TWO-DIMENSIONAL DIGITAL FILTERING AND ITS UNCORRELATED ERROR ANALYSIS*

by

Ming-Duenn Ni and J. K. Aggarwal
Department of Electrical Engineering

Technical Report No. 175
July 15, 1975

INFORMATION SYSTEMS RESEARCH LABORATORY

ELECTRONICS RESEARCH CENTER
THE UNIVERSITY OF TEXAS AT AUSTIN
Austin, Texas 78712

*Research supported in part by the Joint Services Electronics Program under JSEP Contract F44620-71-C-0091 and by the National Science Foundation under Grant GK-42790.

ABSTRACT

The concept of a two-dimensional recursive digitial filter is introduced and block diagram representations are given. Error analyses for both floating-point and fixed-point two-dimensional digital filters are carried out. A systematic way of estimating the mean squared errors due to roundoff, coefficient and input quantizations is discussed. Norm error bounds are also derived. Simulations of the implementations of digital filters are discussed, and corresponding to these simulations error calculations are performed. Numerical examples are given. The derived analytic results are shown to be in good agreement with the simulation results.

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRODUCTION

Digital filtering of two-dimensional digital data is needed in many applications. For example, the processing of seismic records, gravity and magnetic data, and scene analysis and picture processing require two-dimensional filtering. Prior to 1965, the implementation of the two-dimensional digital filtering processes mostly used the two-dimensional direct convolution algorithm which is characterized by the double sum:

$$w_{mn} = \sum_{j=0}^{m} \sum_{k=0}^{n} g_{jk} \, x_{m-j,\, n-k}$$

where $\{x_{jk}\}$ is the input sequence, which, for example, is the digitized recorded image, $\{g_{jk}\}$ is the "weight matrix", which corresponds to the impulse response in the one-dimensional case, and $\{w_{jk}\}$ is the output sequence, which, for example, is the enhanced image. In this method the output, w_{mn}, is the weighted sum of all past values of the input sequence, i.e. $\{x_{jk}\}$, $0 \le j \le m$, and $0 \le k \le n$. A variation of the direct convolution technique is the method where a partial set of the past values of the input sequence is used to obtain the output w_{mn}. Direct convolution has a serious drawback, its requirements of a large number of arithmetic operations (multiplications and additions). The FFT algorithm, discovered in 1965, can provide a great deal of reduction in the number of arithmetic operations, and is being widely used. The FFT makes filtering feasible by use of the frequency domain equation,

$$W(\omega_1, \omega_2) = G(\omega_1, \omega_2)\, X(\omega_1, \omega_2) \ ,$$

where $W(\omega_1, \omega_2)$, $G(\omega_1, \omega_2)$ and $X(\omega_1, \omega_2)$ are discrete Fourier transforms of the sequences $\{w_{mn}\}$, $\{g_{mn}\}$, and $\{x_{mn}\}$ respectively, which are assumed to be periodic, and ω_1 and ω_2 are the two spatial radian frequencies.

The recursive algorithm provides another technique for the implementation of the filtering processes, especially for very large amounts of data. It is already known that the recursive techniques are more powerful (in the sense that its computational and memory requirements are fewer) than the FFT algorithm in a large number of one-dimensional cases. Therefore, it is desirable to study the related problems in two-dimensional recursive filtering.

There are several reports discussing two-dimensional recursive digital filters ([1], [2], [3]). For example, Shanks, Treitel and Justice ([2]) formulate two-dimensional recursive filters by the two-dimensional Z-transform and linear difference equations. They also study the stability of the filters and extend synthesis methods in the one-dimensional case to the two-dimensional case. Huang ([3]) simplifies Shanks' stability theorems ([1]). Farmer and Gooden ([21]) describe some of the computational problems associated with the approximation of unstable recursive digital filters by stable recursive digital filters. Hall ([22]) compares the computation required for the three spatial frequency filtering techniques which are direct convolution, fast Fourier transform, and recursive filtering. Other reports which touch upon two-dimensional recursive filtering are ([23], [24], [25]).

Besides considering the stability and the synthesis problems ([1], [2], [3]), we must also consider the effects of finite word length in the design of two-dimensional recursive digital filters. As in one-dimensional digital filters, the effects of finite word length also lead to the following three sources of error; namely, (1) the quantizations of input signals and initial states, (2) the quantizations of the filter coefficients, and (3) the rounding off of arithmetic operations. These sources of error cause the actual outputs of the digital filters to be different from the ideal outputs.

It is important to know whether a certain accuracy can be achieved if the digital filters are simulated on general purpose computers where the word length is usually fixed, or to determine the minimum word length needed for a specific performance accuracy if the digital filters are constructed with digital hardware. The effects of the three sources of error and the methods of analyzing them generally depend on the types of arithmetic (e.g.,fixed-point arithmetic, floating-point arithmetic, or block floating-point arithmetic). There are many reports on fixed-point one-dimensional digital filters. For example, Jackson ([16], [17]) analyzed roundoff noise for digital filters realized in various forms (i.e.,direct, parallel, and cascade forms). Gold and Rader ([20]) studied the effects of parameter quantization on the poles of a digital filter. Jackson ([18]) and Ebert and etc. ([19]) investigated the limit cycle oscillations due to roundoff after multiplication and overflow at the adder. There are also some papers on floating-point one-dimensional digital filters. Sandberg ([4]) derived an absolute upper bound on the error accumulations due to roundoffs in arithmetic operations. Kaneko and Liu ([8], [9]) derived an expression for the mean squared error caused by roundoff error accumulations assuming that the input signal is zero mean and wide sense stationary. Kan and Aggarwal ([6]) studied the error properties of digital filters realized in canonical form. Oppenheim and Weinstein ([13], [14]) examined and tested the roundoff errors of first and second order digital filters with zeros at infinity and gave expressions of output error to signal ratios for white noise inputs. Generally, the methods (e.g., [4], [6], [8], [9], [13], [14] and etc.) and the expressions (e.g., [4], [6], [8], and etc.) for roundoff errors are quite readily extended to recursive digital filters. However, a fundamental difficulty of two-dimensional digital filters is that we generally cannot factorize a two-dimensional recursive digital filter into a multiplication of lower order digital filters. This fundamental difficulty has caused serious problems in the designing and analyzing of two-dimensional recursive digital filters. For example, we can obtain integral results in closed form for the expressions given by Kaneko and Liu ([8], [9]), but when the expressions are extended to two dimensions we generally cannot get similar results, although numerical approximations are possible.

In this manuscript we review some basic definitions and consequences and give block diagram representations for two-dimensional digital filters. We formulate a systematic way of estimating the output mean squared errors and the output norm error bounds for output errors due to the effects of finite word length for both fixed-point and floating-point two-dimensional digital filters. We give several examples to demonstrate the validity of the method. We also have two appendices. Appendix I describes the properties of the two-dimensional Z-transform needed for the development of the present report. Appendix I also proves a lemma which is essential to the derivation of the output norm error bounds. Appendix II lists some of the programs used to obtain the numerical results. "Two-dimensional" will be designated by "2D" henceforth for brevity.

II. 2D DIGITAL FILTERS

A one-dimensional recursive digital filter is described by the linear difference equation:

$$w_n = \sum_{k=0}^{N} b_k x_{n-k} - \sum_{k=1}^{M} a_k w_{n-k}, \qquad (1)$$

where (i) $n \geq 0$,

(ii) $N < M$,

(iii) $\{x_n\}$ is the input sequence, and
$x_i = 0$, for $i < 0$,

(iv) $\{w_n\}$ is the output sequence, and
$w_j = W(j)$, $j = -1, \ldots, -M$
$= 0$, $j < -M$.

Programs for a general purpose digital computer may be written or special purpose hardware may be built for performing the computation of Eq. (1). Extending the algorithm of Eq. (1) to two dimensions, one gets the following 2D computational algorithm:

$$w_{mn} = \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk}\, x_{m-j,n-k} - \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk} w_{m-j,n-k}, \qquad (2)$$

where (i) $m \geq 0$ and $n \geq 0$,

(ii) $M_b \cdot N_b \leq M_a \cdot N_a$,

(iii) $\{x_{mn}\}$ is the input sequence, and
$x_{jk} = 0$, for $j < 0$ or $k < 0$,

(iv) $\{w_{mn}\}$ is the output sequence, and

$$w_{jk} = W(j,k), \quad k = 0,-1,\ldots,-N_a, \quad j = 0,-1,\ldots,-M_a, \quad j+k \neq 0,$$

$$w_{jk} = 0, \text{ for } k < -N_a \text{ or } j < -M_a.$$

Equation (2) is a 2D linear difference equation and represents a 2D linear discrete system. When the input sequence $\{x_{mn}\}$, the output sequence $\{w_{mn}\}$, and the coefficients a's and b's are digital quantities, it is called a 2D digital filter. Further, if all of the a_{jk}'s ($j+k \neq 0$) are zero, it is nonrecursive, otherwise it is recursive.

In Appendix I, the 2D Z-transform is defined and some of its salient properties are listed. Its use in 2D digital filters is discussed in the following. Taking the 2D Z-transform of the 2D linear difference Eq. (2), and assuming all initial conditions are zero, one gets

$$W(z_1,z_2) = G(z_1,z_2) \cdot X(z_1,z_2) \tag{3}$$

where

$$G(z_1,z_2) = \frac{N(z_1,z_2)}{D(z_1,z_2)} \tag{4}$$

$$= \frac{\displaystyle\sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk} z_1^{-j} z_2^{-k}}{\displaystyle 1 + \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk} z_1^{-j} z_2^{-k}} . \tag{5}$$

$G(z_1,z_2)$ is called a 2D digital transfer function, and may be used to represent a 2D digital filter.

A 2D unit point function is the sequence $\{x_{mn}\}$ such that

$$x_{mn} = 1, \; m = n = 0,$$
$$= 0, \; \text{otherwise},$$

and the response of a digital filter to such an input is called its "point spread response". If

$$G(z_1, z_2) = \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} g_{jk} z_1^{-j} z_2^{-k} , \tag{6}$$

the point spread response is given by the 2D sequence

$$\{g_{jk}\}_{j=0}^{\infty} {}_{k=0}^{\infty} , \tag{7}$$

furthermore, the response of the filter to the input sequence $\{x_{mn}\}$ is given by

$$w_{mn} = \sum_{j=0}^{m} \sum_{k=0}^{n} x_{jk} \; g_{m-j,n-k} = \sum_{j=0}^{m} \sum_{k=0}^{n} x_{m-j,n-k} \; g_{jk} . \tag{8}$$

In the above Eqs. (6), (7), and (8), it is assumed that the digital filter is causal. The inversion formula and other techniques for obtaining point spread response (7) from the transfer function (5) are discussed in Appendix I.

The difference Eq. (2) can be represented by the block diagram as shown in Figure 1. The 2D Z-transform relationship is given by Eqs. (3), (4), and (5). There are many block diagrams with $G(z_1, z_2)$ equivalent to that of Figure 1. Different block diagrams signify different implementations of $G(z_1, z_2)$, and lead to different error properties. Figure 2 is an example in which the block diagram is described by the following pair of equations,

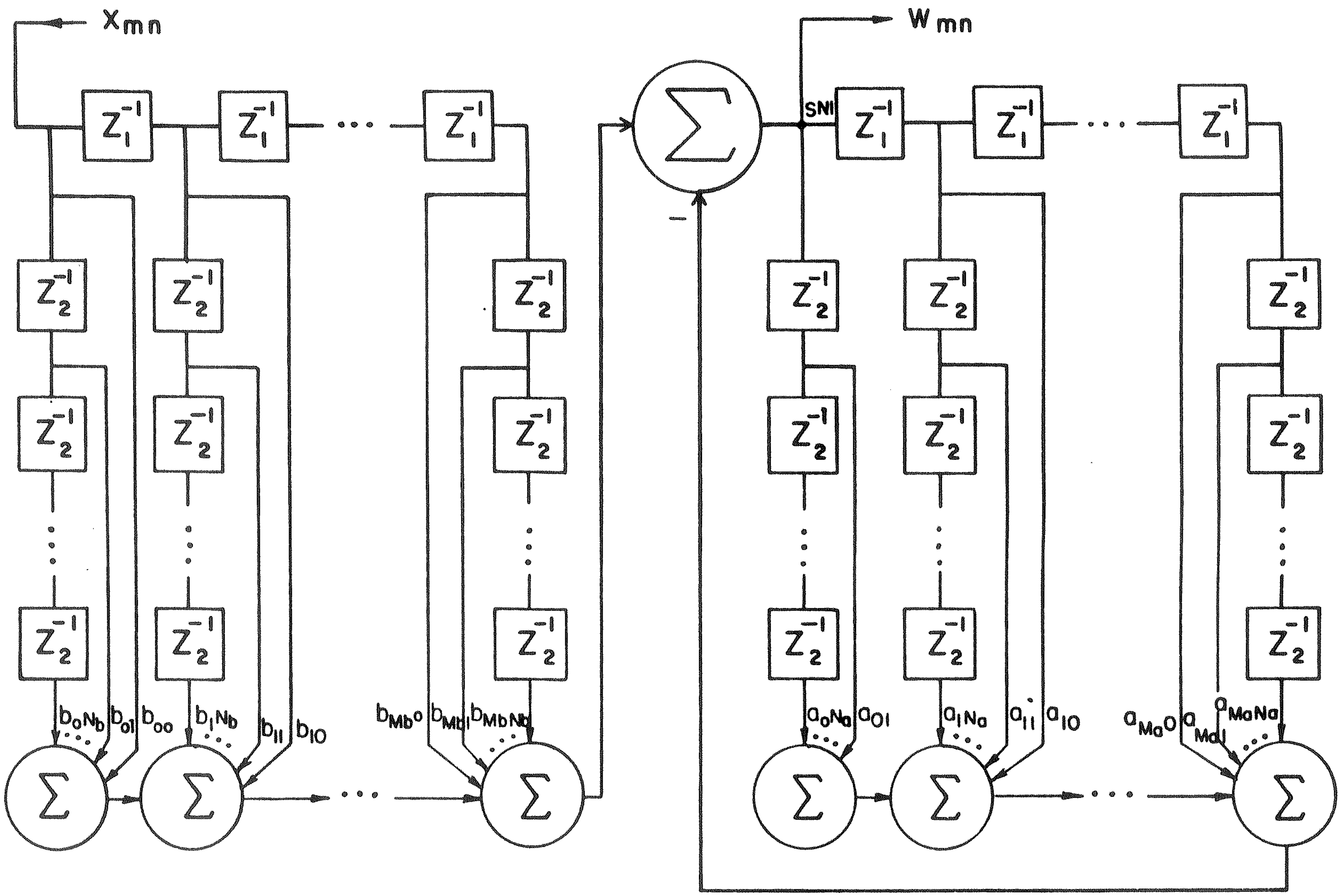

Fig. 1 Block Diagram Representation for Direct Filtering Process

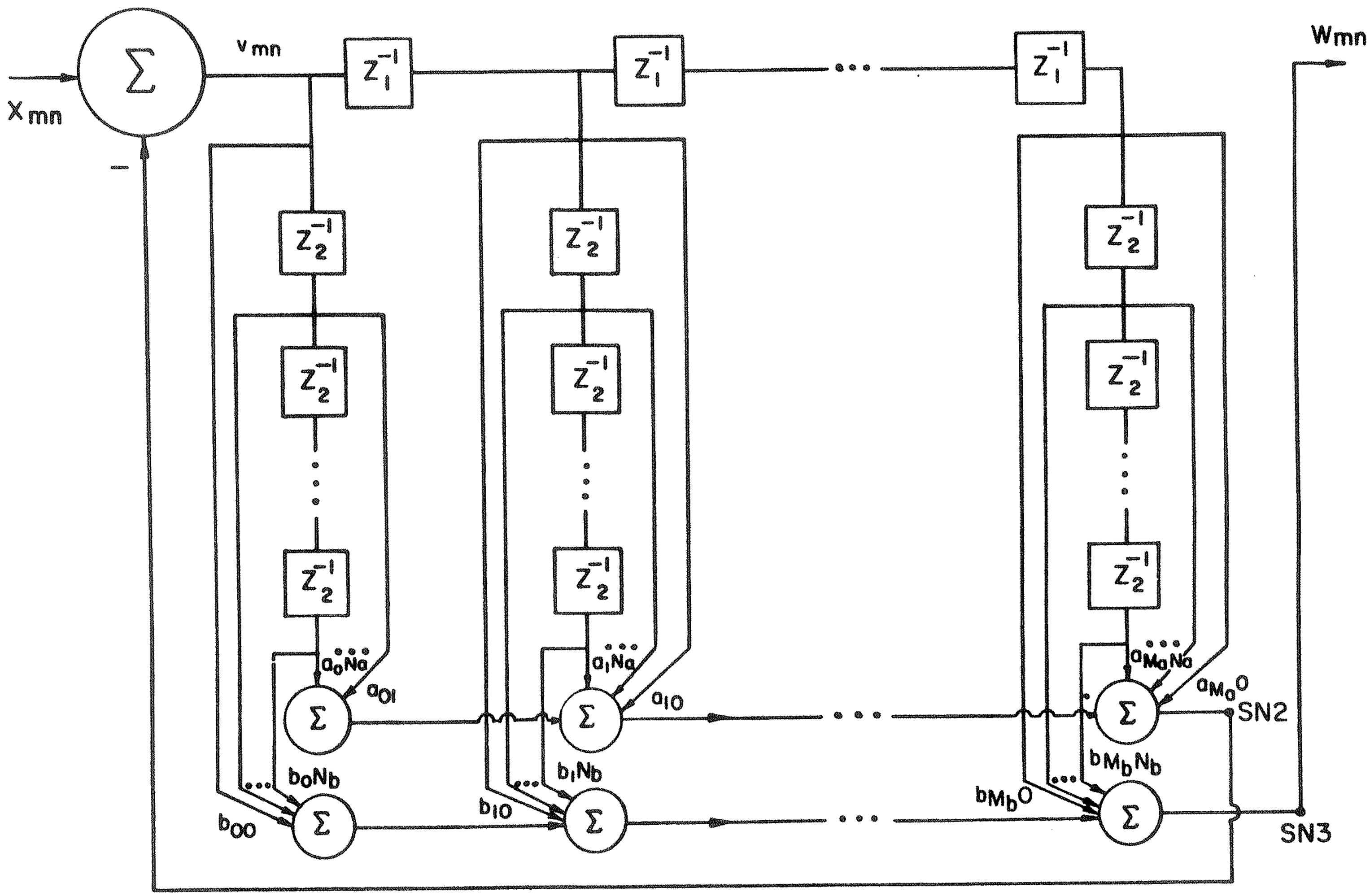

Fig. 2 Block Diagram Representation for Canonic Filtering Process

$$v_{mn} = - \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk} v_{m-j,n-k} + x_{mn}, \tag{9}$$

$$w_{mn} = \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk} v_{m-j,n-k}, \tag{10}$$

where (i) $m \geq 0$ and $n \geq 0$,

(ii) $M_b \cdot N_b \leq M_a \cdot N_a$,

(iii) $\{x_{mn}\}$ is the input sequence,

$x_{jk} = 0$, for $j < 0$ or $k < 0$,

(iv) $\{w_{mn}\}$ is the output sequence,

$w_{jk} = 0$, for $j < 0$ or $k < 0$,

(v) $\{v_{mn}\}$ is the state sequence,

$v_{jk} = V(j,k)$, $j = 0,-1,\ldots,-M_a$, $k = 0,-1,\ldots,-N_a$, $j + k \neq 0$,

$v_{jk} = 0$, for $j < -M_a$ or $k < -N_a$.

In the following sections, Eq. (2) will be called the "direct filtering process", and Eqs. (9) and (10) the "canonic filtering process".

<u>Definition:</u>

The "2D sequence mean square average" norm is defined as

$$\langle x \rangle_{p\ ,\ q}^{K_2,\ K_1} \triangleq \left(\frac{1}{(K_2-p+1)(K_1-q+1)} \sum_{m=p}^{K_2} \sum_{n=q}^{K_1} |x_{mn}|^2 \right)^{1/2}$$

for every real-valued sequence $\{x_{mn}\}$, and every $K_2, K_1 \in I^+$ and every $p,q \in I$, where I^+ is the set of positive integers, and I is the set of integers.

$$[\langle x\rangle_{p,q}^{K_2,K_1}]^2 \text{ is denoted by } {}^2\langle x\rangle_{p,q}^{K_2,K_1} .$$

When both p and q are equal to zero, p and q are omitted, e.g.,

$\langle x\rangle_{p,q}^{K_2,K_1}$ is denoted by $\langle x\rangle^{K_2,K_1}$.

The following lemma is used in the derivation of error bounds. The proof is given in Appendix I.

Lemma 1

If $$f_{mn} = \sum_{k=0}^{m} \sum_{l=0}^{n} C_{m-k,n-l}\, g_{kl}, \text{ for } m \geq 0 \text{ and } n \geq 0$$

with $$\sum_{k=0}^{\infty} \sum_{l=0}^{\infty} |C_{kl}| < \infty,$$

then $$\langle f\rangle^{K_2,K_1} \leq \max_{\substack{0\leq\omega_1\leq 2\pi \\ 0\leq\omega_2\leq 2\pi}} \left| \sum_{k=0}^{\infty} \sum_{l=0}^{\infty} C_{kl} e^{-ik\omega_1} e^{-il\omega_2} \right| \langle g\rangle^{K_2,K_1}$$

for $K_1 \geq 0$ and $K_2 \geq 0$, where $i = \sqrt{-1}$.

We also need the following representations;

$$\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} h_{mn} z_1^{-m} z_2^{-n} = \frac{1}{D(z_1,z_2)} , \tag{11}$$

$$\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} h_{mn}^{(jk)} z_1^{-m} z_2^{-n} = \frac{\sum_{p=j}^{M_a} \sum_{q=k}^{N_a} a_{pq} z_1^{-(p-j)} z_2^{-(q-k)}}{D(z_1,z_2)} , \tag{12}$$

$j=0,\ldots,M_a$,

$k=0,\ldots,N_a$,

$j+k \neq 0$.

III. ERROR ANALYSIS FOR TWO-DIMENSIONAL DIGITAL FILTERS EMPLOYING FIXED-POINT ARITHMETIC

In this chapter, we analyze the effects of finite word length on two-dimensional digital filters employing fixed-point arithmetic. We first find out the actual system equations with finite word length, and then derive formulas for estimating (1) the output mean squared errors and (2) the output norm error bounds. We only consider the case of roundoff errors. By following a similar approach we can obtain similar results for other sources of error. We study both the direct filtering process and the canonic filtering process.

III.1 Actual System Equations with Roundoff Errors

In the implementation of digital filter Eqs. (2), (9), and (10), the finite word length of registers in a general purpose digital computer or special purpose digital hardware produces modification of these difference equations. The actual difference equations corresponding to Eqs. (2), (9), and (10), while taking into account the multiplication roundoff errors, are given here. We assume that:

(i) each machine number q is normalized, so that $|q| < 1$ and is represented by

$$-q_0 + \sum_{k=1}^{t-1} q_k 2^{-k},$$

where t is the number of bits. The q_k's take on values "0" or "1". The following error properties are assumed:

$$fi[\bar{x}] = \bar{x} + \epsilon = x \ , \ |\epsilon| < 2^{-t}, \tag{13}$$

$$fi[x+y] = x + y, \tag{14}$$

$$fi[xy] = xy + \delta \ , \ |\delta| < 2^{-t}, \tag{15}$$

where $\bar{x}$ is a real number, and x and y are machine numbers. It may be observed that the error δ is zero if x or y is zero.

(ii) there is no overflow unless otherwise mentioned.

(iii) the numbers a_{jk}'s, b_{jk}'s, $\{x_{mn}\}$, $\{W(-j,-k)\}_{\substack{j=0,k=0\\ j+k\neq 0}}^{M_a,N_a}$

and $\{V(-j,-k)\}_{\substack{j=0,k=0\\ j+k\neq 0}}^{M_a,N_a}$ are machine numbers.

(iv) the digital filter of Eq. (2) or (9) and (10) is stable ([1],[2]), and we let

$$fi[b_{jk}x_{m-j,n-k}] = b_{jk}x_{m-j,n-k} + \delta_{mn,jk},$$

$$fi[a_{jk}y_{m-j,n-k}] = a_{jk}y_{m-j,n-k} + \eta_{mn,jk},$$

$$fi[b_{jk}u_{m-j,n-k}] = b_{jk}u_{m-j,n-k} + \gamma_{mn,jk},$$

$$fi[a_{jk}u_{m-j,n-k}] = a_{jk}u_{m-j,n-k} + \epsilon_{mn,jk},$$

where the symbols $\delta_{mn,jk}$, $\eta_{mn,jk}$, $\gamma_{mn,jk}$ and $\epsilon_{mn,jk}$ play the same role as δ in Eq. (15) above, and take on values in the interval $(-2^{-t}, 2^{-t})$. The ordering of the arithmetic operations of Eqs. (2), (9) and (10) are according to the flow graphs shown in Figures 3 and 4.

The above assumptions lead to the following actual system of equations:

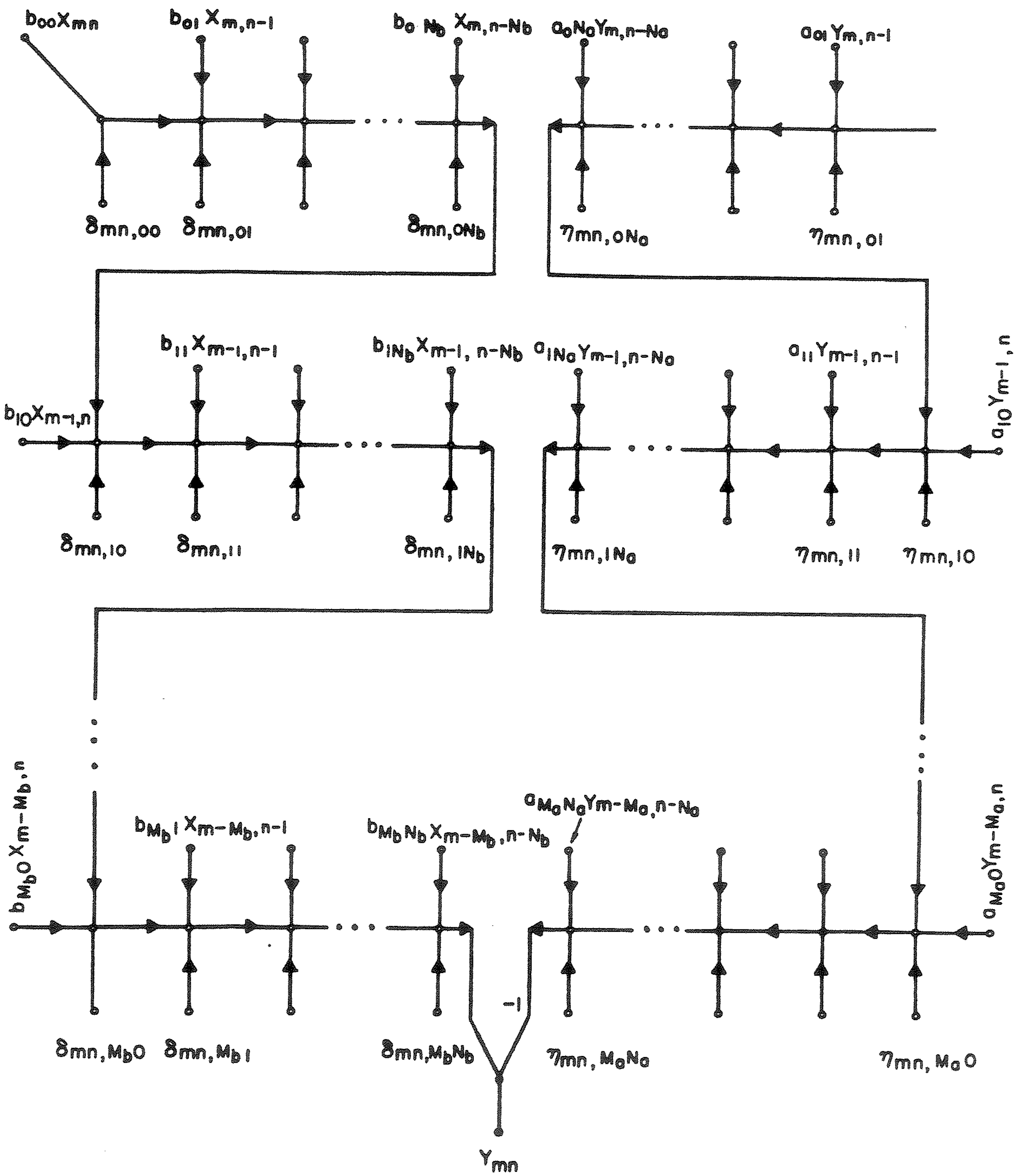

Fig. 3 Flow Graph for Fixed-Point Direct Filtering Process

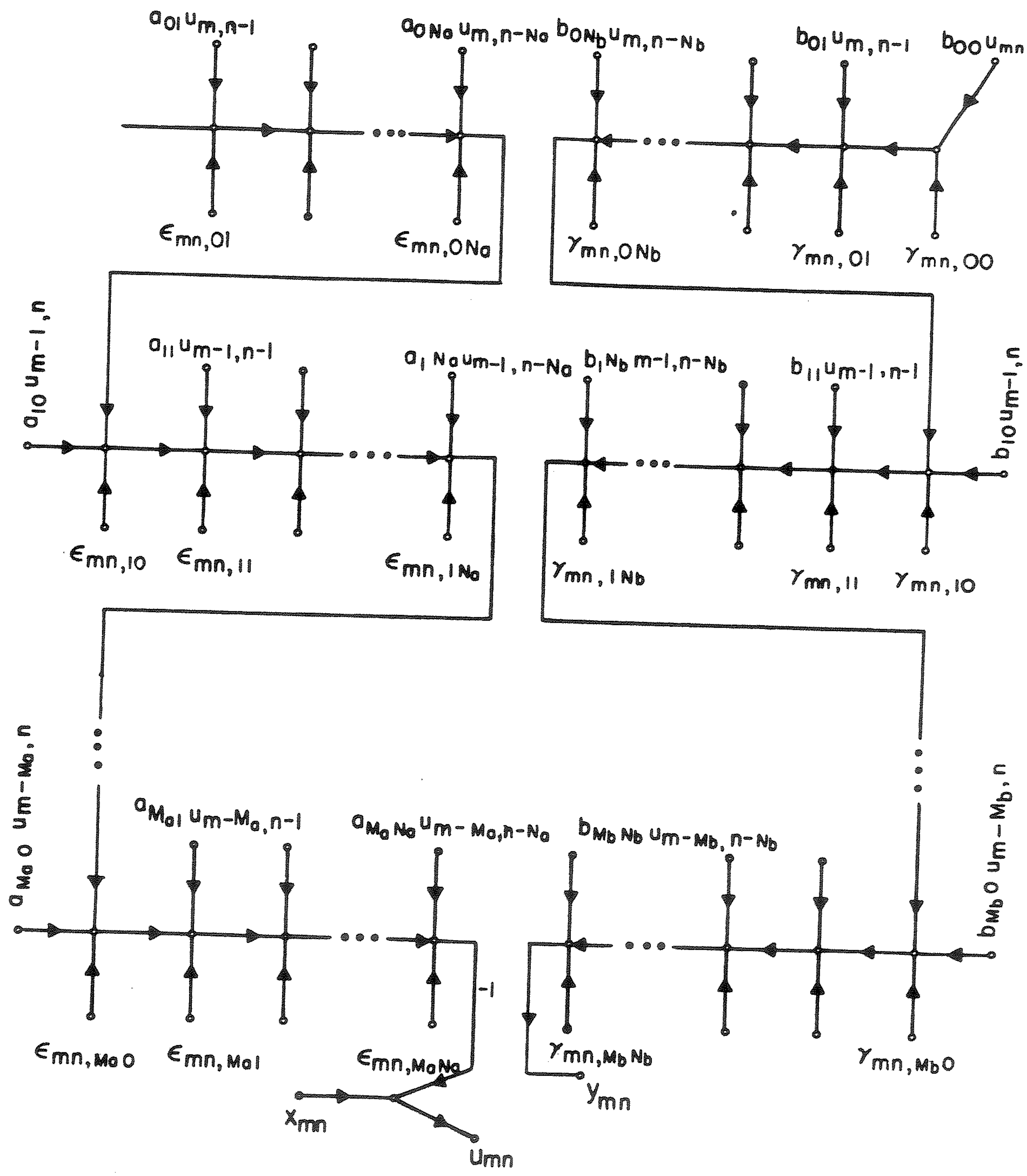

Fig. 4 Flow Graph for Fixed-Point Canonic Filtering Process

(1) Direct Filtering Process

$$y_{mn} = fi\left[\sum_{j=0}^{M_b}\sum_{k=0}^{N_b} b_{jk}x_{m-j,n-k} - \sum_{\substack{j=0\\ j+k\neq 0}}^{M_a}\sum_{k=0}^{N_a} a_{jk}y_{m-j,n-k}\right]$$

$$= \sum_{j=0}^{M_b}\sum_{k=0}^{N_b} b_{jk}x_{m-j,n-k} - \sum_{\substack{j=0\\ j+k\neq 0}}^{M_a}\sum_{k=0}^{N_a} a_{jk}y_{m-j,n-k} \tag{16}$$

$$+ e_{mn}^{(d)}\ , \ m \geq 0 \text{ and } n \geq 0,$$

where

$$e_{mn}^{(d)} = \begin{cases} \displaystyle\sum_{j=0}^{M_b}\sum_{k=0}^{N_b} \delta_{mn,jk} - \sum_{\substack{j=0\\ j+k\neq 0}}^{M_a}\sum_{k=0}^{N_a} \eta_{mn,jk}, & \text{for } m \geq 0 \text{ and } n \geq 0, \\ 0, \text{ otherwise}, & \end{cases} \tag{17}$$

(2) Canonic Filtering Process

$$u_{mn} = fi\left[-\sum_{\substack{j=0\\ j+k\neq 0}}^{M_a}\sum_{k=0}^{N_a} a_{jk}\, u_{m-j,n-k} + x_{mn}\right] \tag{18}$$

$$= -\sum_{\substack{j=0\\ j+k\neq 0}}^{M_a}\sum_{k=0}^{N_a} a_{jk}u_{m-j,n-k} + e_{mn}^{(1)} + x_{mn}, \text{ for } m \geq 0 \text{ and } n \geq 0,$$

$$y_{mn} = fi\left[\sum_{j=0}^{M_b}\sum_{k=0}^{N_b} b_{jk}\, u_{m-j,n-k}\right]$$

$$= \sum_{j=0}^{M_b}\sum_{k=0}^{N_b} b_{jk} u_{m-j,n-k} + e_{mn}^{(2)}, \text{ for } m \geq 0 \text{ and } n \geq 0, \tag{19}$$

where

$$e_{mn}^{(1)} = \begin{cases} -\sum\limits_{\substack{j=0 \\ j+k\neq 0}}^{M_a}\sum\limits_{k=0}^{N_a} \epsilon_{mn,jk}, & m \geq 0 \text{ and } n \geq 0, \\ 0, \text{ otherwise}, \end{cases} \tag{20}$$

$$e_{mn}^{(2)} = \begin{cases} \sum\limits_{j=0}^{M_b}\sum\limits_{k=0}^{N_b} \gamma_{mn,jk}, & m \geq 0 \text{ and } n \geq 0, \\ 0, \text{ otherwise}. \end{cases} \tag{21}$$

III.2 Error Analysis

III.2.1 Behavior of the Sources of Error

Roundoff Error Accumulation

Let

$$e_{mn} = y_{mn} - w_{mn}, \tag{22}$$

$$e'_{mn} = u_{mn} - v_{mn}. \tag{23}$$

Then, for the direct filtering process, by Eqs. (2) and (16),

$$e_{mn} = -\sum_{\substack{j=0 \\ j+k\neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk}\, e_{m-j,n-k} + e_{mn}^{(d)}, \tag{24}$$

or

$$\sum_{j=0}^{M_a} \sum_{k=0}^{N_a} a_{jk} e_{m-j,n-k} = e_{mn}^{(d)}, \text{ with } a_{00} = 1, \tag{25}$$

and for the canonic filtering process, by Eqs. (9), (10), (18) and (19),

$$e'_{mn} = -\sum_{\substack{j=0 \\ j+k\neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk}\, e'_{m-j,n-k} + e_{mn}^{(1)} \tag{26}$$

or

$$\sum_{j=0}^{M_a} \sum_{k=0}^{N_a} a_{jk} e'_{m-j,n-k} = e_{mn}^{(1)}, \text{ with } a_{00} = 1, \tag{27}$$

$$e_{mn} = \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk} e'_{m-j,n-k} + e_{mn}^{(2)} . \tag{28}$$

Equations (24), (25), (26), (27) and (28) can be best illustrated by Figures 5(a) and 6(a), where the roundoff errors $\{e_{mn}^{(d)}\}$, $\{e_{mn}^{(1)}\}$, and $\{e_{mn}^{(2)}\}$ act as noise sources injected at the indicated junctions.

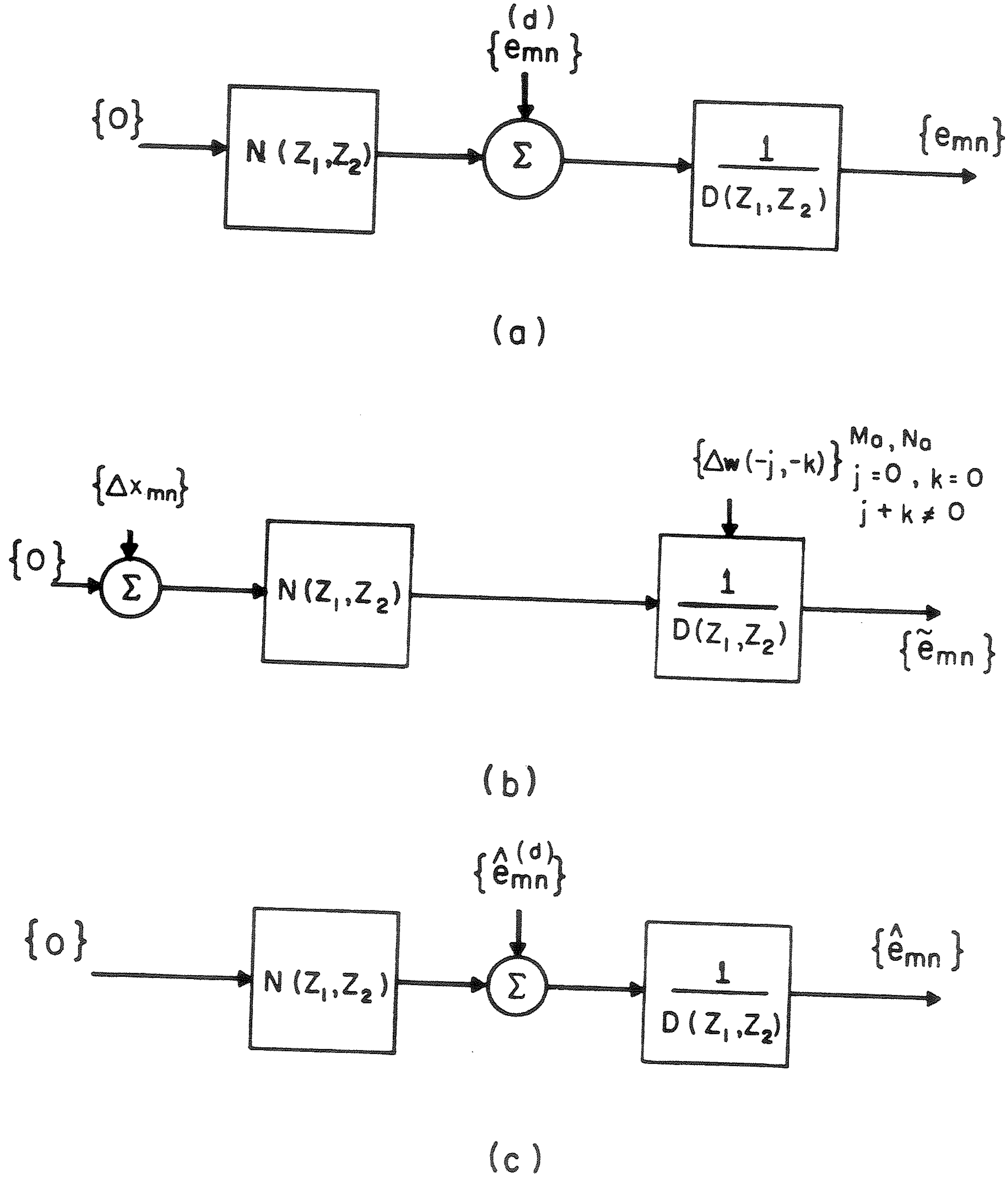

Fig. 5 Block Diagram Interpretations for Direct Filtering Process

(a) Error $\{e_{mn}\}$ due to roundoff

(b) Error $\{\tilde{e}_{mn}\}$ due to quantization of inputs and initial values

(c) Error $\{\hat{e}_{mn}\}$ due to coefficient quantization

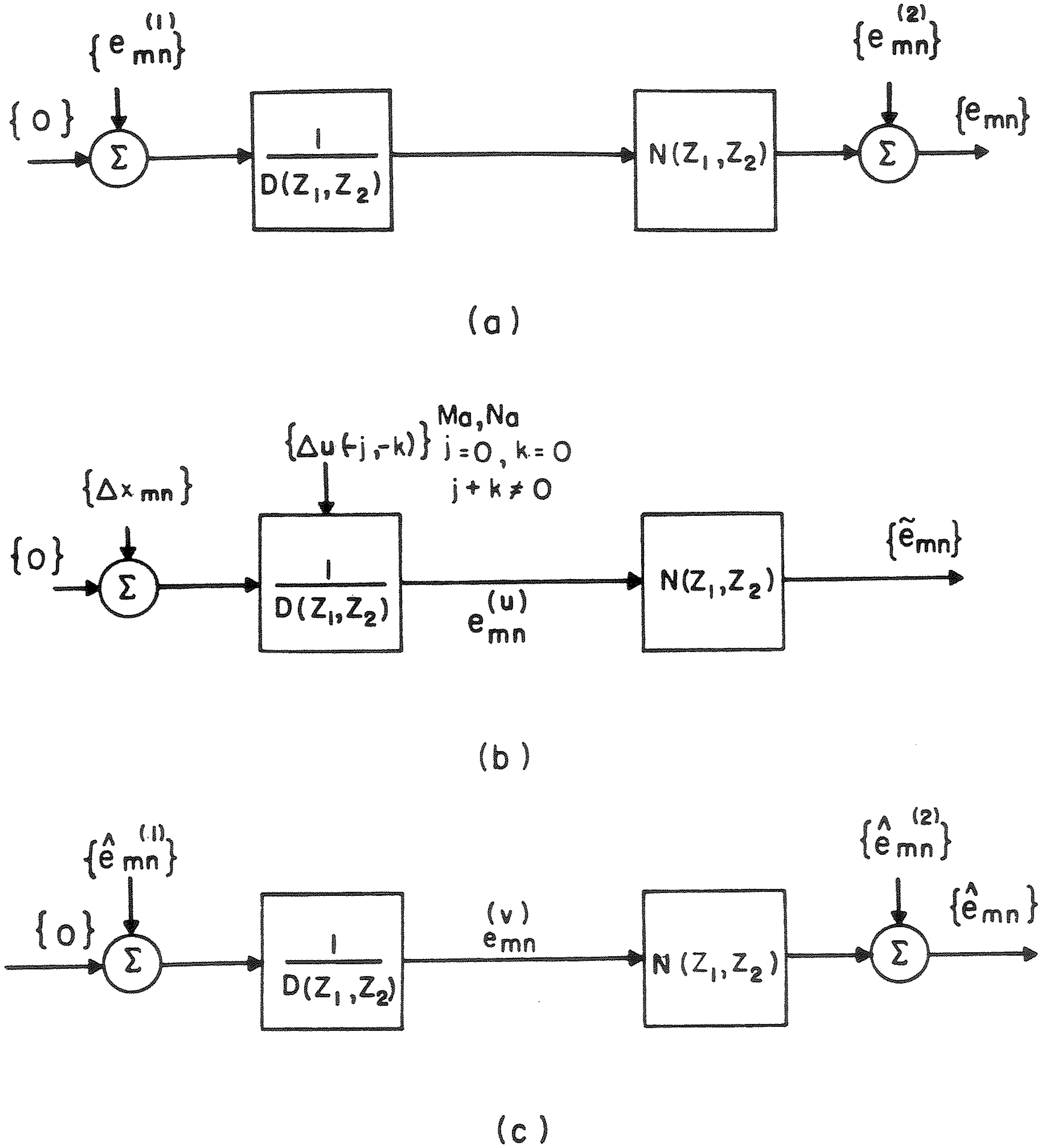

Fig. 6 Block Diagram Interpretations for Canonic Filtering Process

(a) Error $\{e_{mn}\}$ due to roundoff

(b) Error $\{\tilde{e}_{mn}\}$ due to quantization of inputs and initial values

(c) Error $\{\hat{e}_{mn}\}$ due to coefficient quantization

Input and Initial Value Quantization

An interpretation similar to the above is possible for input quantization and initial state quantization. Let the symbols with bars denote infinite precision quantities. For the direct filtering process, the use of the following notation makes the interpretation of noise sources in Figure 5(b) obvious.

(i) $$\Delta x_{mn} \triangleq \begin{cases} x_{mn} - \bar{x}_{mn}, & m \geq 0 \text{ and } n \geq 0, \\ 0, & \text{otherwise}, \end{cases}$$

(ii) $$\Delta w(j,k) \triangleq W(j,k) - \bar{W}(j,k), \quad |\Delta w(j,k)| \leq 2^{-t},$$

(iii) $$\tilde{e}_{mn} \triangleq \begin{cases} w_{mn} - \bar{w}_{mn}, & m \geq 0 \text{ and } n \geq 0, \\ \Delta w(m,n), & m = 0,-1,\ldots,-M_a, \\ & n = 0,-1,\ldots,-N_a,\ m+n \neq 0, \\ 0, & \text{for } m < -M_a \text{ or } n < -N_a. \end{cases}$$

For the canonic filtering process, the use of the following additional notation is needed for suitable interpretation of Figure 6(b).

(iv) $$\Delta u(j,k) \triangleq u(j,k) - \bar{V}(j,k), \quad |\Delta V(j,k)| \leq 2^{-t},$$

(v) $$e^{(u)}_{mn} \triangleq \begin{cases} u_{mn} - \bar{u}_{mn}, & m \geq 0,\ n \geq 0, \\ \Delta u(m,n), & \text{for } m = 0,-1,\ldots,-M_a, \\ & n = 0,-1,\ldots,-N_a,\ m+n \neq 0, \\ 0 & \text{for } m < -M_a, \text{ or } n < -N_a, \end{cases}$$

(vi) $$\tilde{e}_{mn} = w_{mn} - \bar{w}_{mn}, \quad m \geq 0,\ n \geq 0.$$

Coefficient Quantization

Finally, the coefficient quantization introduces output errors as shown in Figures 5(c) and 6(c) for direct and canonic filtering processes, respectively, with the following notation.

(vii) $\Delta b_{jk} \triangleq b_{jk} - \bar{b}_{jk}$,

$\Delta a_{jk} \triangleq a_{jk} - \bar{a}_{jk}$,

$|\Delta b_{jk}| \le 2^{-t}$, $|\Delta a_{jk}| \le 2^{-t}$,

(viii) $$\hat{e}^{(d)}_{mn} \doteq \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} \Delta b_{jk}\, x_{m-j,n-k} - \sum_{\substack{j=0\\ j+k \ne 0}}^{M_a} \sum_{k=0}^{N_a} \Delta a_{jk} w_{m-j,n-k},$$

(ix) $$\hat{e}^{(1)}_{mn} \doteq -\sum_{\substack{j=0\\ j+k \ne 0}}^{M_a} \sum_{k=0}^{N_a} \Delta a_{jk}\, v_{m-j,n-k},$$

$$\hat{e}^{(2)}_{mn} \doteq \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} \Delta b_{jk}\, v_{m-j,n-k},$$

(x) $e^{(v)}_{mn} = v_{mn} - \bar{v}_{mn}$, $m \ge 0$, $n \ge 0$,

$\hat{e}_{mn} = w_{mn} - \bar{w}_{mn}$, $m \ge 0$, $n \ge 0$.

In the later developments, output error bounds due to roundoff error accumulation are derived, mean-square-error analysis is presented, and dynamic range is also discussed. By following the same line of development, similar results for the effects of quantization errors for input and initial conditions, coefficients, and of the combinations of more than two sources of these errors (including roundoff errors) can be obtained without serious difficulty.

III.2.2 Mean Squared Error Analysis

Under certain circumstances, it is reasonable to model the effect of the rounding at each multiplication by the introduction of a white-noise source uniformly distributed with amplitude in the interval $(-E_o/2, E_o/2)$ (i.e. the mean is zero, and the variance is $E_o^2/12$) where E_o is the quantization level. Each of the noise sources is assumed to be independent of each other and of the input. Figure 7 shows how the quantization noise is introduced into the block diagram representation of a second order filter. The noise sources can be replaced by a single noise source as

$$e_{mn} = e_{mn,1} + e_{mn,2} + e_{mn,3}.$$

In general, from Eqs. (17), (20) and (21) the sources of roundoff errors can be represented by single sources $e^{(d)}_{mn}$, $e^{(1)}_{mn}$, and $e^{(2)}_{mn}$ with zero means and with variances as follows:

$$\sigma^2_{e^{(d)}} = \frac{E_o^2}{12} (\hat{M}_a \cdot \hat{N}_a - 1 + \hat{M}_b \cdot \hat{N}_b) , \tag{29}$$

$$\sigma^2_{e^{(1)}} = \frac{E_o^2}{12} (\hat{M}_a \cdot \hat{N}_a - 1) , \tag{30}$$

$$\sigma^2_{e^{(2)}} = \frac{E_o^2}{12} (\hat{M}_b \cdot \hat{N}_b) , \tag{31}$$

where $\hat{M}_b = M_b + 1$, $\hat{N}_b = N_b + 1$, $\hat{M}_a = M_a + 1$, $\hat{N}_a = N_a + 1$, and we have assumed that none of the coefficients, a's and b's, are zero or 1. If any of the coefficients are zero, the error is suitably reduced. We hold this assumption throughout the rest of this chapter.

The output f_{mn}, when the input consists of the noise samples r_{mn}, is

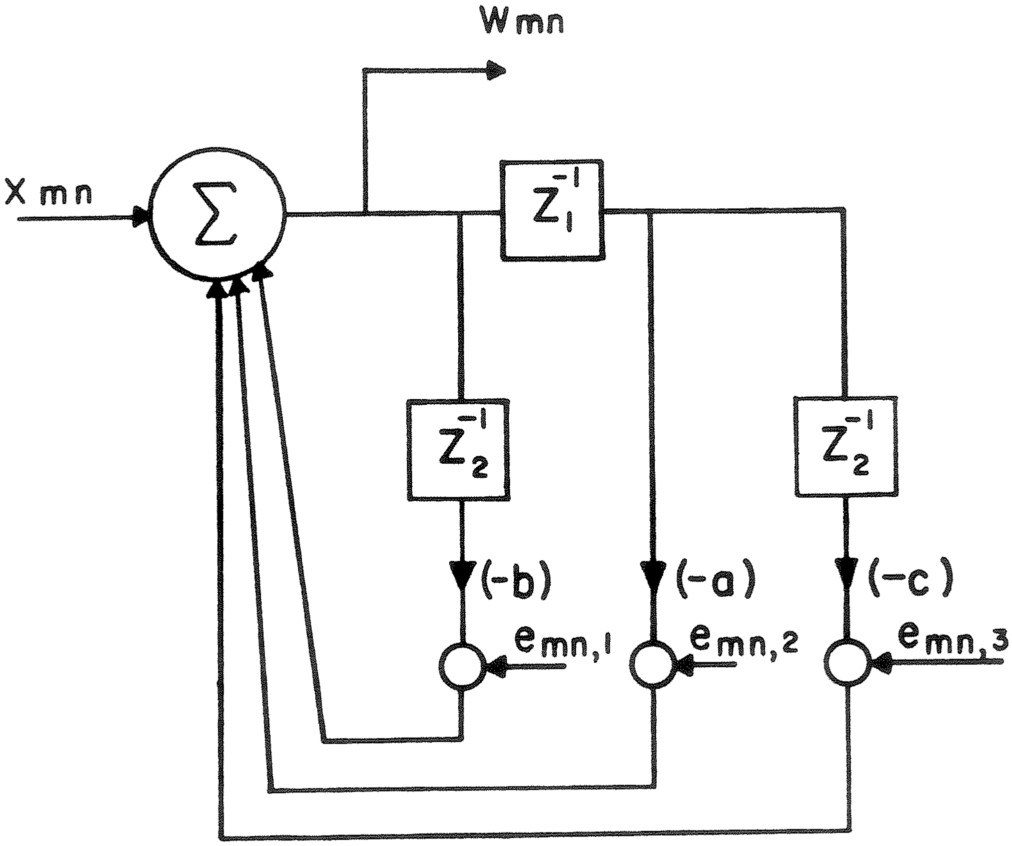

Fig. 7 $G(z_1, z_2) = \dfrac{1}{1 + az_1^{-1} + bz_2^{-1} + cz_1^{-1}z_2^{-1}}$

$$f_{mn} = \sum_{j=0}^{m} \sum_{k=0}^{n} h_{jk} \; r_{m-j,n-k} = \sum_{j=0}^{m} \sum_{k=0}^{n} h_{m-j,n-k} \; r_{jk} \tag{32}$$

where $\{h_{mn}\}$ is the point spread response. Thus, for the direct filtering process,

$$E\{f_{mn}^2\} = \frac{E_o^2}{12} (\hat{M}_a \cdot \hat{N}_a - 1 + \hat{M}_b \cdot \hat{N}_b) \sum_{j=0}^{m} \sum_{k=0}^{n} h_{jk}^2 \; . \tag{33}$$

For the steady state condition,

$$\sigma_t^2 = \frac{E_o^2}{12} (\hat{M}_a \cdot \hat{N}_a - 1 + \hat{M}_b \cdot \hat{N}_b) \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} h_{jk}^2 \; . \tag{34}$$

Generally, regardless of the configuration of the filters, if the impulse response from the (jk)th noise source to the output is $\{h_{mn,jk}\}_{m=0}^{\infty}{}_{,}{}_{n=0}^{\infty}$ then the steady state output-noise variance due to the (jk)th noise source is

$$\sigma_{jk}^2 = \frac{E_o^2}{12} \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} h_{mn,jk}^2 \tag{35}$$

and the total output noise is

$$\sigma_t^2 = \sum_{j} \sum_{k} \sigma_{jk}^2 \; . \tag{36}$$

III.2.3 Norm Error Bounds

In this section we first derive the absolute upper bounds for output errors. We also obtain the corresponding expectation bounds. From Eq. (17), it follows that

$$|e_{mn}^{(d)}| \le 2^{-t}(\hat{M}_b \cdot \hat{N}_b + \hat{M}_a \cdot \hat{N}_a - 1) \tag{37}$$

or

$$\langle e^{(d)} \rangle^{K_2, K_1} \le 2^{-t}(\hat{M}_b \cdot \hat{N}_b + \hat{M}_a \cdot \hat{N}_a - 1)\,. \tag{38}$$

Further, from Eq. (24) it follows that

$$e_{mn} = \sum_{j=0}^{m} \sum_{k=0}^{n} h_{m-j,n-k}\, e_{mn}^{(d)}\,. \tag{39}$$

Thus the bound for the total roundoff error at the output for the direct filtering process is

$$\langle e \rangle^{K_2,K_1} \le \max_{\substack{0 \le \omega_1 \le 2\pi \\ 0 \le \omega_2 \le 2\pi}} |D^{-1}(e^{i\omega_1}, e^{i\omega_2})|\, \langle e^{(d)} \rangle^{K_2,K_1}$$

$$\le 2^{-t}(\hat{M}_b \cdot \hat{N}_b + \hat{M}_a \cdot \hat{N}_a - 1) \cdot \max_{\substack{0 \le \omega_1 \le 2\pi \\ 0 \le \omega_2 \le 2\pi}} |D^{-1}(e^{i\omega_1}, e^{i\omega_2})| \tag{40}$$

by lemma 1 where $K_1, K_2 \ge 0$. For the canonic filtering process

$$|e_{mn}^{(1)}| \le 2^{-t}(\hat{M}_a \cdot \hat{N}_a - 1), \tag{41}$$

$$|e_{mn}^{(2)}| \le 2^{-t}(\hat{M}_b \cdot \hat{N}_b)\,, \tag{42}$$

or

$$\langle e^{(1)} \rangle^{K_2,K_1} \le 2^{-t}(\hat{M}_a \cdot \hat{N}_a - 1)\,, \tag{43}$$

$$\langle e^{(2)} \rangle^{K_2,K_1} \le 2^{-t}(\hat{M}_b \cdot \hat{N}_b)\,. \tag{44}$$

The total roundoff error $\{e_{mn}\}$ accumulated at the output is given by

$$e_{mn} = \sum_{j=0}^{m} \sum_{k=0}^{n} g_{m-j,n-k}\, e_{mn}^{(1)} + e_{mn}^{(2)} , \tag{45}$$

and by lemma 1

$$\langle e-e^{(2)} \rangle^{K_2,K_1} \le \max_{\substack{0 \le \omega_1 \le 2\pi \\ 0 \le \omega_2 \le 2\pi}} | G(e^{i\omega_1}, e^{i\omega_2}) | \langle e^{(1)} \rangle^{K_2,K_1} . \tag{46}$$

Inserting (43) and (44) into (46), one obtains

$$\langle e \rangle^{K_2,K_1} \le \max_{\substack{0 \le \omega_1 \le 2\pi \\ 0 \le \omega_2 \le 2\pi}} |G(e^{i\omega_1}, e^{i\omega_2})| \cdot 2^{-t}(\hat{M}_a \cdot \hat{N}_a - 1) + 2^{-t}(\hat{M}_b \cdot \hat{N}_b) . \tag{47}$$

Eqs. (40) and (47) are absolute upper bounds. If we keep the assumptions for the absolute errors $\delta_{mn,jk}$, $\eta_{mn,jk}, \ldots$, etc. in the beginning of Section III.2.2, we can easily obtain the following expectation bounds by the application of lemma 1 to Eqs. (39) and (45);

$$E\{^2\langle e \rangle^{K_2,K_1}\} \le \frac{E_0^2}{12} (\hat{M}_b \cdot \hat{N}_b + \hat{M}_a \cdot \hat{N}_a - 1) \cdot \max_{\substack{0 \le \omega_1 \le 2\pi \\ 0 \le \omega_2 \le 2\pi}} |D^{-1}(e^{i\omega_1}, e^{i\omega_2})|^2 \tag{48}$$

for the direct filtering process, and

$$E\{^2\langle e \rangle^{K_2,K_1}\} \le \frac{E_o^2}{12} [(\hat{M}_a \cdot \hat{N}_a - 1) \max_{\substack{0 \le \omega_1 \le 2\pi \\ 0 \le \omega_2 \le 2\pi}} |G(e^{i\omega_1}, e^{i\omega_2})|^2 + \hat{M}_b \cdot \hat{N}_b] \tag{49}$$

for the canonic filtering process.

III.3 Dynamic Range Considerations

Overflow occurs when the absolute value of the sum of more than one signal is greater than 1. For the two 2D fixed-point digital filtering processes discussed above inputs must be properly scaled so that overflow will not occur at node SN1 for the direct process shown in Figure 1, and at nodes SN2 and SN3 for the canonic process shown in Figure 2. The maximum value of input signal that will not cause overflow is found for each case as follows.

III.3.1 Direct Filtering Process

At SN1 in Figure 1,

$$w_{mn} = \sum_{j=0}^{m} \sum_{k=0}^{n} x_{m-j,n-k} g_{jk} ,$$

where

$$\sum_{j=0}^{\infty} \sum_{k=0}^{\infty} g_{jk} z_1^{-j} z_2^{-k} = \frac{N(z_1, z_2)}{D(z_1, z_2)} ,$$

and

$$|w_{mn}| \le \sum_{j=0}^{m} \sum_{k=0}^{n} |x_{m-j,n-k}| \, |g_{jk}|$$

$$\le \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} |x_{m-j,n-k}| \, |g_{jk}|$$

$$\le x_{maxd} \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} |g_{jk}|$$

where

$$|x_{jk}| \le x_{maxd}, \text{ for all } j \text{ and } k.$$

In order to guarantee $|w_{mn}| < 1$ for every $m \geq 0$ and $n \geq 0$, it suffices to have

$$x_{maxd} \leq \frac{1}{\sum_{j=0}^{\infty} \sum_{k=0}^{\infty} |g_{jk}|} \quad . \tag{50}$$

III.3.2 Canonic Filtering Process

At SN2 in Figure 2,

$$v_{mn} = \sum_{j=0}^{m} \sum_{k=0}^{n} x_{m-j,n-k} h_{jk}$$

where

$$\sum_{j=0}^{\infty} \sum_{k=0}^{\infty} h_{jk} z_1^{-j} z_2^{-k} = \frac{1}{D(z_1, z_2)}$$

so,

$$|v_{mn}| \leq x_{maxc} \cdot \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} |h_{jk}| \quad . \tag{51}$$

At SN3,

$$w_{mn} = \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk} v_{m-j,n-k}$$

so,

$$|w_{mn}| \leq v_{max} \cdot \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} |b_{jk}| .$$

Let

$$v_{max} \leq \frac{1}{\sum_{j=0}^{M_b} \sum_{k=0}^{N_b} |b_{jk}|} \quad , \tag{52}$$

then $|w_{mn}| < 1$ for every $m > 0$ and $n > 0$

where $|x_{jk}| < x_{maxc}$ and $|v_{jk}| < v_{max}$, for every $j \geq 0$ and $k \geq 0$.

From Eqs. (51) and (52), we have

$$|x_{mn}| < x_{max} \leq \frac{1}{\sum_{j=0}^{\infty}\sum_{k=0}^{\infty} |h_{jk}|} \times \frac{1}{\sum_{j=0}^{M_b}\sum_{k=0}^{N_b} |b_{jk}|}, \qquad (53)$$

for $m \geq 0$ and $n \geq 0$, and for no overflows at SN2 and SN3. It may be observed and emphasized that these are worst case bounds.

III.4 Numerical Results

We have a decimal simulation scheme to simulate the filtering operations. All the quantities involved are in double precision. The finite bit operations are simulated by having a decimal quantizer with quantization level $E_o = 2^{-(t-1)}$ follow each multiplication (or each input signal). When there is no quantizer, the results are taken as infinite precision ones. For the CDC 6600 computer, the mantissa of a number is represented by 48 bits. When double precision is used, the maximum possible error for the mantissa is $2^{-97} \approx 10^{-30}$. The following two examples are given to demonstrate the usefulness of the analytic results stated in Eqs. (35), (36), and (40) and (47). Table 1 and Table 2 show the computer results for the operations from m = 1 and n = 1 to m = 100 and n = 100. The impulse response in each example is reasonably small beyond m = 100 and n = 100.

Columns (I) and (III) list the "actual maximum errors" and "actual mean-square errors" calculated by the decimal simulation scheme on executing Eqs. (2), (9), and (10) respectively. Column (II) gives "theoretical upper bound of errors" by Eqs. (40) and (47). The actual error sequence mean square average norms are obtained by taking the square roots of corresponding values in Column (III). Column (IV) gives "theoretical mean-square errors" by Eqs. (35) and (36), respectively. Inputs in both cases are arbitrarily selected.

Example 1.

$$G(z_1, z_2) = \frac{1}{1 - 0.7z_1^{-1} - 0.5z_2^{-1} + 0.3z_1^{-1}z_2^{-1}},$$

(This filter is given in [2]),

input: $x_{mn} = 0.1 \cos(\omega t)$ where $\omega t = 0.01\, m(n-1)\pi$.

The point spread response dies out slowly. The real theoretical values should be greater than the ones listed in Column (IV).

Table 1

items / bits	(I)	(II)	(III)	(IV)
t = 8	3.38867×10^{-2}	1.17188×10^{-1}	8.73152×10^{-5}	4.54929×10^{-5}
t = 12	1.91280×10^{-3}	7.32422×10^{-3}	3.04815×10^{-7}	1.77707×10^{-7}
t = 16	1.24287×10^{-4}	4.57764×10^{-4}	1.08707×10^{-9}	6.94167×10^{-10}
t = 20	8.71887×10^{-6}	2.86102×10^{-5}	4.59030×10^{-12}	2.71159×10^{-12}
t = 24	4.94035×10^{-7}	1.78814×10^{-6}	1.82150×10^{-14}	1.05921×10^{-14}
t = 28	3.09238×10^{-8}	1.11759×10^{-6}	6.54997×10^{-17}	4.13756×10^{-17}
t = 32	2.05030×10^{-9}	6.98492×10^{-9}	2.75747×10^{-19}	1.61623×10^{-19}

Example 2.

$$G(z_1, z_2) = \frac{N(z_1, z_2)}{D(z_1, z_2)}$$

where

$$N(z_1,z_2) = 1-0.474999z_2^{-1} - 0.636396z_1^{-1}+0.302287z_1^{-1}z_2^{-1},$$

$$\begin{aligned} D(z_1,z_2) = {} & 1-0.949998z_2^{-1} + 0.9025z_2^{-2} + 1.27279z_1^{-1} \\ & -1.20915z_1^{-1}z_2^{-1} + 1.14869z_1^{-1}z_2^{-2} + 0.81z_1^{-2} \\ & -0.769498z_1^{-2}z_2^{-1} + 0.731025z_1^{-2}z_2^{-2}, \end{aligned}$$

input: $x_{mn} = 0.025 \cos \omega t$ and

$$\omega t = 0.1\pi \times [(m-1) \times 100 + n] + 0.01n .$$

The value of maximum gain of this filter is approximately 60.

Table 2

items / bits	(I)		(II)		(III)		(IV)	
	Direct Process	Canonic Process	Direct Process	Canonic Process	Direct Process	Canonic Process	Direct Process	Canonic Process
t = 8	2.19017×10^{-1}	2.05563×10^{-1}	3.78207	1.70522	3.02690×10^{-3}	2.70298×10^{-3}	2.31123×10^{-3}	2.91967×10^{-3}
t = 12	1.04947×10^{-2}	1.34391×10^{-2}	2.36280×10^{-1}	1.06576×10^{-1}	6.14841×10^{-6}	1.06663×10^{-5}	9.02825×10^{-6}	1.14050×10^{-5}
t = 16	6.96397×10^{-4}	7.42209×10^{-4}	1.47737×10^{-2}	6.66101×10^{-3}	2.86827×10^{-8}	4.37726×10^{-8}	3.52666×10^{-8}	4.45507×10^{-8}
t = 20	4.64883×10^{-5}	4.49094×10^{-5}	9.23358×10^{-4}	4.16313×10^{-4}	1.09832×10^{-10}	1.42745×10^{-10}	1.37760×10^{-10}	1.74026×10^{-10}
t = 24	2.64667×10^{-6}	3.04092×10^{-6}	5.77099×10^{-5}	2.60196×10^{-5}	4.47297×10^{-13}	5.33565×10^{-13}	5.38126×10^{-13}	6.79790×10^{-13}
t = 28	1.93636×10^{-7}	2.04656×10^{-7}	3.60687×10^{-6}	1.62622×10^{-6}	2.36728×10^{-15}	2.55708×10^{-15}	2.10205×10^{-15}	2.65543×10^{-15}
t = 32	1.04461×10^{-8}	1.06558×10^{-8}	2.25429×10^{-7}	1.01639×10^{-7}	6.99403×10^{-18}	7.58707×10^{-18}	8.21114×10^{-18}	1.03728×10^{-17}

IV. ERROR ANALYSIS FOR TWO-DIMENSIONAL DIGITAL FILTERS EMPLOYING FLOATING-POINT ARITHMETIC

In this chapter, we analyze the effects of finite word length for two-dimensional digital filters employing floating-point arithmetic. Our procedures are basically the same as in Chapter III. We present a systematic way of estimating (1) the output mean squared errors and (2) the output norm error bounds for all three sources of error. Extensive numerical experiments show that the method leads to satisfactory results. We concentrate our efforts on the direct filtering process of Eq. (2). We begin by deriving the actual system equations with finite word length.

IV.1 Actual Systems of Equations

<u>Basic Assumptions for Digital Filters</u>

(1) each machine number q is equal to "sgn (q) $\cdot$ a $\cdot$ 2^b where the exponent "b" is an integer and "a", the mantissa, is represented by a t bit number. "a" takes on values in $[\frac{1}{2}, 1)$ or $\{0\}$. The following error properties are true under this assumption;

$$fl\ [x] = x(1+\epsilon), \quad |\epsilon| < 2^{-t}, \tag{54}$$

$$fl\ [x+y] = (x+y)(1+\rho), \quad |\rho| < 2^{-t}, \tag{55}$$

$$fl\ [xy] = xy\ (1+\delta), \quad |\delta| < 2^{-t}, \tag{56}$$

where x and y are infinite precision numbers, ϵ, ρ, and δ are relative errors, and fl $[\cdot]$ denotes the floating-point operation with t-bit mantissa,

(2) the range of the values of b is adequate enough to ensure that all the computed numbers lie within the permissible range,

(3) the digital filter of Eq. (2) is stable ([1],[2]),

(4) the ordering of the arithmetic operations is according to the flow diagram shown in Figure 8, where $\delta_{mn,ij}$, $\epsilon_{mn,ij}$, $r_{mn,ij}$, ξ_{mn},

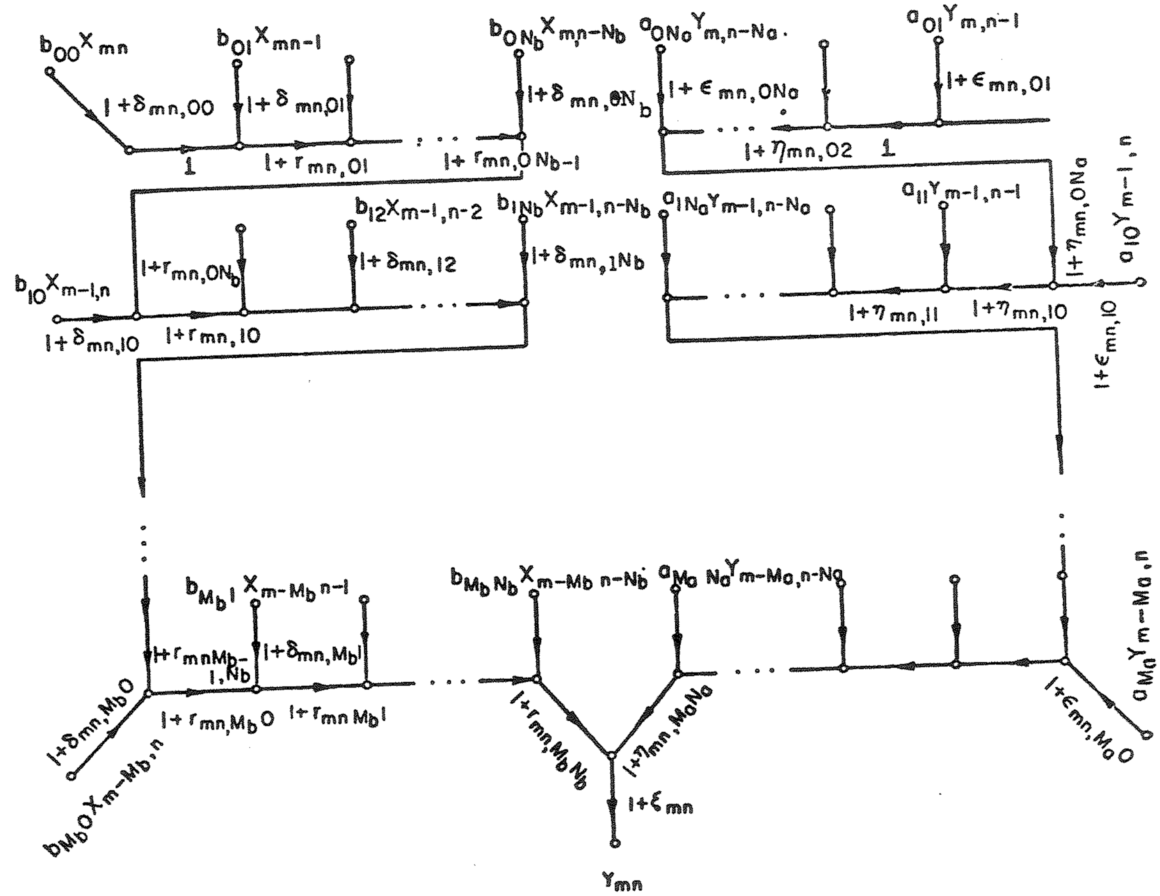

Fig. 8 Flow Graph for Floating-point Direct Filtering Process

$\eta_{mn,ij}$ are relative errors and take on values in $(-2^{-t}, 2^{-t})$.

Actual System of Equations with Roundoff Errors in Multiplications and Additions

The actual system equations for additive and multiplicative round-off errors are

$$y_{mn} = fl \left[\sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk}\, x_{m-j,n-k} - \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk} y_{m-j,n-k} \right] \tag{57}$$

$$= \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk}\, x_{m-j,n-k} - \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk}\, y_{m-j,n-k} + e_{mn}^{(d)} , \tag{58}$$

where y_{mn}, x_{mn}, b_{jk}, and a_{jk} are machine numbers,

$$e_{mn}^{(d)} = \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk}\, (\theta_{mn,jk} - 1)\, x_{m-j,n-k}$$

$$- \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk}\, (\varphi_{mn,jk} - 1)\, y_{m-j,n-k} , \tag{59}$$

$$e_{i\ell}^{(d)} = 0 \quad \text{for } i < 0 \text{ or } \ell < 0 ,$$

$$\theta_{mn,00} = (1 + \xi_{mn})(1 + \delta_{mn,00}) \prod_{\substack{i=0,\ell=0 \\ i+\ell \neq 0}}^{M_b,N_b} (1 + \gamma_{mn,i\ell}) , \tag{60}$$

$$\theta_{mn,jk} = (1+\xi_{mn})(1+\delta_{mn,jk}) \prod_{i=j,\ell=k}^{M_b,N_b} (1+\gamma_{mn,i\ell}) , \quad j = 0,\ldots,M_b, \quad k = 0,\ldots,N_b, \quad j+k \neq 0 , \tag{61}$$

$$\varphi_{mn,01} = (1+\xi_{mn})(1+\epsilon_{mn,01})\left(\prod_{\ell=2}^{N_a}(1+\eta_{mn,0\ell})\right)\left(\prod_{i=1,\ell=0}^{M_a,N_a}(1+\eta_{mn,i\ell})\right), \tag{62}$$

$$\varphi_{mn,0k} = (1+\xi_{mn})(1+\epsilon_{mn,0k})\left(\prod_{\ell=k}^{N_a}(1+\eta_{mn,0k}\right)\left(\prod_{i=1,\ell=0}^{M_a,N_a}(1+\eta_{mn,i\ell})\right) , \quad k = 2,\ldots,N_a \tag{63}$$

and

$$\varphi_{mn,jk} = (1+\xi_{mn})(1+\epsilon_{mn,jk})\left(\prod_{i=j,\ell=k}^{M_a,N_a}(1+\eta_{mn,i\ell})\right), \quad k = 0,\ldots,N_a, \quad j = 1,\ldots,M_a . \tag{64}$$

For Eqs. (60) to (64), we have assumed that $b_{jk} \neq 0$ or 1 for $0 \leq j \leq M_b$ and $0 \leq k \leq N_b$, and that $a_{jk} \neq 0$ or 1 for $0 \leq j \leq M_a$ and $0 \leq k \leq N_a$ except $a_{00} = 1$. If any of the b_{jk} or a_{jk} is 0 or 1, the multiplicative and additive roundoff errors associated with it are zero. The numbers of factors in the expressions for $\theta_{mn,jk}$ and $\varphi_{mn,jk}$ are respectively as follows,

$$\beta'_{jk} = \begin{cases} (M_b+1)(N_b+1)+1, & \text{for } j=0,\ k=0 , \quad (65) \\ (M_b-j+1)(N_b-k+1)+2, & \text{otherwise}, \quad (66) \end{cases}$$

$$\alpha'_{jk} = \begin{cases} (M_a+1)(N_a+1), & \text{for } j = 0,\ k = 1, & (67) \\ (M_a+1)(N_a+1) - k+2, & \text{for } j = 0,\ k = 2, \ldots, N_a, & (68) \\ (M_a-j+1)(N_a-j+1)+2, & \text{otherwise}. & (69) \end{cases}$$

Notice that if some of the filter coefficients are 0 (no addition and multiplication) or 1 (no multiplication), the numbers in Eqs. (65) to (69) will be correspondingly reduced.

Since the digital filter is stable, if the number of bits t is not too small we can approximate Eq. (59) by substituting $w_{m-j,n-k}$ for $y_{m-j,n-k}$ (the mathematical justification is similar to the 1D case in [8]) and obtain

$$e^{(d)}_{mn} = \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk} (\theta_{mn,jk} - 1) x_{m-j,n-k} - \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk} (\varphi_{mn,jk} - 1) w_{m-j,n-k} , \tag{70}$$

where w_{mn} is the ideal output.

Actual System Equations for Coefficient Quantizations

For coefficient quantization the actual system equations are

$$\hat{y}_{mn} = \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk} x_{m-j,n-k} - \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk} \hat{y}_{m-j,n-k} + \hat{e}^{(d)}_{mn} , \tag{71}$$

where $\hat{y}_{mn}$, b_{jk}, a_{jk}, and x_{mn} are machine numbers, and

$$\hat{e}^{(d)}_{mn} = \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} \epsilon_{jk} \bar{b}_{jk} x_{m-j,n-k} - \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} \delta_{jk} \bar{a}_{jk} \hat{y}_{m-j,n-k}, \qquad (72)$$

where $\bar{b}_{jk}$ and $\bar{a}_{jk}$ are ideal filter coefficients,

$| \epsilon_{jk} | < 2^{-t}$, and

$| \delta_{jk} | < 2^{-t}$.

Similar to Eq. (70),we can also approximate Eq. (72) by

$$\hat{e}^{(d)}_{mn} = \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} \epsilon_{jk} \bar{b}_{jk} x_{m-j,n-k} - \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} \delta_{jk} \bar{a}_{jk} w_{m-j,n-k}, \qquad (73)$$

where w_{mn} is the ideal output.

<u>Actual System Equations for Quantizations of Inputs and Initial Conditions</u>

For quantized inputs and initial conditions the actual system equations are

$$\bar{y}_{mn} = \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk} x_{m-j,n-k} - \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk} \bar{y}_{m-j,n-k} + \tilde{e}^{(d)}_{mn}, \qquad (74)$$

where $\bar{y}_{mn}$, b_{jk}, a_{jk}, and x_{mn} are machine numbers,

$$\bar{y}_{jk} = (1 + \epsilon_{jk})\, W(-j,-k)\,, \qquad j = 0, \ldots, M_a\,, \quad k = 0, \ldots, N_a\,, \quad j + k \neq 0\,, \tag{75}$$

where $|\epsilon_{jk}| < 2^{-t}$, and $W(-j,-k)$ are the ideal initial conditions,

and

$$\tilde{e}^{(d)}_{mn} = -\sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk}\, (\delta_{m-j,n-k}\, \bar{x}_{m-j,n-k})\,, \tag{76}$$

where $\bar{x}_{mn}$ is an ideal input, and

$$|\delta_{mn}| < 2^{-t} \quad .$$

IV.2 Behavior of the Sources of Error

Let

$$e_{mn} = y_{mn} - w_{mn}. \tag{77}$$

Subtracting Eq. (2) from Eq. (58), we have

$$e_{mn} = -\sum_{\substack{j=0 \\ j+k\neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk}\, e_{m-j,n-k} + e_{mn}^{(d)} \quad . \tag{78}$$

Eq. (78) says that the output roundoff errors $\{e_{mn}\}$ can be interpreted as resulting from the injection of an error source $e_{mn}^{(d)}$ into the system at the junction indicated in Figure 5(a).

Similarly, from Eqs. (2), (71), and (70), we can interpret the output errors, due to the coefficient inaccuracies or the quantizations of inputs and initial states, as resulting from error sources $\hat{e}_{mn}^{(d)}$ or $\tilde{e}_{mn}^{(d)}$ injected into the system at the junctions shown in Figures 5(b) and 5(c).

IV.3 Mean Squared Error Analysis

Basic Assumptions

As usual, we assume that the relative errors $\delta_{mn,ij}$, $\epsilon_{mn,ij}$, $\gamma_{mn,ij}$, $\xi_{mn,ij}$, etc. are independent random variables and are uniformly distributed in the range $(-2^{-t}, 2^{-t})$. Thus they are zero mean and have a variance of $\sigma^2 = E_o^{\,2}/3$ with $E_o = 2^{-t}$, and they are uncorrelated with each other or with any other signals. We also assume that the input signal x_{mn} is zero mean and wide sense stationary. Since our system is a linear spatially invariant system the output signal is also zero mean and wide sense stationary.

Mean Squared Error Estimations for Roundoff Errors

Expanding Eq. (63), we have

$$\varphi_{mn,jk} = 1 + \xi_{mn} + \epsilon_{mn,jk} + \sum_{i}\sum_{\ell} \eta_{mn,i\ell} + 0\,(2^{-2t}). \tag{79}$$

Take the first order approximation,

$$\varphi_{mn,jk} - 1 \doteq \xi_{mn} + \epsilon_{mn,jk} + \sum_{i}\sum_{\ell} \eta_{mn,i\ell} \quad . \tag{80}$$

The right hand side of Eq. (80) has a total of α'_{jk} (see Eq. (69)) independent identically distributed random variables. We then consider $(\varphi_{mn,jk}-1)$ to be an independent random variable consisting of the sum of α'_{jk} independent identically distributed random variables. Likewise we consider each $(\theta_{mn,jk}-1)$ to be an independent random variable which is the sum of β'_{jk} independent identically distributed random variables.

In Eq. (70), let

$$e^{(a)}_{mn,jk} = (\varphi_{mn,jk}-1)\, a_{jk}\, w_{m-j,n-k}\,, \tag{81}$$

$$e^{(b)}_{mn,jk} = (\theta_{mn,jk}-1)\, b_{jk}\, x_{m-j,n-k}. \tag{82}$$

The means of the random variables $e^{(a)}_{mn,jk}$ and $e^{(b)}_{mn,jk}$ are zero. Let $\sigma^2_{e_{jk}}(a)$ and $\sigma^2_{e_{jk}}(b)$ represent their variances respectively; then,

$$\sigma^2_{e_{jk}}(a) = \frac{E_0^2}{3} \cdot \alpha'_{jk} \cdot \overline{(a_{jk} w_{m-j,n-k})^2}\,, \tag{83}$$

$$\sigma^2_{e_{jk}}(b) = \frac{E_0^2}{3} \cdot \beta'_{jk} \cdot \overline{(b_{jk} x_{m-j,n-k})^2}\,. \tag{84}$$

The bar denotes expected value. We approximate it by the following operation,

$$\overline{A^2} = \lim_{\substack{n\to\infty \\ m\to\infty}} \frac{\sum_{i=0}^{m}\sum_{j=0}^{n} A_{ij}^2}{mn} . \tag{85}$$

Considering $e_{mn}^{(d)}$ (Eqs. (58), (59), (60), and (78)) to be composed of $(M_b+1)(N_b+1)$ of the $e_{mn,jk}^{(b)}$'s and $[(M_a+1)(N_a+1)-1]$ of the $e_{mn,jk}^{(a)}$'s, then from Eqs. (83) and (84), the steady state output error is found to be

$$\sigma_e^2 = \Big(\sum_{\substack{j=0 \\ j+k\neq 0}}^{M_a} \sum_{k=0}^{N_a} \sigma_{e_{jk}}^2(a) + \sum_{j=0}^{M_b}\sum_{k=0}^{N_b} \sigma_{e_{jk}}^2(b) \Big) \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} h_{mn}^2 . \tag{86}$$

Mean Squared Error Estimates for Coefficient Inaccuracies

From Eqs. (71) and (73), the output error due to the quantizations of the filter coefficients can be calculated as

$$\hat{\sigma}_e^2 = \Big(\sum_{\substack{j=0 \\ j+k\neq 0}}^{M_a} \sum_{k=0}^{N_a} \hat{\sigma}_{e_{jk}}^2(a) + \sum_{j=0}^{M_b}\sum_{k=0}^{N_b} \hat{\sigma}_{e_{jk}}^2(b) \Big) \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} h_{mn}^2 , \tag{87}$$

with

$$\hat{\sigma}_{e_{jk}}^2(a) = \overline{(a_{jk}\delta_{jk}w_{m-j,n-k})^2} = \delta_{jk}^2\, \overline{(a_{jk}w_{m-j,n-k})^2} , \tag{88}$$

$$\hat{\sigma}^2_{e_{jk}}(b) = \overline{(b_{jk}\epsilon_{jk}x_{m-j,n-k})^2} = \epsilon^2_{jk} \ \overline{(b_{jk}x_{m-j,n-k})^2} \tag{89}$$

where δ_{jk} and ϵ_{jk} are fixed numbers. For high order filters we may be able to regard δ_{jk} and ϵ_{jk} as random variables, and hence we can replace δ^2_{jk} or ϵ^2_{jk} by $E^2_o/3$ in Eqs. (88) and (89).

Mean Squared Error Estimates for Quantizations of Input Signals and Initial Conditions

The output error due to the quantization of the input signal can be calculated as

$$\bar{\sigma}^2_e = \left(\frac{E^2_o}{3} \ \overline{x^2_{mn}} \right) \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} g^2_{mn} \ . \tag{90}$$

As for the output error due to the quantization of the initial conditions, it usually can be neglected if the impulse response dies out fast enough. The output error is governed by the following equation,

$$\bar{\bar{e}}_{mn} = - \sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} h^{(jk)}_{mn} \ \epsilon_{jk} \ W(-j,-k) , \tag{91}$$

where ϵ_{jk} and $W(-j,-k)$ are defined in Eq. (75), and $h^{(jk)}_{mn}$ is given in Eq. (12).

The derivation of Eq. (91) is not difficult and is omitted here. Note that $h^{(jk)}_{mn} \underset{m,n}{\rightarrow} 0$ and hence $\bar{\bar{e}}_{mn} \rightarrow 0$ when m and/or n $\rightarrow \infty$, since the digital filter is stable. It can be seen that

$$\langle \bar{\bar{e}} \rangle^{K_2,K_1} \le \sum_{j=0}^{M_a} \sum_{k=0}^{N_a} \langle h^{(jk)} \rangle^{K_2,K_1} \, |\epsilon_{jk} W(-j,-k)| \, . \qquad (92)$$

The value of the right hand side dies out as K_2 and K_1 increase.

Discussion

We have derived the steady state output mean squared error for each of the three sources of error. If two or more sources of error are present, the output mean squared errors due to their combined effects can be approximated by the summations of σ_e^2(Eq. (86)), $\hat{\sigma}_e^2$(Eq. (87)), and/or $\bar{\sigma}_e^2$ (Eq. (90)).

For nonstationary or deterministic inputs we cannot obtain similar results to those in Eqs. (86), (89), and (90). As an illustration, for round-off error with deterministic inputs, we have the following equation instead of Eq. (86),

$$\sigma_{e_{mn}}^2 = \frac{E_o^2}{3} \Bigg[\sum_{\substack{j=0 \\ j+k \neq 0}}^{M_a} \sum_{k=0}^{N_a} a_{jk}^2 \cdot \alpha'_{jk} \cdot \Big(\sum_{\mu=0}^{m} \sum_{\nu=0}^{n} w_{\mu-j,\nu-k}^2 \, h_{m-\mu,n-\nu}^2 \Big)$$

$$+ \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk}^2 \cdot \beta'_{jk} \cdot \Big(\sum_{\mu=0}^{m} \sum_{\nu=0}^{n} x_{\mu-j,\nu-k}^2 \, h_{m-\mu,n-\nu}^2 \Big) \Bigg] \, . \qquad (93)$$

IV.4 Norm Bounds

In this section we obtain norm bounds for output errors due to the three sources of error. We still assume that the relative errors $\gamma_{mn,jk}$, $\delta_{mn,jk}$,..., etc. are independent random variables uniformly distributed in $(-2^{-t}, 2^{-t})$. We make no assumption for the input signal.

For roundoff errors, letting $e^{01}_{mn,jk}$ be the output error due to the error source $e^{(a)}_{mn,jk}$,

$$e^{01}_{mn,jk} = \sum_{\mu=0}^{m} \sum_{\nu=0}^{n} (\varphi_{\mu\nu,jk}-1)a_{jk}w_{\mu-j,\nu-k} \cdot h_{m-\mu,n-\nu}. \tag{94}$$

By lemma 1 we can obtain

$$E\{^2\langle e^{01}_{jk}\rangle^{K_2,K_1}\} \le \max_{\substack{0\le\omega_1\le 2\pi \\ 0\le\omega_2\le 2\pi}} |D^{-1}(e^{i\omega_1},e^{i\omega_2})|^2 \cdot \frac{E_o^2}{3} a^2_{jk}\,\alpha'_{jk} \cdot {}^2\langle w\rangle^{K_2,K_1}_{j,k} \tag{95}$$

where $E\{\cdot\}$ denotes the expected value, and we have assumed that the initial conditions of the 2D digital filters are zero. Therefore,

$$E\{^2\langle e\rangle^{K_2,K_1}\} \le \max_{\substack{0\le\omega_1\le 2\pi \\ 0\le\omega_2\le 2\pi}} |D^{-1}(e^{i\omega_1},e^{i\omega_2})|^2 \cdot \frac{E_o^2}{3} \cdot \Big[\sum_{\substack{j=0 \\ j+k\neq 0}}^{M_a} \sum_{k=0}^{N_a} a^2_{jk}\,\alpha'_{jk} \cdot {}^2\langle w\rangle^{K_2,K_1}_{j,k} + \sum_{j=0}^{M_b}\sum_{k=0}^{N_b} b^2_{jk}\,\beta'_{jk} \cdot {}^2\langle x\rangle^{K_2,K_1}_{j,k}\Big]. \tag{96}$$

Similarly, for quantization errors of the filter coefficients, we have

$$E\{{}^{2}\langle\hat{e}\rangle^{K_2,K_1}\} \le \max_{\substack{0\le\omega_1\le 2\pi \\ 0\le\omega_2\le 2\pi}} |D^{-1}(e^{i\omega_1}, e^{i\omega_2})|^2 \cdot [\sum_{\substack{j=0 \\ j+k\neq 0}}^{M_a} \sum_{k=0}^{N_a} \bar{a}_{jk}^{\,2}\, \delta_{jk}^{2} \cdot {}^{2}\langle w\rangle_{j,k}^{K_2,K_1}$$

$$+ \sum_{j=0}^{M_b} \sum_{k=0}^{N_b} \bar{b}_{jk}^{\,2}\, \epsilon_{jk}^{2} \cdot {}^{2}\langle x\rangle_{j,k}^{K_2,K_1}]. \quad (97)$$

For quantization errors of the input signal,

$$E\{{}^{2}\langle\bar{e}\rangle^{K_2,K_1}\} \le \max_{\substack{0\le\omega_1\le 2\pi \\ 0\le\omega_2\le 2\pi}} |G(e^{i\omega_1}, e^{i\omega_2})|^2 \cdot \frac{E_o^2}{3}$$

$$\cdot [\sum_{j=0}^{M_b} \sum_{k=0}^{N_b} b_{jk}^{2}\, \beta'_{jk} \cdot {}^{2}\langle x\rangle_{j,k}^{K_2,K_1}]. \quad (98)$$

IV.5 Numerical Results

A "Decimal simulation program" has been written to simulate, decimally, all the filtering operations on a CDC 6600 computer. The program includes subroutines which simulate floating-point additions, floating-point multiplications, and quantizations. All the floating-point variables are double precision.

The following examples are among those we used to test the validity of the method in Eqs. (86), (87), (90), (96), (97), and (98). For each example with a given input signal the following quantities are calculated; let $\Delta = {}^{2}\langle w \rangle^{K_2,K_1}$ where w_{mn} is the ideal output sequence,

(i) RE : σ_e^2/Δ, where σ_e^2 is the theoretical roundoff error (Eq. (86),

(ii) RA : ${}^{2}\langle e \rangle^{K_2,K_1}/\Delta$, where e_{mn} is the actual output error due to roundoffs,

(iii) RNE : Norm bound for output error due to roundoffs (Eq. (96)) divided by Δ,

(iv) CE : $\hat{\sigma}_e^2$ (Eq. (87)/Δ,

(v) CA : ${}^{2}\langle \hat{e} \rangle^{K_2,K_1}/\Delta$, where $\hat{e}_{mn}$ is the actual output error due to coefficient quantizations,

(vi) CNE : Norm bounds for output error due to coefficient quantizations (Eq. (97)) divided by Δ,

(vii) IE : $\bar{\sigma}_e^2$ (Eq. (90))/Δ,

(viii) IA : ${}^{2}\langle \bar{e} \rangle^{K_2,K_1}/\Delta$, where $\bar{e}_{mn}$ is the actual output error due to input quantizations,

(ix) INE : Norm bound for output error due to input quantizations (Eq. (98)) divided by Δ,

(x) RCE = RE + CE,

(xi) RCA : $^2\langle e^{rc}\rangle^{K_2,K_1}/\Delta$, where e^{rc}_{mn} is the actual output error due to roundoffs and coefficient quantizations,

(xii) RCNE = RNE + CNE,

(xiii) RCIE = RC + CE + IE,

(xiv) RCIA : $^2\langle e^{rci}\rangle^{K_2,K_1}/\Delta$, where e^{rci}_{mn} is the actual output error due to the combined effects of all the three sources of error,

(xv) RCINE = RNE + CNE + INE.

Example 1: Two-dimensional Digital Filter Designed by Shanks' Rotation Method and Modified by a Planar Least Squares Inverse Algorithm ([3]).

$$
\begin{aligned}
N(z_1,z_2) = {} & 4.39\mathrm{x}10^{-3} + 1.317\mathrm{x}10^{-2} z_1^{-1} + 1.317\mathrm{x}10^{-2} z_1^{-2} + 4.39\mathrm{x}10^{-3} z_1^{-3} \\
& + 1.317\mathrm{x}10^{-2} z_2^{-1} + 3.9509\mathrm{x}10^{-2} z_1^{-1} z_2^{-1} + 3.9509\mathrm{x}10^{-2} z_1^{-2} z_2^{-1} \\
& \quad + 1.317\mathrm{x}10^{-2} z_1^{-3} z_2^{-1} \\
& + 1.317\mathrm{x}10^{-2} z_2^{-2} + 3.9509\mathrm{x}10^{-2} z_1^{-1} z_2^{-2} + 3.9509\mathrm{x}10^{-2} z_1^{-2} z_2^{-2} \\
& \quad + 1.317\mathrm{x}10^{-2} z_1^{-3} z_2^{-2} \\
& + 4.39\mathrm{x}10^{-3} z_2^{-3} + 1.317\mathrm{x}10^{-2} z_1^{-1} z_2^{-3} + 1.317\mathrm{x}10^{-2} z_1^{-2} z_2^{-3} \\
& \quad + 4.39\mathrm{x}10^{-3} z_1^{-3} z_2^{-3} \quad .
\end{aligned}
$$

$$
\begin{aligned}
D(z_1, z_2) = {} & 1.+9.72193\times10^{-2} z_1^{-1} + 1.35846\times10^{-2} z_1^{-2} + 6.245431\times10^{-4} z_1^{-3} \\
& + 2.105095\times10^{-2} z_2^{-1} - 1.785671 z_1^{-1} z_2^{-1} - 0.1398205 z_1^{-2} z_2^{-1} \\
& - 1.197869\times10^{-2} z_1^{-3} z_2^{-1} \\
& + 7.184771\times10^{-3} z_2^{-2} - 8.055841\times10^{-2} z_1^{-1} z_2^{-2} + 1.262808 z_1^{-2} z_2^{-2} \\
& + 5.927248\times10^{-2} z_1^{-3} z_2^{-2} \\
& - 7.236237\times10^{-4} z_2^{-3} - 5.415724\times10^{-3} z_1^{-1} z_2^{-3} \\
& + 5.705888\times10^{-2} z_1^{-2} z_2^{-3} - 3.541561\times10^{-1} z_1^{-3} z_2^{-3}.
\end{aligned}
$$

The following three input signals are used:

2I1 : $x_{jk} = \cos(j)\cos(k)$, $j = 0, \ldots, 127$, and $k = 0, \ldots, 127$, (99)

2I2 : $x_{jk} = \cos(j \cdot k)$, $j = 0, \ldots, 127$, and $k = 0, \ldots, 127$, (100)

2I3 : $x_{jk} = \text{noise}(j,k)$, $j = 0, \ldots, 127$, and $k = 0, \ldots, 127$, (101)

where noise (j,k) are generated by a random number generator.

The operations are from $m = 0, \ldots, 127$ and $n = 0, \ldots, 127$. The impulse responses for $1/D(z_1, z_2)$ at $(m,n) = (0,0)$ and $(m,n) = (127,127)$ are 1.0 and 5.732×10^{-16}, respectively. The number of bits for simulations is $t = 16$. Table 3 shows the numerical results.

Table 3

Errors	Inputs 2I1	2I2	2I3	Powers of 10.
RE	2.14062	2.14409	2.15345	10^{-8}
RA	2.76639	1.75509	1.55308	10^{-8}
RNE	1.47309	1.47548	1.48192	10^{-7}
CE	1.18103	1.19667	1.19465	10^{-9}
CA	2.18830	1.63675	1.57992	10^{-9}
CNE	8.12738	8.23503	8.22112	10^{-9}
IE	2.73986	8.64918	7.066	10^{-11}
IA	1.28770	3.15256	3.55109	10^{-11}
INE	0.681206	2.15043	1.75680	10^{-9}
RCE	2.25872	2.26376	2.27292	10^{-8}
RCA	2.02213	1.57048	1.40364	10^{-8}
RCNE	1.55437	1.55783	1.56413	10^{-9}
RCIE	2.26146	2.27241	2.27998	10^{-8}
RCIA	2.08492	1.60492	1.41122	10^{-8}
RCINE	1.56118	1.57934	1.58170	10^{-7}

Example 2: Two-Dimensional High Pass Digital Filter.

$$N(z_1, z_2) = 1.2-3.6z_1^{-1} + 3.6z_1^{-2} - 1.2z_1^{-3} - 3.6z_2^{-1} + 10.8z_1^{-1} z_2^{-1}$$

$$- 10.08z_1^{-2} z_2^{-1} + 3.6z_1^{-3} z_2^{-1} + 3.6z_2^{-2} - 10.8z_1^{-1} z_2^{-2}$$

$$+ 10.08z_1^{-2} z_2^{-2} - 3.6z_1^{-3} z_2^{-2} - 1.2z_2^{-3} + 3.6z_1^{-1} z_2^{-3}$$

$$- 3.6z_1^{-2} z_2^{-3} + 1.2z_1^{-3} z_2^{-3} .$$

$$D(z_1,z_2) = 1. -1.529578z_1^{-1} + 9.6912z_1^{-2} -2.14664z_1^{-3}$$
$$- 1.529578z_2^{-1}+2.339608z_1^{-1}z_2^{-1}-1.482344z_1^{-2}z_2^{-2}$$
$$- 0.208035z_1^{-3}z_2^{-3}$$
$$+ 0.96912z_2^{-2}-1.482344z_1^{-1}z_2^{-2}+0.939193z_1^{-2}z_2^{-2}$$
$$+ 4.6081\times10^{-2}z_1^{-3}z_2^{-3}$$
$$- 0.214644z_2^{-3}-3.28345z_1^{-1}z_2^{-3}-0.208035z_1^{-2}z_2^{-3}$$
$$+ 4.6081\times10^{-2}z_1^{-3}z_2^{-3}.$$

The inputs used are still those in Eqs. (99), (100), and (101). Other operating conditions (number of bits, number of input and output points and etc.) are also the same as in Example 1. Table 4 gives the numerical results.

Table 4

	Inputs			Power of 10
Errors	2I1	2I2	2I3	
RE	6.70882	5.97688	6.89292	10^{-7}
RA	8.27063	1.54102	5.85148	10^{-7}
RNE	8.91590	7.94316	9.16056	10^{-6}
CE	5.36286	4.55883	5.56276	10^{-8}
CA	0.979651	1.53522	1.00613	10^{-8}
CNE	7.12713	6.05860	7.39280	10^{-7}
IE	0.429636	0.761428	1.072392	10^{-10}
IA	0.380303	0.322345	0.66187	10^{-10}
INE	1.37214	1.03698	1.46045	10^{-10}
RCE	7.2451	3.91965	3.95618	10^{-7}
RCA	8.57491	1.54024	6.69687	10^{-7}
RCNE	9.62861	8.54902	9.89984	10^{-6}
RCIE	7.2503	3.92726	3.95725	10^{-7}
RCIA	7.94085	1.62868	5.98533	10^{-7}
RCINE	9.62875	8.54913	9.89998	10^{-6}

Example 3: One-dimensional bandpass digital filter designed by linear programming method ([10]).

Filter Coefficients:

$$N(z) = 0.10202 - 0.3031065z^{-1} + 0.3986085z^{-2} - 0.2786962z^{-3} + 0.0812557z^{-4}$$

$$D(z) = 1. - 2.98247z^{-1} + 3.95745z^{-2} - 2.59993z^{-3} + 0.758117z^{-4}.$$

Input Signals Used:

I1 : $x_j = \cos(10\,j)$, $j=0,\ldots,255$,

I2 : $x_j = \cos(0.1j)$, $j=0,\ldots,255$,

I3 : $x_j = \cos(j)$, $j=0,\ldots,255$,

I4 : $x_j = \text{noise}(j)$, $j=0,\ldots,255$,

I5 : $x_j = \text{noise}(j)\cos(j)$, $j=0,\ldots,255$,

I6 : $x_j = \cos(j+\text{noise}(j))$, $j=0,\ldots,255$,

I7 : $x_j = \cos(\text{noise}(j)\cdot j)$, $j=0,\ldots,255$,

I8 : $x_j = \text{noise}(j)\cdot\cos(j) + \text{noise}^1(j)\cdot\sin(j)$, $j=0,\ldots,255$,

I9 : $x_j = \text{noise}^2(j)\cdot\cos(j) + \text{noise}^3(j)\cdot\sin(j)$, $j=0,\ldots,255$,

where the signals noise (j), $\text{noise}^1(j)$, $\text{noise}^2(j)$, and $\text{noise}^3(j)$ are generated by a random number generator. They have different variances and different starting values.

The operations are from $n = 0$ to $n = 255$. Table 5 lists the numerical results.

Table 5

t = 15 bits

Errors	I1	I2	I3	I4	I5	I6	I7	I8	I9	Powers of 10
RE	6.73	7.05	6.51	1.64	1.60	6.15	1.73	1.60	1.61	10^{-5}
RA	2.25	3.00	2.05	0.75	0.52	2.25	0.71	0.52	0.91	10^{-5}
RNE	7.33	7.68	7.01	1.79	1.74	6.70	1.89	1.74	1.76	10^{-4}
CE	1.51	1.58	1.46	0.30	0.29	1.37	0.32	0.29	0.29	10^{-5}
CA	1.56	1.42	0.85	1.57	1.66	0.84	2.52	1.66	1.59	10^{-5}
CNE	16.4	17.2	15.8	3.29	3.18	14.9	3.53	3.18	3.22	10^{-5}
IE	1.21	1.29	1.15	0.033	0.023	1.08	0.058	0.023	0.026	10^{-8}
IA	0.47	0.26	0.39	0.012	0.014	0.26	0.027	0.014	0.011	10^{-8}
INE	13.7	14.6	13.1	0.37	0.26	12.2	0.62	0.26	0.29	10^{-8}
RCE	8.23	8.63	7.97	1.94	1.89	7.52	2.06	1.89	1.90	10^{-5}
RCA	5.51	9.60	3.38	3.23	2.97	3.02	2.52	2.97	1.72	10^{-5}
RCNE	8.97	9.4	8.7	2.11	2.06	8.2	2.24	2.06	2.08	10^{-4}
RCIE	8.23	8.63	7.97	1.94	1.89	7.52	2.06	1.89	1.90	10^{-5}
RCIA	6.51	8.18	2.26	2.22	2.49	2.90	2.46	2.50	2.23	10^{-5}
RCINE	8.97	9.41	8.7	2.11	2.06	8.2	2.24	2.06	2.08	10^{-4}

Table 5 - (continued)

t = 29 bits										
	Inputs									Powers
Errors	I1	I2	I3	I4	I5	I6	I7	I8	I9	of 10
RE	2.51	2.63	2.43	0.61	0.60	2.29	0.65	0.60	0.60	10^{-13}
RA	1.36	0.90	0.79	0.47	0.25	0.74	0.32	0.25	0.54	10^{-13}
RNE	27.3	28.6	26.5	6.65	6.49	25.0	7.04	6.48	6.54	10^{-13}
CE	1.94	2.01	1.89	0.77	0.76	1.80	0.78	0.076	0.77	10^{-14}
CA	4.39	4.21	1.82	0.36	0.37	1.81	2.96	0.37	0.36	10^{-14}
CNE	21.1	21.9	20.6	8.41	8.31	19.7	8.52	8.31	8.35	10^{-14}
IE	4.52	4.80	4.33	0.123	0.089	4.01	0.22	0.087	0.095	10^{-17}
IA	1.84	1.36	1.98	0.056	0.039	1.30	0.084	0.039	0.054	10^{-17}
INE	51.4	54.6	49.0	1.40	0.98	45.5	0.25	0.98	1.08	10^{-17}
RCE	2.70	2.83	2.61	0.69	0.67	2.47	0.72	0.67	0.68	10^{-13}
RCA	1.22	0.94	1.32	0.47	0.70	0.86	0.40	0.70	0.48	10^{-13}
RCNE	29.4	30.8	28.5	7.50	7.32	26.9	7.89	7.32	7.37	10^{-13}
RCIE	2.70	2.83	2.62	0.69	0.67	2.47	0.72	0.67	0.68	10^{-13}
RCIA	1.45	1.98	1.05	0.34	0.61	0.64	0.70	0.61	0.63	10^{-13}
RCINE	29.4	30.8	28.5	7.50	7.32	27.0	7.89	7.32	7.37	10^{-13}

V. DISCUSSION AND CONCLUSION

V.1 Discussion

We have a unified approach for dealing with uncorrelated errors in both floating-point and fixed-point two-dimensional digital filters. For both kinds of filters we have adopted the same method to derive formulas for estimating the output mean squared errors and output norm error bounds. (Note that mean squared errors and norm error bounds together should furnish us enough information to determine how many bits are needed for a digital filter to have certain desired performance.)

Eqs. (34), (35), (86), (87) and (90) require the calculations of the unit impulse responses ($\sum_m \sum_n h^2_{mn}$ and $\sum_m \sum_n g^2_{mn}$). They could be replaced by the frequency evaluations,

$$\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} h^2_{mn} = \frac{1}{(2\pi i)^2} \oint_{|z_1|=1} \oint_{|z_2|=1} \frac{1}{D(z_1,z_2)D(1/z_1,1/z_2)} \frac{dz_1}{z_1} \frac{dz_2}{z_2} \tag{102}$$

and

$$\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} g^2_{mn} = \frac{1}{(2\pi i)^2} \oint_{|z_1|=1} \oint_{|z_2|=1} G(z_1,z_2)G(\frac{1}{z_1},\frac{1}{z_2}) \frac{dz_1}{z_1} \frac{dz_2}{z_2} . \tag{103}$$

Closed form solutions of Eqs. (102) and (103) for 2D recursive digital filters are usually out of reach. Approximations are possible and one way of doing it is via FFT.

For digital filters with high gain and narrow passband area (these filters are difficult to implement), the present method of estimating the mean squared errors is difficult, since the impulse responses die out slowly, and the approximations of Eqs. (102) and (103) also require very fine step sizes. However, in many important applications ([1]) of 2D signal processing we usually require that the results of the recursive filtering be roughly independent of the initial conditions since they are usually unknown. Thus, we require that h_{mn} and hence g_{mn} die out fast enough. This means that the calculations of $\sum_m \sum_n h^2_{mn}$ and $\sum_m \sum_n g^2_{mn}$ can be cut at a reasonably small length of m and n. So, for some practical 2D recursive digital filters, the present method should serve as an effective way of estimating the output mean squared errors.

The results in Chapters III and IV can be applied to one- and multi-dimensional digital filters, and similar approaches can be adapted to analyzing the error properties of filters realized in different forms. However, for many one-dimensional digital filters, the frequency domain evaluations for $\sum_{n=o}^{\infty} h^2_n$, $\bar{w}^2_n$, and etc. are easier.

V.2 Conclusion

Block diagram representations of a two-dimensional digital filter have been introduced. A systematic treatment of uncorrelated errors for both floating-point and fixed-point 2D digital filters is presented. The analytic results include output norm error bounds (Eqs. (40), (47), (48), (49), (96), (97), and (98)) and output mean squared error estimations (Eqs. (34), (35), (36), (86), (87), and (90)). The numerical experiments have shown that the estimated errors are within an order of magnitude of the actual errors.

APPENDIX I

(i) 2D Z-transform

The 2D Z-transform $X(z_1, z_2)$ of the 2D sequence $\{x_{mn}\}_{m=0,\ n=0}^{\infty\ \ \infty}$ is defined by the double summation as:

$$Z\{x_{mn}\} = X(z_1, z_2) = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} x_{mn} z_1^{-m} z_2^{-n} \tag{A.1}$$

where z_1 and z_2 are complex variables. If the sequence $\{x_{mn}\}$ satisfies the properties:

(a) $|x_{mn}| < \infty$ for all finite m and n,

(b) $|x_{mn}| < K R_1^m R_2^n$, for all $n > \hat{N}$ or $m > \hat{M}$,

where R_1, R_2, $\hat{N}$, $\hat{M}$, and K are constants, the summation (A.1) converges absolutely. The region of the convergence of the series (A.1) is

$$D = \{(z_1, z_2) : |z_1| > R_1 \text{ and } |z_2| > R_2\}.$$

The proof of this result follows easily by considering the summation (A.1) in several parts, i.e. for (i) $m < \hat{M}$, $n < \hat{N}$, (ii) $m > \hat{M}$, $n < \hat{N}$, (iii) $m < \hat{M}$, $n > \hat{N}$, and (iv) $m > \hat{M}$, $n > \hat{N}$. It may be emphasized that the Series (A.1) is a doubly infinite series because of the two indices m and n. There are several ways of summing a series of this nature, and it is absolutely convergent if at least one arrangement for its summation converges absolutely. Further, any rearrangement of an absolutely convergent series leads to an absolutely convergent series, and the sum is the same for all arrangements.

The inversion formula for the above transform is given by

$$x_{jk} = \frac{1}{(2\pi i)^2} \oint_{c_1} \oint_{c_2} X(z_1, z_2) z_1^{j-1} z_2^{k-1} \, dz_1 dz_2, \tag{A.2}$$

where the paths of integration c_1 and c_2 are within the region of the convergence of the Series (A.1). In particular, c_1 is the contour $|z_1| = R_1^* > R_1$, and c_2 is the contour $|z_2| = R_2^* > R_2$. This formula follows by substituting for $X(z_1, z_2)$ in the right-hand side of (A.2), which yields

$$\oint_{c_1} \oint_{c_2} \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} x_{mn} z_1^{j-m-1} z_2^{k-n-1} \, dz_1 dz_2 . \tag{A.3}$$

The interchange of summations and integration gives

$$\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} x_{mn} \oint_{c_1} \oint_{c_2} z_1^{j-m-1} z_2^{k-n-1} \, dz_1 dz_2 . \tag{A.4}$$

The absolute and uniform convergence of the series for $X(z_1, z_2)$ justifies this interchange. By the Cauchy Integral Theorem of two variables (Kaplan [14]), if the paths of integration c_1 and c_2 lie in the region of convergence of the series, as chosen above, it follows that

$$\oint_{c_1} \oint_{c_2} z_1^{j-m-1} z_2^{k-n-1} \, dz_1 dz_2 = \begin{cases} 0, & j \neq m \text{ or } k \neq n , \\ (2\pi i)^2, & j=m \text{ and } k=n . \end{cases} \tag{A.5}$$

There is another expression for the inverse transform given by

$$x_{jk} = \frac{1}{j!k!} \left[\frac{\partial^{j+k}}{(\partial z_1^{-1})^j (\partial z_2^{-1})^k} X(z_1, z_2) \right]_{z_1 = \infty, z_2 = \infty} . \tag{A.6}$$

The expression (A.6) follows from expanding the function $X(z_1, z_2)$ in the neighborhood of $z_1^{-1} = 0 = z_2^{-1}$. In the case $X(z_1, z_2)$ is a given rational function of z_1^{-1} and z_2^{-1}, x_{jk} may be evaluated by determining the Series (A.1) by long division. The relationship (A.1) between $X(z_1, z_2)$ and $\{x_{mn}\}$ is designated by the notation $\{x_{mn}\} \Leftrightarrow X(z_1, z_2)$.

Many properties of the one-dimensional Z-transform are still true when they are extended suitably to the 2D Z-transform. The following properties are needed in the development of the present paper. The proofs are omitted since they follow simply from the definition of the 2D Z-transform.

(a) Linearity:

$$Z\{ax_{mn} + by_{mn}\} = aZ\{x_{mn}\} + bZ\{y_{mn}\} \tag{A.7}$$

where a and b are constants, and $\{x_{mn}\}$, $\{y_{mn}\}$ are 2D sequences.

(b) Shifting:

$$Z\{x_{m+\hat{M},\, n+\hat{N}}\} = z_1^{\hat{M}} z_2^{\hat{N}} Z\{x_{mn}\} - \sum_{\substack{j=0 \\ j+k \neq \hat{M}+\hat{N}}}^{\hat{M}} \sum_{k=0}^{\hat{N}} x_{jk} z_1^{\hat{M}-j} z_2^{\hat{N}-k} \tag{A.8}$$

where $x_{jk} = 0$, for $j<0$ or $k<0$, j and k not equal to zero.

(c) Convolution in space domain:

Let $W(z_1, z_2) \longleftrightarrow \{w_{mn}\}$, $G(z_1, z_2) \longleftrightarrow \{g_{jk}\}$ and $X(z_1, z_2) \longleftrightarrow \{x_{mn}\}$.

If $W(z_1, z_2) = G(z_1, z_2)\, X(z_1, z_2)$, then

$$w_{mn} = \sum_{j=0}^{m} \sum_{k=0}^{n} g_{jk}\, x_{m-j,n-k} = \sum_{j=0}^{m} \sum_{k=0}^{n} g_{m-j,n-k}\, x_{jk}. \tag{A.9}$$

(d) Convolution in frequency domain:

Let $U(z_1,z_2) \leftrightarrow \{U_{mn}\}$, $X(z_1,z_2) \leftrightarrow \{x_{mn}\}$, and $Y(z_1,z_2) \leftrightarrow \{y_{mn}\}$.

If $\{x_{mn}\, y_{mn}\} = \{U_{mn}\}$, then

$$U(z_1,z_2) = \frac{1}{(2\pi i)^2} \oint_{c_1} \oint_{c_2} X\left(\frac{z_1}{v_1}, \frac{z_2}{v_2}\right) Y(v_1,v_2) v_1^{-1} v_2^{-1} dv_1 dv_2 \qquad \text{(A.10)}$$

where the contours are obtained by the simple generalization of the one-dimensional case as in Kuo [15].

A special case of the above result is given by the following equality:

$$\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} x_{mn}^2 = \frac{1}{(2\pi i)^2} \oint_{|v_1|=1} \oint_{|v_2|=1} X\left(\frac{1}{v_1}, \frac{1}{v_2}\right) X(v_1,v_2) v_1^{-1} v_2^{-1} dv_1 dv_2 \qquad \text{(A.11)}$$

where $X(z_1,z_2)$ is convergent for $|z_1| \geq 1$, $|z_2| \geq 1$.

(ii) Proof of Lemma 1

First, we show that

$$\sum_{m=0}^{K_2} \sum_{n=0}^{K_1} |\hat{f}_{mn}|^2 = \frac{1}{4\pi^2} \int_0^{2\pi} \int_0^{2\pi} |F(\omega_1,\omega_2)|^2 \, d\omega_2 d\omega_1 \qquad \text{(A.12)}$$

where

$$F(\omega_1,\omega_2) = \sum_{m=0}^{K_2} \sum_{n=0}^{K_1} \hat{f}_{mn} \, e^{-im\omega_1} e^{-in\omega_2} \qquad \text{(A.13)}$$

and $$\hat{f}_{mn} = \begin{cases} f_{mn} , & m \leq M \text{ and } n \leq N, \\ 0 , & \text{otherwise.} \end{cases}$$

It is known that

if $\hat{y}_n = y_n$; $n \le N$,

$= 0$; $n > N$,

and $$f(\omega) = \sum_{n=0}^{N} \hat{y}_n e^{-in\omega} , \tag{A14}$$

then

$$\sum_{n=0}^{\infty} |\hat{y}_n|^2 = \frac{1}{2\pi} \int_0^{2\pi} |f(\omega)|^2 \, d\omega , \tag{A15}$$

where $\{y_n\}$ can be a complex sequence.

Rewrite (A13) as

$$F(\omega_1, \omega_2) = \sum_{m=0}^{K_2} \left(\sum_{n=0}^{K_1} \hat{f}_{mn} e^{-in\omega_2} \right) e^{-im\omega_1} . \tag{A16}$$

By (L4) it follows that

$$\sum_{m=0}^{\infty} \left| \left(\sum_{n=0}^{K_1} \hat{f}_{mn} e^{-in\omega_2} \right) \right|^2 = \frac{1}{2\pi} \int_0^{2\pi} |F(\omega_1, \omega_2)|^2 d\omega_1 , \tag{A17}$$

and on integrating with respect to ω_2, one obtains:

$$\frac{1}{2\pi} \int_0^{2\pi} \sum_{m=0}^{\infty} \left| \left(\sum_{n=0}^{K_1} \hat{f}_{mn} e^{in\omega_2} \right) \right|^2 d\omega_2 = \frac{1}{2\pi} \int_0^{2\pi} \frac{1}{2\pi} \int_0^{2\pi} |F(\omega_1, \omega_2)|^2 d\omega_1 \; d\omega_2$$

or

$$\sum_{m=0}^{\infty} \left[\frac{1}{2\pi} \int_0^{2\pi} \left| \sum_{n=0}^{\infty} \hat{f}_{mn} e^{-in\omega_2} \right|^2 d\omega_2 \right] \tag{A18}$$

$$= \frac{1}{4\pi^2} \int_0^{2\pi} \int_0^{2\pi} |F(\omega_1, \omega_2)|^2 \, d\omega_1 \; \omega_2 .$$

Again, by (15) one gets (A12). Now, by (A12),

$$\sum_{m=0}^{K_2}\sum_{n=0}^{K_1} |f_{mn}|^2 = \frac{1}{4\pi^2}\int_0^{2\pi}\int_0^{2\pi} \Big| \sum_{m=0}^{K_2}\sum_{n=0}^{K_1} e^{-im\omega_1} e^{-in\omega_2} \sum_{k=0}^{m}\sum_{l=0}^{n} C_{m-k,n-l}\, g_{kl}\Big|^2$$

$$d\omega_2 d\omega_1,$$

$$= \frac{1}{4\pi^2}\int_0^{2\pi}\int_0^{2\pi} \Big| \sum_{m=0}^{K_2}\sum_{n=0}^{K_1} e^{-im\omega_1} e^{-in\omega_2} \sum_{k=0}^{m}\sum_{l=0}^{n} C_{m-k,n-l}\, \hat{g}_{kl}\Big|^2$$

$$d\omega_2 d\omega_1, \qquad \text{(A19)}$$

in which

$$\hat{g}_{kl} = g_{kl} \quad \text{, for } k \le M, \text{ and } l \le N \qquad \text{(A20)}$$

$$= 0 \quad , \quad \text{otherwise.}$$

Further

$$\sum_{m=0}^{K_2}\sum_{n=0}^{K_1} |f_{mn}|^2 \le \frac{1}{4\pi^2}\int_0^{2\pi}\int_0^{2\pi} \Big| \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} e^{-im\omega_1} e^{-in\omega_2} \sum_{k=0}^{m}\sum_{l=0}^{n} C_{m-k,n-l}$$

$$\hat{g}_{kl}\Big|^2 d\omega_2 d\omega_1 , \qquad \text{(A21)}$$

(After reordering and factoring)

$$\le \frac{1}{4\pi^2}\int_0^{2\pi}\int_0^{2\pi} \Big| \sum_{k=0}^{\infty}\sum_{l=0}^{\infty} C_{kl}\, e^{-ik\omega_1} e^{-il\omega_2} \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} e^{-im\omega_1} e^{-in\omega_2}$$

$$\hat{g}_{mn}\Big|^2 d\omega_2\, d\omega_1 , \qquad \text{(A22)}$$

$$\le \max_{\substack{0\le\omega_1\le 2\pi \\ 0\le\omega_2\le 2\pi}} \Big| \sum_{k=0}^{\infty}\sum_{l=0}^{\infty} C_{kl}\, e^{-ik\omega_1} e^{-il\omega_2} \Big|^2 .$$

$$\frac{1}{4\pi^2}\int_0^{2\pi}\int_0^{2\pi}\left|\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} e^{-im\omega_1}\, e^{-in\omega_2}\, \hat{g}_{mn}\right|^2 d\omega_2\, d\omega_1 \tag{A23}$$

$$= \max_{\substack{0\le\omega_1\le 2\pi \\ 0\le\omega_2\le 2\pi}} \left|\sum_{k=0}^{\infty}\sum_{l=0}^{\infty} C_{kl}\, e^{-ik\omega_1}\, e^{-il\omega_2}\right|^2 \sum_{m=0}^{K_2}\sum_{n=0}^{K_1} |g_{mn}|^2 \tag{A24}$$

which proves the Lemma. The present lemma is a generalization of a similar lemma in one-dimension by Sandberg [4].

APPENDIX II

PROGRAM LISTINGS

```
      SUBROUTINE DF2I (IESTRP,ERMST,ERMSTRO,HY,EOSQRE,HH,H,HX,GMX,HMX,
     *ERMSTB,ERMSTBO)
C
C
C
C     PURPOSE :
C        TO CALCULATE (1) THE MEAN SQUARED ERROR ESTIMATIONS AND (2) THE
C     NORM ERROR BOUNDS FOR ALL SOURCES OF ERROR IN TWO-DIMENSIONAL
C     DIGITAL FILTERS EMPLOYING FLOATING-POINT ARITHMETIC.
C
C
C     ATTENTION :
C        WITH SOME CHANGES, THIS SUBROUTINE CAN BE EASILY MODIFIED
C     TO DO THE SAME JOB FOR TWO-DIMENSIONAL DIGITAL FILTERS
C     EMPLOYING FIXED-POINT ARITHMETIC.
C
C
C     SUBROUTINES NEEDED :
C        FLTGR, UNTZU, SHFT.
C
C
C     INPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS) :
C        IESTRP,HY,EOSQRE,HH,H,HX,GMX,HMX,A,B,X,MA,NA,MB,NB,
C        MM,NN,NSTORE.
C
C     OUTPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS) :
C        ERMST,ERMSTRO,ERMSTB,ERMSTBO,HY,HH,H,HX,Y.
C
C     DESCRIPTIONS OF INPUT AND OUTPUT ARGUEMENTS :
C
C     IESTRP=1 :  NO CALCULATIONS FOR HH,H,AND HX.
C     IESTRP=0 :  NO CALCULATIONS FOR HH AND H.
C     IESTRP=-1 :  CALCULATIONS FOR HH,H,AND HX.
C     ERMST(I) :  TO STORE THE ESTIMATED OUTPUT MEAN SQUARED ERROR.
C     HY :  THE OUTPUT SQUARE SUM.
C     ERMSTRO(I) :  ERMST(I)/HY.
C     EOSQRE :  EO*EO, WHERE EO IS THE QUANTIZATION STEP SIZE.
C     HH :  THE IMPULSE RESPONSE SQUARE SUM FOR THE DENOMINATOR FILTER.
C     H :  THE IMPULSE RESPONSE SQUARE SUM FOR THE WHOLE FILTER.
C     HX :  THE INPUT SQUARE SUM.
C     GMX :  THE SQUARED VALUE OF THE MAXIMUM AMPLITUDE FREQUENCY RESPONSE
C           FOR THE WHOLE FILTER.
C     HMX :  THE SQUARED VALUE OF THE MAXIMUM AMPLITUDE FREQUENCY RESPONSE
C           FOR THE DENOMINATOR FILTER.
C     ERMSTB(I) :  TO STORE THE ESTIMATED NORM ERROR BOUND.
C     ERMSTBO(I) :  ERMSTB(I)/HY.
C     ((A(I,J),I=1,...,MA),J=1,....,NA), ((B(I,J),I=1,...,MB),J=1,...,NB) :
C        STORING THE TWO-DIMENSIONAL DIFITAL FILTER COEFFICIENTS,
C        A FOR DENOMINATOR COEFFICIENTS,
C        B FOR NUMERATOR COEFFICIENTS.
C     MM,NN :    SPATIAL RANGE OF FILTERING OPERATIONS,
C               TOTALLY (MM*NN) POINTS.
C     (((X(I,J),Y(I,J)),J=1,...,NN),I=1,....,MM) :
C        INPUT AND OUTPUT POINTS OF THE DIGITAL FILTER.
C     NSTORE :    NN*2, THE NUMBER OF COMPUTER WORDS TRANSFERRED BACK AND
C                FORTH BETWEEN DISK AND CENTRAL MEMORY.
C
C
C
C
C
C
```

```
      COMMON/DF2I1/A(4,4),B(4,4),Y(4,128),X(4,128)
      COMMON/DF2I2/MA,NA,MB,NB,MM,NN,NSTORE
      COMMON/DF2I7/IERROR
      COMMON/SHFT1/STORE(125)
      DIMENSION YMA(4,4),YMB(4,4),ALPHA(4,4),BETA(4,4),QEA(4,4),QEB(4,4)
     *,ERMST(9),ERMSTRO(9),QEBO(4,4),ERMSTB(5),ERMSTBO(5)
      LOGICAL IAUX1L,INDXL
      DOUBLE PRECISION A,B,Y,X,YMA,YMB,V,HY,HM,ALPHA,BETA,QEA,QEB,HD,HR,
     *STORE,H,EOSQRE,AUX1,AUX2,FREF,HX,HBNM,HBONM,HR
C
      CALL IOP(3HREW,1)
      CALL IOP(3HREW,2)
      IF(IESTRP) 191,191,192
  191 DO 24 I=1,MA
      DO126 J=1,NN
  126 Y(I,J)=0.D0
      DO 24 J=1,NA
   24 YMA(I,J)=0.D0
      DO 124 I=1,MB
      DO 27 J=1,NN
   27 X(I,J)=0.D0
      DO124 J=1,NB
  124 YMB(I,J)=0.D0
C
C
      INDXL=.F.
      HX=0.D0
      H=0.D0
      IND=-1
      INDGT=1
      IAUX1L=.F.
C
  647 DO 62 M=1,MM
      CALL SHFT (IND,M)
      DO 62 N=1,NN
      Y(MA,N)=0.D0
      GO TO (644,700),INDGT
C
  700 DO964 J=1,NB
      LJ=N-J+1
      IF(LJ.LT.1) GO TO964
      DO 64 I=1,MB
      LI=MB-I+1
      V=X(LI,LJ)*B(I,J)
      Y(MA,N)=Y(MA,N)+V
      GO TO (64,674),INDGT
  674 YMB(I,J)=YMB(I,J)+V*V
   64 CONTINUE
  964 CONTINUE
C
      GO TO 645
C
  644 IF(IAUX1L) GO TO 645
      IF(IND) 679,669,679
  679 Y(MA,N)=1.D0
      GO TO 659
  669 IF(M.GT.(MB+1)) GO TO 659
      X(MB,N)=0.D0
      IF(M.EQ.1.AND.N.EQ.1) X(MB,N)=1.D0
      GO TO 700
  659 IAUX1L=.T.
C
```

```
 665 DO966 J=1,NA
     LJ=N-J+1
     IF(LJ.LT.1) GO TO966
     IA=1
     IF(J.EQ.1) IA=2
     DO 66 I=IA,MA
     LI=MA-I+1
     V=Y(LI,LJ)*A(I,J)
     Y(MA,N)=Y(MA,N)-V
     GO TO (66,676),INDGT
 676 YMA(I,J)=YMA(I,J)+V*V
  66 CONTINUE
 966 CONTINUE
C
     H=H+Y(MA,N)*Y(MA,N)
     IF(INDXL) HX=HX+X(MA,N)*X(MA,N)
  52 CONTINUE
C
C
     IF(IND) 701,646,702
 701 HH=H
     PRINT 1773,HH
1773 FORMAT(1X//1X,*THE IMPULSE RESPONSE SQRE SUMS OF THE*,
    ** DENOMINATOR FILTER $   *,D17.10)
     H=0.D0
     IF(IERROR.EQ.1) INDXL=.T.
     IND=1
     INDGT=2
     DO 16 I=1,MA
     DO 16 J=1,NN
  16 Y(I,J)=0.
C
C    GO BACK TO FIND THE OUTPUTS WITH INPUTS X
     GO TO 647
C
 702 HY=H
C
     DO 58 I=1,MA
     DO 57 J=1,NN
  57 STORE(J)=Y(I,J)
  58 CALL IOP(2HWB,2,STORE,NSTORE)
C
C    PREPARE TO GO BACK TO FIND THE IMPULSE RESPONSE FOR THE
C    WHOLE FILTER
C
     IND=0
     INDGT=1
     DO 14 I=1,MB
     DO 14 J=1,NN
  14 X(I,J)=0.D0
     H=0.D0
     INDXL=.F.
     DO 15 I=1,MA
     DO 15 J=1,NN
  15 Y(I,J)=0.D0
     IAUXIL=.F.
     GO TO 647
C
C    SET UP VARIOUS PARAMETER VALUES FOR ERROR ESTIMATIONS.
C
C
C**************************************************************************
```

```
C      THE FOLLOWING STATEMENTS SHOULD BE PROPERLY ADJUSTED WHEN ONE OR
C      MORE OF THE FILTER COEFFICIENTS IS ZERO OR ONE.
C*****************************************************************************
  940 DO824 I=1,MB
      DO824 J=1,NB
      IF(I.EQ.1.AND.J.EQ.1) BETA(I,J)=(MB*NB)
      IF(I.NE.1.OR.J.NE.1) BETA(I,J)=((MB-I)*(NB-J)+1)
  824 CONTINUE
      DO 26 I=1,MA
      DO 26 J=1,NA
      IF(I.EQ.1.AND.J.EQ.2) ALPHA(I,J)=MA*NA
      IF(I.EQ.1.AND.J.NE.2) ALPHA(I,J)=(MA*NA-J+2)
      IF(I.EQ.1.AND.J.EQ.1) ALPHA(I,J)=0.
      IF(I.NE.1) ALPHA(I,J)=((MA-I)*(NA-J)+2)
   26 CONTINUE
C*****************************************************************************
      DO 224 I=1,MB
      DO224 J=1,NB
  224 QEBO(I,J)=YMB(I,J)/3.D0*BETA(I,J)
      DO 226 I=1,MA
      DO 226 J=1,NA
  226 QEA(I,J)=YMA(I,J)/3.D0*ALPHA(I,J)
  192 HR=0.
      DO 228 I=1,MA
      DO 228J=1,NA
  228 HR=HR+QEA(I,J)
      HBRNM=0.D0
      DO 230 I=1,MB
      DO 230 J=1,NB
  230 HBRNM=HBRNM+QEBO(I,J)
      MMNN=MM*NN
      HD=0.
      HBNM=0.D0
      DO 771 I=1,MB
      DO 771 J=1,NB
      AUX1=B(I,J)
      CALL FLTGR(AUX1,FREE,KAUX1,.T.,.T.,AUX2)
      AUX1=(AUX1-AUX2)/AUX2
  771 HBNM=HBNM+YMB(I,J)*AUX1*AUX1
      DO 772 I=1,MA
      DO 772 J=1,NA
      AUX1=A(I,J)
      CALL FLTGR(AUX1,FREE,KAUX1,.T.,.T.,AUX2)
      AUX1=(AUX1-AUX2)/AUX2
  772 HD=HD+YMA(I,J)*AUX1*AUX1
      IF(IESTRP.LE.0) PRINT 1774,H
 1774 FORMAT(1X,*THE IMPULSE RESPONSE SQRE SUM OF THE*,
     ** WHOLE FILTER $  *,12X,D17.10//)
C
C      ROUNDOFF ERRORS ESTIMATION
C
      ERMST(1)=(HR+HBRNM)*HH*EDSQRE
      ERMSTRO(1)=ERMST(1)/HY
      ERMST(1)=ERMST(1)/MMNN
C
C      ESTIMATION OF COEFFICIENT QUANTIZATION ERRORS
C
      ERMST(2)=(HD+HBNM)*HH
      ERMSTRO(2)=ERMST(2)/HY
      ERMST(2)=ERMST(2)/MMNN
C
C      ESTIMATION OF INPUT QUANTIZATION ERRORS
```

```
C
      ERMST(3)=HX*H*EUSQRE/3.D0
      ERMSTRO(3)=ERMST(3)/HY
      ERMST(3)=ERMST(3)/MMNN
C
      ERMST(4)=ERMST(1)+ERMST(2)
      ERMSTRO(4)=ERMSTRO(1)+ERMSTRO(2)
      ERMST(5)=ERMST(4)+ERMST(3)
      ERMSTRO(5)=ERMSTRO(4)+ERMSTRO(3)
C
C
C     ESTIMATION OF NORM ERROR BOUNDS
C
      ERMSTB(1)=(HR+HBRNM)*HUX*EUSQRE
      ERMSTB(2)=(HU+HBNM)*HMX
      ERMSTB(3)=HX*GMA*EUSQRE/3.D0
      ERMSTBO(1)=ERMSTB(1)/HY
      ERMSTBO(2)=ERMSTB(2)/HY
      ERMSTBO(3)=ERMSTB(3)/HY
      ERMSTB(1)=ERMSTB(1)/MMNN
      ERMSTB(2)=ERMSTB(2)/MMNN
      ERMSTB(3)=ERMSTB(3)/MMNN
      ERMSTBO(4)=ERMSTBO(1)+ERMSTBO(2)
      ERMSTB(5)=ERMSTB(4)+ERMSTB(3)
      ERMSTB(4)=ERMSTB(1)+ERMSTB(2)
      ERMSTBO(5)=ERMSTBO(4)+ERMSTBO(3)
C
      RETURN
      END
```

```
      SUBROUTINE SHFT(ISTOR,M)
C
C
C     PURPOSE :
C        AN AUXILIARY SUBROUTINE FOR SUBROUTINE DF2I.
C
C
      COMMON/DF2I1/A(4,4),B(4,4),Y(4,128),X(4,128)
      COMMON/DF2I2/MA,NA,MB,NB,MM,NN,NSTORE
      COMMON/SHFT1/STORE(128)
      DOUBLE X,STORE,Y,A,B
C
      IF(ISTOR) 101,104,100
  100 DO 1 I=2,MB
      DO 1 J=1,NN
    1 X(I-1,J)=A(I,J)
      CALL IOP(2HRB,1,STORE,NSTORE)
      DO 2 J=1,NN
    2 X(MB,J)=STORE(J)
      GO TO 101
  104 IF(M.GT.(MB+1)) GO TO 101
      DO 6 I=1,MB
    6 X(I-1,1)=A(I,1)
  101 IF(M.LE.MA) GO TO 102
      DO 3 J=1,NN
    3 STORE(J)=Y(1,J)
      IF(ISTOR) 102,102,110
  110 CALL IOP(2HWB,2,STORE,NSTORE)
  102 DO 4 I=2,MA
      DO 4 J=1,NN
    4 Y(I-1,J)=Y(I,J)
      RETURN
      END
```

```
      SUBROUTINE DF2DR (ERMAX,ERMS,ERMSRU,HY)
C
C
C
C     PURPOSE :
C        TO SIMULATE THE OPERATIONS OF TWO-DIMENSIONAL DIGITAL
C     FILTER EMPLOYING FLOATING POINT ARITHEMATIC.
C
C
C
C     NOTE :
C        THIS SUBROUTINE CAN BE MODIFIED TO BECOME A MUCH SIMPLER
C     VERSION WHICH SIMULATES THE OPERATIONS OF TWO-DIMENSIONAL
C     DIGITAL FILTER EMPLOYING FIXED-POINT ARITHMETIC.
C
C
C     SUBROUTINES NEEDED :
C        FLTGA,FLTGR,FLTGM,QNTZD,SHFTDR.
C
C
C     INPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS) :
C        HY,A,B,Y,X,EO,EOH,EOI,EOVER1,DL2,IS,EOVER2,INDQRL,INDQIL,INDQCL,
C        MM,NN,MA,NA,MB,NB,NSTORE.
C
C     OUTPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS) :
C        ERMAX,ERMS,ERMSRU,YNT.
C
C     DESCRIPTIONS OF INPUT AND OUTPUT ARGUEMENTS :
C
C     HY,A,B,Y,X,MA,NA,MB,NB,MM,NN,NSTORE :
C        PLEASE SEE SUBROUTINE DF2I.
C     EO,EOH,EOI,EOVER1,EOVER2,DL2,IS :  PLEASE SEE SUBROUTINE FLTGR.
C     INDQRL=.T. :  SIMULATIONS INCLUDING ROUNDOFFS.
C     INDQCL=.T. :  SIMULATIONS INCLUDING COEFFICIENT QUANTIZATIONS.
C     INDQIL=.T. :  SIMULATIONS INCLUDING INPUT QUANTIZATIONS.
C     ERMAX :  THE MAXIMUM ACTUAL ERROR.
C     ERMS :  THE ACTUAL MEAN SQUARED ERROR.
C     ERMSRU :  ERMS/HY.
C     ((YNT(I,J),J=1,...,NN),I=1,...,MM) :
C        THE ACTUAL OUTPUTS OF THE DIGITAL FILTER.
C
C
C
C
C
      COMMON/QNTZD1/EO,EOH,EOI,EOVER1,EOVER2
      COMMON/FLTGR1/DL2,IS
      COMMON/SHFT1/STORE(128)
      COMMON/DF2I1/A(4,4),B(4,4),Y(4,128),X(4,128)
      COMMON/DF2I2/MA,NA,MB,NB,MM,NN,NSTORE
      COMMON/FLTGM1/INDQRL/DF2DR1/INDQCL,INDQIL
      COMMON/DF2DR4/VNYNT(4,128),VNX(4,128),KAYNT(4,128),KAX(4,128)
      DIMENSION AA(4,4),BB(4,4),KA(4,4),KB(4,4)
      LOGICAL INDQRL,INDQCL,INDQIL,KAUXIL
      DOUBLE PRECISION A,B,Y,X,YNT,AA,BB,EO,EOH,DL2,HY,FREE,VNYNT,VNX,
     *STORE,AUX1,AUX2,AUXY,EOI,EOVER1,EOVER2
C
      CALL IOP(3HREW,1)
      CALL IOP(3HREW,2)
      ERMS=0.
      ERMAX=0.
C
```

```
      KAUX1L=IFUNCL
      DO 32 I=1,MA
      DO 32 J=1,NA
      AUX1=A(I,J)
   32 CALL FLTGR (AUX1,AA(I,J),KA(I,J),KAUX1L,.F.,FREE)
      DO 30 I=1,MB
      DO 30 J=1,NB
      AUX1=B(I,J)
   30 CALL FLTGR (AUX1,BB(I,J),KB(I,J),KAUX1L,.F.,FREE)
C
C
C
      DO 24 I=1,MA
      DO 24 J=1,NN
      VNYNT(I,J)=0.D0
   24 KAYNT(I,J)=0
      DO 26 I=1,MB
      DO 26 J=1,NN
      VNX(I,J)=0.D0
   26 KAX(I,J)=0
C
      DO962 N=1,MM
      CALL SHFTDR
      DO 62 M=1,NN
      AUXY=0.D0
      KAUXY=0
C
      DO 64 I=1,MB
      LI=MB-I+1
      NBB=NB
      IF(N.LT.NB) NBB=N
      DO 64 J=1,NBB
      LJ=N-J+1
      AUX1=VNX(LI,LJ)
      KAUX1=KAX(LI,LJ)
      CALL FLTGM (AUX2,KAUX2,BB(I,J),KB(I,J),AUX1,KAUX1)
      CALL FLTGA (AUXY,KAUXY,AUX2,KAUX2,.T.)
   64 CONTINUE
C
      DO966 I=1,MA
      JA=1
      IF(I.EQ.1) JA=2
      NBB=NA
      IF(N.LT.NA)NBB=N
      LI=MA-I+1
      IF(NBB.LT.JA) GO TO966
      DO 66 J=JA,NBB
      LJ=N-J+1
      AUX1=VNYNT(LI,LJ)
      KAUX1=KAYNT(LI,LJ)
      CALL FLTGM (AUX2,KAUX2,AA(I,J),KA(I,J),AUX1,KAUX1)
      CALL FLTGA (AUXY,KAUXY,AUX2,KAUX2,.F.)
   66 CONTINUE
  966 CONTINUE
C
      IF(AUXY) 400,401,400
  400 Y(MA,N)=AUXY*2.D0**KAUXY
      GO TO 402
  401 Y(MA,N)=0.D0
  402 VNYNT(MA,N)=AUXY
      KAYNT(MA,N)=KAUXY
      AUX1=STORE(N)-Y(MA,N)
```

```
      ERMS=ERMS+AUX1*AUX1
      IF(DABS(AUX1).GE.ERMAX) ERMAX=AUX1
   62 CONTINUE
  962 CONTINUE
      ERMSRO=ERMS/MY
      ERMS=ERMS/(MM*NN)
      RETURN
      END
```

```
      SUBROUTINE SHFTOR
C
C
C     PURPOSE :
C        AN AUXILIARY SUBROUTINE FOR SUBROUTINE SDF2DR.
C
C
C
      COMMON/DF2DR1/INDQCL,INDWIL
      COMMON/DF2I2/MA,NA,MB,NB,MM,NN,NSTORE
      COMMON/SHFT1/STORE(128)
      COMMON/DF2DR4/VNYNT(4,128),VNX(4,128),KAYNT(4,128),KAX(4,128)
      LOGICAL INDQCL,INDWIL,KAUXIL
      DOUBLE PRECISION STORE,AUX1,AUX2,FREE,VNX,VNYNT
C
      DO 1 I=2,MB
      DO 1 J=1,NN
      VNX(I-1,J)=VNX(I,J)
    1 KAX(I-1,J)=KAX(I,J)
      CALL IOP(2HRB,1,STORE,NSTORE)
      KAUXIL=INDWIL
      DO 2 J=1,NN
      AUX1=STORE(J)
      CALL FLTGR (AUX1,AUX2,KAUX2,KAUXIL,.F.,FREE)
      VNX(MB,J)=AUX2
    2 KAX(MB,J)=KAUX2
      DO 4 I=2,MA
      DO 4 J=1,NN
      VNYNT(I-1,J)=VNYNT(I,J)
    4 KAYNT(I-1,J)=KAYNT(I,J)
      CALL IOP (2HRB,2,STORE,NSTORE)
      RETURN
      END
```

```
      SUBROUTINE FLTGA (VN1,KA1,VN2,KA2,INDPL)
C
C
C     PURPOSE:
C        TO SIMULATE DECIMALLY THE BINARY ADDITION (OR SUBTRACTION)
C     OF TWO FLTG-PT. NUMBERS (VN1*2.D0D0**(KA1)) AND (VN2*2.D0D0**(KA2)).
C
C
C     SUBROUTINE NEEDED:
C        QNTZD.
C
C
C     INPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS):
C        VN1,VN2,KA1,KA2,INDPL,EO,EOH,EOI,DL2,IS
C
C     OUTPUT ARGUEMENTS:
C        VN1,KA1
C
C     DESCRIPTIONS OF INPUT AND OUTPUT ARGUEMENTS:
C
C     VN1,KA1:  AS INPUTS, PLEASE SEE (PURPOSE).
C               AS OUTPUTS,TO STORE THE RESULTS OF THE ADDITION
C               OR SUBTRACTION.
C     VN2,KA2:  PLEASE SEE (PURPOSE).
C     INDPL=.T.:  INDICATING (VN1*2.D0D0**(KA1)) + (VN2*2.D0D0**(KA2)).
C           .F.:  INDICATING                     -
C     INDQRL=.T.:  QUANTIZN. AND OVERFLOW LIMITN. DESIRED IN THE OPERATN.
C     EO,EOH,EOI,EOVER1,EOVER2,DL2,IS:  PLEASE SEE SUBROUTINE FLTGR.
C
C
C
      COMMON/QNTZD1/EO,EOH,EOI,EOVER1,EOVER2
      COMMON/FLTGR1/DL2,IS
      COMMON/FLTGM1/INDQRL
      LOGICAL INDPL,INDQRL
      DOUBLE PRECISION VN1,VN2,DL2,EO,EOH,SN,VN1,EOI,EOVER1,EOVER2
C
C
      IF(VN2.EQ.0.D0) RETURN
      IF(VN1.EQ.0.D0) GO TO 10
C
      N=KA1-KA2
      NN=IABS(N)
      SN=2.D0**(-NN)
      IF(N) 1,2,3
    1 VN1=VN1*SN
      IF(INDQRL) CALL QNTZD (VN1,VN1)
      KA1=KA2
      GO TO 2
    3 VN2=VN2*SN
      IF(INDQRL) CALL QNTZD (VN2,VN2)
    2 IF(INDPL) VN1=VN1+VN2
      IF(.NOT.INDPL) VN1=VN1-VN2
      SN=DABS(VN1)
      IF(IS.EQ.1) GO TO 200
      IF(VN1.LT.1.D0.AND.VN1.GE.(-1.D0)) GO TO 41
      GO TO 400
  200 IF(SN.LT.1.D0) GO TO 41
  400 VN1=VN1*0.5D0
      IF(INDQRL) CALL QNTZD (VN1,VN1)
      KA1=KA1+1
      RETURN
```

```
41  IF(SN.GE.0.5D0.OR.SN.EQ.1.D0) RETURN
    CALL FLTGR(VN1,VN,KA4,F,,F,,SN)
    KA1=KA1+K4
    VN1=VN
    RETURN
100 VN1=VN2
    KA1=KA2
    IF(.NOT.INDPL) VN1=-VN1
    RETURN
    END
```

```
      SUBROUTINE FLTGM (VN1,KA1,VN2,KA2,VN3,KA3)
C
C
C     PURPOSE:
C        TO SIMULATE THE MULTIPLICATION OF TWO FLOATING-POINT
C     NUMBERS (VN2*2.D0**(KA1)) AND (VN3*2.D0**(KA3)).  THE
C     RESULTS ARE STORED AS (VN1*2.D0**(KA1)).
C
C
C     SUBROUTINE NEEDED :
C        QNTZD.
C
C
C     INPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS):
C        VN2,KA2,VN3,KA3,EO,EOH,EOI,DL2,IS
C
C     OUTPUT ARGUEMENTS:
C        VN1,KA1
C
C     DESCRIPTIONS OF INPUT AND OUTPUT PARAMETERS:
C
C     VN1,VN2,VN3,KA1,KA2,KA3:  PLEASE SEE (PURPOSE).
C     EO,EOH,EOI,EOVER1,EOVER2,DL2,IS:  PLEASE SEE SUBROUTINE FLTGR.
C     INDQRL=.T. :  QUANTIZATION OF MULTIPLICATION IN THE OPERATION DESIRED.
C
C
      COMMON/QNTZD1/EO,EOH,EOI,EOVER1,EOVER2
      COMMON/FLTGR1/DL2,IS
      COMMON/FLTGM1/INDQRL
      DOUBLE PRECISION VN1,VN2,VN3,EO,EOH,EOI,DL2,VN,EOVER1,EOVER2
      LOGICAL INDQRL
C
C
      IF(VN3.EQ.0.D0.OR.VN2.EQ.0.D0) GO TO 100
      KA1=KA2+KA3
      VN1=VN2*VN3
      IF(INDQRL) CALL QNTZD (VN1,VN1)
      IF(DABS(VN1)-0.5D0) 10,129,129
   10 VN1=VN1*2.D0
      KA1=KA1-1
      GO TO 129
  100 KA1=0
      VN1=0.D0
  129 RETURN
      END
```

```
      SUBROUTINE FLTGR (V,VN,KA,INDQL,INDQVL,VQ)
C
C
C     PURPOSE:
C        TO REPRESENT A NUMBER (V) AS (VN*2.D0**(KA)), WHERE THE
C     ABSOLUTE VALUE OF (VN) IS BETWEEN 1.D0 AND 0.5D0 OR 0.D0
C     AND (KA) IS AN INTEGER.
C
C
C     SUBROUTINE NEEDED:
C        QNTZD.
C
C
C     INPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS):
C        V,INDQL,INDQVL,EQ,EQH,EQI,DL2,IS
C
C     OUTPUT ARGUEMENTS:
C        VN,KA,VQ[AT INDQVL=.T.]
C
C     DESCRIPTIONS OF INPUT AND OUTPUT PARMETERS:
C
C     V,VN,KA: PLEASE SEE [PURPOSE]
C     INDQL=.T.: QUANTIZATION ON (VN) IS DESIRED.
C     VQ: QUANTIZED VALUES OF (V) IN DECIMAL.
C     INDQVL=.T.:  VQ IS DESIRED.
C     IS=1: SIGN AND MAGNITUDE REPRESENTATION FOR (VN)
C           (DECIMAL EQUIVALENT).
C     EQ,EQH,EQI,EQVER1,EQVER2:  PLEASE SEE SUBROUTINE QNTZD.
C     DL2: 1./DLOG(2.D0).
C
      COMMON/QNTZD1/EQ,EQH,EQI,EQVER1,EQVER2
      COMMON/FLTGR1/DL2,IS
      LOGICAL INDQL,INDQVL
      DOUBLE PRECISION V,VN,VQ,EQ,EQH,EQI,DL2,EQVER1,EQVER2
C
      VN=DABS(V)
      IF(VN.EQ.0.D0) GO TO 100
      IF (VN.LT.2.D0.AND.VN.GE.0.5D0) GO TO 925
C
      A=DLOG(VN)*DL2
      KA=INT(A)
      IF(KA.GE.0) KA=KA+1
      GO TO 926
  925 IF(VN.GE.1) GO TO 929
  928 KA=0
      GO TO 926
  929 KA=1
  926 VN=2.D0**(KA)
      VN=V/VN
      IF(IS.EQ.1) GO TO 200
      IF(VN.LT.1.D0.AND.VN.GE.(-1.D0)) GO TO 300
      GO TO 400
  200 IF(DABS(VN).LT.1.D0) GO TO 300
  400 VN=VN*0.5D0
      KA=KA+1
  300 IF(INDQL) CALL QNTZD (VN,VN)
      GO TO 1234
  100 VN=0.
      KA=-1024
```

```
1234 IF(INDLVL) VW=VN*2.D0**KA
     RETURN
     END
```

```
      SUBROUTINE QNTZD (V,VNN)
C
C
C     PURPOSE:
C        TO QUANTIZE A NUMBER (INCLUDING OVERFLOW LIMITATION).
C
C
C     INPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENT):
C        V,EO,EOH,EOI,EOVER1,EOVER2
C
C     OUTPUT ARGUEMENTS:
C        VNN
C
C     DESCRIPTIONS OF INPUT AND OUTPUT ARGUEMENTS:
C
C     EO:  QUANTIZATION LEVEL OR STEP SIZE.
C     EOH:  HALF OF EO.
C     EOI:  1.D0/EO.
C     V:  THE NUMBER TO BE QUANTIZED AND LIMITED.
C     VNN:  THE OUTPUT NUMBER.
C     EOVER1,EOVER2:  THE LOWER LIMIT AND THE UPPER LIMIT FOR VNN.
C
C
      DOUBLE PRECISION V,VN,EO,EOH,D ,VNN,EOI,EOVER1,EOVER2
      COMMON/QNTZD1/EO,EOH,EOI,EOVER1,EOVER2
C
      V1=V
      K=V*EOI
      VN=EO*FLOAT(K)
      D=V-VN
      VNN=VN
      IF(V1) 1,2,2
    1 IF (D.LE.(-EOH)) VNN=VN-EO
      IF(VNN.LT.EOVER2) VNN=EOVER2
      GO TO 900
    2 IF (D.GE.EOH) VNN= VN+EO
      IF(VNN.GT.EOVER1) VNN=EOVER1
  900 RETURN
      END
```

```
      PROGRAM DFID1 (INPUT,OUTPUT)
C
C
C
C     PURPOSE $
C        TO ANALYZE THE ERROR PROPERTIES FOR ONE-DIMENSIONAL DIGITAL
C     FILTERS EMPLOYING FLOATING-POINT ARITHMETIC BY USING SUBROUTINES
C     DF2IO AND DF2DRO TO ESTIMATE MEAN SQUARED ERRORS AND NORM ERROR
C     BOUNDS AND TO COMPARE THEM WITH THE ACTUAL ERRORS OBTAINED THROUGH
C     SIMULATIONS.
C
C
C     SUBROUTINES NEEDED $
C        DF2IO,DF2DRO,NOISE,FLTGA,FLTGM,FLTGR,QNTZD,FREQZ1.
C
C
C     INPUT ARGUEMENTS BY (FORTRAN IV) READ STATEMENTS $
C        NA,NB,NN,IPRNT,IS,NT1,NT2,NTINCR,INDQ,INDX,AMP,WI,WJ,STD,BMFY,
C        START,STD2,START2,A,B.
C
C     DESCRIPTIONS OF INPUT ARGUEMENTS $
C
C     NA,NB,NN,A,B $  PLEASE SEE SUBROUTINE DF2IO.
C     IPRNT=-1 $  THE FORTRAN VARIABLE NPRINT (PLEASE SEE SUBROUTINE
C                 DF2IO) WILL BE GIVEN THE VALUE (-1).
C     IPRNT=0 $  NPRINT=NN/65+1.
C     IPRNT = 0 $  NPRINT=NN/IPRNT+1.
C     IS=1 $  SIGN AND MAGNITUDE RESPRESENATION FOR MANTISSA OF A FLOAT-
C     IS=1 $  SIGN AND MAGNITUDE RESPRESENATION FOR MANTISSA OF A FLOAT-
C             ING-POINT NUMBER IN SIMULATING OPERATIONS OF THE DIGITAL FILTERS.
C     NT1,NT2,NTINCR $  THE SIMULATIONS AND ESTIMATIONS WILL BE DONE FOR
C                       T (NUMBER OF BINARY BITS) =NT1,NT1+NTINCR,NT1+2
C                       *NTINCR,.....,(AROUND) NT2.
C     INDQ=-1 $  ERROR ANALYSIS FOR INPUT QUANTIZATION ONLY (I).
C     INDQ= 0 $  ERROR ANALYSIS FOR COEFFICIENT QUANTIZN. ONLY (C).
C     INDQ= 1 $  ERROR ANALYSIS FOR ROUND OFF ERROR ONLY (R).
C     INDQ= 2 $  ERROR ANALYSIS FOR COEFFICIENT QUANTIZATIONS AND
C                ROUNDOFF ERRORS (R AND C).
C     INDQ= 3 $  ERROR ANALYSIS FOR INPUT AND COEFFICIENT QUANTIZATIONS
C                AND ROUNDOFF ERRORS (R  C  AND I).
C     INDQ= 4 $  CASES FOR INDQ=0, 1, AND 2.
C     INDQ= 5 $  CASES FOR INDQ=-1, 0, 1, 2, AND 3.
C     INDX,AMP,WI,WJ,STD,START,STD2,START2 $
C        PARAMETERS FOR ASSIGNING THE INPUTS TO THE DIGITAL FILTER.
C        (PLEASE SEE THE SECTION FOR INPUT SETUP IN THIS PROGRAM).
C        INDX=5 OR <5, ONLY ONE OF THE NINE KINDS OF INPUTS IS USED FOR
C        ERROR ANALYSIS,  IND=6 OR >6, NINE KINDS OF INPUTS ARE USED.
C     BMFY=0 $  NO MODIFICATIONS FOR THE FILTER NUMERATOR COEFF NTS, B(I).
C     BMFY=-1 $  ALL B(I) ARE DIVIDED BY B(1)/1.2.
C     BMFY =0 $  ALL B(I) ARE DIVIDED BY BMFY.
C
C
C     COMMENTS $
C     1. THE MAXIMUM VALUE OF THE AMPLITUDE FREQUENCY RESPONSE IS OBTAINED
C        BY SUBROUTINE FREQZ1. IT IS AN APPROXIMATED VALUE, AND THE
C        ACCURACY DEPENDS ON THE PARAMETER SETUPS FOR AND THE WAY OF
C        USING SUBROUTINE FEEQZ1.
C     2. THE COMPUTATION TIME FOR THIS PROGRAM AT INDX=6,INDQ=5,NT1=8,
C        NT2=29, AND NTINCR=7 IS APPROXIMATELY 200 SEC FOR A FIFTH ORDER
C        FILTER WITH ALL THE COEFFICIENTS NOT EQUAL TO 0 OR 1.  IF NO
C        SIMULATIONS ARE INVOLVED, 12 SEC IS NEEDED UNDER THE SAME
C        CONDITIONS.
```

```
C
C
C
      COMMON/DFZI1/A(8),B(8),Y(256),X(256)
      COMMON/DFZI2/NA,NB,NN,NPRINT
      COMMON/DFZI8/YNT(256)
      COMMON/FLTGM1/INDQRL/DF2DR1/INDQCL,INDQIL
      COMMON/FLTGR1/DL2,IS
      COMMON/QNTZD1/EO,EOH,EOI,EOVER1,EOVER2
      COMMON/DFZDR4/VNYNT(256),VNX(256),KAYNT(256),KAX(256)
      DIMENSION ERMST(5),ERMSTO(5),PLINE(130),ERMSTB(5),ERMSTBO(5)
      DOUBLE PRECISION A,B,Y,X,EO,EOH,DL2,HY,VNYNT,VNX,EOI,
     *HR,HI,HX,EOVER1,EOVER2,YNT
      LOGICAL INDQRL,INDQCL,INDQIL,INDIL
      DATA P/1H*/
C
C
 1400 FORMAT(1H ,//)
 1432 FORMAT(1X,*THE ACTUAL OUTPUTS AT FINITE BITS $ */1X,(11D11.3))
 1486 FORMAT(1X,*THE MAXIMUM AMPLITUDE FREQUENCY RESPONSE FOR THE WHOLE
     *FILTER $ *,E13.5)
 1485 FORMAT(1X,*THE MAXIMUM AMPLITUDE FREQUENCY RESPONSE FOR THE DENOMI
     *NATOR FILTER $ *,E13.5)
 1641 FORMAT(1X///1X,*ERROR BOUNDS $ *//1X,*R $ *,2E13.5/1X,*C $ *,2E13.
     *5/1X,*I $ *,2E13.5/1X,*R AND C $ *,2E13.5/1X,*R C AND I $ *,2E13.5
     *///)
 1520 FORMAT(1X/1X,*THE INDICATOR FOR TYPE(S) OF INPUT(S) $ *,I4//1X,*OT
     *HER INPUT PARAMETER VLUES ARE $ *, 9I4,3F7.2 /4X,*RESPECTIVELY FOR
     * NA, NB, IERROR, IS, NT1, NT2, NTINCR, INDG, IPRNT, BMFY, STD2, ST
     *ART2.*///)
 1167 FORMAT(1X,*THE SQRE SUM OF THE IDEAL OUTPUT SIGNAL IS*,D17.5/1X,
     **THE INDICATORS FOR ROUNDOFF ERROR OPERATIONS, COEFFICIENT, AND IN
     *PUT QUANTIZATIONS ARE  INDQRL, INDQCL, AND INDQIL, RESPECTIVELY.*
     */1X,*THEIR VALUES FOLLOW THE NUMERICAL VALUES OF THE ACTUAL ERRORS
     *.*//)
 1003 FORMAT(1X,9F8.4)
 1000 FORMAT(1X,19I4)
 1001 FORMAT(1X,5D13.6)
 1601 FORMAT(1X///1X,*ESTIMATED ERRORS AND ACTUAL ERRORS AT*,I3,* BITS $
     **///1X,*ESTIMATED ERRORS $ *//1X,*R $ *,2E13.5/1X,*C $ *,2E13.5)
 1603 FORMAT(1X,*I $ *,2E13.5/1X,*R AND C $ *,2E13.5/1X,*R C AND I $ *,
     *2E13.5///)
 1405 FORMAT(1X,*R C AND I $ *,3E13.5,4X,*INDICATOR VALUES $ *,3L3,
     *6X,*EST. ER./ACT. $ *,3X,F5.2)
 1404 FORMAT(1X,*R AND C $ *,3E13.5, 6X,*INDICATOR VALUES $ *,3L3,
     *6X,*EST. ER./ACT. $ *,3X,F5.2)
 1403 FORMAT(1X,*R $*,3E13.5,13X,*INDICATOR VALUES $ *,3L3,
     *6X,*EST. ER./ACT. $ *,3X,F5.2)
 1402 FORMAT(1X,*I $*,3E13.5,13X,*INDICATOR VALUES $ *,3L3,
     *6X,*EST. ER./ACT. $ *,3X,F5.2)
 1401 FORMAT(1X,*C $*,3E13.5,13X,*INDICATOR VALUES $ *,3L3,
     *6X,*EST. ER./ACT. $ *,3X,F5.2)
 1010 FORMAT(1H1,*AMPLITUDE (MAX VALUE) $ *,E13.5/1X,*FREQUENCIES FOR SI
     *NUSOIDAL INPUTS  $ *,E13.5,* (WI)  *,E13.5,* (WJ)  */1X,*STARTING
     *VALUE AND STANDARD DEVIATION FOR NOISE INPUT GENERATN. $ *,2E13.5)
 1011 FORMAT(1X,*NUMBER OF INPUT OR OUTPUT POINTS $  *   ,
     *I6//1X,*THE FILTERING COEFFICIENTS ARE $ */)
 1510 FORMAT(1X,130A1,I2)
 1501 FORMAT(1X,*INPUTS ARE  AMP COS(10 WJ WI J).*)
 1502 FORMAT(1X,*INPUTS ARE  AMP COS(WJ/(10 *I) J).*)
 1503 FORMAT(1X,*INPUTS ARE  AMP COS(WJ J).*)
 1504 FORMAT(1X,*INPUTS ARE  AMP NOISE.*)
```

```
1505 FORMAT(1X,*INPUTS ARE  AMP NOISE COS(WJ J).*)
1506 FORMAT(1X,*INPUTS ARE  AMP COS(WJ J + NOISE).*)
1507 FORMAT(1X,*INPUTS ARE  AMP COS(NOISE J).*)
1508 FORMAT(1X,*INPUTS ARE  AMP (X(J) COS(WJ J)+YNT(J) COS(WJ J)).*)
1509 FORMAT(1X,*INPUTS ARE  AMP (X(J) COS(WI J)+YNT(J) COS(WJ J)).*)
1012 FORMAT(1X,5D14.6)
1702 FORMAT(1X,///)
C
     READ 1000,NA,NB,NN,IERROR,IS,NT1,NT2,NTINCR,INDQ,INDX,IPRNT
     READ 1003,AMP,WI,WJ,STD,BMFY,START,STD2,START2
     READ 1001,(A(J),J=1,NA)
     READ 1001,(B(J),J=1,NB)
C
     DO 49 I=1,130
  49 PLINE(I)=P
     IF(IPRNT) 530,531,532
 530 NPRINT=-1
     GO TO 533
 531 IPRNT=65
 532 NPRINT=NN/IPRNT+1
 533 IF(START.EQ.0.) START=0.5
     DL2=1.D0/DLOG(2.D0)
     IF(NT2.EQ.0) NT2=NT1
     IF(NTINCR.EQ.0) NTINCR=4
     IF(WI.EQ.0.) WI=1.
     IF(WJ.EQ.0. ) WJ=1.
     IF(STD.EQ.0.) STD=0.1
     IF(AMP.EQ.0.) AMP=1.
     IF(BMFY) 264,265,266
 264 BMFY=B(1)/1.2
 266 DO 53 J=1,NB
  53 B(J)=B(J)/BMFY
 265 CONTINUE
C
     PRINT 1010,AMP,WI,WJ,STD,START
     PRINT 1011,NN
     PRINT 1012,(A(J),J=1,NA)
     PRINT 1012,(B(J),J=1,NB)
     PRINT 1520,INDX,NA,NB,IERROR,IS,NT1,NT2,NTINCR,INDQ,IPRNT,BMFY,
    *STD2,START2
C
C    CALCULATING THE MAXIMUM VALUE OF THE DENOMINATOR FILTER
C
     PI=3.14159265358
     MPT=180
     WJST=0.
     WJL=2.*PI
     ID=1
C
     CALL FREQZ1 (ID,MPT,WJST,WJL,WJMAX,HMAX)
     PRINT 1485,HMAX
C
     ID=0
     WJL=2.*PI
     WJST=0.
     CALL FREQZ1 (ID,MPT,WJST,WJL,WJMAX,GMAX)
     PRINT 1486,GMAX
     HMAX=HMAX*HMAX
     GMAX=GMAX*GMAX
C
C
C    SET UP INPUTS TO THE ONE-DIMENSIONAL DIGITAL FILTER.
```

```
C
      IESTRP=-1
      IDINDX=INDX
      NINDX=6
      IF(IDINDX.LT.NINDX) GO TO 514
      INDX=-3
  514 PRINT 1400
      PRINT 1510,(PLINE(I),I=1,130),INDX
      IF(INDX) 220,221,221
  220 IF(INDX+2) 371,372,373
  373 DO 16 J=1,NN
   16 X(J)=AMP*COS(WJ*J)
      INDPRT=3
      GO TO 2
  372 WI=WJ*10.*WI
      INDPRT=1
  375 DO 71 J=1,NN
   71 X(J)=AMP*COS(WI*J)
      GO TO 2
  371 IF(IDINDX.GE.NINDX) WI=WI/(100.*WI**2)
      IF(IDINDX.LT.NINDX) WI=WJ/(10.*WI)
      INDPRT=2
      GO TO 375
  221 CALL NOISE(NN,STD,START)
      IF(INDX-1) 256,256,257
  256 DO 231 J=1,NN
  231 X(J)=AMP*X(J)
      INDPRT=4
      IF(INDX.EQ.0) GO TO 2
      DO 258 J=1,NN
  258 X(J)=X(J)*COS(WJ*J)
      INDPRT=5
      GO TO 2
  257 IF(INDX-3) 259,260,296
  259 DO 263 J=1,NN
  263 X(J)=AMP*COS(WJ*J+X(J))
      INDPRT=6
      GO TO 2
  260 DO 5264 J=1,NN
 5264 X(J)=AMP*COS(X(J)*J)
      INDPRT=7
      GO TO 2
  296 IF(INDX-5) 293,293,2
  293 DO 79 J=1,NN
   79 YNT(J)=X(J)
      STD=STD2
      IF(STD.LE.0.) STD=STD*0.5123
      START=START2
      IF(START.LE.0.) START=START*0.793
      INDPRT=8
      CALL NOISE (NN,STD,START)
      IF(INDX.EQ.5) GO TO 476
      WK=WJ
  477 DO 73 J=1,NN
   73 X(J)=AMP*(X(J)*COS(WK*J)+YNT(J)*COS(WJ*J))
      GO TO 2
  476 WK=WI
      IF(WK.EQ.WJ) WK=-1.*WJ
      INDPRT=9
      GO TO 477
    2 CONTINUE
      GO TO (501,502,503,504,505,506,507,508,509),INDPRT
```

```
  501 PRINT 1501
      GO TO 515
  502 PRINT 1502
      GO TO 515
  503 PRINT 1503
      GO TO 515
  504 PRINT 1504
      GO TO 515
  505 PRINT 1505
      GO TO 515
  506 PRINT 1506
      GO TO 515
  507 PRINT 1507
      GO TO 515
  508 PRINT 1508
      GO TO 515
  509 PRINT 1509
C
  515 PRINT 1510,(PLINE(I),I=1,130),INDX
C
      INDQAX=INDQ
      DO 10 I=NT1,NT2,NTINCR
      EO=2.D0**(-I)
      EOI=2.D0**I
      EOH=EO*0.5D0
      EOVER1=1.D0-EO
      EOVER2=-EOVER1
      IF(IS.NE.1) EOVER1=-1
      IF(I.GT.NT1) IESTRP=1
      CALL DF2I(IESTRP,ERMST,ERMSTO,HY,EO*EO,HO,HI,HX,GMAX,HMAX,
     *ERMSTB,ERMSTBO)
      PRINT 1511,I,((ERMST(II),ERMSTO(II)),II=1,2)
      PRINT 1603,((ERMST(II),ERMSTO(II)),II=3,5)
      IF(I.EQ.NT1) PRINT 1167,HY
      PRINT 1641,((ERMSTB(II),ERMSTBO(II)),II=1,5)
      INDIL=.F.
      IF (INDQ-4) 111,112,112
  111 INDQRL=.F.
      INDQCL=.F.
      INDQIL=.F.
      IF(INDQAX) 140,141,142
  142 INDQRL=.T.
      IF(INDQAX-2) 171,172,173
  171 ASSIGN 403 TO ISWITCH
      GO TO 144
  172 INDQCL=.T.
      ASSIGN 404 TO ISWITCH
      GO TO 144
  173 INDQCL=.T.
      INDQIL=.T.
      ASSIGN 405 TO ISWITCH
      GO TO 144
  141 INDQCL=.T.
      ASSIGN 401 TO ISWITCH
      GO TO 144
  140 INDQIL=.T.
      ASSIGN 402 TO ISWITCH
  144 CALL DF2DR (ERMAX,ERMS,ERMSRO,HY)
      INDIL=.T.
      GO TO ISWITCH,(401,402,403,404,405)
  401 RT1=ERMSTO(2)/ERMSRO
      PRINT 1401,ERMS,ERMSRO,ERMAX,INDQRL,INDQCL,INDQIL,RT1
```

```
      GO TO 165
  402 RT1=ERMSTO(3)/ERMSRO
      PRINT 1402,ERMS,ERMSRO,ERMAX,INDQRL,INDQCL,INDQIL,RT1
      GO TO 165
  403 RT1=ERMSTO(1)/ERMSRO
      PRINT 1403,ERMS,ERMSRO,ERMAX,INDQRL,INDQCL,INDQIL,RT1
      GO TO 165
  404 RT1=ERMSTO(4)/ERMSRO
      PRINT 1404,ERMS,ERMSRO,ERMAX,INDQRL,INDQCL,INDQIL,RT1
      GO TO 165
  405 RT1=ERMSTO(5)/ERMSRO
      PRINT 1405,ERMS,ERMSRO,ERMAX,INDQRL,INDQCL,INDQIL,RT1
      GO TO 165
  112 IF(INDQ.EQ.5) GO TO 167
      IAUX1=0
      ISTOP=2
      GO TO 168
  167 IAUX1=-1
      ISTOP=3
  168 IF(IAUX1.GE.ISTOP) GO TO 10
      IF(INDIL) IAUX1=IAUX1+1
      INDQAX=IAUX1
      GO TO 111
  165 IF(INDQ.GT.3) GO TO 168
      IF(NPRINT.LE.0) GO TO 543
      PRINT 1400
      PRINT 1432,(YNT(J),J=1,NN,NPRINT)
  543 PRINT 1702
   10 CONTINUE
C
      IF(IDINDX.LT.NINDX) STOP
      IF(INDX.EQ.(NINDX-1)) STOP
      INDX=INDX+1
      IESTRP=0
      GO TO 514
      END
```

```
      SUBROUTINE DF2IO (IESTRP,ERMST,ERMSTRO,HY,EOSQRE,HH,HI,HX,GMX,HMX,
     *ERMSTB,ERMSTBO)
C
C
C     PURPOSE :
C        TO CALCULATE (1) THE MEAN SQUARED ERROR ESTIMATIONS AND (2) THE
C     NORM ERROR BOUNDS FOR ALL SOURCES OF ERROR IN ONEDIMENSIONAL DIGITAL
C     FILTERS EMPLOYING FLOATING-POINT ARITHMETIC.
C
C
C     SUBROUTINES NEEDED :
C        FLTGR, QNTZD.
C
C
C     INPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS) :
C        IESTRP,HY,EOSQRE,HH,HI,HX,GMX,HMX,A,B,X,NA,NB,NN,NPRINT.
C
C     OUTPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS) :
C        ERMST,ERMSTRO,ERMSTB,ERMSTBO,HY,HH,HI,HX,Y.
C
C     DESCRIPTIONS OF INPUT AND OUTPUT ARGUEMENTS :
C
C     IESTRP=1 :  NO CALCULATIONS FOR HH,H,AND HX.
C     IESTRP=0 :  NO CALCULATIONS FOR HH AND H.
C     IESTRP=-1 :  CALCULATIONS FOR HH,H,AND HX.
C     ERMST(I) :  TO STORE THE ESTIMATED OUTPUT MEAN SQUARED ERROR.
C     HY :  THE OUTPUT SQUARE SUM.
C     ERMSTRO(I) :  ERMST(I)/HY.
C     EOSQRE :  EO*EO, WHERE EO IS THE QUANTIZATION STEP SIZE.
C     HH :  THE IMPULSE RESPONSE SQUARE SUM FOR THE DENOMINATOR FILTER.
C     HI :  THE IMPULSE RESPONSE SQUARE SUM FOR THE WHOLE FILTER.
C     HX :  THE INPUT SQUARE SUM.
C     GMX :  THE SQUARED VALUE OF THE MAXIMUM AMPLITUDE FREQUENCY RESPONSE
C           FOR THE WHOLE FILTER.
C     HMX :  THE SQUARED VALUE OF THE MAXIMUM AMPLITUDE FREQUENCY RESPONSE
C           FOR THE DENOMINATOR FILTER.
C     ERMSTB(I) :  TO STORE THE ESTIMATED NORM ERROR BOUND.
C     ERMSTBO(I) :  ERMSTB(I)/HY.
C     A(I),B(I),NA,NB :  STORING THE COEFFICIENTS OF THE DIGITAL FILTER,
C                        THAT IS,   (B(1)+B(2)*Z**(-1)+ ... +B(NB)*Z**(-NB
C                        +1))/(A(1)+A(2)*Z**(-1)+ ... +A(NA)*Z**(-NA+1))
C     NN :  NUMBER OF POINTS OF THE FILTERING OPERATIONS.
C     NPRINT >0 :  PRINT OUTPUTS FOR ONE OUT OF EVERY (NPRINT) POINTS OF
C                  THE INPUTS AND OUTPUTS OF THE FILTER.
C            OTHERWISE :  NO PRINT OUTPUTS FOR THE INPUTS AND OUTPUTS.
C     X(I),I=1,...,NN :  THE INPUTS OF THE DIGITAL FILTER.
C     Y(I),I=1,...,NN :  THE OUTPUTS OF THE DIGITAL FILT-R.
C
C
C
      COMMON/DF2I1/A(8),B(8),Y(256),X(256)
      COMMON/DF2I2/NA,NB,NN,NPRINT
      COMMON/DF2I8/YNT(256)
      DIMENSION YMA(8),YMB(8),ALPHA(8),BETA(8),QEA(8)
     *,ERMST(5),ERMSTRO(5),QEBO(8),ERMSTB(5),ERMSTBO(5)
      LOGICAL IAUXIL,INDAL
      DOUBLE PRECISION A,B,Y,X,YMA,YMB,V,HY,HH,ALPHA,BETA,QEA,HO,HB,
     *STORE,H,EOSQRE,AUX1,AUX2,FREE,HX,HBRNM,HR,YNT,HI,QEBO,HBNM
C
      IF(IESTRP) 191,191,192
  191 DO126 J=1,NN
```

```
  126 Y(J)=0.D0
      DO 81 J=1,NN
   81 YNT(J)=X(J)
      DO 24 J=1,NA
   24 YMA(J)=0.D0
      DO124 J=1,NB
  124 YMB(J)=0.D0
C
C     PREPARE TO FIND OUTPUT Y WITH REGULAR INPUT X.
C
      HX=0.D0
      H=0.D0
      INDGT=2
      INDXL=.T.
      IND=1
      IAUXIL=.F.
C
  647 DO 62  N=1,NN
      Y(N)=0.D0
      GO TO (644,700),INDGT
C
  700 DO964 J=1,NB
      LJ=N-J+1
      IF(LJ.LT.1) GO TO964
      V=X(LJ)*B(J)
      Y(N)=Y(N)+V
      GO TO (64,674),INDGT
  674 YMB(J)=YMB(J)+V*V
   64 CONTINUE
  964 CONTINUE
C
      GO TO 645
C
  644 IF(IAUXIL) GO TO 645
      IF(IND) 679,669,679
  679 Y(N)=1.D0
      GO TO 659
  669 IF(N.GT.(NB+1)) GO TO 659
      X(N)=0.D0
      IF(N.EQ.1) X(N)=1.D0
      GO TO 700
  659 IAUXIL=.T.
C
  645 DO 66 J=2,NA
      LJ=N-J+1
      IF(LJ.LT.1) GO TO66
      V=Y(LJ)*A(J)
      Y(N)=Y(N)-V
      GO TO (66,676),INDGT
  676 YMA(J)=YMA(J)+V*V
   66 CONTINUE
C
      H=H+Y(N)*Y(N)
      IF(INDXL) HX=HX+X(N)*X(N)
   62 CONTINUE
C
C
      IF(IND) 701,646,702
  701 HH=H
      IAUXIL=.F.
      IND=0
      INDGT=1
```

```
      H=0.D0
      PRINT 1775,HH
      IF(NPRINT) 861,861,862
  861 PRINT 1235,Y(1),Y(NN)
      GO TO 647
  862 PRINT 1636,(Y(J),J=1,NN,NPRINT)
C
C     GO BACK TO FIND THE IMPULSE RESPONSE FOR THE WHOLE FILTER.
C
      GO TO 647
C
  792 HY=H
C
      IF(IESTRP) 793,794,793
C
C     PREPARE TO GO BACK TO FIND THE IMPULSE RESPONSE FOR THE
C     DENOMINATOR FILTER.
C
  793 IND=-1
      INDGT=1
      H=0.D0
      INDXL=.F.
      IAUXIL=.F.
      DO 29 J=1,NN
   29 X(J)=Y(J)
      NXSTRE=NH+1
      DO 31 J=1,NXSTRE
   31 QEA(J)=X(J)
      GO TO 647
C
C
C     SET UP VARIOUS VALUES FOR ERROR ESTIMATIONS.
C
  646 HI=H
C
C*********************************************************************
C     THE FOLLOWING STATEMENTS SHOULD BE PROPERLY ADJUSTED WHEN ONE OR
C     MORE OF THE FILTER COEFFICIENTS IS ZERO OR ONE.
C*********************************************************************
  794 DO824 J=1,NB
      IF(J.EQ.1) BETA(J)=(NB)
      IF(J.NE.1) BETA(J)=(NB-J+1)
  824 CONTINUE
      DO 26 J=1,NA
      IF(J.EQ.2) ALPHA(J)=NA
      IF(J.NE.2) ALPHA(J)=(NA-J+2)
      IF(J.EQ.1) ALPHA(J)=0.
   26 CONTINUE
C*********************************************************************
      IF(IESTRP.GE.0) GO TO 799
      DO 35 J=1,NXSTRE
   35 X(J)=QEA(J)
  799 DO224 J=1,NB
  224 QEBO(J)=YMB(J)/3.D0*BETA(J)
      DO 226 J=1,NA
  226 QEA(J)=YMA(J)/3.D0*ALPHA(J)
  192 HR=0.
      DO 228J=1,NA
  228 HR=HR+QEA(J)
      HBRNM=0.D0
      DO 230 J=1,NB
  230 HBRNM=HBRNM+QEBO(J)
```

```
C
      HD=0.
      HBNM=0.D0
      DO 771 J=1,NB
      AUX1=B(J)
      CALL FLTGR(AUX1,FREE,KAUX1,.T.,.T.,AUX2)
      AUX1=((AUX1-AUX2)/AUX1)**2
  771 HBNM=HBNM+YMB(J)*AUX1
      DO 772 J=1,NA
      AUX1=A(J)
      CALL FLTGR(AUX1,FREE,KAUX1,.T.,.T.,AUX2)
      AUX1=((AUX1-AUX2)/AUX1)**2
  772 HD=HD+YMA(J)*AUX1
      IF(IESTRF) 548,547,539
  548 PRINT 1774,H
      IF(NPRINT) 865,865,866
  865 PRINT 1230,Y(1),Y(NN)
      GO TO 867
  866 PRINT 1648,(Y(J),J=1,NN,NPRINT)
  667 PRINT 1400
      PRINT 1400
      DO 39 J=1,NN
   39 Y(J)=X(J)
      DO 86 J=1,NN
   86 X(J)=YNT(J)
  547 IF(NPRINT) 539,539,869
  869 PRINT 1547,(Y(J),J=1,NN,NPRINT)
      PRINT 1400
      PRINT 1638,(A(J),J=1,NN,NPRINT)
  539 CONTINUE
C
C     ROUNDOFF ERRORS ESTIMAION
C
      ERMST(1)=(HR+HBRNM)*HH*EOSQRE
      ERMSTRO(1)=ERMST(1)/HY
      ERMST(1)=ERMST(1)/NN
C
C     ESTIMATION OF COEFFICIENT QUANTIZATION ERRORS
C
      ERMST(2)=(HD+HBNM)*HH
      ERMSTRO(2)=ERMST(2)/HY
      ERMST(2)=ERMST(2)/NN
C
C     ESTIMATION OF INPUT QUANTIZATION ERRORS
C
      ERMST(3)=HX*H1*EOSQRE/3.D0
      ERMSTRO(3)=ERMST(3)/HY
      ERMST(3)=ERMST(3)/NN
C
      ERMST(4)=ERMST(1)+ERMST(2)
      ERMSTRO(4)=ERMSTRO(1)+ERMSTRO(2)
      ERMST(5)=ERMST(4)+ERMST(3)
      ERMSTRO(5)=ERMSTRO(4)+ERMSTRO(3)
C
C
C     ESTIMATION OF NORM ERROR BOUNDS
C
      ERMSTB(1)=(HR+HBRNM)*HMX*EOSQRE
      ERMSTB(2)=(HD+HBNM)*HMX
      ERMSTB(3)=HX*GMX*EOSQRE/3.D0
      ERMSTBO(1)=ERMSTB(1)/HY
      ERMSTBO(2)=ERMSTB(2)/HY
```

```
      ERMSTBO(3)=ERMSTB(3)/NY
      ERMSTB(1)=ERMSTB(1)/NN
      ERMSTB(2)=ERMSTB(2)/NN
      ERMSTB(3)=ERMSTB(3)/NN
      ERMSTBO(4)=ERMSTBO(1)+ERMSTBO(2)
      ERMSTB(5)=ERMSTB(4)+ERMSTB(3)
      ERMSTB(4)=ERMSTB(1)+ERMSTB(2)
      ERMSTBO(5)=ERMSTBO(4)+ERMSTBO(3)
C
 1638 FORMAT(1X,11D11.3)
 1773 FORMAT(1X//1X,*THE IMPULSE REWPONSE SQRE SUMS OF THE*,
     ** DENOMINATOR FILTER :  *,D17.10)
 1230 FORMAT(1X,*THE FIRST AND THE LAST VALUES OF THE RESPONSES ARE *,
     *2D14.5,* RESPECTIVELY.*)
 1774 FORMAT(1X//1X,*THE IMPULSE RESPONSE SQRE SUM OF THE*,
     ** WHOLE FILTER :  *,12X,D17.10)
 1400 FORMAT(1H ,/)
 1547 FORMAT(1X,*THE REGULAR OUTPUTS AND INPUTS : *//(1X,11D11.3))
C
      RETURN
      END
```

```
      SUBROUTINE DF2DRO (ERMAX,ERMS,ERMSRO,HY)
C
C
C     PURPOSE :
C        TO SIMULATE THE OPERATIONS OF ONE-DIMENSIONAL DIGITAL
C     FILTER EMPLOYING FLOATING POINT ARITHEMATIC.
C
C
C     SUBROUTINES NEEDED :
C        FLTGA,FLTGR,FLTGM,QNTZD.
C
C
C     INPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS) :
C        HY,A,B,Y,X,EO,EOH,EOI,EOVER1,DL2,IS,EOVER2,INDQRL,INDQIL,INDQCL.
C
C     OUTPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS) :
C        ERMAX,ERMS,ERMSRO,YNT.
C
C     DESCRIPTIONS OF INPUT AND OUTPUT ARGUEMENTS :
C
C     HY,A,B,Y,X,NA,NB,NN :  PLEASE SEE SUBROUTINE DF210.
C     EO,EOH,EOI,EOVER1,EOVER2,DL2,IS :  PLEASE SEE SUBROUTINE FLTGR.
C     INDQRL=.T. :  SIMULATIONS INCLUDING ROUNDOFFS.
C     INDQCL=.T. :  SIMULATIONS INCLUDING COEFFICIENT QUANTIZATIONS.
C     INDQIL=.T. :  SIMULATIONS INCLUDING INPUT QUANTIZATIONS.
C     ERMAX :  THE MAXIMUM ACTUAL ERROR.
C     ERMS :  THE ACTUAL MEAN SQUARED ERROR.
C     ERMSRO :  ERMS/HY.
C     YNT(I),I=1,...,NN :  THE ACTUAL OUTPUTS OF THE DIGITAL FILTER.
C
C
C
      COMMON/QNTZD1/EO,EOH,EOI,EOVER1,EOVER2
      COMMON/FLTGR1/DL2,IS
      COMMON/DF211/A(8),B(8),Y(256),X(256)
      COMMON/DF218/YNT(256)
      COMMON/DF212/NA,NB,NN,NPRINT
      COMMON/FLTGM1/INDQRL/DF2DR1/INDQCL,INDQIL
      COMMON/DF2DR4/VNYNT(256),VNX(256),KAYNT(256),KAX(256)
      DIMENSION AA(8),BB(8),KA(8),KB(8)
      LOGICAL INDQRL,INDQCL,INDQIL,KAUXIL
      DOUBLE PRECISION A,B,Y,X,YNT,AA,BB,EO,EOH,DL2,HY,FREE,VNYNT,VNX,
     *STORE,AUX1,AUX2,AUXY,EOI,EOVER1,EOVER2
C
      ERMS=0.
      ERMAX=0.
C
      KAUXIL=INDQCL
      DO 32 J=1,NA
      AUX1=A(J)
   32 CALL FLTGR (AUX1,AA(J),KA(J),KAUXIL,.F.,FREE)
      DO 30 J=1,NB
      AUX1=B(J)
   30 CALL FLTGR (AUX1,BB(J),KB(J),KAUXIL,.F.,FREE)
C
C
C
      DO 24 J=1,NN
      VNYNT(J)=0.D0
   24 KAYNT(J)=0
      KAUXIL=INDQIL
```

```
      DO 26 I=1,NN
      AUX1=X(I)
      CALL FLTGM(AUX1,AUX2,KAUX2,KAUX1L,.F.,FREE)
      VNX(I)=AUX2
   26 KAX(I)=KAUX2
C
      DO 62 N=1,NN
      AUXY=0.D0
      KAUXY=0
C
      NBB=NB
      IF(N.LT.NB) NBB=N
      DO 64 J=1,NBB
      LJ=N-J+1
      AUX1=VNX(LJ)
      KAUX1=KAX(LJ)
      CALL FLTGM (AUX2,KAUX2,BB(J),KB(J),AUX1,KAUX1)
      CALL FLTGA (AUXY,KAUXY,AUX2,KAUX2,.T.)
   64 CONTINUE
C
      JA=2
      NBB=NA
      IF(N.LT.NA)NBB=N
      IF(NBB.LT.JA) GO TO966
      DO 66 J=JA,NBB
      LJ=N-J+1
      AUX1=VNYNT(LJ)
      KAUX1=KAYNT(LJ)
      CALL FLTGM (AUX2,KAUX2,AA(J),KA(J),AUX1,KAUX1)
      CALL FLTGA (AUXY,KAUXY,AUX2,KAUX2,.F.)
   66 CONTINUE
  966 CONTINUE
C
      IF(AUXY) 400,401,400
  400 YNT(N)=AUXY*2.D0**KAUXY
      GO TO 402
  401 YNT(N)=0.D0
  402 VNYNT(N)=AUXY
      KAYNT(N)=KAUXY
      AUX1=Y(N)-YNT(N)
      ERMS=ERMS+AUX1*AUX1
      IF(DABS(AUX1).GE.ERMAX) ERMAX=AUX1
   62 CONTINUE
  962 CONTINUE
      ERMSRO=ERMS/HY
      ERMS=ERMS/NN
      RETURN
      END
```

```
      SUBROUTINE NOISE(N,STD,START)
C
C
C     PURPOSE :
C        TO GENERATE A NOISE SIGNAL WITH SUBROUTINE RANF.
C
C
C     INPUT ARGUEMENTS :
C        N,STD,START.
C
C     OUTPUT ARGUEMENT (IN COMMON STATEMENT) :
C        X.
C
C     DESCRIPTIONS OF INPUT AND OUTPUT ARGUEMENTS :
C
C     N :  NUMBER OF POINTS DESIRED.
C     STD :  STANDARD DEVIATION.
C     START :  STARTING VALUE FOR SIGNAL GENERATION.
C     X(I),I=1,...,N :  GENERATED SIGNAL.
C
C
      COMMON/DF2I1/A(8),B(8),Y(256),X(256)
      DOUBLE PRECISION A,B,X,Y
C
      PI=3.1415927
      IF(START) 1,2,1
    1 DUMB=RANF(START)
    2 DO3I=1,N
      X(I)=(-2.*ALOG(RANF(0.)))**.5
    3 X(I)=STD*X(I)*COS(2.*PI*RANF(0.))
      RETURN
      END
```

```
      SUBROUTINE FREQZ1 (ID,MPT,WJST,WJL,WJMAX,AMAX)
C
C
C     PURPOSE :
C        TO CALCULATE THE FREQUENCY RESPONSE (AMPLITUDE) OF A
C     ONE-DIMENSIONAL DIGITAL FILTER.
C
C
C     NOTE :
C        THIS SUBROUTINE CAN BE MODIFIED TO DO THE SAME JOB FOR
C     TWO-DIMENSIONAL DIGITAL FILTER.
C
C
C     SUBPROGRAM NEEDED:
C        CPV.
C
C
C     INPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS) :
C        ID,MPT,WJST,WJL,A,B,NA,NB.
C     OUTPUT ARGUEMENTS (INCLUDING THOSE IN COMMON STATEMENTS) :
C        WJST,WJL,WJMAX,AMAX,Y.
C
C     DESCRIPTIONS OF INPUT AND OUTPUT ARGUEMENTS :
C
C     ID=1 :  CALCULATIONS OF FREQUENCY RESPONSE ARE FOR THE WHOLE FILTER.
C     ID=0 :  CALCULATTONS ARE FOR THE DENOMINATOR FILTER.
C     WJL (AS INPUT) :  LENGTH OF FREQUENCY RANGE.
C     MPT, WJST (AS INPUT) :  THE CALCULATIONS WILL BE O  THE FOLLOWING
C        FREQUENCY POINTS,  WJST,WJST+WJINCR,WJST+2*WJINCR,...,
C        WJST+(MPT-1)*WJINCR, WHERE WJINCR=WJL/(MPT-1).
C     A,B,NA,NB : FILTER COEFFICIENTS. (PLEASE SEE SUBROUTINE DF2IO).
C     AMAX,WJMAX :  THE MAXIMUM VALUE OF THE AMPLITUDE FREQUENCY RESPONSE
C                   IS AMAX,  IT OCCURS AT THE FREQUENCY POINT WJMAX.
C     Y(I),I=1,...,MPT :  THE AMPLITUDE FREQUENCY RESPONSES.
C     WJST (AS OUTPUT) :  WJMAX-WJINCR.
C     WJL (AS OUTPUT) :  WJINCR*2.
C
C
C
      COMPLEX CPV,ZWJ,BZ(6),AZ(6),BZV,AZV
      COMMON/DF2I1/A(8),B(8),Y(256),X(256)
      COMMON/DF2I2/NA,NB,NN,NPRINT
      DOUBLE PRECISION A,B,Y,X
      LOGICAL IDNML
C
C     INITIAL SET UP.
C
      WJICR=WJL/(MPT-1)
      AMAX=0.
      IF(ID) 143,143,142
  143 DO 12 J=1,NB
      AUX1=B(J)
   12 BZ(J)=CMPLX(AUX1,0.)
      IDNML=.T.
      GO TO 144
  142 BZV=CMPLX(1.,0.)
      IDNML=.F.
  144 DO 11 J=1,NA
      AUX1=A(J)
   11 AZ(J)=CMPLX(AUX1,0.)
      ERAMG=10.**(-6)
```

```
C
C     STARTING TO FIND THE FREQUENCY RESPONSE.
C
      DO 1 M=1,MPT
      WJ=WJST+WJICH*(M-1)
      ZWJ=CMPLX(COS(WJ),SIN(WJ))
C
      IF(IDNML) BZV=CPV(ZWJ,BZ,NB,0,NB)
      AZV=CPV(ZWJ,AZ,NA,0,NA)
C
      AMG=CABS(AZV)
      IF(AMG-ERAMG) 120,120,123
  120 AMG=10.**6
      GO TO 125
  123 AMG=CABS(BZV)/AMG
  125 Y(M)=AMG
      IF(AMAX.GE.AMG) GO TO 1
      AMAX=AMG
      WJMAX=WJ
    1 CONTINUE
C
      WJST=WJ-WJICH
      WJL=WJICH*2.
      RETURN
      END
```

```
      FUNCTION CPV (Z,AZ,NORDR1,NSTART,NTOTAL)
C
C
C     PURPOSE :
C        AN AUXILIARY PROGRAM FOR SUBROUTINE FPEQZ1.
C
C
C
      COMPLEX CPV,Z,AZ(NTOTAL)
C
      CPV=CMPLX(0.,0.)
      DO 24 I=1,NORDR1
   24 CPV=CPV*Z+AZ(NORDR1-I+1+NSTART)
      RETURN
      END
```

REFERENCES

[1]. Huang, T.S., "Stability of two-dimensional recursive filters," IEEE Trans. on Audio and Electroacoustics, Vol. AU-20, No. 2, June 1972.

[2]. Shanks, J.L., TREITEL, S., and Justice, J.H., "Stability and synthesis of two-dimensional recursive filters," IEEE Trans. on Audio and Electroacoustics, Vol. AU-20, No. 2, June 1972.

[3]. Shanks, J.L., "Two-dimensional recursive filters," in 1969 SWIEEECO Rec., pp. 19E1 - 19E8.

[4]. Sandberg, I.W., "Floating-point-roundoff accumulation in digital filter realizations," Bell Syst. Tech. J., Vol. 46, Oct. 1967, pp. 1775-1791.

[5]. Ni, M.D. and Aggarwal, J.K., "Two-dimensional digital filtering and its error analysis," IEEE Trans. on Computers, Vol. c-23, No. 9, September 1974.

[6]. Kan, E.P.F. and Aggarwal, J.K., "Error analysis of digital filter employing floating-point arithmetic," IEEE Trans. Circuit Theory, Vol. CT-18, No. 6, November 1971.

[7]. Kan, E.P.F. and Aggarwal, J.K., "Correction to 'Error analysis of digital filters employing floating-point arithmetic,'" IEEE Trans. Circuit Theory, Vol. CT-20, No. 3, September 1973.

[8]. Liu, B., "Effect of finite word length on the accuracy of digital filters - A review," IEEE Trans. Circuit Theory, Vol. CT-18, No. 6, November 1971.

[9]. Liu, B. and Kaneko, T., "Error analysis of digital filters realized with floating-point arithmetic," Proc. IEEE, Vol. 57, Oct. 1969, pp. 1735-1747.

[10]. Thajchayapong, P. and Rayner, P.J.W., "Recursive digital filter design by linear programming," IEEE Trans. Audio and Electroacoustics, Vol. AU-21, No. 2, April 1973.

[11]. Gold, B. and Rader, C.M., Digital Processing of Signals, New York: McGraw-Hill, 1969.

[12]. Jury, E.I., Theory and Application of the z-Transformation method, John Wiley & Sons, Inc., New York, 1964.

[13]. Oppenheim, A. V. and Weinstein, C. J., "Effects of finite register length in digital filtering and the fast Fourier transform", Proc. of IEEE, Vol. 60, No. 8, August 1972.

[14]. Oppenheim, A. V. and Weinstein, C. J., "A comparison of round-off noise in floating point and fixed point digital filter realization", Proc. IEEE, Vol. 57, pp. 1181-1183, June 1969.

[15]. Jackson, L. B., Kaiser, J. F., and McDonald, H. S., "An approach to the implementation of digital filters", IEEE Trans. on Audio and Electro-acoustics, Vol. AU-16, pp. 413-421, September 1968.

[16]. Jackson, L. B., "On the interaction of roundoff noise and dynamic range in digital filters", Bell Systems Technical Journal, Vol. 49, pp. 159-184, February 1970.

[17]. Jackson, L. B., "Roundoff-noise analysis for fixed-point digital filters realized in cascade or parallel form", IEEE Trans. Audio Electro-acoustics, Vol. AU-18, pp. 107-122, June 1970.

[18]. Jackson, L. B., "An analysis of limit cycles due to multiplication rounding in recursive digital (sub) filters", Proc. 7th Annual Allerton Conf. Circuit System Theory, pp. 69-78, 1969.

[19]. Ebert, P. M., Mazo, J. E., Taylor, M. G., "Overflow oscillations in digital filters", Bell System Journal, Vol. 48, pp. 2999-3020, November 1969.

[20]. Rader, C. M. and Gold, B., "Effects of parameter quantization on the poles of a digital filter", Proc. IEEE, Vol. 55, pp. 688-689, May 1967.

[21]. Farmer, C. H., and Gooden, D. S., "Rotation and stability of a recursive digital filter", Proc. of Two Dimensional Digital Signal Processing Conference, October 1971, Columbia, Missouri.

[22]. Hall, E. L., "Comparison of computations for spatial frequency filtering", Proc. IEEE, Vol. 60, No. 7, pp. 887-891, July 1972.

[23]. Hunt, B. R., "Computational considerations in digital image enhancement", Proc. of Two Dimensional Digital Signal Processing Conf., October 1971, Columbia, Missouri.

[24]. Hall, E. L. and Kahveci, A., "High resolution image enhancement techniques", Proc. of Two Dimensional Digital Signal Processing Conf., October 1971, Columbia, Missouri.

[25]. Bednar, J. B. and Farmer, C., "Stability of spatial digital filters", Mathematical Biosciences, Vol. 14, pp. 113-119, 1972.

[26]. Aggarwal, J. K., "Input quantization and arithmetic roundoff in digital filters - A review", Network and Signal Theory, Peter Peregrinus Ltd., pp. 315-343, September 1972.

[27]. Jackson, L. B., Kaiser, J. F., and McDonald, H. S., "An approach to the implementation of digital filters", IEEE Trans. on Audio and Electroacoustics, Vol. AU-16, pp. 413-421, September 1968.

[28]. Kaplan, W., "Introduction to Analytic Functions", Addison-Wesley Publishing Company, Chapter 9, p. 165, 1966.

[29]. Kuo, B. C., "Discrete-Data Control Systems", Prentice-Hall, Chapter 3, p. 53, 1970.

DISTRIBUTION LIST*

Current AFOSR Contract F4460-71-C-0091

Joint Services Electronics Program Distribution List dated 18 August 1975

DEPARTMENT OF DEFENSE

Chief, R & D Division (340)
Defense Communications Agency
Washington, D. C. 20301

Defense Documentation Center (12)
ATTN: DDC-TCA (Mrs. V. Caponio)
Cameron Station
Alexandria, Virginia 22314

Dr. A. D. Schnitzler
Institute for Defense Analyses
Science and Technology Division
400 Army-Navy Drive
Arlington, Virginia 22202

Dr. George H. Heilmeier
Office of Director of Defense
Research and Engineering
The Pentagon
Washington, D. C. 20315

Director, National Security Agency
Fort George G. Meade, Maryland 20755
ATTN: Dr. T. J. Beahn

DEPARTMENT OF THE AIR FORCE

HQ/USAF (AF/RDPE)
Washington, D. C. 20330

HQ USAF/RDPS
Washington, D. C. 20330

Rome Air Development Center
ATTN: Documents Library (TILD)
Griffiss AFB, New York 13440

Mr. H. E. Webb, Jr. (ISCP)
Rome Air Development Center
Griffiss AFB, New York 13440

AFSC (CCJ/Mr. Irving R. Mirman)
Andrews AFB
Washington, D. C. 20334

Directorate of Electronics & Weapons
HQ AFSC/DLC
Andrews AFB, Maryland 20334

Directorate of Science
HQ/AFSC/DLS
Andrews AFB, Washington, D. C. 20331

Mr. Carl Sletten
AFCLR/LZ
L. G. Hanscom Fld. Bedford, MA 01730

Dr. Richard Picard
AFCLR/OP
L. G. Hanscom Fld. Bedford, MA 01730

LTC J. W. Gregory (5)
AF Member, TAC
Air Force Office of Scientific Research
1400 Wilson Blvd.
Arlington, Virginia 22209

Mr. Robert Barrett
AFCLR/LQ
L. G. Hanscom Fld. Bedford, MA 01730

Dr. John N. Howard
AFCRL (CA)
L. G. Hanscom Field
Bedford, Massachusetts 01730

HQ ESD (DRI/Stop 22)
L. G. Hanscom Field
Bedford, Massachusetts 01730

Professor R. E. Fontana
Head Dept of Electrical Engineering
AF IT/ENE
Wright-Patterson AFB, Ohio 45433

AFAL/TE, Dr. W. C. Eppers, Jr.
Chief, Electronics Technology Division
Air Force Avionics Laboratory
Wright-Patterson AFB, Ohio 45433

AF Avionics Lab/CA
ATTN: Dr. Robert J. Doran
Acting Chief Scientist
AF Avionics Laboratory
Wright-Patterson AFB Ohio 45433

AFAL/TEA (Mr. R. D. Larson)
Wright-Patterson AFB, Ohio 45433

Faculty Secretariat (DFSS)
US Air Force Academy
Colorado 80840

Howard H. Steenbergen
Chief, Microelectronics Development
& Utilization Group/TE
Air Force Avionics Laboratory
Wright-Patterson AFB, Ohio 45433

Dr. Richard B. Mack
Physicist
Radiation and Reflection Branch (LZR)
Air Force Cambridge Research Laboratories
L. G. Hanscom Field, Bedford, MA 01730

Charles S. Sahagian
Chief, Preparation and Growth Branch (LQ)
Air Force Cambridge Laboratories
L. G. Hanscom Field, Bedford, MA 01730

Major William Patterson
Assistant Chief, Information Processing Branch (ISI)
Rome Air Development Center
Griffiss AFB, N Y 13441

LTC Richard J. Gowen
Professor and Deputy Department Head
Dept. of Electrical Engineering
USAF Academy, Colorado 80840

Director, USAF Project RAND
Via: Air Force Liaison Office
The RAND Corporation
ATTN: Library D
1700 Main Street
Santa Monica, California 90406

AUL/LSE-9663
Maxwell AFB, Alabama 36112

AFETR Technical Library
P. O. Box 4608, MU 5650
Patrick AFB, Florida 32925

ADTC (SSLT)
Eglin AFB, Florida 32542

HQ AMD (RDR/Col Godden)
Brooks AFB, Texas 78235

USAFSAM (RAT)
Brooks AFB, Texas 78235

Commander (2)
ATTN: STEWS-AD-L, Technical Library
White Sands Missile Range, New Mexico 88002

USAF European Office of Aerospace Research
Technical Information Office
Box 14, FPO New York 09510

VELA Seismological Center
312 Montgomery Street
Alexandria, Virginia 22314

Dr. Carl E. Baum
AFWL (ES)
Kirtland AFB, New Mexico 87117

Hqs Elct Sys Division (AFSC)
ATTN: ESD/MCIT/Stop 36
Mr. John Mott/Smith
Laurence G. Hanscom Field,
Bedford, Mass 01730

USAFSAM/RAL
Brooks AFB, Texas 78235

Paul M. Kalaghan, Supervisor
Prog. Div., Geophy. Dept.
Smithsonian Institution
60 Garden Street
Cambridge, Mass. 02138

DEPARTMENT OF THE ARMY

HQDA (DARD-ARS-P)
Washington, DC 20310

Commander
US Army Security Agency
ATTN: IARD-T
Arlington Hall Station
Arlington, Virginia 22212

HQ Army Materiel Command
Technical Library Rm 7B 35
5001 Eisenhower Avenue
Alexandria, Virginia 22304

Commander (AMXRD-BAD)
US Army Ballistics Research Laboratory
Aberdeen Proving Ground
Aberdeen, Maryland 21005

Commander
Picatinny Arsenal
Dover, N J 07701
ATTN: Science & Tech Info Br
SMUPA-TS-T-S

Dr. Hermann Robl
US Army Research Office
P. O. Box 12211
Research Triangle Park
North Carolina 27709
6

Richard O. Ulsh (CRDARD-IP)
US Army Research Office
P. O. Box 12211
Research Triangle Park
North Carolina 27709 16

Mr. George C. White, Jr.
Deputy Director, L1000, 64-4
Pitman-Dunn Laboratory
Frankford Arsenal
Philadelphia, Pennsylvania 19137

Redstone Scientific Information Center
ATTN: Chief, Document Section
US Army Missile Command
Redstone Arsenal, Alabama 35809

Commander
US Army Missile Command
ATTN: AMSMI-RR
Redstone Arsenal, Alabama 35809

COL Robert W. Noce
Senior Standardization Representative
US Army Standardization Group, Canada
Canadian Force Headquarters
Ottawa, Ontario, Canada KIA OK2

Dr. Homer F. Priest
Chief, Materials Sciences Division, Bldg. 292
Army Materials and Mechanics Research Center
Watertown, Massachusetts 02172

John E. Rosenberg
Harry Diamond Laboratories
Connecticut Ave & Van Ness Street N. W.
Washington, DC 20438

Commandant
US Army Air Defense School
ATTN: ATSAD-T-CSM
Fort Bliss, Texas 79916

*The Joint Services Technical Advisory Committee has established this list for the regular distribution of reports on the electronics research program of the University of Texas at Austin. Additional addresses may be included on their written request to:

Mr. I. A. Balton (AMSEL-TL-DC)
Executive Secretary, TAC/JSEP
US Army Electronics Command
Fort Monmouth, New Jersey 07703

An appropriate endorsement by a Department of Defense sponsor is required except on request from a Federal Agency.

Commandant
US Army Command and General Staff College
ATTN: Acquisitions, Lib Div
Fort Leavenworth, Kansas 66027

Dr. Hans K. Ziegler (AMSEL-TL-D)
Army Member, TAC/JSEP
US Army Electronics Command
Fort Monmouth, New Jersey 07703

Mr. I. A. Balton, (AMSEL-TL-DC) (5)
Executive Secretary, TAC/JSEP
US Army Electronics Command
Fort Monmouth, New Jersey 07703

Mr. A. D. Bedrosian, Rm 26-131
US Army Scientific Liaison Office
Mass Institute of Technology
77 Massachusetts Avenue
Cambridge, Massachusetts 02139

Director (NV-D)
Night Vision Laboratory, USAECOM
Fort Belvoir, Virginia 22060

Commander/Director
Atmospheric Sciences Laboratory
ATTN: AMSEL-BL-DD
White Sands Missile Range, New Mexico 88002

Atmospheric Sciences Laboratory
US Army Electronics Command
ATTN: AMSEL-BL-RA (Dr. Holt)
White Sands Missile Range, New Mexico 88002

Chief, Missile EW Tech Area
Electronic Warfare Laboratory, ECOM
ATTN: AMSEL-WL-MY
White Sands Missile Range, New Mexico 88002

US Army Armaments
ATTN: AMSAR-RD
Rock Island, Illinois 61201

US Army ABMDA
(ATTN: RDMD-NC, Mr. Gold)
1300 Wilson Blvd.
Arlington, VA 22209

Harry C. Holloway, M. D. Col, MC
Director, DIV of Neuropsychiatry
Walter Reed Army Institute of Research
Washington, DC 20012

Commander, USASATCOM
AMCPM-SC
Fort Monmouth, New Jersey 07703

Director, TRI-TAC
ATTN: TT-AD (Mrs. Briller)
Fort Monmouth, N. J. 07703

Commander
US Army R & D Group (Far East)
APO, San Francisco, California 96343

Commander, US Army Communications Command
ATTN: Director, Advanced Concepts Office
Fort Huachuca, AZ 85613

Project Manager, ARTADS
(AMCPM-TDS)
EAI Building
West Long Branch, NJ 07764

US Army White Sands Missile Range
STEWS-ID-R (ATTN: Dr. Alton L. Gilbert)
White Sands Missile Range, NM 88002

Mr. William T. Kawai
US Army R & D Group (FAR EAST)
APO, San Francisco, California 96343

Commander
US Army Electronics Command
Fort Monmouth, New Jersey 07703

ATTN: AMSEL-RD-O (Dr. W. S. McAfee)
CT-L (Dr. G. Buser)
CT-LE (Dr. S. Epstein)
BL-FM-A
CT-D
CT-R
NL-O (Dr. H. S. Bennett)
NL-T (Mr. R. Kulinyi)
NL-C
NL-PB
NL-F
NL-M
TL-I
TL-B
VL-D
WL-D
TL-MM (Mr. Lipetz)
BL-FM (Dr. Edward Collett)
NL-O
NL-X
NL-H Schwering
NL-Y
TL-DR
TL-E (Dr. S. Kronenberg)
TL-E (Dr. J. Kohn)
TL-I (Dr. C. Thornton)
NL-B (Dr. S. Amoroso)

DEPARTMENT OF THE NAVY

Director, Electronic Programs
ATTN: Code 427
Office of Naval Research
800 North Quincy Street
Arlington, Virginia 22217

Director
Naval Research Laboratory
ATTN: Mr. A. Brodzinsky, Code 5200
Washington, DC 20390

Director
Naval Research Laboratory
ATTN: Library, Code 2629 (ONRL)
Washington, DC 20390

Dr. G. M. R. Winkler
Director, Time Service Division
US Naval Observatory
Washington, DC 20390

Naval Weapons Center
Technical Library (Code 753)
China Lake, California 93555

Director
Information Systems Program (437)
Office of Naval Research
Arlington, Virginia 22217

Director, Naval Research Lab (Code 6400)
4555 Overlook Avenue, S. W.
Washington, DC 20375

Director, Naval Research Laboratory (Code 6470)
4555 Overlook Avenue, S. W.
Washington, DC 20375

Dr. Leo Young (Code 5203)
Electronics Division
Naval Research Laboratory
Washington, DC 20375

Commander
Naval Training Equipment Center
Orlando, Florida 32813

Dr. A. L. Slafkosky
Scientific Advisor, Code AX
Hqs. US Marine Corps
Washington, DC 20380

US Naval Weapons Laboratory
Dahlgren, Virginia 22448

Commander
US Naval Ordnance Laboratory
Silver Spring, Maryland 29010
ATTN: Tech Library & Info Services Div.

Director
Office of Naval Research
Boston Branch
495 Summer Street
Boston, Massachusetts 02210

Commander
Naval Missile Center
ATTN: 5632.2, Technical Library
Point Mugu, California 93042

Commander
Naval Electronics Laboratory Center
ATTN: Library
San Diego, California 92152

Deputy Director and Chief Scientist
Office of Naval Research Branch Office
1030 East Green Street
Pasadena, California 91106

Superintendent
Naval Post Graduate School
Monterey, California 93940
ATTN: Library (Code 2124)

Officer in Charge, New London Lab
Naval Underwater Systems Center (TECH Library)
New London, Connecticut 06320

Commander
Naval Avionics Facility
ATTN: D/035 Technical Library
Indianapolis, Indiana 46241

Commander
Office of Naval Research Branch Office
536 South Clark Street
Chicago, Illinois 60605

Naval Air Development Center
ATTN: Technical Library
Johnsville
Warminster, Pennsylvania 18974

Naval Oceanographic Office
Technical Library (Code 1640)
Suitland, Maryland 20390

Naval Ship Research and Development Center
Central Library (Code L42 and L43)
Washington, DC 20007

OTHER GOVERNMENT AGENCIES

Mr. F. C. Schwenk, RD-T
National Aeronautics & Space Administration
Washington, DC 20546

Los Alamos Scientific Laboratory
ATTN: Reports Library
P O Box 1663
Los Alamos, New Mexico 87544

M. Zane Thornton
Deputy Director Institute for Computer Sciences & Technology
National Bureau of Standards
Washington, DC 20234

Director, Office of Postal Technology (R&D)
US Postal Service
11711 Parklawn Drive
Rockville, Maryland 20852

NASA Lewis Research Center
ATTN: Library
21000 Brookpark Road
Cleveland, Ohio 44135

Library -R51
Bureau of Standards
Acquisition
Boulder, Colorado 80302

MIT Lincoln Laboratory
ATTN: Library A-082
P. O. Box 73
Lexington, Massachusetts 02173

Dr. Jay Harris
Program Director, Devices and Waves Program
NSF
1800 G Street
Washington, DC 20550

Dr. Howard W. Etzel
Deputy Director, Div. of Materials Rsch
NSF
1800 G Street
Washington, DC 20550

Dr. Dean Mitchell
Program Director, Solid-State Physics
Division of Materials Research
National Science Foundation
1800 G Street
Washington, DC 20550

NON-GOVERNMENT AGENCIES

Director
Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Director
Microwave Research Institute
Polytechnic Institute of New York
Long Island Graduate Center, Route 110
Farmingdale, New York 11735

Mr. Jerome Fox, Research Coordinator
Polytechnic Institute of New York
333 Jay Street
Brooklyn, New York 11201

Director
Columbia Radiation Laboratory
Dept. of Physics
Columbia University
538 West 120th Street
New York, New York 10027

Director
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

Director
Stanford Electronics Laboratory
Stanford University
Stanford, California 94305

Director
Microwave Laboratory
Stanford University
Stanford, California 94305

Director
Electronics Research Laboratory
University of California
Berkeley, California 94720

Director
Electronics Sciences Laboratory
University of Southern California
Los Angeles, California 90007

Director
Electronics Research Center
The University of Texas at Austin
Engineering-Science Bldg. 112
Austin, Texas 78712

Director of Laboratories
Division of Engineering & Applied Physics
Harvard University
Pierce Hall
Cambridge, Massachusetts 02138