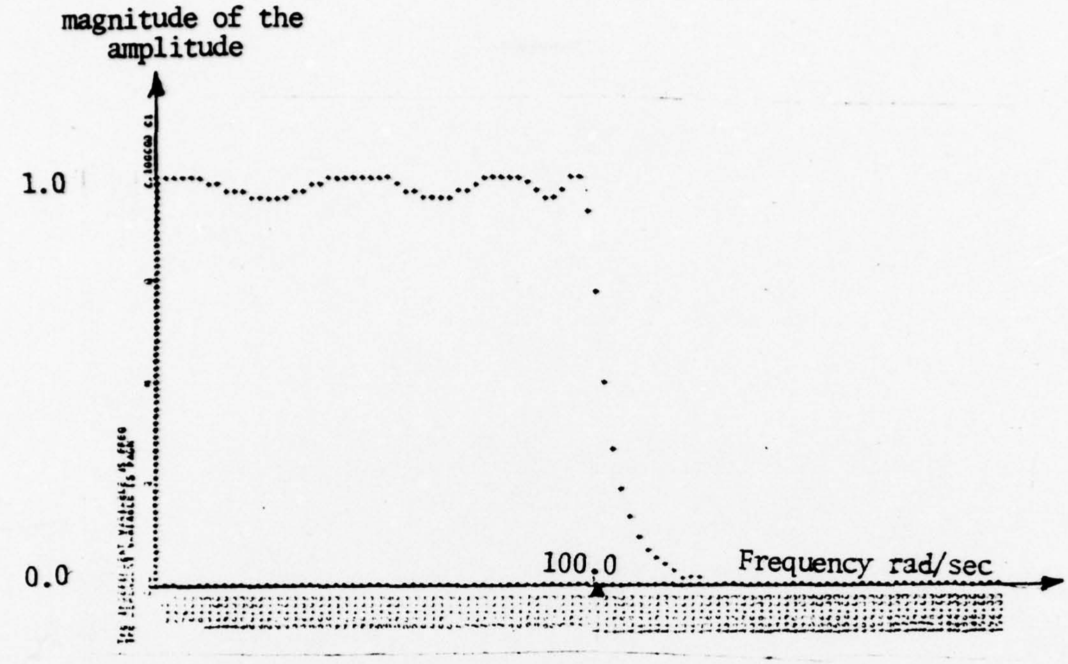


Fig. A.4. Graph of the transfer function of the wave digital filter with the given specification in Section 1.

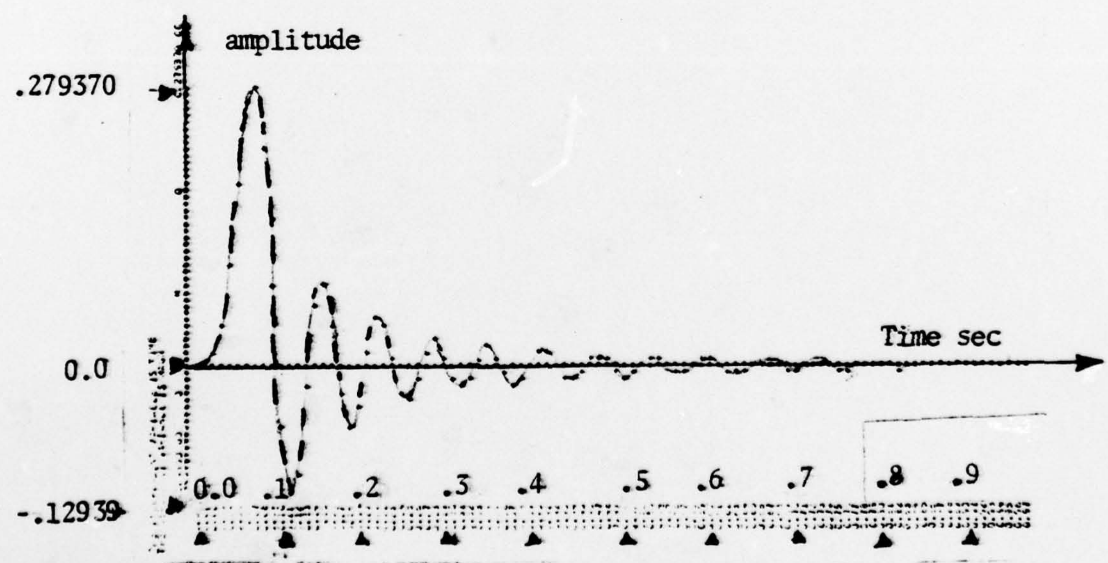


Fig. A.5. Graph of the unity impulse response of wave digital filter with the given specification in Section I.

B. Computer Program No. 1. Program for unity impulse response of the wave digital filter with no delay free path on port two with the given specification .

```
C
C      ********  WAVE DIGITAL FILTER  *************
C
C          ****   TIME RESPONCE    ****
C
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DATA(210,3)
C
C       **********.5DB CHEBECHEV LOW-PASS FILTER WITH RS=2.0   ********
C     COMPONENT VALUES
C     INDUCTANCE AND CAPACITANCE VALUES IN HENRIES,ANC FARADS
C     NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C     WITH   RL=1
C
      RS=2.0D0
      AL1=2.4275D0
      C2=.7470D0
      AL3=4.3695D0
      C4=.8377D0
      AL5=4.4886D0
      C6=.8137D0
      AL7=3.4050D0
      RL=1.0D0
      WRITE(6,27)
 27   FORMAT(5X,'THE UNSCALED COMPONENT VALUES ARE',//)
      WRITE(6,18) RL,RS
      WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
C
C     SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
      OMGAC=100.0D0
C
C     SAMPLING TIME IS T
      T=.01D0
C
C     IMPEDANCE SCALE FACTOR IS    SIMP
      SIMP=50.0D0
C     THE EFFECT OF SAMPLING TIME IS  SFREQ
C     FREQUENCY SCALING WITH PREWARPING AS WELL AS TAKING INTO ACOUNT
      SFREQ=DTAN(OMGAC*T/2)
      RS=RS*SIMP
      AL1=AL1*SIMP/SFREQ
      C2=C2/(SIMP*SFREQ)
      AL3=AL3*SIMP/SFREQ
      C4=C4/(SIMP*SFREQ)
      AL5=AL5*SIMP/SFREQ
      C6=C6/(SIMP*SFREQ)
      AL7=AL7*SIMP/SFREQ
      RL=RL*SIMP
      WRITE(6,29)
 29   FORMAT(5X,'THE   SCALED COMPONENT VALUES ARE',//)
      WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
 22   FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
     1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
      WRITE(6,18) RL,RS
 18   FORMAT(10X,'    RL=',E12.5,5X,'RS=',E12.5,//)
C     FILTER SCALE FACTOR FROM DATA IS COEF
      COEF=(RS+RL)/RL
      WRITE(6,37) COEF
 37   FORMAT(5X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
C
C     CALCULATING THE WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
C     -----------------------------------------------------------
C
      R1=RS
      R2=R1+AL1
      SIGMA1=R1/R2
      G2=1.0D0/R2
      G3=G2+C2
      SIGMA2=G2/G3
      R3=1.0D0/G3
      R4=R3+AL3
```

```
      SIGMA3=R3/R4
      G4=1.CD0/R4
      G5=G4+C4
      SIGMA4=G4/G5
      R5=1.CD0/G5
      R6=R5+AL5
      SIGMA5=R5/R6
      G6=1.CD0/R6
      G7=G6+C6
      SIGMA6=G6/G7
      R7=1.CD0/G7
      R8=R7+AL7
      SIGMA7=R7/R8
      PHI=(RL-R8)/(RL+R8)
      WRITE(6,42)
42    FORMAT(5X,'THE WAVE DIGITAL FILTER MULTIPLIER CCEFFICIENTS ARE....
     *',//)
      WRITE(6,10) SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PHI
10    FORMAT(/,4X,'SIGMA1=',F6.4,4X,'SIGMA2=',F6.4,4X,'SIGMA3=',F6.4,4X,
     *'SIGMA4=',F6.4,4X,'SIGMA5=',F6.4,4X,'SIGMA6=',F6.4,4X,'SIGMA7=',F6
     *.4,4X,'PHI=',F6.4)
C
C     INITIAL VALUES
C     ------------------------------------
C
C     INPUT IN TIME DOMAIN IS    AS
      AS=1.CD0
      DLTAT=0.0D0
      A11=AS*COEF
      X11=0.0D0
      X13=0.0D0
      X11=0.0D0
      X13=0.0D0
      X14=0.0D0
      X23=0.0D0
      X24=0.CD0
      X33=0.0D0
      X34=C.CD0
      X43=0.0D0
      X44=0.0D0
      X53=0.0D0
      X54=0.0D0
      X63=0.0D0
      X64=0.0D0
      X72=0.0D0
      X73=0.0D0
      X74=0.CD0
C
C     ITTERATION IN THE TIME DOMAIN
C     -----------------------------
C
      DC 10C I=1,98
      B12=A11+X11-X23+SIGMA1*(X23-X14)
      B22=X33+SIGMA2*(B12+X14-X33-X24)
      B32=B22+X24-X43+SIGMA3*(X43-X34)
      B42=X53+SIGMA4*(B32+X34-X53-X44)
      B52=B42+X44-X63+SIGMA5*(X63-X54)
      B62=X73+SIGMA6*(B52+X54-X73-X64)
      B72=B62+X64-X72+SIGMA7*(X72-X74)
      A72=B72*PHI
      B71=B62+SIGMA7*(A72-B62+X72-X73)
      B61=B71-B52+X73+SIGMA6*(B52-X63)
      B51=B42+SIGMA5*(B61-B42+X63-X53)
      B41=B51-B32+X53+SIGMA4*(B32-X43)
      B31=B22+SIGMA3*(B41-B22+X43-X33)
      B21=B31-B12+X33+SIGMA2*(B12-X23)
      B11=A11+SIGMA1*(B21-A11+X23-X13)
C
C     UPDATED VALUES
C     -----------------------------
C
      X11=A11
      X13=B11
```

```
      X14=B12
      X23=B21
      X24=B22
      X33=B31
      X34=B32
      X43=B41
      X44=B42
      X53=B51
      X54=B52
      X63=B61
      X64=B62
      X72=A72
      X73=B71
      X74=B72
C
C     FOR IMPULSE RESPONCE A11 IS SET TO ZERO FOR THE NEXT ITTERATION
      A11=0.0D0
C
C     ARRANGING THE OUTPUT DATA
      DATA(I,1)=DLTAT
      DATA(I,2)=B72*(1.0D0+PHI)/2
      DATA(I,3)=0.0D0
      DLTAT=DLTAT+T
  100 CONTINUE
C
C
      WRITE(6,54)
   54 FORMAT('1')
      WRITE(6,55)
   55 FORMAT(20X,'TIME',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DATA,98,4HTIME,9HMAGNITUDE)
   20 FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      STOP
      END
```

GRAPHX

```
      SUBROUTINE GRAPHX (DATA,N,VINDEP,VARDEP)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DATA(210,3),B(121)
      DATA DOT/'+'/,STAR/'*'/,BLANK/' '/
      WRITE(6,300) VINDEP
300   FORMAT(1H1,' THE INDEPENDENT VARIABLE IS ',A4)
      WRITE(6,400) VARDEP
400   FORMAT(1H,' THE DEPENDENT VARIABLE IS ',A4)
      WRITE(6,500)
500   FORMAT(' ')
      BIGEST=DATA(1,2)
      SMAL=DATA(1,2)
      DO 1 I=2,N
      IF(DATA(I,2).GT.BIGEST)BIGEST=DATA(I,2)
      IF(DATA(I,2).LT.SMAL)SMAL=DATA(I,2)
1     CONTINUE
      DO 2 I=1,N
      IF(DATA(I,3).GT.BIGEST)BIGEST=DATA(I,3)
      IF(DATA(I,3).LT.SMAL)SMAL=DATA(I,3)
2     CONTINUE
      IF(SMAL.GE.0.0D0) SMAL=0.0D0
      WRITE(6,200) SMAL,BIGEST
      BMINS=BIGEST-SMAL
      DO 3 I=1,62
3     B(I)=BLANK
      DO 4 I=1,N
      DATA(I,2)=(DATA(I,2)-SMAL)*61.0D0/BMINS+1.0D0
      DATA(I,3)=(DATA(I,3)-SMAL)*61.0D0/BMINS+1.0D0
      INDEX=DATA(I,2)
      JNDEX=DATA(I,3)
      B(INDEX)=DOT
      B(JNDEX)=STAR
      WRITE(6,100) DATA(I,1),(B(NN1),NN1=1,62)
      B(INDEX)=BLANK
      B(JNDEX)=BLANK
4     CONTINUE
200   FORMAT(8X,E12.5,5X,'Q',14X,'M',14X,'Q',12X,E12.5,/10X,61(1H*))
100   FORMAT(1X,F8.2,1X,62A1)
      RETURN
      END
```

C. Computer Program* No. 2. Program for transfer function of the wave digital filter with no delay free path on port two with the given specification.

```
C        ********  WAVE DIGITAL FILTER   *************
C
C          ****  FREQUENCY RESPONCE    ****
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DATA(210,3)
      COMPLEX*16 H2,W1,Z
C
C      **********.5DB CHEBECHEV LOW-PASS FILTER WITH RS=2.0   ********
C     COMPONENT VALUES
C     INDUCTANCE AND CAPACITANCE VALUES IN HENRIES,AND FARADS
C     NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C     WITH   RL=1
C
      RS=2.0D0
      AL1=2.42750D0
      C2=.74700D0
      AL3=4.36550D0
      C4=.83770D0
      AL5=4.48860D0
      C6=.81370D0
      AL7=3.40500D0
      RL=1.0D0
      WRITE(6,27)
   27 FORMAT(5X,'THE UNSCALED COMPONENT VALUES ARE',//)
      WRITE(6,18) RL,RS
      WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
C
C     SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
      OMGAC=100.0D0
C
C     SAMPLING TIME IS T
      T=.01D0
C
C     IMPEDANCE SCALE FACTOR IS   SIMP
      SIMP=50.0D0
C
C
C     FREQUENCY SCALING WITH PREWARPING AS WELL AS TAKING INTO ACOUNT
C     THE EFFECT OF SAMPLING TIME IS   SFREQ
      SFREQ=DTAN(OMGAC*T/2)
      RS=RS*SIMP
      AL1=AL1*SIMP/SFREQ
      C2=C2/(SIMP*SFREQ)
      AL3=AL3*SIMP/SFREQ
      C4=C4/(SIMP*SFREQ)
      AL5=AL5*SIMP/SFREQ
      C6=C6/(SIMP*SFREQ)
      AL7=AL7*SIMP/SFREQ
      RL=RL*SIMP
      WRITE(6,29)
   29 FORMAT(5X,' THE   SCALED COMPONENT VALUES ARE',//)
      WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
   22 FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
     1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
      WRITE(6,18) RL,RS
   18 FORMAT(10X,'    RL=',E12.5,5X,'RS=',E12.5,//)
C
C     FILTER SCALE FACTOR FROM DATA IS COEF
      COEF=(RS+RL)/RL
      WRITE(6,37) COEF
   37 FORMAT(5X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
C
C     CALCULATING THE WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
C     ------------------------------------------------------------
C
C
      R1=RS
      R2=R1+AL1
      SIGMA1=R1/R2
      G2=1.0D0/R2
      G3=G2+C2
      SIGMA2=G2/G3
```


*Graphx, subroutine is given in Computer Program No. 1.

```
      R3=1.CD0/G3
      R4=R3+AL3
      SIGMA3=R3/R4
      G4=1.CD0/R4
      G5=G4+C4
      SIGMA4=C4/G5
      R5=1.CD0/G5
      R6=R5+AL5
      SIGMA5=R5/R6
      G6=1.CD0/R6
      G7=G6+C6
      SIGMA6=G6/G7
      R7=1.CD0/G7
      R8=R7+AL7
      SIGMA7=R7/R8
      PHI=(RL-R8)/(RL+R8)
C
      WRITE(6,42)
 42   FORMAT(5X,'THE WAVE DIGITAL FILTER MULTIPLIER CCEFFICIENTS ARE....
     *',//)
C
      WRITE(6,10) SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PHI
 10   FORMAT(/,4X,'SIGMA1=',F6.4,4X,'SIGMA2=',F6.4,4X,'SIGMA3=',F6.4,4X,
     1'SIGMA4=',F6.4,4X,'SIGMA5=',F6.4,4X,'SIGMA6=',F6.4,4X,'SIGMA7=',F6
     2.4,4X,'PHI=',F6.4,/)
C
C
C     FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
C     FREQUENCY INCREMENT
      DLTAW=OMGAC/50.D0
C
C     INITIAL VALUES IN FREQUENCY DOMAIN
C
C     INPUT IN TIME DOMAIN IS     AS
      AS=1.CD0
      W=0.0D0
C
C     ITTERATION IN THE FREQUENCY DOMAIN
C     -----------------------------------
      DO 110 J=1,98
C
C     INITIAL VALUES IN TIME DOMAIN
C
      A11=AS*COEF
      H2=DCMPLX(0.CD0,0.0D0)
      TT=0.0D0
      X14=0.CD0
      X23=0.0D0
      X24=0.0D0
      X33=0.CD0
      X34=0.0D0
      X43=0.0D0
      X44=0.CD0
      X53=0.0D0
      X54=0.0D0
      X63=0.0D0
      X64=0.0D0
      X72=0.CD0
      X73=0.0D0
      X74=0.0D0
C
C     ITTERATION IN THE TIME DOMAIN
C     ------------------------------
      DO 100 I=1,500
      B12=A11+X11-X23+SIGMA1*(X23-X14)
      B22=X33+SIGMA2*(B12+X14-X33-X24)
      B32=B22+X24-X43+SIGMA3*(X43-X34)
      B42=X53+SIGMA4*(B32+X34-X53-X44)
      B52=B42+X44-X63+SIGMA5*(X63-X54)
      B62=X73+SIGMA6*(B52+X54-X73-X64)
      B72=B62+X64-X72+SIGMA7*(X72-X74)
      A72=B72*PHI
      B71=B62+SIGMA7*(A72-B62+X72-X73)
```

```
      B61=B71-B52+X73+SIGMA6*(B52-X63)
      B51=B42+SIGMA5*(B61-B42+X63-X53)
      B41=B51-B32+X53+SIGMA4*(B32-X43)
      B31=B22+SIGMA3*(B41-B22+X43-X33)
      B21=B31-B12+X33+SIGMA2*(B12-X23)
      B11=A11+SIGMA1*(B21-A11+X23-X13)
C
C     UPDATED VALUES
C     ------------------------------
      X11=A11
      X13=B11
      X14=B12
      X23=B21
      X24=B22
      X33=B31
      X34=B32
      X43=B41
      X44=B42
      X53=B51
      X54=B52
      X63=B61
      X64=B62
      X72=A72
      X73=B71
      X74=B72
      A11=0.0D0
      WT=W*TT
      W1=DCMPLX(0.0D0,-WT)
      Z=CDEXP(W1)
      H2=H2+B72*Z
      TT=TT+T
  100 CONTINUE
C
      DB=CDABS(H2)*(1.0D0+PHI)/2
C     ARRANGING THE OUTPUT DATA
      DATA(J,1)=W
      DATA(J,2)=DB
      DATA(J,3)=0.0D0
      W=W+DLTAW
  110 CONTINUE
C
      WRITE(6,54)
   54 FORMAT('1')
      WRITE(6,55)
   55 FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DATA,98,4HFREQ,9HMAGNITUDE)
   20 FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      STOP
      END
```

AD-A073 962 NAVAL POSTGRADUATE SCHOOL MONTEREY CA F/G 9/2

AN IN-DEPTH STUDY OF THE SENSITIVITY OF WAVE DIGITAL FILTERS,(U)

JUN 79 M SANAIE-FARD

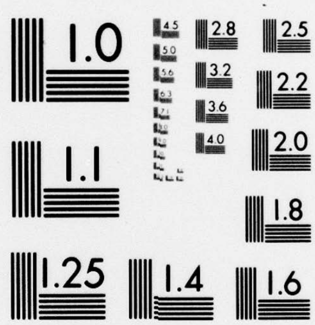
UNCLASSIFIED NL

3 OF 3

AD
A073962

END

DATE
FILMED
10-79
DDC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

D. Computer Program* No. 3. Program for unity impulse response of the conventional cascaded digital filter with the given specification .

```
C     *********TIME RESPONCE ************
C
C        CONNVENSIONAL  DIGITAL FILTER DESIGN
C
C
         IMPLICIT REAL*8 (A-H,O-Z)
         DIMENSION DATA(210,3)
         COMPLEX*16 Z
         DIMENSION A(8),Z(7)
C
C
C         **********.5DB CHEBECHEV LOW-PASS FILTER WITH RS=2.0  ********
C        COMPONENT VALUES
C
C        INDUCTANCE AND CAPACITANCE VALUES IN HENRIES,AND FARADS
C        NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C        WITH    RL=1
C
         RS=2.0D0
         AL1=2.42750D0
         C2=.74700D0
         AL3=4.36950D0
         C4=.83770D0
         AL5=4.48860D0
         C6=.81370D0
         AL7=3.40500D0
         RL=1.0D0
         WRITE(6,27)
 27      FORMAT(5X,'THE UNSCALED COMPONENT VALUES ARE',//)
         WRITE(6,18) RL,RS
         WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
C
C        SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
         OMGAC=100.0D0
C
C        SAMPLING TIME IS T
         T=.01D0
C
C        IMPEDANCE SCALE FACTOR IS   SIMP
         SIMP=50.0D0
C
C        FREQUENCY SCALING WITH PREWARPING IS  SFREQ
         SFREQ=2*DTAN(OMGAC*T/2)/T
         RS=RS*SIMP
         AL1=AL1*SIMP/SFREQ
         C2=C2/(SIMP*SFREQ)
         AL3=AL3*SIMP/SFREQ
         C4=C4/(SIMP*SFREQ)
         AL5=AL5*SIMP/SFREQ
         C6=C6/(SIMP*SFREQ)
         AL7=AL7*SIMP/SFREQ
         RL=RL*SIMP
         WRITE(6,29)
 29      FORMAT(5X,'THE   SCALED COMPONENT VALUES ARE',//)
         WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
 22      FORMAT(5X,' L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
        1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
         WRITE(6,18) RL,RS
 18      FORMAT(10X,'   RL=',E12.5,5X,'RS=',E12.5,//)
C
C        CALCULATION OF THE COEFFICIENTS OF
C        ------------------------------------
C
C         H(S)=K/(S7+A6*S6+A5*S5+A4*S4+A3*S3+A2*S2+A1*S1+AC)
C
         AAC=(RL+RS)/RL
C
C        FILTER SCALE FACTOR FROM DATA IS COEF
         COEF=AA0
         WRITE(6,37) COEF
 37      FORMAT(5X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
C
         AA1=(AL1+AL3+AL5+AL7)/RL+RS*(C2+C4+C6)
```

*Graphx, subroutine is given in Computer Program No. 1.

```
      AA2=(C4+C6)*(AL1+AL3+AL7*RS/RL)+C2*(AL1+(AL3+AL5+AL7)*RS/RL)+AL5*(
     *C6+C4*RS/RL)
      AA3=(AL1+AL3)*(AL5+AL7)*C4/RL+(AL3+AL5)*(C2*C6*RS+(C6*AL7+C2*AL1)/
     *RL)+C4*RS*(C2*AL3+AL5*C6)+AL1*AL7*(C2+C6)/RL
      AA4=(AL1+AL7*RS/RL)*(AL5*C6*(C2+C4)+AL3*C2*(C4+C6))+C4*AL5*AL3*(C6
     *+C2*RS/RL)
      AA5=AL1*(C2*AL3*(C4*(AL5+AL7)+C6*AL7)+AL5*C6*AL7*(C2+C4))/RL+AL3*C
     *4*AL5*C6*(AL7/RL+C2*RS)
      AA6=C2*AL3*C4*AL5*C6*(AL7*RS/RL+AL1)
      AA7=(AL1*C2*AL3*C4*AL5*C6*AL7)/RL
C
      A(1)=1.0D0
      A(2)=AA6/AA7
      A(3)=AA5/AA7
      A(4)=AA4/AA7
      A(5)=AA3/AA7
      A(6)=AA2/AA7
      A(7)=AA1/AA7
      A(8)=AA0/AA7
C
      COEF1=COEF/AA7
      WRITE(6,41)
   41 FORMAT(5X,'THE COEF.OF    H(S)=K/(S7+A6*S6+A5*S5+A4*S4+A3*S3+A2*S2+
     1A1*S1+A0)     ARE',/)
      WRITE(6,40) A(2),A(3),A(4),A(5),A(6),A(7),A(8)
   40 FORMAT(5X,'A6=',E12.5,3X,'A5=',E12.5,3X,'A4=',E12.5,3X,'A3=',E12.5
     1,3X,'A2=',E12.5,3X,'A1=',E12.5,3X,'A0=',E12.5,/)
      WRITE(6,43) COEF1
   43 FORMAT(5X,'K=',E12.5,//)
C
C     CALCULATION OF THE COEF. OF THE EQN.
C     ----------------------------------------------------
C                 H(S)=  K/(S2+A1*S+B1)(S2+A2*S+B2)(S2+A3*S+B3)(S+B4)
C
C
      NDEG=7
      CALL ZPOLR(A,NDEG,Z,IER)
      P1=-2.0D0*REAL(Z(1))
      F1=REAL(Z(1))**2+AIMAG(Z(1))**2
      P2=-2.0D0*REAL(Z(3))
      F2=REAL(Z(3))**2+AIMAG(Z(3))**2
      P3=-2.0D0*REAL(Z(5))
      F3=REAL(Z(5))**2+AIMAG(Z(5))**2
      P4=-REAL(Z(7))
      PP=1.0D0/AA7
      WRITE(6,42)
   42 FORMAT(5X,'THE COEF. OF H(S) ARE.....
     1      H(S)=K/(S2+A1*S+B1)(S2+A2*S+B2)(S2+A3*S+B3)(S+B4)',//)
      WRITE(6,60) P1,F1,P2,F2,P3,F3,P4,COEF1
   60 FORMAT(1X,'A1=',E12.5,1X,'B1=',E12.5,2X,'A2=',E12.5,1X,'B2=',E12.5
     1,2X,'A3=',E12.5,1X,'B3=',E12.5,2X,'A4=',E12.5,3X,'K=',E12.5,//)
C
C     CALCULATION OF THE COEF OF H(Z)  WITH SAMPLING PERIOD OF T
C     -------------------------------------------------------------
C
C                 -1 7          -1          -1     -2            -1     -2          -1    -2
C     H(Z)=K*(1+Z  ) /(1+A4*Z  )(1+A1*Z  +B1*Z  )(1+A2*Z  +B2*Z  )(1+A3Z  +B3Z  )
C
C
      F=2.0D0/T
      FF=F*F
      B1=2.0D0*(F1-FF)/(FF+P1*F+F1)
      D1=(FF-P1*F+F1)/(FF+P1*F+F1)
      B2=2.0D0*(F2-FF)/(FF+P2*F+F2)
      D2=(FF-P2*F+F2)/(FF+P2*F+F2)
      B3=2.0D0*(F3-FF)/(FF+P3*F+F3)
      D3=(FF-P3*F+F3)/(FF+P3*F+F3)
      B4=(P4-F)/(P4+F)
      COEF2=COEF1/((P4+F)*(FF+F*P1+F1)*(FF+F*P2+F2)*(FF+F*P3+F3))
      WRITE(6,66)
   66 FORMAT(5X,'THE COEF. OF Z**-1 AND Z**-2 IN QUADRATURE FORM AND TOT
     1AL MULTLYING FACTOR K ARE...',/)
      WRITE(6,60) B1,D1,B2,D2,B3,D3,B4,COEF2
```

```
C
C       INITIAL VALUES
C       ----------------------
C       INPUT IN TIME DOMAIN IS     UP
        UP=1.0D0
        UPP=UP*COEF2
        X1=0.0D0
        X2=0.0D0
        X3=0.0D0
        X4=0.0D0
        X5=0.0D0
        X6=0.0D0
        X7=0.0D0
        DLTAT=0.0D0
C
C       ITTERATION IN THE TIME DOMAIN
C       -----------------------------
C
        DO 100 I=1,98
        V1=UPP+X1
        V2=V1+X2
        V3=V2+X3
        V=V3+X4
        X1=UPP-B4*V1
        X2=2*V1-B1*V2+X5
        X3=2*V2-B2*V3+X6
        X4=2*V3-B3*V+X7
        X5=V1-D1*V2
        X6=V2-D2*V3
        X7=V3-D3*V
C
C       FOR IMPULSE RESPONCE UPP IS SET TO ZERO FOR THE NEXT ITTERATION
        UPP=0.0D0
C
C       ARRANGING THE OUTPUT DATA
        DATA(I,1)=DLTAT
        DATA(I,2)=V
        DATA(I,3)=0.0D0
        DLTAT=DLTAT+T
  100   CONTINUE
C
        WRITE(6,54)
   54   FORMAT('1')
        WRITE(6,55)
   55   FORMAT(20X,'TIME',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
        WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
        CALL GRAPHX(DATA,98,4HTIME,9HMAGNITUDE)
   20   FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
        STOP
        END
```

E. Computer Program* No. 4. Program for transfer function of the conventional cascaded digital filter with the given specification.

```
C        **** FREQUENCY RESPONCE    ****
C
C     CONNVENSIONAL  DIGITAL FILTER DESIGN
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 WT,Z,Y,H
      DIMENSION DATA(210,3)
      DIMENSION A(8),Y(7)
C
C
C     COMPONENT VALUES
C      **********.508 CHEBECHEV LOW-PASS FILTER WITH RS=2.0   ********
C
C     INDUCTANCE AND CAPACITANCE VALUES IN HENRIES, AND FARADS
C     NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C     WITH   RL=1
C
      RS=2.0D0
      AL1=2.4275D0
      C2=.7470D0
      AL3=4.3695D0
      C4=.8377D0
      AL5=4.4886D0
      C6=.8137D0
      AL7=3.4050D0
      RL=1.0D0
      WRITE(6,27)
   27 FORMAT(5X,'THE UNSCALED COMPONENT VALUES ARE',//)
      WRITE(6,18) RL,RS
      WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
C
C     SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
      OMGAC=100.0D0
C
C     SAMPLING TIME IS T
      T=.01D0
C
C     IMPEDANCE SCALE FACTOR IS    SIMP
      SIMP=50.0D0
C
C     FREQUENCY SCALING WITH PREWARPING IS  SFREQ
      SFREQ=2*DTAN(OMGAC*T/2)/T
      RS=RS*SIMP
      AL1=AL1*SIMP/SFREQ
      C2=C2/(SIMP*SFREQ)
      AL3=AL3*SIMP/SFREQ
      C4=C4/(SIMP*SFREQ)
      AL5=AL5*SIMP/SFREQ
      C6=C6/(SIMP*SFREQ)
      AL7=AL7*SIMP/SFREQ
      RL=RL*SIMP
      WRITE(6,29)
   29 FORMAT(5X,'THE   SCALED COMPONENT VALUES ARE',//)
      WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
   22 FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
     1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
      WRITE(6,18) RL,RS
   18 FORMAT(10X,'    RL=',E12.5,5X,'RS=',E12.5,//)
C
C     CALCULATION OF THE COEFFICIENTS OF
C     ------------------------------------
C
C      H(S)=K/(S7+A6*S6+A5*S5+A4*S4+A3*S3+A2*S2+A1*S1+A0)
C
      AA0=(RL+RS)/RL
C
C     FILTER SCALE FACTOR FROM DATA IS COEF
      COEF=AA0
      WRITE(6,37) COEF
   37 FORMAT(5X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
C
      AA1=(AL1+AL3+AL5+AL7)/RL+RS*(C2+C4+C6)
```

*Graphx, subroutine is given in Computer Program No. 1.

```
      AA2=(C4+C6)*(AL1+AL3+AL7*RS/RL)+C2*(AL1+(AL3+AL5+AL7)*RS/RL)+AL5*(
     *C6+C4*RS/RL)
      AA3=(AL1+AL3)*(AL5+AL7)*C4/RL+(AL3+AL5)*(C2*C6*RS+(C6*AL7+C2*AL1)/
     *RL)+C4*RS*(C2*AL3+AL5*C6)+AL1*AL7*(C2+C6)/RL
      AA4=(AL1+AL7*RS/RL)*(AL5*C6*(C2+C4)+AL3*C2*(C4+C6))+C4*AL5*AL3*(C6
     *+C2*RS/RL)
      AA5=AL1*(C2*AL3*(C4*(AL5+AL7)+C6*AL7)+AL5*C6*AL7*(C2+C4))/RL+AL3*C
     *4*AL5*C6*(AL7/RL+C2*RS)
      AA6=C2*AL3*C4*AL5*C6*(AL7*RS/RL+AL1)
      AA7=(AL1*C2*AL3*C4*AL5*C6*AL7)/RL
C
      A(1)=1.0D0
      A(2)=AA6/AA7
      A(3)=AA5/AA7
      A(4)=AA4/AA7
      A(5)=AA3/AA7
      A(6)=AA2/AA7
      A(7)=AA1/AA7
      A(8)=AA0/AA7
C
      COEF1=COEF/AA7
      WRITE(6,41)
   41 FORMAT(5X,'THE COEF.OF    H(S)=K/(S7+A6*S6+A5*S5+A4*S4+A3*S3+A2*S2+
     1A1*S1+A0)    ARE',/)
      WRITE(6,40) A(2),A(3),A(4),A(5),A(6),A(7),A(8)
   40 FORMAT(5X,'A6=',E12.5,3X,'A5=',E12.5,3X,'A4=',E12.5,3X,'A3=',E12.5
     1,3X,'A2=',E12.5,3X,'A1=',E12.5,3X,'A0=',E12.5,/)
      WRITE(6,43) COEF1
   43 FORMAT(5X,'K=',E12.5,//)
C
C     CALCULATION OF THE COEF. OF THE EQN.
C     ----------------------------------------------
C                 H(S)=  K/(S2+A1*S+B1)(S2+A2*S+B2)(S2+A3*S+B3)(S+B4)
C
C
      NDEG=7
      CALL ZPOLR(A,NDEG,Y,IER)
      P1=-2.0D0*REAL(Y(1))
      F1=REAL(Y(1))**2+AIMAG(Y(1))**2
      P2=-2.0D0*REAL(Y(3))
      F2=REAL(Y(3))**2+AIMAG(Y(3))**2
      P3=-2.0D0*REAL(Y(5))
      F3=REAL(Y(5))**2+AIMAG(Y(5))**2
      P4=-REAL(Y(7))
      PP=1.0D0/AA7
      WRITE(6,42)
   42 FORMAT(5X,'THE COEF. OF H(S) ARE.....
     1        H(S)=K/(S2+A1*S+B1)(S2+A2*S+B2)(S2+A3*S+B3)(S+B4)',//)
      WRITE(6,60) P1,F1,P2,F2,P3,F3,P4,COEF1
   60 FORMAT(1X,'A1=',E12.5,1X,'B1=',E12.5,2X,'A2=',E12.5,1X,'B2=',E12.5
     1,2X,'A3=',E12.5,1X,'B3=',E12.5,2X,'A4=',E12.5,3X,'K=',E12.5,//)
C
C     CALCULATION OF THE COEF OF H(Z)  WITH SAMPLING PERIOD OF T
C     ---------------------------------------------------------------
C
C                -1  7          -1         -1       -2         -1      -2        -1      -2
C     H(Z)=K*(1+Z  ) /(1+A4*Z  )(1+A1*Z  +B1*Z  )(1+A2*Z  +B2*Z)(1+A3Z  +B3Z  )
C
C
      F=2.0D0/T
      FF=F*F
      B1=2.0D0*(F1-FF)/(FF+P1*F+F1)
      D1=(FF-P1*F+F1)/(FF+P1*F+F1)
      B2=2.0D0*(F2-FF)/(FF+P2*F+F2)
      D2=(FF-P2*F+F2)/(FF+P2*F+F2)
      B3=2.0D0*(F3-FF)/(FF+P3*F+F3)
      D3=(FF-P3*F+F3)/(FF+P3*F+F3)
      B4=(P4-F)/(P4+F)
      COEF2=COEF1/((P4+F)*(FF+F*P1+F1)*(FF+F*P2+F2)*(FF+F*P3+F3))
      WRITE(6,66)
   66 FORMAT(5X,'THE COEF. OF Z**-1 AND Z**-2 IN QUADRATURE FORM AND TOT
     1AL MULTLYING FACTOR K ARE...',/)
      WRITE(6,60) B1,D1,B2,D2,B3,D3,B4,COEF2
```

```
C
C     INPUT IN TIME DOMAIN IS     UP
      UP=1.0D0
      UPP=UP*CCEF2
C
C     FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
C     FREQUENCY INCREMENT
      DLTAW=OMGAC/50.D0
C
C     INITIAL VALUES IN FREQUENCY DOMAIN
C
      W=0.0D0
C
C     ITTERATION IN THE FREQUENCY DOMAIN
C     ----------------------------------
C
      DO 110 J=1,98
C
C     INITIAL VALUES IN TIME DOMAIN
C
      UP1=UPP
      TT=0.D0
      H=DCMPLX(0.0D0,0.0D0)
      X1=0.D0
      X2=0.0D0
      X3=0.0D0
      X4=0.D0
      X5=0.0D0
      X6=0.D0
      X7=0.D0
C
C     ITTERATION IN THE TIME DOMAIN
C     -----------------------------
C
      DO 100 I=1,500
      V1=UP1+X1
      V2=V1+X2
      V3=V2+X3
      V=V3+X4
      X1=UP1-B4*V1
      X2=2*V1-B1*V2+X5
      X3=2*V2-B2*V3+X6
      X4=2*V3-B3*V+X7
      X5=V1-D1*V2
      X6=V2-D2*V3
      X7=V3-D3*V
      UP1=0.D0
      W1=W*TT
      WT=DCMPLX(0.0D0,-W1)
      Z=CDEXP(WT)
      H=H+V*Z
      TT=TT+T
  100 CONTINUE
C
      DA=CDABS(H)
C     ARRANGING THE OUTPUT DATA
      DATA(J,1)=W
      DATA(J,2)=DA
      DATA(J,3)=0.0D0
      W=W+DLTAW
  110 CONTINUE
C
      WRITE(6,54)
   54 FORMAT('1')
      WRITE(6,55)
   55 FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DATA,98,4HFREQ,4HMAGN)
   20 FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      STOP
      END
```

Reference

[1] Anatol I. Zverev, Handbook of Filter Synthesis, 1967, published by John Wiley and Sons, Inc.

APPENDIX 2

A - Computer Program* No. 5. Program to calculate the rms error due to truncation in the number of bits of wave digital filter multiplier coefficients.

```
C                    **** FREQUENCY RESPONCE ****
C
C        ROOT MEAN SQUARE ERROR DUE TO TRUNCATION IN NO. OF BITS *****
C        FOR SIMPLE SECTION WAVE DIGITAL FILTER
C
C
         IMPLICIT REAL*8 (A-H,O-Z)
         DIMENSION DATA(210,3)
         DIMENSION DATA3(210,3)
C
C
C         **********.5DB CHEBECHEV LOW-PASS FILTER WITH RS=1.0   ********
C        COMPONENT VALUES
C        INDUCTANCE AND CAPACITANCE VALUES IN HENRIES,AND FARADS
C        NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C        WITH   RL=1
C
         RS=1.0D0
         AL1=1.7896D0
         C2=1.2961D0
         AL3=2.7177D0
         C4=1.3848D0
         AL5=2.7177D0
         C6=1.2961D0
         AL7=1.7896D0
         RL=1.0D0
         WRITE(6,18) RL,RS
 18      FORMAT(10X,'     RL=',F6.4,5X,'RS=',F6.4,/)
         WRITE(6,22)   AL1,C2,AL3,C4,AL5,C6,AL7
 22      FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
        1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
C
C        FILTER SCALE FACTOR IS COEF
         COEF=(RL+RS)/RL
         WRITE(6,37) COEF
 37      FORMAT(7X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
C
C        SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
         OMGAC=1.0D0
C
C        SAMPLING PERIOD IS DLTAT
         DLTAT=1.0D0
C
C        FREQUENCY SCALING WITH PREWARPING AS WELL AS TAKING INTO ACOUNT
C        THE EFFECT OF SAMPLING TIME IS  SCALE
         SCALE=1.0D0/DTAN(OMGAC*DLTAT/2)
         AL1=AL1*SCALE
         C2=C2*SCALE
         AL3=AL3*SCALE
         C4=C4*SCALE
         AL5=AL5*SCALE
         C6=C6*SCALE
         AL7=AL7*SCALE
C
C        CALCULATION TO FIND THE TERMINATING RESISTANCE OF EACH ELEMENT
C        AND WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
C        WITH NO DELAY FREE PATH ON PORT TWO
C        ------------------------------------------------------------------
         R1=RS
         R2=R1+AL1
         SIGMA1=R1/R2
         G2=1.0D0/R2
         G3=G2+C2
         SIGMA2=G2/G3
         R3=1.0D0/G3
         R4=R3+AL3
         SIGMA3=R3/R4
         G4=1.0D0/R4
         G5=G4+C4
         SIGMA4=G4/G5
         R5=1.0D0/G5
         R6=R5+AL5
         SIGMA5=R5/R6
```

*Subroutine Graphx is given in Computer Program No. 1 of Appendix 1.

```
      G6=1.0D0/R6
      G7=G6+C6
      SIGMA6=G6/G7
      R7=1.0D0/G7
      R8=R7+AL7
      SIGMA7=R7/R8
      PHI=(RL-R8)/(RL+R8)
C
      WRITE(6,42)
 42   FORMAT(5X,'THE WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS ARE....
     *',//)
C
      WRITE(6,10) SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PHI
 10   FORMAT(/,4X,'SIGMA1=',F6.4,4X,'SIGMA2=',F6.4,4X,'SIGMA3=',F6.4,4X,
     1'SIGMA4=',F6.4,4X,'SIGMA5=',F6.4,4X,'SIGMA6=',F6.4,4X,'SIGMA7=',F6
     2.4,4X,'PHI=',F6.4,/)
C
C     INPUT IN TIME DOMAIN IS     AS
      AS=1.0D0
      UP=AS*COEF
C
C
C     CALCULATION OF THE FREQUENCY RESPONCE OF THE
C     FILTER WITH PRACTICALLY INFINITE PRECESION
C     IN ORDER TO SET THE REFFERENCE DATA
C
      CALL FILTER(UP,SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PH
     *I,DATA,OMGAC,DLTAT)
C
      DO 220 JJ=1,20
      NOBTS=JJ
C
C
C     CALCULATION OF THE ERROR IN THE FREQUENCY DOMAIN DUE
C     TO QUANTIZATION OF THE WAVE DIGITAL FILTER PARAMETERS TO THE
C     REQUIRED NO. OF BITS
C
      CALL FILERR(UP,SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PH
     *I,DATA,NOBTS,ER,OMGAC,DLTAT)
C
C     OUTPUT DATA SET UP
      DATA3(JJ,1)=JJ
      DATA3(JJ,2)=ER
      DATA3(JJ,3)=0.0D0
C
220   CONTINUE
C
C
C     PLOT OF THE R.M.S. ERROR DUE TO QUANTIZATION IN THE NO OF
C     BITS  OF WAVE DIGITAL FILTER PARAMETERS VERSUS THE NO. OF BITS
C
      WRITE(6,54)
 54   FORMAT('1')
      WRITE(6,55)
 55   FORMAT(20X,'NO. OF BITS',15X,'MEAN SQUAR ERROR',/)
      WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,20)
      CALL GRAPHX(DATA3,20,4HNBIT,4HMSER)
 20   FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      STOP
      END
```

FILTER

```
      SUBROUTINE FILTER(UP,SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIG
     *MA7,PHI,DATA,OMGAC,DLTAT)
C
C     SUBROUTINE FILTER TAKES THE VALUE OF THE INPUT IN THE TIME
C     DOMAIN AND PLOTS THE MAGNITUDE VERS. FREQUENCY CURVE
C     OF THE WAVE DIGITAL FILTER
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 H2,W1,Z
      DIMENSION DATA(210,3),DT(210,3)
C
C     FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
C     FREQUENCY INCREMENT
      DLW=OMGAC/50
C
C     INITIAL VALUES IN THE FREQUENCY DOMAIN
      W=0.0D0
C     ITTERATION IN THE FREQUENCY DOMAIN
C     -----------------------------------
      DO 110 J=1,98
C
C     INITIAL VALUES IN TIME DOMAIN
C
      H2=DCMPLX(0.0D0,0.0D0)
      TT=0.0D0
      X11=0.0D0
      X13=0.0D0
      X14=0.0D0
      X23=0.0D0
      X24=0.0D0
      X33=0.0D0
      X34=0.0D0
      X43=0.0D0
      X44=0.0D0
      X53=0.0D0
      X54=0.0D0
      X63=0.0D0
      X64=0.0D0
      X72=0.0D0
      X73=0.0D0
      X74=0.0D0
      A11=UP
C
C     ITTERATION IN THE TIME DOMAIN
C     ------------------------------
      DO 100 I=1,500
      B12=A11+X11-X23+SIGMA1*(X23-X14)
      B22=X33+SIGMA2*(B12+X14-X33-X24)
      B32=B22+X24-X43+SIGMA3*(X43-X34)
      B42=X53+SIGMA4*(B32+X34-X53-X44)
      B52=B42+X44-X63+SIGMA5*(X63-X54)
      B62=X73+SIGMA6*(B52+X54-X73-X64)
      B72=B62+X64-X72+SIGMA7*(X72-X74)
      A72=B72*PHI
      B71=B62+SIGMA7*(A72-B62+X72-X73)
      B61=B71-B52+X73+SIGMA6*(B52-X63)
      B51=B42+SIGMA5*(B61-B42+X63-X53)
      B41=B51-B32+X53+SIGMA4*(B32-X43)
      B31=B22+SIGMA3*(B41-B22+X43-X33)
      B21=B31-B12+X33+SIGMA2*(B12-X23)
      B11=A11+SIGMA1*(B21-A11+X23-X13)
C
C     UPDATED VALUES
C
      X11=A11
      X13=B11
      X14=B12
      X23=B21
      X24=B22
      X33=B31
      X34=B32
      X43=B41
      X44=B42
```

```
      X53=B51
      X54=B52
      X63=B61
      X64=B62
      X72=A72
      X73=B71
      X74=B72
      A11=0.0D0
      WT=W*TT
      W1=DCMPLX(0.0D0,-WT)
      Z=CDEXP(W1)
      H2=H2+B72*Z
      TT=TT+DLTAT
C
  100 CONTINUE
      DB=CDABS(H2)*(1.0D0+PHI)/2
C
C     ARRANGIGNG THE OUTPUT DATA
      DATA(J,1)=W
      DATA(J,2)=DB
      DATA(J,3)=0.0D0
C
C     ARRANGING THE DATA FOR PLOT SUBROUTINE
      DT(J,1)=DATA(J,1)
      DT(J,2)=DATA(J,2)
      DT(J,3)=DATA(J,3)
C
      W=W+DLW
  110 CONTINUE
C
C     WRITING AND PLOTING THE FREQUENCY RESPONCE
      WRITE(6,54)
   54 FORMAT('1')
      WRITE(6,55)
   55 FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DT,98,4HFREQ,4HMAGN)
   20 FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      RETURN
      END
```

FILERR

```
      SUBROUTINE FILERR(UP,SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIG
     *MA7,PHI,DATA,NOBTS,ER,OMGAC,DLTAT)
C
C     SUBROUTINE FILERR TAKES THE ROOT MEAN SQUARE VALUE OF THE ERROR
C     BETWEEN UNTRUNCATED AND TRUNCATED VALUE OF THE OUT PUT IN THE
C     FREQUENCY DOMAIN  AND ALSO PLOTS THE FINITE PRECESION
C     FREQUENCY RESPONCE
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DATA2(210,3)
      DIMENSION DATA(210,3)
C
      WRITE(6,54)
 54   FORMAT('1')
      WRITE(6,21)
 21   FORMAT(10X,'THE UNTRUNCATED VALUES ARE')
      WRITE(6,10) SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PHI
 10   FORMAT(/,4X,'SIGMA1=',F6.4,4X,'SIGMA2=',F6.4,4X,'SIGMA3=',F6.4,4X,
     *'SIGMA4=',F6.4,4X,'SIGMA5=',F6.4,4X,'SIGMA6=',F6.4,4X,'SIGMA7=',F6
     *.4,4X,'PHI=',F6.4)
C
      A=SIGMA1
      B=SIGMA2
      C=SIGMA3
      D=SIGMA4
      E=SIGMA5
      P=SIGMA6
      G=SIGMA7
      H=PHI
C
      WRITE(6,33) NOBTS
 33   FORMAT(10X,'     NO OF BITS =',I2,/)
C
C     PERFORMING THE TRUNCATION PROCESS  TO REQUIRED NO. OF BITS ,
C     ' NOBTS'
      SIGMA1=TRUNC(A,NOBTS)
      SIGMA2=TRUNC(B,NOBTS)
      SIGMA3=TRUNC(C,NOBTS)
      SIGMA4=TRUNC(D,NOBTS)
      SIGMA5=TRUNC(E,NOBTS)
      SIGMA6=TRUNC(P,NOBTS)
      SIGMA7=TRUNC(G,NOBTS)
      PHI=TRUNC(H,NOBTS)
      WRITE(6,22)
 22   FORMAT(10X,'THE TRUNCATED VALUES ARE')
      WRITE(6,10) SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PHI
C
C     CALCULATION OF THE FREQUENCY RESPONCE WITH WAVE DIGITAL
C     PARAMETERS OF FINITE PRECESION
      CALL FILTER(UP,SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PH
     *I,DATA2,OMGAC,DLTAT)
C
C     CALCULATION OF ROOT MEAN SQUARE ERROR
      ER=0.0D0
      DO 213 KK=1,98
      ER=ER+(DATA(KK,2)-DATA2(KK,2))**2
 213  CONTINUE
      ER=ER/98.0D0
      ER=DSQRT(ER)
C
      SIGMA1=A
      SIGMA2=B
      SIGMA3=C
      SIGMA4=D
      SIGMA5=E
      SIGMA6=P
      SIGMA7=G
      PHI=H
C
      RETURN
      END
```

TRUNC

```
      FUNCTION TRUNC (VALUE,NOBTS)
C
C     FUNCTION TRUNC  CONVERTS THE VALUE INTC BINOMIAL FLOATING
C     POINT ARITHMATIC AND THEN TRUNCATES THE VALUE TC THE DESIRED
C     NO. OF BITS AND THEN CONVERTS BACK THE VALUE INTO DECIMAL
C     FIXED POINT ARITHMATIC
C     MAX. NO. OF BITS(NOBTS)  MUST NOT EXCEED  30
C     AND THE ABSOLUTE VALUE OF 'VALUE' MUST BE LESS THAN 1.0D9
C     AND THE ABSOLUTE VALUE OF 'VALUE' MUST BE GREATER THAN 1.0D-15
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION MLTPL (30)
      DIMENSION MLT(30)
C
      A=DABS(VALUE)
      JJ=0
      IF(A.LE.1.0D-15) TRUNC=0.0D0
      IF(A.LE.1.0D-15) RETURN
      DSIGN=VALUE/DABS(VALUE)
      IF(A.GE.1.0D0) GO TO 1
      FRACT=A
      C=0.0D0
      NBITS=NOBTS
C     THE MAGNITUDE OF A IS GREATER THAN 1
C
      GO TO 20
    1 N=A
      B=N
      FRACT=A-B
      NONT=0
      JJ=1
    2 CONTINUE
      M=N
      IF(N.EQ.0) GO TO 88
      N=N/2
      MM=M-2*N
      IF(MM.EQ.0) GO TO 23
      NONT=NONT+1
      MLT(NONT)=1
      GO TO 2
   23 NONT=NONT+1
      MLT(NONT)=0
      GO TO 2
   88 CONTINUE
      IF(NOBTS.LE.NONT) GO TO 68
      C=B
      NBITS=NOBTS-NONT
      GO TO 20
   68 NP=NONT-NOBTS+1
      C=0.0D0
      DO 16 I=NP,NONT
      C=C+MLT(I)*2**(I-1)
   16 CONTINUE
      TRUNC=C*DSIGN
      RETURN
   20 CONTINUE
      DO 6 I=1,NBITS
      FRACT=FRACT*2
      IF (FRACT.GE.1.0D0) GO TO 7
      MLTPL(I)=0
      IF((JJ+MLTPL(I)).EQ.0) NBITS=NBITS+1
      GO TO 6
    7 CONTINUE
      FRACT=FRACT-1.0D0
      MLTPL(I)=1
      JJ=1
    6 CONTINUE
      RMANT=0.0D0
      DO 8 I=1,NBITS
      RMANT=RMANT+MLTPL(I)*2.0D0**(-I)
    8 CONTINUE
      TRUNC=DSIGN*(C+RMANT)
      RETURN
      END
```

B - Computer Program* No. 6. Program to calculate the rms error due to truncation in the number of bits of complex wave digital filter multiplier coefficients.

```
C                    **** FREQUENCY RESPONCE ****
C
C      ROOT MEAN SQUARE ERROR DUE TO TRUNCATION IN NO. OF BITS *****
C      FOR COMPLEX WAVE DIGITAL FILTER
C
       IMPLICIT REAL*8 (A-H,O-Z)
       DIMENSION DATA(210,3)
       DIMENSION DATA3(210,3)
C
C       ***********.5DB CHEBECHEV LOW-PASS FILTER WITH RS=1.0  ********
C      COMPONENT VALUES
C      INDUCTANCE AND CAPACITANCE VALUES IN HENRIES,AND FARADS
C      NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C      WITH    RL=1
C
       RS=1.0D0
       AL1=1.7896D0
       C2=1.2961D0
       AL3=2.7177D0
       C4=1.3848D0
       AL5=2.7177D0
       C6=1.2961D0
       AL7=1.7896D0
       RL=1.0D0
       WRITE(6,18) RL,RS
 18    FORMAT(10X,'     RL=',F6.4,5X,'RS=',F6.4,/)
       WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
 22    FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
      1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
C
C      FILTER SCALE FACTOR IS COEF
       COEF=(RL+RS)/RL
       WRITE(6,37) COEF
 37    FORMAT(7X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
C
C      SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
       OMGAC=1.0D0
C
C      SAMPLING PERIOD IS DLTAT
       DLTAT=1.0D0
C
C      FREQUENCY SCALING WITH PREWARPING AS WELL AS TAKING INTO ACOUNT
C      THE EFFECT OF SAMPLING TIME IS  SCALE
       SCALE=1.0D0/DTAN(OMGAC*DLTAT/2)
       AL1=AL1*SCALE
       C2=C2*SCALE
       AL3=AL3*SCALE
       C4=C4*SCALE
       AL5=AL5*SCALE
       C6=C6*SCALE
       AL7=AL7*SCALE
C
C      CALCULATION TO FIND THE TERMINATING RESISTANCE OF EACH ELEMENT
C      AND COMPLEX WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
C      WITH NO DELAY FREE PATH ON PORT TWO
C      -----------------------------------------------------------------
       R1=RS
       R2=(R1+AL1)/(1.0D0+C2*(R1+AL1))
       R3=(R2+AL3)/(1.0D0+C4*(R2+AL3))
       R4=(R3+AL5)/(1.0D0+C6*(R3+AL5))
       R5=R4+AL7
       BETA11=(2*R1+AL1+R1*R1*C2-AL1*AL1*C2)/((R1+AL1)*(1.0D0+R1*C2+AL1*C
      *2))
       GAMA11=R1/((R1+AL1)*(1.0D0+R1*C2+AL1*C2))
       BETA12=(R1*R1*C2+2*R1*AL1*C2-AL1+AL1*AL1*C2)/((R1+AL1)*(1.0D0+R1*C
      *2+AL1*C2))
       GAMA12=(R1*R1*C2-AL1-AL1*AL1*C2)/((R1+AL1)*(1.0D0+R1*C2+AL1*C2))
       BETA13=1.0D0/(1.0D0+R1*C2+AL1*C2)
       BETA14=R1/(R1+AL1)
       BETA21=(2*R2+AL3+R2*R2*C4-AL3*AL3*C4)/((R2+AL3)*(1.0D0+R2*C4+AL3*C
      *4))
       GAMA21=R2/((R2+AL3)*(1.0D0+R2*C4+AL3*C4))
       BETA22=(R2*R2*C4+2*R2*AL3*C4-AL3+AL3*AL3*C4)/((R2+AL3)*(1.0D0+R2*C
```

*Subroutine Graphx is given in Computer Program No. 1 of Appendix 1 and Function Trunc is given in Computer Program No. 5.

```
     *4+AL3*C4))
      GAMA22=(R2*R2*C4-AL3-AL3*AL3*C4)/((R2+AL3)*(1.0D0+R2*C4+AL3*C4))
      BETA23=1.0D0/(1.0D0+R2*C4+AL3*C4)
      BETA24=R2/(R2+AL3)
      BETA31=(2*R3+AL5+R3*R3*C6-AL5*AL5*C6)/((R3+AL5)*(1.0D0+R3*C6+AL5*C
     *6))
      GAMA31=R3/((R3+AL5)*(1.0D0+R3*C6+AL5*C6))
      BETA32=(R3*R3*C6+2*R3*AL5*C6-AL5+AL5*AL5*C6)/((R3+AL5)*(1.0D0+R3*C
     *6+AL5*C6))
      GAMA32=(R3*R3*C6-AL5-AL5*AL5*C6)/((R3+AL5)*(1.0D0+R3*C6+AL5*C6))
      BETA33=1.0D0/(1.0D0+R3*C6+AL5*C6)
      BETA34=R3/(R3+AL5)
      SIGM41=R4/R5
      PHI=(RL-R5)/(RL+R5)
C
      WRITE(6,42)
   42 FORMAT(5X,'THE COMPLEX WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
     * ARE...',//)
C
      WRITE(6,88) BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14
   88 FORMAT(1X,'BETA11=',E12.5,1X,'GAMA11=',E12.5,1X,'BETA12=',E12.5,1X
     *,'GAMA12=',E12.5,1X,'BETA13=',E12.5,1X,'BETA14=',E12.5,//)
      WRITE(6,89) BETA21,GAMA21,BETA22,GAMA22,BETA23,BETA24
   89 FORMAT(1X,'BETA21=',E12.5,1X,'GAMA21=',E12.5,1X,'BETA22=',E12.5,1X
     *,'GAMA22=',E12.5,1X,'BETA23=',E12.5,1X,'BETA14=',E12.5,//)
      WRITE(6,90) BETA31,GAMA31,BETA32,GAMA32,BETA33,BETA34
   90 FORMAT(1X,'BETA31=',E12.5,1X,'GAMA31=',E12.5,1X,'BETA32=',E12.5,1X
     *,'GAMA32=',E12.5,1X,'BETA33=',E12.5,1X,'BETA34=',E12.5,//)
      WRITE(6,91) SIGM41,PHI
   91 FORMAT(5X,'SIGM41=',E12.5,5X,'PHI=',E12.5,//)
C
C     INPUT IN TIME DOMAIN IS     AS
      AS=1.0D0
      UP=AS*COEF
C
C     CALCULATION OF THE FREQUENCY RESPONCE OF THE
C     FILTER WITH PRACTICALLY INFINITE PRECESION
C     IN ORDER TO SET THE REFFERENCE DATA
C
      CALL FILTER(UP,BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14,BETA21,GA
     *MA21,BETA22,GAMA22,BETA23,BETA24,BETA31,GAMA31,BETA32,GAMA32,BETA3
     *3,BETA34,SIGM41,PHI,DATA,OMGAC,DLTAT)
C
      DO 220 JJ=1,20
      NOBTS=JJ
C
C     CALCULATION OF THE ERROR IN THE FREQUENCY DOMAIN DUE
C     TO QUANTIZATION OF THE WAVE DIGITAL FILTER PARAMETERS TO THE
C     REQUIRED NO. OF BITS
C
      CALL FILERR(UP,BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14,BETA21,GA
     *MA21,BETA22,GAMA22,BETA23,BETA24,BETA31,GAMA31,BETA32,GAMA32,BETA3
     *3,BETA34,SIGM41,PHI,DATA,NOBTS,ER,OMGAC,DLTAT)
C
C     OUTPUT DATA SET UP
      DATA3(JJ,1)=JJ
      DATA3(JJ,2)=ER
      DATA3(JJ,3)=0.0D0
C
  220 CONTINUE
C
C     PLOT OF THE R.M.S. ERROR DUE TO QUANTIZATION IN THE NO OF
C     BITS  OF WAVE DIGITAL FILTER PARAMETERS VERSUS THE NO. OF BITS
C
      WRITE(6,54)
   54 FORMAT('1')
      WRITE(6,55)
   55 FORMAT(20X,'NO. OF BITS',15X,'MEAN SQUAR ERROR',/)
      WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,20)
      CALL GRAPHX(DATA3,20,4HNBIT,4HMSER)
   20 FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      STOP
      END
```

FILTER

```
      SUBROUTINE FILTER(UP,BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14,BET
     *A21,GAMA21,BETA22,GAMA22,BETA23,BETA24,BETA31,GAMA31,BETA32,GAMA32
     *,BETA33,BETA34,SIGM41,PHI,DATA,OMGAC,DLTAT)
C
C     SUBROUTINE FILTER TAKES THE VALUE OF THE INPUT IN THE TIME
C     DOMAIN AND PLOTS THE MAGNITUDE VERS. FREQUENCY CURVE
C     OF THE COMPLEX WAVE DIGITAL FILTER
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 H2,W1,Z
      DIMENSION DATA(210,3),DT(210,3)
C
C     FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
C     FREQUENCY INCREMENT
      DLTAW=OMGAC/50.D0
C
C     INITIAL VALUES IN THE FREQUENCY DOMAIN
      W=0.0D0
C     ITTERATION IN THE FREQUENCY DOMAIN
C     ------------------------------------
      DO 110 J=1,98
C
C     INITIAL VALUES IN TIME DOMAIN
C
      H2=DCMPLX(0.0D0,0.0D0)
      TT=0.0D0
      X11=0.0D0
      X12=0.0D0
      X13=0.0D0
      X14=0.0D0
      X15=0.0D0
      X16=0.0D0
      X17=0.0D0
      X18=0.0D0
      X21=0.0D0
      X22=0.0D0
      X23=0.0D0
      X24=0.0D0
      X25=0.0D0
      X26=0.0D0
      X27=0.0D0
      X28=0.0D0
      X31=0.0D0
      X32=0.0D0
      X33=0.0D0
      X34=0.0D0
      X35=0.0D0
      X36=0.0D0
      X37=0.0D0
      X38=0.0D0
      X41=0.0D0
      X42=0.0D0
      X43=0.0D0
      X44=0.0D0
      A11=UP
C
C     ITTERATION IN THE TIME DOMAIN
C     ------------------------------
      DO 100 I=1,500
      B12=BETA13*(A11+2*X11+X12)+BETA12*X13+GAMA12*X14-BETA11*X17-GAMA11
     **X18
      A21=B12
      B22=BETA23*(A21+2*X21+X22)+BETA22*X23+GAMA22*X24-BETA21*X27-GAMA21
     **X28
      A31=B22
      B32=BETA33*(A31+2*X31+X32)+BETA32*X33+GAMA32*X34-BETA31*X37-GAMA31
     **X38
      A41=B32
      B42=A41+X41-X42+SIGM41*(X42-X44)
      A42=B42*PHI
      B41=A41+SIGM41*(A42-A41+X42-X43)
      A32=B41
      B31=BETA34*(A32+2*X33+X34)-BETA32*X31-GAMA32*A31-BETA31*X35-GAMA31
```

```
     **X36
C
C     UPDATED VALUES
C
      A22=B31
      B21=BETA24*(A22+2*X23+X24)-BETA22*X21-GAMA22*A21-BETA21*X25-GAMA21
     **X26
      A12=B21
      B11=BETA14*(A12+2*X13+X14)-BETA12*X11-GAMA12*A11-BETA11*X15-GAMA11
     **X16
C     UPDATED EQUATIONS
      X12=X11
      X11=A11
      X14=X13
      X13=A12
      X16=X15
      X15=B11
      X18=X17
      X17=B12
      X22=X21
      X21=A21
      X24=X23
      X23=A22
      X26=X25
      X25=B21
      X28=X27
      X27=B22
      X32=X31
      X31=A31
      X34=X33
      X33=A32
      X36=X35
      X35=B31
      X38=X37
      X37=B32
      X41=A41
      X42=A42
      X43=B41
      X44=B42
      A11=0.0D0
      WT=W*TT
      W1=DCMPLX(0.0D0,-WT)
      Z=CDEXP(W1)
      H2=H2+B42*Z
      TT=TT+DLTAT
C
 100  CONTINUE
      DB=CDABS(H2)*(1.0D0+PHI)/2
C
C     ARRANGIGNG THE OUTPUT DATA
      DATA(J,1)=W
      DATA(J,2)=DB
      DATA(J,3)=0.0D0
C
C     ARRANGING THE DATA FOR PLOT SUBROUTINE
      DT(J,1)=DATA(J,1)
      DT(J,2)=DATA(J,2)
      DT(J,3)=DATA(J,3)
C
      W=W+DLTAW
 110  CONTINUE
C
C     WRITING AND PLOTING THE FREQUENCY RESPONCE
      WRITE(6,54)
 54   FORMAT('1')
      WRITE(6,55)
 55   FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DT,98,4HFREQ,4HMAGN)
 20   FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      RETURN
      END
```

```
      SUBROUTINE FILERR(UP,BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14,BET
     *A21,GAMA21,BETA22,GAMA22,BETA23,BETA24,BETA31,GAMA31,BETA32,GAMA32
     *,BETA33,BETA34,SIGM41,PHI,DATA,NOBTS,ER,OMGAC,DLTAT)
C
C     SUBROUTINE FILERR TAKES THE ROOT MEAN SQUARE VALUE OF THE ERROR
C     BETWEEN UNTRUNCATED AND TRUNCATED VALUE OF THE OUT PUT IN THE
C     FREQUENCY DOMAIN  AND ALSO PLOTS THE FINITE PRECESION
C     FREQUENCY RESPONCE
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DATA2(210,3)
      DIMENSION DATA(210,3)
C
      WRITE(6,54)
   54 FORMAT('1')
      WRITE(6,21)
   21 FORMAT(10X,'THE UNTRUNCATED VALUES ARE')
      WRITE(6,88) BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14
   88 FORMAT(1X,'BETA11=',E12.5,1X,'GAMA11=',E12.5,1X,'BETA12=',E12.5,1X
     *,'GAMA12=',E12.5,1X,'BETA13=',E12.5,1X,'BETA14=',E12.5,//)
      WRITE(6,89) BETA21,GAMA21,BETA22,GAMA22,BETA23,BETA24
   89 FORMAT(1X,'BETA21=',E12.5,1X,'GAMA21=',E12.5,1X,'BETA22=',E12.5,1X
     *,'GAMA22=',E12.5,1X,'BETA23=',E12.5,1X,'BETA14=',E12.5,//)
      WRITE(6,90) BETA31,GAMA31,BETA32,GAMA32,BETA33,BETA34
   90 FORMAT(1X,'BETA31=',E12.5,1X,'GAMA31=',E12.5,1X,'BETA32=',E12.5,1X
     *,'GAMA32=',E12.5,1X,'BETA33=',E12.5,1X,'BETA34=',E12.5,//)
      WRITE(6,91) SIGM41,PHI
   91 FORMAT(5X,'SIGM41=',E12.5,5X,'PHI=',E12.5,//)
C
      AA1=BETA11
      AA2=GAMA11
      AA3=BETA12
      AA4=GAMA12
      AA5=BETA13
      AA6=BETA14
      BB1=BETA21
      BB2=GAMA21
      BB3=BETA22
      BB4=GAMA22
      BB5=BETA23
      BB6=BETA24
      CC1=BETA31
      CC2=GAMA31
      CC3=BETA32
      CC4=GAMA32
      CC5=BETA33
      CC6=BETA34
      DD1=SIGM41
      H=PHI
C
      WRITE(6,33) NOBTS
   33 FORMAT(10X,'     NO OF BITS =',I2,/)
C
C     PERFORMING THE TRUNCATION PROCESS  TO REQUIRED NO. OF BITS ,
C     ' NOBTS'
      BETA11=TRUNC(AA1,NOBTS)
      GAMA11=TRUNC(AA2,NOBTS)
      BETA12=TRUNC(AA3,NOBTS)
      GAMA12=TRUNC(AA4,NOBTS)
      BETA13=TRUNC(AA5,NOBTS)
      BETA14=TRUNC(AA6,NOBTS)
      BETA21=TRUNC(BB1,NOBTS)
      GAMA21=TRUNC(BB2,NOBTS)
      BETA22=TRUNC(BB3,NOBTS)
      GAMA22=TRUNC(BB4,NOBTS)
      BETA23=TRUNC(BB5,NOBTS)
      BETA24=TRUNC(BB6,NOBTS)
      BETA31=TRUNC(CC1,NOBTS)
      GAMA31=TRUNC(CC2,NOBTS)
      BETA32=TRUNC(CC3,NOBTS)
      GAMA32=TRUNC(CC4,NOBTS)
      BETA33=TRUNC(CC5,NOBTS)
      BETA34=TRUNC(CC6,NOBTS)
```

```
      SIGM41=TRUNC(DD1,NOBTS)
      PHI=TRUNC(H,NOBTS)
      WRITE(6,22)
   22 FORMAT(10X,'THE TRUNCATED VALUES ARE')
      WRITE(6,88) BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14
      WRITE(6,89) BETA21,GAMA21,BETA22,GAMA22,BETA23,BETA24
      WRITE(6,90) BETA31,GAMA31,BETA32,GAMA32,BETA33,BETA34
      WRITE(6,91) SIGM41,PHI
C
C     CALCULATION OF THE FREQUENCY RESPONCE WITH WAVE DIGITAL
C     PARAMETERS OF FINITE PRECESION
      CALL FILTER(UP,BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14,BETA21,GA
     *MA21,BETA22,GAMA22,BETA23,BETA24,BETA31,GAMA31,BETA32,GAMA32,BETA3
     *3,BETA34,SIGM41,PHI,DATA2,OMGAC,DLTAT)
C
C     CALCULATION OF ROOT MEAN SQUARE ERROR
      ER=0.0D0
      DO 213 KK=1,98
      ER=ER+(DATA(KK,2)-DATA2(KK,2))**2
  213 CONTINUE
      ER=ER/98.0D0
      ER=DSQRT(ER)
C
      BETA11=AA1
      GAMA11=AA2
      BETA12=AA3
      GAMA12=AA4
      BETA13=AA5
      BETA14=AA6
      BETA34=CC6
      BETA21=BB1
      GAMA21=BB2
      BETA22=BB3
      GAMA22=BB4
      BETA23=BB5
      BETA24=BB6
      BETA33=CC5
      GAMA32=CC4
      BETA32=CC3
      GAMA31=CC2
      BETA31=CC1
      SIGM41=DD1
      PHI=H
C
      RETURN
      END
```

C - <u>Computer Program</u>*<u> No. 7</u>. Program to calculate the rms error due to truncation in the number of bits of reduced parameter complex wave digital filter multiplier coefficients.

```
C                       **** FREQUENCY RESPONCE ****
C
C       ROOT MEAN SQUARE ERROR DUE TO TRUNCATION IN NO. OF BITS *****
C       FOR REDUCED PARAMETER COMPLEX WAVE DIGITAL FILTER
C
        IMPLICIT REAL*8 (A-H,O-Z)
        DIMENSION DATA(210,3)
        DIMENSION DATA3(210,3)
C
C        ***********.5DB CHEBECHEV LOW-PASS FILTER WITH RS=1.0   ********
C       COMPONENT VALUES
C       INDUCTANCE AND CAPACITANCE VALUES IN HENRIES,AND FARADS
C       NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C       WITH    RL=1
C
        RS=1.0D0
        AL1=1.7896D0
        C2=1.2961D0
        AL3=2.7177D0
        C4=1.3848D0
        AL5=2.7177D0
        C6=1.2961D0
        AL7=1.7896D0
        RL=1.0D0
        WRITE(6,18) RL,RS
   18   FORMAT(10X,'     RL=',F6.4,5X,'RS=',F6.4,/)
        WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
   22   FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
       1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
C
C       FILTER SCALE FACTOR IS COEF
        COEF=(RL+RS)/RL
        WRITE(6,37) COEF
   37   FORMAT(7X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
C
C       SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
        OMGAC=1.0D0
C
C       SAMPLING PERIOD IS DLTAT
        DLTAT=1.0D0
C
C       FREQUENCY SCALING WITH PREWARPING AS WELL AS TAKING INTO ACOUNT
C       THE EFFECT OF SAMPLING TIME IS  SCALE
        SCALE=1.0D0/DTAN(OMGAC*DLTAT/2)
        AL1=AL1*SCALE
        C2=C2*SCALE
        AL3=AL3*SCALE
        C4=C4*SCALE
        AL5=AL5*SCALE
        C6=C6*SCALE
        AL7=AL7*SCALE
C
C       CALCULATION TO FIND THE TERMINATING RESISTANCE OF EACH ELEMENT
C       AND REDUCED PARAMETER COMPLEX WAVE DIGITAL FILTER COEFFICIENTS
C       WITH NO DELAY FREE PATH ON PORT TWO
C       ----------------------------------------------------------------
        R1=RS
        ALFA1=AL1/(R1+AL1)
        BETA1=1.0D0/(1.0D0+R1*C2+AL1*C2)
        R2=AL1*BETA1/ALFA1
        ALFA2=AL3/(R2+AL3)
        BETA2=1.0D0/(1.0D0+R2*C4+AL3*C4)
        R3=AL3*BETA2/ALFA2
        ALFA3=AL5/(R3+AL5)
        BETA3=1.0D0/(1.0D0+R3*C6+AL5*C6)
        R4=AL5*BETA3/ALFA3
        R5=R4+AL7
        SIGM41=R4/R5
        PHI=(RL-R5)/(RL+R5)
        WRITE(6,42)
   42   FORMAT(5X,'THE REDUCED PARAMETER COMPLEX WAVE DIGITAL FILTER COEFF
```

*Subroutine Graphx is given in Computer Program No. 1 Appendix 1 and Function Trunc is given in Computer Program No. 5.

```
     *ICIENTS ARE...',//)
C
      WRITE(6,88) ALFA1,BETA1,ALFA2,BETA2,ALFA3,BETA3
 88   FORMAT(1X,' ALFA1=',E12.5,1X,'BETA1 =',E12.5,1X,'ALFA2 =',E12.5,1X
     *,'BETA2 =',E12.5,1X,'ALFA3 =',E12.5,1X,'BETA3 =',E12.5,//)
      WRITE(6,91) SIGM41,PHI
 91   FORMAT(5X,'SIGM41=',E12.5,5X,'PHI=',E12.5,//)
C
C     INPUT IN TIME DOMAIN IS     AS
      AS=1.000
      UP=AS*COEF
C
C     CALCULATION OF THE FREQUENCY RESPONCE OF THE
C     FILTER WITH PRACTICALLY INFINITE PRECESION
C     IN ORDER TO SET THE REFFERENCE DATA
C
      CALL FILTER(UP,ALFA1,BETA1,ALFA2,BETA2,ALFA3,BETA3,SIGM41,PHI,DATA
     *,OMGAC,DLTAT)
C
      DO 220 JJ=1,20
      NOBTS=JJ
C
C     CALCULATION OF THE ERROR IN THE FREQUENCY DOMAIN DUE
C     TO QUANTIZATION OF THE WAVE DIGITAL FILTER PARAMETERS TO THE
C     REQUIRED NO. OF BITS
C
      CALL FILERR(UP,ALFA1,BETA1,ALFA2,BETA2,ALFA3,BETA3,SIGM41,PHI,DATA
     *,NOBTS,ER,OMGAC,DLTAT)
C
C     OUTPUT DATA SET UP
      DATA3(JJ,1)=JJ
      DATA3(JJ,2)=ER
      DATA3(JJ,3)=0.000
C
 220  CONTINUE
C
C     PLOT OF THE R.M.S. ERROR DUE TO QUANTIZATION IN THE NO OF
C     BITS  OF WAVE DIGITAL FILTER PARAMETERS VERSUS THE NO. OF BITS
C
      WRITE(6,54)
 54   FORMAT('1')
      WRITE(6,55)
 55   FORMAT(20X,'NO. OF BITS',15X,'MEAN SQUAR ERROR',/)
      WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,20)
      CALL GRAPHX(DATA3,20,4HNBIT,4HMSER)
 20   FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      STOP
      END
```

FILTER

```
      SUBROUTINE FILTER(UP,ALFA1,BETA1,ALFA2,BETA2,ALFA3,BETA3,SIGM41,PH
     *I,DATA,OMGAC,DLTAT)
C
C     SUBROUTINE FILTER TAKES THE VALUE OF THE INPUT IN THE TIME
C     DOMAIN AND PLOTS THE MAGNITUDE VERS. FREQUENCY CURVE
C     OF THE REDUCED PARAMETER COMPLEX WAVE DIGITAL FILTER
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 H2,W1,Z
      DIMENSION DATA(210,3),DT(210,3)
C
C     FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
C     FREQUENCY INCREMENT
      DLTAW=OMGAC/50.D0
C
C     INITIAL VALUES IN THE FREQUENCY DOMAIN
      W=0.0D0
C
C     PREMULTIPLICATIONS
      GAMA1=2*ALFA1
      GAMA2=2*ALFA2
      GAMA3=2*ALFA3
      THETA1=ALFA1*BETA1
      THETA2=ALFA2*BETA2
      THETA3=ALFA3*BETA3
C
C     ITTERATION IN THE FREQUENCY DOMAIN
C     ----------------------------------
      DO 110 J=1,98
C
C     INITIAL VALUES IN TIME DOMAIN
C
      H2=DCMPLX(0.0D0,0.0D0)
      TT=0.0D0
      X11=0.0D0
      X12=0.0D0
      X13=0.0D0
      X14=0.0D0
      X15=0.0D0
      X16=0.0D0
      X17=0.0D0
      X18=0.0D0
      X21=0.0D0
      X22=0.0D0
      X23=0.0D0
      X24=0.0D0
      X25=0.0D0
      X26=0.0D0
      X27=0.0D0
      X28=0.0D0
      X31=0.0D0
      X32=0.0D0
      X33=0.0D0
      X34=0.0D0
      X35=0.0D0
      X36=0.0D0
      X37=0.0D0
      X38=0.0D0
      X41=0.0D0
      X42=0.0D0
      X43=0.0D0
      X44=0.0D0
      A11=UP
C
C     ITTERATION IN THE TIME DOMAIN
C     -----------------------------
      DO 100 I=1,500
      B12=X13+X14-X17+GAMA1*(X17-X14)+BETA1*(A11+2*X11+X12-X13-X14-X17-X
     *18)+THETA1*(X18-X13-X17+X14)
      A21=B12
      B22=X23+X24-X27+GAMA2*(X27-X24)+BETA2*(A21+2*X21+X22-X23-X24-X27-X
     *28)+THETA2*(X28-X23-X27+X24)
      A31=B22
```

```
      B32=X33+X34-X37+GAMA3*(X37-X34)+BETA3*(A31+2*X31+X32-X33-X34-X37-X
     *38)+THETA3*(X38-X33-X37+X34)
      A41=B32
      B42=A41+X41-X42+SIGM41*(X42-X44)
      A42=B42*PHI
      B41=A41+SIGM41*(A42-A41+X42-X43)
      A32=B41
      B31=A32-A31-X31+2*X33+X34-X35+ALFA3*(2*(A31-X33+X35)-A32-X34)+BETA
     *3*(A31+X31-X35-X36)+THETA3*(X31-A31-X35+X36)
      A22=B31
      B21=A22-A21-X21+2*X23+X24-X25+ALFA2*(2*(A21-X23+X25)-A22-X24)+BETA
     *2*(A21+X21-X25-X26)+THETA2*(X21-A21-X25+X26)
      A12=B21
      B11=A12-A11-X11+2*X13+X14-X15+ALFA1*(2*(A11-X13+X15)-A12-X14)+BETA
     *1*(A11+X11-X15-X16)+THETA1*(X11-A11-X15+X16)
C
C     UPDATED EQUATIONS
      X12=X11
      X11=A11
      X14=X13
      X13=A12
      X16=X15
      X15=B11
      X18=X17
      X17=B12
      X22=X21
      X21=A21
      X24=X23
      X23=A22
      X26=X25
      X25=B21
      X28=X27
      X27=B22
      X32=X31
      X31=A31
      X34=X33
      X33=A32
      X36=X35
      X35=B31
      X38=X37
      X37=B32
      X41=A41
      X42=A42
      X43=B41
      X44=B42
      A11=0.0D0
C
      WT=W*TT
      W1=DCMPLX(0.0D0,-WT)
      Z=CDEXP(W1)
      H2=H2+B42*Z
      TT=TT+DLTAT
C
  100 CONTINUE
      DB=CDABS(H2)*(1.0D0+PHI)/2
C
C     ARRANGIGNG THE OUTPUT DATA
      DATA(J,1)=W
      DATA(J,2)=DB
      DATA(J,3)=0.0D0
C
C     ARRANGING THE DATA FOR PLOT SUBROUTINE
      DT(J,1)=DATA(J,1)
      DT(J,2)=DATA(J,2)
      DT(J,3)=DATA(J,3)
C
      W=W+DLTAW
  110 CONTINUE
C
C     WRITING AND PLOTING THE FREQUENCY RESPONCE
      WRITE(6,54)
   54 FORMAT('1')
      WRITE(6,55)
   55 FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DT,98,4HFREQ,4HMAGN)
   20 FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      RETURN
      END
```

FILERR

```
      SUBROUTINE FILERR(UP,ALFA1,BETA1,ALFA2,BETA2,ALFA3,BETA3,SIGM41,PH
     *I,DATA,NOBTS,ER,OMGAC,DLTAT)
C
C     SUBROUTINE FILERR TAKES THE ROOT MEAN SQUARE VALUE OF THE ERROR
C     BETWEEN UNTRUNCATED AND TRUNCATED VALUE OF THE OUT PUT IN THE
C     FREQUENCY DOMAIN  AND ALSO PLOTS THE FINITE PRECESION
C     FREQUENCY RESPONCE
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DATA2(210,3)
      DIMENSION DATA(210,3)
C
      WRITE(6,54)
 54   FORMAT('1')
      WRITE(6,21)
 21   FORMAT(10X,'THE UNTRUNCATED VALUES ARE')
      WRITE(6,88) ALFA1,BETA1,ALFA2,BETA2,ALFA3,BETA3
 88   FORMAT(1X,' ALFA1=',E12.5,1X,'BETA1 =',E12.5,1X,'ALFA2 =',E12.5,1X
     *,'BETA2 =',E12.5,1X,'ALFA3 =',E12.5,1X,'BETA3 =',E12.5,//)
      WRITE(6,91) SIGM41,PHI
 91   FORMAT(5X,'SIGM41=',E12.5,5X,'PHI=',E12.5,//)
C
      AA1=ALFA1
      AA2=BETA1
      AA3=ALFA2
      AA4=BETA2
      AA5=ALFA3
      AA6=BETA3
      DD1=SIGM41
      H=PHI
      WRITE(6,33) NOBTS
 33   FORMAT(10X,'     NO OF BITS =',I2,/)
C
C     PERFORMING THE TRUNCATION PROCESS  TO REQUIRED NO. OF BITS ,
C     ' NOBTS'
      ALFA1=TRUNC(AA1,NOBTS)
      BETA1=TRUNC(AA2,NOBTS)
      ALFA2=TRUNC(AA3,NOBTS)
      BETA2=TRUNC(AA4,NOBTS)
      ALFA3=TRUNC(AA5,NOBTS)
      BETA3=TRUNC(AA6,NOBTS)
      SIGM41=TRUNC(DD1,NOBTS)
      PHI=TRUNC(H,NOBTS)
      WRITE(6,22)
 22   FORMAT(10X,'THE TRUNCATED VALUES ARE')
      WRITE(6,88) ALFA1,BETA1,ALFA2,BETA2,ALFA3,BETA3
      WRITE(6,91) SIGM41,PHI
C
C     CALCULATION OF THE FREQUENCY RESPONCE WITH WAVE DIGITAL
C     PARAMETERS OF FINITE PRECESION
      CALL FILTER(UP,ALFA1,BETA1,ALFA2,BETA2,ALFA3,BETA3,SIGM41,PHI,DATA
     *2,OMGAC,DLTAT)
C
C     CALCULATION OF ROOT MEAN SQUARE ERROR
      ER=0.0D0
      DO 213 KK=1,98
      ER=ER+(DATA(KK,2)-DATA2(KK,2))**2
 213  CONTINUE
      ER=ER/98.0D0
      ER=DSQRT(ER)
C
      ALFA1=AA1
      BETA1=AA2
      ALFA2=AA3
      BETA2=AA4
      ALFA3=AA5
      BETA3=AA6
      SIGM41=DD1
      PHI=H
C
      RETURN
      END
```

D - Computer Program* No. 8. Program to calculate the rms error due to truncation in the number of bits of the RLC components of the conventional direct digital filter.

```
C                    **** FREQUENCY RESPONCE ****
C
C     ROOT MEAN SQUARE ERROR DUE TO TRUNCATION IN NO. OF BITS
C     FOR CONVENTIONAL DIRECT DIGITAL FILTER DESIGN
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DATA(210,3)
      DIMENSION DATA3(210,3)
C
C
C      **********.5DB CHEBECHEV LOW-PASS FILTER WITH RS=1.0   ********
C     COMPONENT VALUES
C     INDUCTANCE AND CAPACITANCE VALUES IN HENRIES,AND FARADS
C     NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C     WITH   RL=1
C
      RS=1.0D0
      AL1=1.7896D0
      C2=1.2961D0
      AL3=2.7177D0
      C4=1.3848D0
      AL5=2.7177D0
      C6=1.2961D0
      AL7=1.7896D0
      RL=1.0D0
C
C     FILTER SCALE FACTOR IS COEF
      COEF=(RL+RS)/RL
      WRITE(6,37) COEF
 37   FORMAT(7X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
      WRITE(6,27)
 27   FORMAT(5X,'THE UNSCALED COMPONENT VALUES ARE',//)
      WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
 22   FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
     1,'L5=',E12.5,3X,'C5=',E12.5,3X,'L7=',E12.5,//)
      WRITE(6,28) RS,RL
 28   FORMAT(20X,'RS=',E12.5,5X,'RL=',E12.5,//)
C
C
C     SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
      OMGAC=1.0D0
C
C     SAMPLING TIME IS T
      T=1.0D0
C
C     FREQUENCY SCALING WITH PREWARPING IS  SFREQ
      SFREQ=2*DTAN(OMGAC*T/2)/T
      AL1=AL1/SFREQ
      C2=C2/SFREQ
      AL3=AL3/SFREQ
      C4=C4/SFREQ
      AL5=AL5/SFREQ
      C6=C6/SFREQ
      AL7=AL7/SFREQ
      WRITE(6,29)
 29   FORMAT(5X,'THE   SCALED COMPONENT VALUES ARE',//)
      WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
C
C     INPUT IN TIME DOMAIN IS    UP
      UP=1.0D0
C
C     CALCULATION OF THE FREQUENCY RESPONCE OF THE
C     FILTER WITH PRACTICALLY INFINITE PRECESION
C     IN ORDER TO SET THE REFFERENCE DATA
C
      CALL FILTER(UP ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,T)
C
      DO 220 JJ=1,20
      NOBTS=JJ
C
C     CALCULATION OF THE ERROR IN THE FREQUENCY DOMAIN DUE
C     TO QUANTIZATION OF THE FILTER COMPONENT VALUES TO THE REQUIRED NO.
```

*Subroutine Graphx is given in Computer Program No. 1 of Appendix 1 and Function Trunc is given in Computer Program No. 5.

```
C      OF BITS
C
       CALL FILERR(UP  ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,NOBTS,ER
      *,T)
C
C      OUTPUT DATA SET UP
       DATA3(JJ,1)=JJ
       DATA3(JJ,2)=ER
       DATA3(JJ,3)=0.0D0
C
  220  CONTINUE
C
C      PLOT OF THE R.M.S. ERROR DUE TO QUANTIZATION IN THE NO OF
C      BITS OF FILTER COMPONENT VALUES VERSUS THE NO. OF BITS
C
       WRITE(6,54)
   54  FORMAT('1')
       WRITE(6,55)
   55  FORMAT(20X,'NO. OF BITS',15X,'MEAN SQUAR ERROR',
       WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,20)
       CALL GRAPHX(DATA3,20,4HNBIT,4HMSER)
   20  FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
       STOP
       END
```

FILTER

```
      SUBROUTINE FILTER(UP ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,T)
C
C     SUBROUTINE FILTER TAKES THE VALUE OF THE INPUT IN THE TIME
C     DOMAIN AND PLOTS THE MAGNITUDE VERS. FREQUENCY CURVE
C     OF THE CONVENTIONAL DIRECT DIGITAL FILTER
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 WT,Z,Y,H
      DIMENSION DATA(210,3),DT(210,3)
      DIMENSION A(8),Y(7)
C
C     CALCULATION OF      COEFFICIENTS OF  THE EQUATION
C
C      H(S)=K/(S7+A6*S6+A5*S5+A4*S4+A3*S3+A2*S2+A1*S1+A0)
C
C
      AA0=(RL+RS)/RL
      AA1=(AL1+AL3+AL5+AL7)/RL+RS*(C2+C4+C6)
      AA2=(C4+C6)*(AL1+AL3+AL7*RS/RL)+C2*(AL1+(AL3+AL5+AL7)*RS/RL)+AL5*(
     *C6+C4*RS/RL)
      AA3=(AL1+AL3)*(AL5+AL7)*C4/RL+(AL3+AL5)*(C2*C6*RS+(C6*AL7+C2*AL1)/
     *RL)+C4*RS*(C2*AL3+AL5*C6)+AL1*AL7*(C2+C6)/RL
      AA4=(AL1+AL7*RS/RL)*(AL5*C6*(C2+C4)+AL3*C2*(C4+C6))+C4*AL5*AL3*(C6
     *+C2*RS/RL)
      AA5=AL1*(C2*AL3*(C4*(AL5+AL7)+C6*AL7)+AL5*C6*AL7*(C2+C4))/RL+AL3*C
     *4*AL5*C6*(AL7/RL+C2*RS)
      AA6=C2*AL3*C4*AL5*C6*(AL7*RS/RL+AL1)
      AA7=(AL1*C2*AL3*C4*AL5*C6*AL7)/RL
      A(1)=1.0D0
      A(2)=AA6/AA7
      A(3)=AA5/AA7
      A(4)=AA4/AA7
      A(5)=AA3/AA7
      A(6)=AA2/AA7
      A(7)=AA1/AA7
      A(8)=AA0/AA7
      COEF1=A(8)
      WRITE(6,41)
41    FORMAT(5X,'THE COEF.OF    H(S)=K/(S7+A6*S6+A5*S5+A4*S4+A3*S3+A2*S2+
     1A1*S1+A0)     ARE',/)
      WRITE(6,40) A(2),A(3),A(4),A(5),A(6),A(7),A(8)
40    FORMAT(5X,'A6=',E12.5,3X,'A5=',E12.5,3X,'A4=',E12.5,3X,'A3=',E12.5
     1,3X,'A2=',E12.5,3X,'A1=',E12.5,3X,'A0=',E12.5,/)
      WRITE(6,43) COEF1
43    FORMAT(5X,'K=',E12.5,//)
C
C
C     CALCULATION OF THE COEF. OF THE EQN.
C
C                  7  7     6     5     4     3     2     1
C     H(Z)=K*(1+Z)/(Z+A6Z +A5Z +A4Z +A3Z +A2Z +A1Z +A0)
C
      F1=2.0D0/T
      F2=F1*F1
      F3=F2*F1
      F4=F3*F1
      F5=F4*F1
      F6=F5*F1
      F7=F6*F1
      ALFA7=F7+F6*A(2)+F5*A(3)+F4*A(4)+F3*A(5)+F2*A(6)+F1*A(7)+A(8)
      ALFA6=-7*F7-5*F6*A(2)-3*F5*A(3)-F4*A(4)+F3*A(5)+3*F2*A(6)+5*F1*A(7
     *)+7*A(8)
      ALFA5=21*F7+9*F6*A(2)+F5*A(3)-3*F4*A(4)-3*F3*A(5)+F2*A(6)+9*F1*A(7
     *)+21*A(8)
      ALFA4=-35*F7-5*F6*A(2)+5*F5*A(3)+3*F4*A(4)-3*F3*A(5)-5*F2*A(6)+5*F
     *1*A(7)+35*A(8)
      ALFA3=35*F7-5*F6*A(2)-5*F5*A(3)+3*F4*A(4)+3*F3*A(5)-5*F2*A(6)-5*F1
     **A(7)+35*A(8)
      ALFA2=-21*F7+9*F6*A(2)-F5*A(3)-3*F4*A(4)+3*F3*A(5)+F2*A(6)-9*F1*A(
     *7)+21*A(8)
      ALFA1=7*F7-5*F6*A(2)+3*F5*A(3)-F4*A(4)-F3*A(5)+3*F2*A(6)-5*F1*A(7)
     *+7*A(8)
      ALFA0=-F7+F6*A(2)-F5*A(3)+F4*A(4)-F3*A(5)+F2*A(6)-F1*A(7)+A(8)
```

```
      ALFA6=ALFA6/ALFA7
      ALFA5=ALFA5/ALFA7
      ALFA4=ALFA4/ALFA7
      ALFA3=ALFA3/ALFA7
      ALFA2=ALFA2/ALFA7
      ALFA1=ALFA1/ALFA7
      ALFA0=ALFA0/ALFA7
      COEF2=COEF1/ALFA7
      WRITE(6,38)
 38   FORMAT(5X,'THE COEFFICIENTS OF    H(Z)=K(1+Z)7/(Z7+A6Z6+A5Z5+A4Z4+A
     *3Z3+A2Z2+A1Z+A0)  ARE',///)
      WRITE(6,40) ALFA6,ALFA5,ALFA4,ALFA3,ALFA2,ALFA1,ALFA0
      WRITE(6,43) COEF2
C
      UPP=UP*COEF2
C
C     FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
C     FREQUENCY INCREMENT
      DLW=OMGAC/50
C
C     INITIAL VALUES IN THE FREQUENCY DOMAIN
      W=0.0D0
C
C     ITTERATION IN THE FREQUENCY DOMAIN
C     ------------------------------------
      DO 110 J=1,98
C
C     INITIAL VALUES IN TIME DOMAIN
C
      UP1=UPP
      TT=0.0D0
      H=DCMPLX(0.0D0,0.0D0)
      X1=0.0
      X2=0.0
      X3=0.0
      X4=0.0
      X5=0.0
      X6=0.0
      X7=0.0
C
C     ITTERATION IN THE TIME DOMAIN
C     -------------------------------
      DO 100 I=1,500
      V=UP1+X1
      X1=7*UP1-ALFA6*V+X2
      X2=21*UP1-ALFA5*V+X3
      X3=35*UP1-ALFA4*V+X4
      X4=35*UP1-ALFA3*V+X5
      X5=21*UP1-ALFA2*V+X6
      X6=7*UP1-ALFA1*V+X7
      X7=UP1-ALFA0*V
      UP1=0.0D0
      W1=W*TT
      WT=DCMPLX(0.0D0,-W1)
      Z=CDEXP(WT)
      H=H+V*Z
      TT=TT+T
C
 100  CONTINUE
      DA=CDABS(H)
C
C     ARRANGIGNG THE OUTPUT DATA
      DATA(J,1)=W
      DATA(J,2)=DA
      DATA(J,3)=0.0D0
C
C     ARRANGING THE DATA FOR PLOT SUBROUTINE
      DT(J,1)=DATA(J,1)
      DT(J,2)=DATA(J,2)
      DT(J,3)=DATA(J,3)
C
      W=W+DLW
 110  CONTINUE
```

```
C
C     WRITING AND PLOTING THE FREQUENCY RESPONCE
      WRITE(6,54)
 54   FORMAT('1')
      WRITE(6,55)
 55   FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DT,98,4HFREQ,4HMAGN)
 20   FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      RETURN
      END
```

```
      SUBROUTINE FILERR(UP  ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,NO
     *BTS,ER,T)
C
C     SUBROUTINE FILERR TAKES THE ROOT MEAN SQUARE VALUE OF THE ERROR
C     BETWEEN UNTRUNCATED AND TRUNCATED VALUE OF THE OUT PUT IN THE
C     FREQUENCY DOMAIN AND ALSO PLOTS THE FINITE PRECESION FREQUENCY
C     RESPONCE
C
      IMPLICIT REAL*8 (A-H, O-Z)
      DIMENSION DATA2(210,3)
      DIMENSION DATA(210,3)
C
      WRITE(6,54)
 54   FORMAT('1')
      WRITE(6,21)
 21   FORMAT(10X,'THE UNTRUNCATED VALUES ARE',/)
      WRITE(6,22)   AL1,C2,AL3,C4,AL5,C6,AL7
 22   FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,3X,'L3=',E12.5,3X,'C4=',E12.5
     *,3X,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
      WRITE(6,28) RS,RL
 28   FORMAT(20X,'RS=',E12.5,5X,'RL=',E12.5,//)
C
      A=RS
      B=AL1
      C=C2
      D=AL3
      E=C4
      F=AL5
      G=C6
      H=AL7
      P=RL
      WRITE(6,33) NOBTS
 33   FORMAT(10X,'      NO OF BITS =',I2,/)
C
C     PERFORMING THE TRUNCATION PROCESS  TO REQUIRED NO. OF BITS ,
C     ' NOBTS'
      RS=TRUNC(RS,NOBTS)
      AL1=TRUNC(AL1,NOBTS)
      C2=TRUNC(C2 ,NOBTS)
      AL3=TRUNC(AL3,NOBTS)
      C4=TRUNC(C4 ,NOBTS)
      AL5=TRUNC(AL5 ,NOBTS)
      C6=TRUNC(C6 ,NOBTS)
      AL7=TRUNC(AL7 ,NOBTS)
      RL=TRUNC(RL ,NOBTS)
      WRITE(6,25)
 25   FORMAT(10X,'THE TRUNCATED VALUES ARE')
      WRITE(6,22)   AL1,C2,AL3,C4,AL5,C6,AL7
      WRITE(6,28) RS,RL
C
C     CALCULATION OF THE FREQUENCY RESPONCE WITH COMPLONENTS OF
C     FINITE PRECESION
      CALL FILTER(UP  ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA2,T)
C
C     CALCULATION OF ROOT MEAN SQUARE ERROR
      ER=0.0D0
      DO 213 KK=1,98
      ER=ER+(DATA(KK,2)-DATA2(KK,2))**2
 213  CONTINUE
      ER=ER/98.0D0
      ER=DSQRT(ER)
C
      RS=A
      AL1=B
      C2=C
      AL3=D
      C4=E
      AL5=F
      C6=G
      AL7=H
      RL=P
C
      RETURN
      END
```

E - Computer Program No. 9.* Program to calculate the rms error due to truncation in the number of bits of the RLC components of the conventional cascaded digital filter.

```
C                   **** FREQUENCY RESPONCE ****
C
C      ROOT MEAN SQUARE ERROR DUE TO TRUNCATION IN NO. OF BITS
C      FOR CONVENTIONAL CASCADED DIGITAL FILTER
C
       IMPLICIT REAL*8 (A-H,O-Z)
       DIMENSION DATA(210,3)
       DIMENSION DATA3(210,3)
C
C
C       **********.5DB CHEBECHEV LOW-PASS FILTER WITH RS=1.0   ********
C      COMPONENT VALUES
C      INDUCTANCE AND CAPACITANCE VALUES IN HENRIES,AND FARADS
C      NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C      WITH    RL=1
C
       RS=1.0D0
       AL1=1.7896D0
       C2=1.2961D0
       AL3=2.7177D0
       C4=1.3848D0
       AL5=2.7177D0
       C6=1.2961D0
       AL7=1.7896D0
       RL=1.0D0
C
C      FILTER SCALE FACTOR IS COEF
       COEF=(RL+RS)/RL
       WRITE(6,37) COEF
 37    FORMAT(7X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
       WRITE(6,27)
 27    FORMAT(5X,'THE UNSCALED COMPONENT VALUES ARE',//)
       WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
 22    FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
      1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
       WRITE(6,28) RS,RL
 28    FORMAT(20X,'RS=',E12.5,5X,'RL=',E12.5,//)
C
C
C      SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
       OMGAC=1.0D0
C
C      SAMPLING TIME IS T
       T=1.0D0
C
C      FREQUENCY SCALING WITH PREWARPING IS  SFREQ
       SFREQ=2*DTAN(OMGAC*T/2)/T
       AL1=AL1/SFREQ
       C2=C2/SFREQ
       AL3=AL3/SFREQ
       C4=C4/SFREQ
       AL5=AL5/SFREQ
       C6=C6/SFREQ
       AL7=AL7/SFREQ
       WRITE(6,29)
 29    FORMAT(5X,'THE   SCALED COMPONENT VALUES ARE',//)
       WRITE(6,22)  AL1,C2,AL3,C4,AL5,C6,AL7
C
C      INPUT IN TIME DOMAIN IS    UP
       UP=1.0D0
C
C      CALCULATION OF THE FREQUENCY RESPONCE OF THE
C      FILTER WITH PRACTICALLY INFINITE PRECESION
C      IN ORDER TO SET THE REFFERENCE DATA
C
       CALL  FILTER(UP ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,T)
C
       DO 220 JJ=1,20
       NOBTS=JJ
C
C      CALCULATION OF THE ERROR IN THE FREQUENCY DOMAIN DUE
C      TO QUANTIZATION OF THE FILTER COMPONENT VALUES TO THE REQUIRED NO.
```

*Subroutine Graphx is given in Computer Program No. 1 of Appendix 1 and Frunction Trunc is given in the Computer Program No. 5.

```
C        OF BITS
C
         CALL FILERR(UP ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,CMGAC,DATA,NOBTS,ER
        *,T)
C
C        OUTPUT DATA SET UP
         DATA3(JJ,1)=JJ
         DATA3(JJ,2)=ER
         DATA3(JJ,3)=0.0D0
C
 220     CONTINUE
C
C        PLOT OF THE R.M.S. ERROR DUE TO QUANTIZATION IN THE NO OF
C        BITS OF FILTER COMPONENT VALUES VERSUS THE NO. OF BITS
C
         WRITE(6,54)
  54     FORMAT('1')
         WRITE(6,55)
  55     FORMAT(20X,'NO. OF BITS',15X,'MEAN SQUAR ERROR',/)
         WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,20)
         CALL GRAPHX(DATA3,20,4HNBIT,4HMSER)
  20     FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
         STOP
         END
```

FILTER

```
      SUBROUTINE FILTER(UP ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,T)
C
C     SUBROUTINE FILTER TAKES THE VALUE OF THE INPUT IN THE TIME
C     DOMAIN AND PLOTS THE MAGNITUDE VERS. FREQUENCY CURVE
C     OF THE CONVENTIONAL CASCADED DIGITAL FILTER
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 WT,Z,Y,H
      DIMENSION DATA(210,3),DT(210,3)
      DIMENSION A(8),Y(7)
C
C     CALCULATION OF      COEFFICIENTS OF  THE EQUATION
C
C      H(S)=K/(S7+A6*S6+A5*S5+A4*S4+A3*S3+A2*S2+A1*S1+A0)
C
C
      AA0=(RL+RS)/RL
      AA1=(AL1+AL3+AL5+AL7)/RL+RS*(C2+C4+C6)
      AA2=(C4+C6)*(AL1+AL3+AL7*RS/RL)+C2*(AL1+(AL3+AL5+AL7)*RS/RL)+AL5*(
     *C6+C4*RS/RL)
      AA3=(AL1+AL3)*(AL5+AL7)*C4/RL+(AL3+AL5)*(C2*C6*RS+(C6*AL7+C2*AL1)/
     *RL)+C4*RS*(C2*AL3+AL5*C6)+AL1*AL7*(C2+C6)/RL
      AA4=(AL1+AL7*RS/RL)*(AL5*C6*(C2+C4)+AL3*C2*(C4+C6))+C4*AL5*AL3*(C6
     *+C2*RS/RL)
      AA5=AL1*(C2*AL3*(C4*(AL5+AL7)+C6*AL7)+AL5*C6*AL7*(C2+C4))/RL+AL3*C
     *4*AL5*C6*(AL7/RL+C2*RS)
      AA6=C2*AL3*C4*AL5*C6*(AL7*RS/RL+AL1)
      AA7=(AL1*C2*AL3*C4*AL5*C6*AL7)/RL
      A(1)=1.0D0
      A(2)=AA6/AA7
      A(3)=AA5/AA7
      A(4)=AA4/AA7
      A(5)=AA3/AA7
      A(6)=AA2/AA7
      A(7)=AA1/AA7
      A(8)=AA0/AA7
      COEF1=A(8)
      WRITE(6,41)
   41 FORMAT(5X,'THE COEF.OF    H(S)=K/(S7+A1*S6+A2*S5+A3*S4+A4*S3+A5*S2+
     1A6*S1+A7)    ARE',/)
      WRITE(6,40) A(2),A(3),A(4),A(5),A(6),A(7),A(8)
   40 FORMAT(5X,'A1=',E12.5,3X,'A2=',E12.5,3X,'A3=',E12.5,3X,'A4=',E12.5
     1,3X,'A5=',E12.5,3X,'A6=',E12.5,3X,'A7=',E12.5,/)
      WRITE(6,43) COEF1
C
C
C     CALCULATION OF THE COEF. OF THE EQN.
C               H(S)=  K/(S2+A1*S+B1)(S2+A2*S+B2)(S2+A3*S+B3)(S+B4)
C
C
   43 FORMAT(5X,'K=',E12.5,//)
      NDEG=7
C
C     SUBROUTINE ZPOLR GIVES THE ROOTS OF THE POLYNOMIAL OF DEGREE 'N'
      CALL ZPOLR(A,NDEG,Y,IER)
      P1=-2.0D0*REAL(Y(1))
      F1=REAL(Y(1))**2+AIMAG(Y(1))**2
      P2=-2.0D0*REAL(Y(3))
      F2=REAL(Y(3))**2+AIMAG(Y(3))**2
      P3=-2.0D0*REAL(Y(5))
      F3=REAL(Y(5))**2+AIMAG(Y(5))**2
      P4=-REAL(Y(7))
      PP=1.0D0/AA7
      WRITE(6,42)
   42 FORMAT(5X,'THE COEF. OF H(S) ARE......
     1       H(S)=K/(S2+A1*S+B1)(S2+A2*S+B2)(S2+A3*S+B3)(S+B4)',//)
      WRITE(6,60) P1,F1,P2,F2,P3,F3,P4,COEF1
C
C     CALCULATION OF THE COEF OF H(Z)  WITH SAMPLING PERIOD OF T
C
C                  -1 7          -1           -1      -2            -1      -2         -1     -2
C     H(Z)=K*(1+Z  )  /(1+A1*Z  )(1+A2*Z  +B2*Z  )(1+A3*Z  +B3*Z  )(1+A4Z  +B4Z  )
C
```

```
C
60    FORMAT(1X,'A1=',E12.5,1X,'B1=',E12.5,2X,'A2=',E12.5,1X,'B2=',E12.5
     1,2X,'A3=',E12.5,1X,'B3=',E12.5,2X,'A4=',E12.5,3X,'K=',E12.5,//)
      F=2.0D0/T
      FF=F*F
      B1=2.0D0*(F1-FF)/(FF+P1*F+F1)
      D1=(FF-P1*F+F1)/(FF+P1*F+F1)
      B2=2.0D0*(F2-FF)/(FF+P2*F+F2)
      D2=(FF-P2*F+F2)/(FF+P2*F+F2)
      B3=2.0D0*(F3-FF)/(FF+P3*F+F3)
      D3=(FF-P3*F+F3)/(FF+P3*F+F3)
      B4=(P4-F)/(P4+F)
      COEF2=COEF1/((P4+F)*(FF+F*P1+F1)*(FF+F*P2+F2)*(FF+F*P3+F3))
      WRITE(6,66)
66    FORMAT(5X,'THE COEF. OF Z**-1 AND Z**-2 IN QUADRATURE FORM AND TOT
     1AL MULTLYING FACTOR K ARE...',/)
      WRITE(6,60) B1,D1,B2,D2,B3,D3,B4,COEF2
C
      UPP=UP*COEF2
C     FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
C     FREQUENCY INCREMENT
      DLW=OMGAC/50
C
C     INITIAL VALUES IN THE FREQUENCY DOMAIN
      W=0.0D0
C
C     ITTERATION IN THE FREQUENCY DOMAIN
C     ----------------------------------
      DO 110 J=1,98
C
C     INITIAL VALUES IN TIME DOMAIN
C
      UP1=UPP
      TT=0.0D0
      H=DCMPLX(0.0D0,0.0D0)
      X1=0.0
      X2=0.0
      X3=0.0
      X4=0.0
      X5=0.0
      X6=0.0
      X7=0.0
C
C     ITTERATION IN THE TIME DOMAIN
C     -----------------------------
      DO 100 I=1,500
      V1=UP1+X1
      V2=V1+X2
      V3=V2+X3
      V=V3+X4
      X1=UP1-B4*V1
      X2=2*V1-B1*V2+X5
      X3=2*V2-B2*V3+X6
      X4=2*V3-B3*V+X7
      X5=V1-D1*V2
      X6=V2-D2*V3
      X7=V3-D3*V
      UP1=0.0D0
      W1=W*TT
      WT=DCMPLX(0.0D0,-W1)
      Z=CDEXP(WT)
      H=H+V*Z
      TT=TT+T
C
100   CONTINUE
      DA=CDABS(H)
C
C     ARRANGIGNG THE OUTPUT DATA
      DATA(J,1)=W
      DATA(J,2)=DA
      DATA(J,3)=0.0D0
C
C     ARRANGING THE DATA FOR PLOT SUBROUTINE
```

```
      DT(J,1)=DATA(J,1)
      DT(J,2)=DATA(J,2)
      DT(J,3)=DATA(J,3)
C
      W=W+DLW
 110  CONTINUE
C
C     WRITING AND PLOTING THE FREQUENCY RESPONCE
      WRITE(6,54)
 54   FORMAT('1')
      WRITE(6,55)
 55   FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DT,98,4HFREQ,4HMAGN)
 20   FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      RETURN
      END
```

FILERR

```
      SUBROUTINE FILERR(UP  ,RS ,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,NO
     *BTS,ER,T)
C
C     SUBROUTINE FILERR TAKES THE ROOT MEAN SQUARE VALUE OF THE ERROR
C     BETWEEN UNTRUNCATED AND TRUNCATED VALUE OF THE OUT PUT IN THE
C     FREQUENCY DOMAIN AND ALSO PLOTS THE FINITE PRECESION FREQUENCY
C     RESPONCE
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION DATA2(210,3)
      DIMENSION DATA(210,3)
C
      WRITE(6,54)
   54 FORMAT('1')
      WRITE(6,21)
   21 FORMAT(10X,'THE UNTRUNCATED VALUES ARE',/)
      WRITE(6,22)   AL1,C2,AL3,C4,AL5,C6,AL7
   22 FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,3X,'L3=',E12.5,3X,'C4=',E12.5
     *,3X,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
      WRITE(6,28)  RS,RL
   28 FORMAT(20X,'RS=',E12.5,5X,'RL=',E12.5,//)
C
      A=RS
      B=AL1
      C=C2
      D=AL3
      E=C4
      F=AL5
      G=C6
      H=AL7
      P=RL
      WRITE(6,33)  NOBTS
   33 FORMAT(10X,'     NO OF BITS =',I2,/)
C
C     PERFORMING THE TRUNCATION PROCESS  TO REQUIRED NO. OF BITS ,
C     ' NOBTS'
      RS=TRUNC(RS,NOBTS)
      AL1=TRUNC(AL1,NOBTS)
      C2=TRUNC(C2,NOBTS)
      AL3=TRUNC(AL3,NOBTS)
      C4=TRUNC(C4,NOBTS)
      AL5=TRUNC(AL5,NOBTS)
      C6=TRUNC(C6,NOBTS)
      AL7=TRUNC(AL7,NOBTS)
      RL=TRUNC(RL,NOBTS)
      WRITE(6,25)
   25 FORMAT(10X,'THE TRUNCATED VALUES ARE')
      WRITE(6,22)   AL1,C2,AL3,C4,AL5,C6,AL7
      WRITE(6,28) RS,RL
C
C     CALCULATION OF THE FREQUENCY RESPONCE WITH COMPLONENTS OF
C     FINITE PRECESION
      CALL  FILTER(UP  ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA2,T)
C
C     CALCULATION OF ROOT MEAN SQUARE ERROR
      ER=0.0D0
      DO 213 KK=1,98
      ER=ER+(DATA(KK,2)-DATA2(KK,2))**2
  213 CONTINUE
      ER=ER/98.0D0
      ER=DSQRT(ER)
C
      RS=A
      AL1=B
      C2=C
      AL3=D
      C4=E
      AL5=F
      C6=G
      AL7=H
      RL=P
C
      RETURN

      END
```

A. Computer Program* No. 10. Program to calculate the sensitivity function of the seventh order low pass wave digital filter with respect to both wave digital filter multiplier coefficients and the original filter component values.

```
C
C
C        ***********  FREQUENCY DOMAIN *************
C
C        *******  SENSITIVITY FUNCTION  OF THE FILTER *************
C
C        WITH RESPECT TO L'S AND C'S ,RS,AND RL
C        AND ALSO WITH RESPECT TO WAVE DIGITAL PARAMETERS
C        ON THE SAME GRAPH FOR COMPARISON PURPOSES
C
C
C
C
         IMPLICIT REAL*8 (A-H,O-Z)
C
         DIMENSION DATA0(210,3)
         DIMENSION DATA1(210,3)
         DIMENSION DATA2(210,3)
         DIMENSION DATA3(210,3)
         DIMENSION DATA4(210,3)
         DIMENSION DATA5(210,3)
         DIMENSION DATA6(210,3)
         DIMENSION DATA7(210,3)
         DIMENSION DATA8(210,3)
C
         COMPLEX*16 H1,H2,H3,H4,H5,H6,H7,H8
         COMPLEX*16 T1,T2,T3,T4,T5,T6,T7,T8,TO
         COMPLEX*16 W1,Z,H
C
C
C         ***********.5DB CHEBECHEV LOW-PASS FILTER WITH RS=1.0   ********
         RS=1.0D0
         AL1=1.7896D0
         C2=1.2961D0
         AL3=2.7177D0
         C4=1.3848D0
         AL5=2.7177D0
         C6=1.2961D0
         AL7=1.7896D0
         RL=1.0D0
C
C        FILTER SCALE FACTOR FROM DATA IS COEF
         COEF=(RL+RS)/RL
         WRITE(6,37) COEF
  37     FORMAT(7X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
C
         WRITE(6,19) AL1,C2,AL3,C4,AL5,C6,AL7
  19     FORMAT(5X,' SER L1=',F8.4,2X,'SHT C2=',F8.4,2X,'SER L3=',F8.4,2X,'
        *SHT C4=',F8.4,2X,'SER L5=',F8.4,2X,'SHT C6=',F8.4,2X,'SER L7=',F8.
        *4,/)
         WRITE(6,18) RL,RS
  18     FORMAT( 10X,'     RL=',F8.4,5X,'RS=',F8.4,/)
C
C        SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
         OMGAC=1.0D0
C
C        SAMPLING PERIOD IS DLTAT
         DLTAT=1.0D0
C
C
C        SCALE TO NORMALIZE THE ELEMENT VALUES ,AS WELL AS TAKING INTO
C        ACCOUNT THE EFFECT OF SAMPLING TIME AND FREQUENCY SCALE
         SCALE=1.0D0/DTAN(OMGAC*DLTAT/2)
         AL1=AL1*SCALE
         C2=C2*SCALE
         AL3=AL3*SCALE
         C4=C4*SCALE
         AL5=AL5*SCALE
         C6=C6*SCALE
         AL7=AL7*SCALE
C
C        CALCULATION TO FIND THE TERMINATING RESISTANCE OF EACH ELEMENT
C        AND WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
C        WITH NO DELAY FREE PATH ON PORT TWO
```

*Subroutine Graphyx is given in Computer Program No. 1, Appendix 1.

```
      R1=RS
      R2=R1+AL1
      SIGMA1=R1/R2
      G2=1.0D0/R2
      G3=G2+C2
      SIGMA2=G2/G3
      R3=1.0D0/G3
      R4=R3+AL3
      SIGMA3=R3/R4
      G4=1.0D0/R4
      G5=G4+C4
      SIGMA4=G4/G5
      R5=1.0D0/G5
      R6=R5+AL5
      SIGMA5=R5/R6
      G6=1.0D0/R6
      G7=G6+C6
      SIGMA6=G6/G7
      R7=1.0D0/G7
      R8=R7+AL7
      SIGMA7=R7/R8
      PHI=(RL-R8)/(RL+R8)
      WRITE(6,42)
 42   FORMAT(5X,'THE WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS ARE....
     *',//)
      WRITE(6,10) SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PHI
 10   FORMAT(/,4X,'SIGMA1=',F6.4,4X,'SIGMA2=',F6.4,4X,'SIGMA3=',F6.4,4X,
     *'SIGMA4=',F6.4,4X,'SIGMA5=',F6.4,4X,'SIGMA6=',F6.4,4X,'SIGMA7=',F6
     *.4,4X,'PHI=',F6.4)
C
C
C
C     IMPULSE INPUT TO THE FILTER WITH MAGNITUDE OF   AS=1
      AS=1.0D0
      UP=AS*COEF
C
C     FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
C     FREQUENCY INCREMENT
      DLTAW=OMGAC/50.D0
C
C
C
C     PREMULTIPLICATION OF
C     COEFFICIENTS OF PARTIAL DIFFRENTIATION OF  SIGMA'S AND PHI
C     WITH RESPECT TO L'S AND C'S ,RS,AND RL
C
      A1=AL1/(R1+AL1)**2
      A2=A1*C2*R1/(SIGMA1+R1*C2)**2
      A3=A2*R2*AL3/(SIGMA2*R2+AL3)**2
      A4=A3*G3*C4/(SIGMA3*G3+C4)**2
      A5=A4*R4*AL5/(SIGMA4*R4+AL5)**2
      A6=A5*G5*C6/(SIGMA5*G5+C6)**2
      A7=A6*R6*AL7/(SIGMA6*R6+AL7)**2
      A8=-2*A7*G7*RL/(SIGMA7*G7+RL)**2
C
      B1=-R1/(R1+AL1)**2
      B2=-R1*R1*C2/((SIGMA1+R1*C2)*(R1+AL1))**2
      B3=B2*R2*AL3/(SIGMA2*R2+AL3)**2
      B4=B3*G3*C4/(SIGMA3*G3+C4)**2
      B5=B4*R4*AL5/(SIGMA4*R4+AL5)**2
      B6=B5*G5*C6/(SIGMA5*G5+C6)**2
      B7=B6*R6*AL7/(SIGMA6*R6+AL7)**2
      B8=-2*B7*G7*RL/(SIGMA7*G7+RL)**2
C
      D2=-G2/(G2+C2)**2
      D3=D2*R2*AL3/(SIGMA2*R2+AL3)**2
      D4=D3*G3*C4/(SIGMA3*G3+C4)**2
      D5=D4*R4*AL5/(SIGMA4*R4+AL5)**2
      D6=D5*G5*C6/(SIGMA5*G5+C6)**2
      D7=D6*R6*AL7/(SIGMA6*R6+AL7)**2
      D8=-2*D7*G7*RL/(SIGMA7*G7+RL)**2
C
      E3=-R3/(R3+AL3)**2
      E4=E3*G3*C4/(SIGMA3*G3+C4)**2
```

```
      E5=E4*R4*AL5/(SIGMA4*R4+AL5)**2
      E6=E5*G5*C6/(SIGMA5*G5+C6)**2
      E7=E6*R6*AL7/(SIGMA6*R6+AL7)**2
      E8=-2*E7*G7*RL/(SIGMA7*G7+RL)**2
C
      P4=-G4/(G4+C4)**2
      P5=P4*R4*AL5/(SIGMA4*R4+AL5)**2
      P6=P5*G5*C6/(SIGMA5*G5+C6)**2
      P7=P6*R6*AL7/(SIGMA6*R6+AL7)**2
      P8=-2*P7*G7*RL/(SIGMA7*G7+RL)**2
C
      Q5=-R5/(R5+AL5)**2
      Q6=Q5*G5*C6/(SIGMA5*G5+C6)**2
      Q7=Q6*R6*AL7/(SIGMA6*R6+AL7)**2
      Q8=-2*Q7*G7*RL/(SIGMA7*G7+RL)**2
C
      S6=-G6/(G6+C6)**2
      S7=S6*R6*AL7/(SIGMA6*R6+AL7)**2
      S8=-2*S7*G7*RL/(SIGMA7*G7+RL)**2
C
      U7=-R7/(R7+AL7)**2
      U8=-2*U7*G7*RL/(SIGMA7*G7+RL)**2
C
      V8=2*G7*SIGMA7/(RL+G7*SIGMA7)**2
C
C     INITIAL VALUES IN FREQUENCY DOMAIN
C     ----------------------------------------------
      W=0.0D0
C
C     ITTERATION IN THE FREQUENCY DOMAIN
C     **************************************
      DO 110 J=1,98
C
C     INITIAL VALUES IN THE TIME DOMAIN
C     ------------------------------------
C
      C11=UP
C
      TT=0.0D0
      H=DCMPLX(0.0D0,0.0D0)
      H1=DCMPLX(0.0D0,0.0D0)
      H2=DCMPLX(0.0D0,0.0D0)
      H3=DCMPLX(0.0D0,0.0D0)
      H4=DCMPLX(0.0D0,0.0D0)
      H5=DCMPLX(0.0D0,0.0D0)
      H6=DCMPLX(0.0D0,0.0D0)
      H7=DCMPLX(0.0D0,0.0D0)
      H8=DCMPLX(0.0D0,0.0D0)
C
      Y11=0.0D0
      Y13=0.0D0
      Y14=0.0D0
      Y23=0.0D0
      Y24=0.0D0
      Y33=0.0D0
      Y34=0.0D0
      Y43=0.0D0
      Y44=0.0D0
      Y53=0.0D0
      Y54=0.0D0
      Y63=0.0D0
      Y64=0.0D0
      Y72=0.0D0
      Y73=0.0D0
      Y74=0.0D0
C
      DX111=0.0D0
      DX113=0.0D0
      DX114=0.0D0
      DX123=0.0D0
      DX124=0.0D0
      DX133=0.0D0
      DX134=0.0D0
```

```
      DX143=0.0D0
      DX144=0.0D0
      DX153=0.0D0
      DX154=0.0D0
      DX163=0.0D0
      DX164=0.0D0
      DX172=0.0D0
      DX173=0.0D0
      DX174=0.0D0
C
      DX211=0.0D0
      DX213=0.0D0
      DX214=0.0D0
      DX223=0.0D0
      DX224=0.0D0
      DX233=0.0D0
      DX234=0.0D0
      DX243=0.0D0
      DX244=0.0D0
      DX253=0.0D0
      DX254=0.0D0
      DX263=0.0D0
      DX264=0.0D0
      DX272=0.0D0
      DX273=0.0D0
      DX274=0.0D0
C
      DX311=0.0D0
      DX313=0.0D0
      DX314=0.0D0
      DX323=0.0D0
      DX324=0.0D0
      DX333=0.0D0
      DX334=0.0D0
      DX343=0.0D0
      DX344=0.0D0
      DX353=0.0D0
      DX354=0.0D0
      DX363=0.0D0
      DX364=0.0D0
      DX372=0.0D0
      DX373=0.0D0
      DX374=0.0D0
C
      DX411=0.0D0
      DX413=0.0D0
      DX414=0.0D0
      DX423=0.0D0
      DX424=0.0D0
      DX433=0.0D0
      DX434=0.0D0
      DX443=0.0D0
      DX444=0.0D0
      DX453=0.0D0
      DX454=0.0D0
      DX463=0.0D0
      DX464=0.0D0
      DX472=0.0D0
      DX473=0.0D0
      DX474=0.0D0
C
      DX511=0.0D0
      DX513=0.0D0
      DX514=0.0D0
      DX523=0.0D0
      DX524=0.0D0
      DX533=0.0D0
      DX534=0.0D0
      DX543=0.0D0
      DX544=0.0D0
      DX553=0.0D0
      DX554=0.0D0
      DX563=0.0D0
```

```
      DX564=0.0D0
      DX572=0.0D0
      DX573=0.0D0
      DX574=0.0D0
C
      DX611=0.0D0
      DX613=0.0D0
      DX614=0.0D0
      DX623=0.0D0
      DX624=0.0D0
      DX633=0.0D0
      DX634=0.0D0
      DX643=0.0D0
      DX644=0.0D0
      DX653=0.0D0
      DX654=0.0D0
      DX663=0.0D0
      DX664=0.0D0
      DX672=0.0D0
      DX673=0.0D0
      DX674=0.0D0
C
      DX711=0.0D0
      DX713=0.0D0
      DX714=0.0D0
      DX723=0.0D0
      DX724=0.0D0
      DX733=0.0D0
      DX734=0.0D0
      DX743=0.0D0
      DX744=0.0D0
      DX753=0.0D0
      DX754=0.0D0
      DX763=0.0D0
      DX764=0.0D0
      DX772=0.0D0
      DX773=0.0D0
      DX774=0.0D0
C
      DX811=0.0D0
      DX813=0.0D0
      DX814=0.0D0
      DX823=0.0D0
      DX824=0.0D0
      DX833=0.0D0
      DX834=0.0D0
      DX843=0.0D0
      DX844=0.0D0
      DX853=0.0D0
      DX854=0.0D0
      DX863=0.0D0
      DX864=0.0D0
      DX872=0.0D0
      DX873=0.0D0
      DX874=0.0D0
C
C
C     ITTERATION IN THE TIME DOMAIN
C     ************************************
      DO 100 I=1,480
C
      D12=C11+Y11-Y23+SIGMA1*(Y23-Y14)
      D22=Y33+SIGMA2*(D12+Y14-Y33-Y24)
      D32=D22+Y24-Y43+SIGMA3*(Y43-Y34)
      D42=Y53+SIGMA4*(D32+Y34-Y53-Y44)
      D52=D42+Y44-Y63+SIGMA5*(Y63-Y54)
      D62=Y73+SIGMA6*(D52+Y54-Y73-Y64)
      D72=D62+Y64-Y72+SIGMA7*(Y72-Y74)
      C72=D72*PHI
      D71=D62+SIGMA7*(C72-D62+Y72-Y73)
      D61=D71-D52+Y73+SIGMA6*(D52-Y63)
      D51=D42+SIGMA5*(D61-D42+Y63-Y53)
      D41=D51-D32+Y53+SIGMA4*(D32-Y43)
```

```
      D31=D22+SIGMA3*(D41-D22+Y43-Y33)
      D21=D31-D12+Y33+SIGMA2*(D12-Y23)
      D11=C11+SIGMA1*(D21-C11+Y23-Y13)
C
C     DIFFRENTIATION WITH RESPECT TO SIGMA1
C     ------------------------------------------------------
C
      DB112=Y23-Y14-DX123+SIGMA1*(DX123-DX114)
      DB122=DX133+SIGMA2*(DB112+DX114-DX133-DX124)
      DB132=DB122+DX124-DX143+SIGMA3*(DX143-DX134)
      DB142=DX153+SIGMA4*(DB132+DX134-DX153-DX144)
      DB152=DB142+DX144-DX163+SIGMA5*(DX163-DX154)
      DB162=DX173+SIGMA6*(DB152+DX154-DX173-DX164)
      DB172=DB162+DX164-DX172+SIGMA7*(DX172-DX174)
      DA172=DB172*PHI
      DB171=DB162+SIGMA7*(DA172-DB162+DX172-DX173)
      DB161=DB171-DB152+DX173+SIGMA6*(DB152-DX163)
      DB151=DB142+SIGMA5*(DB161-DB142+DX163-DX153)
      DB141=DB151-DB132+DX153+SIGMA4*(DB132-DX143)
      DB131=DB122+SIGMA3*(DB141-DB122+DX143-DX133)
      DB121=DB131-DB112+DX133+SIGMA2*(DB112-DX123)
      DB111=D21-C11+Y23-Y13+SIGMA1*(DB121+DX123-DX113)
C
C     DIFFRENTIATION WITH RESPECT TO SIGMA2
C     ------------------------------------------------------
C
      DB212=-DX223+SIGMA1*(DX223-DX214)
      DB222=DX233+SIGMA2*(DB212+DX214-DX233-DX224)+D12+Y14-Y33-Y24
      DB232=DB222+DX224-DX243+SIGMA3*(DX243-DX234)
      DB242=DX253+SIGMA4*(DB232+DX234-DX253-DX244)
      DB252=DB242+DX244-DX263+SIGMA5*(DX263-DX254)
      DB262=DX273+SIGMA6*(DB252+DX254-DX273-DX264)
      DB272=DB262+DX264-DX272+SIGMA7*(DX272-DX274)
      DA272=DB272*PHI
      DB271=DB262+SIGMA7*(DA272-DB262+DX272-DX273)
      DB261=DB271-DB252+DX273+SIGMA6*(DB252-DX263)
      DB251=DB242+SIGMA5*(DB261-DB242+DX263-DX253)
      DB241=DB251-DB232+DX253+SIGMA4*(DB232-DX243)
      DB231=DB222+SIGMA3*(DB241-DB222+DX243-DX233)
      DB221=DB231-DB212+DX233+SIGMA2*(DB212-DX223)+D12-Y23
      DB211=SIGMA1*(DB221+DX223-DX213)
C
C     DIFFRENTIATION WITH RESPECT TO SIGMA3
C     ------------------------------------------------------
C
      DB312=-DX323+SIGMA1*(DX323-DX314)
      DB322=DX333+SIGMA2*(DB312+DX314-DX333-DX324)
      DB332=DB322+DX324-DX343+SIGMA3*(DX343-DX334)+Y43-Y34
      DB342=DX353+SIGMA4*(DB332+DX334-DX353-DX344)
      DB352=DB342+DX344-DX363+SIGMA5*(DX363-DX354)
      DB362=DX373+SIGMA6*(DB352+DX354-DX373-DX364)
      DB372=DB362+DX364-DX372+SIGMA7*(DX372-DX374)
      DA372=DB372*PHI
      DB371=DB362+SIGMA7*(DA372-DB362+DX372-DX373)
      DB361=DB371-DB352+DX373+SIGMA6*(DB352-DX363)
      DB351=DB342+SIGMA5*(DB361-DB342+DX363-DX353)
      DB341=DB351-DB332+DX353+SIGMA4*(DB332-DX343)
      DB331=DB322+SIGMA3*(DB341-DB322+DX343-DX333)+D41-D22+Y43-Y33
      DB321=DB331-DB312+DX333+SIGMA2*(DB312-DX323)
      DB311=SIGMA1*(DB321+DX323-DX313)
C
C     DIFFRENTIATION WITH RESPECT TO SIGMA4
C     ------------------------------------------------------
C
      DB412=-DX423+SIGMA1*(DX423-DX414)
      DB422=DX433+SIGMA2*(DB412+DX414-DX433-DX424)
      DB432=DB422+DX424-DX443+SIGMA3*(DX443-DX434)
      DB442=DX453+SIGMA4*(DB432+DX434-DX453-DX444)+D32+Y34-Y53-Y44
      DB452=DB442+DX444-DX463+SIGMA5*(DX463-DX454)
      DB462=DX473+SIGMA6*(DB452+DX454-DX473-DX464)
      DB472=DB462+DX464-DX472+SIGMA7*(DX472-DX474)
      DA472=DB472*PHI
```

```
      DB471=DB462+SIGMA7*(DA472-DB462+DX472-DX473)
      DB461=DB471-DB452+DX473+SIGMA6*(DB452-DX463)
      DB451=DB442+SIGMA5*(DB461-DB442+DX463-DX453)
      DB441=DB451-DB432+DX453+SIGMA4*(DB432-DX443)+D32-Y43
      DB431=DB422+SIGMA3*(DB441-DB422+DX443-DX433)
      DB421=DB431-DB412+DX433+SIGMA2*(DB412-DX423)
      DB411=SIGMA1*(DB421+DX423-DX413)
C
C     DIFFRENTIATION WITH RESPECT TO SIGMA5
C     ---------------------------------------------
C
      DB512=-DX523+SIGMA1*(DX523-DX514)
      DB522=DX533+SIGMA2*(DB512+DX514-DX533-DX524)
      DB532=DB522+DX524-DX543+SIGMA3*(DX543-DX534)
      DB542=DX553+SIGMA4*(DB532+DX534-DX553-DX544)
      DB552=DB542+DX544-DX563+SIGMA5*(DX563-DX554)+Y63-Y54
      DB562=DX573+SIGMA6*(DB552+DX554-DX573-DX564)
      DB572=DB562+DX564-DX572+SIGMA7*(DX572-DX574)
      DA572=DB572*PHI
      DB571=DB562+SIGMA7*(DA572-DB562+DX572-DX573)
      DB561=DB571-DB552+DX573+SIGMA6*(DB552-DX563)
      DB551=DB542+SIGMA5*(DB561-DB542+DX563-DX553)+D61-D42+Y63-Y53
      DB541=DB551-DB532+DX553+SIGMA4*(DB532-DX543)
      DB531=DB522+SIGMA3*(DB541-DB522+DX543-DX533)
      DB521=DB531-DB512+DX533+SIGMA2*(DB512-DX523)
      DB511=SIGMA1*(DB521+DX523-DX513)
C
C     DIFFRENTIATION WITH RESPECT TO SIGMA6
C     ---------------------------------------------
C
      DB612=-DX623+SIGMA1*(DX623-DX614)
      DB622=DX633+SIGMA2*(DB612+DX614-DX633-DX624)
      DB632=DB622+DX624-DX643+SIGMA3*(DX643-DX634)
      DB642=DX653+SIGMA4*(DB632+DX634-DX653-DX644)
      DB652=DB642+DX644-DX663+SIGMA5*(DX663-DX654)
      DB662=DX673+SIGMA6*(DB652+DX654-DX673-DX664)+D52+Y54-Y73-Y64
      DB672=DB662+DX664-DX672+SIGMA7*(DX672-DX674)
      DA672=DB672*PHI
      DB671=DB662+SIGMA7*(DA672-DB662+DX672-DX673)
      DB661=DB671-DB652+DX673+SIGMA6*(DB652-DX663)+D52-Y63
      DB651=DB642+SIGMA5*(DB661-DB642+DX663-DX653)
      DB641=DB651-DB632+DX653+SIGMA4*(DB632-DX643)
      DB631=DB622+SIGMA3*(DB641-DB622+DX643-DX633)
      DB621=DB631-DB612+DX633+SIGMA2*(DB612-DX623)
      DB611=SIGMA1*(DB621+DX623-DX613)
C
C     DIFFRENTIATION WITH RESPECT TO SIGMA7
C     ---------------------------------------------
C
      DB712=-DX723+SIGMA1*(DX723-DX714)
      DB722=DX733+SIGMA2*(DB712+DX714-DX733-DX724)
      DB732=DB722+DX724-DX743+SIGMA3*(DX743-DX734)
      DB742=DX753+SIGMA4*(DB732+DX734-DX753-DX744)
      DB752=DB742+DX744-DX763+SIGMA5*(DX763-DX754)
      DB762=DX773+SIGMA6*(DB752+DX754-DX773-DX764)
      DB772=DB762+DX764-DX772+SIGMA7*(DX772-DX774)+Y72-Y74
      DA772=DB772*PHI
      DB771=DB762+SIGMA7*(DA772-DB762+DX772-DX773)+C72-C62+Y72-Y73
      DB761=DB771-DB752+DX773+SIGMA6*(DB752-DX763)
      DB751=DB742+SIGMA5*(DB761-DB742+DX763-DX753)
      DB741=DB751-DB732+DX753+SIGMA4*(DB732-DX743)
      DB731=DB722+SIGMA3*(DB741-DB722+DX743-DX733)
      DB721=DB731-DB712+DX733+SIGMA2*(DB712-DX723)
      DB711=SIGMA1*(DB721+DX723-DX713)
C
C     DIFFRENTIATION WITH RESPECT TO PHI
C     ---------------------------------------------
C
      DB812=-DX823+SIGMA1*(DX823-DX814)
      DB822=DX833+SIGMA2*(DB812+DX814-DX833-DX824)
      DB832=DB822+DX824-DX843+SIGMA3*(DX843-DX834)
      DB842=DX853+SIGMA4*(DB832+DX834-DX853-DX844)
      DB852=DB842+DX844-DX863+SIGMA5*(DX863-DX854)
```

```
      DB862=DX873+SIGMA6*(DB852+DX854-DX873-DX864)
      DB872=DB862+DX864-DX872+SIGMA7*(DX872-DX874)
      DA872=DB872*PHI+D72
      DB871=DB862+SIGMA7*(DA872-DB862+DX872-DX873)
      DB861=DB871-DB852+DX873+SIGMA6*(DB852-DX863)
      DB851=DB842+SIGMA5*(DB861-DB842+DX863-DX853)
      DB841=DB851-DB832+DX853+SIGMA4*(DB832-DX843)
      DB831=DB822+SIGMA3*(DB841-DB822+DX843-DX833)
      DB821=DB831-DB812+DX833+SIGMA2*(DB812-DX823)
      DB811=SIGMA1*(DB821+DX823-DX813)
C
C     UPDATED VALUES FOR NEXT ITTERATION
C     ---------------------------------------------------
C
      Y11=C11
      Y13=D11
      Y14=D12
      Y23=D21
      Y24=D22
      Y33=D31
      Y34=D32
      Y43=D41
      Y44=D42
      Y53=D51
      Y54=D52
      Y63=D61
      Y64=D62
      Y72=C72
      Y73=D71
      Y74=D72
C
      DX113=DB111
      DX114=DB112
      DX123=DB121
      DX124=DB122
      DX133=DB131
      DX134=DB132
      DX143=DB141
      DX144=DB142
      DX153=DB151
      DX154=DB152
      DX163=DB161
      DX164=DB162
      DX172=DA172
      DX173=DB171
      DX174=DB172
C
      DX213=DB211
      DX214=DB212
      DX223=DB221
      DX224=DB222
      DX233=DB231
      DX234=DB232
      DX243=DB241
      DX244=DB242
      DX253=DB251
      DX254=DB252
      DX263=DB261
      DX264=DB262
      DX272=DA272
      DX273=DB271
      DX274=DB272
C
      DX313=DB311
      DX314=DB312
      DX323=DB321
      DX324=DB322
      DX333=DB331
      DX334=DB332
      DX343=DB341
      DX344=DB342
      DX353=DB351
      DX354=DB352
```

```
      DX363=DB361
      DX364=DB362
      DX372=DA372
      DX373=DB371
      DX374=DB372
C
      DX413=DB411
      DX414=DB412
      DX423=DB421
      DX424=DB422
      DX433=DB431
      DX434=DB432
      DX443=DB441
      DX444=DB442
      DX453=DB451
      DX454=DB452
      DX463=DB461
      DX464=DB462
      DX472=DA472
      DX473=DB471
      DX474=DB472
C
      DX513=DB511
      DX514=DB512
      DX523=DB521
      DX524=DB522
      DX533=DB531
      DX534=DB532
      DX543=DB541
      DX544=DB542
      DX553=DB551
      DX554=DB552
      DX563=DB561
      DX564=DB562
      DX572=DA572
      DX573=DB571
      DX574=DB572
C
      DX613=DB611
      DX614=DB612
      DX623=DB621
      DX624=DB622
      DX633=DB631
      DX634=DB632
      DX643=DB641
      DX644=DB642
      DX653=DB651
      DX654=DB652
      DX663=DB661
      DX664=DB662
      DX672=DA672
      DX673=DB671
      DX674=DB672
      DX713=DB711
      DX714=DB712
      DX723=DB721
      DX724=DB722
      DX733=DB731
      DX734=DB732
      DX743=DB741
      DX744=DB742
      DX753=DB751
      DX754=DB752
      DX763=DB761
      DX764=DB762
      DX772=DA772
      DX773=DB771
      DX774=DB772
C
      DX813=DB811
      DX814=DB812
      DX823=DB821
      DX824=DB822
```

```
      DX833=DB831
      DX834=DB832
      DX843=DB841
      DX844=DB842
      DX853=DB851
      DX854=DB852
      DX863=DB861
      DX864=DB862
      DX872=DA872
      DX873=DB871
      DX874=DB872
C
      C11=0.0D0
C
      WT=W*TT
      W1=DCMPLX(0.0D0,-WT)
      Z=CDEXP(W1)
      H1=H1+DB172*Z
      H2=H2+DB272*Z
      H3=H3+DB372*Z
      H4=H4+DB472*Z
      H5=H5+DB572*Z
      H6=H6+DB672*Z
      H7=H7+DB772*Z
      H8=H8+DB872*Z
      H=H+D72*Z
C     UPDATING THE TIME INCREMENT
      TT=TT+DLTAT
C
  100 CONTINUE
C
      H1=H1*(1.0D0+PHI)/2
      H2=H2*(1.0D0+PHI)/2
      H3=H3*(1.0D0+PHI)/2
      H4=H4*(1.0D0+PHI)/2
      H5=H5*(1.0D0+PHI)/2
      H6=H6*(1.0D0+PHI)/2
      H7=H7*(1.0D0+PHI)/2
      H8=H8*(1.0D0+PHI)/2+H/2
C
C     ARRANGING THE OUTPUT DATA IN FREQUENCY DOMAIN
C     ---------------------------------------------
C
      DATA0(J,1)=W
      DATA1(J,1)=W
      DATA2(J,1)=W
      DATA3(J,1)=W
      DATA4(J,1)=W
      DATA5(J,1)=W
      DATA6(J,1)=W
      DATA7(J,1)=W
      DATA8(J,1)=W
C
C     ARRANGING THE NORMALIZED OUTPUT DERIVATIVES WITH RESPECT TO
C     SIGMA'S,AND PHI
C
      DATA0(J,2)=0.0D0
      DATA1(J,2)=CDABS(H1)*SIGMA1
      IF(RS.LE.1.0D-15) DATA1(J,2)=CDABS(H1)
C     IF RS=0 THEN SIGMA1=0  THUS FOR THIS CASE ONLY NON NORMALIZED
C     VALUES FOR DERIVATIVES ARE CALCULATED AND PLOTED
      DATA2(J,2)=CDABS(H2)*SIGMA2
      DATA3(J,2)=CDABS(H3)*SIGMA3
      DATA4(J,2)=CDABS(H4)*SIGMA4
      DATA5(J,2)=CDABS(H5)*SIGMA5
      DATA6(J,2)=CDABS(H6)*SIGMA6
      DATA7(J,2)=CDABS(H7)*SIGMA7
      DATA8(J,2)=CDABS(H8)*DABS(PHI)
C
C
C
      T0=A1*H1+A2*H2+A3*H3+A4*H4+A5*H5+A6*H6+A7*H7+A8*H8
      T1=B1*H1+B2*H2+B3*H3+B4*H4+B5*H5+B6*H6+B7*H7+B8*H8
```

```
      T2=D2*H2+D3*H3+D4*H4+D5*H5+D6*H6+D7*H7+D8*H8
      T3=E3*H3+E4*H4+E5*H5+E6*H6+E7*H7+E8*H8
      T4=P4*H4+P5*H5+P6*H6+P7*H7+P8*H8
      T5=Q5*H5+Q6*H6+Q7*H7+Q8*H8
      T6=S6*H6+S7*H7+S8*H8
      T7=U7*H7+U8*H8
      T8=V8*H8
C
C
C     ARRANGING THE NORMALIZED OUTPUT DERIVATIVES WITH RESPECT TO
C     L'S,C'S,RS,AND RL
C     NOTE THAT FOR THE CASE OF RS=0 NON NORMALIZED VALUES ARE
C     ARRANGED FOR SECTION ONE ONLY
C
      DATA0(J,3)=CDABS(T0)*RS
      IF(RS.LE.1.0D-15) DATA0(J,3)=CDABS(T0)
      DATA1(J,3)=CDABS(T1)*AL1
      IF(RS.LE.1.0D-15) DATA1(J,3)=CDABS(T1)
      DATA2(J,3)=CDABS(T2)*C2
      DATA3(J,3)=CDABS(T3)*AL3
      DATA4(J,3)=CDABS(T4)*C4
      DATA5(J,3)=CDABS(T5)*AL5
      DATA6(J,3)=CDABS(T6)*C6
      DATA7(J,3)=CDABS(T7)*AL7
      DATA8(J,3)=CDABS(T8)*RL
C
C     UPDATING THE FREQUENCY INCREMENT
      W=W+DLTAW
  110 CONTINUE
C
C
C
      WRITE(6,54)
   54 FORMAT('1')
      IF(RS.GT.1.0D-15) GO TO 555
      WRITE(6,50)
   50 FORMAT(35X,'NON NORMALIZED VALUES',//)
      WRITE(6,49)
   49 FORMAT(20X,'FREQ',40X,'D(O/P)/D(RS)',//)
      GO TO 666
  555 CONTINUE
      WRITE(6,48)
   48 FORMAT(20X,'FREQ',40X,'(D(O/P)/D(RS))*RS',//)
  666 CONTINUE
      WRITE(6,20) ((DATA0(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DATA0,98,4HFREQ,4HMAG.)
C
      WRITE(6,54)
      IF(RS.GT.1.0D-15) GO TO 333
      WRITE(6,50)
      WRITE(6,51)
   51 FORMAT(20X,'FREQ',15X,'D(O/P)/D(SIGMA1)',9X,'D(O/P)/D(RL)',//)
      GO TO 444
  333 CONTINUE
      WRITE(6,55)
   55 FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA1))*SIGMA1',4X,'(D(O/P)/D(L1
     *))*L1',//)
  444 CONTINUE
      WRITE(6,20) ((DATA1(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DATA1,98,4HFREQ,4HMAG.)
C
      WRITE(6,54)
      WRITE(6,56)
   56 FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA2))*SIGMA2',4X,'(D(O/P)/D(L2
     *))*C2',//)
      WRITE(6,20) ((DATA2(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DATA2,98,4HFREQ,4HMAG.)
C
      WRITE(6,54)
      WRITE(6,57)
   57 FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA3))*SIGMA3',4X,'(D(O/P)/D(L3
     *))*L3',//)
      WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,98)
```

```
      CALL GRAPHX(DATA3,98,4HFREQ,4HMAG.)
C
      WRITE(6,54)
      WRITE(6,58)
 58   FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA4))*SIGMA4',4X,'(D(O/P)/D(L4
     *))*C4',//)
      WRITE(6,20) ((DATA4(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DATA4,98,4HFREQ,4HMAG.)
C
      WRITE(6,54)
      WRITE(6,59)
 59   FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA5))*SIGMA5',4X,'(D(O/P)/D(L5
     *))*L5',//)
      WRITE(6,20) ((DATA5(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DATA5,98,4HFREQ,4HMAG.)
C
      WRITE(6,54)
      WRITE(6,60)
 60   FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA6))*SIGMA6',4X,'(D(O/P)/D(L6
     *))*C6',//)
      WRITE(6,20) ((DATA6(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DATA6,98,4HFREQ,4HMAG.)
C
      WRITE(6,54)
      WRITE(6,61)
 61   FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA7))*SIGMA7',4X,'(D(O/P)/D(L7
     *))*L7',//)
      WRITE(6,20) ((DATA7(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DATA7,98,4HFREQ,4HMAG.)
C
      WRITE(6,54)
      WRITE(6,62)
 62   FORMAT(20X,'FREQ',11X,'(D(O/P)/D(PHI))*ABS(PHI)',4X,'(D(O/P)/D(RL)
     *)*RL',//)
      WRITE(6,20) ((DATA8(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DATA8,98,4HFREQ,4HMAG.)
C
C
 20   FORMAT(20X,E12.5 ,10X,E12.5 ,10X,E12.5)
C
      STOP
      END
```

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Professor Sydney R. Parker, Code 62Px Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	10
4.	Professor Richard W. Hamming, Code 52Hg Department of Computer Sciences Naval Postgraduate School Monterey, California 93940	1
5.	CDR Masoud Sanaie-Fard Iranian Navy c/o National Iranian Navy Training Office Suite 1110 800 North Quincy Street Arlington, Virginia 22203	3
6.	Department of Electrical Engineering Univerity of Tehran Tehran, IRAN	1
7.	Department of Electrical Engineering University of Azerbaijan Tabriz, IRAN	1
8.	Professor B. Peikari Electrical Engineering Department Institute of Technology Southern Methodist University Dallas, Texas 75275	1
9.	Professor Donald E. Kirk, Code 62 Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1