(NASA-TM-X-56038) A FORTRAN PROGRAM FOR THE
ANALYSIS OF LINEAR CONTINUOUS AND
SAMPLED-DATA SYSTEMS (NASA)

N76-18823

Unclas
G3/61 18568

NASA TM X-56038

A FORTRAN PROGRAM FOR THE ANALYSIS OF LINEAR CONTINUOUS AND SAMPLED-DATA SYSTEMS

John W. Edwards

January 1976

NASA high-number Technical Memorandums are issued to provide rapid transmittal of technical information from the researcher to the user. As such, they are not subject to the usual NASA review process.

NASA Dryden Flight Research Center
Edwards, California 93523

NOTICE

1. Report No.
TM X-56038

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
A FORTRAN PROGRAM FOR THE ANALYSIS OF LINEAR
CONTINUOUS AND SAMPLED-DATA SYSTEMS

5. Report Date
January 1976

6. Performing Organization Code

7. Author(s)
John W. Edwards

8. Performing Organization Report No

10. Work Unit No.
505-06-91

9. Performing Organization Name and Address
NASA Dryden Flight Research Center
P. O. Box 273
Edwards, California 93523

11. Contract or Grant No.

13. Type of Report and Period Covered
Technical Memorandum

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, D. C. 20546

14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A FORTRAN digital computer program which performs the general analysis of linearized control systems is described. State variable techniques are used to analyze continuous, discrete, and sampled-data systems. Analysis options include the calculation of system eigenvalues, transfer functions, root loci, root contours, frequency responses, power spectra, and transient responses for open- and closed-loop systems. A flexible data input format allows the user to define systems in a variety of representations. Data may be entered by inputting explicit data matrices or matrices constructed in user-written subroutines, by specifying transfer function block diagrams, or by using a combination of these methods.

ORIGINAL PAGE IS
OF POOR QUALITY

17. Key Words (Suggested by Author(s))
FORTRAN linear analysis program
Continuous and sampled data analysis
Control system analysis

18. Distribution Statement
Unclassified - Unlimited

19. Security Classif. (of this report)
Unclassified

20. Security Classif. (of this page)
Unclassified

*For sale by the National Technical Information Service, Springfield, Virginia 22151

CONTENTS

Pages ii – iv BLANK

A FORTRAN PROGRAM FOR THE ANALYSIS OF LINEAR CONTINUOUS AND SAMPLED-DATA SYSTEMS

John W. Edwards
Dryden Flight Research Center

INTRODUCTION

A FORTRAN digital computer program is described which analyzes linear continuous or sampled-data systems using state variable techniques. Open- and closed-loop systems may be analyzed using frequency response or transient response techniques. Root locus and root contour options are also available. Systems may be defined by inputting explicit data matrices, by constructing matrices in user written subroutines, or by specifying transfer function block diagrams. The program also allows the user to define a system using a combination of the above methods. For instance, the plant may be described as a set of coupled differential equations and its control system described by a block diagram. The program allows the user to analyze such a system without converting the subsystems into a common representation.

SYMBOLS

$A, B, C, H, G, F,$ $K1, K2, K3, K4, D$	matrices used to define system models
$G(\cdot)$	transfer function
I	identity matrix
m	fractional computational time delay for sampled-data systems

NX, NY, NU	dimensions of state, output, and input vectors
R, S	matrices used to define system interconnections for MIXED option
s, z, w	Laplace, z-, and w- transform variables
z, v	output and input vectors for block diagram portions of MIXED systems
x, y, u	state, output, and input vectors
t	time-sec.
T	sampled-data system sample period-sec.
ϵ	incremental time period-sec.
$\phi(t)$	state transition matrix
[illegible](t)	integral of $\phi(t)$
Subscripts:	
com	command signal
ext	external signal
n	time index for sampled-data systems
Superscripts:	
c	designates continuous subsystem of sampled-data system
d	designates discrete subsystem of sampled-data system
T	transpose of a matrix

PROGRAM DESCRIPTION

CONTROL is a FORTRAN digital program capable of performing general analysis of linearized control systems problems. It utilizes state variable matrix operations to find system eigenvalues, transfer functions, root contours, root loci, frequency responses, power spectra, and transient responses. Continuous, discrete, and sampled-data system analyses may be accomplished. The data input format is quite flexible, allowing data to be entered in matrix form, block diagram form, general parameter input form, or a combination of these forms. The analysis options available are listed in Table I.

The basic systems which CONTROL analyses are

$\dot{x} = Ax + Bu$ (plant equation) (1a)

$y = Hx + Fu$ (output equation) (1b)

$u = K1x + Du_{com}$ (control law) (1c)

for continuous systems and

$x_{n+1} = Ax_n + Bu_n$ (plant equation) (2a)

$y_n = Hx_n + Fu_n$ (output equation) (2b)

$u_n = K1x_n + Du_{com_n}$ (control law) (2c)

for discrete systems. The state vector, x, is NX dimensional; the output vector, y, is NY dimensional; and the input vector, u, is NU dimensional.

To allow user flexibility, additional matrix equations are used in the program for definition of feedback control laws, etc., and will be discussed later. In all cases, the system equations are reduced to the form of systems (1) or (2) before being analyzed.

CONTROL program analysis is performed in the CNTRLR subroutine which is diagrammed in figure 1. The use of each subroutine is briefly described below.

CARD - reads data cards which specify program options chosen

LOAD, MATRIX, CHANGE, CLASS - data input subroutines

LOAD - reads data input matrices

MATRIX - a user written subprogram which constructs system matrices from basic data

CHANGE - a user written subprogram which changes data already in the program. Used for parameter variation studies

CLASS - constructs system matrices from data describing a block diagram of transfer functions

SETUP - reduces input data matrices to the system equations (1) or (2)

EIGEN - determines system eigenvalues from the A matrix using QR-algorithm

ROOT - performs root loci analysis as a function of two independently incremented feedback gains

NMRATR - determines numerator zeroes of transfer functions defined by the NU inputs, u, and NY outputs, y.

FRQRSP - computes a frequency response at discrete frequencies for each transfer function determined by NMRATR

PSD - computes a power spectrum of the transfer functions determined by NMRATR assuming unity variance white noise excitation

THIST - computes a transient response of the system at discrete time points

INPUT - a user written subroutine which constructs the input vector to the transient response

The CONTROL program is on disc in the CDC CYBER 70 system and is called by the user with appropriate control cards. To utilize the full flexibility of CONTROL, the user may write several of the subroutines to perform data preparation for his specific problem. These subroutines are:

MAIN
MATRIX
CHANGE
INPUT

These subroutines may be compiled in SOURCE language. Note that they are not required. These subroutines are defined in the program on disc and are overridden by the user's source subroutines. If a specific output format is desired, any of the CONTROL subroutines may be modified by the user and compiled to override the disc versions. A brief description of the use of these routines is given below:

MAIN - Provides variable dimensioning capability. The size of all arrays used in CONTROL is declared here and passed through COMMON to all other subroutines. Thus the size of the system matrixes may be quickly and easily changed. The disc program is provided with:

MX = 15 = n + 1; n = dimension of maximum state

MY = 15 = dimension of maximum output vector

MY = 10 = dimension of maximum input vector

MATRIX - A user written subroutine which constructs the system matrices. For instance, MATRIX may read the nondimensional aircraft stability derivatives, perform axes transformations, dimensionalize the derivatives, and insert the proper numbers into the system matrixes. General MATRIX routines are available for the standard lateral-directional and longitudinal linearized equations of motion which are generally used at FRC.

CHANGE - A user written subroutine which changes specified elements in the system matrices (which are already constructed through MATRIX, LOAD, or CLASS). This routine allows the capability of doing parameter variation studies on a basic system configuration without having to reload the problem data for each variation.

INPUT - This user written routine constructs the input vector for a time history calculation. Thus the user may generate step, impulse, ramp, sinusoidal, random inputs as desired. A basic routine which simply reads 1 card for the input vector $\underline{u}$ is provided.

All of these routines have specified calling sequences which are given in Appendix 1. Each routine has specific COMMON and DIMENSION statements which are required and which the user will not change. The structure of the CONTROL deck is shown in figure 2. The job control language for CONTROL is given in Table II. All figures, tables and information required to set up data for the CONTROL program are gathered at the end of this document for easy reference by the user.

CONTINUOUS SYSTEM MODELS

The basic continuous system model which CONTROL analyzes is given by (1) and is repeated here:

$$\dot{x} = Ax + Bu$$
$$y = Hx + Fu$$
$$u = K1x + Du_{com}$$

CONTROL analyzes three configurations of continuous system models: open loop, closed loop, and root loci. These three configurations are under the control of SYSTEM. For ease in problem setup, the additional matrices D, K1, K2, K3, K4, C, and G are defined.

Open Loop (SYSTEM = 1)

$$C\dot{x} = Ax + Bu \tag{3a}$$
$$y = Hx + G\dot{x} + Fu \tag{3b}$$

Closed Loop (SYSTEM = 2)

$$C\dot{x} = Ax + Bu \tag{4a}$$
$$u = K1x + K2\dot{x} + Du_{com} \tag{4b}$$
$$y = Hx + G\dot{x} + Fu \tag{4c}$$

Subscript COM indicates a "command" signal. Note that the inclusion of the C matrix allows inertial cross-coupling between states. K2 and G allow for state derivative feedback and output. D allows for controller interconnections and feedforward gains.

CONTROL reduces the systems (3) and (4) with the substitutions

$$A \longleftarrow C^{-1}A + C^{-1}B\,(I - K2\,C^{-1}B)^{-1}(K1 + K2\,C^{-1}A)$$
$$B \longleftarrow C^{-1}B\,(I - K2\,C^{-1}B)^{-1}D$$
$$H \longleftarrow H + GC^{-1}A + (F + GC^{-1}B)(I - K2\,C^{-1}B)^{-1}D$$
$$F \longleftarrow (F + GC^{-1}B)\,(I - K2\,C^{-1}B)^{-1}D$$

to the system,

$$\dot{x} = Ax + Bu_{com} \qquad (5a)$$
$$y = Hx + Fu_{com} \qquad (5b)$$

Note that (5) is identical to the basic system (1) with u replaced by u_{COM}. Obviously, for the open-loop system D = I, K1 = K2 = 0, and $u = u_{COM}$.

Root Loci (SYSTEM = 3)

$$C\dot{x} = Ax + Bu \qquad (6a)$$
$$u = k_1(K1\,x + K2\,\dot{x}) + k_3(K3\,x + K4\,\dot{x}) \qquad (6b)$$

This root loci model allows root loci as a function of two independent feedback variables. The first feedback variable is defined by the matrices K1 and K2 (e.g., normal acceleration = $\frac{V}{g}(\dot{\alpha} - q)$). K3 and K4 define the second feedback variable (commonly K3 = K4 = 0). The condition codes N1, N2, GAIN1, and GAIN2 determine the locus gain and number of locus points as follows:

$$k_1 = r * \mathrm{GAIN1} \qquad r = \begin{cases} 0, 1, 2, 3, \ldots, N1; & N1 > 0 \\ 0, 1, 2, 4, \ldots, 2^{(|N1|-2)}; & N1 < 0 \end{cases}$$

$$k_3 = t * \mathrm{GAIN2} \qquad t = \begin{cases} 0, 1, 2, 3, \ldots, N2; & N1 > 0 \\ 0, 1, 2, 4, \ldots, 2^{(|N2|-2)}; & N1 < 0 \end{cases}$$

CONTROL computes the closed-loop system matrix as:

$$\dot{x} = A x$$

where

$$A \longleftarrow A' + B'\left(I - k_1 K' - k_3 K^2\right)^{-1}\left(k_1 K^3 + k_3 K^4\right)$$

$$A' \longleftarrow C^{-1} A$$

$$B' \longleftarrow C^{-1} B$$

$$K' \longleftarrow K2\, B'$$

$$K^2 \longleftarrow K4\, B'$$

$$K^3 \longleftarrow K1 + K2 A'$$

$$K^4 \longleftarrow K3 + K4 A'$$

The eigenvalues of A are then determined to find the system root locus. The zeroes corresponding to the first feedback variable are determined by forming the observation equation:

$$y = Hx + Fu$$

where

$$H = \begin{bmatrix} K1 + K2A' \\ \hdashline K3 + K4A' \end{bmatrix}$$

$$F = \begin{bmatrix} K2\, B' \\ \hdashline K4\, B' \end{bmatrix}$$

A subsequent call to NMRATR then determines the zeroes. NMRATR determines the zeroes of the transfer function specified by the first nonzero row of these derived H and F matrices. If PLOT $\neq 0$, an additional data card is required for each root locus specifying the maximum and minimum axis coordinates desired for the plot (see Table II).

The continuous system models given above are summarized in Table III(a).

MIXED Systems

This problem formulation can be used to model a system which is described by a combination of differential equations and Laplace transform blocks. An example of such a system is an aircraft-control system. The aircraft equations of motion are usually known as differential equations while the control system is often given in block diagram form. The MIXED option gives the user the capability of easily modeling such a system. This option, which is a great convenience in analyzing continuous systems, becomes indispensable in the analysis of sampled-data systems. There it is used to correctly connect the continuous and discrete subsystems and to discretize to the continuous subsystem at the proper times.

The MIXED option involves a two-step data input stream. In the first step, the plant equations of motion are loaded as an open-loop system. The second step loads the block diagram control system data (CLASS subroutine) and augments the open-loop plant matrices with the control system dynamics.

Finally, the two subsystems are coupled together. The open-loop, closed-loop, and root locus options are available with the MIXED option.

The CLASS subroutine may be used by itself to construct state space formulations of systems described entirely by block diagrams. The input format for the MIXED option will be given after the CLASS loading option has been discussed.

DATA DECK FORMAT

CONTROL problem definition is accomplished in CARD. Condition codes defining the analysis options chosen, data input mode, and data handling procedures are read in a namelist format (CODE). Each pass through CNTRLR (figure 1) defines a case.

The CONTROL program data deck structure is shown in figure 3. The title card, namelist CODE, output label card, input label card, system data, and transient response input data formats are given in Table IV.

The format for entering data using namelist is illustrated in figure 4. Column 1 of each card must be blank. Column 2 of card 1 begins the name of the namelist (CODE) which is followed by a blank. Variable names in the namelist must be right justified with respect to the "=" sign, variable values must be right justified with respect to the "," sign and be of the proper type (real, integer). The namelist data entry is closed by the code ",&END."

The analysis options chosen, data input format, and system structure are defined in the namelist, CODE. The variables which are defined by the namelist are:

Integer variables

READ, SYSTEM, OUTPUT, MIXED, DIGITL, FRPS, NUMERS, TRESP, NX, NY, NU, NXC, NUC, ZOH, N1, N2, CONTUR, MULTRT, MODEL, NSCALE, CMAT, NK2, FORM, IPT, IGO, SAV, IFLAG, READ3

Real variables

DELT, FINALT, IFREQ, FFREQ, DELFRQ, GAIN1, GAIN2, M

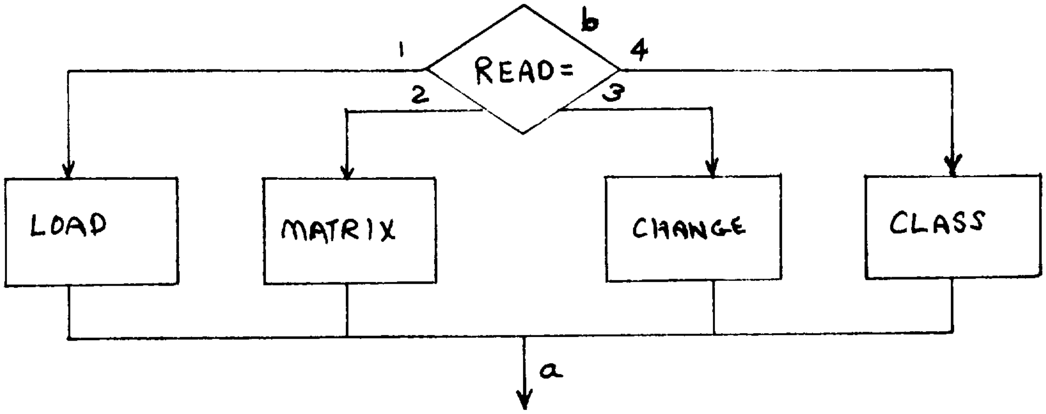
These variables are described in Appendix 2. All variables are initialized to zero at the start of each case (unless SAV = 1) and only nonzero variables need be defined in the namelist.

For root locus cases in which plots are requested, one card is required for each root locus by the plotter program to define the $j\omega$ scale on the plot. This card(s) follows the PLOT CARDS card.

Figure 1 indicates that CONTROL will return to the top of the program and look for more data at the end of each case. Thus data decks may be stacked together to analyze many cases on a single computer run.

DATA INPUT OPTIONS

The system data may be input in four different methods. The control loading option is determined by the condition code READ:



<u>CHANGE</u> (READ = 3) The CHANGE option is meant to be used on cases following an initial case. For instance, on case 1, the system matrices are loaded or constructed in LOAD, MATRIX, or CLASS. These "original" system matrices (as defined at point (a) on case 1) may be saved and made available at point (b) in case 2. If CHANGE is utilized in case 2, the "original" system matrices may be altered as desired. Thus a parameter variation study may be accomplished using a simple CHANGE subroutine.

The remaining three options--LOAD, MATRIX, and CLASS--will now be described.

<u>LOAD</u> (READ = 1) The subroutine LOAD reads explicit data matrices under the control of the condition codes. Each data matrix (e.g., A) is read row by row with an (8F10.4) format. Each data matrix must be preceded by a dimension card (2I10) giving the number of rows and columns in the matrix. For example, a 5 x 5 matrix would require 6 data cards (1 dimension card, 1 card per row) while an 8 x 9 matrix would require 19 data cards (1 dimension card, 2 cards per row). The matrices required are listed in STEP 1 of Table V.

MATRIX (READ = 2) MATRIX is a user written subroutine which constructs the system matrices required by the condition codes. These matrices are the same as those given in STEP 1 of Table V. Data defined in MATRIX are not destroyed by CONTROL. Thus the user may input data to the MATRIX subroutine for case 1 and reuse the data on subsequent cases. CLASS (READ = 4) If READ = 4 the system matrices are constructed in CLASS from block diagram input data. If MIXED = 1, the control system dynamics, in block diagram form, are added on to the plant dynamics by the CLASS subroutine. Thus the CLASS subroutine has a dual function in CONTROL. In both cases, the data format describing the block diagram is the same. The data format is given as STEP 2 of Table V. Two options of inputting the block diagram information are provided. The first option describes the block diagram by inputting the transfer function polynomial coefficients explicitly for general fourth-order transfer functions. The second option allows the user to pick standard form constant, first-order, or second-order blocks to describe the system. These blocks are defined by a small number of parameters and are given in Table VI. The first option is used only if a particular transfer function form cannot be found in Table VI. The first option will not be described. A description of the second option begins on page 21.

The data required to describe a system given entirely by block diagrams are summarized below. The additional data required if MIXED = 1 will be described later.

NBLOCK = No. of blocks in block diagram

GRAPH = Integer matrix describing interconnection of blocks

BLOCK = Integer matrix describing dimension of numerator and denominator polynomials of each block

NUMER = Matrix of polynomial coefficients of numerators of each block (ordered from $s^{\circ} \longrightarrow s^{m}$)

DENOM = Matrix of polynomial coefficients of denominators of each block (ordered from $s^{\circ} \longrightarrow s^{n}$)

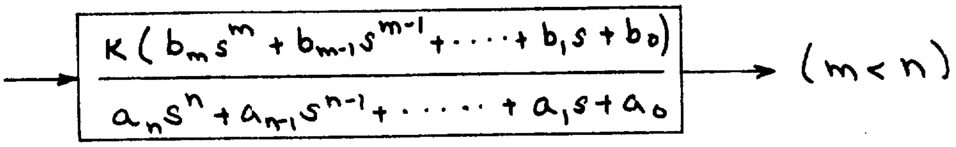
GAIN = Vector of gains of each block

ITHINY = Integer vector describing which block outputs are to be saved in final H matrix

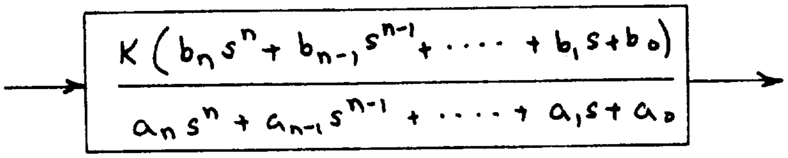
Allowable forms of blocks:

Form

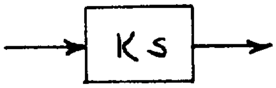
1



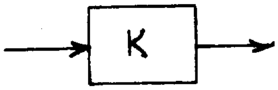
2



3



4



Restrictions:

a.) Inputs into a type 3.) block must be outputs of a type 1.) block.

b.) $n \leqslant 4$

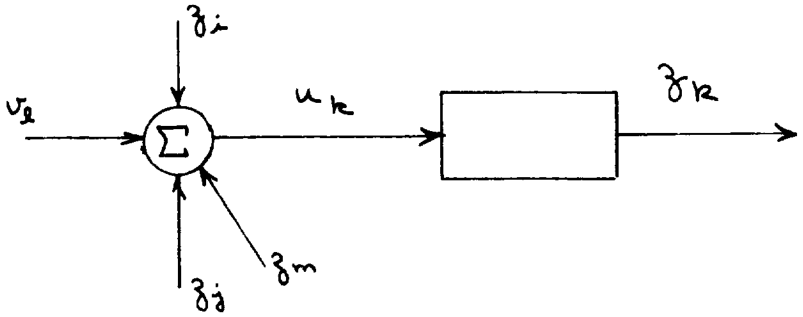
c.) Number of blocks $\leqslant$ 20

Interconnection of blocks:

The interconnection of blocks may be described with the concept of internal and external inputs to a block:

Internal input - an input which is an output of a block in the system.

External input - any input other than internal inputs, designated by v . The input to any block may be the sum of three or less internal inputs and one external input.

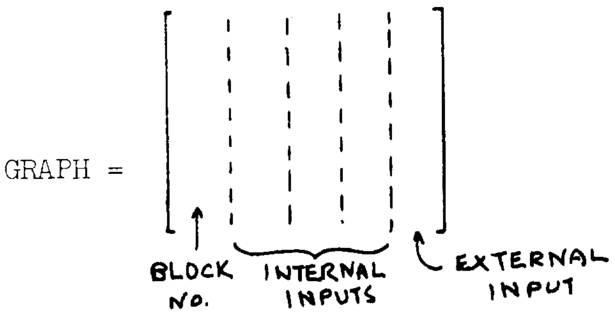


$$u_k = \pm v_l \pm z_i \pm z_j \pm z_m$$

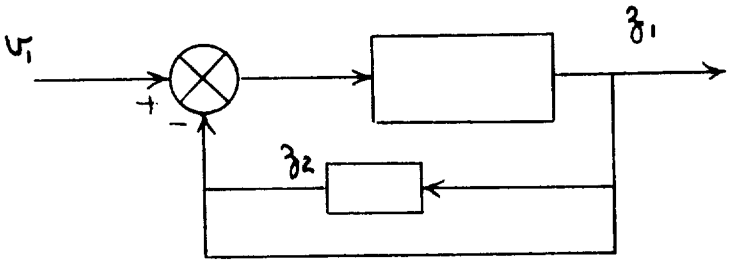
The ± indicates that either sign may be used on any of the inputs. Whenever an external input is defined, a summing junction is implied at the input to the block.

Problem formulation

1. Label the outputs of the blocks consecutively z_i, i = 1,2,3,....
2. Label the external inputs consecutively v_j , j = 1,2,3,...
3. Construct the GRAPH Matrix (dimension NBLOCK x 5) where NBLOCK = number of blocks in the diagram:



A simple example will illustrate the method:



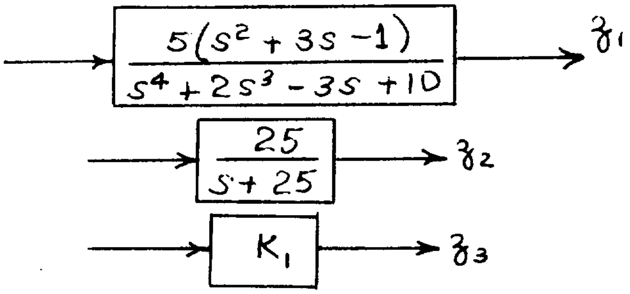
$$\text{GRAPH} = \begin{bmatrix} 1 & -2 & -1 & 0 & 1 \\ 2 & 1 & 0 & 0 & 0 \end{bmatrix}$$

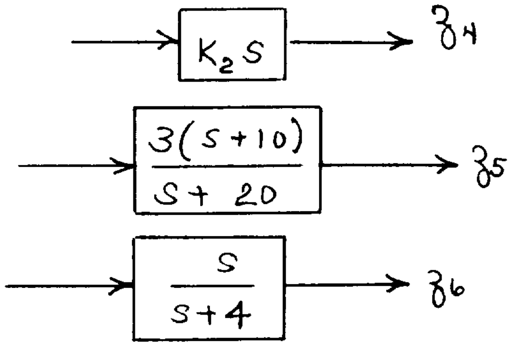
Note that negative inputs to summing junctions are indicated by the negative sign attached to the input label.

The following input data describe the internal composition of each block:

BLOCK - Describes the number of coefficients of the numerator and denominator polynomials which will be loaded (dimension NBLOCK x 3). The first column contains the block number (the sequence must correspond to the sequence of the first column of GRAPH). The second column contains the number of numerator coefficients to be loaded. The third column contains the number of denominator coefficients to be loaded.

example:





$$\text{BLOCK} = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 1 & 2 \\ 3 & 1 & 1 \\ 4 & 2 & 1 \\ 5 & 2 & 2 \\ 6 & 2 & 2 \end{bmatrix}$$

Note that a constant block K is treated as $K\frac{1}{1}$.

Also

$$\boxed{Ks} = \boxed{\frac{K(s+0)}{1}}$$

NUMER (NBLOCK x 5)

NUMER contains the numerator coefficients.

For the above example:

$$\text{NUMER} = \begin{bmatrix} -1. & 3. & 1. & 0 & 0 \\ 25. & 0 & 0 & 0 & 0 \\ 1. & 0 & 0 & 0 & 0 \\ 0 & 1. & 0 & 0 & 0 \\ 10. & 1. & 0 & 0 & 0 \\ 0 & 1. & 0 & 0 & 0 \end{bmatrix}$$

DENOM (NBLOCK x 5)

Denom contains the denominator coefficients.

For the above example:

$$\text{DENOM} = \begin{bmatrix} 10. & -3. & 0 & 2. & 1. \\ 25. & 1. & 0 & 0 & 0 \\ 1. & 0 & 0 & 0 & 0 \\ 1. & 0 & 0 & 0 & 0 \\ 20. & 1. & 0 & 0 & 0 \\ 4. & 1. & 0 & 0 & 0 \end{bmatrix}$$

GAIN (NBLOCK)

Gain is a vector containing the gain constants of the blocks.

For the above example:

$$\text{GAIN} = \begin{bmatrix} 5 & 1 & K_1 & K_2 & 3 & 1 \end{bmatrix}$$

ITHINY ($\leqslant$ NBLOCK)

In constructing the system matrices, CLASS constructs the output equation

$$y = Hx + Fu$$

where y is an NBLOCK vector. That is, the output of each block is defined as a component of the y vector. Ordinarily, the user will want to study only a few of these outputs. The ITHINY vector "thins" out the y vector. For the above example if;

$$\text{ITHINY} = \begin{bmatrix} 1 & 3 & 4 \end{bmatrix}$$

then $y^t = (z_1, z_3, z_4)^t$

The second option of defining a block diagram will now be described. This option allows the user to pick standard form transfer functions given in Table VI to describe the block diagram. Only one data card per block is required to describe the system. The option is chosen by setting NIT = 1 on the first card of STEP 2. The data required to specify each block are NUM, TYPE, CONNEC, MOD, PARAM where

NUM is the block number.

TYPE is the type of transfer function from Table VI.

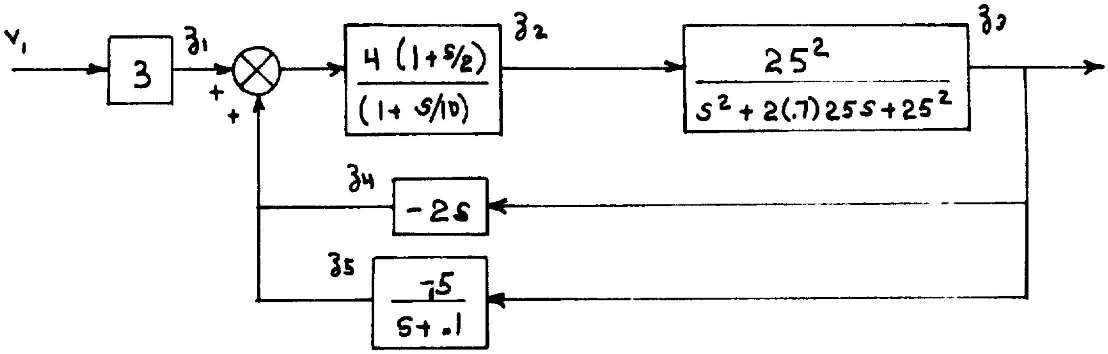
CONNEC specifies the connections between blocks and the external inputs. (This is the last four columns of GRAPH.)

MOD specifies whether the transfer function is an s, z, or w -transform.

PARAM gives the parameters defining the block as indicated in Table VI.

Following the cards giving the above data for each block, one card for ITHINY is read.

example: set up the input data required to describe the system



The data required are

```
5    1
1    1    0    0    0    1    0    3.
2    5    1    4    5    0    0    4.   10.   2.
3    8    2    0    0    0    0    1.   25.   .7
4    2    3    0    0    0    0   -2.
5    4    3    0    0    0    0   -5.    .1
1    2    3    4    5
```

CLASS constructs a state space representation of each individual block using the phase variable canonical form. For example, the first block of the previous example:

$$G(s) = \frac{5(s^2 + 3s - 1)}{s^4 + 2s^3 - 3s + 10}$$

is modeled as

$$\dot{x} = Ax + Bu$$
$$y = Hx + Fu$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -10 & 3 & 0 & -2 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$H = \begin{bmatrix} -5 & 15 & 5 & 0 \end{bmatrix} \qquad F = \begin{bmatrix} 0 \end{bmatrix}$$

CLASS constructs the state space representation of the entire block diagram in the following manner:

1. Write the output vector of the collection of blocks as

$$Cz = Hx + Fv$$
$$z = C^{-1}Hx + C^{-1}Fv$$

2. Write the state equation describing the internal structure of each block (in phase variables) ignoring all connections between blocks and defining the "internal" input

vector, $\underline{u}$

$$\dot{x} = Ax + Bu$$

3. Define the connection matrix, G, as

$$u = Gz$$

where

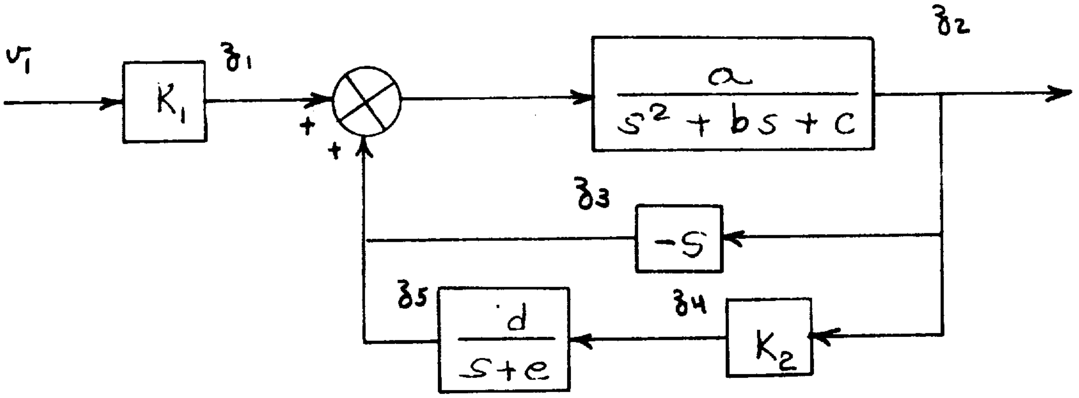
$$G(i, \text{GRAPH}(i,j)) = 1 \quad \text{if} \quad \text{GRAPH}(i,j) \neq 0 \quad j = 2,3,4$$
$$= 0 \quad \text{otherwise}$$

4. Couple the blocks together as

$$\dot{x} = (A + BGC^{-1}H)x + (BGC^{-1}F)v$$
$$y = (C^{-1}H)x + (C^{-1}F)v$$

The process of constructing the state variable representation is illustrated in the following example.

example: The operations described above will be performed for the system in the diagram.



$$\text{GRAPH} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 2 & 1 & -3 & 5 & 0 \\ 3 & 2 & 0 & 0 & 0 \\ 4 & 2 & 0 & 0 & 0 \\ 5 & 4 & 0 & 0 & 0 \end{bmatrix}$$

1. Write the output equation.

$$C z \quad = \quad H x \quad + \quad F v$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & -K_2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ a & 0 & 0 \\ 0 & -a & 0 \\ 0 & 0 & 0 \\ 0 & 0 & d \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} K_1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} v_1 \end{bmatrix}$$

$$z = C^{-1} H x + C^{-1} F v$$

$$z = \begin{bmatrix} 0 & 0 & 0 \\ a & 0 & 0 \\ 0 & -a & 0 \\ aK_2 & 0 & 0 \\ 0 & 0 & d \end{bmatrix} x + \begin{bmatrix} K_1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} v$$

2. Write the state equation of the uncoupled blocks.

$$\dot{x} = Ax + Bu$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -c & -b & 0 \\ 0 & 0 & -e \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix}$$

3. Write the connection matrix.

$$u = G z$$

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{bmatrix}$$

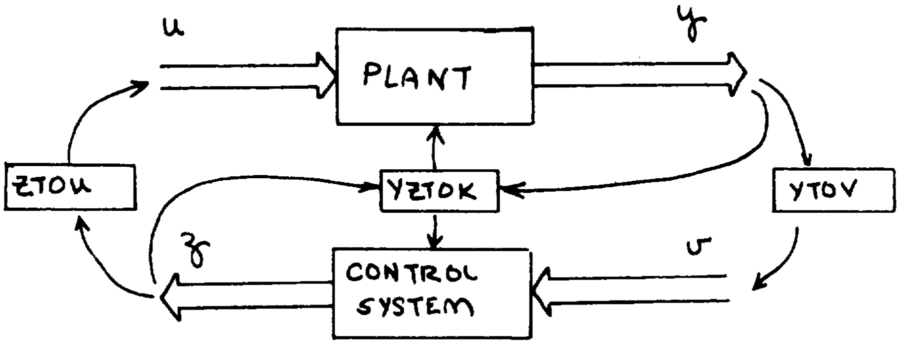
4. Couple the blocks together.

$$\dot{x} = (A + BGC^{-1}H)\,x + (BGC^{-1}F)\,v$$

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ -c & -(a+b) & d \\ aK_2 & 0 & -e \end{bmatrix} x + \begin{bmatrix} 0 \\ K_1 \\ 0 \end{bmatrix} v$$

MIXED INPUT OPTION (CONTINUOUS SYSTEMS)

Mixed systems are characterized by having a portion of the system described by a set of differential equations and the remainder of the system described in block diagram form. The MIXED input option allows these systems to be analyzed without requiring the user to convert the system into a common representation. Typically the plant equations of motion are given and the control system is represented in block diagram form. The MIXED option constructs the state space representation of the system in a two-step process, the first step defining the plant and the second step augmenting the system with the control system block diagram data.



The input-output pairs for the plant and control system are designated ($\underline{u}$, $\underline{y}$) and ($\underline{v}$, $\underline{z}$), respectively. The state of the plant is $\underline{x}_1$, and the state of the control system is $\underline{x}_2$. The blocks YTOV, ZTOU, and YZTOK in the above diagram are additional connection data required by CLASS. They are used to connect the plant and controller. YZTOK defines a feedback gain matrix to allow a conventional root locus to be performed while connections specified by YTOV and ZTOU are incorporated directly into the total system A matrix. Table V indicates the input data required for the MIXED option.

STEP 1 The plant is modeled as an open-loop system regardless of the value of SYSTEM. (In this case, SYSTEM refers to the total, augmented system).

$$C_1 \dot{x}_1 = A_1 x_1 + B_1 u_1$$

$$y = H_1 x_1 + G_1 \dot{x}_1 + Fn$$

The quantities NX, NY, NU in the namelist must refer to this step. Additional dimensions required in step 2 are added by the program.

STEP 2 The control system is modeled by CLASS in the same manner described in the previous section. The required data are:

NBLOCK, GRAPH, BLOCK, NUMER, DENOM, GAIN,
ITHINY, ITHINU, NYTOV, NZTOU, NYZTOK, YTOV,
ZTOU, YZTOK.

The first seven quantities were defined in the last section while the remaining quantities specify how the system is connected.

ITHINY - Integer vector numbering, in sequence, those components of the augmented output vector $(y^T, z^T)^T$ to be saved for analysis.

ITHINU - Integer vector numbering, in sequence those components of the augmented input vector $(u^T, v^T)^T$ to be saved for analysis.

NYTOV - Number of connections from y to v .

NZTOU - " " " z to u .

NYZTOK - Number of feedback paths defined (with root locus option, the first connection specifies K1 and the second connection specifies K3.).

YTOV - Integer matrix of dimension (NYTOV) x 2. Each row of the matrix describes a connection from y to v. The first element of a row specifies the component of y to be connected to the element of v specified by the second element.

ZTOU - Integer matrix of dimension (NZTOU) x 2 specifying connections between z and u in the same fashion as YTOV.

YZTOK - Integer matrix of dimension (NYZTOK) x 2 specifying a feedback gain matrix. The first number of a row specifies which element of the augmented output $(y^T, z^T)^T$ is to be fed back to the element of the augmented input $(u^T, v^T)^T$ specified by the second number of the row.

Note that numbers in YTOV and ZTOU refer to indexing in the individual vectors u, v, y, z while numbers in YZTOK refer to indexing in the augmented vectors $(u^T, v^T)^T$ and $(y^T, z^T)^T$. In all cases, the indexing is specified before any thinning of the output or input has occurred.

The CLASS subroutine constructs the state space representation of the control system with the above data. The system is:

$$\dot{x}_2 = A_2 x_2 + B_2 v$$
$$z = H_2 x_2 + F_2 v$$

The total system, at this point, is in the form of uncoupled diagonal blocks:

$$\left[\begin{array}{c} \dot{x}_1 \\ \hdashline \dot{x}_2 \end{array}\right] = \left[\begin{array}{c:c} A_1 & 0 \\ \hdashline 0 & A_2 \end{array}\right] \left[\begin{array}{c} x_1 \\ \hdashline x_2 \end{array}\right] + \left[\begin{array}{c:c} B_1 & 0 \\ \hdashline 0 & B_2 \end{array}\right] \left[\begin{array}{c} u \\ \hdashline v \end{array}\right] \qquad (7a)$$

$$\left[\begin{array}{c} y \\ \hdashline z \end{array}\right] = \left[\begin{array}{c:c} H_1 & 0 \\ \hdashline 0 & H_2 \end{array}\right] \left[\begin{array}{c} x_1 \\ \hdashline x_2 \end{array}\right] + \left[\begin{array}{c:c} F_1 & 0 \\ \hdashline 0 & F_2 \end{array}\right] \left[\begin{array}{c} u \\ \hdashline v \end{array}\right] \qquad (7b)$$

where the dimensions are

x_1 - NX
x_2 - NX1
y - NY
z - NY1
u - NU
v - NU1

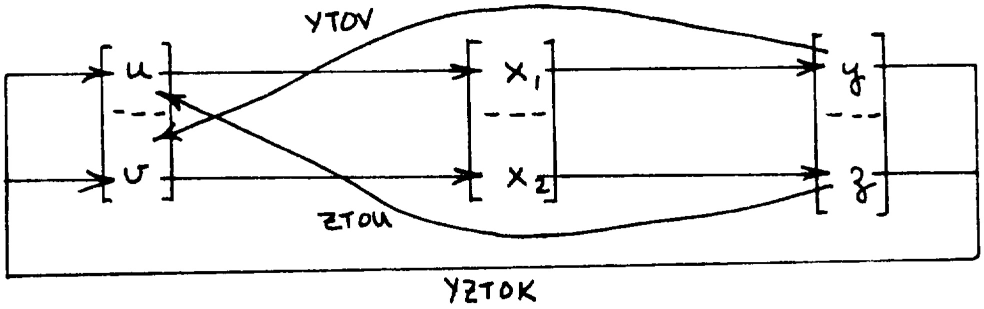
$$\begin{bmatrix} x_1 \\ --- \\ x_2 \end{bmatrix} - \quad NXT = NX + NX1$$

$$\begin{bmatrix} y \\ --- \\ z \end{bmatrix} - \quad NYT = NY + NY1$$

$$\begin{bmatrix} u \\ --- \\ v \end{bmatrix} - \quad NUT = NU + NU1$$

The last three vectors above are the augmented state, augmented output, and augmented input vectors, respectively.

The system is now coupled together using YTOV, ZTOU, and YZTOK. The connections between the augmented input, state, and output vectors may be diagramed as follows.



$$\begin{bmatrix} u \\ --- \\ v \end{bmatrix} = \left[\begin{array}{c:c} 0 & R_1 \\ \hdashline R_2 & 0 \end{array}\right] \begin{bmatrix} y \\ --- \\ z \end{bmatrix} + S \begin{bmatrix} u_{com} \\ ------ \\ v_{com} \end{bmatrix} \tag{8}$$

where subscript "com" designates an external "command" input. The dimensions of R_1, R_2, and S are

$$R_1 - NU \times NY1$$

$$R_2 - NU1 \times NY$$

$$S - NUT \times (\#\text{ of elements in } ITHINU)$$

R_1, R_2, and S are constructed according to the rules:

$$r_{1_{ij}} = \begin{cases} 1.0 & \text{if } ZTOU(k,1) = j \text{ and } ZTOU(k,2) = i \;;\; k = 1,2,\ldots,NZTOU \\ 0 & \text{otherwise} \end{cases}$$

$$r_{2_{ij}} = \begin{cases} 1.0 & \text{if } YTOV(k,1) = j \text{ and } YTOV(k,2) = i \;;\; k = 1,2,\ldots,NYTOV \\ 0 & \text{otherwise} \end{cases}$$

$$s_{ij} = \begin{cases} 1.0 & \text{if } \; i = ITHINU(j) \;;\; j = 1,2,\ldots, \\ 0 & \text{otherwise} \end{cases}$$

The submatrices R_1 and R_2 will be used to couple A_1 and A_2 while S is used to thin out the augmented input vector of unnecessary inputs. Unwanted outputs are thinned out at the very end by simply deleting rows of the final H and F output matrices (ITHINY).

The six partitioned matrices in (7) and (8) are defined to be A, B, H, F, R, and S and the system of equations is reduced to the basic system (1) with the substitutions

$$A \leftarrow A + BR(I-FR)^{-1}H$$

$$B \leftarrow BR(I-FR)^{-1}FS + BS$$

$$H \leftarrow (I-FR)^{-1}H$$

$$F \leftarrow (I-FR)^{-1}FS$$

Now YZTOK is utilized to construct a feedback gain matrix if a root locus is desired. The feedback control law is

$$u = K1\, x + K2\, \dot{x}$$

The feedback gain matrices, K1 and K2, are constructed by the rule:

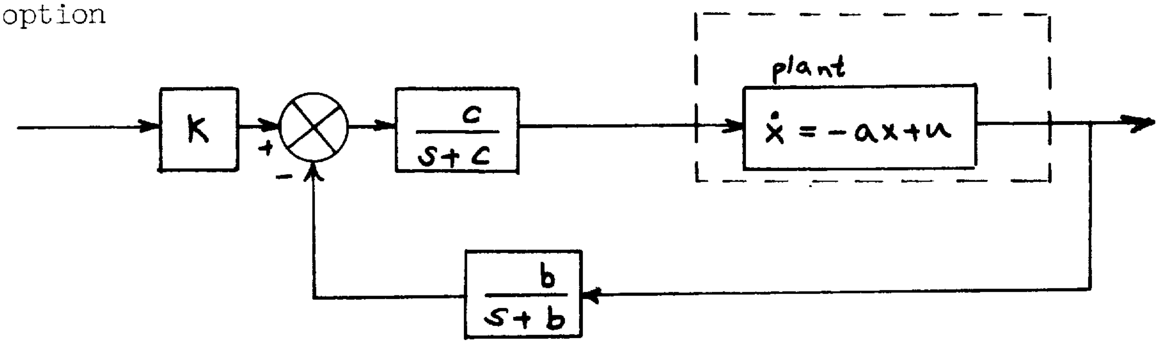
> let YZTOK (1, 1) = i
>
> YZTOK (1, 2) = j
>
> then the jth rows of K1 and K2 are copied from the ith row of the output matrices H and F. Thus feedback is defined from the ith output to the jth input.

If SYSTEM = 3 (root locus) a second row of YZTOK (if any) would be used to construct a second feedback variable into K3 and K4, which would define the second feedback of a two-dimensional root locus.

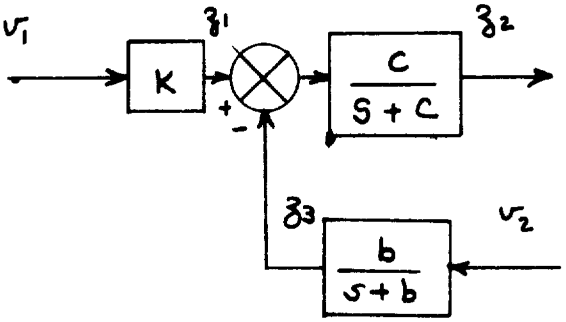
If a root locus is desired in which feedback is defined to the jth input, then the jth input cannot be thinned out of the system (i.e., j must appear in ITHINU).

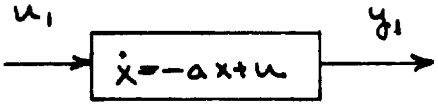
If SYSTEM = 2 (closed loop) any number of feedback paths may be defined in YZTOK.

example: Set up the system indicated below using the MIXED option



The plant is described by a first order differential equation (STEP 1) and the control system is described by its block diagram representation. Redrawing the diagram:





$$\text{GRAPH} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 2 & 1 & -3 & 0 & 0 \\ 3 & 0 & 0 & 0 & 2 \end{bmatrix}$$

a. If a closed-loop analysis is desired

$$YTOV = \begin{bmatrix} 1 & 2 \end{bmatrix}$$

$$ZTOU = \begin{bmatrix} 2 & 1 \end{bmatrix}$$

will connect the system correctly.

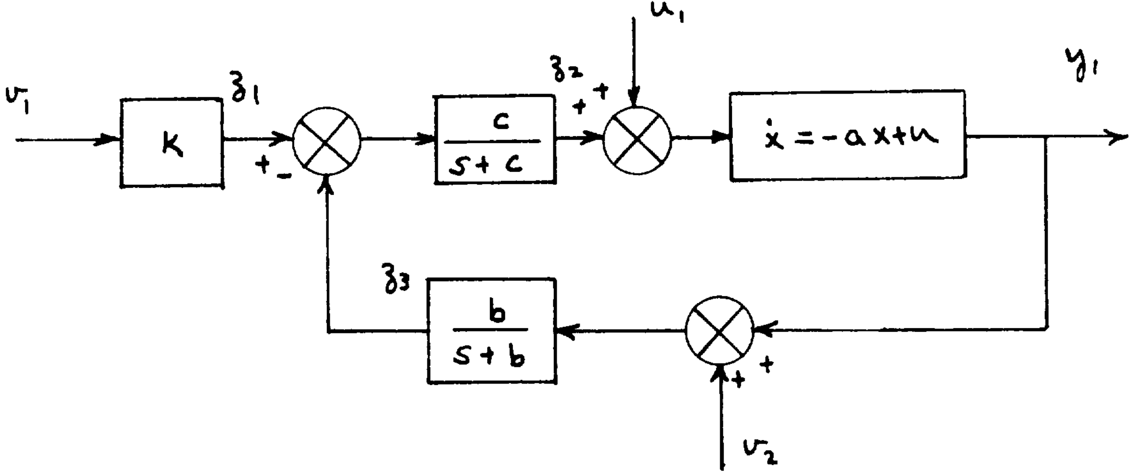
b. If a root locus is desired

$$YTOV = \begin{bmatrix} 1 & 2 \end{bmatrix}$$

$$YZTOK = \begin{bmatrix} 3 & 1 \end{bmatrix}$$

will provide the required feedback matrix (alternatively ZTOU = $\begin{bmatrix} 2 & 1 \end{bmatrix}$, YZTOK = $\begin{bmatrix} 1 & 3 \end{bmatrix}$ will also give the desired locus, defining the feedback at a different point of the system).

Returning to the closed-loop system of a., the resulting closed-loop system will be:



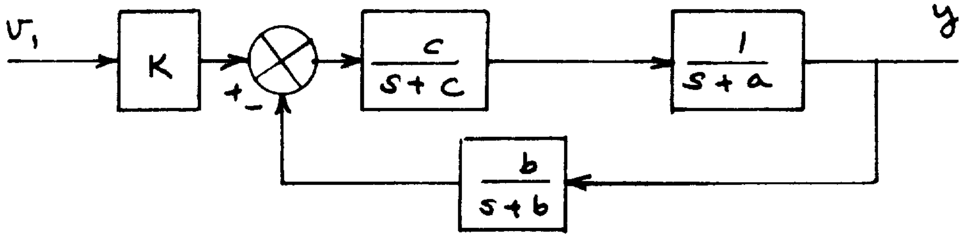
with

$$u = \begin{bmatrix} u_1 \\ \hdashline v_1 \\ v_2 \end{bmatrix} \qquad y^T = \begin{bmatrix} y_1 & z_1 & z_2 & z_3 \end{bmatrix}$$

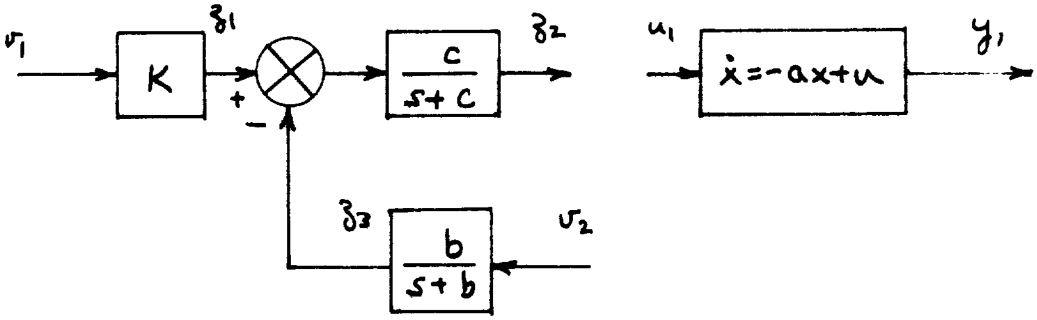
Notice that the connections defined by YTOV and ZTOU create summing junctions. Following the connection of the system the input label (e.g., v_2) refers to the "external input" to the summing junction and not to the "error signal" (output of the summing junction). In many cases these external inputs are not of interest and will be thinned out with ITHINU. Similar thinning will usually be done on the output vector. For instance

$$\text{ITHINU} = [2]$$
$$\text{ITHINY} = [1]$$

results in the system



example: set up the system of the previous example to do a root locus

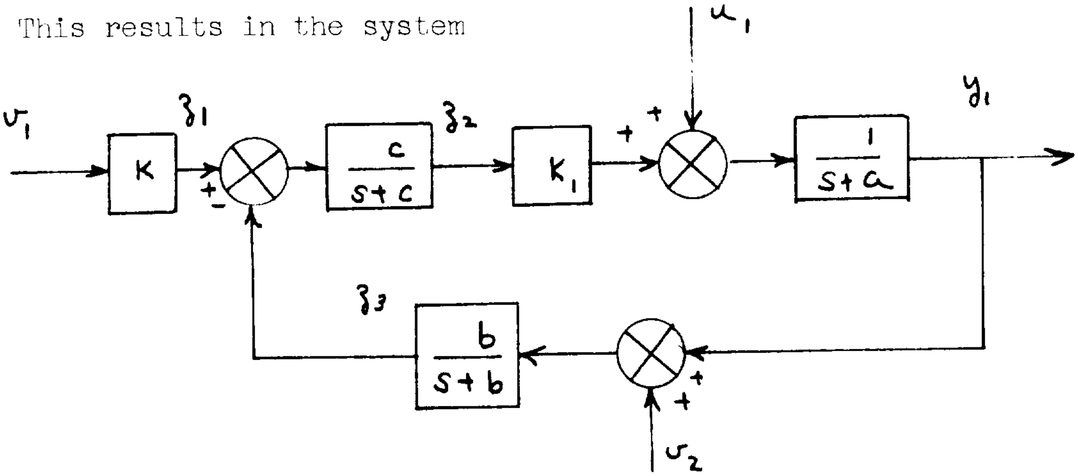


The above system can be put into the standard root locus form by defining

$$\text{YTOV} = \begin{bmatrix} 1 & 2 \end{bmatrix}$$

$$\text{YZTOK} = \begin{bmatrix} 3 & 1 \end{bmatrix}$$

This results in the system

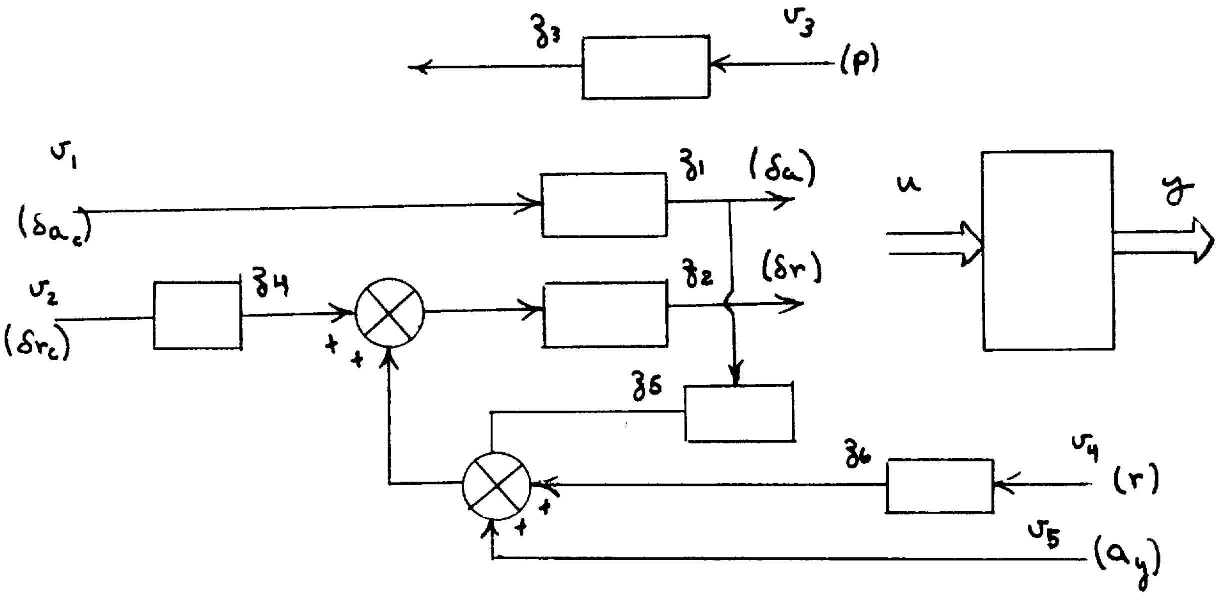


where K_1 is the root locus gain. For this system v_1 and v_2 may be thinned out but $\underline{u_1}$ cannot be thinned out since a feedback variable has been defined to it. No output equation is defined for a root locus case, but the open-loop zeroes of z_2 due to an input at u_1 will be determined.

example: Suppose that STEP 1 of the MIXED option has defined a model of the lateral-directional dynamics of an airplane (A_1, B_1, H_1, F_1) with the plant input and output vectors:

$$u = \begin{bmatrix} \delta a \\ \delta r \end{bmatrix} \qquad y = \begin{bmatrix} p \\ r \\ \beta \\ \phi \\ a_y \end{bmatrix}$$

It is desired to add a control system with actuator dynamics, roll rate, yaw rate, and sideforce feedbacks, and an aileron-to-rudder interconnect. Finally, a root locus of the roll rate feedback is desired with the yaw rate and sideforce feedback loops closed. The block diagram is:



(The internal dynamics of the blocks are not considered in this example since we are concerned with the method of connecting the system together.) The augmented input and output vectors are

$$u = \begin{bmatrix} \delta a \\ \delta r \\ \hdashline \delta a_c \\ \delta r_c \\ p \\ r \\ a_y \end{bmatrix} \qquad\qquad y = \begin{bmatrix} p \\ r \\ \beta \\ \phi \\ a_y \\ \hdashline \delta a \\ \delta r \\ z_3 \\ z_4 \\ z_5 \\ z_6 \end{bmatrix}$$

The pertinent input data are

$$GRAPH = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 2 & 4 & 5 & 6 & 5 \\ 3 & 0 & 0 & 0 & 3 \\ 4 & 0 & 0 & 0 & 2 \\ 5 & 1 & 0 & 0 & 0 \\ 6 & 0 & 0 & 0 & 4 \end{bmatrix}$$

$$YTOV = \begin{bmatrix} 1 & 3 \\ 2 & 4 \\ 5 & 5 \end{bmatrix}$$

$$ZTOU = \begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix}$$

$$YZTOK = \begin{bmatrix} 8 & 3 \end{bmatrix}$$

The loading options which have been described above are summarized in Table V.

1. If READ = 1,2,3 and MIXED = 0, use STEP 1 of Table V. The required matrices must be defined via LOAD, MATRIX, or CHANGE.
2. If READ = 4 and MIXED = 0, use only STEP 2 of Table V. The required matrices will be constructed in CLASS from the block diagram data (GRAPH, etc.).
3. If MIXED = 1, the open-loop plant is defined by STEP 1 of Table V and the control system block diagram added as STEP 2 of Table V. In STEP 1, define the open-loop

plant as if SYSTEM = 1 regardless of the actual value of SYSTEM. SYSTEM, in this case, refers to the augmented system.

DISCRETE SYSTEM MODELS

If a system is known in a completely discrete form, then the system can be described by a combination of the vector difference equations,

$$x_{n+1} = A x_n + B u_n \qquad (10a)$$

$$u_n = K1 x_n + D u_{com_n} \qquad (10b)$$

$$y_n = H x_n + F u_n \qquad (10c)$$

The matrices G, C, K2, and K4 are not defined for discrete systems. This model is algebraically equivalent to the continuous system models (3), (4), and (6) and allows open-loop, closed-loop, and root locus models to be defined. The same computer algorithms which were used to generate eigenvalues for continuous systems can then be used for the discrete system. The resulting transfer functions are Z-transform transfer functions. The vector output sequence, y_n, corresponds to the continuous system transient response and may be generated directly from equations (10). The namelist parameters are defined in the same fashion as for continuous systems and all of the analysis options are available. To specify completely discrete system analysis, set DIGITL = 2 and DELT = T where T is the sample period. The completely discrete system models are given in Table IIIb. Frequency responses are generated in the w-plane by means of the bilinear transformation

$$z = \frac{1 + w}{1 - w}$$

SAMPLED-DATA SYSTEM MODELS

Sampled-data systems are composed of a continuous dynamical subsystem and a discrete subsystem. The continuous subsystem is called the plant and the discrete subsystem is called the digital controller. Usually, but not always, the digital controller will be a dynamical system. To analyze a sampled-data system, the continuous plant must be discretized so that the two subsystems have a common representation. Careful attention must be given to the structure of the system in this discretizing process and the interconnection of the two subsystems. The MIXED option assumes a central role in the sampled-data analysis with the YTOV, ZTOU, and YZTOU options used to define the connections between the subsystem.

The sampled-data system is block diagramed in figure 5 in which $D(z)$ is the digital controller described by the triple (u^d_n, x^d_n, y^d_n) and $G(s)$ is the continuous plant described by the triple (u^c, x^c, y^c) . The plant and digital controller dynamics are given by

$$\dot{x}^c = A_c x^c + B_c u^c \tag{11a}$$

$$y^c = H_c x^c + F_c u^c \tag{11b}$$

$$x^d_{n+1} = A_d x^d_n + B_d u^d_n \qquad (12a)$$

$$y^d_n = H_d x^d_n + F_d u^d_n \qquad (12b)$$

The dimension of x^c is NXC and the dimension of u^c is NUC. (The $G\dot{x}$ term in the output equation for y^c is allowed and has already been eliminated in equation (11)b. These two systems are combined in the partitioned matrix form,

$$\begin{bmatrix} \dot{x}^c \\ \hdashline x^d_{n+1} \end{bmatrix} = \begin{bmatrix} A_c & \vdots & 0 \\ \hdashline 0 & \vdots & A_d \end{bmatrix} \begin{bmatrix} x^c \\ \hdashline x^d_n \end{bmatrix} + \begin{bmatrix} B_c & \vdots & 0 \\ \hdashline 0 & \vdots & B_d \end{bmatrix} \begin{bmatrix} u^c \\ \hdashline u^d_n \end{bmatrix} \qquad (13a)$$

$$\begin{bmatrix} y^c \\ \hdashline y^d_n \end{bmatrix} = \begin{bmatrix} H_c & \vdots & 0 \\ \hdashline 0 & \vdots & H_d \end{bmatrix} \begin{bmatrix} x^c \\ \hdashline x^d_n \end{bmatrix} + \begin{bmatrix} F_c & \vdots & 0 \\ \hdashline 0 & \vdots & F_d \end{bmatrix} \begin{bmatrix} u^c \\ \hdashline u^d_n \end{bmatrix} \qquad (13b)$$

Thus the order of the augmented state and output vectors is

(a) plant states (outputs)

(b) digital controller states (outputs)

CONTROL discretizes the upper left hand (NXC) x (NXC) submatrix of the augmented system A matrix.

The plant is generally assumed to be an open-loop system but analog feedback loops may be defined within the plant. Any such analog feedback must be defined explicitly in the plant A matrix, in YTOV, or ZTOU. Analog actuator dynamics and sensor dynamics are included in the plant. Actuators and sensors may be modeled in block diagram form using the MIXED option. The digital controller will usually be comprised of a summing junction and digital filters. External inputs to the plant and digital controller may be defined. External inputs to the digital controller are considered to be sequences of numbers, $u^{d}_{ext_n}$, separated in time by the sample period, T. External inputs to the plant, u^{c}_{ext} , are considered as sampled continuous inputs. The inputs to the plant are comprised of the outputs of the digital controller and the sampled external inputs. These inputs may be applied to the plant as

(a) outputs of samplers (pulse trains)

(b) outputs of zero-order-hold elements,

The sampled-data block diagram of figure 5 is capable of representing a wide range of sampled-data systems. Figure 6 shows several of the possible configurations. Figure 6(a) shows an open-loop plant with a sampled (pulse train) input, figure 6(b) shows a closed-loop plant with a sampled error signal, figure 6(c) shows a closed-loop plant with digital compensation in the feedback path, and figure 6(d) shows a closed-loop plant with digital compensation in the forward path. Questions arising in the analysis of these systems involves the stability of the closed-loop system, the system transient response, and the synthesis of (digital) control systems. The CONTROL program allows these questions to be studied using the z-plane root locus, the w-plane frequency response (both standard and modified z-transforms), and the system transient response.

In order to perform this analysis, it is necessary to discretize the plant to obtain a discrete system model of the entire sampled-data system.

The form of the input function determines the proper discretization of the plant. CONTROL treats the first ZOH elements of u^c as inputs from zero-order-hold elements and the remaining (NUC - ZOH) elements of u^c as sampled inputs. Thus the ordering of the augmented input vector is

(a) zero-order-hold inputs to plant
(b) sampled inputs to plant
(c) discrete inputs to digital controller

CONTROL discretizes the upper left hand (NXC) x (NUC) submatrix of the augmented system B matrix. The first ZOH columns of the submatrix are discretized to account for the zero-order-hold effect and the remaining (NUC-ZOH) columns are discretized to account for the sampling effect.

The system which results from the discretization of the plant is block diagramed in figure 7. All of the sequences x_n^d , x_n^c , y_n^c , etc., are defined at the same instants of time

$$t = nT \ ; \qquad n = 0, 1, 2, \ldots$$

However, since the system actually contains a continuous subsystem which has a continuous state trajectory it is necessary to define the exact meaning of sequences such as y_n^c . For instance, y_n^c has the two interpretations

$$y_n^{c^+} \equiv \lim_{\epsilon_2 \to 0} y^c(nT+\epsilon_2) \qquad \epsilon_2 > 0 \qquad (14a)$$

$$y_n^{c^-} \equiv y^c(nT-\epsilon_1) \qquad \epsilon_1 > 0 \qquad (14b)$$

Which of these two definitions is used has an important bearing on the resulting discretized system. The CONTROL program assumes

$$x_n^c \equiv \lim_{\epsilon \to 0} x^c(nT-\epsilon) \qquad \epsilon > 0$$

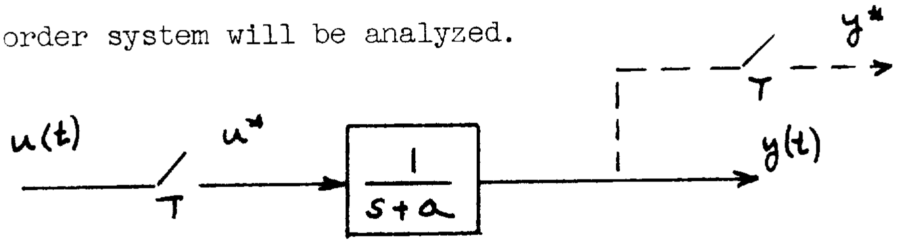
$$u_n^c \equiv u^c(nT)$$

The interpretation of these definitions is that; the state, x_n^c , is defined at time $t=nT$ prior to the application of the input, u_n^c . The updated input, u_n^c , is applied to the system at the sampling instant, $t=nT$. Thus all events are timed with respect to u_n^c as occurring before or after the application of u_n^c .

Figure 8(a) shows the time sequence of events in the complete sampled-data system. At $t=nT-\epsilon_1$, the plant is sampled to provide $y_n^{c^-}$ and $u_{ext_n}^d$ is input to the digital controller. Following the computational delay, ϵ_1 , the digital controller output, y_n^d , is updated (using $y_n^{c^-}$ and $u_{ext_n}^d$), the plant input, u_n^c , is defined (using y_n^d and $u_{ext_n}^c$) and applied to the plant. At $t=nT+\epsilon_2$ the plant output is $y_n^{c^+}$. The time delay ϵ_2 may be regarded as an infinitesimal (i.e., the limit goes to zero in eq. (14(a)) since the plant responds instantaneously to u^c_n . The time delay e_1 cannot be regarded as an infinitesimal since the digital controller requires a finite time to compute the update. y^d_n. However, if $\epsilon_1 << T$ it is customary to assume $\epsilon_1 \longrightarrow 0$.

Figure 8(b) shows the idealized time sequence for the sampled-data system. In this idealized model, events occur only at the sample instants and they occur instantaneously.

To illustrate the discretizing process an open-loop first order system will be analyzed.



The state equation of the plant driven by the inputer train is

$$\dot{x} = -ax + u$$

$$u = u^* = u(t)\,\delta(t-nT) \qquad n=0,1,2,\ldots$$

The solution of the state equation for $0 \leq t < T$ is

$$x(t) = \phi(t)\,x_0 + \int_0^t \phi(t-\tau)\,u^*(\tau)\,d\tau$$

$$= e^{-at}x_0 + \int_0^t e^{-a(t-\tau)}\,\delta(\tau)\,u(\tau)\,d\tau$$

$$= e^{-at} x_0 + e^{-at} u_0$$

and in general

$$x_{n+1} \equiv x[(n+1)T] = e^{-aT} x_n + e^{-aT} u_n \tag{15a}$$

The output equation depends upon the choice of output; y_n^- or y_n^+ . At $t = nT - \epsilon \ (\epsilon \neq 0)$ $u^* = 0$ and

$$y_n^- = x_n \tag{15b}$$

At $t = nT + \epsilon$ the impulse at $t = nT$ will have changed the state x_n to

$$x_n^+ = x(nT+\epsilon) = \lim_{\epsilon \to 0^+} \left[\phi(\epsilon) x_n + \int_{nT}^{nT+\epsilon} \phi(nT+\epsilon-\tau) u^*(\tau) \, d\tau \right]$$

$$= \lim_{\epsilon \to 0^+} \left[\phi(\epsilon) x_n + \phi(\epsilon) u_n \right]$$

$$x_n^+ = x_n + u_n$$

since $\phi(0) = 1$ and $u^*(\tau) = u_n \delta(\tau - nT)$

The output y_n^+ is then

$$y_n^+ = x_n + u_n \tag{15c}$$

since $u(nT+\epsilon) = 0$.

The discretized system is given by equations (15a) and (15b) if y_n^- is chosen as the output, or by (15a) and (15c) if y_n^+ is chosen as the output. The corresponding z-transfer functions are

$$G(z) = \frac{e^{-aT}}{z - e^{-aT}} \quad \text{if} \quad y_n \equiv y_n^- \tag{16a}$$

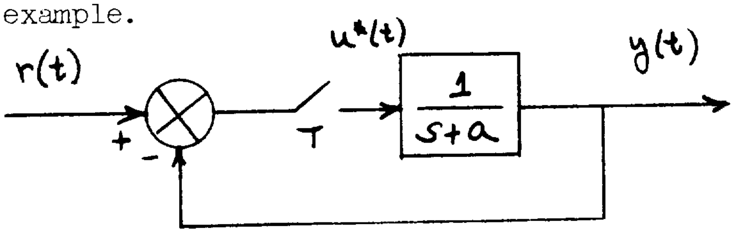
$$G(z) = \frac{z}{z - e^{-aT}} \quad \text{if} \quad y_n \equiv y_n^+ \tag{16b}$$

An interesting result of this example is that (16b) is the standard z-transfer function (pulse transfer function) of $G(s) = \frac{1}{s+a}$. Thus to generate standard z-transfer functions for the plant (11) the following convention should be used:

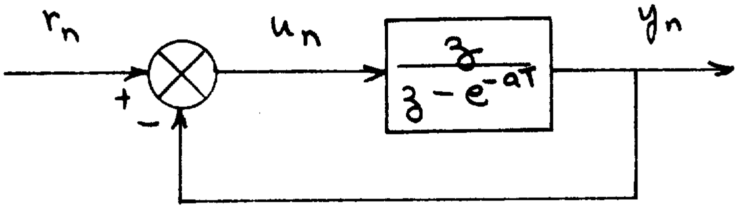
$$y_n \equiv y_n^+ = y(nT+\epsilon) \tag{17a}$$

$$x_n \equiv x_n^- = x(nT-\epsilon) \tag{17b}$$

Although this convention will generate correct pulse transfer functions, it is not the convention required for analyzing closed-loop sampled-data systems. This is evident from the following example.



This system is replaced by the equivalent discrete system:



whose state equation is (assuming $y_n = y_n^+$)

$$x_{n+1} = e^{-aT} x_n + e^{-aT} u_n$$

$$y_n = x_n + u_n$$

The control law is $u_n = r_n - y_n = r_n - x_n - u_n$

This control law is nonphysical since it requires the digital controller to compute, at $t = nT - \epsilon$, a control law requiring data not available until $t = nT$ (see fig. 8). Thus for a closed-loop sampled data system the convention $y_n \equiv y_n^+$ can lead to a nonphysical realization of the system. The correct convention for closed-loop sampled-data systems is

$$y_n \equiv y_n^- = y(nT - \epsilon) \qquad \epsilon > 0$$

since this equation represents the measurements (observations) available to the digital controller at the time at which it must compute the control law.

Table VII lists the correctly discretized plant models for the various input and output definitions. Table VII(a) gives the models for the output convention $y_n \equiv y_n^-$ and Table VII(b) gives the models for the output convention $y_n \equiv y_n^+$. In Table VII(a) F = 0. This is required for the pulsed input since a continuous system with F $\neq$ 0 cannot be driven by a pulse train. The ZOH input system (Table VII(a)) cannot have F $\neq$ 0 since this would make y_n dependent on u_{n-1} and the analysis options of CONTROL are not applicable to such a system.

The only differences in Table VII(a) and (b) are in the direct feedforward term of the output equations. It is interesting to note that these terms in Table VII(b) (HBu_n for pulsed inputs and Fu_n for ZOH inputs) will be zero if

a. all input-output transfer paths have at least 2 more poles than zeroes for the pulsed input system

b. all input-output transfer paths have at least 1 more pole than zeroes for the ZOH system.

Inputs from digital controllers to continuous mechanical plants are almost always applied through zero-order-hold elements whose outputs drive actuators. If actuator dynamics are included in the plant then there will always be at least 1 more pole than zeroes for all transfer paths and there will be no difference in the two ZOH input cases of Table VII(a) and (b).

From Table VI the operations required to discretize the continuous plant are now apparent. CONTROL replaces the augmented system model of equations (13) with

$$\left[\begin{array}{c} x^c_{n+1} \\ \hline x^d_{n+1} \end{array}\right] = \left[\begin{array}{c|c} \phi & 0 \\ \hline 0 & A_d \end{array}\right] \left[\begin{array}{c} x^c_n \\ \hline x^d_n \end{array}\right] + \left[\begin{array}{c} [\Theta B_c] \;\vdots\; [\phi B_c] \;\vdots\; 0 \\ \hline \underbrace{\underbrace{0}_{ZOH} \;\vdots\; 0}_{NUC} \;\vdots\; B_d \end{array}\right] \left[\begin{array}{c} u^c_n \\ \hline u^d_n \end{array}\right] \Big\} NXC \tag{18a}$$

$$\left[\begin{array}{c} y^c_n \\ \hline y^d_n \end{array}\right] = \left[\begin{array}{c|c} H_c & 0 \\ \hline 0 & H_d \end{array}\right] \left[\begin{array}{c} x^c_n \\ \hline x^d_n \end{array}\right] + \left[\begin{array}{c|c|c} [F_c] & [H_c B_c] & 0 \\ \hline 0 & 0 & F_d \end{array}\right] \left[\begin{array}{c} u^c_n \\ \hline u^d_n \end{array}\right] \tag{18b}$$

The partitioned submatrices $\left([\circledast B_c], [\phi B_c], [F_c], [H_c B_c] \right)$ are interpreted as follows: The first NXC rows and ZOH columns of $\circledast B_c$ are copied into the corresponding location of the B matrix, etc. At this point the sampled-data system is completely discretized and is given by

$$x_{n+1} = A x_n + B u_n \tag{19a}$$

$$y_n = H x_n + F u_n \tag{19b}$$

where the augmented vectors and matrices are:

$$x_n = \begin{bmatrix} x^c_n \\ \hdashline x^d_n \end{bmatrix} \qquad u_n = \begin{bmatrix} u^c_n \\ \hdashline u^d_n \end{bmatrix} \qquad y_n = \begin{bmatrix} y^c_n \\ \hdashline y^d_n \end{bmatrix}$$

$$A = \left[\begin{array}{c:c} \phi(T) & 0 \\ \hdashline 0 & A_d \end{array} \right] \qquad B = \left[\begin{array}{c:c:c} [\circledast B_c] & [\phi B_c] & 0 \\ \hdashline 0 & 0 & B_d \end{array} \right]$$

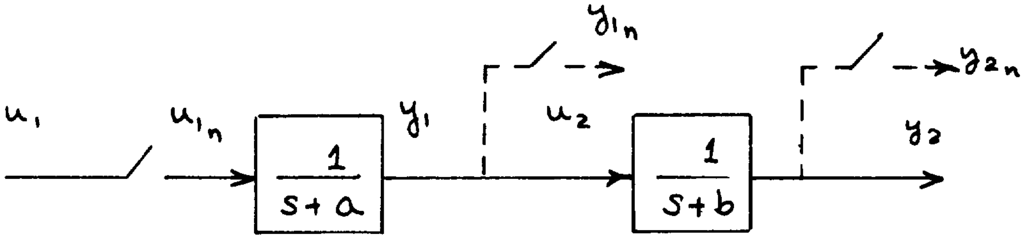
$$H = \left[\begin{array}{c:c} H_c & 0 \\ \hdashline 0 & H_d \end{array} \right] \qquad F = \left[\begin{array}{c:c:c} [F_c] & [H_c B_c] & 0 \\ \hdashline 0 & 0 & F_d \end{array} \right]$$

In light of the preceding discussion concerning the possibility of defining nonphysical feedback control laws, the submatrices $[F_c]$ and $[H_c B_c]$ must be interpreted carefully. CONTROL does not use $[F_c]$ or $[H_c B_c]$ to define connections from the plant to the discrete subsystem or to define sampled-data feedback laws. This amounts to using $y_n^{c-} = y^c(nT-\epsilon)$ instead of $y_n^{c+} = y(nT+\epsilon)$ for these connections. These submatrices are used in computing z-transfer functions, frequency responses, and transient responses. Thus pulse transfer functions such as that of the above example can be computed. This completes the discussion concerning the discretization of the plant dynamics.

The sampled-data system has been reduced to a set of uncoupled matrix difference equations (19). The remaining step in the definition of the system is the method of connecting the digital controller and the discretized plant. These connections are best illustrated by the following examples. The examples illustrate the various possible combinations of continuous and discrete subsystems. Each example contains two systems. The appropriate uncoupled state and output equations are written for each example followed by the equation giving the connection between the two systems. The state equations are discretized, the connection equation incorporated at the appropriate time, and the final coupled discretized system is given.

example: Determine the properly discretized and connected equations for the following systems.

a.)



1. Original system.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -a & 0 \\ 0 & -b \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

2. Connection equation.

$$u_2 = y_1 = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

3. Coupled system.

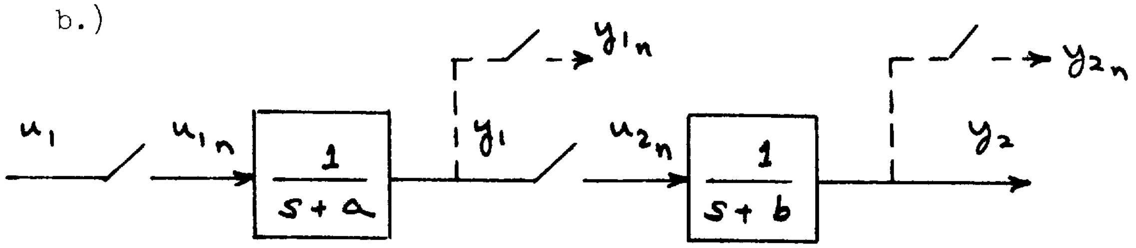
$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -a & 0 \\ 1 & -b \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_1$$

4. Final system.

$$\begin{bmatrix} x_{1_{n+1}} \\ x_{2_{n+1}} \end{bmatrix} = \begin{bmatrix} e^{-aT} & 0 \\ \frac{e^{-aT}-e^{-bT}}{(b-a)} & e^{-bT} \end{bmatrix} \begin{bmatrix} x_{1_n} \\ x_{2_n} \end{bmatrix} + \begin{bmatrix} e^{-aT} \\ 0 \end{bmatrix} \begin{bmatrix} u_{1_n} \end{bmatrix}$$

$$\begin{bmatrix} y_{1_n} \\ y_{2_n} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{1_n} \\ x_{2_n} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} u_{1_n} \end{bmatrix}$$

b.)



1. Original system.

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}-a & 0\\ 0 & -b\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix}$$

$$\begin{bmatrix}y_1\\ y_2\end{bmatrix} = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}$$

2. Discretized system.

$$\begin{bmatrix}x_{1n+1}\\ x_{2n+1}\end{bmatrix} = \begin{bmatrix}e^{-aT} & 0\\ 0 & e^{-bT}\end{bmatrix}\begin{bmatrix}x_{1n}\\ x_{2n}\end{bmatrix} + \begin{bmatrix}e^{-aT} & 0\\ 0 & e^{-bT}\end{bmatrix}\begin{bmatrix}u_{1n}\\ u_{2n}\end{bmatrix}$$

$$\begin{bmatrix}y^{+}_{1n}\\ y^{+}_{2n}\end{bmatrix} = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}x_{1n}\\ x_{2n}\end{bmatrix} + \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}u_{1n}\\ u_{2n}\end{bmatrix}$$

3. Connection equation.

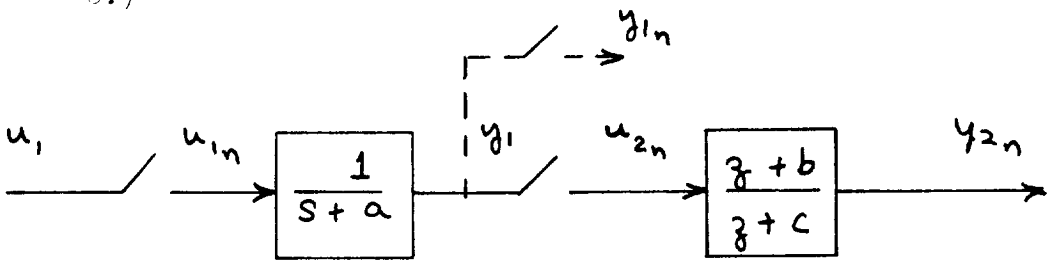
$$u_{2n} = y^{-}_{1n} = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_{1n}\\ x_{2n}\end{bmatrix}$$

4. Final system.

$$\begin{bmatrix}x_{1n+1}\\ x_{2n+1}\end{bmatrix} = \begin{bmatrix}e^{-aT} & 0\\ e^{-bT} & e^{-bT}\end{bmatrix}\begin{vmatrix}x_{1n}\\ x_{2n}\end{vmatrix} + \begin{bmatrix}e^{-aT}\\ 0\end{bmatrix}u_{1n}$$

$$\begin{bmatrix}y^{+}_{1n}\\ y^{+}_{2n}\end{bmatrix} = \begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}\begin{bmatrix}x_{1n}\\ x_{2n}\end{bmatrix} + \begin{bmatrix}1\\ 1\end{bmatrix}u_{1n}$$

c.)



1. Original system.

$$\begin{bmatrix}\dot{x}_1\\ x_{2n+1}\end{bmatrix} = \begin{bmatrix}-a & 0\\ 0 & -c\end{bmatrix}\begin{bmatrix}x_1\\ x_{2n}\end{bmatrix} + \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}u_1\\ u_{2n}\end{bmatrix}$$

$$\begin{bmatrix}y_1^+\\ y_{2n}\end{bmatrix} = \begin{bmatrix}1 & 0\\ 0 & (b-c)\end{bmatrix}\begin{bmatrix}x_1\\ x_{2n}\end{bmatrix} + \begin{bmatrix}0 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}u_1\\ u_{2n}\end{bmatrix}$$

2. Discretized system.

$$\begin{bmatrix}x_{1n+1}\\ x_{2n+1}\end{bmatrix} = \begin{bmatrix}e^{-aT} & 0\\ 0 & -c\end{bmatrix}\begin{bmatrix}x_{1n}\\ x_{2n}\end{bmatrix} + \begin{bmatrix}e^{-aT} & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}u_{1n}\\ u_{2n}\end{bmatrix}$$

3. Connection equation.

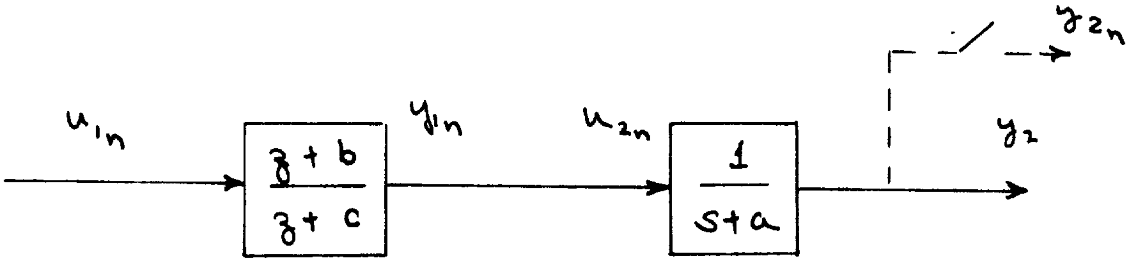
$$u_{2n} = y_{1n}^- = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_{1n}\\ x_{2n}\end{bmatrix}$$

4. Final system.

$$\begin{bmatrix}x_{1n+1}\\ x_{2n+1}\end{bmatrix} = \begin{bmatrix}e^{-aT} & 0\\ 1 & -c\end{bmatrix}\begin{bmatrix}x_{1n}\\ x_{2n}\end{bmatrix} + \begin{bmatrix}e^{-aT}\\ 0\end{bmatrix} u_{1n}$$

$$\begin{bmatrix}y_{1n}^+\\ y_{2n}\end{bmatrix} = \begin{bmatrix}1 & 0\\ 1 & (b-c)\end{bmatrix}\begin{bmatrix}x_{1n}\\ x_{2n}\end{bmatrix} + \begin{bmatrix}1\\ 0\end{bmatrix} u_{1n}$$

d.)



1. Original system.

$$\begin{bmatrix}\dot{x}_2\\ x_{1n+1}\end{bmatrix} = \begin{bmatrix}-a & 0\\ 0 & -c\end{bmatrix}\begin{bmatrix}x_2\\ x_{1n}\end{bmatrix} + \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}u_2\\ u_{1n}\end{bmatrix}$$

$$\begin{bmatrix}y_2\\ y_1\end{bmatrix} = \begin{bmatrix}1 & 0\\ 0 & (b-c)\end{bmatrix}\begin{bmatrix}x_2\\ x_{1n}\end{bmatrix} + \begin{bmatrix}0 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}u_2\\ u_{1n}\end{bmatrix}$$

2. Discretized system.

$$\begin{bmatrix}x_{2n+1}\\ x_{1n+1}\end{bmatrix} = \begin{bmatrix}e^{-aT} & 0\\ 0 & -c\end{bmatrix}\begin{bmatrix}x_{2n}\\ x_{1n}\end{bmatrix} + \begin{bmatrix}e^{-aT} & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}u_{2n}\\ u_{1n}\end{bmatrix}$$

$$\begin{bmatrix}y^{+}_{2n}\\ y_{1n}\end{bmatrix} = \begin{bmatrix}1 & 0\\ 0 & (b-c)\end{bmatrix}\begin{bmatrix}x_{2n}\\ x_{1n}\end{bmatrix} + \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}u_{2n}\\ u_{1n}\end{bmatrix}$$

3. Connection equation.

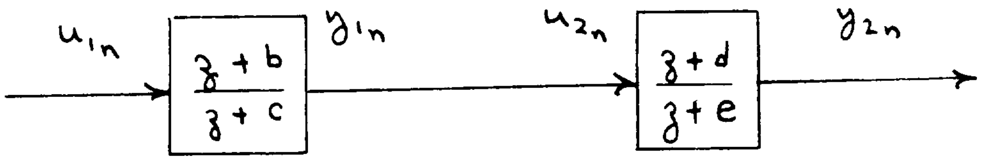
$$u_{2n} = y_{1n} = \begin{bmatrix}0 & (b-c)\end{bmatrix}\begin{bmatrix}x_{2n}\\ x_{1n}\end{bmatrix} + \begin{bmatrix}0 & 1\end{bmatrix}\begin{bmatrix}u_{2n}\\ u_{1n}\end{bmatrix}$$

4. Final system.

$$\begin{bmatrix}x_{2n+1}\\ x_{1n+1}\end{bmatrix} = \begin{bmatrix}e^{-aT} & (b-c)e^{-aT}\\ 0 & -c\end{bmatrix}\begin{bmatrix}x_{2n}\\ x_{1n}\end{bmatrix} + \begin{bmatrix}e^{-aT}\\ 1\end{bmatrix}\begin{bmatrix}u_{1n}\end{bmatrix}$$

$$\begin{bmatrix}y^{+}_{2n}\\ y_{1n}\end{bmatrix} = \begin{bmatrix}1 & (b-c)\\ 0 & (b-c)\end{bmatrix}\begin{bmatrix}x_{2n}\\ x_{1n}\end{bmatrix} + \begin{bmatrix}1\\ 1\end{bmatrix}\begin{bmatrix}u_{1n}\end{bmatrix}$$

e.)



1. Original system.

$$\begin{bmatrix} x_{1n+1} \\ x_{2n+1} \end{bmatrix} = \begin{bmatrix} -c & 0 \\ 0 & -e \end{bmatrix} \begin{bmatrix} x_{1n} \\ x_{2n} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_{1n} \\ u_{2n} \end{bmatrix}$$

$$\begin{bmatrix} y_{1n} \\ y_{2n} \end{bmatrix} = \begin{bmatrix} (b-c) & 0 \\ 0 & (d-e) \end{bmatrix} \begin{bmatrix} x_{1n} \\ x_{2n} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_{1n} \\ u_{2n} \end{bmatrix}$$

2. Connection equation.

$$u_{2n} = y_{1n} = \begin{bmatrix} (b-c) & 0 \end{bmatrix} \begin{bmatrix} x_{1n} \\ x_{2n} \end{bmatrix} + \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} u_{1n} \\ u_{2n} \end{bmatrix}$$

3. Final system.

$$\begin{bmatrix} x_{1n+1} \\ x_{2n+1} \end{bmatrix} = \begin{bmatrix} -c & 0 \\ b-c & -e \end{bmatrix} \begin{bmatrix} x_{1n} \\ x_{2n} \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} u_{1n} \end{bmatrix}$$

$$\begin{bmatrix} y_{1n} \\ y_{2n} \end{bmatrix} = \begin{bmatrix} b-c & 0 \\ b-c & d-e \end{bmatrix} \begin{bmatrix} x_{1n} \\ x_{2n} \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} u_{1n} \end{bmatrix}$$

A close study of these examples reveals that:

i.) Connections between two continuous systems must be made before they are discretized (a).

ii.) Connections to a continuous system from a sampler, zero-order-hold, or digital system must be made after the continuous system is discretized (b and d). Such a continuous system must have an input defined and the connection to the input is specified in a "feedback" gain matrix.

iii.) Connections to a digital system from a continuous system may be made either before or after the system is discretized (c). The discretization has no effect on the connection equation.

iv.) Connections between discrete systems may be made at any time since the discretization does not affect these systems (e).

SAMPLED-DATA SYSTEM ANALYSIS USING THE MIXED OPTION

The use of the MIXED option of the CONTROL program to model sampled-data systems may be described now. The MIXED option involves a two-step process in which a set of linear differential equations are augmented with a block diagram control system.

Sampled-data systems involve a continuous subsystem, the plant, and a discrete subsystem, the digital controller. The CONTROL program constructs the sampled-data system model by defining the plant in STEP 1 and the digital controller in STEP 2 of the MIXED option. It is not required that the plant correspond completely with STEP 1 of the MIXED option. For example, the block diagram control system (STEP 2) may contain a mixture of Laplace transformed blocks (actuator and sensor dynamics) and $\mathfrak{z}$-transformed blocks (the digital filters of the digital controller). The Laplace transform blocks are thus part of the plant. Conversely, difference equations defining digital filters may be written explicitly in the A matrix in STEP 1. The required ordering of inputs, outputs, and states is given in Table VIII(a). This ordering produces a preliminary sampled-data system which is a composite of the system of equations (7) and (13). (Due to the possibility of defining part of the plant in STEP 2, the correspondence between vectors (e.g., x_1 in eq. (7) and x^c in eq. (13)) is not exact.)

This preliminary system is connected together by connections specified in YTOV, ZTOU, and YZTOK. In light of the results of the examples given above, provision must be made for making some connections before the plant is discretized and the remainder after the discretization. To achieve this, the following sequence of operations is performed:

i.) YTOV and ZTOU connections completed

ii.) plant discretized

iii.) YZTOK defines feedback gain matrix, K1

iv.) feedback gain matrix, K1, incorporated into total system A matrix (closed-loop analysis) or used to perform a root locus

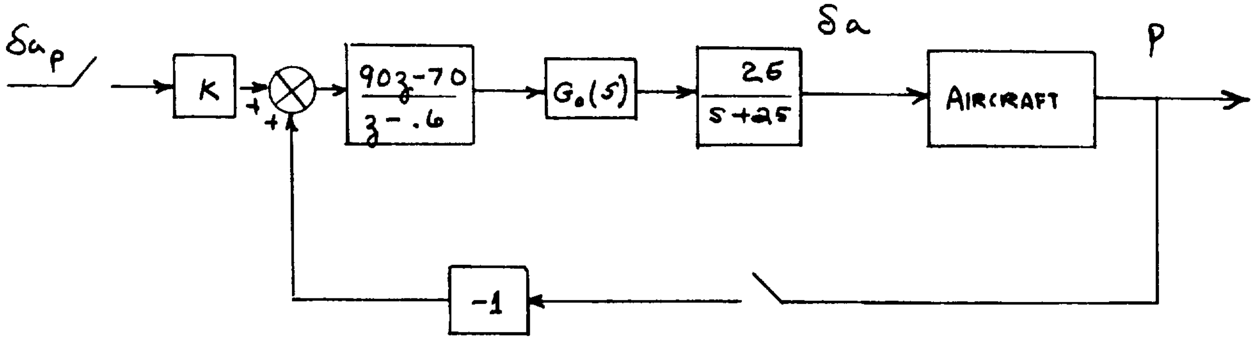
Thus connections which must be completed before the system is discretized are defined in YTOV, ZTOU, GRAPH, or the A_1 matrix of STEP 1. Connections from discrete systems to the plant (which must be made after the plant is discretized) must be defined in YZTOK. Table VIII(b) summarized the types of connections allowed and how they must be defined. $G(s)$ and $D(z)$ represent continuous and discrete subsystems, respectively. The sampled-data system model and its construction are given in Table III(c). The step indicated in Table III(c)(4) is the critical step in constructing the sampled-data system model. This step connects the digital controller to the plant in the correct manner. If a root locus is desired, this step defines the appropriate feedback gain matrix. If a root locus is called for, then the first connection specified by YZTOK will generate the feedback gain matrix, K1, and the second element of YZTOK (if any) will generate a second feedback gain matrix, K3. The control law is then

$$u_n = K1\,x_n + K3\,x_n$$

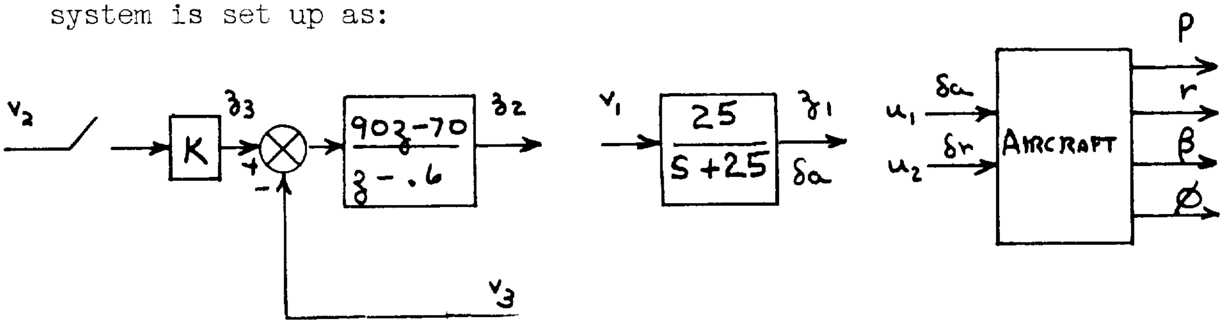
and K1, K3 would generate a root locus grid.

The formulation of the sampled-data system is illustrated in the following examples.

example: A lateral-directional aircraft plant with a roll rate feedback to a digital controller is to be modeled. The block diagram of the system is:



The block diagram is comprised of a pilot input gain, a digital filter, a zero-order-hold, and the aileron actuator transfer function. The aircraft state vector and output vector is $y = x = (p\ r\ \beta\ \phi)^T$ and the aircraft equations of motion are input to CONTROL in STEP 1. The system is set up as:



STEP 1 defines the aircraft with $u = (\delta a\ \delta r)^T$. In STEP 2 the actuator block input and output must be numbered first since it is part of the plant. The appropriate data for STEP 2 is:

NBLOCK 3

GRAPH
$$\begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 2 & 3 & 0 & 0 & -3 \\ 3 & 0 & 0 & 0 & 2 \end{bmatrix}$$

BLOCK
$$\begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & 2 \\ 3 & 1 & 1 \end{bmatrix}$$

NUMER
$$\begin{bmatrix} 25. & & & \\ -70. & 90. & & \\ 1. & & & \end{bmatrix}$$

DENOM
$$\begin{bmatrix} 25. & 1. & & \\ -.6 & 1. & & \\ 1. & & & \end{bmatrix}$$

GAIN $\begin{bmatrix} 1. & 1. & K \end{bmatrix}$

ITHINY $\begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix}$

ITHINU $\begin{bmatrix} 3 & 4 \end{bmatrix}$

NYTOV,NZTOU,
NYZTOK $\begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$

YTOV [1 3]

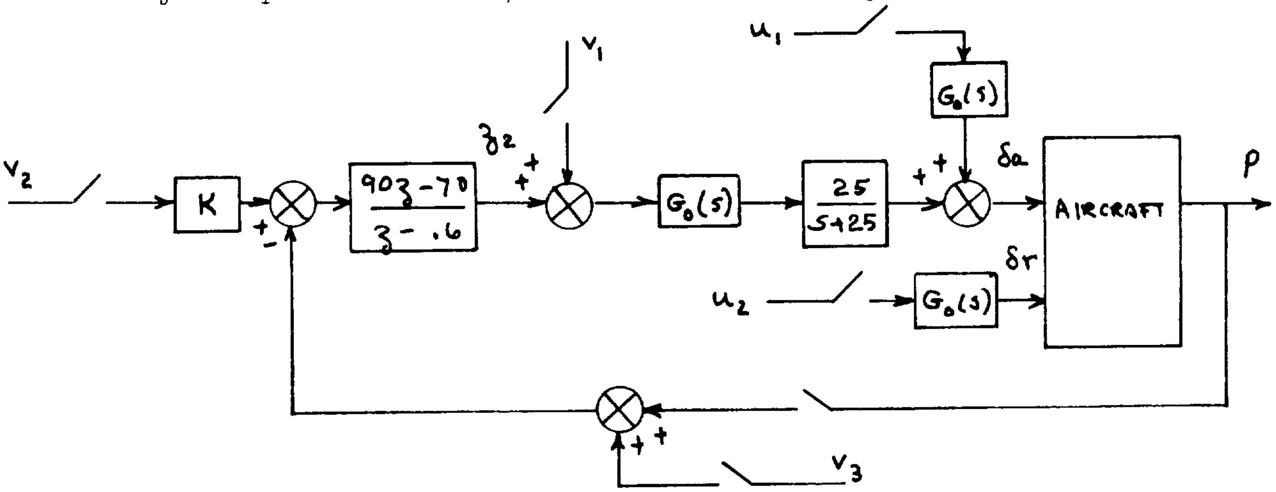
ZTOU [1 1]

YZTOK [6 3]

The pertinent namelist parameters are

DIGITL = 1, NXC = 5, NDC = 3, ZOH = 3, ...

(Other parameters are required to specify STEP 1 and the analysis options chosen.) CONTROL sets the system up as



The connection from z_2 to the summing junction before the zero-order-hold is constructed as a feedback control law

$$u = \begin{bmatrix} \delta a \\ \delta r \\ v_1 \\ v_2 \\ v_3 \end{bmatrix} = K1\,x = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -90. & 0. & 0. & 0. & 0. & -16. \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ r \\ \beta \\ \phi \\ \delta a \\ x \end{bmatrix}$$

If SYSTEM = 2, this "feedback" connection is completed and closed-loop $\mathfrak{z}$-transfer functions, w-plane frequency responses, or transient responses may be obtained. If SYSTEM = 3, the feedback gain matrix is used to calculate the $\mathfrak{z}$-plane root locus.

The ITHINU vector in the above example specifies that only the 3rd and 4th elements of the augmented input vector will be retained. The 3rd element (v_1) must be retained if SYSTEM = 3 since a feedback is defined to this input. If v_1 is not retained the root locus could not be obtained.

COMPUTATIONAL DELAY AND MODIFIED $\mathfrak{z}$-TRANSFORMS

The digital controller requires a finite time, ϵ_1 , to compute the updated command signals to the plant (fig. 8(a)). The preceding analysis has assumed that $\epsilon_1 << T$. If this condition is not met the computational delay may be critical to the system stability. The modified $\mathfrak{z}$-transform allows this case to be analyzed using open-loop frequency response techniques. A linear analysis of a sampled-data system requires that all events pertinent to the system (state, input, output) be defined at one instant in time, the sampling instant. But if $\epsilon \approx T$, the digital filter computes commands based on the system output at time $nT-\epsilon_1$, while the plant state is defined only at time nT . The situation is shown in figure 9. At $t=(n+m-1)T$, the digital controller samples the plant output, $y_n^c(m)$, and inputs

u_n^d . The digital controller then computes the updated plant input u_n^c . This requires $(1-m)T$ seconds where $0 \leq m \leq 1$. If $m = 1$ there is no delay and if $m = 0$ there is a one sample period delay.

The output $y_n^c(m)$ is dependent upon the state of the plant at $t=(n-1)T$ and the form of the input u^c from $t=(n-1)T$ to $t=(n+m-1)T$. If the input to the plant is from a sampler then

$$\begin{aligned} y_n^c(m) &= H\, x^c\left[(n+m-1)T\right] \\ &= H\left[\phi(mT)\, x^c_{n-1} + \phi(mT) B\, u^c_{n-1}\right] \qquad (20a) \end{aligned}$$

where

$$\phi(mT) = \int_0^{mT} e^{-A(mT-\tau)}\, d\tau$$

If the input to the plant is from a zero-order-hold then

$$y_n^c(m) = H\phi(mT)x^c_{n-1} + \left[H\oplus(mT)B + F\right]u^c_{n-1} \qquad (20b)$$

where

$$\oplus(mT) = \int_0^{mT} \phi(mT-\tau)\, d\tau$$

In either case, $y_n^c(m)$ is described by (20a) or (20b) as a linear combination of x^c_{n-1} and u^c_{n-1} . Modified z-transform analysis is performed by idealizing the time sequence model of figure 9 and considering $y_n^c(m)$ and the digital controller update that it generates to occur instantaneously with the plant update at $t = nT$. The resulting system model for sampled inputs is

$$x^c_{n+1} = \phi(T)\, x^c_n + \phi(T) B\, u^c_n \tag{21a}$$

$$y^c_n(m) = H\, \phi(mT)\, x_{n-1} + H\, \phi(mT)\, u^c_{n-1} \tag{21b}$$

The z-transform of (21) yields the transfer function matrix $G(z,m)$ where

$$\begin{aligned} Y_m(z) &\equiv G(z,m)\, U(z) \\ &= \frac{1}{z}\left\{ H\, \phi(mT)\left[(zI - \phi(T))^{-1}\phi(T) B + B\right]\right\} U(z) \end{aligned}$$

Thus the model of equations (21) will generate modified z-transfer functions with the identifications

$$\begin{aligned} A &\longleftarrow \phi(T) \\ B &\longleftarrow \phi(T) B \\ H &\longleftarrow H\phi(mT) \\ F &\longleftarrow H\phi(mT) B \end{aligned}$$

A similar transfer matrix may be derived for zero-order-hold inputs.

The resulting z-transfer functions which the CONTROL program generates must be multiplied by z^{-1} to account for the sample period delay in the output equation (21b).

example: find $G(z,m)$ for $G(s) = \frac{1}{s+a}$

The plant equations are

$$\dot{x} = -ax + u$$

$$y = x$$

then $\phi(T) = e^{-aT}$, $\phi(mT) = e^{-amT}$ and

$$x_{n+1} = e^{-aT} x_n + e^{-aT} u_n$$

$$y_n(m) = e^{-amT} x_{n-1} + e^{-amT} u_{n-1}$$

$$Y_m(z) = \frac{1}{z}\left\{ e^{-amT} \frac{e^{-aT}}{z-e^{aT}} + e^{-amT} \right\} U(z)$$

$$= \frac{1}{z} \frac{z e^{-amT}}{z-e^{-aT}} U(z)$$

or $$G(z, m) = \frac{e^{-amT}}{z-e^{-aT}}$$

To obtain modified z-transforms of open-loop sampled-data systems the parameter m must be specified in the namelist. If $m = 1.$ the standard z-transform will result if $y_n^{c+} = y_n^{c-}$. Numerical errors limit m to $m > .2$. All other namelist parameters required for sampled-data system analysis are unchanged. Only open-loop analysis can be accomplished with $m \neq 0$. Digital filters may be cascaded with the "open loop" plant using the MIXED option as described above. If the digital filter output drives the plant, recall that a quasi "closed-loop" system is set up with YZTOK defining a "feedback" law and SYSTEM must be set to 2. Table VII(c) lists the modified z-transform models.

METHOD OF ANALYSIS

The methods of analysis used by the CONTROL program are summarized in this section.

Frequency Response and Power Spectra

The basic system equations ((1.) or (2.)) may be transformed to yield

$$pX(p) = AX(p) + BU_{com}(p) \tag{22a}$$

$$Y(p) = HX(p) + FU_{com}(p) \tag{22b}$$

where $p = s$ or z depending on the type of system. Equation (22) is rewritten to display the system transfer matrix,

$$Y(p) = G(p)U_{com}(p) \tag{23}$$

where

$$G(p) = H(pI-A)^{-1}B + F$$

The transfer matrix, $G(p)$, is an $NY \times NU$ matrix of transfer functions. If transfer functions and/or frequency responses are requested, CONTROL computes these $(NY)\cdot(NU)$ functions. Frequency responses are generated at discrete frequencies in the following manner:

1. Continuous systems - $G(p) = G(s)$. The frequency response of the ith output due to the jth input is generated by setting $s = j\omega$ in $g_{ij}(s)$ and allowing ω to take on discrete values as specified in Appendix 2.
2. Discrete and sampled-data systems - $G(p) = G(z)$. Frequency responses of z-transformed functions are

accomplished either in the w -plane or the z -plane under the control of FRPS. If FRPS = 1, $G(z)$ is transformed to a w -plane transfer matrix, $G(w)$, by the transformation

$$z = \frac{1+w}{1-w}$$

The frequency response is accomplished by the substitution $w = j\nu$ at discrete points along the positive imaginary axis in the w -plane.

If FRPS = -1, the frequency response is accomplished by the substitution $z = \cos \omega T + j \sin \omega T$ at discrete points along the upper unit semicircle in the z -plane. The advantage of the w -plane frequency response is that asymptotic Bode plot methods may be used. (This is due to the frequency response being a polynomial function of the frequency.)

3. Continuous power spectra - $G(p) = G(s)$

Power spectra are computed for continuous systems from the relation

$$S_{y_i}(\omega) = |g_{ij}(\omega)|^2 S_{u_j}(\omega)$$

where $S_{y_i}(\omega)$ and $S_{u_j}(\omega)$ are the power spectra of the ith output and jth input, respectively, and $g_{ij}(s)$ is the corresponding transfer function. CONTROL assumes that

$$S_{u_j}(\omega) = 1$$

i.e., the input is unity variance white noise. Thus, to compute power spectra, a "shaping filter" will usually be added to the system dynamics and driven by the white noise input. The output of the filter then drives the system with the desired spectral content.

Eigenvalues

CONTROL uses the QR algorithm to determine the system eigenvalues. HESSEN transforms the matrix to upper Hessenburg form. QREIG then determines the eigenvalues via calls to QRT. The subroutines QREIG, HESSEN, and QRT are contained in the IBM Share Program No. 3006.01 written by P. Imiad Fawzi and J. E. VanNess, Northwestern University. The QR algorithm is discussed in reference 1.

Transfer Function Numerators

CONTROL determines the transfer function numerators as the eigenvalues of a matrix derived from the A, B, H, and F matrices. Details may be found in reference 2.

Transient Responses and Discretization of Sampled-Data Systems

To compute transient responses of continuous systems and to discretize sampled-data systems, the transition matrix and its integral are required. They are computed in the EAT subroutine by summing the partial series

$$\phi(t) = e^{At} = I + At + \frac{1}{2!}A^2t^2 + \frac{1}{3!}A^3t^3 + \cdots + \frac{1}{n!}A^nt^n$$

$$\Theta(t) = \int_0^t e^{A(t-\tau)}d\tau = It + \frac{1}{2!}At^2 + \frac{1}{3!}A^2t^3 + \cdots + \frac{1}{n!}A^{(n-1)}t^n$$

The series are terminated when the last terms in both series cause changes to each element of both series less than 10^{-3} times the respective element or when the series has not converged in 24 terms.

In the computation of ϕ for sampled-data systems, it is common to have eigenvalues whose magnitudes are comparable to the half-sample frequency resulting in slow convergence of the series. In this case, the user does not have the flexibility of using a smaller integration step size since the sample period is fixed. To help alleviate this problem, CONTROL computes $\phi(T/8)$ and $\Theta(T/8)$ and then finds $\phi(T)$ and $\Theta(T)$ as

$$\phi(T) = \left[\phi(T/8)\right]^8$$

$$\Theta(T) = \Theta(T/8)\left[\sum_{i=0}^{7} \phi(iT/8)\right]$$

Details may be found in reference 3.

Transient responses for continuous systems are calculated using $\phi(T)$ and $\Theta(T)$ as

$$x[(n+1)T] = \phi(T)x(nT) + \Theta(T)Bu(nT)$$

$$y(nT) = Hx(nT) + Fu(nT)$$

where $u(nT)$ is defined in the INPUT subroutine. The input, $u(nT)$, is held constant between sample periods.

Transient responses for discrete and sampled-data systems are computed in a similar manner from the difference equations given in Table VII.

For sampled-data systems, the MULTRT option allows the user to compute the intersample response of the system. The system is then described by:

$$
\begin{aligned}
x_{n+1} &= A x_n + B u_n \\
y_n &= H x_n + F u_n \\
u_n &= K1 x_n + D u_{com_n}
\end{aligned}
$$

where A and B are obtained by discretizing the plant for the time period, T/MULTRT and u_n is updated every T seconds. That is, the plant is discretized as though the sample period was T/MULTRT but u_n is held constant over T seconds. Thus, MULTRT intersample transient response points will be computed. Only transient responses are allowed with this option.

Digital Filtering

In synthesizing sampled-data systems, much time and effort can go into the computation of digital filter coefficients to give desired filtering to a signal. The CONTROL program allows the user to choose from a table of standard filters (Table VI) the filtering action he desires. The filter may be specified in the s, z, or w-plane. The transformation of s- and w-plane filters to z-plane filters can be carried out automatically by the program, allowing the user to draw upon experience in analog filtering techniques. The transformation of a w-plane filter to a z-plane filter is accomplished by replacing w by

$$w = \frac{z-1}{z+1}$$

The transformation of an s -plane filter to a w -plane filter is

$$u = \tanh\left(\frac{aT}{2}\right)$$

where a is the s -plane first order pole or zero and u is the corresponding w -plane pole or zero. For complex poles and zeroes

$$\left(1 + \frac{2\zeta}{\omega}s + s^2/\omega^2\right)$$

the transformation is

$$\left(1 + \frac{2\zeta_w}{\omega_w}w + w^2/\omega_w^2\right)$$

where

$$\omega_w = \sqrt{u^2 + \nu^2}$$
$$\zeta_w = -u/\omega_w$$
$$u = \frac{\sinh(\alpha T)}{\cosh(\alpha T) + \cos(\beta T)}$$
$$\nu = \frac{\sin(\alpha T)}{\cosh(\alpha T) + \cos(\beta T)}$$

with

$$\omega^2 = \alpha^2 + \beta^2$$
$$\alpha = -\zeta\omega$$

Digital filters derived from this two-step process (called the bilinear transformation with frequency prewarping) maintain a close resemblance of the original s -plane filters over a wide range of the half-sample frequency. References 4 and 5 give discussions of digital filters.

Model Following - If MODEL = 1, CONTROL computes frequency responses appropriate to the evaluation of model following systems. Let y_1 be the model output, y_2 be the model follower output, and let the model have inputs u_1, and u_2. The CONTROL program will compute the frequency responses (with FRPS set appropriately)

1 $\frac{y_1}{u_1}(j\omega)$

2 $\frac{y_2}{u_1}(j\omega)$; $\frac{y_2/u_1}{y_1/u_1}(j\omega)$

3 $\frac{y_1}{u_2}(j\omega)$

4 $\frac{y_2}{u_2}(j\omega)$; $\frac{y_2/u_2}{y_1/u_2}(j\omega)$

Thus, with MODEL set, CONTROL will provide the frequency responses

$$\frac{y_{2i+1}/u_l}{y_{2i}/u_l}(j\omega) \qquad \begin{cases} i = 1,2,\ldots, NY/2 \\ l = 1,2,\ldots, NU \end{cases}$$

in addition to the standard frequency responses.

Similarity Transformation - Numerical problems in calculating eigenvalues can sometimes be traced to the eigenvalues of a matrix being small compared to its norm. In this event, diagonal similarity transformations may aid in eigenvalue computation. For the autonomous system

$$\dot{x} = A x \qquad (24)$$

a similarity transformation is defined as the change of variables

$$z = P x$$

for P nonsingular. The system (24) becomes

$$\dot{z} = P \dot{x} = P A P^{-1} z = A' z$$

with $A' = P A P^{-1}$. The matrix A' has the same eigenvalues as A and if P is chosen properly, A' will have a smaller norm than A.

The CONTROL program includes an option (NSCALE) which the user may select if numerical problems are suspected in eigenvalue computations. In order to maintain the same input-output relationship, the program performs the following operations when the A matrix is scaled.

$$PB \longrightarrow B$$
$$HP^{-1} \longrightarrow H$$
$$K1P^{-1} \longrightarrow K1$$
$$K3P^{-1} \longrightarrow K3$$

The scaling technique is described in references 6 and 7.

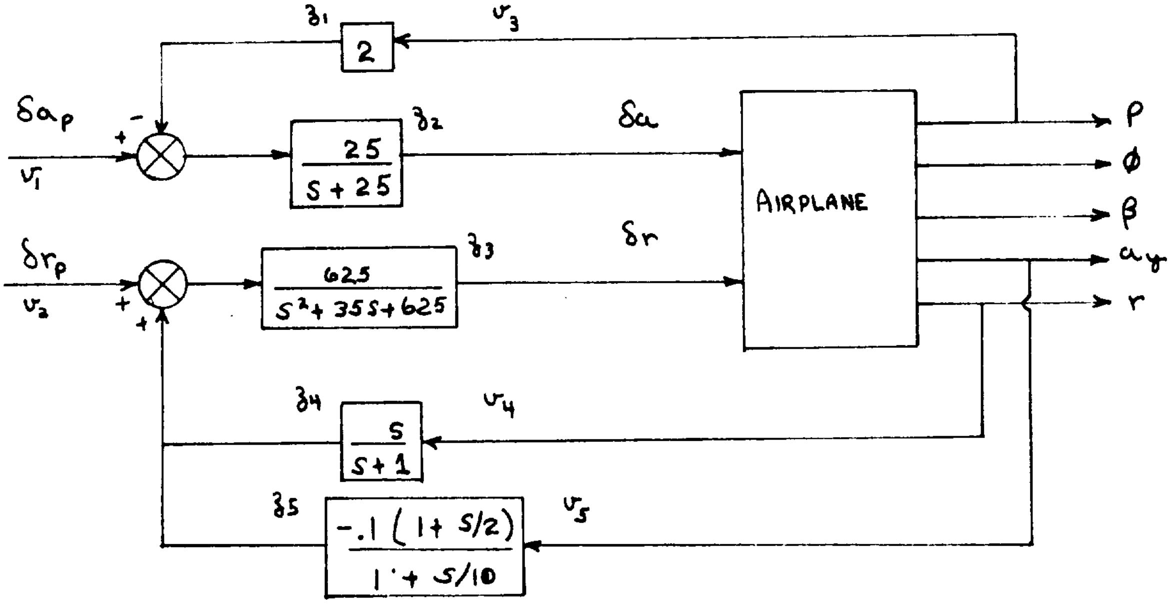
PROGRAM SIZE AND TIMING

The CONTROL program requires 131 K_8 words of computer memory. It has operated on a CDC CYBER 70 computer utilizing a segmented structure which reduced the memory requirement to 72 K_8 words. Included in these size requirements is memory allocated to data matrices for maximum state, output, and input vector dimensions of 15, 15, and 10, respectively. The data matrix storage requirements are a function of the system dimensions.

The example program of the next section was executed on the CDC CYBER 70 and required 17 sec. of CPU execution time to generate the problem setup and calculate ten transfer functions, ten frequency responses, and a transient response.

EXAMPLE PROBLEM

An example problem is given to illustrate the problem formulation, data deck, and output listing. The input and output listings are given in Appendix 3. The problem involves a lateral-directional airplane model with a control system consisting of aileron and rudder actuators, a roll rate feedback to δa, and yaw rate and side force feedback to δr through a washout and lead-lag filter, respectively. The system is shown in the block diagram.



The airplane equations of motion are

$$\begin{bmatrix} \dot{p} \\ \dot{r} \\ \dot{\beta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} L_p & L_r & L_\beta & 0 \\ N_p & N_r & N_\beta & 0 \\ \alpha_t & -1 & Y_\beta & g/V \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ r \\ \beta \\ \phi \end{bmatrix} + \begin{bmatrix} L_{\delta a} & L_{\delta r} \\ N_{\delta a} & N_{\delta r} \\ 0 & Y_{\delta r} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta a \\ \delta r \end{bmatrix}$$

$$\begin{bmatrix} p \\ r \\ \beta \\ \phi \\ a_y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -V/g\,\alpha_t & V/g & 0 & -1 \end{bmatrix} \begin{bmatrix} p \\ r \\ \beta \\ \phi \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & V/g & 0 \end{bmatrix} \begin{bmatrix} \dot{p} \\ \dot{r} \\ \dot{\beta} \\ \dot{\phi} \end{bmatrix}$$

the control system is described using the MIXED option and frequency responses and transient responses are obtained for roll rate and yaw rate.

CONCLUDING REMARKS

A FORTRAN digital computer program for the analysis of linear continuous and sampled-data systems has been described. The program features a flexible input format allowing the program user to define systems in a variety of representations. All systems are analyzed using state variable techniques. Analysis options of the program are: transfer functions, frequency responses, power spectra, root loci, root contours, and transient responses.

Dryden Flight Research Center
National Aeronautics and Space Administration
Edwards, Calif., January 12, 1976

REFERENCES

1. Parlett, B. N.: The LU and QR Algorithms. Mathematical Methods for Digital Computers, Volume II, Anthony Ralston and Herbert S. Wilf, eds., John Wiley & Sons, Inc., c.1967, pp. 116-130.

2. Sandberg, Irwin W.; and So, Hing C.: A Two-Sets-of-Eigenvalues Approach to the Computer Analysis of Linear Systems. IEEE Trans. Circuit Theory, vol. CT-16, no. 4, Nov. 1969, pp. 509-517.

3. Johnson, J. C.; and Phillips, C. L.: An Algorithm for the Computation of the Integral of the State Transition Matrix. IEEE Trans. Automatic Control, vol. AC-16, no. 2, Apr. 1971, pp. 204-205.

4. Kaiser, J. F.: Digital Filters. System Analysis by Digital Computer, Franklin F. Kuo and James F. Kaiser, eds., John Wiley & Sons, Inc., c.1966, pp. 218-285.

5. IEEE Trans. Audio and Electroacoustics. Vol. AU-16, no. 3, Sept. 1968.

6. Osborne, E. E.: On Pre-Conditioning of Matrices. J. Assn. for Computing Machinery, vol. 7, c.1960, pp. 338-345.

7. Harary, Frank: A Graph Theoretic Method for the Complete Reduction of a Matrix With a View Toward Finding Its Eigenvalues. J. Math. Phys., vol. 38, 1959, pp. 104-111.

APPENDIX 1

FORTRAN LISTING OF CONTROL

A brief description of the various subroutines of CONTROL follows:

CONTROL - is the MAIN subroutine as described on page 5.

ADD - performs matrix addition.

ASCALE - scales the A matrix with a diagonal similarity transformation.

CARD - is described on page 4.

CHANGE - is described on page 6.

CLASS - is described on page 4.

CNTRLR - serves as the executive routine for CONTROL.

CPMT - computes complex roots.

EAT - computes the transition matrix and its integral.

EIGEN - is described on page 4

FRQRSP - is described on page 4.

HESSEN - transforms a matrix to the upper Hessenburg form.

INPUTV - is the INPUT subroutine described on page 6.

INVR - determines the inverse of a matrix.

LOAD - is described on page 4.

LOAD1 - is used in conjunction with LOAD to load the system matrices.

MAKE - makes two matrices equivalent.

MATRIX - is described on page 4.

MULT - performs matrix multiplication.

NMRATR - is described on page 4.

PSP - is described on page 4.

OREIG - determines the system eigenvalues.

QRT - is used in conjunction with OREIG.

RDISC - reads input matrices from the disc storage units.

RDISC1 - is used in conjunction with RDISC.

REDUCE - determines the irreducible submatrices of a matrix (used with ASCALE).

ROOT - is described on page 4.

SETUP - is described on page 4.

SPIT - outputs matrices on the printer.

SPIT1 - is used in conjunction with SPIT.

THIST - is described on page 5.
(starts at label SPIT1 16)

SWZ - transforms s-and w-plane filters to z-plane filters.

TANG - computes complex arc tangents.

WDISC - writes input matrices on to disc storage.

WDISC1 - is used in conjunction with WDISC.

ZOT - initializes the system matrices to zero.

ZOT 1 - is used in conjunction with ZOT.

ZTOW - converts z-plane transfer functions to w-plane transfer functions.

COPO - plots data on CALCOMP plotter.

READO - plots zeros for a root locus.

CSTAR - contains envelope curves for the C* options.

SUBSCL - computes the scaling factor for root locus plots.

```
      PROGRAM CONTROL (INPUT=65,OUTPUT=65,TAPE1=INPUT,TAPE3=OUTPUT,TAPE5 CONTROL     2
     1=65,TAPE6=65,TAPE7=65,TAPE8=65,TAPE9=65)                           CONTROL     3
      REAL K1,K2,K3,K4                                                   CONTROL     4
      DIMENSION W1(15,15),W2(15,15),W3(15,15),ROOTR(15),ROOTI(15),       CONTROL     5
     1ROTR(15),ROTI(15),SAV1(200),SAV2(200),U(15),V(15)                  CONTROL     6
      DIMENSION A(15,15),B(15,10),C(15,15),H(15,15),G(15,1 ),F(1',10),   CONTROL     7
     1K1(10,15),K2(10,15),K3(10,15),K4(10,15),D(10,10)                   CONTROL     8
      COMMON/SUBWRIT/ ISUBNAM                                            CONTROL     9
      DATA ISUBNAM/0/                                                    CONTROL    10
C                                                                        CONTROL    11
C     CONTROL MAIN PROGRAM.  ABSOLUTE DIMENSIONS FOR ALL ARRAYS ARE      CONTROL    12
C     DECLARED HERE AND PASSED THROUGHOUT THE PROGRAM IN THE UDIM        CONTROL    13
C     COMMON BLOCK.  TO CHANGE DIMENSIONS OF ARRAYS REQUIRES CHANGES ONL CONTROL    14
C     IN THIS ROUTINE.  MX MUST BE SET EQUAL TO \NX^1' WHERE NX IS THE   CONTROL    15
C     MAXIMUM DIMENSION OF THE STATE VECTOR.                             CONTROL    16
C                                                                        CONTROL    17
      IF (ISUBNAM .GE. 2) WRITE (3,990)                                  CONTROL    18
  990 FORMAT(1X,*MAIN*)                                                  CONTROL    19
      MX=15                                                              CONTROL    20
      MY=10                                                              CONTROL    21
      MU=10                                                              CONTROL    22
      MS=200                                                             CONTROL    23
      CALL      CNTRLR (A,B,C,H,G,F,K1,K2,K3,K4,D,W1,W2,W3,ROOTR,ROOTI,  CONTROL    24
     1ROTR,ROTI,SAV1,SAV2,U,V,                                           CONTROL    25
     2MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         CONTROL    26
      STOP                                                               CONTROL    27
      END                                                                CONTROL    28
```

```
                   SUBROUTINE ADD (X,A,Y,B,C,N,M,
                  1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
                   DIMENSION   A(MAT1,MAT2),B(MAT3,MAT4),C(MAT5,MAT6)
                   COMMON/SUBWRIT/ ISUBNAM
 c                 IF(ISUBNAM.GE.2) WRITE(3,990)
               990 FORMAT(1X,*ADD*)
                   DO 10 I=1,N
                   DO 10 J=1,M
                   C(I,J)=X*A(I,J)+Y*B(I,J)
10              10 CONTINUE
                   RETURN
                   END
```

```
      SUBROUTINE ASCALE (N,J,M,MM,P,A,B,C,DIBAR,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
C
C       THIS SUBROUTINE SCALES THE MATRIX A (OF DIMENSION N) WITH
C       A DIAGONAL SIMILARITY TRANSFORMATION. SUBROUTINE REDUCE IS
C       CALLED TO DETERMINE THE IRREDUCIBLE SUBMATRICES OF A.
C
C          J- NUMBER OF IRREDUCIBLE SUBMATRICES
C     MM(I)- DIMENSION OF THE ITH SUBMATRIX
C        M - MATRIX WHOSE ITH ROW CONTAINS THE ORIGINAL ROW AND COLUMN
C              NUMBERS OF THE ITH SUBMATRIX
C              TRANSFORMATION MATRIX
C        P - VECTOR CONTAINING THE DIAGONAL ELEMENTS OF THE
C
C         THE ROUTINE PRODUCES THE SCALED A MATRIX
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,
     2IGC,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP
      INTEGER DIGITL,SCAPLT,ZOH
      DIMENSION DIBAR (MX)
      DIMENSION A(MX,MX),B(MX,MX),C(MX,MX)
      DIMENSION M(10,20),MM(20),P(20)
      COMMON/SUBWRIT/ ISUBNAM
      IF(ISUBNAM.GE.2) WRITE(3,990)
  990 FORMAT(1X,*SCALE*)
      NSCALE=0
      ITEST=0
      CALL REDUCE (N,J,MM,M,A,B,C,ITEST,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (ITEST.EQ.0) GO TO 110
      DO 111 I=1,N
      P(I)=1.0
  111 CONTINUE
      WRITE (3,112)
  112 FORMAT (/*    SIMILARITY TRANSFORMATION MATRIX SET EQUAL TO IDENTI
     1TY*/)
      RETURN
  110 CONTINUE
      DO 75 I=1,N
      P(I)=0.0
   75 CONTINUE
      STOP=1.0
      DO 100 I=1,J
      NI=MM(I)
      IF (NI.NE.1) GO TO 70
      P(M(I,1))=1.0
      GO TO 100
   70 CONTINUE
      DO 50 II=1,NI
      DO 51 JJ=1,NI
      B(II,JJ)=A(M(I,II),M(I,JJ))
   51 CONTINUE
      DIBAR(II)=0.0
      P(M(I,II))=1.0
   50 CONTINUE
      ICTCC=0
```

```
      K=0                                                               ASCALE    59
    4 DO 10 II=1,NI                                                     ASCALE    60
      SAPI=0.0                                                          ASCALE    61
      RAIP =0.0                                                         ASCALE    62
      DO 20 IP=1,NI                                                     ASCALE    63
      IF (II.EQ.IP)GO TO 20                                             ASCALE    64
      RAIP=RAIP+B(II,IP)**2                                             ASCALE    65
      SAPI=SAPI+B(IP,II)**2                                             ASCALE    66
   20 CONTINUE                                                          ASCALE    67
      DITTY=SQRT(SAPI/RAIP)                                             ASCALE    68
      DIBAR(II)=SQRT(DITTY)                                             ASCALE    69
      DIBARI=1.0/DIBAR(II)                                              ASCALE    70
      P(M(I,II))=P(M(I,II))*DIBAR(II)                                   ASCALE    71
      DO 30 L=1,NI                                                      ASCALE    72
      B(II,L)=B(II,L)*DIBAR(II)                                         ASCALE    73
   30 CONTINUE                                                          ASCALE    74
      DO 40 L=1,NI                                                      ASCALE    75
      B(L,II)=B(L,II)*DIBARI                                            ASCALE    76
   40 CONTINUE                                                          ASCALE    77
      K=K+1                                                             ASCALE    78
      IF ((DIBAR(II)).LE.(1.+STOP))ICTCO=ICTCO+1                        ASCALE    79
      IF ((DIBAR(II)).GT.(1.+STOP))ICTCO=0                              ASCALE    80
      IF (ICTCO.EQ.NI) GO TO 3                                          ASCALE    81
   10 CONTINUE                                                          ASCALE    82
      IF (K.LT.40) GO TO 4                                              ASCALE    83
    3 CONTINUE                                                          ASCALE    84
  100 CONTINUE                                                          ASCALE    85
      DO 60 I=1,N                                                       ASCALE    86
      DO 60 L=1,N                                                       ASCALE    87
      A(I,L)=P(I)*A(I,L)/P(L)                                           ASCALE    88
   60 CONTINUE                                                          ASCALE    89
      IF (IPT.LT.1) GO TO 200                                           ASCALE    90
      WRITE (3,201) (P(I),I=1,N)                                        ASCALE    91
  201 FORMAT (/10X,* THE DIAGONAL TRANSFORMATION MATRIX IS*/40X,(E20.8)) ASCALE   92
  200 CONTINUE                                                          ASCALE    93
      RETURN                                                            ASCALE    94
      END                                                               ASCALE    95
```

```
      SUBROUTINE CARD                                                   CARD       2
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,     CARD       3
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,         CARD       4
     1IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT                  CARD       5
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP CARD       6
      INTEGER DIGITL, SCAPLT, ZOH                                       CARD       7
      COMMON/ACOND/DELT,FINALT,IFREQ,FFREQ,DELFRQ, GAIN1,GAIN2,M        CARD       8
      COMMON/LABEL/INPT,OUTPT,TITLE                                     CARD       9
      COMMON/SUBWRIT/ ISUBNAM                                           CARD      10
      REAL   INPT(10), OUTPT(20),TITLE(8)                               CARD      11
      REAL IFREQ,M                                                      CARD      12
      NAMELIST/CODE/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,   CARD      13
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,IGO,     CARD      14
     2FORM,IPT,READ3,KOUNT,DELT,FINALT,IFREQ,FFREQ,DELFRQ,GAIN1,GAIN2,  CARD      15
     3MIXED,MULTRT,SCAPLT,                                              CARD      16
     4ZOH,M,ISUBNAM                                                     CARD      17
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     CARD      18
      DATA PST/10HROCO      /,PLOT/10HPLOT      /                       CARD      19
  990 FORMAT(1X,*CARD*)                                                 CARD      20
    1 FORMAT (8A10)                                                     CARD      21
    2 FORMAT (40I2)                                                     CARD      22
      IF (KOUNT.NE.1) GO TO 10                                          CARD      23
      WRITE (3,11)                                                      CARD      24
   11 FORMAT (/,"      CONTROL PROGRAM PERMANENT FILE JWELIB,CYCLE 13, JA CARD      25
     1NUARY 1,1976 ",/)                                                 CARD      26
   10 CONTINUE                                                          CARD      27
      IF (CONTUR.EQ.1.AND.KOUNT.GT.1) GO TO 5                           CARD      28
      READ (1,1) TITLE                                                  CARD      29
      IF (TITLE(1).EQ.PLOT) GO TO 300                                   CARD      30
      IF (EOF(1).NE.0) STOP                                             CARD      31
      WRITE (3,3) TITLE                                                 CARD      32
    3 FORMAT (1H1,8A10)                                                 CARD      33
      IF (IFLAG .EQ.1.AND.KOUNT.GT.1) RETURN                            CARD      34
      READ (1,CODE)                                                     CARD      35
  140 FORMAT (10X,*CONTINUOUS SYSTEM*)                                  CARD      36
  141 FORMAT (10X,*SAMPLED-DATA SYSTEM*)                                CARD      37
  142 FORMAT (10X,*DISCRETE SYSTEM*)                                    CARD      38
  243 FORMAT (10X,*MIXED OPTION*)                                       CARD      39
  143 FORMAT (10X,*OPEN LOOP*)                                          CARD      40
  144 FORMAT (10X,*CLOSED LOOP*)                                        CARD      41
  145 FORMAT (10X,*ROOT LOCUS*)                                         CARD      42
  245 FORMAT (10X,*ROOT CONTOUR*)                                       CARD      43
  146 FORMAT (10X,*LOAD ROUTINE INPUT*)                                 CARD      44
  147 FORMAT (10X,*MATRIX ROUTINE INPUT*)                               CARD      45
  148 FORMAT (10X,*CHANGE ROUTINE INPUT*)                               CARD      46
  149 FORMAT (10X,*CLASS ROUTINE INPUT*)                                CARD      47
  151 FORMAT (10X,*TRANSFER FUNCTIONS*)                                 CARD      48
  152 FORMAT (10X,*EIGENVALUES*)                                        CARD      49
  153 FORMAT (10X,*FREQUENCY RESPONSES*)                                CARD      50
  154 FORMAT (10X,*POWER SPECTRA*)                                      CARD      51
  155 FORMAT (10X,*TRANSIENT RESPONSES*)                                CARD      52
      IS=DIGITL+1                                                       CARD      53
      GO TO (170,171,172),IS                                            CARD      54
  170 WRITE (3,140)                                                     CARD      55
      GO TO 200                                                         CARD      56
  171 WRITE (3,141)                                                     CARD      57
      GO TO 200                                                         CARD      58
```

```
  172 WRITE (3,142)                                                       CARD      59
  200 IF (MIXED .EQ. 1) WRITE (3,243)                                     CARD      60
      GO TO (173,174,175),SYSTEM                                          CARD      61
  173 WRITE (3,143)                                                       CARD      62
      GO TO 201                                                           CARD      63
  174 WRITE (3,144)                                                       CARD      64
      GO TO 201                                                           CARD      65
  175 WRITE (3,145)                                                       CARD      66
  201 GO TO (176,177,178,179),READ                                        CARD      67
  176 WRITE (3,146)                                                       CARD      68
      GO TO 202                                                           CARD      69
  177 WRITE (3,147)                                                       CARD      70
      GO TO 202                                                           CARD      71
  178 WRITE (3,148)                                                       CARD      72
      GO TO 202                                                           CARD      73
  179 WRITE (3,149)                                                       CARD      74
  202 IF (NUMERS.EQ.0.AND.SYSTEM.NE.3.AND.CONTUR.EQ.0) WRITE (3,151)      CARD      75
      IF (NUMERS .EQ. 1) WRITE (3,152)                                    CARD      76
      IF (CONTUR .EQ. 1) WRITE (3,245)                                    CARD      77
      IF (FRPS .EQ. 1 .OR. FRPS .EQ. -1) WRITE (3,153)                    CARD      78
      IF (FRPS .EQ. 2) WRITE (3,154)                                      CARD      79
      IF (TRESP .NE. 0) WRITE (3,155)                                     CARD      80
      WRITE (3,150) NX,READ,TRESP,CMAT,DELT,NY,SYSTEM,FRPS,NK2,FINALT,    CARD      81
     1NU,MIXED,NUMERS,IFLAG,IFREQ,NXC,OUTPUT,FORM,IGO,DELFRQ,NUC,DIGITL,  CARD      82
     2CONTUR,READ3,FFREQ,ZOH,IPT,MULTRT,SAV,GAIN1,N1,KOUNT,MODEL,NSCALE,  CARD      83
     3GAIN2,N2,M                                                          CARD      84
  150 FORMAT (//10X,*NX   =*,I4,11X,*READ   =*,I4,8X,*TRESP  =*,I4,8X,    CARD      85
     1*CMAT   =*,I4,8X,*DELT   =*,F7.3/10X,*NY  =*,I4,11X,*SYSTEM =*,     CARD      86
     2I4,8X,*FRPS   =*,I4,8X,*NK2    =*,I4,8X,*FINALT =*,F7.3/10X,        CARD      87
     3*NU  =*,I4,11X,*MIXED  =*,I4,8X,*NUMERS =*,I4,8X,*IFLAG  =*,I4,     CARD      88
     48X,*IFREQ  =*,F7.3/10X,*NXC =*,I4,11X,*OUTPUT =*,I4,8X,*FORM   =*,  CARD      89
     5I4,8X,*IGO    =*,I4,8X,*DELFRQ =*,F7.3/10X,*NUC =*,I4,11X,          CARD      90
     6*DIGITL =*,I4,8X,*CONTUR =*,I4,8X,*READ3  =*,I4,8X,*FFREQ  =*,      CARD      91
     7F7.3/10X,*ZOH =*,I4,11X,*IPT    =*,I4,8X,*MULTRT =*,I4,8X,          CARD      92
     8*SAV    =*,I4,8X,*GAIN1  =*,F7.3/10X,*N1  =*,I4,11X,*KOUNT  =*      CARD      93
     9I4,8X,*MODEL  =*,I4,8X,*NSCALE =*,I4,8X,*GAIN2  =*,F7.3/10X,        CARD      94
     1*N2  =*,I4,71X,*M       =*,F7.3)                                    CARD      95
      IF (MULTRT.GT.0) DELT=DELT/MULTRT                                   CARD      96
      IF (NY.EQ.0) GO TO 50                                               CARD      97
      IF (MIXED.NE.0.AND.NY.LT.8) GO TO 50                                CARD      98
      READ (1,1)(OUTPT(I),I=1,NY)                                         CARD      99
      IF (EOF(1).NE.0) STOP                                               CARD     100
      GO TO 51                                                            CARD     101
   50 READ (1,1)(OUTPT(I),I=1,8)                                          CARD     102
      IF (EOF(1).NE.0) STOP                                               CARD     103
   51 IF (NU.EQ.0) GO TO 52                                               CARD     104
      IF (MIXED.NE.0.AND.NU.LT.8) GO TO 52                                CARD     105
      READ (1,1) (INPT(I),I=1,NU)                                         CARD     106
      IF (EOF(1).NE.0) STOP                                               CARD     107
      GO TO 53                                                            CARD     108
   52 READ(1,1)(INPT(I),I=1,8)                                            CARD     109
      IF (EOF(1).NE.0) STOP                                               CARD     110
   53 CONTINUE                                                            CARD     111
      RETURN                                                              CARD     112
  300 CALL COPO                                                           CARD     113
      STOP                                                                CARD     114
    5 READ (1,1) TITLE                                                    CARD     115
```

```
      IF (EOF(1).NE.0) GO TO 98                                   CARD     116
      WRITE (3,8) TITLE                                           CARD     117
    8 FORMAT (/8A10/)                                             CARD     118
      RETURN                                                      CARD     119
   98 IF (FORM.EQ.0) GO TO 99                                     CARD     120
      ND=3                                                        CARD     121
      XYX=-1.                                                     CARD     122
      WRITE (7) ND,XYX,XYX,XYX                                    CARD     123
   99 STOP                                                        CARD     124
      END                                                         CARD     125
```

```
      SUBROUTINE CHANGE (A,B,C,H,G,F,K1,K2,K3,K4,D,W1,W2,W3,            CHANGE     2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        CHANGE     3
      DIMENSION W1(MX,MX),W2(MX,MX),W3(MX,MX)                           CHANGE     4
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,     CHANGE     5
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,         CHANGE     6
     1IGC,FORM,IPT,READ3,MIXEC,MULTRT,SCAPLT,ZOH,KOUNT                  CHANGE     7
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP CHANGE     8
      INTEGER DIGITL, SCAPLT, ZOH                                       CHANGE     9
      COMMON/ACOND/DELT,FINALT,IFREQ,FFREQ,DELFREQ,GAIN1,GAIN2,M        CHANGE    10
      REAL K1, K2, K3, K4, IFREQ,M                                      CHANGE    11
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),  CHANGE    12
     1K1(MU,MX),K2(MU,MX),K3(MU,MX),K4(MU,MX),D(MU,MU)                  CHANGE    13
      COMMON/BLKDAT/NUMER,DENOM,GAIN,GRAPH,BLOCK,STATE,YTOV,ZTOU,YZTOK, CHANGE    14
     1ITHINY,ITHINU,NBLOCK,NYTOV,NZTOU,NXYU,NYZTOK,NXT,NYT,NUT,NY1,NU1  CHANGE    15
      REAL NUMER                                                        CHANGE    16
      INTEGER GRAPH,BLOCK,STATE,YTOV,ZTOU,YZTOK                         CHANGE    17
      DIMENSION GRAPH(20,5),BLOCK(20,3),NUMER(20,5),DENOM(20,5),        CHANGE    18
     XGAIN(20),STATE(20,4),ITHINY(30),ITHINU(20),YTOV(20,2),            CHANGE    19
     X  ZTOU(20,2),NXYU(8),YZTOK(20,2)                                  CHANGE    20
C                                                                       CHANGE    21
C     USER WRITTEN SUBROUTINE TO CHANGE SYSTEM PARAMETERS SET UP IN     CHANGE    22
C     PREVIOUS CASE                                                     CHANGE    23
C                                                                       CHANGE    24
      COMMON/SUBWRIT/ ISUBNAM                                           CHANGE    25
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     CHANGE    26
  990 FORMAT(1X,*CHANGE*)                                               CHANGE    27
      RETURN                                                            CHANGE    28
      END                                                               CHANGE    29
```

```
      SUBROUTINE CLASS (A,B,C,H,G,F,D,W1,W2,W3,                         CLASS      2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        CLASS      3
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,     CLASS      4
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,         CLASS      5
     2IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT                  CLASS      6
      COMMON/ACOND/DELT,FINALT,IFREQ,FFREQ,DELFRQ, GAIN1,GAIN2,MMM      CLASS      7
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP CLASS      8
      INTEGER DIGITL,SCAPLT,ZOH                                         CLASS      9
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),  CLASS     10
     1D(MU,MU)                                                          CLASS     11
      DIMENSION W1(MX,MX),W2(MX,MX),W3(MX,MX)                           CLASS     12
      REAL IFREQ,NUMER,MMM                                              CLASS     13
      COMMON/BLKDAT/NUMER,DENOM,GAIN,GRAPH,BLOCK,STATE,YTOV,ZTOU,YZTOK, CLASS     14
     1ITHINY,ITHINU,NBLOCK,NYTOV,NZTOU,NXYU,NYZTOK,NXT,NYT,NUT,NY1,NU1  CLASS     15
      INTEGER GRAPH,BLOCK,STATE,YTOV,ZTOU,YZTOK                         CLASS     16
      DIMENSION GRAPH(20,5),BLOCK(20,3),NUMER(20,5),DENOM(20,5),        CLASS     17
     XGAIN(20),STATE(20,4),ITHINY(30),ITHINU(20),YTOV(20,2),            CLASS     18
     X  ZTOU(20,2),NXYU(8),YZTOK(20,2)                                  CLASS     19
    1 FORMAT (16I5)                                                     CLASS     20
      COMMON/SUBWRIT/ ISUBNAM                                           CLASS     21
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     CLASS     22
  990 FORMAT(1X,*CLASS*)                                                CLASS     23
    2 FORMAT (8F10.4)                                                   CLASS     24
      NIN=3                                                             CLASS     25
      NX1=NX                                                            CLASS     26
      NY1=NY                                                            CLASS     27
      NU1=NU                                                            CLASS     28
      NXT=NX                                                            CLASS     29
      NYT=NY                                                            CLASS     30
      NUT=NU                                                            CLASS     31
      IF (IGO.EQ.1) GO TO 224                                           CLASS     32
      NYTOV=0                                                           CLASS     33
      NZTOU=0                                                           CLASS     34
      NYZTOK=0                                                          CLASS     35
  224 IF (MIXED.EQ.1) GO TO 210                                         CLASS     36
      NX1=0                                                             CLASS     37
      NY1=0                                                             CLASS     38
      NU1=0                                                             CLASS     39
  210 CONTINUE                                                          CLASS     40
      IF (IGO.EQ.1.AND.KOUNT.GT.1) GO TO 86                             CLASS     41
      DO 222 I=1,30                                                     CLASS     42
      ITHINY(I)=0                                                       CLASS     43
  222 CONTINUE                                                          CLASS     44
      DO 223 I=1,20                                                     CLASS     45
      ITHINU(I)=0                                                       CLASS     46
  223 CONTINUE                                                          CLASS     47
      READ (1,1) NBLOCK,NIT                                             CLASS     48
      IF (EOF(1).NE.0) STOP                                             CLASS     49
      IF (NBLOCK.EQ.0) GO TO 230                                        CLASS     50
      IF (NIT .EQ. 0) GO TO 736                                         CLASS     51
      CALL SWZ (DELT,IGO,                                               CLASS     52
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        CLASS     53
      GO TO 86                                                          CLASS     54
  736 DO 91 I=1,NBLOCK                                                  CLASS     55
      READ (1,1) (GRAPH(I,J),J=1,5)                                     CLASS     56
      IF (EOF(1).NE.0) STOP                                             CLASS     57
   91 CONTINUE                                                          CLASS     58
```

```
      DO 92 I=1,NBLOCK                                                  CLASS      59
      READ (1,1) (BLOCK(I,J),J=1,3)                                     CLASS      60
      IF (EOF(1).NE.0) STOP                                             CLASS      61
   92 CONTINUE                                                          CLASS      62
      DO 93 I=1,NBLOCK                                                  CLASS      63
      READ (1,2) (NUMER(I,J),J=1,5)                                     CLASS      64
      IF (EOF(1).NE.0) STOP                                             CLASS      65
   93 CONTINUE                                                          CLASS      66
      DO 94 I=1,NBLOCK                                                  CLASS      67
      READ (1,2) (DENOM(I,J),J=1,5)                                     CLASS      68
      IF (EOF(1).NE.0) STOP                                             CLASS      69
   94 CONTINUE                                                          CLASS      70
      READ (1,2) (GAIN(I),I=1,NBLOCK)                                   CLASS      71
      IF (EOF(1).NE.0) STOP                                             CLASS      72
   86 CONTINUE                                                          CLASS      73
      DO 5 I=1,NBLOCK                                                   CLASS      74
      DO 5 J=1,4                                                        CLASS      75
      STATE (I,J)=0.0                                                   CLASS      76
    5 CONTINUE                                                          CLASS      77
      NX=0                                                              CLASS      78
      DO 20 I=1,NBLOCK                                                  CLASS      79
      STATE(I,1)=GRAPH(I,1)                                             CLASS      80
      IF (BLOCK(I,2).EQ.1.AND.BLOCK(I,3).EQ.1) GO TO 21                 CLASS      81
      IF (BLOCK(I,2).GT.BLOCK(I,3)) GO TO 22                            CLASS      82
      IF (BLOCK(I,2).EQ.BLOCK(I,3)) GO TO 23                            CLASS      83
      STATE(I,3)=1                                                      CLASS      84
      GO TO 24                                                          CLASS      85
   23 STATE(I,3)=2                                                      CLASS      86
   24 STATE(I,2)=NX+1                                                   CLASS      87
      STATE(I,4)=BLOCK(I,3)-1                                           CLASS      88
      NX=NX+BLOCK(I,3)-1                                                CLASS      89
      GO TO 20                                                          CLASS      90
   21 STATE(I,3)=4                                                      CLASS      91
      GO TO 20                                                          CLASS      92
   22 STATE(I,3)=3                                                      CLASS      93
   20 CONTINUE                                                          CLASS      94
      NU=0                                                              CLASS      95
      DO 25 K=1,NBLOCK                                                  CLASS      96
      IF (IABS(GRAPH(K,5)).GT.NU) NU=IABS(GRAPH(K,5))                   CLASS      97
   25 CONTINUE                                                          CLASS      98
      NXT=NX+NX1                                                        CLASS      99
      NYT=NBLOCK+NY1                                                    CLASS     100
      NUT=NU+NU1                                                        CLASS     101
      IF (IGC.EQ.1.AND.KOUNT.GT.1) GO TO 271                            CLASS     102
      READ (1,1) (ITHINY(I),I=1,NYT)                                    CLASS     103
      IF (EOF(1).NE.0) STOP                                             CLASS     104
   87 FORMAT (/19X,*ITHINY*/)                                           CLASS     105
      IF (READ.EQ.4.AND.SYSTEM.EQ.3) GO TO 232                          CLASS     106
      IF (MIXED.NE.1) GO TO 243                                         CLASS     107
  232 READ (1,1)(ITHINU(I),I=1,NUT)                                     CLASS     108
      IF (EOF(1).NE.0) STOP                                             CLASS     109
  230 READ (1,1) NYTOV,NZTOU,NYZTOK                                     CLASS     110
      IF (EOF(1).NE.0) STOP                                             CLASS     111
      IF (NYTOV.EQ.0) GO TO 231                                         CLASS     112
      DO 212 I=1,NYTOV                                                  CLASS     113
      READ (1,1)(YTOV(I,J),J=1,2)                                       CLASS     114
      IF (EOF(1).NE.0) STOP                                             CLASS     115
```

```
  212 CONTINUE
  231 IF (NZTOU.EQ.0) GO TO 211
      DO 213 I=1,NZTOU
      READ (1,1)(ZTOU(I,J),J=1,2)
      IF (EOF(1).NE.0) STOP
  213 CONTINUE
  211 IF (NYZTOK.EQ.0) GO TO 243
      DO 219 I=1,NYZTOK
      READ (1,1) (YZTOK(I,J),J=1,2)
      IF (EOF(1).NE.0) STOP
  219 CONTINUE
      IF (SYSTEM.EQ.1) SYSTEM=2
  243 CONTINUE
      IF (NBLOCK.EQ.0) GO TO 241
  214 FORMAT (/10X,*ITHINU*/)
  215 FORMAT (/10X,*YTOV*/)
  217 FORMAT (/10X,*ZTOU*/)
  220 FORMAT (/10X,*YZTOK*/)
      GO TO 272
  271 CONTINUE
      NX1=NXYU(1)
      NY1=NXYU(2)
      NU1=NXYU(3)
      NXT=NXYU(4)
      NYT=NXYU(5)
      NUT=NXYU(6)
      NX=NXYU(7)
      NU=NXYU(8)
      GO TO 273
  272 NXYU(1)=NX1
      NXYU(2)=NY1
      NXYU(3)=NU1
      NXYU(4)=NXT
      NXYU(5)=NYT
      NXYU(6)=NUT
      NXYU(7)=NX
      NXYU(8)=NU
  273 CONTINU
      IF (NIT .EQ. 1) GO TO 340
      WRITE (3,95)
   95 FORMAT (//* BLOCK DIAGRAM INPUT PARAMETERS ARE*//)
      WRITE (3,96)
   96 FORMAT (10X,* GRAPH*/)
      DO 97 I=1,NBLOCK
      WRITE (3,1) (GRAPH(I,J),J=1,5)
   97 CONTINUE
      WRITE (3,98)
   98 FORMAT (/10X,*BLOCK*/)
      DO 99 I=1,NBLOCK
      WRITE (3,1) (BLOCK(I,J),J=1,3)
   99 CONTINUE
      WRITE (3,81)
   81 FORMAT (/10X,*NUMER*/)
      DO 82 I=1,NBLOCK
      WRITE (3,2) (NUMER(I,J),J=1,5)
   82 CONTINUE
      WRITE (3,83)
```

C.2

```
   83 FORMAT (/10X,*DENOM  */)                                        CLASS     173
      DO 84 I=1,NBLOCK                                                CLASS     174
      WRITE (3,2) (DENOM(I,J),J=1,5)                                  CLASS     175
   84 CONTINUE                                                        CLASS     176
      WRITE (3,85)                                                    CLASS     177
   85 FORMAT (/10X,*GAIN*/)                                           CLASS     178
      WRITE (3,2) (GAIN(I),I=1,NBLOCK)                                CLASS     179
  340 WRITE (3,87)                                                    CLASS     180
      WRITE (3,1)(ITHINY(I),I=1,NYT)                                  CLASS     181
      IF (READ.EQ.4.AND.SYSTEM.EQ.3) GO TO 341                        CLASS     182
      IF (MIXED.NE.1) GO TO 242                                       CLASS     183
  341 WRITE (3,214)                                                   CLASS     184
      WRITE (3,1)(ITHINU(I),I=1,NUT)                                  CLASS     185
      IF (NYTOV.EQ.0) GO TO 240                                       CLASS     186
      WRITE (3,215)                                                   CLASS     187
      DO 216 I=1,NYTOV                                                CLASS     188
      WRITE (3,1)(YTOV(I,J),J=1,2)                                    CLASS     189
  216 CONTINUE                                                        CLASS     190
  240 IF (NZTOU.EQ.0) GO TO 241                                       CLASS     191
      WRITE (3,217)                                                   CLASS     192
      DO 218 I=1,NZTOU                                                CLASS     193
      WRITE (3,1)(ZTOU(I,J),J=1,2)                                    CLASS     194
  218 CONTINUE                                                        CLASS     195
  241 IF (NYZTOK.EQ.0) GO TO 242                                      CLASS     196
      WRITE (3,220)                                                   CLASS     197
      DO 221 I=1,NYZTOK                                               CLASS     198
      WRITE (3,1) (YZTOK(I,J),J=1,2)                                  CLASS     199
  221 CONTINUE                                                        CLASS     200
  242 CONTINUE                                                        CLASS     201
      IF (NBLOCK.EQ.0) RETURN                                         CLASS     202
      DO 440 I=1,NBLOCK                                               CLASS     203
      IF (DENOM(I,BLOCK(I,3)).EQ.1.) GO TO 440                        CLASS     204
      N9= BLOCK(I,3)                                                  CLASS     205
      XX=DENOM(I,N9)                                                  CLASS     206
      IF (XX.NE. 0.0) GO TO 442                                       CLASS     207
      WRITE (3,443) I                                                 CLASS     208
  443 FORMAT (/10X,* LEADING COEFFICIENT OF NO.*,I5,* BLOCK IS ZERO*/) CLASS    209
      GO TO 440                                                       CLASS     210
  442 CONTINUE                                                        CLASS     211
      DO 441 J=1,N9                                                   CLASS     212
      DENOM(I,J)= DENOM(I,J)/XX                                       CLASS     213
  441 CONTINUE                                                        CLASS     214
      GAIN(I)= GAIN(I)/XX                                             CLASS     215
  440 CONTINUE                                                        CLASS     216
      DO 30 I=1,NBLOCK                                                CLASS     217
      IF (STATE(I,3).NE.4) GO TO 40                                   CLASS     218
      DO 31 J=1,NIN                                                   CLASS     219
      IF (GRAPH(I,J+1).EQ.0) GO TO 31                                 CLASS     220
      C(I+NY1,IABS(GRAPH(I,J+1))+NY1)=-ISIGN(1,GRAPH(I,J+1))*GAIN(I)  CLASS     221
     1*NUMER(I,1)/DENOM(I,1)                                          CLASS     222
   31 CONTINUE                                                        CLASS     223
      IF (GRAPH(I,5).EQ.0) GO TO 30                                   CLASS     224
      F(I+NY1,IABS(GRAPH(I,5))+NU1)=ISIGN(1,GRAPH(I,5))*GAIN(I)       CLASS     225
     1*NUMER(I,1)/DENOM(I,1)                                          CLASS     226
      GO TO 30                                                        CLASS     227
   40 IF (STATE(I,3).EQ.3) GO TO 50                                   CLASS     228
      NOST1=STATE(I,4)+1                                              CLASS     229
```

```
      NOST=NOST1-1                                                      CLASS    230
      IF (NOST.NE.0) GO TO 45                                           CLASS    231
      WRITE (3,46)                                                      CLASS    232
   46 FORMAT (/10X,*INCONSISTENT DATA IN CLASS*)                        CLASS    233
      GO TO 30                                                          CLASS    234
   45 DO 42 J=1,NOST                                                    CLASS    235
      H(I+NY1,STATE(I,2)+J-1+NX1)=GAIN(I)*(NUMER(I,J)-DENOM(I,J)*NUMER  CLASS    236
     1(I,NOST1))                                                        CLASS    237
   42 CONTINUE                                                          CLASS    238
      IF (STATE(I,3).EQ.1) GO TO 41                                     CLASS    239
      IF (GRAPH(I,5).EQ.0) GO TO 43                                     CLASS    240
      F(I+NY1,IABS(GRAPH(I,5))+NU1)=ISIGN(1,GRAPH(I,5))*GAIN(I)*NUMER   CLASS    241
     1(I,BLOCK(I,2))                                                    CLASS    242
   43 CONTINUE                                                          CLASS    243
      DO 44 J=1,NIN                                                     CLASS    244
      IF (GRAPH(I,J+1).EQ.0) GO TO 44                                   CLASS    245
      C(I+NY1,IABS(GRAPH(I,J+1))+NY1)=-ISIGN(1,GRAPH(I,J+1))*GAIN(I)    CLASS    246
     1*NUMER(I,BLOCK(I,2))                                              CLASS    247
   44 CONTINUE                                                          CLASS    248
   41 CONTINUE                                                          CLASS    249
      GO TO 30                                                          CLASS    250
   50 CONTINUE                                                          CLASS    251
      DO 60 J=1,NIN                                                     CLASS    252
      IF (GRAPH(I,J+1).EQ.0) GO TO 60                                   CLASS    253
      NOST=STATE(IABS(GRAPH(I,J+1)),4)                                  CLASS    254
      IF (NOST.GE.2) GO TO 47                                           CLASS    255
      WRITE (3,48)                                                      CLASS    256
   48 FORMAT (/10X,*DIFFERENTIATOR INPUT NOT ALLOWED*)                  CLASS    257
      GO TO 60                                                          CLASS    258
   47 DO 72 L=2,NOST                                                    CLASS    259
      H(I+NY1,STATE(IABS(GRAPH(I,J+1)),2)+L-1+NX1)=H(I+NY1,STATE(IABS   CLASS    260
     1(GRAPH(I,J+1)),2)+L-1+NX1)+ISIGN(1,GRAPH(I,J+1))*GAIN(I)*GAIN(IABS CLASS    261
     2(GRAPH(I,J+1)))*(NUMER(IABS(GRAPH(I,J+1)),L-1)-NUMER(IABS(GRAPH   CLASS    262
     3(I,J+1)),NOST)*DENOM(IABS(GRAPH(I,J+1)),L))                       CLASS    263
   72 CONTINUE                                                          CLASS    264
      IF (BLOCK(IABS(GRAPH(I,J+1)),2)+1.LT.BLOCK(IABS(GRAPH(I,J+1)),3)  CLASS    265
     1)GO TO 60                                                         CLASS    266
      H(I+NY1,STATE(IABS(GRAPH(I,J+1)),2)+NX1)=H(I+NY1,STATE(IABS(GRAPH CLASS    267
     1(I,J+1)),2)+NX1)+GAIN(I)*GAIN(IABS(GRAPH(I,J+1)))*ISIGN(1,GRAPH   CLASS    268
     2(I,J+1))*(-1.)*NUMER(IABS(GRAPH(I,J+1)),NOST)*DENOM(IABS(GRAPH    CLASS    269
     3(I,J+1)),1)                                                       CLASS    270
      IF (GRAPH(IABS(GRAPH(I,J+1)),5).EQ.0) GO TO 73                    CLASS    271
      F(I+NY1,IABS(GRAPH(IABS(GRAPH(I,J+1)),5))+NU1)=ISIGN(1,GRAPH(IABS CLASS    272
     1(GRAPH(I,J+1)),5))*GAIN(I)*GAIN(IABS(GRAPH(I,J+1)))*NUMER(IABS    CLASS    273
     2(GRAPH(I,J+1)),BLOCK(IABS(GRAPH(I,J+1)),2)-1)*ISIGN(1,GRAPH(I,J+1) CLASS    274
     3)+F(I+NY1,IABS(GRAPH(IABS(GRAPH(I,J+1)),5))+NU1)                  CLASS    275
   73 DO 74 L=1,NIN                                                     CLASS    276
      C(I+NY1,IABS(GRAPH(IABS(GRAPH(I,J+1)),L+1))+NY1)=C(I+NY1,IABS(GRAP CLASS    277
     1H(IABS(GRAPH(I,J+1)),L+1))+NY1)-ISIGN(1,GRAPH(IABS(GRAPH(I,J+1)), CLASS    278
     2L+1))*ISIGN(1,GRAPH(I,J+1))*GAIN(I)*GAIN(IABS(GRAPH(I,J+1)))      CLASS    279
   74 CONTINUE                                                          CLASS    280
   60 CONTINUE                                                          CLASS    281
   30 CONTINUE                                                          CLASS    282
      II1=NY1+1                                                         CLASS    283
      II2=NBLOCK+NY1                                                    CLASS    284
       DO 80II=II1,II2                                                  CLASS    285
      C(II,II)=C(II,II)+1.0                                             CLASS    286
```

```
   80 CONTINUE
      CALL INVR (C,W1,NYT,1,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MX
      MAT2=MX
      MAT3=MY
      MAT4=MX
      MAT5=MX
      MAT6=MX
      CALL MULT (W1,H,W2,NYT,NYT,NXT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MY
      MAT3=MX
      CALL MAKE (H,W2,NYT,NXT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MY
      MAT3=MY
      MAT4=MU
      CALL MULT (W1,F,W2,NYT,NYT,NUT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MY
      MAT2=MU
      MAT3=MX
      MAT4=MX
      CALL MAKE (F,W2,NYT,NUT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      DO 200 I=1,NBLOCK
      NOST=STATE(I,4)
      IF (NOST.EQ.0) GO TO 200
      DO 201 J=1,NOST
      A(STATE(I,2)+NOST-1+NX1,STATE(I,2)+J-1+NX1)=-DENOM(I,J)
      IF (J.EQ.NOST) GO TO 201
      A(STATE(I,2)-1+J+NX1,STATE(I,2)+J+NX1)=1.0
  201 CONTINUE
      IF (IABS(GRAPH(I,5)).EQ.0) GO TO 110
      B(STATE(I,2)+NOST-1+NX1,IABS(GRAPH(I,5))+NU1)=ISIGN(1,GRAPH(I,5))
  110 CONTINUE
      DO 120 J=1,NIN
      IF (GRAPH(I,J+1).EQ.0) GO TO 120
      DO 121 K=1,NX
      A(STATE(I,2)+NOST-1+NX1,K+NX1)=A(STATE(I,2)+NOST-1+NX1,K+NX1)
     1+H(IABS(GRAPH(I,J+1))+NY1,K+NX1)*ISIGN(1,GRAPH(I,J+1))
  121 CONTINUE
      DO 122 K=1,NU
      IF (F(I,K).NE.0.0) OUTPUT=3
      B(STATE(I,2)+NOST-1+NX1,K+NU1)=B(STATE(I,2)+NOST-1+NX1,K+NU1)
     1+F(IABS(GRAPH(I,J+1))+NY1,K+NU1)*ISIGN(1,GRAPH(I,J+1))
  122 CONTINUE
  120 CONTINUE
  200 CONTINUE
      DO 280 I=1,NX
      DO 281 J=1,NX
      C(I+NX1,J+NX1)=0.0
  281 CONTINUE
      C(I+NX1,I+NX1)=1.0
  280 CONTINUE
      DO 90 I=1,NUT
```

```
          DO 191 J=1,NUT                                                CLASS     344
          D(I,J)=0.0                                                    CLASS     345
      191 CONTINUE                                                      CLASS     346
          D(I,I)=1.0                                                    CLASS     347
       90 CONTINUE                                                      CLASS     348
          DO 523 I=1,NYT                                                CLASS     349
          DO 522 J=1,NUT                                                CLASS     350
          IF (F(I,J).EQ.0.0) GO TO 522                                  CLASS     351
          OUTPUT=3                                                      CLASS     352
          GO TO 523                                                     CLASS     353
      522 CONTINUE                                                      CLASS     354
          OUTPUT=1                                                      CLASS     355
      523 CONTINUE                                                      CLASS     356
          NU=NUT                                                        CLASS     357
          NX=NXT                                                        CLASS     358
          NY=NYT                                                        CLASS     359
          RETURN                                                        CLASS     360
          END                                                           CLASS     361
```

```
      SUBROUTINE CNTRLR (A,B,C,H,G,F,K1,K2,K3,K4,D,W1,W2,W3,ROOTR,ROOTI, CNTRLR       2
C    1ROTR,ROTI,SAV1,SAV2,U,V,                                           CNTRLR       3
     1ROTR,ROTI,Z,ZZ,U,V,                                                CNTRLR       4
     2MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         CNTRLR       5
C                                                                        CNTRLR       6
C         THIS SUBROUTINE SERVES AS THE EXECUTIVE ROUTINE FOR THE        CNTRLR       7
C         CONTROL PROGRAM. THE CONTROL PROGRAM IS CAPABLE OF PERFORMING  CNTRLR       8
C         ,FOR LINEAR SYSTEMS, THE FOLLOWING OPERATIONS                  CNTRLR       9
C                                                                        CNTRLR      10
C           1. ROOT LOCII AS A FUNCTION OF TWO FEEDBACK GAINS            CNTRLR      11
C                                                                        CNTRLR      12
C           2. DETERMINATION OF SYSTEM EIGENVALUES FOR OPEN AND          CNTRLR      13
C              CLOSED-LOOP SYSTEMS                                       CNTRLR      14
C                                                                        CNTRLR      15
C           3. DETERMINATION OF SYSTEM TRANSFER FUNCTIONS FOR            CNTRLR      16
C              ARBITRARY INPUT-OUTPUT VARIABLES                          CNTRLR      17
C                                                                        CNTRLR      18
C           4. CALCULATION OF TABULATED FREQUENCY RESPONSES              CNTRLR      19
C                                                                        CNTRLR      20
C           5. CALCULATION OF TABULATED POWER SPECTRAL DENSITY           CNTRLR      21
C              FUNCTIONS.                                                CNTRLR      22
C                                                                        CNTRLR      23
C           5. TABULATED TIME HISTORY RESPONSE                           CNTRLR      24
C                                                                        CNTRLR      25
C        COMPUTATIONS ARE PERFORMED USING STATE VARIABLE MATRIX          CNTRLR      26
C        NOTATION.                                                       CNTRLR      27
C                                                                        CNTRLR      28
C      CORRECTION MADE BY G. NORRIS JULY 5 73                            CNTRLR      29
C      Z AND SAV1, ZZ AND SAV2 ARE SAME MATRIX                           CNTRLR      30
C                                                                        CNTRLR      31
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,      CNTRLR      32
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,          CNTRLR      33
     1IGC,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT                   CNTRLR      34
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP  CNTRLR      35
      INTEGER DIGITL,SCAPLT,ZOH                                          CNTRLR      36
      COMMON/ACOND/ DELT,FINALT,IFREQ,FFREQ,DELFRQ,GAIN1,GAIN2,MN        CNTRLR      37
      REAL   INPT(10), OUTPT(20),TITLE(8)                                CNTRLR      38
      COMMON/LABEL/INPT,OUTPT,TITLE                                      CNTRLR      39
      REAL IFREQ,K1,K2,K3,K4,MN                                          CNTRLR      40
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),   CNTRLR      41
     1K1(MU,MX),K2(MU,MX),K3(MU,MX),K4(MU,MX),D(MU,MU),                  CNTRLR      42
     2W1(MX,MX),W2(MX,MX),W3(MX,MX),ROOTR(MX),ROOTI(MX),ROTR(MX),        CNTRLR      43
     3ROTI(MX),U(MX),V(MX),Z(MS),ZZ(MS)                                  CNTRLR      44
C     DIMENSION A(15,15), B(15,10), C(15,15), H(15,15), G(15,15),        CNTRLR      45
C    1 F(15,10), K1(10,15), K2(10,15), K3(10,15), K4(10,15), D(10,10)    CNTRLR      46
C     DIMENSION W1(15,15), W2(15,15), W3(15,15), ROOTR(15), ROOTI(15),   CNTRLR      47
C    1 ROTR(15), ROTI(15), U(15), V(15), Z(200), ZZ(200)                 CNTRLR      48
      DIMENSION M(10,20),MM(20),P(20),ICOND(29),JCOND(29),RCOND(8)       CNTRLR      49
      EQUIVALENCE (READ,ICOND(1)),(DELT,RCOND(1))                        CNTRLR      50
      COMMON/BLKDAT/NUMER,DENOM,GAIN,GRAPH,BLOCK,STATE,YTOV,ZTOU,YZTOK,  CNTRLR      51
     1ITHINY,ITHINU,NBLOCK,NYTOV,NZTOU,NXYU,NYZTOK,NXT,NYT,NUT,NY1,NU1   CNTRLR      52
      DIMENSION GRAPH(20,3),BLOCK(20,3),NUMER(20,5),DENOM(20,5),         CNTRLR      53
     XGAIN(20),STATE(20,4),ITHINY(30),ITHINU(20),YTOV(20,2),             CNTRLR      54
     X  ZTOU(20,2),NXYU(8),YZTOK(20,2)                                   CNTRLR      55
      INTEGER GRAPH,BLOCK,STATE,YTOV,ZTOU,YZTOK                          CNTRLR      56
      REAL NUMER                                                         CNTRLR      57
      COMMON/SUBWRIT/ ISUBNAM                                            CNTRLR      58
```

```
           DATA  RCON/10HROCO       /                                  CNTRLR      59
           IF(ISURNAM.GE.2) WRITE(3,999)                               CNTRLR      60
       999 FORMAT(1X,*CNTRLR*)                                         CNTRLR      61
           KOUNT=0                                                     CNTRLR      62
           ISAV=0                                                      CNTRLR      63
       500 CONTINUE                                                    CNTRLR      64
           ITT=0                                                       CNTRLR      65
           DO 202 I=1,29                                               CNTRLR      66
           JCOND(I)=0                                                  CNTRLR      67
       202 CONTINUE                                                    CNTRLR      68
           DO 203 I=1,8                                                CNTRLR      69
           BCOND(I)=0.0                                                CNTRLR      70
       203 CONTINUE                                                    CNTRLR      71
       501 CONTINUE                                                    CNTRLR      72
           REWIND 8                                                    CNTRLR      73
           KOUNT=KOUNT+1                                               CNTRLR      74
           DO 200 I=1,29                                               CNTRLR      75
           ICOND(I)=JCOND(I)                                           CNTRLR      76
       200 CONTINUE                                                    CNTRLR      77
           CALL CARD                                                   CNTRLR      78
           IF(READ.NE.3) ISAV=0                                        CNTRLR      79
           DO 201 I=1,29                                               CNTRLR      80
           JCOND(I)=ICOND(I)                                           CNTRLR      81
       201 CONTINUE                                                    CNTRLR      82
           IF((CONTUR.EQ.1).AND.(FORM.GT.0).AND.(ITT.EQ.0)) GO TO 91   CNTRLR      83
           GO TO 92                                                    CNTRLR      84
        91 NPLOT=1                                                     CNTRLR      85
           WRITE (7) NPLOT                                             CNTRLR      86
           WRITE (7) RCON,TITLE,SYSTEM,MODEL,DIGITL,SCAPLT             CNTRLR      87
        92 IF (CONTUR.EQ.1) ITT=1                                      CNTRLR      88
           CALL ZOT (A,B,C,H,G,F,K1,K2,K3,K4,D,                        CNTRLR      89
          1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                  CNTRLR      90
           GO TO (30,40,50,55),READ                                    CNTRLR      91
        30 CALL LOAD     (A,B,C,H,G,F,K1,K2,K3,K4,D,                   CNTRLR      92
          1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                  CNTRLR      93
           GO TO 60                                                    CNTRLR      94
        40 CALL MATRIX (A,B,C,H,G,F,K1,K2,K3,K4,D,W1,W2,W3,            CNTRLR      95
          1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                  CNTRLR      96
      C                                                                CNTRLR      97
           DO 45 I=1,29                                                CNTRLR      98
        45 JCOND(I)=ICOND(I)                                           CNTRLR      99
      C                                                                CNTRLR     100
           GO TO 60                                                    CNTRLR     101
        50 IF (ISAV.EQ.0) GO TO 51                                     CNTRLR     102
           CALL RCTSC (A,B,C,H,G,F,K1,K2,K3,K4,D,                      CNTRLR     103
          1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                  CNTRLR     104
        51 CALL CHANGE (A,B,C,H,G,F,K1,K2,K3,K4,D,W1,W2,W3,            CNTRLR     105
          1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                  CNTRLR     106
           IF (MIXED.EQ.0.AND.ISAV.EQ.0) GO TO 555                     CNTRLR     107
           GO TO 60                                                    CNTRLR     108
        55 IGO =0                                                      CNTRLR     109
       555 CONTINUE                                                    CNTRLR     110
           CALL CLASS       (A,B,C,H,G,F,D,W1,W2,W3,                   CNTRLR     111
          1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                  CNTRLR     112
        60 CONTINUE                                                    CNTRLR     113
           IF (READ3.EQ.1) JCOND(1)=3                                  CNTRLR     114
           IF (SAV.EQ.1)ISAV=1                                         CNTRLR     115
```

```
      IF (CONTUR.EQ.1.AND.KOUNT.GT.1) GO TO 69
      CALL OPTT     (A,B,C,H,G,F,K1,K2,K3,K4,D,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
   69 IF (SAV.EQ.0) GO TO 70
      CALL NOISC    (A,B,C,H,G,F,K1,K2,K3,K4,D,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      JCOND(18)=0
   70 CONTINUE
      CALL        SETUP (J,M,MM,P,A,B,C,H,G,F,K1,K2,K3,K4,D,W1,W2,W3,
     1 ROOTR,
     2 MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (TRESP.GT.0) NPLOT=1
      IF (TRESP.GT.0) NPLOT=TRESP
      IF (FRPS.NE.0) NPLOT=NY*NU
      IF (FRPS.GT.0.AND.TRESP.GT.0) NPLOT=NY*NU+1
      IF (FRPS.NE.0.AND.TRESP.GT.0) NPLOT=NY*NU+TRESP
      IF (SYSTEM.EQ.3) NPLOT=1
      IF (FORM.GT.0.AND.CONTUR.EQ.0) WRITE (7) NPLOT
      IF (MULTRT.NE.0) GO TO 101
      IF (SYSTEM.NE.3) GO TO 80
      CALL ROOT (A,B,C,H,G,F,K1,K2,K3,K4,D,W1,W2,W3,ROOTR,ROOTI,U,V,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      GO TO 90
   80 CONTINUE
      WRITE (3,81)
   81 FORMAT   (//10X,*THE EIGEN VALUES OF THE SYSTEM ARE*//20X,
     1*REAL PART*,15X,* IMAGINARY PART*/)
      CALL EIGEN (NX,W1,W2,W3,ROOTR,ROOTI,U,V,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (IFLAG.EQ.0.AND.CONTUR.EQ.1) GO TO 499
      IF (CONTUR.EQ.1) GO TO 501
      GO TO 502
  499 IF (FORM.EQ.0) GO TO 500
      ND=1
      XYX=-1.0
      WRITE(7)ND,XYX,XYX
      GO TO 500
  502 CONTINUE
      CALL CPMT (7,ROOTR,ROOTI,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      WRITE (3,82)
   82 FORMAT (//10X,*THE COEFFICIENTS OF THE CHARACTERISTIC EQUATION ORD
     1ERED FROM THE LOWEST POWER OF S*//)
      NX1=NX+1
      WRITE (3,83) (Z(I),I=1,NX1)
   83 FORMAT (E20.8)
   90 CONTINUE
      IF (NUMERS.EQ.1) GO TO 100
      NN=NX
C     CALL        NMRATR (NN,A,B,C,H,G,F,D,ROOTR,ROOTI,ROTR,ROTI,Z,V,
      CALL        NMRATR (NN,A,B,C,H,G,F,D,ROOTR,ROOTI,ROTR,ROTI,U,V,
C    1W1,W2,W3,SAV1,SAV2,
     1W1,W2,W3,Z,ZZ,
     2MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
  100 CONTINUE
      IF(TRESP.EQ.0) GO TO 507
  101 CONTINUE
```

```
                     CALL       THIST (A,B,C,H,F,W1,W2,W3,ROOTR,ROOTI,U,K1,D,Z,V,       CNTRLR    173
                     CALL       THIST (A,B,C,H,F,W1,W2,W3,ROOTR,ROOTI,U,K1,C,ROTI,ROTR, CNTRLR    174
                    1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         CNTRLR    175
175              507 IF (SCAPLT.EQ.2) JCOND(28)=1                                       CNTRLR    176
                     IF (IFLAG.EQ.0) GO TO 100                                          CNTRLR    177
                     GO TO 501                                                          CNTRLR    178
                     END                                                                CNTRLR    179
```

```
      SUBROUTINE CPMT(C,ROOTR,ROOTI,N,                                  CPMT
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        CPMT
      COMPLEX A,B,D,E                                                   CPMT
      DIMENSION A(25),B(25),C(MX),ROOTR(MX),ROOTI(MX),D(25),E(25)       CPMT
      COMMON/SUBWRIT/ ISUBNAM                                           CPMT
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     CPMT
  990 FORMAT(1X,*CPMT*)                                                 CPMT
      IF (N.GT.1) GO TO 10                                              CPMT
      C(1)=-ROOTR(1)                                                    CPMT
      C(2)=1.0                                                          CPMT
      RETURN                                                            CPMT
   10 CONTINUE                                                          CPMT
      A(1)=CMPLX(-ROOTR(1),ROOTI(1))                                    CPMT
      A(2)=CMPLX(1.0,0.0)                                               CPMT
      NX=2                                                              CPMT
      DO 4 II=2,N                                                       CPMT
      NY=NX+1                                                           CPMT
      DO 1 I=1,NY                                                       CPMT
      D(I)=CMPLX(0.0,0.0)                                               CPMT
    1 CONTINUE                                                          CPMT
      B(1)=CMPLX(-ROOTR(II),ROOTI(II))                                  CPMT
      B(2)=CMPLX(1.0,0.0)                                               CPMT
      DO 3 I=1,2                                                        CPMT
      DO 3 J=1,NX                                                       CPMT
      K=I+J-1                                                           CPMT
      D(K)=A(J)*B(I)+D(K)                                               CPMT
    3 CONTINUE                                                          CPMT
      NX=NX+1                                                           CPMT
      DO 5 JJ=1,NX                                                      CPMT
      A(JJ)=D(JJ)                                                       CPMT
    5 CONTINUE                                                          CPMT
    4 CONTINUE                                                          CPMT
      DO 7 I=1,NX                                                       CPMT
      E(I)=D(I)                                                         CPMT
    7 CONTINUE                                                          CPMT
      DO 6 I=1,NX                                                       CPMT
      C(I)=REAL(D(I))                                                   CPMT
    6 CONTINUE                                                          CPMT
      RETURN                                                            CPMT
      END                                                               CPMT
```

```
      SUBROUTINE EAT (T,A,W1,W2,W3,C,N,                                 EAT        2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        EAT        3
C                                                                       EAT        4
C     THIS SUBROUTINE COMPUTES THE TRANSITION MATRIX AND IT@S INTEGRAL. EAT        5
C     THE SERIES IS TRUNCATED WHEN THE LARGEST ELEMENT OF THE LAST TERM EAT        6
C     THE SERIES IS LESS THAN 1.E-03 TIMES THE SMALLEST ELEMENT OF THE S EAT       7
C     SERIES.  WRITTEN BY R. MAINE  8/17/71                             EAT        8
C                                                                       EAT        9
      DIMENSION A(MX,MX),W1(MX,MX),W2(MX,MX),W3(MX,MX),C(MX,MX)         EAT       10
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,     EAT       11
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,         EAT       12
     1IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT                  EAT       13
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP EAT       14
      INTEGER DIGITL                                                    EAT       15
      COMMON/SUBWRIT/ ISUBNAM                                           EAT       16
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     EAT       17
  990 FORMAT(1X,*EAT*)                                                  EAT       18
      MAT1=MX                                                           EAT       19
      MAT2=MX                                                           EAT       20
      MAT3=MX                                                           EAT       21
      MAT4=MX                                                           EAT       22
      MAT5=MX                                                           EAT       23
      MAT6=MX                                                           EAT       24
      NT=24                                                             EAT       25
      CALL ZCT1(W1,N,N,                                                 EAT       26
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        EAT       27
      CALL MAKE (W2,W1,N,N,                                             EAT       28
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        EAT       29
      DO 1 I=1,N                                                        EAT       30
      W1(I,I)=1.0                                                       EAT       31
    1 CONTINUE                                                          EAT       32
      CALL MAKE (W3,W1,N,N,                                             EAT       33
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        EAT       34
      G=1.0                                                             EAT       35
      W3MAX= 1.E+50                                                     EAT       36
      T= T/8.                                                           EAT       37
      DO 7 I=1,NT                                                       EAT       38
      BB=I                                                              EAT       39
      G=G*T/BB                                                          EAT       40
      W1MIN= 1.E+50                                                     EAT       41
      W2MIN= 1.E+50                                                     EAT       42
      DO 30 J=1,NX                                                      EAT       43
      DO 30 K=1,NX                                                      EAT       44
      IF (W1(J,K).NE.0.0.AND.ABS(W1(J,K)).LT. W1MIN) W1MIN=ABS(W1(J,K)) EAT       45
      IF (W2(J,K).NE.0.0.AND.ABS(W2(J,K)).LT. W2MIN) W2MIN=ABS(W2(J,K)) EAT       46
   30 CONTINUE                                                          EAT       47
      W3MAX1= W3MAX*T/BB                                                EAT       48
      CALL ADD (1.0,W2,G,W3,W2,N,N,                                     EAT       49
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        EAT       50
      CALL MULT (A,W3,C,N,N,N,                                          EAT       51
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        EAT       52
      CALL MAKE (W3,C,N,N,                                              EAT       53
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        EAT       54
      W3MAX =0.0                                                        EAT       55
      DO 40 J=1,NX                                                      EAT       56
      DO 40 K=1,NX                                                      EAT       57
      IF (ABS(W3(J,K)).GT.W3MAX) W3MAX= (ABS(W3(J,K)))                  EAT       58
```

```
   40 CONTINUE                                                         EAT      59
      W3MAX= W3MAX*G                                                   EAT      60
      CALL ADD (1.0, W1, G, W3, W1, N, N,                              EAT      61
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       EAT      62
      IF (W3MAX1.LT.W2MIN* 1.0E-03 .AND.W3MAX.LT.W1MIN* 1.0E-03) GO TO 70 EAT   63
    7 CONTINUE                                                         EAT      64
      WRITE (3,1000) W1MIN, W3MAX, W2MIN, W3MAX1                       EAT      65
 1000 FORMAT(* ERROR IN EAT*,5X,*W1MIN =*,E15.6,5X,*W3MAX =*,E15.6,/* *, EAT    66
     117X,*W2MIN =*,E15.6,5X,*W3MAX1 =*,E15.6)                         EAT      67
   70 CONTINUE                                                         EAT      68
      DO 90 K=1,3                                                      EAT      69
      CALL MAKE (W3, W1, N, N,                                         EAT      70
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       EAT      71
      CALL MULT (W1, W3, C, N, N, N,                                   EAT      72
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       EAT      73
      CALL MAKE (W1, C, N, N,                                          EAT      74
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       EAT      75
      DO 80 J=1,N                                                      EAT      76
      W3(J,J)= W3(J,J)+1.                                              EAT      77
   80 CONTINUE                                                         EAT      78
      CALL MULT (W2, W3, C, N, N, N,                                   EAT      79
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       EAT      80
      CALL MAKE (W2, C, N, N,                                          EAT      81
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       EAT      82
   90 CONTINUE                                                         EAT      83
      I= I*3.                                                          EAT      84
      WRITE (3,51) I                                                   EAT      85
   51 FORMAT (/*     THE TRANSITION MATRIX *,I5,*  TERMS*/)            EAT      86
      CALL SPIT1 (W1,NX,NX,                                            EAT      87
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       EAT      88
      RETURN                                                           EAT      89
      END                                                              EAT      90
```

```
      SUBROUTINE EIGEN (N,W1,W2,W3,ROOTR,ROOTI,ROTR,ROTI,                EIGEN     2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         EIGEN     3
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,      EIGEN     4
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,          EIGEN     5
     1IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT                   EIGEN     6
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP  EIGEN     7
      INTEGER DIGITL,SCAPLT,ZOH                                          EIGEN     8
      DIMENSION M(10,20),MM(20),P(20),KD(25)                             EIGEN     9
      DIMENSION W1(MX,MX),W2(MX,MX),W3(MX,MX)                            EIGEN    10
      DIMENSION ROOTR(MX),ROOTI(MX),ROTR(MX),ROTI(MX)                    EIGEN    11
C     COMMON/UDIM/MX,MY,MU,MS                                            EIGEN    12
C                                                                        EIGEN    13
C       THIS SUBROUTINE FINDS THE EIGENVALUES OF THE INPUT MATRIX (W1)   EIGEN    14
C                                                                        EIGEN    15
C       NSCALE=0  NO PRECONDITIONING OF THE INPUT MATRIX IS DONE         EIGEN    16
C                                                                        EIGEN    17
C       NSCALE=1  SCALE IS CALLED TO SCALE THE INPUT MATRIX BY A         EIGEN    18
C                 DIAGONAL SIMILARITY TRANSFORMATION. THEN THE           EIGEN    19
C                 EIGENVALUES OF THE IRREDUCIBLE SUBMATRICES ARE         EIGEN    20
C                 DETERMINED.                                            EIGEN    21
C       NSCALE=2  REDUCE IS CALLED TO DETERMINE THE IRREDUCIBLE          EIGEN    22
C                 SUBMATRICES OF W1. THEN THE EIGENVALUES OF THE         EIGEN    23
C                 INDIVIDUAL SUBMATRICES ARE DETERMINED.                 EIGEN    24
C                                                                        EIGEN    25
C                                                                        EIGEN    26
      COMMON/SUBWRIT/ ISUBNAM                                            EIGEN    27
      IF(ISUBNAM.GE.2) WRITE(3,990)                                      EIGEN    28
  990 FORMAT(1X,*EIGEN*)                                                 EIGEN    29
      IPRNT=0                                                            EIGEN    30
      IF (NSCALE.EQ.0) GO TO 50                                          EIGEN    31
      IF (NSCALE.EQ.2) GO TO 100                                         EIGEN    32
      CALL ASCALE (N,J,M,MM,P,W1,W2,W3,ROOTR,                            EIGEN    33
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         EIGEN    34
      GO TO 101                                                          EIGEN    35
  100 CALL REDUCE (N,J,MM,M,W1,W2,W3,                                    EIGEN    36
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         EIGEN    37
      GO TO 101                                                          EIGEN    38
   50 DO 51 I=1,N                                                        EIGEN    39
      M(1,I)=I                                                           EIGEN    40
   51 CONTINUE                                                           EIGEN    41
      MM(1)=N                                                            EIGEN    42
      J=1                                                                EIGEN    43
  101 CONTINUE                                                           EIGEN    44
      KRUNT=0                                                            EIGEN    45
      DO 200 I=1,J                                                       EIGEN    46
      NS=MM(I)                                                           EIGEN    47
      DO 110 K=1,NS                                                      EIGEN    48
      DO 110 L=1,NS                                                      EIGEN    49
      W2(K,L)=W1(M(I,K),M(I,L))                                          EIGEN    50
  110 CONTINUE                                                           EIGEN    51
      CALL HESSEN (W2,NS,ROTR,                                           EIGEN    52
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         EIGEN    53
      CALL QREIG (NS,ROTR,ROTI,IPRNT,W2,                                 EIGEN    54
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         EIGEN    55
      IF(SYSTEM.EQ.3) GO TO 3                                            EIGEN    56
      IF(CONTUR.EQ.1) GO TO 3                                            EIGEN    57
      GO TO 5                                                            EIGEN    58
```

```
      4 IF(FORM.EQ.3) GO TO 5                                         EIGEN    59
        J3=0                                                          EIGEN    60
        DO 418 J4=1,N5                                                EIGEN    61
        IF(ROTI(J4).LT.0.) GO TO 418                                  EIGEN    62
        J3=J3+1                                                       EIGEN    63
        KD(J3)=J4                                                     EIGEN    64
    418 CONTINUE                                                      EIGEN    65
        WRITE(7)J3,(ROTR(KD(K)),ROTI(KD(K)),K=1,J3)                   EIGEN    66
        IF(FORM.EQ.2) GO TO 7                                         EIGEN    67
      6 WRITE (3,300) (ROTR(II),ROTI(II),II=1,N5)                     EIGEN    68
      7 CONTINUE                                                      EIGEN    69
    300 FORMAT (/(2E30.8))                                            EIGEN    70
        DO 120 K=1,N5                                                 EIGEN    71
        ROOTR(KRUNT+K)=ROTR(K)                                        EIGEN    72
        ROOTI(KRUNT+K)=ROTI(K)                                        EIGEN    73
    120 CONTINUE                                                      EIGEN    74
        KRUNT=KRUNT+N5                                                EIGEN    75
    200 CONTINUE                                                      EIGEN    76
        RETURN                                                        EIGEN    77
        END                                                           EIGEN    78
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE FRQRSP (NNUM,NN,GAIN,IMOD,ROOTR,ROOTI,ROTR,ROTI,SAV1,  FRQRSP       2
     1SAV2,INY,INU,                                                     FRQRSP       3
     2MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        FRQRSP       4
      REAL IFREQ,M                                                      FRQRSP       5
      DIMENSION ROOTR(MX),ROOTI(MX),ROTR(MX),ROTI(MX),SAV1(MS),SAV2(MS) FRQRSP       6
      COMPLEX RN1,RN2,RD1,RD2                                           FRQRSP       7
      INTEGER TWO, FOUR                                                 FRQRSP       8
      COMMON/LABEL/INPT,OUTPT,TITLE                                     FRQRSP       9
      REAL   INPT(10), OUTPT(20),TITLE(8)                               FRQRSP      10
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,     FRQRSP      11
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,         FRQRSP      12
     1IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT                  FRQRSP      13
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP FRQRSP      14
      INTEGER DIGITL, SCAPLT, ZOH                                       FRQRSP      15
      COMMON/ACOND/ DELT,FINALT,IFREQ,FFREQ,DELFRQ,GAIN1,GAIN2,M        FRQRSP      16
      COMMON/SUBWRIT/ ISUBNAM                                           FRQRSP      17
      REAL PST                                                          FRQRSP      18
      DATA TWO/2/,FOUR/4/                                               FRQRSP      19
      DATA PST/10HFREQ      /                                           FRQRSP      20
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     FRQRSP      21
  990 FORMAT(1X,*FRQRSP*)                                               FRQRSP      22
      IF(FORM.GT.0) WRITE(7)PST,TITLE,SYSTEM,MODEL,DIGITL,SCAPLT        FRQRSP      23
      J=1                                                               FRQRSP      24
      IF (IMOD.EQ.2) GO TO 70                                           FRQRSP      25
      IF (DIGITL.NE.0) GO TO 200                                        FRQRSP      26
      WRITE (3,71) OUTPT(INY),INPT(INU)                                 FRQRSP      27
   71 FORMAT (//5X,A10*/*A10,*FREQUENCY RESPONSE  S-PLANE*//* FREQUENCY FRQRSP      28
     1*,6X,*AMPLITUDE RATIO*,9X,*PHASE ANGLE*/*  RAD/SEC*,12X,*DB*,17X,* FRQRSP      29
     2DEGREES*/)                                                        FRQRSP      30
      GO TO 75                                                          FRQRSP      31
  200 WRITE (3,76) OUTPT(INY),INPT(INU)                                 FRQRSP      32
   76 FORMAT (//5X,A10,*/*A10,* FREQUENCY RESPONSE  W-PLANE*//*  W-PLANE FRQRSP      33
     1   S-PLANE*/* FREQUNCY      FREQUENCY       AMPLITUDE RATIO       FRQRSP      34
     2PHASE ANGLE*/12X,*  RAD/SEC*,17X,*DB*,17X,*DEGREES*/)             FRQRSP      35
   75 IF (FORM.GT.0) WRITE (7)TWO,OUTPT(INY),INPT(INU)                  FRQRSP      36
      GO TO 72                                                          FRQRSP      37
   70 IF (DIGITL.NE.0) GO TO 201                                        FRQRSP      38
      WRITE (3,73)  OUTPT(INY),INPT(INU),OUTPT(INY),OUTPT(INY-1)        FRQRSP      39
   73 FORMAT (//6X,*S-PLANE*,A10,*/*,A10,* FREQUENCY RESPONSE*3X,A10,*/* FRQRSP      40
     1A10,*FREQUENCY RESPONSE*/* FREQUENCY     AMPLITUDE RATIO*9X,*PHASE FRQRSP      41
     2ANGLE  AMPLITUDE RATIO*9X,*PHASE ANGLE*/*  RAD/SEC*10X,*DB*17X,*DE FRQRSP      42
     3GREES*,13X,*DB*,17X,*DEGREES*/)                                   FRQRSP      43
      GO TO 77                                                          FRQRSP      44
  201 WRITE (3,78) OUTPT(INY),INPT(INU),OUTPT(INY),OUTPT(INY-1)         FRQRSP      45
   78 FORMAT (//6X,*W-PLANE  *,A10,*/*,A10,* FREQUENCY RESPONSE*3X,A10,* FRQRSP      46
     1/*,A10,* FREQUENCY RESPONSE*/* W-PLANE   S-PLANE*/* FREQUENCY     FRQRSP      47
     2AMPLITUDE RATIO*9X,*PHASE ANGLE  AMPLITUDE RATIO*9X,*PHASE ANGLE*/ FRQRSP      48
     3*  RAD/SEC*10X,*DB*17X,*DEGREES*,13X,*DB*,18X,*DEGREES*/)         FRQRSP      49
   77 IF (FORM.GT.0) WRITE(7)FOUR,OUTPT(INY),INPT(INU),OUTPT(INY),OUTPT FRQRSP      50
     1(INY-1)                                                           FRQRSP      51
   72 CONTINUE                                                          FRQRSP      52
      FREQ=.1/1.15                                                      FRQRSP      53
      IF (DIGITL.EQ.0.OR.INY.GT.1.OR.INU.GT.1.OR.FRPS.EQ.-1) GO TO 91   FRQRSP      54
      IF (IFREQ.EQ.0) GO TO 90                                          FRQRSP      55
      IFREQ= TAN(IFREQ*DELT*.5)                                         FRQRSP      56
      FFREQ= TAN(FFREQ*DELT*.5)                                         FRQRSP      57
      GO TO 91                                                          FRQRSP      58
```

```
 40 FFREQ= TAN(87./57.3)
    IFREQ= TAN(FREQ     * DELT*.5)
    DELFRQ= 1.15
 91 CONTINUE
100 CONTINUE
    IF (IFREQ.EQ.0..AND.DIGITL.EQ.0) GO TO 80
    IF (IFREQ.NE.0..AND.DIGITL.EQ.0) GO TO 81
    IF (IFREQ.EQ.0..AND.DIGITL.NE.0) GO TO 82
    IF (IFREQ.NE.0..AND.DIGITL.NE.0) GO TO 83
 80 FREQ=1.15*FREQ
    IF (J.EQ.1) FFREQ=150.
    GO TO 84
 81 IF (J.EQ.1) GO TO 85
    FREQ=FREQ*DELFRQ
    GO TO 84
 85 FREQ=IFREQ
    GO TO 84
 82 IF (FRPS.EQ.-1) GO TO 80
    IF (J.EQ.1) GO TO 85
    FREQ=1.15*FREQ
    GO TO 84
 86 FREQ-TAN(FREQ     *DELT*.5)
    GO TO 84
 83 IF (J.EQ.1) GO TO 85
    FREQ=FREQ*DELFRQ
 84 CONTINUE
    RN1=CMPLX (1.0,0.0)
    RD1=CMPLX (1.0,0.0)
    XR= 0.0
    XI= FREQ
    DO 20 I=1,NN
    IF (FRPS.NE.-1) GO TO 21
    XR= COS(FREQ*DELT)
    XI= SIN(FREQ*DELT)
 21 IF (I.GT.NNUM) GO TO 5
    RN2= CMPLX(XR-ROTR(I), XI-ROTI(I))
    RN1= RN1*RN2
  5 RD2= CMPLX(XR-ROOTR(I), XI-ROOTI(I))
    RD1=RD1*RD2
 20 CONTINUE
    RD1=GAIN*RN1/RD1
    PHI=TANG(RD1)
    REAL1=REAL(RD1)
    AMAG=AIMAG(RD1)
    AMPRAT =SQRT(AMAG**2+REAL1**2)
    DB=20.*ALOG10(AMPRAT)
    IF (IMCO.EQ.0) GO TO 30
    IF (IMCO.EQ.2) GO TO 40
    SAV1(J)=DB
    SAV2(J)=PHI
 30 IF(FORM.EQ.2) GO TO 6
    IF (DIGITL.NE. 0    .AND. FRPS .NE. -1) GO TO 79
    WRITE (3,50) FREQ,DB,PHI
    GO TO 11
 79 OMEGA= ATAN(FREQ)*2./DELT
    WRITE (3,51) FREQ,OMEGA,DB,PHI
 11 CONTINUE
```

```
      IF(FORM.EQ.0) GO TO 60
    6 WRITE (7) FREQ,DB,PHI
  121 CONTINUE
      GO TO 60
   40 A=DB-SAV1(J)
      B=PHI-SAV2(J)
      IF(FORM.EQ.2) GO TO 7
      IF (DIGITL.NE. 0   .AND. FRPS .NE. -1) GO TO 12
      WRITE (3,50) FREQ,DB,PHI,A,B
      GO TO 13
   12 WRITE (3,51) FREQ,OMEGA,DB,PHI,A,B
   13 CONTINUE
      IF(FORM.EQ.0) GO TO 60
    7 WRITE(7) FREQ,DB,PHI,A,B
  131 CONTINUE
   50 FORMAT (F10.4,4E20.3)
   51 FORMAT (2F10.4,4E20.5)
   60 J=J+1
      IF (FREQ.LE.FFREQ) GO TO 100
   10 CONTINUE
      IF(FORM.EQ.0) GO TO 8
      JD=99
      IF(IMOD.EQ.2) WRITE(7)JD,DB,PHI,A,B
      IF(IMOD.NE.2) WRITE(7)JD,DB,PHI
    8 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE HESSEN (A,M,B,                                         HESSEN     2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        HESSEN     3
      DIMENSION B(MX),A(MX,MX)                                          HESSEN     4
      COMMON/SUBWRIT/ ISUBNAM                                           HESSEN     5
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     HESSEN     6
  990 FORMAT(1X,*HESSEN*)                                               HESSEN     7
      IF (M-2) 30,30,32                                                 HESSEN     8
   32 DO 40 LC = 3,M                                                    HESSEN     9
      N = M - LC + 3                                                    HESSEN    10
      N1 = N - 1                                                        HESSEN    11
      N2 = N - 2                                                        HESSEN    12
      NI = N1                                                           HESSEN    13
      DIV= ABS(A(N,N-1))                                                HESSEN    14
      DO 2 J = 1,N2                                                     HESSEN    15
      IF( ABS(A(N,J))-DIV) 2,2,1                                        HESSEN    16
    1 NI = J                                                            HESSEN    17
      DIV=  ABS(A(N,J))                                                 HESSEN    18
    2 CONTINUE                                                          HESSEN    19
      IF(DIV) 3,40,3                                                    HESSEN    20
    3 IF(NI - N1) 4, 7,4                                                HESSEN    21
    4 DO 5 J = 1,M                                                      HESSEN    22
      DIV = A(J,NI)                                                     HESSEN    23
      A(J,NI) = A(J,N1)                                                 HESSEN    24
      A(J,N1) = DIV                                                     HESSEN    25
    5 CONTINUE                                                          HESSEN    26
      DO 6 J = 1,M                                                      HESSEN    27
      DIV = A(NI,J)                                                     HESSEN    28
      A(NI,J) = A(N1,J)                                                 HESSEN    29
      A(N1,J) = DIV                                                     HESSEN    30
    6 CONTINUE                                                          HESSEN    31
    7 DO 26 K = 1, N1                                                   HESSEN    32
      B(K) = A(N,K)/A(N,N-1)                                            HESSEN    33
   26 CONTINUE                                                          HESSEN    34
      DO 45 J = 1,M                                                     HESSEN    35
      SUM = 0.0                                                         HESSEN    36
      IF (J - N1) 46,43,43                                              HESSEN    37
   46 IF(B(J))   41,43,41                                               HESSEN    38
   41 A(N,J) = 0.0                                                      HESSEN    39
      DO 42 K = 1,N1                                                    HESSEN    40
      A(K,J) = A(K,J) - A(K,N1)*B(J)                                    HESSEN    41
      SUM = SUM + A(K,J)*B(K)                                           HESSEN    42
   42 CONTINUE                                                          HESSEN    43
      GO TO 45                                                          HESSEN    44
   43 DO 44 K = 1,N1                                                    HESSEN    45
      SUM = SUM + A(K,J)*B(K)                                           HESSEN    46
   44 CONTINUE                                                          HESSEN    47
   45 A(N1,J) = SUM                                                     HESSEN    48
   40 CONTINUE                                                          HESSEN    49
   30 RETURN                                                            HESSEN    50
       END                                                              HESSEN    51
```

```
      SUBROUTINE INPUTV(DELT,T,U,                                       INPUTV     2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        INPUTV     3
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,     INPUTV     4
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,         INPUTV     5
     2IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT                  INPUTV     6
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP INPUTV     7
      INTEGER DIGITL,SCAPLT,ZOH                                         INPUTV     8
      DIMENSION U(MX)                                                   INPUTV     9
C                                                                       INPUTV    10
C     USER WRITTEN SUBROUTINE CONSTRUCTING INPUT VECTOR FOR TRANSIENT   INPUTV    11
C     RESPONSE.                                                         INPUTV    12
C                                                                       INPUTV    13
      COMMON/SUBWRIT/ ISUBNAM                                           INPUTV    14
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     INPUTV    15
  990 FORMAT(1X,*INPUTV*)                                               INPUTV    16
      IF (T.GT.0.0) RETURN                                              INPUTV    17
      READ (1,1) (U(I),I=1,NU)                                          INPUTV    18
      IF (EOF(1).NE.0) STOP                                             INPUTV    19
    1 FORMAT (8F10.4)                                                   INPUTV    20
      RETURN                                                            INPUTV    21
      END                                                               INPUTV    22
```

```
      SUBROUTINE INVR (A,B,JJJ,IT,                                      INVR       2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        INVR       3
C  PROGRAM AUTHORS R.E. FUNDERIC AND R.G. EDWARDS,                      INVR       4
C  COMPUTING TECHNOLOGY CENTER, UNION CARBIDE CORP., NUCLEAR DIV.,      INVR       5
C  OAK RIDGE, TENN.                                                     INVR       6
C                                                                       INVR       7
C  CTC ORD PROGRAM NO. 9067.1                                           INVR       8
      DIMENSION A(MX,MX),B(MX,MX)                                       INVR       9
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,     INVR      10
     1CONTUR,NUMBRS,FRPS,TRESP,MODEL,NSCALE,SAV,CHAT,NK2,IFLAG,         INVR      11
     1IGO,FORM,IPT,READ3,MIXSQ,MULTRT,SCAPLT,ZOH,KOUNT                  INVR      12
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CHAT,READ3, FRPS,TRESP INVR      13
      INTEGER DIGITL, SCAPLT, ZOH                                       INVR      14
      COMMON/SUBWRIT/ ISUBNAM                                           INVR      15
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     INVR      16
  990 FORMAT(1X,*INVR*)                                                 INVR      17
      MAT1=MX                                                           INVR      18
      MAT2=MX                                                           INVR      19
      IF (IPT.LT.1) GO TO 70                                            INVR      20
      WRITE (3,71)                                                      INVR      21
   71 FORMAT (/*    MATRIX ENTERING INVR AND ITS INVERSE*/)             INVR      22
      CALL SPIT1 (A,JJJ,JJJ,                                            INVR      23
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        INVR      24
   70 CONTINUE                                                          INVR      25
      IF (JJJ.NE.1) GO TO 50                                            INVR      26
      B(1,1)=1./A(1,1)                                                  INVR      27
      RETURN                                                            INVR      28
   50 CONTINUE                                                          INVR      29
      DO 21 I=1,JJJ                                                     INVR      30
      DO 20 J=1,JJJ                                                     INVR      31
      B(I,J)=0.0                                                        INVR      32
   20 CONTINUE                                                          INVR      33
      B(I,I)=1.0                                                        INVR      34
   21 CONTINUE                                                          INVR      35
      KK=JJJ                                                            INVR      36
      NV=JJJ                                                            INVR      37
      D=1.                                                              INVR      38
      IF (JJJ.LT.0)D=0.                                                 INVR      39
      KKM=KK-1                                                          INVR      40
      DO 9 I=1,KKM                                                      INVR      41
      S=0.0                                                             INVR      42
      DO 1 J=I,KK                                                       INVR      43
      R=ABS(A(J,I))                                                     INVR      44
      IF (R.LT.S) GO TO 1                                               INVR      45
      S=R                                                               INVR      46
      L=J                                                               INVR      47
    1 CONTINUE                                                          INVR      48
      IF (L.EQ.I) GO TO 5                                               INVR      49
      DO 2 J=I,KK                                                       INVR      50
      S=A(I,J)                                                          INVR      51
      A(I,J)=A(L,J)                                                     INVR      52
      A(L,J)=S                                                          INVR      53
    2 CONTINUE                                                          INVR      54
      IF(NV.LE.0)GO TO 4                                                INVR      55
      DO 3 J=1,NV                                                       INVR      56
      S=B(I,J)                                                          INVR      57
      B(I,J)=B(L,J)                                                     INVR      58
```

```
      B(L,J)=0                                                          INVR      59
    3 CONTINUE                                                          INVR      60
    4 D=-D                                                              INVR      61
    5 IF(A(I,I).EQ.0.)GO TO 9                                           INVR      62
      IPO=I+1                                                           INVR      63
      DO 8  J=IPO,KK                                                    INVR      64
      IF (A(J,I).EQ.0.) GO TO 8                                         INVR      65
      S=A(J,I)/A(I,I)                                                   INVR      66
      A(J,I)=0.0                                                        INVR      67
      DO 6  K=IPO,KK                                                    INVR      68
      A(J,K)=A(J,K)-A(I,K)*S                                            INVR      69
    6 CONTINUE                                                          INVR      70
      IF (NV.LE.0) GO TO 8                                              INVR      71
      DO 7 K=1,NV                                                       INVR      72
      B(J,K)=B(J,K)-B(I,K)*S                                            INVR      73
    7 CONTINUE                                                          INVR      74
    8 CONTINUE                                                          INVR      75
    9 CONTINUE                                                          INVR      76
      DO 10 I=1,KK                                                      INVR      77
      D=D*A(I,I)                                                        INVR      78
   10 CONTINUE                                                          INVR      79
      IF(NV.LE.0)GO TO 13                                               INVR      80
      KMO=KK-1                                                          INVR      81
      DO 12 K=1,NV                                                      INVR      82
      B(KK,K)=B(KK,K)/A(KK,KK)                                          INVR      83
      DO 12 I=1,KMO                                                     INVR      84
      N=KK-I                                                            INVR      85
      DO 11 J=N,KMO                                                     INVR      86
      B(N,K)=B(N,K)-A(N,J+1)*B(J+1,K)                                   INVR      87
   11 CONTINUE                                                          INVR      88
      B(N,K)=B(N,K)/A(N,N)                                              INVR      89
   12 CONTINUE                                                          INVR      90
   13 DMAT=D=D                                                          INVR      91
      IF (IPT.LT.1) GO TO 72                                            INVR      92
      CALL SPIT1 (B,JJJ,JJJ,                                            INVR      93
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        INVR      94
   72 CONTINUE                                                          INVR      95
      IF (IT.EQ.0) RETURN                                               INVR      96
      DO 30 I=1,JJJ                                                     INVR      97
      DO 30 J=1,JJJ                                                     INVR      98
      IF (ABS(B(I,J)).LT.1.E-5)B(I,J)=0.                                INVR      99
   30 CONTINUE                                                          INVR     100
      RETURN                                                            INVR     101
      END                                                               INVR     102
```

```
      SUBROUTINE LOAD     (A,B,C,H,G,F,K1,K2,K3,K4,D,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
C
C       THIS SUBROUTINE LOADS ALL MATRICES ACCORDING TO THE
C       PARAMETERS, SYSTEM AND OUTPUT, USING THE SUBROUTINE LOAD1
C
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,
     1CONTUR,NUMERS,ERPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,
     1IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, ERPS,TRESP
      INTEGER DIGITL, SCAPLT, ZOH
      REAL K1,K2,K3,K4
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),
     1K1(MU,MX),K2(MU,MX),K3(MU,MX),K4(MU,MX),D(MU,MU)
      COMMON/SUBWRIT/ ISUBNAM
      IF(ISUBNAM.GE.2) WRITE(3,990)
  990 FORMAT(1X,*LOAD*)
      MAT1=MX
      MAT2=MX
      NMAT=0
      CALL LOAD1 (A,NX,NX,NMAT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MU
      CALL LOAD1 (B,NX,NU,NMAT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (CMAT.EQ.0) GO TO 20
      MAT1=MX
      MAT2=MX
      CALL LOAD1 (C,NX,NX,NMAT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      GO TO 40
   20 MAT2=MX
      CALL ZCT1(C,NX,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      DO 21 I=1,NX
   21 C(I,I)=1.0
   40 CONTINUE
      IF (MIXED.EQ.1) GO TO 50
      GO TO  (50,50,60),SYSTEM
   60 MAT1=MU
      MAT2=MX
      CALL LOAD1 (K1,NU,NX,NMAT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (NK2.EQ.0) GO TO 62
      CALL LOAD1 (K2,NU,NX,NMAT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
   62 CONTINUE
      IF (N2.EQ.0) GO TO 64
      CALL LOAD1 (K3,NU,NX,NMAT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (NK2.EQ.0) GO TO 64
      CALL LOAD1 (K4,NU,NX,NMAT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
   64 CONTINUE
      GO TO 200
   50 MAT1=MY
      MAT2=MX
```

```
      CALL LOAD1 (H,NY,NX,NMAT,                                         LOAD  59
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        LOAD  60
      GO TO(100,56,57,58),OUTPUT                                        LOAD  61
   56 CALL LOAD1 (G,NY,NX,NMAT,                                         LOAD  62
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        LOAD  63
      GO TO 100                                                         LOAD  64
   57 MAT2=MU                                                           LOAD  65
      CALL LOAD1 (F,NY,NU,NMAT,                                         LOAD  66
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        LOAD  67
      GO TO 100                                                         LOAD  68
   58 CALL LOAD1 (G,NY,NX,NMAT,                                         LOAD  69
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        LOAD  70
      MAT2=MU                                                           LOAD  71
      CALL LOAD1 (F,NY,NU,NMAT,                                         LOAD  72
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        LOAD  73
  100 IF (MIXED .EQ. 1) GO TO 200                                       LOAD  74
      GO TO (200,110,200),SYSTEM                                        LOAD  75
  110 MAT1=MU                                                           LOAD  76
      MAT2=MX                                                           LOAD  77
      CALL LOAD1 (K1,NU,NX,NMAT,                                        LOAD  78
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        LOAD  79
      IF (NK2.EQ.0) GO TO 66                                            LOAD  80
      CALL LOAD1 (K2,NU,NX,NMAT,                                        LOAD  81
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        LOAD  82
   66 CONTINUE                                                          LOAD  83
      MAT2=MU                                                           LOAD  84
      CALL LOAD1 (D,NU,NU,NMAT,                                         LOAD  85
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        LOAD  86
  200 RETURN                                                            LOAD  87
      END                                                               LOAD  88
```

```
      SUBROUTINE LOAD1 (A,N,M,NMAT,                                     LOAD1          2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        LOAD1          3
      DIMENSION A(MAT1,MAT2)                                            LOAD1          4
      COMMON/SUBWRIT/ ISUBNAM                                           LOAD1          5
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     LOAD1          6
  990 FORMAT(1X,*LOAD1*)                                                LOAD1          7
      NMAT=NMAT+1                                                       LOAD1          8
   10 FORMAT (2I10)                                                     LOAD1          9
      READ (1,10) N1,N2                                                 LOAD1         10
      IF (EOF(1).NE.0) STOP                                             LOAD1         11
      IF (N1.EQ.N.AND.N2.EQ.M) GO TO 20                                 LOAD1         12
      WRITE (3,100) NMAT,N,M,N1,N2                                      LOAD1         13
  100 FORMAT (///,10X,* WARNING  DIMENSION OF NUMBER*,I2,* MATRIX SHOULD LOAD1         14
     1 BE*,I5,* BY*,I2,*  BUT IS*,I5,* BY*,I2,//)                       LOAD1         15
   20 DO 30 I=1,N                                                       LOAD1         16
      READ (1,200) (A(I,J),J=1,M)                                       LOAD1         17
      IF (EOF(1).NE.0) STOP                                             LOAD1         18
   30 CONTINUE                                                          LOAD1         19
  200 FORMAT (8F10.4)                                                   LOAD1         20
      RETURN                                                            LOAD1         21
      END                                                               LOAD1         22
```

```
SUBROUTINE MAKE         73/74    OPT=1                                    FTN 4.2+75060        01/09/76  14.12.48.

                      SUBROUTINE MAKE (A,B,N,M,                                          MAKE       2
                     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         MAKE       3
                      DIMENSION A(MAT1,MAT2),B(MAT3,MAT4)                                MAKE       4
                      COMMON/SUBWRIT/ ISUBNAM                                            MAKE       5
                      IF(ISUBNAM.GE.2) WRITE(3,990)                                      MAKE       6
                  990 FORMAT(1X,*MAKE*)                                                  MAKE       7
                      DO 10 I=1,N                                                        MAKE       8
                      DO 10 J=1,M                                                        MAKE       9
                      A(I,J)=B(I,J)                                                      MAKE      10
                   10 CONTINUE                                                           MAKE      11
                      RETURN                                                             MAKE      12
                      END                                                                MAKE      13
```

```
      SUBROUTINE MATRIX (A,B,C,H,G,F,K1,K2,K3,K4,D,W1,W2,W3,             MATRIX      2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         MATRIX      3
      COMMON/COND/READ,SYSTEM,OUTPUT,NK,NY,NU,NKC,NUC,N1,N2,DIGITL,      MATRIX      4
     1CONTUR ,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,         MATRIX      5
     1IGC,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT                   MATRIX      6
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,FRPS,TRESP,READ3   MATRIX      7
      INTEGER DIGITL, SCAPLT, ZOH                                        MATRIX      8
      COMMON/ACOND/ DELT,FINALT,IFREQ,FFREQ,DELFRQ,GAIN1,GAIN2,M         MATRIX      9
      REAL K1,K2,K3,K4,IFREQ,M                                           MATRIX     10
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),   MATRIX     11
     1K1(MU,MX),K2(MU,MX),K3(MU,MX),K4(MU,MX),D(MU,MU),                  MATRIX     12
     2W1(MX,MX),W2(MX,MX),W3(MX,MX)                                      MATRIX     13
      DIMENSION GRAPH(20,5),BLOCK(20,3),NUMER(20,5),DENOM(20,5),         MATRIX     14
     XGAIN(20),STATE(20,4),ITHINY(30),ITHINU(20),YTOV(20,2),             MATRIX     15
     X  ZTOU(20,2),NXYU(8),YZTOK(20,2)                                   MATRIX     16
      REAL NUMER                                                         MATRIX     17
      INTEGER GRAPH,BLOCK,STATE,YTOV,ZTOU,YZTOK                          MATRIX     18
      COMMON /BLKDAT/ NUMER,DENOM,GAIN,GRAPH,BLOCK,STATE,YTOV,           MATRIX     19
     X  ZTOU,YZTOK,ITHINY,ITHINU,NBLOCK,NYTOV,NZTOU,NXYU,NYZTOK,         MATRIX     20
     X  NXT,NYT,NUT,NY1,NU1                                              MATRIX     21
C                                                                        MATRIX     22
C     USER WRITTEN SUBROUTINE TO CONSTRUCT SYSTEM MATRICES UNDER CONTROL MATRIX     23
C     OF CONDITION CODES                                                 MATRIX     24
C                                                                        MATRIX     25
      COMMON/SUBWRIT/ ISUBNAM                                            MATRIX     26
      IF(ISUBNAM.GE.2) WRITE(3,990)                                      MATRIX     27
  990 FORMAT(1X,*MATRIX SUB 2*)                                          MATRIX     28
      RETURN                                                             MATRIX     29
      END                                                                MATRIX     30
```

```
      SUBROUTINE MULT (A,B,C,N,M,K,                                     MULT       2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        MULT       3
      DIMENSION A(MAT1,MAT2),B(MAT3,MAT4),C(MAT5,MAT6)                  MULT       4
      COMMON/SUBWRIT/ ISUBNAM                                           MULT       5
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     MULT       6
  990 FORMAT(1X,*MULT*)                                                 MULT       7
      DO 10 I=1,N                                                       MULT       8
      DO 10 L=1,K                                                       MULT       9
      XX=0.0                                                            MULT      10
      DO 11 J=1,M                                                       MULT      11
      XX=XX+A(I,J)*B(J,L)                                               MULT      12
   11 CONTINUE                                                          MULT      13
      C(I,L)=XX                                                         MULT      14
   10 CONTINUE                                                          MULT      15
      RETURN                                                            MULT      16
      END                                                               MULT      17
```

```
      SUBROUTINE NMRATR (NN,A,B,C,H,G,F,D,ROOTR,ROOTI,ROTR,ROTI,Z,V,
     1W1,W2,W3,SAV1,SAV2,
     2MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5)
C
C      THIS SUBROUTINE DETERMINES THE NUMERATORS OF TRANSFER
C      FUNCTIONS BY FINDING THE MATRIX WHOSE EIGENVALUES ARE
C     THE DESIRED Z ROOTS. SUBROUTINE EIGEN IS CALLED TO FIND
C      THE EIGENVALUES
C      CORRECTION MADE BY G. NORRIS JULY 5 73
C      V AND ROTR, AND Z AND ROTI ARE SAME MATRIX
C
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,
     1CONTUR,NUMER,FRES,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,
     2IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KCUNT
      INTEGER READ, SYSTEM, OUTPUT, FORM, CONTUR, SAV, CMAT, READ3,
     1 FRES, TRESP
      INTEGER DIGITL,SCAPLT,ZOH
      COMMON/ACOND/ DELT,FINALT,IFREQ,FFREQ,DELFRQ,GAIN1,GAIN2,MN
      REAL    INPT(10), OUTPT(20),TITLE(8)
      COMMON/LABEL/INPT,OUTPT,TITLE
      REAL IFREQ,MN
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),
     1D(MU,MU),
     2W1(MX,MX),W2(MX,MX),W3(MX,MX),ROOTR(MX),ROOTI(MX),ROTR(MX),
     3ROTI(MX),Z(MX),V(MX),SAV1(MS),SAV2(MS)
C     DIMENSION A(15,15), B(15,10), C(15,15), H(15,15), G(15,15),
C    1 F(15,10), D(10,10)
C     DIMENSION W1(15,15), W2(15,15), W3(15,15)
C     DIMENSION ROOTR(15), ROOTI(15), ROTR(15), ROTI(15)
C     DIMENSION SAV1(200), SAV2(200)
C     DIMENSION V(15), Z(15)
      COMMON/SUBWRIT/ ISUBNAM
      INTEGER ONE
      DATA ONE /1/,DUMY/0.0/
      IF(ISUBNAM.GE.2) WRITE(3,990)
  990 FORMAT(1X,*NMRATR*)
      DO 300 JJ=1,NU
      IMOD=0
      DO 300 I=1,NY
      N=NN
      DO 500 IX=1,NX
      IF (H(I,IX).NE.0.0) GO TO 501
  500 CONTINUE
      WRITE (3,502)I
  502 FORMAT (//10X,*ROW *,I2,* OF H MATRIX IS NULL *//)
      WRITE(7)ONE,DUMY,DUMY
      GO TO 300
  501 CONTINUE
      IF (F(I,JJ).EQ.0.0) GO TO 410
      NNUM=NN
      GO TO 431
  410 DO 420 K=1,NN
      ROTR(K)=0.
      IF (H(I,K).NE.0.0)ROTR(K)=1.
  420 CONTINUE
      DO 430 LL=1,NN
      DO 421 K=1,NN
```

```
      IF (ROTR(K).EQ.0.0.OR.B(K,JJ).EQ.0.0) GO TO 421                   NMRATR    59
      NNUM=NN-LL                                                        NMRATR    60
      GO TO 431                                                         NMRATR    61
  421 CONTINUE                                                          NMRATR    62
      ITEST=0                                                           NMRATR    63
      DO 442 K=1,NN                                                     NMRATR    64
      ROTI(K)=0.                                                        NMRATR    65
  442 CONTINUE                                                          NMRATR    66
      DO 422 K=1,NN                                                     NMRATR    67
      DO 422 L=1,NN                                                     NMRATR    68
      IF (K.EQ.L) GO TO 422                                             NMRATR    69
      IF (ROTR(K).EQ.0.0.OR.A(K,L).EQ.0.0) GO TO 422                    NMRATR    70
      ROTI(L)=1.                                                        NMRATR    71
      ITEST=1                                                           NMRATR    72
  422 CONTINUE                                                          NMRATR    73
      IF (ITEST.EQ.0) GO TO 440                                         NMRATR    74
      DO 423 K=1,NN                                                     NMRATR    75
      ROTR(K)=ROTI(K)                                                   NMRATR    76
  423 CONTINUE                                                          NMRATR    77
  430 CONTINUE                                                          NMRATR    78
  440 WRITE (3,441) JJ,I                                                NMRATR    79
  441 FORMAT (/10X,*NUMBER*,I4,*  INPUT DOES NOT EXCITE NUMBER*,I4,*  OU NMRATR    80
     1TPUT*/)                                                           NMRATR    81
      GO TO 300                                                         NMRATR    82
  431 W2(1,1)=F(I,JJ)                                                   NMRATR    83
      MM=N+1                                                            NMRATR    84
      DO 5 K=2,MM                                                       NMRATR    85
      W2(1,K)=-H(I,K-1)                                                 NMRATR    86
      W2(K,1)=B(K-1,JJ)                                                 NMRATR    87
      DO 4 L=2,MM                                                       NMRATR    88
      W2(K,L)=-A(K-1,L-1)                                               NMRATR    89
    4 CONTINUE                                                          NMRATR    90
    5 CONTINUE                                                          NMRATR    91
      IF (IPT.LT.2) GO TO 240                                           NMRATR    92
      WRITE (3,241)                                                     NMRATR    93
  241 FORMAT (/* NUMERATOR MATRICES*/)                                  NMRATR    94
      MAT1=MX                                                           NMRATR    95
      MAT2=MX                                                           NMRATR    96
      CALL PRIT1 (W2,MM,MM,                                             NMRATR    97
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        NMRATR    98
  240 CONTINUE                                                          NMRATR    99
      II=0                                                              NMRATR   100
      IF (W2(1,1)) 181,8,181                                            NMRATR   101
    8 DO 180 II=1,LL                                                    NMRATR   102
      DO 10 K=2,MM                                                      NMRATR   103
      IF (W2(K,1).NE.0.0) GO TO 22                                      NMRATR   104
   10 CONTINUE                                                          NMRATR   105
      GO TO 200                                                         NMRATR   106
   22 M=K                                                               NMRATR   107
      Z(II)=W2(M,1)                                                     NMRATR   108
      DO 20 K=1,N                                                       NMRATR   109
      DO 15 L=1,N                                                       NMRATR   110
      W3(K,L)=-(W2(K+1,1)/Z(II))*(W2(1,L+1)+W2(M,L+1))                  NMRATR   111
   15 CONTINUE                                                          NMRATR   112
   20 CONTINUE                                                          NMRATR   113
      DO 30 K=1,N                                                       NMRATR   114
      DO 25 L=1,N                                                       NMRATR   115
```

```
      W3(K,L)=W3(K,L)+W2(K+1,L+1)                                       NMRATR   116
   20 CONTINUE                                                          NMRATR   117
   30 CONTINUE                                                          NMRATR   118
      IF (M-2) 32,42,32                                                 NMRATR   119
   32 DO 40 K=1,N                                                       NMRATR   120
      STORE=W3(1,K)                                                     NMRATR   121
      W3(1,K)=W3(M-1,K)                                                 NMRATR   122
      W3(M-1,K)=STORE                                                   NMRATR   123
   40 CONTINUE                                                          NMRATR   124
   42 DO 50 K=1,N                                                       NMRATR   125
      G1=0.0                                                            NMRATR   126
      DO 45 L=1,N                                                       NMRATR   127
      G1=W3(K,L)*(W2(L+1,1)/Z(II))+G1                                   NMRATR   128
   45 CONTINUE                                                          NMRATR   129
      W3(K,M-1)=G1                                                      NMRATR   130
   50 CONTINUE                                                          NMRATR   131
      IF (M-2) 52,62,52                                                 NMRATR   132
   52 DO 60 K=1,N                                                       NMRATR   133
      STORE=W3(K,1)                                                     NMRATR   134
      W3(K,1)=W3(K,M-1)                                                 NMRATR   135
      W3(K,M-1)=STORE                                                   NMRATR   136
   60 CONTINUE                                                          NMRATR   137
   62 G1=0.0                                                            NMRATR   138
      DO 71 KKK=1,N                                                     NMRATR   139
      G1=-W2(1,KKK+1)*W2(KKK+1,1)+G1                                    NMRATR   140
   71 CONTINUE                                                          NMRATR   141
      DO 70 K=1,N                                                       NMRATR   142
      DO 70 L=1,N                                                       NMRATR   143
      W2(K,L)=W3(K,L)                                                   NMRATR   144
   70 CONTINUE                                                          NMRATR   145
      W2(1,1)=G1/Z(II)                                                  NMRATR   146
      N=N-1                                                             NMRATR   147
      MM=N+1                                                            NMRATR   148
      IF (IPT.LT.2) GO TO 180                                           NMRATR   149
      CALL SPIT1 (W2,MM,MM,                                             NMRATR   150
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        NMRATR   151
  180 CONTINUE                                                          NMRATR   152
  181 CONTINUE                                                          NMRATR   153
C   FOLLOWING STATEMENT ADDED  SEPT 1972 DUE TO CDC IBM DO LOOP         NMRATR   154
C    INDEX DIFFERENCE                                                   NMRATR   155
      IF(II.NE.0) II=II-1                                               NMRATR   156
C    IN THE FORM---                                                     NMRATR   157
C      DO 180 II=1,LL    WHERE LL=1                                     NMRATR   158
C    II TERMINAL VALUE IS 2 ON CDC                                      NMRATR   159
C    II TERMINAL VALUE IS 1 ON IBM                                      NMRATR   160
      IF (N.EQ.0) GO TO 197                                             NMRATR   161
      DO 190 K=1,N                                                      NMRATR   162
      DO 185 L=1,N                                                      NMRATR   163
      W3(K,L)=-W2(K+1,L+1)+ ((W2(K+1,1)*W2(1,L+1))/W2(1,1))             NMRATR   164
  185 CONTINUE                                                          NMRATR   165
  190 CONTINUE                                                          NMRATR   166
  197 DE= W2(1,1)                                                       NMRATR   167
      IF (II) 193,196,193                                               NMRATR   168
  193 DO 195 KJ=1,II                                                    NMRATR   169
      DE=DE*Z(KJ)                                                       NMRATR   170
  195 CONTINUE                                                          NMRATR   171
  196 CONTINUE                                                          NMRATR   172
```

```
            IF (N.NE.0) GO TO 199                                      NMRATR     173
            WRITE (3,210) OUTPT(I), INPT(JJ),DE                        NMRATR     174
        210 FORMAT (//10X,*THE*,A10,*/*,A10,*NUMERATOR GAIN IS * ,E12.4,    NMRATR     175
           1*THERE ARE NO ZEROES*/)                                    NMRATR     176
            GO TO 198                                                  NMRATR     177
        199 DO 400 K=1,N                                               NMRATR     178
            DO 400 J=1,N                                               NMRATR     179
            W1(K,J)=W3(K,J)                                            NMRATR     180
        400 CONTINUE                                                   NMRATR     181
            IF (IPT.LT.2) GO TO 243                                    NMRATR     182
            CALL SPIT1(W1,N,N,                                         NMRATR     183
           1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                 NMRATR     184
        243 CONTINUE                                                   NMRATR     185
            WRITE (3,209) OUTPT(I),INPT(JJ),DE                         NMRATR     186
        209 FORMAT (//10X,,*THE *,A10,*/*A10,* NUMERATOR GAIN IS*,E12.4//* THE NMRATR     187
           1 ZEROES OF THE TRANSFER FUNCTION ARE*//20X,*REAL PART*,15X,*IMAGIN NMRATR     188
           2ARY PART*/)                                                NMRATR     189
            IF (MN.NE.0.AND. SYSTEM .NE. 3)  WRITE (3,1000)            NMRATR     190
       1000 FORMAT (/ * FOR MODIFIED Z-TRANSFORMS, ADD Z=0. POLE*/)    NMRATR     191
            CALL EIGEN (N,W1,W2,W3,ROTR,ROTI,Z,V,                      NMRATR     192
           1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                 NMRATR     193
            CALL CPMT (Z,ROTR,ROTI,N,                                  NMRATR     194
           1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                 NMRATR     195
            WRITE (3,82)                                               NMRATR     196
         82 FORMAT (//10X,* THE COEFFICIENTS OF THE NUMERATOR  POLYNOMIAL ORDE NMRATR     197
           1RED FROM THE LOWEST POWER OF S*//)                         NMRATR     198
            NX1=N+1                                                    NMRATR     199
            WRITE (3,83) (Z(IA),IA=1,NX1)                              NMRATR     200
         83 FORMAT (E20.8)                                             NMRATR     201
        198 CONTINUE                                                   NMRATR     202
            IF (DIGITL.EQ.0) GO TO 110                                 NMRATR     203
            IF (MN .EQ. 0) GO TO 113                                   NMRATR     204
            NSAV=NX                                                    NMRATR     205
            NN=NN+1                                                    NMRATR     206
            NX=NX+1                                                    NMRATR     207
            IF (I .GT. 1 .OR. JJ .GT. 1) GO TO 113                     NMRATR     208
            ROOTR(NX)= 0.0                                             NMRATR     209
            ROOTI(NX)= 0.0                                             NMRATR     210
        113 CONTINUE                                                   NMRATR     211
            IF (DIGITL .EQ. 0 .OR. SYSTEM .EQ. 3 .OR. FRPS .EQ. -1) GO TO 110  NMRATR     212
            IF (I .GT. 1 .OR. JJ .GT. 1) GO TO 110                     NMRATR     213
            CALL ZTOW(N,NN,DE,1,1,ROOTR,ROOTI,ROTR,ROTI,L1,DEE,        NMRATR     214
           1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                 NMRATR     215
            WRITE (3,111)                                              NMRATR     216
        111 FORMAT (/10X,*  THE W-PLANE POLES ARE *//)                 NMRATR     217
            WRITE (3,112) (ROOTR(K),ROOTI(K),K=1,NN)                   NMRATR     218
        112 FORMAT (/(2E30.8))                                         NMRATR     219
        110 CONTINUE                                                   NMRATR     220
            IF (DIGITL.EQ.0.OR.SYSTEM.EQ.3.OR.FRPS.EQ.-1) GO TO 120    NMRATR     221
            CALL ZTOW (N,NN,DE,2,2,ROOTR,ROOTI,ROTR,ROTI,L1,DEE,       NMRATR     222
           1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                 NMRATR     223
            WRITE (3,121) DE                                           NMRATR     224
        121 FORMAT (//10X,*  THE W-PLANE TRANSFER FUNCTION GAIN IS   *,E12.4/* NMRATR     225
           1          THE ZEROES ARE*//)                               NMRATR     226
            WRITE (3,112) (ROTR(K),ROTI(K),K=1,N)                      NMRATR     227
        120 CONTINUE                                                   NMRATR     228
            IF (FRPS.EQ.-1) GO TO 301                                  NMRATR     229
```

```
              IFR=FRPS+1                                                NMRATR     230
              GO TO (311,310,320) IFR                                   NMRATR     231
          310 IF (MODEL.EQ.0) GO TO 301                                 NMRATR     232
              IF((MOD(I,2)).EQ.1) GO TO 302                             NMRATR     233
              IF (IMOD.EQ.2) GO TO 302                                  NMRATR     234
              IMOD=2                                                    NMRATR     235
              GO TO 301                                                 NMRATR     236
          302 IMOD=1                                                    NMRATR     237
          301 CALL FRQRSP (N,NN,DE,IMOD,ROOTR,ROOTI,ROTR,ROTI,SAV1,SAV2, NMRATR     238
             1I,JJ,                                                     NMRATR     239
             1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                NMRATR     240
              GO TO 311                                                 NMRATR     241
          320 CALL PSD (N,NN,DE,ROOTR,ROOTI,ROTR,ROTI,SAV1,SAV2,I,JJ,   NMRATR     242
             1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                NMRATR     243
              GO TO 311                                                 NMRATR     244
          200 WRITE (3,211) JJ                                          NMRATR     245
          211 FORMAT (//10X,*COLUMN *,I2,*  OF THE B MATRIX IS NULL*/)  NMRATR     246
          311 IF (MN.EQ.0) GO TO 300                                    NMRATR     247
              NX=NSAV                                                   NMRATR     248
              NN=NSAV                                                   NMRATR     249
          300 CONTINUE                                                  NMRATR     250
              END                                                       NMRATR     251
```

```
      SUBROUTINE PSD (NNUM,NN,GAIN,ROOTR,ROOTI,ROTR,ROTI,SAV1,SAV2,       PSD        2
     1INY,INU,                                                            PSD        3
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                          PSD        4
C                                                                         PSD        5
C       THIS SUBROUTINE COMPUTES THE POWER SPECTRUM CORRESPONDING        PSD        6
C       TO THE TRANSFER FUNCTION WHOSE POLES ARE (ROOTR,ROOTI) AND       PSD        7
C       WHOSE ZEROES ARE (ROTR,ROTI). IT IS ASSUMED THAT THE INPUT       PSD        8
C       TO THE TRANSFER FUNCTION IS A WHITE NOISE PROCESS OF UNITY       PSD        9
C       VARIANCE AND HENCE THE PSD IS GIVEN BY THE SQUARE OF THE         PSD       10
C       MODULUS OF THE TRANSFER FUNCTION. THUS ANY CORRELATION           PSD       11
C       DESIRED IN THE RANDOM PROCESS EXCITING THE SYSTEM SHOULD         PSD       12
C       BE INCLUDED AS A SHAPING FILTER IN THE *A* MATRIX WHICH          PSD       13
C       IS EXCITED BY WHITE NOISE. NOTE THAT ONLY EXPONENTIALLY          PSD       14
C       CORRELATED PROCESSES MAY BE SO STRUCTURED.                       PSD       15
C                                                                         PSD       16
      DIMENSION ROOTR(MX),ROOTI(MX),ROTR(MX),ROTI(MX),SAV1(MS),SAV2(MS)   PSD       17
      COMPLEX RN1,RN2,RD1,RD2                                             PSD       18
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,       PSD       19
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,           PSD       20
     2IGC,FORM,IPT,READ3,MIXED,MULTPT,SCAPLT,ZOH,KOUNT                    PSD       21
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP   PSD       22
      COMMON/ACOND/ DELT,FINALT,IFREQ,FFREQ,DELFRQ,GAIN1,GAIN2,M          PSD       23
      INTEGER DIGITL, SCAPLT, ZOH                                         PSD       24
      REAL IFREQ,M                                                        PSD       25
      REAL PST                                                            PSD       26
      COMMON/LABEL/INPT,OUTPT,TITLE                                       PSD       27
      REAL   INPT(10), OUTPT(20),TITLE(8)                                 PSD       28
      COMMON/SUBWRIT/ ISUBNAM                                             PSD       29
      DATA PST/10HSPEC      /                                             PSD       30
      IF(ISUBNAM.GE.2) WRITE(3,990)                                       PSD       31
  990 FORMAT(1X,*PSD*)                                                    PSD       32
      IF(FORM.GT.0) WRITE(7)PST,TITLE,SYSTEM,MODEL,DIGITL,SCAPLT          PSD       33
      VAR=0.0                                                             PSD       34
      X=0.0                                                               PSD       35
      J=1                                                                 PSD       36
      IF(FORM.GT.0) WRITE(7)OUTPT(INY),INPT(INU)                          PSD       37
      WRITE (3,60) OUTP T(INY),INP T(INU)                                 PSD       38
   60 FORMAT (/10X,A10,*/*A10,* POWER SPECTRAL DENSITY*/10X,*FREQUENCY,R  PSD       39
     1AD/SEC*4X,*PSD MAGNITUDE*/)                                         PSD       40
      FREQ=IFREQ                                                          PSD       41
      IF (IFREQ.NE.0.) GO TO 70                                           PSD       42
      FREQ=.1                                                             PSD       43
      FFREQ=150.                                                          PSD       44
   70 CONTINUE                                                            PSD       45
  100 CONTINUE                                                            PSD       46
      IF (IFREQ.NE.0.) GO TO 71                                           PSD       47
      FREQ=1.15*FREQ                                                      PSD       48
      GO TO 72                                                            PSD       49
   71 FREQ=DELFRQ*FREQ                                                    PSD       50
   72 CONTINUE                                                            PSD       51
      RN1=CMPLX (1.0,0.0)                                                 PSD       52
      RD1=CMPLX (1.0,0.0)                                                 PSD       53
      DO 20 I=1,NN                                                        PSD       54
      IF (I.GT.NNUM) GO TO 5                                              PSD       55
      RN2=CMPLX (-ROTR(I),FREQ    -ROTI(I))                               PSD       56
      RN1=RN1*RN2                                                         PSD       57
      RN2=CMPLX (-ROTR(I),-FREQ    -ROTI(I))                              PSD       58
```

```
          RN1=RN1*RN2                                                     PSD       59
          RD2=CMPLX (-ROOTR(I),FREQ     -ROOTI(I))                        PSD       60
          RD1=RD1*RD2                                                     PSD       61
          RD2=CMPLX (-ROOTR(I),-FREQ     -ROOTI(I))                       PSD       62
          RD1=RD1*RD2                                                     PSD       63
       20 CONTINUE                                                        PSD       64
          SAV1(J)=GAIN**2*RN1/RD1                                         PSD       65
          VAR=VAR+SAV1(J)*(FREQ     -X)                                   PSD       66
          IF(FORM.EQ.2) GO TO 6                                           PSD       67
          WRITE (3,50) FREQ     ,SAV1(J)                                  PSD       68
          IF(FORM.EQ.0) GO TO 40                                          PSD       69
        6 WRITE(7)FREQ     ,SAV1(J)                                       PSD       70
       40 CONTINUE                                                        PSD       71
       50 FORMAT (14X,F8.4.12X,E12.4)                                     PSD       72
          X=FREQ                                                          PSD       73
          J=J+1                                                           PSD       74
          IF (FREQ.LE.FFREQ) GO TO 100                                    PSD       75
       10 CONTINUE                                                        PSD       76
          WRITE (3,30) VAR                                                PSD       77
       30 FORMAT (10X,*THE VARIANCE IS*E12.4)                             PSD       78
          JD=99                                                           PSD       79
          WRITE(7) JD,SAV1(I)                                             PSD       80
          RETURN                                                          PSD       81
          END                                                             PSD       82
```

```
      SUBROUTINE QREIG (M,ROOTR,ROOTI,IPRNT,A,                          QREIG      2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        QREIG      3
      DIMENSION A(MX,MY),ROOTR(MX),ROOTI(MX)                            QREIG      4
      COMMON/SUBWRIT/ ISUBNAM                                           QREIG      5
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     QREIG      6
  990 FORMAT(1X,*QREIG*)                                                QREIG      7
      N = M                                                             QREIG      8
      IF(IPRNT)  80,81,80                                               QREIG      9
   80 WRITE (3,104)                                                     QREIG     10
   81 ZERO = 0.0                                                        QREIG     11
      JJ=1                                                              QREIG     12
  177 XNN=0.0                                                           QREIG     13
      XN2=0.0                                                           QREIG     14
      AA = 0.0                                                          QREIG     15
      B = 0.0                                                           QREIG     16
      C = 0.0                                                           QREIG     17
      DD = 0.0                                                          QREIG     18
      R=0.0                                                             QREIG     19
      SIG=0.0                                                           QREIG     20
      ITER = 0                                                          QREIG     21
   17 IF(N-2)     13,14,12                                              QREIG     22
   13 IF(IPRNT)  82,83,82                                               QREIG     23
   82 WRITE (3,105)A(1,1)                                               QREIG     24
   83 ROOTR(1) = A(1,1)                                                 QREIG     25
      ROOTI(1) = 0.0                                                    QREIG     26
    1 CONTINUE                                                          QREIG     27
      RETURN                                                            QREIG     28
   14 JJ=-1                                                             QREIG     29
   12 X =  (A(N-1,N-1) - A(N,N))**2                                     QREIG     30
      S = 4.0*A(N,N-1)*A(N-1,N)                                         QREIG     31
      ITER = ITER + 1                                                   QREIG     32
C     IF (X.EQ.0.0.OR. ABS(S/X) .GT. 1.0E-8) GO TO 15                   QREIG     33
      IF(X.EQ.0.0)GO TO 15                                              QREIG     34
      IF(ABS(S/X).GT.1.0E-8) GO TO 15                                   QREIG     35
   16 IF ( ABS(A(N-1,N-1))- ABS(A(N,N))) 32,32,31                       QREIG     36
   31 E = A(N-1,N-1)                                                    QREIG     37
      G = A(N,N)                                                        QREIG     38
      GO TO 33                                                          QREIG     39
   32 G = A(N-1,N-1)                                                    QREIG     40
      E = A(N,N)                                                        QREIG     41
   33 F = 0.                                                            QREIG     42
      H = 0.                                                            QREIG     43
      GO TO 24                                                          QREIG     44
   15 S = X + S                                                         QREIG     45
      X = A(N-1,N-1) + A(N,N)                                           QREIG     46
      IF(S)     18,19,19                                                QREIG     47
   19 SQ= SQRT(S)                                                       QREIG     48
      F=0.0                                                             QREIG     49
      H=0.0                                                             QREIG     50
      IF (X) 21,21,22                                                   QREIG     51
   21 E=(X-SQ)/2.0                                                      QREIG     52
      G=(X+SQ)/2.0                                                      QREIG     53
      GO TO 24                                                          QREIG     54
   22 G=(X-SQ)/2.0                                                      QREIG     55
      E=(X+SQ)/2.0                                                      QREIG     56
      GO TO 24                                                          QREIG     57
   18 F= SQRT (-S)/2.0                                                  QREIG     58
```

```
      E=X/2.0                                                           QREIG     59
      G=E                                                               QREIG     60
      H=-F                                                              QREIG     61
   24 IF(JJ)  28,70,70                                                  QREIG     62
   70 D=1.0E-10*( ABS(G)+F)                                             QREIG     63
      IF( ABS(A(N-1,N-2)).GT. D) GO TO 26                               QREIG     64
   28 IF(IPRNT)  84,85,84                                               QREIG     65
   84 WRITE (3,105)E,F,ITER                                             QREIG     66
      WRITE (3,105)G,H                                                  QREIG     67
   85 ROOTR(N) = E                                                      QREIG     68
      ROOTI(N) = F                                                      QREIG     69
      ROOTR(N-1) = G                                                    QREIG     70
      ROOTI(N-1) = H                                                    QREIG     71
      N=N-2                                                             QREIG     72
      IF(JJ)  1,177,177                                                 QREIG     73
   26 IF( ABS(A(N,N-1)) .GT. 1.0E-10* ABS(A(N,N))) GO TO 50             QREIG     74
   29 IF(IPRNT)  86,87,86                                               QREIG     75
   86 WRITE (3,105)A(N,N),ZERO,ITER                                     QREIG     76
   87 ROOTR(N) = A(N,N)                                                 QREIG     77
      ROOTI(N) = 0.0                                                    QREIG     78
      N=N-1                                                             QREIG     79
      GO TO 177                                                         QREIG     80
   50 IF( ABS( ABS(XNN/A(N,N-1))-1.0)-1.0E-6) 63,63,62                  QREIG     81
   62 IF( ABS( ABS(XN2/A(N-1,N-2))-1.0)-1.0E-6) 63,63,700               QREIG     82
   63 VO= ABS(A(N,N-1))- ABS(A(N-1,N-2))                                QREIG     83
      IF (ITER-15) 53,164,64                                            QREIG     84
  164 IF(VO) 165,165,166                                                QREIG     85
  165 P = A(N-1,N-2)**2                                                 QREIG     86
      SIG = 2.0*A(N-1,N-2)                                              QREIG     87
      GO TO 60                                                          QREIG     88
  166 P = A(N,N-1)**2                                                   QREIG     89
      SIG = 2.0*A(N,N-1)                                                QREIG     90
      GO TO 60                                                          QREIG     91
   64 IF(VO)   67,67,66                                                 QREIG     92
   66 IF(IPRNT)  88,85,88                                               QREIG     93
   88 WRITE (3,107)A(N-1,N-2)                                           QREIG     94
      GO TO 84                                                          QREIG     95
   67 IF(IPRNT)  89,87,89                                               QREIG     96
   89 WRITE (3,107)A(N,N-1)                                             QREIG     97
      GO TO 86                                                          QREIG     98
  700 IF(ITER .GT. 50) GO TO 63                                         QREIG     99
      IF(ITER .GT. 5 ) GO TO 53                                         QREIG    100
  701 ET=E*E+F*F                                                        QREIG    101
      IF (ET) 702,601,702                                               QREIG    102
  702 GT=G*G+H*H                                                        QREIG    103
      IF (GT) 703,601,703                                               QREIG    104
  703 CONTINUE                                                          QREIG    105
      Z1=          ((E-AA)**2+(F-B)**2)/(E*E+F*F)                       QREIG    106
      Z2=        ((G-C)**2+(H-DD)**2)/(G*G+H*H)                         QREIG    107
      IF(Z1-0.25)  51,51,52                                             QREIG    108
   51 IF(Z2-0.25)    53,53,54                                           QREIG    109
   53 P=E*G-F*H                                                         QREIG    110
      SIG=E+G                                                           QREIG    111
      GO TO 60                                                          QREIG    112
   54 P=E*E                                                             QREIG    113
      SIG=E+E                                                           QREIG    114
      GO TO 60                                                          QREIG    115
```

```
       57 IF(ZZ-0.25) 55,57,601                              QREIG    116
       55 R=0.0                                              QREIG    117
          SIG=5.0                                            QREIG    118
          GO TO 60                                           QREIG    119
      601 R = 0.0                                            QREIG    120
          SIG = 0.0                                          QREIG    121
       60 XNN=A(N,N-1)                                       QREIG    122
          XN2=A(N-1,N-2)                                     QREIG    123
          CALL QRT (N,R,SIG,D,A,                             QREIG    124
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)         QREIG    125
          AA=                                                QREIG    126
          B=F                                                QREIG    127
          C=G                                                QREIG    128
          DD=H                                               QREIG    129
          GO TO 12                                           QREIG    130
      104 FORMAT(////1X, 9HREAL PART  6X  14HIMAGINARY PART, 25X QREIG 131
         1  13HTAKEN AS ZERO 5X 6HITER //)                   QREIG    132
      105 FORMAT(1X,E15.8,3X,E15.8, 42X I3)                  QREIG    133
      107 FORMAT (74X, E15.8)                                QREIG    134
           END                                               QREIG    135
```

```
      SUBROUTINE QRT (N,R,SIG,D,A,                                      QRT        2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        QRT        3
      DIMENSION A(MX,MX),PSI(2),G(3)                                    QRT        4
      COMMON/SUBWRIT/ ISUBNAM                                           QRT        5
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     QRT        6
  990 FORMAT(1X,*QRT*)                                                  QRT        7
      N1 = N - 1                                                        QRT        8
      IA = N - 2                                                        QRT        9
      IP = IA                                                           QRT       10
      IF(N-3) 101,10,60                                                 QRT       11
   60 DO 12 J = 3,N1                                                    QRT       12
      J1 = N - J                                                        QRT       13
      IF( ABS(A(J1+1,J1))-D) 10,10,11                                   QRT       14
   11 DEN = A(J1+1,J1+1)*(A(J1+1,J1+1)-SIG)+A(J1+1,J1+2)*A(J1+2,J1+1)+R QRT       15
      IF(DEN) 61,12,61                                                  QRT       16
   61 IF( ABS(A(J1+1,J1)*A(J1+2,J1+1)*( ABS(A(J1+1,J1+1)+A(J1+2,J1+2)  QRT       17
     1-SIG)+ ABS(A(J1+3,J1+2)))/DEN)-D) 10,10,12                        QRT       18
   12 IP= J1                                                            QRT       19
   10 DO  14  J=1,IP                                                    QRT       20
      J1=IP-J+1                                                         QRT       21
      IF ( ABS(A(J1+1,J1))-D) 13,13,14                                  QRT       22
   14 IQ= J1                                                            QRT       23
   13 DO  100  I=IP,N1                                                  QRT       24
      IF(I-IP)    16,15,16                                              QRT       25
   15 G(1)=A(IP,IP)*(A(IP,IP)-SIG)+A(IP,IP+1)*A(IP+1,IP)+R              QRT       26
      G(2)=A(IP+1,IP)*(A(IP,IP)+A(IP+1,IP+1)-SIG)                       QRT       27
      G(3)=A(IP+1,IP)*A(IP+2,IP+1)                                      QRT       28
      A(IP+2,IP)=0.0                                                    QRT       29
      GO TO 19                                                          QRT       30
   16 G(1)=A(I,I-1)                                                     QRT       31
      G(2)=A(I+1,I-1)                                                   QRT       32
      IF(I-IA)    17,17,18                                              QRT       33
   17 G(3)=A(I+2,I-1)                                                   QRT       34
      GO TO 19                                                          QRT       35
   18 G(3)=0.0                                                          QRT       36
   19 XK=  SIGN( SQRT(G(1)**2+G(2)**2+G(3)**2), G(1))                   QRT       37
   22 IF(XK)    23,24,23                                                QRT       38
   23 AL=G(1)/XK+1.0                                                    QRT       39
      PSI(1)=G(2)/(G(1)+XK)                                             QRT       40
      PSI(2)=G(3)/(G(1)+XK)                                             QRT       41
      GO TO 25                                                          QRT       42
   24 AL=2.0                                                            QRT       43
      PSI(1)=0.0                                                        QRT       44
      PSI(2)=0.0                                                        QRT       45
   25 IF(I-IQ)    26,27,26                                              QRT       46
   26 IF(I-IP)     29,28,29                                             QRT       47
   28 A(I,I-1)=-A(I,I-1)                                                QRT       48
      GO TO 27                                                          QRT       49
   29 A(I,I-1)=-XK                                                      QRT       50
   27 DO 30  J=I,N                                                      QRT       51
      IF(I-IA)   31,31,32                                               QRT       52
   31 C=PSI(2)*A(I+2,J)                                                 QRT       53
      GO TO 33                                                          QRT       54
   32 C=0.0                                                             QRT       55
   33 E=AL*(A(I,J)+PSI(1)*A(I+1,J)+C)                                   QRT       56
      A(I,J)=A(I,J)-E                                                   QRT       57
      A(I+1,J)=A(I+1,J)-PSI(1)*E                                        QRT       58
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      IF(I-IA)    34,34,30                                          QRT      59
   34 A(I+2,J)=A(I+2,J)-PSI(2)*E                                    QRT      60
   30 CONTINUE                                                      QRT      61
      IF(I-IA)    35,35,36                                          QRT      62
   35 L=I+2                                                         QRT      63
      GO TO  37                                                     QRT      64
   36 L=N                                                           QRT      65
   37 DO  40  J=IO,L                                                QRT      66
      IF(I-IA)   38,38,39                                           QRT      67
   38 C=PSI(2)*A(J,I+2)                                             QRT      68
      GO TO  41                                                     QRT      69
   39 C=0.0                                                         QRT      70
   41 E=AL*(A(J,I)+PSI(1)*A(J,I+1)+C)                               QRT      71
      A(J,I)=A(J,I)-E                                               QRT      72
      A(J,I+1)=A(J,I+1)-PSI(1)*E                                    QRT      73
      IF(I-IA)   42,42,40                                           QRT      74
   42 A(J,I+2)=A(J,I+2)-PSI(2)*E                                    QRT      75
   40 CONTINUE                                                      QRT      76
      IF(I-N+3)    43,43,100                                        QRT      77
   43 E=AL*PSI(2)*A(I+3,I+2)                                        QRT      78
      A(I+3,I)=-E                                                   QRT      79
      A(I+3,I+1)=-PSI(1)*E                                          QRT      80
      A(I+3,I+2)=A(I+3,I+2)-PSI(2)*E                                QRT      81
  100 CONTINUE                                                      QRT      82
  101 RETURN                                                        QRT      83
      END                                                           QRT      84
```

```
      SUBROUTINE RDISC   (A,B,C,H,G,F,K1,K2,K3,K4,D,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
C
C       THIS SUBROUTINE READS ALL INPUT MATRICES FROM DISCS ACCORDING
C       TO THE  PARAMETERS, SYSTEM AND OUTPUT, USING SUBROUTINE RDISC1
C
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,
     1CONTUR,NUMERR,ERRS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,
     2IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, ERRS,TRESP
      INTEGER        DIGITL,SCAPLT, ZOH
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),
     1K1(MU,MX),K2(MU,MX),K3(MU,MX),K4(MU,MX),D(MU,MU)
      REAL K1,K2,K3,K4
      COMMON/SUBWRIT/ ISUBNAM
      IF(ISUBNAM.GE.2) WRITE(3,990)
  990 FORMAT (1X,*RDISC*)
      MAT1=MX
      MAT2=MX
      CALL RDISC1 (A,NX,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MU
      CALL RDISC1 (B,NX,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MX
      CALL RDISC1 (C,NX,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MU
      CALL RDISC1 (K1,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL RDISC1 (K2,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL RDISC1 (K3,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL RDISC1 (K4,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MY
      CALL RDISC1 (H,NY,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL RDISC1 (G,NY,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MU
      CALL RDISC1 (F,NY,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MU
      CALL RDISC1 (D,NU,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      REWIND 8
      RETURN
      END
```

```
SUBROUTINE RDISC1       73/74    OPT=1                      FTN 4.2+75060        01/09/76  14.13.53.

                    SUBROUTINE RDISC1 (A,N,M,                                          RDISC1        2
                   1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         RDISC1        3
                    DIMENSION A(MAT1,MAT2)                                             RDISC1        4
                    COMMON/SUBWRIT/ ISUBNAM                                            RDISC1        5
 F                  IF(ISUBNAM.GE.2) WRITE(3,940)                                      RDISC1        6
                940 FORMAT(1X,*RDISC1*)                                                RDISC1        7
                    DO 10 I=1,N                                                        RDISC1        8
                    READ (8)  (A(I,J),J=1,M)                                           RDISC1        9
                    IF(EOF(8).NE.0) STOP1                                              RDISC1       10
10               10 CONTINUE                                                           RDISC1       11
                    RETURN                                                             RDISC1       12
                    END                                                                RDISC1       13
```

```
      SUBROUTINE REDUCE (N,J,MM,M,A,B,C,ITEST,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
C
C        THIS SUBROUTINE DETERMINES THE IRREDUCIBLE SUBMATRICES OF
C        THE MATRIX A WITH DIMENSION N.
C
C         J- NUMBER OF IRREDUCIBLE SUBMATRICES (1-5)
C     MM(I)- DIMENSION OF THE ITH SUBMATRIX
C        M - MATRIX WHOSE ITH ROW CONTAINS THE ORIGINAL ROW AND COLUMN
C              NUMBERS OF THE ITH SUBMATRIX
C
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,
     2IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP
      INTEGER DIGITL,SCAPLT,ZOH
      DIMENSION A(MX,MX),B(MX,MX),C(MX,MX)
      INTEGER SCAN
      DIMENSION M(10,20),MM(20),SCAN(20)
      COMMON/SUBWRIT/ ISUBNAM
      IF(ISUBNAM.GE.2) WRITE(3,990)
  990 FORMAT(1X,*REDUCE*)
      DO 60 I=1,20
      DO 61 J=1,10
      M(J,I)=0
   61 CONTINUE
      MM(I)=0
   60 CONTINUE
      DO 10 I=1,N
      DO 11 J=1,N
      B(I,J)=0.0
      C(I,J)=0.0
      IF (A(I,J).NE.0.0) B(I,J)=1.0
   11 CONTINUE
      B(I,I)=1.0
   10 CONTINUE
      KRUNT=1
   22 CONTINUE
      DO 20 I=1,N
      DO 20 J=1,N
      DO 21 K=1,N
      IF (B(I,K).EQ.0.0.OR.B(K,J).EQ.0.0) GO TO 21
      C(I,J)=1.0
      GO TO 20
   21 CONTINUE
   20 CONTINUE
      KRUNT=2*KRUNT
      IF (KRUNT.GE.N-1) GO TO 23
      DO 24 I=1,N
      DO 24 J=1,N
      B(I,J)=C(I,J)
   24 CONTINUE
      GO TO 22
   23 CONTINUE
      DO 30 I=1,N
      SCAN(I)=I
   30 CONTINUE
```

```
      K=N                                                               REDUCE      59
      DO 35  J=1,10                                                     REDUCE      60
      MM(J)=0                                                           REDUCE      61
      NK=1                                                              REDUCE      62
      JJ=1                                                              REDUCE      63
      DO 36 I=1,K                                                       REDUCE      64
      IF (C(SCAN(1),SCAN(I)).EQ.1.0.AND.C(SCAN(I),SCAN(1)).EQ.1.0)      REDUCE      65
     1 GO TO 37                                                         REDUCE      66
      GO TO 36                                                          REDUCE      67
   37 M(J,JJ)=SCAN(I)                                                   REDUCE      68
      MM(J)=MM(J)+1                                                     REDUCE      69
      NK=NK+1                                                           REDUCE      70
      JJ=JJ+1                                                           REDUCE      71
   36 CONTINUE                                                          REDUCE      72
      K=K-NK                                                            REDUCE      73
      IF (K.LE.0) GO TO 38                                              REDUCE      74
      LL=1                                                              REDUCE      75
      KRUNT=0                                                           REDUCE      76
      KONT=0                                                            REDUCE      77
      JX=K+NK                                                           REDUCE      78
      DO 39 L=1,JX                                                      REDUCE      79
      IF (SCAN(L).NE.M(J,LL)) GO TO 40                                  REDUCE      80
      KRUNT=KRUNT+1                                                     REDUCE      81
      LL=LL+1                                                           REDUCE      82
      GO TO 39                                                          REDUCE      83
   40 KRUNT=KRUNT+1                                                     REDUCE      84
      KONT=KONT+1                                                       REDUCE      85
      SCAN(KONT)=SCAN(KRUNT)                                            REDUCE      86
   39 CONTINUE                                                          REDUCE      87
   35 CONTINUE                                                          REDUCE      88
   38 CONTINUE                                                          REDUCE      89
      IF (K.GT.0)WRITE (3,100)                                          REDUCE      90
  100 FORMAT (//,* MORE THAN 10 REDUCIBLE SUBMATRICES*//)               REDUCE      91
      IF (K.GT.0) ITEST=1                                               REDUCE      92
      IF (IPT.LT.2) GO TO 200                                           REDUCE      93
      WRITE (3,201)                                                     REDUCE      94
  201 FORMAT (/* MATRIX M IN REDUCE*/)                                  REDUCE      95
C     IN STATEMENT DO 35 J=1,10 IF THE LOOP GOES TO COMPLETION          REDUCE      96
C     J WILL = 11 ON CDC                                                REDUCE      97
C     J WILL = 10 ON IBM                                                REDUCE      98
      IF(J.GE.11) J=J-1                                                 REDUCE      99
      DO 202 I=1,J                                                      REDUCE     100
      MMM=MM(I)                                                         REDUCE     101
      WRITE (3,203) (M(I,L),L=1,MMM)                                    REDUCE     102
  202 CONTINUE                                                          REDUCE     103
  203 FORMAT (20I5)                                                     REDUCE     104
  200 CONTINUE                                                          REDUCE     105
      RETURN                                                            REDUCE     106
      END                                                               REDUCE     107
```

```
      SUBROUTINE ROOT (A,B,C,H,G,F,K1,K2,K3,K4,D,W1,W2,W3,ROOTR,ROOTI,
     1U,V,
     2MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
C
C       THIS SUBROUTINE COMPUTES ROOT LOCII AS A FUNCTION OF TWO
C       FEEDBACK GAINS DEFINED BY K1,K2 AND K3,K4. THE OPEN LOOP
C       SYSTEM IS SCALED BY A SIMILARITY TRANSFORMATION. AT THE
C       CONCLUSION OF THE ROOT LOCII COMPUTATIONS THE NUMERATOR
C       ZEROES AND GAINS CORRESPONDING TO THE OPEN LOOP SYSTEM
C       ARE CALCULATED.
C
C        N1- NUMBER OF ROOT LOCUS POINTS FOR THE K1,K2 GAIN
C        N2- NUMBER OF ROOT LOCUS POINTS FOR THE K3,K4 GAIN
C
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,
     2IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KCUNT
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP
      INTEGER DIGITL,SCAPLT,ZOH
      REAL IFREQ,K1,K2,K3,K4,MN
      COMMON/ACOND/ DELT,FINALT,IFREQ,FFREQ,DELFRQ,GAIN1,GAIN2,MN
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),
     1K1(MU,MX),K2(MU,MX),K3(MU,MX),K4(MU,MX),D(MU,MU)
      DIMENSION W1(MX,MX),W2(MX,MX),W3(MX,MX)
      DIMENSION ROOTR(MX),ROOTI(MX),U(MX),V(MX)
      DIMENSION M(10,20),MM(20),P(20)
      COMMON/LABEL/INPT,OUTPT,TITLE
      REAL   INPT(10), OUTPT(20),TITLE(8)
      COMMON/SUBWRIT/ ISUBNAM
      DATA BLANK/8H        /
      DATA PST/10HPCLO      /
      IF(ISUBNAM.GE.2) WRITE(3,990)
      IF (ISUBNAM .EQ. 2) WRITE (3,990)
  990 FORMAT(1X,*ROOT*)
      IF (GAIN1.EQ.0.0) GAIN1=1.0
      IF (GAIN2.EQ.0.0) GAIN2=1.0
      WRITE (3,300) GAIN1,GAIN2
  300 FORMAT (/10X,*GAIN1 = *F10.4,10X,*GAIN2 = *F10.4/)
      MAT1=MX
      MAT2=MX
      MAT3=MX
      MAT4=MX
      MAT5=MX
      MAT6=MX
      IF(FORM.GT.0) WRITE(7)PST,TITLE,SYSTEM,MODEL,DIGITL,SCAPLT
      IF(FORM.GT.0) WRITE(7)N2,N1
      KK=1
      N1A=IABS(N1)
      RR=0.0
      TT=0.0
      II=0
      JJ=0
      NK3=NK2+1
      DO 40 I=1,N2
      IF (N1.LT.0) GO TO 141
      II=I-1
      RR=(I-1)*GAIN2
```

```
      GO TO 142
  141 RR=2.*RR
      II=2*II
      IF (I.EQ.1)II=0
      IF (I.EQ.2)II=1
      IF (I.EQ.1) RR=0.0
      IF (I.EQ.2)RR=GAIN2
  142 CONTINUE
      DO 40 J=1,N1A
      IF (N1.LT.0) GO TO 41
      JJ= J-1
      TT=  (J-1) * GAIN1
      GO TO 42
   41 TT=2. * TT
      JJ=2* JJ
      IF (J.EQ.1)  JJ=0
      IF (J.EQ.2)  JJ=1
      IF (J.EQ.1)  TT=0.0
      IF (J.EQ.2)  TT=GAIN1
   42 CONTINUE
      GO TO (101,100),NK3
  100 MAT1=MU
      MAT2=MX
      MAT3=MU
      MAT4=MX
      CALL ADD (TT,K2,RR,K4,W1,NU,NL,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MX
      MAT3=MX
      DO 43 IA=1,NU
      DO 44 JA=1,NU
      W1(IA,JA)=-W1(IA,JA)
   44 CONTINUE
      W1(IA,IA)=1.0+W1(IA,IA)
   43 CONTINUE
      CALL INVR (W1,W2,NU,0,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MU
      MAT3=MU
      CALL ADD (TT,K1,RR,K3,W1,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MX
      MAT3=MX
      CALL MULT (W2,W1,W3,NU,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      GO TO 102
  101 MAT1=MU
      MAT2=MX
      MAT3=MU
      MAT4=MX
      CALL ADD (TT,K1,RR,K3,W3,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
  102 CONTINUE
      IF (J.EQ.2.AND.I.EQ.1) GO TO 200
      IF (J.EQ.1.AND.I.EQ.2) GO TO 201
      GO TO 202
  200 DO 203 IX=1,NU
```

```
      DO 205 JX=1,NX
      IF (NK2.EQ.1) GO TO 209
      H(KK,JX)=W3(IX,JX)/TT
      GO TO 205
  209 H(KK,JX)=W1(IX,JX)/TT
  205 CONTINUE
      IF (NK2.EQ.0) GO TO 203
      DO 207 JX=1,NU
      F(KK,JX)=K2(IX,JX)
  207 CONTINUE
  203 KK=KK+1
      NY=NU
      GO TO 202
  201 DO 204 IX=1,NU
      DO 206 JX=1,NX
      IF (NK2.EQ.1) GO TO 210
      H(KK,JX)=W3(IX,JX)/TT
      GO TO 206
  210 H(KK,JX)=W1(IX,JX)/RR
  206 CONTINUE
      IF (NK2.EQ.0) GO TO 204
      DO 208 JX=1,NU
      F(KK,JX)=K4(IX,JX)
  208 CONTINUE
  204 KK=KK+1
      NY=2*NU
  202 CONTINUE
      MAT1=MX
      MAT2=MU
      MAT3=MX
      MAT4=MY
      CALL MULT (B,W3,W2,NX,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MX
      CALL ADD (1.0,A,1.0,W2,W1,NX,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (N2.NE.0) GO TO 220
      WRITE (3,221) J, TT
  221  FORMAT (//,10X,*NO. *,I5,* ITERATION, 1ST GAIN =*,E12.4/
     120X,* REAL PART*,15X,*IMAGINARY PART*/)
      GO TO 222
  220 WRITE (3,7) J, TT, I, RR
    7 FORMAT (//,10X,*NO. *,I5,* ITERATION, 1ST GAIN =*,E12.4,10X,*NO. *
     1,I5,* ITERATION, 2ND GAIN =*,E12.4/20X,*REAL PART*,15X,
     2*IMAGINARY PART*/)
  222 CONTINUE
      CALL EIGEN (NX,W1,W2,W3,ROOTR,ROOTI,U,V,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
   45 CONTINUE
      DO 74 I=1,NU
      DO 74 J=1,NU
      D(I,J)=0.0
   74 CONTINUE
      OUTPUT=3
      IF (NK2.EQ.0) OUTPUT=1
      ERRS=0
      NUMERS=0
```

```
          OUTPT(1)=BLANK                                             ROOT      173
          INPT(1)=BLANK                                              ROOT      174
          NUU=NU*2                                                   ROOT      175
          DO 50 I=1,NUU                                              ROOT      176
          DO 51 J=1,NX                                               ROOT      177
          IF (H(I,J).EQ.0.0) GO TO 51                                ROOT      178
          GO TO 52                                                   ROOT      179
       51 CONTINUE                                                   ROOT      180
       50 CONTINUE                                                   ROOT      181
       52  IF (I.EQ.1) GO TO 54                                      ROOT      182
          IF (I.GT.NU) I=NUU-NU                                      ROOT      183
          DO 53 J=1,NX                                               ROOT      184
          H(1,J)=H(I,J)                                              ROOT      185
          B(J,1)=B(J,I)                                              ROOT      186
       53 CONTINUE                                                   ROOT      187
          DO 56 J=1,NU                                               ROOT      188
          F(1,J) =F(I,J)                                             ROOT      189
       56 CONTINUE                                                   ROOT      190
       54 NY=1                                                       ROOT      191
          NU=1                                                       ROOT      192
          WRITE (3,55)                                               ROOT      193
       55 FORMAT (//,10X,*ROOT LOCUS OPTION COMPUTES THE ZEROES OF THE FIRST ROOT      194
         1 NON ZERO ROW*/10X,*OF K1 OR K3 RESPECTIVELY DUE TO INDICATED    ROOT      195
         2 INPUT*/)                                                  ROOT      196
          RETURN                                                     ROOT      197
          END                                                        ROOT      198
```

```
      SUBROUTINE SETUP (J,M,MM,P,A,B,C,H,G,F,K1,K2,K3,K4,D,W1,W2,W3,        SETUP      2
     1 ROOTR,                                                               SETUP      3
     2 MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                           SETUP      4
C                                                                           SETUP      5
C       THIS SUBPROGRAM REDUCES THE ORIGINAL INPUT MATRICES FOR             SETUP      6
C       THE OPEN AND CLOSED LOOP SYSTEMS TO STANDARD FORM AND               SETUP      7
C       SCALES THE RESULTING SYSTEM WITH A DIAGONAL SIMILARITY              SETUP      8
C       TRANSFORMATION.                                                     SETUP      9
C     CORR CTION MADE BY G. NORRIS JULY 5 73                                SETUP     10
C                                                                           SETUP     11
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,         SETUP     12
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,             SETUP     13
     2IGC,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT                      SETUP     14
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP     SETUP     15
      INTEGER DIGITL,SCAPLT,ZOH                                             SETUP     16
      COMMON/ACOND/ DELT,FINALT,IFREQ,FFREQ,DELFRQ,GAIN1,GAIN2,MMM          SETUP     17
      REAL K1, K2, K3, K4, IFREQ, MMM                                       SETUP     18
      DIMENSION M(10,20),MM(20),P(20)                                       SETUP     19
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),      SETUP     20
     1K1(MU,MX),K2(MU,MX),K3(MU,MX),K4(MU,MX),D(MU,MU),                     SETUP     21
     2W1(MX,MX),W2(MX,MX),W3(MX,MX),ROOTR(MX)                               SETUP     22
C     DIMENSION W1(15,15), W2(15,15), W3(15,15)                             SETUP     23
C     DIMENSION A(15,15), B(15,10), C(15,15), H(15,15), G(15,15)            SETUP     24
C     DIMENSION F(15,10), D(10,10), ROOTR(15),                              SETUP     25
C    1 K1(10,15), K2(10,15), K3(10,15), K4(10,15)                           SETUP     26
      COMMON/BLKDAT/NUMER,DENOM,GAIN,GRAPH,BLOCK,STATE,YTOV,ZTOU,YZTOK,     SETUP     27
     1ITHINY,ITHINU,NBLOCK,NYTOV,NZTOU,NXYU,NYZTOK,NXT,NYT,NUT,NY1,NU1      SETUP     28
      REAL NUMER                                                            SETUP     29
      INTEGER GRAPH,BLOCK,STATE,YTOV,ZTOU,YZTOK                             SETUP     30
      DIMENSION GRAPH(20,5),BLOCK(20,3),NUMER(20,5),DENOM(20,5),            SETUP     31
     XGAIN(20),STATE(20,4),ITHINY(30),ITHINU(20),YTOV(20,2),                SETUP     32
     X  ZTOU(20,2),NXYU(8),YZTOK(20,2)                                      SETUP     33
      COMMON/SUBWRIT/ ISUBNAM                                               SETUP     34
      IF(ISUBNAM.GE.2) WRITE(3,990)                                         SETUP     35
  990 FORMAT(1X,*SETUP*)                                                    SETUP     36
      IMIX=0                                                                SETUP     37
      ICK=0                                                                 SETUP     38
      MAT1=MX                                                               SETUP     39
      MAT2=MX                                                               SETUP     40
      MAT3=MX                                                               SETUP     41
      MAT4=MX                                                               SETUP     42
      MAT5=MX                                                               SETUP     43
      MAT6=MX                                                               SETUP     44
      IF (CMAT.EQ.0) GO TO 5                                                SETUP     45
C                                                                           SETUP     46
C     ELIMINATE C MATRIX                                                    SETUP     47
C                                                                           SETUP     48
      CALL INVR (C,W2,NX,1,                                                 SETUP     49
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                            SETUP     50
      CALL MULT (W2,A,W1,NX,NX,NX,                                          SETUP     51
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                            SETUP     52
      CALL MAKE (A,W1,NX,NX,                                                SETUP     53
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                            SETUP     54
      MAT4=MU                                                               SETUP     55
      CALL MULT (W2,B,W1,NX,NX,NU,                                          SETUP     56
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                            SETUP     57
      MAT2=MU                                                               SETUP     58
```

```
      MAT4=MX
      CALL MAKE (B,W1,NX,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CMAT=0
    5 GO TO (50,10,10),SYSTEM
   10 IF (NK2.EQ.0) GO TO 40
C
C     ELIMINATE K2,K4 MATRICES
C
      IF (SYSTEM.EQ.3.AND.MIXED.NE.1) GO TO 660
      MAT1=MU
      MAT2=MX
      MAT4=MU
      CALL MULT(K2,B,W1,NU,NX,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      DO 12 I=1,NU
      DO 13 J=1,NU
      W2(I,J)=-W1(I,J)
   13 CONTINUE
      W2(I,I)=1.0+W2(I,I)
   12 CONTINUE
      IF (DIGITL .NE. 1) GO TO 838
      DO 339 I=1,NU
      DO 339 J=1,NUC
      W2(I,J)= 0.0
      IF (I .EQ. J) W2(I,I)= 1.0
  339 CONTINUE
  838 CALL INVR (W2,W1,NU,0,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MU
      MAT4=MX
      CALL MULT (K2,A,W2,NU,NX,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL ADD  (1.0,K1,1.0,W2,W3,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MX
      CALL MULT (W1,W3,W2,NU,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT3=MU
      MAT4=MU
      CALL MULT (W1,D,W3,NU,NU,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MU
      MAT2=MU
      MAT3=MX
      MAT4=MX
      CALL MAKE (D,W3,NU,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MX
      CALL MAKE (K1,W2,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      NK2=0
      GO TO 40
  660 MAT1=MU
      MAT2=MX
      CALL MULT (K2,A,W1,NU,NX,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
```

```
      CALL MULT (K4,A,W2,NU,NX,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL ADD (1.0,K1,1.0,W1,W3,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL ADD (1.0,K3,1.0,W2,W1,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL MAKE (K1,W3,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL MAKE (K3,W1,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT4=MU
      CALL MULT (K2,B,W1,NU,NX,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL MULT (K4,B,W2,NU,NX,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT4=MX
      CALL MAKE (K2,W1,NU,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL MAKE (K4,W2,NU,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
   40 CONTINUE
   50 IF (OUTPUT.EQ.1.OR.OUTPUT.EQ.3) GO TO 51
C
C     ELIMINATE G MATRIX
C
      MAT1=MY
      MAT2=MX
      MAT3=MY
      MAT4=MU
      CALL MULT (G,B,W1,NY,NX,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MU
      MAT4=MX
      MAT3=MX
      CALL ADD (1.0,F,1.0,W1,W2,NY,NU,
     1MX,MY,MU,MS,MAT1,MAT3,MAT4,MAT5,MAT6)
      CALL MAKE (F,W2,NY,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MX
      CALL MULT (G,A,W1,NY,NX,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL ADD (1.0,H,1.0,W1,W2,NY,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL MAKE (H,W2,NY,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (OUTPUT.EQ.2) OUTPUT=3
      IF (OUTPUT.EQ.4) OUTPUT=3
   51 IF (MIXED.NE.1.OR.READ.EQ.4) GO TO 210
C
C     AUGMENT SYSTEM WITH CONTROL SYSTEM MODELED IN CLASS
C
      MAT1=MX
      MAT2=MX
      CALL ZOT1 (C,MX,MX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      DO 211 I=1,NY
      C(I,I)=1.0
```

```
  211 CONTINUE                                                         SETUP     173
      CALL CLASS        (A,B,C,H,G,F,D,W1,W2,W3,                       SETUP     174
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       SETUP     175
  210 CONTINUE                                                         SETUP     176
      MAT1=MX                                                          SETUP     177
      MAT2=MY                                                          SETUP     178
      MAT3=MX                                                          SETUP     179
      MAT4=MX                                                          SETUP     180
      CALL MAKE (W1,A,NX,NX,                                           SETUP     181
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       SETUP     182
      IF (NSCALE.EQ.0) GO TO 52                                        SETUP     183
      IF (NSCALE.EQ.2) GO TO 52                                        SETUP     184
C                                                                      SETUP     185
C     SYSTEM SCALED BY A DIAGONAL SIMILARITY TRANSFORMATION            SETUP     186
C                                                                      SETUP     187
      ICK=1                                                            SETUP     188
      CALL ASCALE (NX,J,M,MM,P,W1,W2,W3,ROOTR,                         SETUP     189
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       SETUP     190
      DO 60 I= 1,NX                                                    SETUP     191
      DO 60 K= 1,NU                                                    SETUP     192
      K1(K,I)=K1(K,I)/P(I)                                             SETUP     193
      IF (SYSTEM.NE.3.AND.NK2.NE.1) GO TO 661                          SETUP     194
      K3(K,I)=K3(K,I)/P(I)                                             SETUP     195
  661 B(I,K)=B(I,K)*P(I)                                               SETUP     196
   60 CONTINUE                                                         SETUP     197
      IF (SYSTEM.EQ.3.OR.MIXED.EQ.1) GO TO 662                         SETUP     198
      DO 70 I=1,NY                                                     SETUP     199
      DO 70 K=1,NX                                                     SETUP     200
      H(I,K)=H(I,K)/P(K)                                               SETUP     201
   70 CONTINUE                                                         SETUP     202
  662 MAT1=MX                                                          SETUP     203
      MAT2=MX                                                          SETUP     204
      CALL MAKE (A,W1,NX,NX,                                           SETUP     205
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       SETUP     206
      IF (IPT.LT.1) GO TO 52                                           SETUP     207
      WRITE (3,110)                                                    SETUP     208
  110 FORMAT (1H1,10X,*THE REDUCED AND SCALED SYSTEM IS*)              SETUP     209
      CALL SFIT (A,B,C,H,G,F,K1,K2,K3,K4,D,                            SETUP     210
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       SETUP     211
   52 IF (DIGITL.NE.1) GO TO 80                                        SETUP     212
      IF(NZTOU.EQ.0.AND.NYTOV.EQ.0) GO TO 465                          SETUP     213
      IMIX=1                                                           SETUP     214
      GO TO 80                                                         SETUP     215
C                                                                      SETUP     216
C     DISCRETIZE CONTINUOUS PORTION OF SAMPLED-DATA SYSTEM             SETUP     217
C                                                                      SETUP     218
  465 IF (MMM.EQ.0.0) GO TO 455                                        SETUP     219
      IF (SYSTEM.NE.1) GO TO 456                                       SETUP     220
      GO TO 457                                                        SETUP     221
  456 WRITE (3,458)                                                    SETUP     222
  458 FORMAT (/,* ROOT LOCUS AND CLOSED LOOP OPTIONS NOT ALLOWED WITH MO SETUP     223
     -DIFIED Z-TRANSFORMS*/)                                           SETUP     224
  457 XM=MMM*DELT                                                      SETUP     225
      CALL EAT (XM,A,W1,W2,W3,C,NXC,                                   SETUP     226
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                       SETUP     227
      IF (ZOH.EQ. 0) GO TO 459                                         SETUP     228
      OUTPUT=3                                                         SETUP     229
```

```
      MAT1=MY
      CALL MULT (H,W2,W3,NY,NXC,NXC,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT4=MU
      MAT1=MX
      CALL MULT (W3,P,C,NY,NXC,NUC,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MY
      MAT3=MY
      MAT4=MU
      MAT5=MY
      MAT6=MU
      CALL ADD (1.,C,1.,F,F,NY,NUC,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
  459 MAT1=MY
      MAT3=MX
      MAT4=MX
      MAT5=MY
      MAT6=MX
      CALL MULT (H,W1,W3,NY,NXC,NXC,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT5)
      CALL MAKE (H,W3,NY,NXC,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (ZOH.EQ.NUC) GO TO 455
      NN1=ZOH+1
      DO 460 I=1,NY
      DO 460 J=NN1,NUC
      IF (F(I,J).NE.0) GO TO 461
  460 CONTINUE
      GO TO 462
  461 WRITE (3,448)
  462 DO 463 I=1,NY
      DO 463 J=NN1,NUC
      XX=0.0
      DO 464 K=1,NXC
      XX=XX+H(I,K)*B(K,J)
  464 CONTINUE
      F(I,J)=XX
  463 CONTINUE
      OUTPUT=3
  455 CONTINUE
      CALL EAT (DELT,A,W1,W2,W3,C,NXC,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      NN1=ZOH+1
      DO 7 I=1,NXC
      DO 7 J=1,NXC
      A(I,J)= W1(I,J)
    7 CONTINUE
      IF (ZOH.EQ.0) GO TO 441
      MAT1=MX
      MAT4=MU
      CALL MULT (W2, B, W3, NXC, NXC, ZOH,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MU
      MAT4=MX
      CALL MAKE (B,W3,NXC,ZOH,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
```

```
  441 IF (ZOH.EQ.NUC) GO TO 442
      IF (ZOH.LT.NUC) GO TO 443
      WRITE (3,444)
  444 FORMAT (/,* ZOH GREATER THAN NUC*/)
  443 DO 445 I=1,NY
      DO 445 J=NN1,NUC
      IF (F(I,J).NE.0.) GO TO 446
  445 CONTINUE
      GO TO 447
  446 WRITE (3,448)
  448 FORMAT (/* F MATRIX NOT NULL *)
  447 DO 449 I=1,NXC
      DO 449 J=NN1,NUC
      XX=0.0
      DO 450 K=1,NXC
      XX=XX+W1(I,K)*B(K,J)
  450 CONTINUE
      W3(I,J)= XX
  449 CONTINUE
      IF (MMM.NE.0.0) GO TO 470
      DO 452 I=1,NY
      DO 452 J=NN1,NUC
      XX=0.0
      DO 453 K=1,NXC
      XX=XX+H(I,K)* B(K,J)
  453 CONTINUE
      F(I,J) = XX
  452 CONTINUE
  470 DO 451 I=1,NXC
      DO 451 J=NN1,NUC
      B(I,J)= W3(I,J)
  451 CONTINUE
      OUTPUT=3
  442 CONTINUE
   15 IF (IPT.LT.1) GO TO 80
      WRITE (3,111)
  111 FORMAT (1H1,10X,* THE DESCRETIZED SYSTEM IS*)
      CALL SPIT (A, B, C, H, G, F, K1, K2, K3, K4, D,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
   80 CONTINUE
      IF (MIXED.NE.1) GO TO 500
      IF (IMIX.EQ.2) GO TO 500
C
C     SUBSYSTEMS DESCRIBED IN STEP1 AND STEP2 OF MIXED LOADING OPTION
C     ARE COUPLED USING YTOV AND ZTCL
C
      IF (NBLOCK.EQ.0) GO TO 300
      MAT1=MX
      MAT2=MY
      MAT3=MX
      MAT4=MY
      CALL ZCT1(W1,NUT,NYT,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (NZTOU.EQ.0) GO TO 300
      DO 301 I=1,NZTOU
      W1(ZTOU(I,2),ZTOU(I,1)+NY1)=1.0
  301 CONTINUE
```

```
      300 IF (NYTOV.EQ.0) GO TO 302                                       SETUP     344
          DO 303 I=1,NYTOV                                                SETUP     345
          W1(YTOV(I,2)+NU1,YTOV(I,1))=1.0                                 SETUP     346
      303 CONTINUE                                                        SETUP     347
          MAT1=MY                                                         SETUP     348
          MAT2=MU                                                         SETUP     349
      302 CALL MULT (F,W1,W3,NYT,NUT,NYT,                                 SETUP     350
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                      SETUP     351
          DO 306 I=1,NYT                                                  SETUP     352
          DO 307 J=1,NYT                                                  SETUP     353
          W3(I,J)=-W3(I,J)                                                SETUP     354
      307 CONTINUE                                                        SETUP     355
          W3(I,I)=1.0+W3(I,I)                                             SETUP     356
      306 CONTINUE                                                        SETUP     357
          CALL INVR(W3,C,NYT,1,                                           SETUP     358
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                      SETUP     359
          MAT1=MX                                                         SETUP     360
          MAT2=MX                                                         SETUP     361
          MAT3=MY                                                         SETUP     362
          CALL MULT (C,H,W3,NYT,NYT,NXT,                                  SETUP     363
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                      SETUP     364
          MAT1=MY                                                         SETUP     365
          MAT3=MX                                                         SETUP     366
          CALL MAKE (H,W3,NYT,NXT,                                        SETUP     367
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                      SETUP     368
          IF (ICK.NE.1) GO TO 312                                         SETUP     369
          DO 313 I=1,NY                                                   SETUP     370
          DO 313 J=1,NX                                                   SETUP     371
          H(I,J)=H(I,J)/P(J)                                              SETUP     372
      313 CONTINUE                                                        SETUP     373
      312 CONTINUE                                                        SETUP     374
          MAT1=MX                                                         SETUP     375
          MAT2=MX                                                         SETUP     376
          MAT3=MY                                                         SETUP     377
          MAT4=MU                                                         SETUP     378
          CALL MULT (C,F,W3,NYT,NYT,NUT,                                  SETUP     379
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                      SETUP     380
          MAT1=MY                                                         SETUP     381
          MAT2=MU                                                         SETUP     382
          MAT3=MX                                                         SETUP     383
          MAT4=MX                                                         SETUP     384
          CALL MAKE(F,W3,NYT,NUT,                                         SETUP     385
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                      SETUP     386
          MAT1=MX                                                         SETUP     387
          MAT2=MU                                                         SETUP     388
          CALL MULT (B,W1,W3,NXT,NUT,NYT,                                 SETUP     389
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                      SETUP     390
          MAT2=MX                                                         SETUP     391
          MAT3=MY                                                         SETUP     392
          CALL MULT (W3,H,W1,NXT,NYT,NXT,                                 SETUP     393
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                      SETUP     394
          MAT3=MX                                                         SETUP     395
          CALL ADD (1.0,A,1.0,W1,A,NXT,NXT,                               SETUP     396
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                      SETUP     397
          MAT3=MY                                                         SETUP     398
          MAT4=MU                                                         SETUP     399
          CALL MULT (W3,F,W1,NXT,NYT,NUT,                                 SETUP     400
```

```
                 1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                  SETUP     401
                  IF (ICK.NE.1) GO TO 310                                     SETUP     402
                  DO 311 I=1,NX                                               SETUP     403
                  DO 311 J=1,NX                                               SETUP     404
                  W1(I,J)=W1(I,J)/P(J)                                        SETUP     405
              311 CONTINUE                                                    SETUP     406
              310 CONTINUE                                                    SETUP     407
                  MAT3=MX                                                     SETUP     408
                  MAT4=MY                                                     SETUP     409
                  MAT1=MX                                                     SETUP     410
                  MAT2=MU                                                     SETUP     411
                  CALL ADD (1.0,P ,1.0,W1,W2,NXT,NUT,                         SETUP     412
                 1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                  SETUP     413
                  MAT2=MU                                                     SETUP     414
                  CALL MAKE (U,W2,NXT,NUT,                                    SETUP     415
                 1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                  SETUP     416
              500 CONTINUE                                                    SETUP     417
                  IF (ITMIX.NE. 1) GO TO 466                                  SETUP     418
                  ITMIX=2                                                     SETUP     419
                  GO TO 465                                                   SETUP     420
              456 IF (READ.EQ.4.OR.MIXED.EQ.1) GO TO 728                      SETUP     421
                  GO TO 720                                                   SETUP     422
              728 IF (NYZTOK.EQ.0) GO TO 720                                  SETUP     423
            C                                                                 SETUP     424
            C     CONSTRUCT FEEDBACK MATRICES K1 AND K3 FROM YZTOK            SETUP     425
            C                                                                 SETUP     426
                  IF (SYSTEM.EQ.1) SYSTEM=2                                   SETUP     427
                  K=1                                                         SETUP     428
                  I=1                                                         SETUP     429
              716 MAT1=MX                                                     SETUP     430
                  MAT2=MX                                                     SETUP     431
                  CALL ZCT1(W1,NUT,NX,                                        SETUP     432
                 1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                  SETUP     433
                  CALL ZCT1(W2,NUT,NUT,                                       SETUP     434
                 1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                  SETUP     435
              713 CONTINUE                                                    SETUP     436
                  DO 714 J=1,NX                                               SETUP     437
                  W1(YZTOK(I,2),J)= H(YZTOK(I,1),J) +W1(YZTOK(I,2),J)         SETUP     438
              714 CONTINUE                                                    SETUP     439
                  DO 715 J=1,NUT                                              SETUP     440
                  W2(YZTOK(I,2),J)= F(YZTOK(I,1),J) +W2(YZTOK(I,2),J)         SETUP     441
              715 CONTINUE                                                    SETUP     442
                  I=I+1                                                       SETUP     443
                  IF (SYSTEM .EQ. 2) GO TO 730                                SETUP     444
                  DO 800 J=1,NYZTOK                                           SETUP     445
                  DO 801 L=1,20                                               SETUP     446
                  IF (YZTOK(J,2) .EQ. ITHINU(L)) GO TO 800                    SETUP     447
              801 CONTINUE                                                    SETUP     448
                  WRITE (3,804) YZTOK(J,2)                                    SETUP     449
              804 FORMAT (/10X,*NO. *,I5,* INPUT MUST BE SAVED FOR FEEDBACK*/)  SETUP     450
                  DO 802 L=1,20                                               SETUP     451
                  IF (ITHINU(21-L).EQ.0)  GO TO 802                           SETUP     452
                  IF (YZTOK(J,2).GT.ITHINU(21-L)) GO TO 803                   SETUP     453
                  ITHINU(22-L)=ITHINU(21-L)                                   SETUP     454
              802 CONTINUE                                                    SETUP     455
              803 ITHINU(22-L)=YZTOK(J,2)                                     SETUP     456
              800 CONTINUE                                                    SETUP     457
```

```
      MAT1=MU                                                           SETUP    458
      MAT2=MU                                                           SETUP    459
      IF (K .EQ. 2) GO TO 731                                           SETUP    460
      CALL MAKE (K1,W1,NUT,NX,                                          SETUP    461
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SETUP    462
      CALL MAKE (K2,W2,NUT,NUT,                                         SETUP    463
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SETUP    464
      IF (NY2TOK .EQ. 1) GO TO 732                                      SETUP    465
      K=2                                                               SETUP    466
      GO TO 716                                                         SETUP    467
  731 CALL MAKE (K3,W1,NUT,NX,                                          SETUP    468
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SETUP    469
      CALL MAKE (K4,W2,NUT,NUT,                                         SETUP    470
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SETUP    471
      IF (NY2TOK .GT. 2) WRITE (3,722)                                  SETUP    472
  722 FORMAT (/*ONLY 2 LOOPS ALLOWED*)                                  SETUP    473
  732 DO 733 L=1,NUT                                                    SETUP    474
      DO 733 J=1,NUT                                                    SETUP    475
      IF (K2(L,J) .NE. 0.0) GO TO 734                                   SETUP    476
      IF (K4(L,J) .NE. 0.0) GO TO 734                                   SETUP    477
  733 CONTINUE                                                          SETUP    478
      GO TO 720                                                         SETUP    479
  734 NK2=1                                                             SETUP    480
      GO TO 720                                                         SETUP    481
  730 IF (I .LE. NY2TOK) GO TO 713                                      SETUP    482
      DO 736 J=1,NUT                                                    SETUP    483
      DO 737 L=1,NUT                                                    SETUP    484
      W2(J,L)=-W2(J,L)                                                  SETUP    485
  737 CONTINUE                                                          SETUP    486
      W2(J,J)=1.0+W2(J,J)                                               SETUP    487
  736 CONTINUE                                                          SETUP    488
      MAT1=MX                                                           SETUP    489
      MAT2=MX                                                           SETUP    490
      CALL INVP (W2,W3,NUT,1,                                           SETUP    491
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SETUP    492
      MAT1=MU                                                           SETUP    493
      MAT2=MU                                                           SETUP    494
      MAT3=MX                                                           SETUP    495
      MAT4=MX                                                           SETUP    496
      CALL MAKE (D,W3,NUT,NUT,                                          SETUP    497
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SETUP    498
      MAT1=MX                                                           SETUP    499
      MAT2=MX                                                           SETUP    500
      CALL MULT (W3,W1,W2,NUT,NUT,NX,                                   SETUP    501
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SETUP    502
      MAT1=MU                                                           SETUP    503
      CALL MAKE (K1,W2,NUT,NX,                                          SETUP    504
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SETUP    505
  720 CONTINUE                                                          SETUP    506
      IF (DIGITL .NE. 1) GO TO 6                                        SETUP    507
      DO 837 I=1,NU                                                     SETUP    508
      DO 837 J=1,NUC                                                    SETUP    509
      K2(I,J)=0.0                                                       SETUP    510
      K4(I,J)=0.0                                                       SETUP    511
  837 CONTINUE                                                          SETUP    512
    6 CONTINUE                                                          SETUP    513
      IF (MULTPT .NE. 0) GO TO 727                                      SETUP    514
```

```
      IF (SYSTEM .NE. 2) GO TO 727
C
C     FOR CLOSED LOOP ANALYSIS, INCORPORATE FEEDBACK MATRX,K1
C     AND FEEDFORWARD MATRIX,C
C
      MAT1=MX
      MAT2=MU
      MAT3=MU
      MAT4=MY
      CALL MULT (B,K1,W1,NX,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MX
      MAT3=MX
      CALL ADD (1.0,A,1.0,W1,W2,NX,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL MAKE (A,W2,NX,NX,
     1MX,MU,MY,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MU
      MAT3=MU
      MAT4=MU
      CALL MULT (B,D,W1,NX,NU,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT3=MX
      MAT4=MX
      CALL MAKE (B,W1,NX,NU,
     1MX,MU,MY,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MY
      MAT2=MU
      MAT3=MU
      CALL MULT (F,K1,W1,NY,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MX
      MAT3=MX
      CALL ADD (1.0,H,1.0,W1,W2,NY,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL MAKE (H,W2,NY,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MU
      MAT3=MU
      MAT4=MU
      CALL MULT (F,D,W1,NY,NU,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT3=MX
      MAT4=MX
      CALL MAKE (F,W1,NY,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      SYSTEM=1
  727 CONTINUE
      IF (READ.EQ.4.AND.SYSTEM.EQ.3) GO TO 828
      IF (MIXED.NE.1) GO TO 721
  828   IF (NBLOCK.EQ.0) GO TO 520
C
C    THIN THE INPUT VECTOR (MIXED LOADING OPTION)
C
      DO 394 I=1,10
      DO 321 J=1,NX
      B(J,I)=B(J,ITHINU(I))
```

```
  321 CONTINUE
      DO 323 J=1,NY
      F(J,I)=F(J,ITHINU(I))
  323 CONTINUE
      IF (SYSTEM .EQ. 3 .OR.  MULTPT .NE. 0) GO TO 322
      GO TO 305
  322 DO 325 J=1,NX
      K1(I,J)=K1(ITHINU(I),J)
      K3(I,J)=K3(ITHINU(I),J)
  325 CONTINUE
      DO 326 J=1,NUT
      D(J,I)=D(J,ITHINU(I))
  326 CONTINUE
      DO 327 J=1,NUT
      C(I,J)=D(ITHINU(I),J)
  327 CONTINUE
  305 IF (ITHINU(I+1) .EQ. 0) GO TO 320
  304 CONTINUE
  320 IF (I.LT.NUT) NU=I
  721 CONTINUE
      IF (READ.EQ.4.OR.MIXED.EQ.1.OR.READ.EQ.3) GO TO 729
      GO TO 520
  729 CONTINUE
C
C     THIN THE OUTPUT VECTOR (MIXED LOADING OPTION)
C
      DO 501 I=1,30
      DO 502 J=1,NXT
      H(I,J)=H(ITHINY(I),J)
  502 CONTINUE
      DO 503 J=1,NUT
      F(I,J)=F(ITHINY(I),J)
  503 CONTINUE
      IF (ITHINY(I+1) .EQ. 0) GO TO 521
  501 CONTINUE
  521 IF (I.LT.NY) NY=I
      IF (IMIX.NE.1) GO TO 520
      IMIX=2
      GO TO 465
  520 MAT1=MX
      MAT2=MX
      MAT3=MX
      MAT4=MX
      CALL MAKE (W1,A,NX,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (MIXED.EQ.1)                        GO TO 102
      IF (IPT.LT.1) GO TO 200
  102 WRITE (3,100)
  100 FORMAT(1H1,10X,* THE FINAL REDUCED SYSTEM IS*//)
      CALL SPIT (A,B,C,H,G,F,K1,K2,K3,K4,D,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
  200 CONTINUE
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE SPIT    (A,B,C,H,G,F,K1,K2,K3,K4,D,                    SPIT       2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SPIT       3
C                                                                       SPIT       4
C        THIS SUBROUTINE PRINTS ALL MATRICES ACCORDING TO THE           SPIT       5
C        PARAMETERS, SYSTEM AND OUTPUT, USING SUBROUTINE SPIT1          SPIT       6
C                                                                       SPIT       7
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,     SPIT       8
     1CONTUR,NUMERS,ERPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,         SPIT       9
     2IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOM,KCUNT                  SPIT      10
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, ERPS,TRESP SPIT      11
      INTEGER DIGITL,SCAPLT,ZOM                                         SPIT      12
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),  SPIT      13
     1K1(MU,MX),K2(MU,MX),K3(MU,MX),K4(MU,MX),D(MU,MU)                  SPIT      14
      REAL K1,K2,K3,K4                                                  SPIT      15
      COMMON/SUBWRIT/ ISUBNAM                                           SPIT      16
      IF(ISUBNAM.GE.2) WRITE(3,990)                                     SPIT      17
  990 FORMAT(1X,*SPIT*)                                                 SPIT      18
      MAT1=MX                                                           SPIT      19
      MAT2=MX                                                           SPIT      20
      WRITE (3,10)                                                      SPIT      21
   10 FORMAT (// 10X,*THE A MATRIX IS*/)                                SPIT      22
      CALL SPIT1 (A,NX,NX,                                              SPIT      23
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SPIT      24
      WRITE (3,11)                                                      SPIT      25
   11 FORMAT (// 10X,*THE B MATRIX IS*/)                                SPIT      26
      MAT2=MU                                                           SPIT      27
      CALL SPIT1 (B,NX,NU,                                              SPIT      28
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SPIT      29
      MAT2=MX                                                           SPIT      30
      IF (CMAT.EQ.0) GO TO 21                                           SPIT      31
      WRITE (3,20)                                                      SPIT      32
   20 FORMAT (//10X,*THE C MATRIX IS*/)                                 SPIT      33
      CALL SPIT1 (C,NX,NX,                                              SPIT      34
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SPIT      35
   21 CONTINUE                                                          SPIT      36
      GO TO (50,50,60),SYSTEM                                           SPIT      37
   60 WRITE (3,12)                                                      SPIT      38
   12 FORMAT (// 10X,*THE K1 MATRIX IS*/)                               SPIT      39
      MAT1=MU                                                           SPIT      40
      CALL SPIT1 (K1,NU,NX,                                             SPIT      41
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SPIT      42
      IF (NK2.EQ.0) GO TO 62                                            SPIT      43
      WRITE (3,13)                                                      SPIT      44
   13 FORMAT (// 10X,*THE K2 MATRIX IS*/)                               SPIT      45
      CALL SPIT1 (K2,NU,NX,                                             SPIT      46
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SPIT      47
   62 CONTINUE                                                          SPIT      48
      IF (N2.EQ.0)GO TO 64                                              SPIT      49
      WRITE (3,14)                                                      SPIT      50
   14 FORMAT (// 10X,*THE K3 MATRIX IS*/)                               SPIT      51
      CALL SPIT1 (K3,NU,NX,                                             SPIT      52
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SPIT      53
      IF (NK2.EQ.0) GO TO 64                                            SPIT      54
      WRITE (3,15)                                                      SPIT      55
   15 FORMAT (// 10X,*THE K4 MATRIX IS*/)                               SPIT      56
      CALL SPIT1 (K4,NU,NX,                                             SPIT      57
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                        SPIT      58
```

```
   64 CONTINUE
      IF (MIXED.EQ.1) GO TO 50
      GO TO 200
   50 WRITE (3,16)
   16 FORMAT (// 10X,*THE H MATRIX IS*/)
   17 FORMAT (// 10X,*THE G MATRIX IS*/)
   18 FORMAT (//10X,*THE F MATRIX IS*/)
   19 FORMAT (//10X,*THE D MATRIX IS*/)
      MAT1=MY
      CALL SPIT1 (H,NY,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      GO TO (100,56,57,58),OUTPUT
   56 WRITE (3,17)
      CALL SPIT1 (G,NY,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      GO TO 100
   57 WRITE (3,18)
      MAT2=MU
      CALL SPIT1 (F,NY,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      GO TO 100
   58 WRITE (3,17)
      CALL SPIT1 (G,NY,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      WRITE (3,18)
      MAT2=MU
      CALL SPIT1 (F,NY,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
  100 GO TO (200,110,200),SYSTEM
  110 WRITE (3,12)
      MAT1=MU
      MAT2=MX
      CALL SPIT1 (K1,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      IF (NK2.EQ.0) GO TO 66
      WRITE (3,13)
      CALL SPIT1 (K2,NU,NX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
   66 CONTINUE
      WRITE (3,19)
      MAT2=MU
      CALL SPIT1 (D,NU,NU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
  200 CONTINUE
      RETURN
      END
```

```
SUBROUTINE SPIT1        73/74    OPT=1                              FTN 4.2+75060        01/09/76  14.15.49.

                  SUBROUTINE SPIT1 (A,N,M,                                                SPIT1         2
                 1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                              SPIT1         3
                  DIMENSION A(MAT1,MAT2)                                                  SPIT1         4
                  COMMON/SUBWRIT/ ISUBNAM                                                 SPIT1         5
                  IF(ISUBNAM.GE.2) WRITE(3,990)                                           SPIT1         6
              990 FORMAT(1X,*SPIT1*)                                                      SPIT1         7
                  WRITE (3,10) N,M                                                        SPIT1         8
               10 FORMAT (2I10/)                                                          SPIT1         9
                  DO 20 I=1,N                                                             SPIT1        10
                  WRITE (3,30) (A(I,J),J=1,M)                                             SPIT1        11
               20 CONTINUE                                                                SPIT1        12
               30 FORMAT (10E12.4)                                                        SPIT1        13
                  RETURN                                                                  SPIT1        14
                  END                                                                     SPIT1        15
```

```
      SUBROUTINE THIST (A,B,C,H,F,W1,W2,W3,ROOTR,ROOTI,U,K1,D,Z,V,        SPIT1      16
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                          SPIT1      17
C                                                                         SPIT1      18
C       THIS SUBROUTINE COMPUTES A TABULATED TIME HISTORY RESPONSE        SPIT1      19
C       OF THE SYSTEM.  INPUT TO THE SYSTEM IS UNDER USER CONTROL         SPIT1      20
C       THROUGH A CALL TO SUBROUTINE INPUT (DELT,T,U) WHERE DELT IS       SPIT1      21
C       THE TIME INCREMENT, T IS THE TIME, AND U IS THE INPUT VECTOR      SPIT1      22
C       OF DIMENSION MX.  THE RESPONSE IS COMPUTED USING THE STATE        SPIT1      23
C       TRANSITION MATRIX.  THE TRANSITION MATRIX IS COMPUTED BY THE      SPIT1      24
C       SUBROUTINE EAT USING THE SERIES EXPANSION TECHNIQUE.  TEN TERMS   SPIT1      25
C       OF THE SERIES ARE USED.                                           SPIT1      26
C     CORRECTION MADE BY G. NORRIS JULY 5 73                              SPIT1      27
C                                                                         SPIT1      28
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,       SPIT1      29
     1CONTUR,NUMERS,FRES,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,           SPIT1      30
     2IGC,FORM,IPT,READ3,MIXEC,MULTRT,SCAPLT,ZOH,KCLNT                    SPIT1      31
      COMMON/ACOND/ DELT,FINALT,IFREQ,FFREQ,DELFRQ,GAIN1,GAIN2,M          SPIT1      32
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRES,TRESP   SPIT1      33
      INTEGER DIGITL,SCAPLT,ZOH                                           SPIT1      34
      REAL K1, IFREQ, M                                                   SPIT1      35
      COMMON/LABEL/INPT,OUTPT,TITLE                                       SPIT1      36
      REAL   INPT(10), OUTPT(20),TITLE(8)                                 SPIT1      37
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),F(MY,MU),             SPIT1      38
     1W1(MX,MX),W2(MX,MX),W3(MX,MX),ROOTR(MX),ROOTI(MX),U(MX),            SPIT1      39
     2K1(MU,MX),D(MU,MU),Z(MX),V(MX)                                      SPIT1      40
C     DIMENSION A(15,15), B(15,10), C(15,15), H(15,15)                    SPIT1      41
C     DIMENSION F(15,10),K1(10,15), D(10,10), Z(15), V(15), U(15)         SPIT1      42
C     DIMENSION W1(15,15), W2(15,15), W3(15,15)                           SPIT1      43
C     DIMENSION ROOTR(15), ROOTI(15)                                      SPIT1      44
C     DIMENSION K2(10,15), K3(10,15), K4(10,15)                           SPIT1      45
      REAL PST                                                            SPIT1      46
      COMMON/SUBWRIT/ ISUBNAM                                             SPIT1      47
      DATA PST/10HTIME      /                                             SPIT1      48
      IF(ISUBNAM.GE.2) WRITE(3,990)                                       SPIT1      49
  990 FORMAT(1X,*THIST*)                                                  SPIT1      50
      IF (DIGITL.NE.0) GO TO 60                                           SPIT1      51
      IF (MULTRT.GT.0) GO TO 60                                           SPIT1      52
      CALL EAT (DELT,A,W1,W2,W3,C,NX,                                     SPIT1      53
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                          SPIT1      54
      MAT1=MX                                                             SPIT1      55
      MAT2=MX                                                             SPIT1      56
      MAT3=MX                                                             SPIT1      57
      MAT4=MX                                                             SPIT1      58
      MAT5=MX                                                             SPIT1      59
      MAT6=MX                                                             SPIT1      60
      CALL MAKE (A,W1,NX,NX,                                              SPIT1      61
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                          SPIT1      62
      MAT4=MU                                                             SPIT1      63
      CALL MULT (W2,B,W3,NX,NX,NU,                                        SPIT1      64
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                          SPIT1      65
      MAT2=MU                                                             SPIT1      66
      MAT4=MX                                                             SPIT1      67
      CALL MAKE (B,W3,NX,NU,                                              SPIT1      68
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                          SPIT1      69
   60 CONTINUE                                                            SPIT1      70
      DO 400 I5=1,TRESP                                                   SPIT1      71
      T=0.0                                                               SPIT1      72
```

```
          DO 1 I=1,NX                                                   SPIT1     73
          ROOTR(I)=0.0                                                  SPIT1     74
          ROOTI(I)=0.0                                                  SPIT1     75
          Z(I)=0.0                                                      SPIT1     76
          V(I)=0.0                                                      SPIT1     77
          W1(I,1)=0.0                                                   SPIT1     78
          U(I)=0.0                                                      SPIT1     79
        1 CONTINUE                                                      SPIT1     80
          NXX=NXC+1                                                     SPIT1     81
          NUU=NUC+1                                                     SPIT1     82
          IF(FORM.GT.0) WRITE(7)PST,TITLE,SYSTEM,MODEL,DIGITL,SCAPLT    SPIT1     83
          NYU=NY+NU                                                     SPIT1     84
          IF (MULTRT.EQ.1) NYU=NY+NUC                                   SPIT1     85
          IF(FORM.GT.0) WRITE(7)NYU,(OUTPT(I),I=1,NY),(INPT(J),J=1,NU)  SPIT1     86
          WRITE (3,13) (OUTP T(I),I=1,NY),(INP T(J),J=1,NU)             SPIT1     87
       13 FORMAT (//*     TIME     *(10(2X,A10))//)                     SPIT1     88
          IF (MULTRT.NE.0) GO TO 98                                     SPIT1     89
       20 CONTINUE                                                      SPIT1     90
          CALL INPUTV(DELT,T,U,                                         SPIT1     91
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                    SPIT1     92
          DO 30 I=1,NY                                                  SPIT1     93
          ROOTI(I)=0.0                                                  SPIT1     94
          DO 30 J=1,NX                                                  SPIT1     95
          ROOTI(I)=ROOTI(I)+H(I,J)*ROOTR(J)                             SPIT1     96
       30 CONTINUE                                                      SPIT1     97
          DO 31 I=1,NY                                                  SPIT1     98
          DO 31 J=1,NU                                                  SPIT1     99
          ROOTI(I)=ROOTI(I)+F(I,J)*U(J)                                 SPIT1    100
       31 CONTINUE                                                      SPIT1    101
          IF(FORM.EQ.2) GO TO 6                                         SPIT1    102
          WRITE (3,10) T,(ROOTI(I),I=1,NY),(U(J),J=1,NU)                SPIT1    103
          IF(FORM.EQ.0) GO TO 7                                         SPIT1    104
        6 WRITE(7)T,(ROOTI(I),I=1,NY),(U(J),J=1,NU)                     SPIT1    105
        7 CONTINUE                                                      SPIT1    106
       10 FORMAT (11E12.3)                                              SPIT1    107
          DO 40 I=1,NX                                                  SPIT1    108
          W1(I,1)=0.0                                                   SPIT1    109
          DO 40 J=1,NX                                                  SPIT1    110
          W1(I,1)=W1(I,1)+A(I,J)*ROOTR(J)                               SPIT1    111
       40 CONTINUE                                                      SPIT1    112
          DO 41 I=1,NX                                                  SPIT1    113
          DO 41 J=1,NU                                                  SPIT1    114
          W1(I,1)=W1(I,1)+B(I,J)*U(J)                                   SPIT1    115
       41 CONTINUE                                                      SPIT1    116
          DO 42 I=1,NX                                                  SPIT1    117
          ROOTR(I)=W1(I,1)                                              SPIT1    118
       42 CONTINUE                                                      SPIT1    119
          T=T+DELT                                                      SPIT1    120
          IF (T.LE.FINALT) GO TO 20                                     SPIT1    121
          GO TO 96                                                      SPIT1    122
       98 KCT=1                                                         SPIT1    123
          ICK=0                                                         SPIT1    124
      120 CONTINUE                                                      SPIT1    125
          IF (ICK.EQ.1) GO TO 70                                        SPIT1    126
          CALL INPUTV(DELT,T,U,                                         SPIT1    127
         1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                    SPIT1    128
          DO 75 I=1,NXC                                                 SPIT1    129
```

```
            V(I)=ROOTR(I)                                          SPIT1     130
         75 CONTINUE                                               SPIT1     131
            DO 71 I=1,NU                                           SPIT1     132
            XX=0.                                                  SPIT1     133
            DO 72 J=1,NU                                           SPIT1     134
            XX=XX+D(I,J)*U(J)                                      SPIT1     135
         72 CONTINUE                                               SPIT1     136
            Z(I)=XX                                                SPIT1     137
         71 CONTINUE                                               SPIT1     138
            DO 73 I=1,NU                                           SPIT1     139
            XX=0.                                                  SPIT1     140
            DO 74 J=1,NX                                           SPIT1     141
            XX=XX+K1(I,J)*ROOTR(J)                                 SPIT1     142
         74 CONTINUE                                               SPIT1     143
            Z(I)=Z(I)+XX                                           SPIT1     144
         73 CONTINUE                                               SPIT1     145
            ICK=1                                                  SPIT1     146
         70 CONTINUE                                               SPIT1     147
            DO 80 I=1,NY                                           SPIT1     148
            XX=0.                                                  SPIT1     149
            DO 81 J=1,NXC                                          SPIT1     150
            XX=XX+H(I,J)*ROOTR(J)                                  SPIT1     151
         81 CONTINUE                                               SPIT1     152
            ROOTI(I)=XX                                            SPIT1     153
         80 CONTINUE                                               SPIT1     154
            DO 82 I=1,NY                                           SPIT1     155
            XX=0.                                                  SPIT1     156
            DO 83 J=1,NUC                                          SPIT1     157
            XX=XX+F(I,J)*Z(J)                                      SPIT1     158
         83 CONTINUE                                               SPIT1     159
            ROOTI(I)=ROOTI(I)+XX                                   SPIT1     160
         82 CONTINUE                                               SPIT1     161
            IF (FORM.EQ.2) GO TO 16                                SPIT1     162
            WRITE (3,10) T,(ROOTI(I),I=1,NY),(Z(J),J=1,NUC)        SPIT1     163
            IF (FORM.EQ.0) GO TO 17                                SPIT1     164
         16 WRITE (7) T, (ROOTI(I),I=1,NY), (Z(J),J=1,NUC)         SPIT1     165
         17 CONTINUE                                               SPIT1     166
            DO 85 I=1,NXC                                          SPIT1     167
            XX=0.                                                  SPIT1     168
            DO 86 J=1,NXC                                          SPIT1     169
            XX=XX+A(I,J)*ROOTR(J)                                  SPIT1     170
         86 CONTINUE                                               SPIT1     171
            W1(I,1)=XX                                             SPIT1     172
         85 CONTINUE                                               SPIT1     173
            DO 87 I=1,NXC                                          SPIT1     174
            XX=0.                                                  SPIT1     175
            DO 88 J=1,NUC                                          SPIT1     176
            XX=XX+B(I,J)*Z(J)                                      SPIT1     177
         88 CONTINUE                                               SPIT1     178
            W1(I,1)=W1(I,1)+XX                                     SPIT1     179
         87 CONTINUE                                               SPIT1     180
            DO 89 I=1,NXC                                          SPIT1     181
            ROOTR(I)=W1(I,1)                                       SPIT1     182
         89 CONTINUE                                               SPIT1     183
            T=T+DELT                                               SPIT1     184
            KCT=KCT+1                                              SPIT1     185
            IF (T.GT.FINALT) GO TO 96                              SPIT1     186
```

```
      IF (KCT.LE.MULTRT) GO TO 120                                      SPIT1     187
      KCT=1                                                             SPIT1     188
      ICK=0                                                             SPIT1     189
      IF (NXX.GT.NX) GO TO 120                                          SPIT1     190
      DO 90 I=NXX,NX                                                    SPIT1     191
      XX=0.                                                             SPIT1     192
      DO 91 J=NXX,NX                                                    SPIT1     193
      XX=XX+A(I,J)*ROOTR(J)                                             SPIT1     194
   91 CONTINUE                                                          SPIT1     195
      W1(I,1)=XX                                                        SPIT1     196
   90 CONTINUE                                                          SPIT1     197
      DO 92 I=NXX,NX                                                    SPIT1     198
      XX=0.                                                             SPIT1     199
      DO 93 J=1,NXC                                                     SPIT1     200
      XX=XX+A(I,J)*V(J)                                                 SPIT1     201
   93 CONTINUE                                                          SPIT1     202
      W1(I,1)=W1(I,1)+XX                                                SPIT1     203
   92 CONTINUE                                                          SPIT1     204
      IF (NUU.GT.NU) GO TO 130                                          SPIT1     205
      DO 94 I=NXX,NX                                                    SPIT1     206
      XX=0.                                                             SPIT1     207
      DO 95 J=NUU,NU                                                    SPIT1     208
      XX=XX+B(I,J)*Z(J)                                                 SPIT1     209
   95 CONTINUE                                                          SPIT1     210
      ROOTR(I)=W1(I,1)+XX                                               SPIT1     211
   94 CONTINUE                                                          SPIT1     212
      GO TO 120                                                         SPIT1     213
  130 DO 131 I=NXX,NX                                                   SPIT1     214
      ROOTR(I)=W1(I,1)                                                  SPIT1     215
  131 CONTINUE                                                          SPIT1     216
      GO TO 120                                                         SPIT1     217
   96 TT=-1.0                                                           SPIT1     218
      IF(FORM.GT.0) WRITE(7)TT,TT,TT,TT,TT,TT,TT,TT,TT,TT,TT,TT,TT,     SPIT1     219
     XTT,TT,TT,TT,TT,TT,TT,TT,TT,TT                                     SPIT1     220
  400 CONTINUE                                                          SPIT1     221
      END                                                               SPIT1     222
```

```
      SUBROUTINE SWZ (DELT,IGO,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
C     UPDATED 1-9-73 TO INCLUDE DIFFERENTIATOR BLOCK
C     UPDATED 2-13-73 TO CORRECT GAIN COMPUTATION WITH MOD=1
C     UPDATED 4-25-73
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CHAT,NK2,IFLAG,
     2IGOO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CHAT,READ3, FRPS,TRESP
      INTEGER DIGITL,SCAPLT,ZOH
      COMMON/BLKDAT/NUMER,DENOM,GAIN,GRAPH,BLOCK,STATE,YTOV,ZTOU,YZTOK,
     1ITHINY,ITHINU,NBLOCK,NYTOV,NZTOU,NXYU,NYZTOK,NXT,NYT,NUT,NY1,NU1
      REAL NUMER
      INTEGER GRAPH,BLOCK,STATE,YTOV,ZTOU,YZTOK
      DIMENSION GRAPH(20,5),BLOCK(20,3),NUMER(20,5),DENOM(20,5),
     XGAIN(20),STATE(20,4),ITHINY(30),ITHINU(20),YTOV(20,2),
     X  ZTOU(20,2),NXYU(8),YZTOK(20,2)
      DIMENSION NIP(20,7),PARAM(20,5),CARAM(20,5)
   11 FORMAT (I2,I3,5I5,5F10.4)
      DO 300 I=1,NBLOCK
      DO 300 K=1,5
      DENOM(I,K)=0.
  300 NUMER(I,K)=0.
      IF(IGO.EQ.1.AND.KOUNT.GT.1) GO TO 310
C     IF (IGO.EQ.1) GO TO 310
      WRITE (3,500)
   14 FORMAT (7I5,7F10.4)
  500 FORMAT (  //* BLOCK DIAGRAM INPUT PARAMETERS*//* NO. TYPE*,7X,*CON
     1NEC*,5X,*MOD*,19X,*PARAM*/)
      DO 10 J=1,NBLOCK
      READ (1,11) (NIP(J,L),L=1,7),(CARAM(J,K),K=1,5)
      WRITE (3,14) (NIP(J,L),L=1,7),(CARAM(J,K),K=1,5)
   10 CONTINUE
  310 DO 311 I=1,NBLOCK
      DO 311 K=1,5
  311 PARAM(I,K)=CARAM(I,K)
      DO 12 J=1,NBLOCK
      GAIN(NIP(J,1))=PARAM(J,1)
      GRAPH(NIP(J,1),1)=NIP(J,1)
      DO 13 K=3,6
   13 GRAPH (NIP(J,1),K-1)=NIP(J,K)
   12 BLOCK(NIP(J,1),1)=NIP(J,1)
      DO 100 I=1,NBLOCK
      L=NIP(I,1)
      II=NIP(I,2)
      GO TO (101,102,103,104,105,106,107,108,108,108,108),II
  101 BLOCK(        L,2)=1
      BLOCK(        L,3)=1
      NUMER(L,1)=1.
      DENOM(        L,1)=1.
      GO TO 100
  102 IF (NIP(I,7).NE.0) WRITE (3,15)
   15 FORMAT (/* THIS DIGITAL DIFFERENTIATOR IS NO GOOD*/)
      BLOCK(L,2)=2
      NUMER(L,1)=0.
      NUMER(L,2)=1.
      DENOM(L,1)=1.
```

```
      GO TO 100                                                         SWZ      59
  103 IF (NIP(I,7).EQ.1) GO TO 120                                      SWZ      60
      IF (NIP(I,7).EQ.2) GO TO 121                                      SWZ      61
      BLOCK(L,2)=1                                                      SWZ      62
      BLOCK(L,3)=2                                                      SWZ      63
      NUMER(L,1)=1.                                                     SWZ      64
      DENOM(L,1)=0.                                                     SWZ      65
      DENOM(L,2)=1.                                                     SWZ      66
      GO TO 100                                                         SWZ      67
  120 GAIN(L)=GAIN(L)*.5*DELT                                           SWZ      68
  121 BLOCK(L,2)=2                                                      SWZ      69
      BLOCK(L,3)=2                                                      SWZ      70
      NUMER(L,1)=1.                                                     SWZ      71
      NUMER(L,2)=1.                                                     SWZ      72
      DENOM(L,1)=-1.                                                    SWZ      73
      DENOM(L,2)=1.                                                     SWZ      74
      GO TO 100                                                         SWZ      75
  104 IF (NIP(I,7).EQ.1) GO TO 141                                      SWZ      76
      IF (NIP(I,7).EQ.2) GO TO 142                                      SWZ      77
      BLOCK(L,2)=1                                                      SWZ      78
      BLOCK(L,3)=2                                                      SWZ      79
      NUMER(L,1)=1.                                                     SWZ      80
      DENOM(L,1)=1.                                                     SWZ      81
      DENOM(L,2)=1./PARAM(I,2)                                          SWZ      82
      GO TO 100                                                         SWZ      83
  141 PARAM(I,2)=TANH(.5*PARAM(I,2)*DELT)                               SWZ      84
  142 XNU=PARAM(I,2)                                                    SWZ      85
      BLOCK(L,2)=2                                                      SWZ      86
      BLOCK(L,3)=2                                                      SWZ      87
      NUMER(L,1)=XNU/(1.+XNU)                                           SWZ      88
      NUMER(L,2)=NUMER(L,1)                                             SWZ      89
      DENOM(L,2)=1.                                                     SWZ      90
      DENOM(L,1)=(XNU-1.)/(XNU+1.)                                      SWZ      91
      GO TO 100                                                         SWZ      92
  105 BLOCK(L,2)=2                                                      SWZ      93
      BLOCK(L,3)=2                                                      SWZ      94
      IF (NIP(I,7).EQ.1) GO TO 151                                      SWZ      95
      IF (NIP(I,7).EQ.2) GO TO 152                                      SWZ      96
      NUMER(L,1)=1.                                                     SWZ      97
      NUMER(L,2)=1./PARAM(I,3)                                          SWZ      98
      DENOM(L,1)=1.                                                     SWZ      99
      DENOM(L,2)=1./PARAM(I,2)                                          SWZ     100
      GO TO 100                                                         SWZ     101
  151 PARAM(I,2)=TANH(.5*PARAM(I,2)*DELT)                               SWZ     102
      PARAM(I,3)=TANH(.5*PARAM(I,3)*DELT)                               SWZ     103
  152 XNU=PARAM(I,2)                                                    SWZ     104
      XN=PARAM(I,3)                                                     SWZ     105
      DENOM(L,1)=(XNU-1.)/(XNU+1.)                                      SWZ     106
      DENOM(L,2)=1.                                                     SWZ     107
      NUMER(L,1)=(XNU*(XN-1.))/(XN*(XNU+1.))                            SWZ     108
      NUMER(L,2)=(XNU*(XN+1.))/(XN*(XNU+1.))                            SWZ     109
      GO TO 100                                                         SWZ     110
  106 BLOCK(L,2)=2                                                      SWZ     111
      BLOCK(L,3)=2                                                      SWZ     112
      IF (NIP(I,7).EQ.1) GO TO 161                                      SWZ     113
      IF (NIP(I,7).EQ.2) GO TO 162                                      SWZ     114
      NUMER(L,1)=0.                                                     SWZ     115
```

```
      NUMER(L,2)=1.                                                     SWZ       116
      DENOM(L,1)=PARAM(I,2)                                             SWZ       117
      DENOM(L,2)=1.                                                     SWZ       118
      GO TO 100                                                         SWZ       119
  161 PARAM(I,2)=TANH(.5*PARAM(I,2)*DELT)                               SWZ       120
  162 XN=PARAM(I,2)                                                     SWZ       121
      NUMER(L,1)=-1./(1.+XN)                                            SWZ       122
      NUMER(L,2)=-NUMER(L,1)                                            SWZ       123
      DENOM(L,1)=(XN-1.)/(XN+1.)                                        SWZ       124
      DENOM(L,2)=1.                                                     SWZ       125
      GO TO 100                                                         SWZ       126
  107 IF (NIP(I,7).EQ.1) GO TO 171                                      SWZ       127
      IF (NIP(I,7).EQ.2) GO TO 172                                      SWZ       128
      BLOCK(L,2)=1                                                      SWZ       129
      BLOCK(L,3)=3                                                      SWZ       130
      NUMER(L,1)=1.                                                     SWZ       131
      DENOM(L,3)=1./(PARAM(I,2)*PARAM(I,3))                             SWZ       132
      DENOM(L,2)=1./PARAM(I,2)+1./ PARAM(I,3)                           SWZ       133
      DENOM(L,1)=1.                                                     SWZ       134
      GO TO 100                                                         SWZ       135
  171 PARAM(I,2)=TANH(.5*PARAM(I,2)*DELT)                               SWZ       136
      PARAM(I,3)=TANH(.5*PARAM(I,3)*DELT)                               SWZ       137
  172 XNU=PARAM(I,2)                                                    SWZ       138
      XN=PARAM(I,3)                                                     SWZ       139
      GAIN(L)=GAIN(L)*XN*XNU/((XN+1.)*(XNU+1.))                         SWZ       140
      BLOCK(L,2)=3                                                      SWZ       141
      BLOCK(L,3)=3                                                      SWZ       142
      NUMER(L,1)=1.                                                     SWZ       143
      NUMER(L,2)=2.                                                     SWZ       144
      NUMER(L,3)=1.                                                     SWZ       145
      DENOM(L,1)=((XN-1.)*(XNU-1.))/((XN+1.)*(XNU+1.))                  SWZ       146
      DENOM(L,2)=(XN-1.)/(XN+1.)+(XNU-1.)/(XNU+1.)                      SWZ       147
      DENOM(L,3)=1.                                                     SWZ       148
      GO TO 100                                                         SWZ       149
  108 IF (NIP(I,7).EQ.1) GO TO 181                                      SWZ       150
      IF (NIP(I,7).EQ.2) GO TO 182                                      SWZ       151
      BLOCK(L,2)=1                                                      SWZ       152
      BLOCK(L,3)=3                                                      SWZ       153
      NUMER(L,1)=1.                                                     SWZ       154
      DENOM(L,3)=1./(PARAM(I,2)**2)                                     SWZ       155
      DENOM(L,2)=2.*PARAM(I,3)/PARAM(I,2)                               SWZ       156
      DENOM(L,1)=1.                                                     SWZ       157
      GO TO 109                                                         SWZ       158
  181 ALFA=-PARAM(I,2)*       PARAM(I,3)                                SWZ       159
      BETA=SQRT(PARAM(I,2)**2-ALFA**2)                                  SWZ       160
      A=ALFA*DELT                                                       SWZ       161
      B=BETA*DELT                                                       SWZ       162
      D=(EXP(A)*COS(B)+1.)**2+SIN(B)**2*EXP(2.*A)                       SWZ       163
      U=(EXP(2.*A)                     -1.)/D                           SWZ       164
      V=2.*SIN(B)*EXP(A)/D                                              SWZ       165
      W=SQRT(U**2+V**2)                                                 SWZ       166
      OMEGA=PARAM(L,2)                                                  SWZ       167
      ZETA=PARAM(L,3)                                                   SWZ       168
      PARAM(I,2)=W                                                      SWZ       169
      PARAM(I,3)=-U/W                                                   SWZ       170
  182 BLOCK(L,2)=3                                                      SWZ       171
      BLOCK(L,3)=3                                                      SWZ       172
```

```
          XN=PARAM(I,2)                                                     SWZ
          DENOM(L,1)=(1.-2.*PARAM(I,3)*XN+XN**2)/(1.+2.*PARAM(I,3)*XN+XN**2) SWZ
          DENOM(L,2)=2.*(XN**2-1.)/(1.+2.*PARAM(I,3)*XN+XN**2)              SWZ
          DENOM(L,3)=1.                                                     SWZ
          GAIN(L)=GAIN(L)*XN**2/(1.+2.*PARAM(I,3)*XN+XN**2)                 SWZ
      109 IF (NIP(I,7).NE.0) GO TO 110                                      SWZ
          II=NIP(I,2)-7                                                     SWZ
          GO TO (100,221,222,223),II                                        SWZ
      221 BLOCK(L,2)=3                                                      SWZ
          NUMER(L,3)=1./PARAM(I,4)**2                                       SWZ
          NUMER(L,2)=2.*PARAM(I,5)/PARAM(I,4)                               SWZ
          NUMER(L,1)=1.                                                     SWZ
          GO TO 100                                                         SWZ
      222 BLOCK(L,2)=2                                                      SWZ
          NUMER(L,2)=1./PARAM(I,4)                                          SWZ
          NUMER(L,1)=1.                                                     SWZ
          GO TO 100                                                         SWZ
      223 BLOCK(L,2)=2                                                      SWZ
          NUMER(L,1)=0.                                                     SWZ
          NUMER(L,2)=1.                                                     SWZ
          GO TO 100                                                         SWZ
      110 II=NIP(I,2)-7                                                     SWZ
          GO TO (230,232,231,230),II                                        SWZ
      231 IF (NIP(I,7).EQ.2) GO TO 230                                      SWZ
          PARAM(I,4)=TANH(.5*PARAM(I,4)*DELT)                               SWZ
          GO TO 230                                                         SWZ
      232 IF (NIP(I,7).EQ.2) GO TO 230                                      SWZ
          ALFA=-PARAM(I,4)*       PARAM(I,5)                                SWZ
          BETA=SQRT(PARAM(I,4)**2-ALFA**2)                                  SWZ
          A=ALFA*DELT                                                       SWZ
          B=BETA*DELT                                                       SWZ
          D=(EXP(A)*COS(B)+1.)**2+SIN(B)**2*EXP(2.*A)                       SWZ
          U=(EXP(2.*A)                        -1.)/D                        SWZ
          V=2.*SIN(B)*EXP(A)/D                                              SWZ
          W=SQRT(U**2+V**2)                                                 SWZ
          PARAM(I,4)=W                                                      SWZ
          PARAM(I,5)=-U/W                                                   SWZ
      230 GO TO (240,241,242,243),II                                        SWZ
      240 NUMER(L,1)=1.                                                     SWZ
          NUMER(L,2)=2.                                                     SWZ
          NUMER(L,3)=1.                                                     SWZ
          GO TO 100                                                         SWZ
      241 YN=PARAM(I,4)                                                     SWZ
          NUMER(L,2)=2.*(YN**2-1.)/(1.+2.*PARAM(I,5)*YN+YN**2)              SWZ
          NUMER(L,1)=(1.-2.*PARAM(I,5)*YN+YN**2)/(1.+2.*PARAM(I,5)*YN+YN**2) SWZ
          NUMER(L,3)=1.                                                     SWZ
          GAIN(L)=GAIN(L)*(1.+2.*PARAM(I,5)*YN+YN**2)/YN**2                 SWZ
          GO TO 100                                                         SWZ
      242 YN=PARAM(I,4)                                                     SWZ
          NUMER(L,1)=(YN-1.)/(YN+1.)                                        SWZ
          NUMER(L,2)=2.*YN/(YN+1.)                                          SWZ
          NUMER(L,3)=1.                                                     SWZ
          GAIN(L)=GAIN(L)*(1.+YN)/YN                                        SWZ
          GO TO 100                                                         SWZ
      243 NUMER(L,1)=-1.                                                    SWZ
          NUMER(L,2)=0.                                                     SWZ
          NUMER(L,3)=1.                                                     SWZ
```

```
                 IF (NIP(I,7).EQ.2) GO TO 100                               SWZ      230
                 GAIN(L)=GAIN(L)*(1.+2.*PARAM(I,3)*XN+XN**2)/XN**2          SWZ      231
                 GAIN(L)=GAIN(L)*(1.+DENOM(L,1)+DENOM(L,2))                 SWZ      232
                1/(2.*DELT)                                                 SWZ      233
             100 CONTINUE                                                   SWZ      234
                 RETURN                                                     SWZ      235
                 END                                                        SWZ      236
```

```
          FUNCTION TANG (A)                                                 TANG       2
          COMPLEX A,C                                                       TANG       3
          EQUIVALENCE (C,B(1))                                              TANG       4
          DIMENSION B(2)                                                    TANG       5
          C=A                                                               TANG       6
          PHI=ATAN2 (B(2),B(1))*57.3                                        TANG       7
          IF (B(2).GT.0.0.AND.B(1).LT.0.0) GO TO 10                         TANG       8
          TANG=PHI                                                          TANG       9
          GO TO 20                                                          TANG      10
       10 TANG=PHI-360.                                                     TANG      11
       20 RETURN                                                            TANG      12
          END                                                               TANG      13
```

```
      SUBROUTINE WDISC    (A,B,C,H,G,F,K1,K2,K3,K4,D,                    WDISC       2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC       3
C                                                                        WDISC       4
C        THIS SUBROUTINE WRITES ALL INPUT MATRICES ON DISCS ACCORDING    WDISC       5
C        TO THE PARAMETERS,SYSTEM AND OUTPUT, USING SUBROUTINE WDISC1    WDISC       6
C                                                                        WDISC       7
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,      WDISC       8
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,          WDISC       9
     2IGO,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KOUNT                   WDISC      10
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP  WDISC      11
      INTEGER DIGITL,SCAPLT,ZOH                                          WDISC      12
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),   WDISC      13
     1K1(MU,MX),K2(MU,MX),K3(MU,MX),K4(MU,MX),D(MU,MU)                   WDISC      14
      REAL K1,K2,K3,K4                                                   WDISC      15
      COMMON/SUBWRIT/ ISUBNAM                                            WDISC      16
      IF(ISUBNAM.GE.2) WRITE(3,990)                                      WDISC      17
  990 FORMAT(1X,*WDISC*)                                                 WDISC      18
      MAT1=MY                                                            WDISC      19
      MAT2=MX                                                            WDISC      20
      CALL WDISC1 (A,NX,NX,                                              WDISC      21
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC      22
      MAT2=MU                                                            WDISC      23
      CALL WDISC1 (B,NX,NU,                                              WDISC      24
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC      25
      MAT2=MX                                                            WDISC      26
      CALL WDISC1 (C,NX,NX,                                              WDISC      27
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC      28
      MAT1=MU                                                            WDISC      29
      CALL WDISC1 (K1,NU,NX,                                             WDISC      30
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC      31
      CALL WDISC1 (K2,NU,NX,                                             WDISC      32
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC      33
      CALL WDISC1 (K3,NU,NX,                                             WDISC      34
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC      35
      CALL WDISC1 (K4,NU,NX,                                             WDISC      36
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC      37
      MAT1=MY                                                            WDISC      38
      CALL WDISC1 (H,NY,NX,                                              WDISC      39
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC      40
      CALL WDISC1 (G,NY,NX,                                              WDISC      41
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC      42
      MAT2=MU                                                            WDISC      43
      CALL WDISC1 (F,NY,NU,                                              WDISC      44
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC      45
      MAT1=MU                                                            WDISC      46
      CALL WDISC1 (D,NU,NU,                                              WDISC      47
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC      48
      REWIND 8                                                           WDISC      49
      RETURN                                                             WDISC      50
      END                                                                WDISC      51
```

```
SUBROUTINE WDISC1      73/74    OPT=1                          FTN 4.2+75060       01/09/75  14.19.11.

                  SUBROUTINE WDISC1 (A,N,M,                                          WDISC1      2
                 1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                         WDISC1      3
                  DIMENSION A(MAT1,MAT2)                                             WDISC1      4
                  COMMON/SUBWRIT/ ISUBNAM                                            WDISC1      5
                  IF(ISUBNAM.GE.2) WRITE(3,990)                                      WDISC1      6
              990 FORMAT(1X,*WDISC1*)                                                WDISC1      7
                  DO 10 I=1,N                                                        WDISC1      8
                  WRITE (8) (A(I,J),J=1,M)                                           WDISC1      9
               10 CONTINUE                                                           WDISC1     10
                  RETURN                                                             WDISC1     11
                  END                                                                WDISC1     12
```

```
      SUBROUTINE ZOT (A,B,C,H,G,F,K1,K2,K3,K4,D,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      COMMON/COND/READ,SYSTEM,OUTPUT,NX,NY,NU,NXC,NUC,N1,N2,DIGITL,
     1CONTUR,NUMERS,FRPS,TRESP,MODEL,NSCALE,SAV,CMAT,NK2,IFLAG,
     2IGC,FORM,IPT,READ3,MIXED,MULTRT,SCAPLT,ZOH,KCUNT
      INTEGER DIGITL,SCAPLT,ZOH
      INTEGER READ,SYSTEM,OUTPUT,FORM,CONTUR,SAV,CMAT,READ3, FRPS,TRESP

      REAL K1,K2,K3,K4
      DIMENSION A(MX,MX),B(MX,MU),C(MX,MX),H(MY,MX),G(MY,MX),F(MY,MU),
     1K1(MU,MX),K2(MU,MX),K3(MU,MX),K4(MU,MX),D(MU,MU)
C
C       THIS SUBROUTINE INITIALIZES THE SYSTEM MATRICES TO ZERO
C       USING ZOT1
C
C
      COMMON/SUBWRIT/ ISUBNAM
      IF(ISUBNAM.GE.2) WRITE(3,990)
  990 FORMAT(1X,*ZOT*)
      MAT1=MX
      MAT2=MX
      CALL ZOT1 (A,MX,MX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL ZOT1 (C,MX,MX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MY
      CALL ZOT1 (H,MY,MX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL ZOT1 (G,MY,MX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MU
      CALL ZOT1 (F,MY,MU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MX
      CALL ZOT1 (B,MX,MU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT1=MU
      MAT2=MX
      CALL ZOT1 (K1,MU,MX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL ZOT1 (K2,MU,MX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL ZOT1 (K3,MU,MX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      CALL ZOT1 (K4,MU,MX,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      MAT2=MU
      CALL ZOT1 (D,MU,MU,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
      RETURN
      END
```

```
      SUBROUTINE ZOT1 (A,N,M,                                    ZOT1         2
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)                 ZOT1         3
      DIMENSION A(MAT1,MAT2)                                     ZOT1         4
      COMMON/SUBWRIT/ ISUBNAM                                    ZOT1         5
      IF(ISUBNAM.GE.2) WRITE(3,990)                              ZOT1         6
  990 FORMAT(1X,*ZOT1*)                                          ZOT1         7
      DO 10 I=1,N                                                ZOT1         8
      DO 10 J=1,M                                                ZOT1         9
      A(I,J)=0.0                                                 ZOT1        10
   10 CONTINUE                                                   ZOT1        11
      RETURN                                                     ZOT1        12
      END                                                        ZOT1        13
```

```
      SUBROUTINE ZTOW (N,NN,DE,I,JJ,ROOTR,ROOTI,ROTR,ROTI,L1,DEE,
     1MX,MY,MU,MS,MAT1,MAT2,MAT3,MAT4,MAT5,MAT6)
C
C       THIS SUBROUTINE CONVERTS Z-PLANE TRANSFER FUNCTIONS TO
C        W-PLANE TRANSFER FUNCTIONS
C
      DIMENSION ROOTR(MX),ROOTI(MX),ROTR(MX),ROTI(MX)
      COMMON/SUBWRIT/ ISUBNAM
      IF(ISUBNAM.GE.2) WRITE(3,990)
  990 FORMAT(1X,*ZTOW*)
      L2=0
      K=1
      IF (I.GT.1.OR.JJ.GT.1) GO TO 1
      DEE=1.0
      L1=0
    2 IF (ABS((ROOTR(K)-1.)    +ROOTI(K)    ).GT.10.**(-6)) GO TO 10
      L1=L1+1
      DEE=DEE/2.
      ROOTR(K)=0.0
      ROOTI(K)=0.0
      K=K+1
      GO TO 11
   10 IF (ROOTI(K).NE.0.0) GO TO 12
      DEE=DEE/(1.+ROOTR(K))
      ROOTR(K)=-(1.-ROOTR(K))/(1.+ROOTR(K))
      K=K+1
      GO TO 11
   12 A=ROOTR(K)**2
      B=ROOTI(K)**2
      G=(1.+ROOTR(K))**2
      DEE=DEE/(G+B)
      ROOTR(K)=(A+B-1.)/(G+B)
      ROOTR(K+1)=ROOTR(K)
      ROOTI(K)=(2.*ROOTI(K))/(G+B)
      ROOTI(K+1)=-ROOTI(K)
      K=K+2
   11 IF (K.LE.NN) GO TO 2
      RETURN
    1 DE=DE*DEE
      NSAV=0
   13 IF (ABS((ROTR(K)+1.)+ROTI(K)).LT.10.**(-6)) GO TO 18
      IF (ABS((ROTR(K)-1.)+ROTI(K)).GT.10.**(-6)) GO TO 14
      DE=DE*2.
      L2=L2+1
      ROTR(K)=0.
      ROTI(K)=0.
      K=K+1
      GO TO 15
   18 DE=DE*2.
      ROTR(K)=-1000.
      ROTI(K)=0.
      K=K+1
      NSAV=NSAV+1
      GO TO 15
   14 IF (ROTI(K).NE.0.) GO TO 16
      DE=DE*(1.+ROTR(K))
      ROTR(K)=-(1.-ROTR(K))/(1.+ROTR(K))
```

```
      K=K+1
      GO TO 15
   13 A=ROTR(K)**2
      B=ROTI(K)**2
      G=(1.+ROTR(K))**2
      DE=DE*(G+B)
      ROTR(K)=(A+B-1.)/(G+B)
      ROTR(K+1)=ROTR(K)
      ROTI(K)=(2.*ROTI(K))/(G+B)
      ROTI(K+1)=-ROTI(K)
      K=K+2
   15 IF (K.LE.N) GO TO 13
      LMN=NN-N
      DE=DE*(-1.)**LMN
      II=1
      DO 20 I=1,N
      IF (ROTR(II).NE.-1000.) GO TO 22
      K=II
      DO 21 L=K,N
      ROTR(L)=ROTR(L+1)
      ROTI(L)=ROTI(L+1)
   21 CONTINUE
      GO TO 20
   22 II=II+1
   20 CONTINUE
      N=N-NSAV
      IF (N.EQ.NN)RETURN
      N1=N+1
      N=NN-NSAV
      IF (N.LE.N1)RETURN
      DO 17 I=N1,N
      ROTR(I)=1.0
      ROTI(I)=0.
   17 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE COPO                                                   COPO       2
C                                                                       COPO       3
C---------------------------------------------------------------------- COPO       4
C**********     SUBROUTINE  C O P O  PLOTS DATA GENERATED BY THE  C O N COPO       5
C               PROGRAM. THE FOLLOWING TYPES OF PLOTS ARE AVAILABLE.    COPO       6
C                      1. TIME HISTORY                                  COPO       7
C                      2. TIME HISTORY WITH CSTAR ENVELOPE              COPO       8
C                      3. FREQUENCY RESPONSE                            COPO       9
C                      4. POWER SPECTRAL DENSITY                        COPO      10
C                      5. ROOT LOCUS                                    COPO      11
C                      6. Z-PLANE ROOT LOCUS                            COPO      12
C                      7. ROOT CONTOUR                                  COPO      13
C                                                                       COPO      14
C               THE MAXIMUM NUMBER OF POINTS FOR ANY ONE PLOT IS 998. TH COPO     15
C               MAXIMUM NUMBER OF TIME HISTORY PLOTS IS 25. AN AUTOMATIC COPO     16
C               PLOT REQUEST IS GENERATED AT THE END OF TH. JOB         COPO      17
C---------------------------------------------------------------------- COPO      18
C                                                                       COPO      19
      EXTERNAL CSTAR                                                    COPO      20
      DOUBLE PRECISION SNAME                                            COPO      21
      COMMON/CSTARC/CTML,CTMU,CENVL,CENVU/CSTARP/PTML,PTMU,PENVL,PENVU   COPO      22
C                                                                       COPO      23
C    ARRAYS IN FOLLOWING DIMENSION STATEMENT DEFINE THE POWER AND CURVED COPO     24
C    APPROACH CSTAR OPTION ENVELOPES FOR TIME HISTORY PLOTS.            COPO      25
C                                                                       COPO      26
      DIMENSION CTML(20),CTMU(20),CENVL(20),CENVU(20),                  COPO      27
     X          PTML(12),PTMU(23),PENVL(12),PENVU(23)                   COPO      28
      REAL UNCLE(1024),VP(1000) ,TM,FR,VP2(1000),                       COPO      29
     XBIG(10),YLOG(4,9),LOG(9),NOTA,PTITLE(8),COMMENT(12),              COPO      30
     X     PS,RL,TIME(1000),V(25),S10(1000),S20(1000)                   COPO      31
      INTEGER ORDER(10),LOW,HIGH,LOWY,HIGHY                             COPO      32
      INTEGER YEP                                                       COPO      33
      INTEGER TWO,FOUR,SYSTEM                                           COPO      34
      EQUIVALENCE (FREQ,IFRE)                                           COPO      35
      INTEGER REMUS,TITLE(8),LAB(4),CNE,NAME(25),ENV,CSCRUV,CSPOA        COPO      36
      DATA ORDER/1,0,2,5,10,14,12,11,3,7/,                              COPO      37
     XCSCRUV/10HCSTARC    /,                                            COPO      38
     XCSPOA/10HCSTARP    /                                              COPO      39
      DATA TWO/2/,FOUR/4/,REMUS/1024/,                                  COPO      40
     XRC/10HROCO      /,                                                COPO      41
     XFR/10HFREQ      /,                                                COPO      42
     XPS/10HSPEC      /,                                                COPO      43
     XTM/10HTIME      /,                                                COPO      44
     XRL/10HPOLO      /,                                                COPO      45
     XNCPLOTS/0/,NOTA/10HNOT APP   /,                                   COPO      46
     XPTITLE/"         *","***** PLEA","SE READJUS","T PEN EVER",       COPO      47
     X"Y FIVE TO ","TEN PLOT P","AGES *****","*         "/,             COPO      48
     XCOMMENT/"  ****** "," P L E A S"," E  U S E ","  ******  ",       COPO      49
     X"  * P E R"," F O R A T"," E D  P A ","P E R *   ",               COPO      50
     X"         ","         ","         ","         "/,                 COPO      51
     X     BIG/2*.18,.25,.20,.18,.20,.18,.25,.18,.20/,                  COPO      52
     X            YLOG/.1139,.1761,.2304,.2788,.3617,.3979,.4314,       COPO      53
     X                     .4624,.5185,.5441,.5682,.5911,.6335,.6532,   COPO      54
     X                     .6721,.6902,.7243,.7404,0.,0.,.7993,.8129,   COPO      55
     X                     0.,0.,.8633,.8751,0.,0.,.9191,.9294,0.,0.,   COPO      56
     X                     .9685,.9777,0.,0./,                          COPO      57
     X     LOG/.3010,.4771,.6021,.6990,.7782,.8451,.9031,.9542,1./,     COPO      58
```

```
     XLABEL/10H MAG-DB   /,LABEL1/10HPHI-DEG   /,LABEL2/10H  DUE TO  /   COPO      59
C                                                                         COPO      60
C    SET UP PLOT FACTORS AND READ DATA FOR AUTO PLOT REQUEST              COPO      61
C                                                                         COPO      62
      HALF=2./2.54                                                        COPO      63
      CM=10./25.4                                                         COPO      64
      CALL PLOTS (UNCLE,REMUS,6)                                          COPO      65
      CALL FACTOR (1.)                                                    COPO      66
C     READ(1,504)SNAME,SUBTASK                                            COPO      67
      READ(1,504) IVSN,SNAME,SUBTASK                                      COPO      68
C 504 FORMAT (15X,2A10,5X,I4)                                             COPO      69
  504 FORMAT (I4,11X,2A10,5X,I4)                                          COPO      70
      WRITE (3,806)                                                       COPO      71
  806 FORMAT ("1",10X," PLOTTING HAS BEGUN")                              COPO      72
C                                                                         COPO      73
C    DATA SET 7 CONTAINS PLOT DATA WRITTEN BY  C O N T R O L  PROGRAM     COPO      74
C    FOR EACH CASE RUN IN THE CONTROL PROGRAM THE FOLLOWING IS WRITTEN O COPO      75
C                  RECORD ONE  NPLOT=NO. OF PLOTS                         COPO      76
C                  RECORD TWO  TYPE=TYPE OF PLOT (MUST BE TIME,FREQ,SPEC,ROL COPO      77
C                              TITLE=80 CHARACTER PLOT TITLE              COPO      78
C                              SYSTEM=3 IMPLIES ROOT LOCUS                COPO      79
C                              MODEL=NOT USED                             COPO      80
C                              IDIG=1 INDICATES Z-PLANE ROOT LOCUS        COPO      81
C                              SCAPLT=NOT USED                            COPO      82
C                                                                         COPO      83
      YEP=0                                                               COPO      84
      KLP=0                                                               COPO      85
      NCKP=0                                                              COPO      86
      NOKP=0                                                              COPO      87
      NNKP=0                                                              COPO      88
 8061 CONTINUE                                                            COPO      89
      NOKP=NOKP+1                                                         COPO      90
      IF ((YEP.NE.1).AND.(NOKP.EQ.2)) GO TO 99                            COPO      91
      IF (NOKP.EQ.2) REWIND 9                                             COPO      92
      IF (NOKP.EQ.2) GO TO 113                                            COPO      93
      IF(NOKP.EQ.3) GO TO 99                                              COPO      94
      REWIND 7                                                            COPO      95
  113 CONTINUE                                                            COPO      96
      NCO=0                                                               COPO      97
  999 IF(NCO.NE.0) GO TO 18                                               COPO      98
      IF (NOKP.NE.2) GO TO 1115                                           COPO      99
      READ(9) NPLOT                                                       COPO     100
      IF (EOF(9).NE.0) GO TO 99                                           COPO     101
      IF (NOKP.EQ.2) GO TO 114                                            COPO     102
 1115 CONTINUE                                                            COPO     103
      READ(7)NPLOT                                                        COPO     104
      IF (EOF(7) .NE. 0) GO TO 8061                                       COPO     105
  114 CONTINUE                                                            COPO     106
      IF (NOKP.EQ.2) NCKP=1                                               COPO     107
      IF (NOKP.EQ.2) NNKP=KLP                                             COPO     108
      IF (NOKP.EQ.1) KLP=KLP+1                                            COPO     109
   18 NCO=NCO+1                                                           COPO     110
      IF(NCO.EQ.NPLOT) NCO=0                                              COPO     111
      IF (NCKP.NE.2) GO TO 111                                            COPO     112
                     READ (9) TYPE,TITLE,SYSTEM,MODEL,IDIG,SCAPLT         COPO     113
      IF (EOF(9).NE.0) GO TO 99                                           COPO     114
      IF (NOKP.EQ.2) GO TO 116                                            COPO     115
```

```
  111 READ (7) TYPE,TITLE,SYSTEM,MODEL,IDIG,SCAPLT
      IF (EOF(7) .NE. 0) GO TO 99
      IF((TYPE.EQ.PL).AND.(IDIG.EQ.1)) WRITE(9) NPLOT
      IF((TYPE.EQ.PL).AND.(IDIG.EQ.1)) WRITE(9) TYPE,TITLE,SYSTEM,MODEL,
     1IDIG,SCAPLT
  115 CONTINUE
      IF((TYPE.EQ.TM).OR.(TYPE.EQ.FR).OR.(TYPE.EQ.PS).OR.(TYPE.EQ.RL).OR
     X.(TYPE.EQ.PC)) GO TO 1
      WRITE (3,905)
  905 FORMAT (10X,"TYPE FIELD DOES NOT CONTAIN VALID OPTION--WILL CONTIN
     XUE SEARCHING FOR RECORD WITH VALID TYPE FIELD")
      GO TO 999
    1 CONTINUE
      IF((NCO.EQ.0).OR.(NCO.EQ.1)) WRITE(3,500)TITLE
  500 FORMAT (/5X,8A10)
C
C     IS THIS A TIME HISTORY PLOT?
      IF(TYPE.NE.TM) GO TO 2
C***************** TIME HISTORY PLOT SECTION *****************
      READ(7)NYU,(NAME(J),J=1,NYU)
      M=1
    4 READ(7)TIME(M),(V(K),K=1,NYU)
      IF(TIME(M).LT.0.) GO TO 3
C
C     DATA SET 5 IS A TEMPORARY TIME HISTORY DATA STORAGE
C
      WRITE(5)(V(K),K=1,NYU)
      M=M+1
      GO TO 4
    3 M=M-1
      REWIND 5
      CALL SCALE (TIME,8.5,M,1)
      TCM=TIME(M+2)
C
C-----------------------------------------------------------------------
C----DO LOOP 90 WILL PLOT  NYU  TIME HISTORY PLOTS----------
      DO 90 KK=1,NYU
      ENV=0
      DO 10 KC=1,M
      READ(5) (V(K),K=1,NYU)
   10 VP(KC)=V(KK)
C
C     CHECK FOR A CSTAR TYPE TIME HISTORY
C
      IF((NAME(KK).NE.CSCRUV).AND.(NAME(KK).NE.CSPOA)) ENV=1
      IF(ENV.EQ.1) GO TO 9
C
C     IF CSTAR OPTION, NORMALIZE DATA BY DIVIDING BY LAST DATA VALUE.
C
      DO 25 KC=1,M
   25 VP(KC)=VP(KC)/VP(M)
    9 REWIND 5
      IF(MOD(KK,4).NE.1) GO TO 31
      IF(KK.EQ.1) GO TO 32
C
C     SPACE TO NEXT PLOT PAGE
      CALL PLOT(28.4950*CM,-2.*CM,-3)
```

```
   32 CALL PLOT (2.*CM,2.*CM,-3)                                  COPO     173
      CALL FACTOR(HALF)                                           COPO     174
      CALL AX90 (0.,0.,"TIME HISTORY---SECONDS",-22,8.5,0.,TIME(M+1), COPO     175
     XTCM)                                                        COPO     176
      CALL FACTOR(1.)                                             COPO     177
      IF(KK.EQ.1) TIME(M+2)=TIME(M+2)*1.27                        COPO     178
      YO=-20.                                                     COPO     179
      YN=13.                                                      COPO     180
      CALL SYMBOL (.5*CM,23.2*CM,.20*CM,TITLE,0.,80)              COPO     181
      GO TO 33                                                    COPO     182
   31 YN=-6.                                                      COPO     183
      YO=YO+6.                                                    COPO     184
   33 CALL PLOT (0.,YN*CM,-3)                                     COPO     185
      CALL SCALE (VP,2.5,M,1)                                     COPO     186
C                                                                 COPO     187
C     IF ENV NOT EQUAL TO ONE, GENERATE APPROPRIATE CSTAR ENVELOPE COPO     188
C                                                                 COPO     189
      IF(ENV.EQ.1) GO TO 29                                       COPO     190
      IF((NAME(KK).EQ.CSCRUV).AND.(VP(M+2).LT..8)) VP(M+2)=.8     COPO     191
      IF((NAME(KK).EQ.CSPOA).AND.(VP(M+2).LT.1.)) VP(M+2)=1.      COPO     192
   29 CONTINUE                                                    COPO     193
      CALL FACTOR(HALF)                                           COPO     194
      CALL AX90 (0.,0.,NAME(KK),10,2.5,90.,VP(M+1),VP(M+2))       COPO     195
      VP(M+2)=VP(M+2)*1.27                                        COPO     196
      CALL FACTOR(1.)                                             COPO     197
C                                                                 COPO     198
C     IF ENV NOT EQUAL TO ONE, GENERATE APPPOPRIATE CSTAR ENVELOPE COPO     199
C                                                                 COPO     200
      IF(ENV.EQ.1) GO TO 26                                       COPO     201
      IF(NAME(KK).NE.CSCRUV) GO TO 28                             COPO     202
      CTML(20)=TIME(M+2)                                          COPO     203
      CENVL(19)=VP(M+1)                                           COPO     204
      CENVL(20)=VP(M+2)                                           COPO     205
      CALL LINE (CTML,CENVL,18,1,0,0)                             COPO     206
      CTMU(20)=CTML(20)                                           COPO     207
      CENVU(19)=CENVL(19)                                         COPO     208
      CENVU(20)=CENVL(20)                                         COPO     209
      CALL LINE (CTMU,CENVU,18,1,0,0)                             COPO     210
      GO TO 26                                                    COPO     211
   28 PTML(12)=TIME(M+2)                                          COPO     212
      PENVL(11)=VP(M+1)                                           COPO     213
      PENVL(12)=VP(M+2)                                           COPO     214
      CALL LINE(PTML,PENVL,10,1,0,0)                              COPO     215
      PTMU(23)=PTML(12)                                           COPO     216
      PENVU(22)=PENVL(11)                                         COPO     217
      PENVL(23)=PENVL(12)                                         COPO     218
      CALL LINE (PTMU,PENVU,21,1,0,0)                             COPO     219
   26 CONTINUE                                                    COPO     220
      CALL LINE (TIME,VP,M,1,0,0)                                 COPO     221
C                                                                 COPO     222
C     NOPLOTS KEEPS TRACK OF NUMBER OF PLOTS                      COPO     223
      NOPLOTS=NOPLOTS+1                                           COPO     224
      WRITE(3,502)NAME(KK)                                        COPO     225
  502 FORMAT (10X,A10,"  TIME HISTORY PLOT COMPLETED")            COPO     226
   90 CONTINUE                                                    COPO     227
C-------------------------- END OF DO LOOP 90 -------------------------- COPO     228
C                                                                 COPO     229
```

```
C     SPACE TO NEXT PLOT PAGE
       CALL PLOT (24.4950*CM,YO*CM,-3)
       GO TO 999
C
C     IS THIS A FREQUENCY RESPONSE PLOT?
    2 IF(TYPE.NE.FR) GO TO 5
C
C-------------------------------------------------------------------
C***** FREQUENCY RESPONSE AND POWER SPECTRAL DENSITY PLOT SECTION *****
C
      READ(7)NUDE,(NAME(J),J=1,NUDE)
      K=1
C
C----READ AND CONVERT FREQUENCY RESPONSE DATA----------
C     CHECK TO SEE IF THERE ARE TWO OR FOUR PLOTS
C
    7 IF(NUDE.EQ.TWO) READ(7)FREQ,VP(K),VP2(K)
      IF(NUDE.EQ.FOUR) READ(7)FREQ,VP(K),VP2(K),S10(K),S20(K)
      IF(IFRE.EQ.99) GO TO 6
      TIME(K)=ALOG10(FREQ)
      K=K+1
      GO TO 7
    6 K=K-1
      GO TO 17
C
C     IS THIS A POWER SPECTRAL DENSITY PLOT?
    5 IF(TYPE.NE.PS) GO TO 23
C
C----THIS SECTION WILL READ AND CONVERT POWER SPECTRAL DENSITY DATA
C
      READ(7) (NAME(J),J=1,2)
      YI=16.
      YA=-18.
      K=1
   16 READ(7)FREQ,VALUE
      IF(IFRE.EQ.99) GO TO 13
      FREQ=FREQ/6.2832
      TIME(K)=ALOG10(FREQ)
      IF(K.EQ.1) GO TO 35
      IF(TIME(K).LE.PTIME) GO TO 13
   35 PTIME=TIME(K)
      IF(VALUE)14,24,34
   14 VP(K)=ALOG10(-VALUE)
      GO TO 44
   34 VP(K)=ALOG10(VALUE)
      GO TO 44
   24 VP(K)=0
   44 IF((VP(K)).LT.YI) YI=VP(K)
      IF((VP(K)).GT.YA) YA=VP(K)
   15 K=K+1
      GO TO 16
   13 K=K-1
C-------------------------------------------------------------------
C
C     DETERMINE X AXIS MINIMUM AND INCREMENT
   17 DO 20 J=1,11
      IF(TIME(1).LT.FLOAT(6-J))LOW=FLOAT(5-J)
```

```
      IF(TIME(K).GT.FLOAT(J-8))HIGH=FLOAT(J-7)                          COPO     287
   20 CONTINUE                                                          COPO     288
      S=LOW                                                             COPO     289
      DO 30 J=1,5                                                       COPO     290
      L=J                                                               COPO     291
      S=S+1                                                             COPO     292
      IF(S.EQ.HIGH) GO TO 8                                             COPO     293
   30 CONTINUE                                                          COPO     294
    8 XLEN=17.                                                          COPO     295
      IF(TYPE.EQ.PS) XLEN=18.                                           COPO     296
      XDIST=(XLEN/L)*CM                                                 COPO     297
      TIME(K+1)=LOW                                                     COPO     298
      TIME(K+2)=1./XDIST                                                COPO     299
      NUDIF=1                                                           COPO     300
C                                                                       COPO     301
C----DO LOOP 101 WILL PLOT ONE PSD  OR TWO FREQ. RESPONSE PLOTS--------- COPO     302
C                                                                       COPO     303
   55 DO 101 KK=1,2                                                     COPO     304
      X=2.*CM                                                           COPO     305
      Y=13.*CM                                                          COPO     306
      IF(KK.EQ.1) GO TO 221                                             COPO     307
      X=0.                                                              COPO     308
      Y=-13.*CM                                                         COPO     309
      IF(TYPE.NE.PS) GO TO 221                                          COPO     310
      X=0.0                                                             COPO     311
      Y=13.*CM                                                          COPO     312
  221 IF(KK.NE.2) GO TO 703                                             COPO     313
      CALL PLOT(X,Y,-3)                                                 COPO     314
      GO TO      11                                                     COPO     315
C                                                                       COPO     316
C     GENERATE X AXIS (LOG) FOR PSD OR FREQUENCY RESPONSE               COPO     317
C                                                                       COPO     318
  703 IF(TYPE.EQ.FR) GO TO 700                                          COPO     319
      CALL PLOT(CM,0.,-3)                                               COPO     320
      CALL SYMBOL(.5*CM,25.2*CM ,.20*CM,TITLE,0.,80)                    COPO     321
      GO TO 701                                                         COPO     322
  700 CALL PLOT(X,Y,-3)                                                 COPO     323
      CALL SYMBOL (.5*CM,12.2*CM,.20*CM,TITLE,0.,80)                    COPO     324
  701 XN=0.                                                             COPO     325
      YN=0.                                                             COPO     326
      IF(TYPE.EQ.FR)YN=-13.*CM                                          COPO     327
      LK=L+1                                                            COPO     328
      PL=1.                                                             COPO     329
      DO 40 J=1,LK                                                      COPO     330
      CALL SYMBOL (XN,YN,.65*CM,13,0.,-1)                               COPO     331
      IF(J.EQ.LK) GO TO 40                                              COPO     332
      DO 50 J1=1,8                                                      COPO     333
      XP=XN+XDIST*LOG(J1)                                               COPO     334
      CALL SYMBOL (XP,YN,.2*CM,13,0.,-1)                                COPO     335
      PL=PL+1.                                                          COPO     336
      XPO=XP-.05*CM                                                     COPO     337
      CALL NUMBER (XPO,YN-.30*CM,.15*CM,PL,0.,-1)                       COPO     338
   50 CALL PLOT (XP,YN,3)                                               COPO     339
      PL=1.                                                             COPO     340
   40 XN=XN+XDIST                                                       COPO     341
      CALL PLOT (0.,YN,2)                                               COPO     342
      XPM=-.2*CM                                                        COPO     343
```

```
      XFM=.4*CM
      XNO=LOW
      DO 60 J=1,LK
      CALL NUMBER (XBM,YN-.70*CM,.25*CM,10.,0.,-1)
      CALL NUMBER (XFM,YN-.50*CM,.125*CM,XNO,0.,-1)
      XNO=XNO+1.
      XBM=XBM+XDIST
   60 XFM=XFM+XDIST
C
C     IS THIS A PSD PLOT?
      IF(TYPE.EQ.PS) GO TO 19
C
C    GENERATE Y AXIS (CM) FOR FREQUENCY RESPONSE AND PLOT DATA LINE
C
      CALL SYMBOL (5.5*CM,YN-1.15*CM,.25*CM,"FREQUENCY RESPONSE--RADIANS
     X/SEC",0.,31)
  901 FORMAT (F10.5)
      READ (1,901) DDB
      IF (EOF(1) .NE. 0)                                  GO TO 902
      IF (DDB .EQ. 0.)                                    GO TO 902
      VMAX=-100.
      DO 904 IJK=1,K
         IF (VP(IJK) .GT. VMAX)   VMAX=VP(IJK)
  904    CONTINUE
      VMAX=FLOAT(IFIX(VMAX+DDB/2.))
      VP(K+1)=VMAX-6.0*DDB
      VP(K+2)=DDB
      VMIN=VMAX-11.5*DDB
      DO 906 JKL=1,K
         IF (VP(JKL) .LT. VMIN)   VP(JKL)=VMIN
  906    CONTINUE
                                                          GO TO 903
  902 CALL SCALE (VP,6.0,K,1)
  903 CONTINUE
      LAB(1)=LABEL
      LAB(3)=LABEL2
      J=2
      IF(NUDIE.EQ.1) J=J-2
      LAB(2)=NAME(J+1)
      LAB(4)=NAME(J+2)
      CALL FACTOR(HALF)
      CALL AX9C (0.,0.,LAB,40,6.,90.,VP(K+1),VP(K+2))
      VP(K+2)=VP(K+2)*1.27
      CALL FACTOR(1.)
      NOPLOTS=NOPLOTS+1
      CALL LINE (TIME,VP,K,1,0,0)
      GO TO 12
C
C    GENERATE Y AXIS (LOG) FOR PSD
C
   19 CALL SYMBOL (5.25*CM,-1.15*CM,.25*CM,"POWER SPECTRAL DENSITY--CYCL
     XES/SEC",0.,34)
      DO 70 J=1,32
      IF(YI.LT.FLOAT(16-J))LOWY=FLOAT(15-J)
      IF(YA.GT.FLOAT(J-18))HIGHY=FLOAT(J-17)
   70 CONTINUE
      S=LOWY
```

```
      DO 80 J=1,10                                             COPO     401
      LY=J                                                     COPO     402
      S=S+1                                                    COPO     403
      IF(S.EQ.HIGHY) GO TO 21                                  COPO     404
   80 CONTINUE                                                 COPO     405
   21 YDIST=(25./LY)*CM                                        COPO     406
      INO=25./LY                                               COPO     407
      VP(K+1)=LOWY                                             COPO     408
      VP(K+2)=1./YDIST                                         COPO     409
      YN=0.                                                    COPO     410
      LKY=LY+1                                                 COPO     411
      PL=1.                                                    COPO     412
      DO 140 J=1,LKY                                           COPO     413
      CALL SYMBOL (0.,YN,.3*CM,13,90.,-1)                      COPO     414
      IF(J.EQ.LKY) GO TO 140                                   COPO     415
      DO 100 J1=1,9                                            COPO     416
      IF(INO.LT.4) GO TO 76                                    COPO     417
      IF(INO.LT.6) GO TO 77                                    COPO     418
      IF(INO.LE.8) GO TO 78                                    COPO     419
      IF(INO.LE.12) GO TO 79                                   COPO     420
      IF(J1.GT.5) GO TO 78                                     COPO     421
      DO 110 J2=1,4                                            COPO     422
      YQ=YN+YDIST*YLOG(J2,J1)                                  COPO     423
  110 CALL SYMBOL (0.,YQ,.2*CM,13,90.,-1)                      COPO     424
      GO TO 76                                                 COPO     425
   77 IF(J1.GT.3) GO TO 76                                     COPO     426
      GO TO 79                                                 COPO     427
   78 IF(J1.GT.6) GO TO 76                                     COPO     428
   79 DO 120 J2=1,2                                            COPO     429
      YQ=YN+YDIST*YLOG(J2,J1)                                  COPO     430
  120 CALL SYMBOL (0.,YQ,.2*CM,13,90.,-1)                      COPO     431
   76 IF(J1.EQ.9) GO TO 100                                    COPO     432
      YQ=YN+YDIST*LOG(J1)                                      COPO     433
      CALL SYMBOL (0.,YQ,.3*CM,13,90.,-1)                      COPO     434
      IF(INO.LT.4) GO TO 100                                   COPO     435
      PL=PL+1.                                                 COPO     436
      CALL NUMBER (-.25*CM,YQ,.15*CM,PL,0.,-1)                 COPO     437
  100 CONTINUE                                                 COPO     438
      PL=1.                                                    COPO     439
  140 YN=YN+YDIST                                              COPO     440
      CALL PLOT (0.,0.,2)                                      COPO     441
      YNO=LOWY                                                 COPO     442
      YEM=.4*CM                                                COPO     443
      YBM=-.2*CM                                               COPO     444
      DO 130 J=1,LKY                                           COPO     445
      CALL NUMBER (-.5*CM,YBM,.3*CM,10.,90.,-1)                COPO     446
      CALL NUMBER (-.8*CM,YEM,.2*CM,YNO,90.,-1)                COPO     447
      YNO=YNO+1.                                               COPO     448
      YBM=YBM+YDIST                                            COPO     449
  130 YEM=YEM+YDIST                                            COPO     450
      LAB(1)=NAME(1)                                           COPO     451
      LAB(2)=LABEL2                                            COPO     452
      LAB(3)=NAME(2)                                           COPO     453
      CALL SYMBOL (-1.3*CM,10.3*CM,.3*CM,LAB,90.,30)           COPO     454
      GO TO 22                                                 COPO     455
C                                                              COPO     456
C     GENERATE SECOND FREQUENCY RESPONSE Y AXIS AND PLOT DATA LINE COPO     457
```

```
   11 VP2(K+1)=-270.
      VP2(K+2)=60.*1.27
      LAB(1)=LABEL1
      LAB(3)=LABEL2
      J=2
      IF(NUDIE.EQ.1) J=J-2
      LAB(2)=NAME(J+1)
      LAB(4)=NAME(J+2)
      CALL FACTOR(HALF)
      CALL AX90 (0.,0.,LAB,40,6.,90.,-270.,60.)
      CALL FACTOR(1.)
      NOPLOTS=NOPLOTS+1
      CALL LINE (TIME,VP2,K,1,0,0)
C
C     SPACE TO NEXT PLOT PAGE
   12 IF(KK.EQ.2) CALL PLOT (28.4950*CM,0.,-3)
      IF(KK.EQ.1) GO TO 101
      GO TO 101
C
C     PLOT PSD DATA LINE
C
   22 CALL LINE (TIME,VP,K,1,0,0)
      NOPLOTS=NOPLOTS+1
   42 IF(IFRE.EQ.99) GO TO 36
      READ(7)FREQ,VALUE,VALUE
      GO TO 42
   36 CONTINUE
C
C     SPACE TO NEXT PLOT PAGE
      CALL PLOT (29.4950*CM,0.,-3)
      WRITE(3,1113)
 1113 FORMAT(/10X,"P S D  PLOTS COMPLETED"/)
      GO TO 98
  101 CONTINUE
C
C------------------------- END OF CC LOOP 101 -------------------------
C
      WRITE(3,1112)
 1112 FORMAT (/11X,"FREQUENCY RESPONSE PLOTS COMPLETED"/)
C
C     CHECK FOR MORE THAN TWO FREQUENCY RESPONSE PLOTS
C
      IF(TYPE.EQ.PS) GO TO 999
      IF(NUDC.EQ.TWO) GO TO 999
      IF(NUDIE.EQ.2) GO TO 999
      NUDIE=2
      DO 200 JKS=1,K
      VP(JKS)=S10(JKS)
  200 VP2(JKS)=S20(JKS)
      GO TO 56
C
C---------------------------------------------------------------------
C******ROOT LOCUS AND ROOT CONTOUR SECTION**********
C
C
C     IS THIS A ROOT LOCUS PLOT?
```

```
   23 IF(TYPE.EQ.RL) READ(7)N,M                                          COPO
      M=IABS(M)                                                          COPO
C                                                                        COPO
C     CHECK FOR A Z-PLANE ROOT LOCUS (IDIG=1)                            COPO
C                                                                        COPO
      IF(NOKP.EQ.2) GO TO 1901                                           COPO
      IF(IDIG.NE.1) GO TO 201                                            COPO
      YEP= 1                                                             COPO
                                                                         COPO
C----------------------------------------------------------------------- COPO
C*****Z-PLANE ROOT LOCUS SECTION**********                               COPO
C     GENERATE HALF CIRCLE AND EACH AXES                                 COPO
C                                                                        COPO
      CALL PLOT (17.*CM,2.*CM,-3)                                        COPO
      CALL PLOT (0.,20.*CM,2)                                            COPO
      Z=1.                                                               COPO
      Y=20.*CM                                                           COPO
      DO 202 J=1,5                                                       COPO
      CALL SYMBOL (0.,Y,.2*CM,13,90.,-1)                                 COPO
      CALL NUMBER (.5*CM,Y-.3*CM,.25*CM,Z,90.,1)                         COPO
      Z=Z-.5                                                             COPO
  202 Y=Y-5.*CM                                                          COPO
      CALL SYMBOL (2.*CM,6.*CM,.3*CM,"Z - PLANE ROOT LOCUS   (.1 UNIT/CM COPO
     X)",90.,35)                                                         COPO
      Y=10.*CM                                                           COPO
      CALL PLOT (0.,Y,3)                                                 COPO
      X=-12.*CM                                                          COPO
      CALL PLOT (X,Y,2)                                                  COPO
      Z=1.2                                                              COPO
      DO 203 J=1,3                                                       COPO
      CALL PLOT (X,Y,3)                                                  COPO
      CALL PLOT (X,Y,2)                                                  COPO
      CALL SYMBOL (X,Y,.2*CM,13,0.,-1)                                   COPO
      CALL NUMBER (X,Y+.15*CM,.125*CM,Z,90.,1)                           COPO
      CALL PLOT (X,Y+.15*CM,3)                                           COPO
      Z=Z-.4                                                             COPO
  203 X=X+4.*CM                                                          COPO
      CALL SYMBOL (-16.0*CM,.5*CM,.24*CM,TITLE(1),90.,80)                COPO
C                                                                        COPO
C    FOR ROOT LOCUS EACH GAIN INCREMENT HAS A DIFFERENT SYMBOL PLOTTED   COPO
C                                                                        COPO
      CALL SYMBOL (-15.*CM,3.*CM,.24*CM,"GAIN INCREMENT ORDER---",       COPO
     X90.,23)                                                            COPO
      Y=9.1*CM                                                           COPO
      X=-12.15*CM                                                        COPO
      DO 204 J=1,12                                                      COPO
      CALL SYMBOL (X,Y,BIG(J)*CM*1.2,ORDER(J),0.,-1)                     COPO
  204 Y=Y+.85*CM                                                         COPO
      S1=1.01                                                            COPO
      DO 205 J=1,100                                                     COPO
      S1=S1-.01                                                          COPO
      S10(J)=S1                                                          COPO
  205 TIME(J)=-SQRT(1-S10(J)**2)                                         COPO
      S10(101)=-1.2                                                      COPO
      S10(102)=.254                                                      COPO
      TIME(101)=-1.2                                                     COPO
      TIME(102)=.254                                                     COPO
```

```
      CALL PLOT (-12.*CM,0.,-3)
      CALL LINE (TIME,S11,100,1,0,0)
      S1 =-1.01
      DO 206 J=1,100
      S1=S1+.01
  206 S10(J)=S1
      CALL LINE (TIME,S10,100,1,0,0)
      XDT=.1*2.54
      YCREY=.1*2.54
      XMIN=-1.3
      XMAX=1.3
      YMIN=0.
      YMAX=1.0
      GO TO 207
C
C-----------------------------------------------------------------------
C
C***** FOLLOWING SECTION INVOLVES ROOT LOCUS OR ROOT CONTOUR
C
C     READ MAXIMUM AND MINIMUM VALUES FOR Y AXIS AND X AXIS GENERATION
C
  201 READ (1,900)YMIN,YMAX
  900 FORMAT (2F10.0)
      IF (EOF(1) .NE. 0) GO TO 998
      WRITE(3,803)YMIN,YMAX
  803 FORMAT (10X,"YMIN=",F6.1,5X,"YMAX=",F6.1)
C
C     GENERATE X AXIS
C
      CALL SUBSCL(YMAX,YMIN,25.0,1.,10.,YCREY)
      XDT=YCREY
      XMIN=-14.*XDT
      XMAX=4.*XDT
      CALL FACTOR(HALF)
      IF(TYPE.EQ.RL) CALL AXIS (0.,0.,"ROOT LOCUS",-10,9.,0.,XMIN,
     XXDT*2.)
      IF(TYPE.NE.RL) CALL AXIS (0.,0.,"ROOT CONTOUR",-12,9.,0.,XMIN,
     XXDT*2.)
      CALL FACTOR(1.)
      CALL PLOT (14.*CM,0.,3)
      CALL PLOT (14.*CM,25.*CM,2)
      CALL SYMBOL (CM,25.2*CM,.20*CM,TITLE(1),0.,80)
C
C     IS THIS A ROOT CONTOUR PLOT
      IF(TYPE.NE.RL) GO TO 61
C
C     FOR ROOT LOCUS EACH GAIN INCREMENT HAS A DIFFERENT SYMBOL PLOTTED
C
      CALL SYMBOL (CM,26.0*CM,.24*CM,"GAIN INCREMENT ORDER---",0.,23)
      XN=7.10*CM
      YN=26.15*CM
      DO 170 J=1,10
      CALL SYMBOL (XN,YN,.24*CM,ORDER(J),0.,-1)
  170 XN=XN+.85*CM
   61 CONTINUE
      XN=18.3*CM
      YN=25.*CM
```

```
             Y5=YCREY*5.                                                     COPO     629
             YDO=YMIN+25.*YCREY                                              COPO     630
             DO 180 J=1,6                                                    COPO     631
             CALL PLOT (19.*CM,YN,3)                                         COPO     632
             CALL SYMBOL (18.*CM,YN,.4*CM,13,90.,-1)                         COPO     633
             CALL NUMBER (XN,YN,.26*CM,YDO,0.,1)                             COPO     634
             CALL PLOT (18.*CM,YN,3)                                         COPO     635
             YN=YN-5.*CM                                                     COPO     636
         180 YDO=YDO-Y5                                                      COPO     637
             YCREY=YCREY*2.54                                                COPO     638
             XDT=XDT*2.54                                                    COPO     639
             IF(TYPE.NE.RL) GO TO 62                                         COPO     640
         207 CONTINUE                                                        COPO     641
       C                                                                     COPO     642
       C-------------------------------------------------------------------- COPO     643
       C----DO LOOP 190 WILL READ THE ROOTS WITH N BEING THE NO. OF GAIN INCREM COPO  644
       C----AND M BEING THE NO. OF ROOTS TO BE PLOTTED WITHIN EACH GAIN INCREME COPO  645
       C----           BOTH Z-PLANE AND REGULAR ROOT LOCUS ARE PLOTTED HERE  COPO     646
       C                                                                     COPO     647
             DO 190 J=1,N                                                    COPO     648
             DO 190 K=1,M                                                    COPO     649
             IF(K.EQ.1) PTF=1.                                               COPO     650
             IF (NOKP.NE.2) GO TO 1955                                       COPO     651
             READ (9) I,(TIME(L),VP(L),L=1,I)                                COPO     652
             IF (EOF(9).NE.0)  GO TO 1900                                    COPO     653
             IF (NOKP.EQ.2) GO TO 115                                        COPO     654
        1955 CONTINUE                                                        COPO     655
             READ(7)I,(TIME(L),VP(L),L=1,I)                                  COPO     656
             IF (IDIG.EQ.1) WRITE(9)I,(TIME(L),VP(L),L=1,I)                  COPO     657
         115 CONTINUE                                                        COPO     658
             LL=0                                                            COPO     659
             DO 160 L=1,I                                                    COPO     660
       C                                                                     COPO     661
       C     THROW OUT ALL POINTS OUTSIDE X AND Y BOUNDS                     COPO     662
       C                                                                     COPO     663
             IF(((VP(L)).GT.YMAX).OR.((VP(L)).LT.YMIN).OR.((TIME(L)).GT.XMAX).O COPO  664
            XR.((TIME(L)).LT.XMIN)) GO TO 160                                COPO     665
             IF(((VP(L)).EQ.0.).AND.((TIME(L)).EQ.0.)) GO TO 160             COPO     666
             IF((J.NE.1).OR.(K.NE.1)) GO TO 27                               COPO     667
             XN=(TIME(L)-XMIN)/XDT                                           COPO     668
             YN=(VP(L)-YMIN)/YCREY                                           COPO     669
             IF(IDIG.EQ.1) XN=(1.2-VP(L))/XDT                                COPO     670
             IF(IDIG.EQ.1) YN=(1.+TIME(L))/YCREY                             COPO     671
             IF(NOKP.EQ.2) XN=-VP(L)/XDT                                     COPO     672
             IF(NOKP.EQ.2) YN=(16.+(TIME(L)-1.)/.005)/2.54                   COPO     673
             CALL SYMBOL (XN,YN,.35*CM,4,0.,-1)                              COPO     674
             GO TO 160                                                       COPO     675
          27 LL=LL+1                                                         COPO     676
             TIME(LL)=TIME(L)                                                COPO     677
             VP(LL)=VP(L)                                                    COPO     678
         160 CONTINUE                                                        COPO     679
             L=LL                                                            COPO     680
             IF(IDIG.NE.1) GO TO 208                                         COPO     681
             DO 209 JJ=1,L                                                   COPO     682
             VT=VP(JJ)                                                       COPO     683
             VP(JJ)=TIME(JJ)                                                 COPO     684
         209 TIME(JJ)=-VT                                                    COPO     685
```

```
              VP(L+1)=-1.                                                      COPO     686
              VP(L+2)=YCREY                                                    COPO     687
              TIME(L+1)=-1.2                                                   COPO     688
              TIME(L+2)=XDT                                                    COPO     689
              IF (NOKP.NE.2) GO TO 211                                         COPO     690
              VP(L+1)=.92                                                      COPO     691
              TIME(L+1)=0.0                                                    COPO     692
              GO TO 211                                                        COPO     693
          228 VP(L+1)=YMIN                                                     COPO     694
              VP(L+2)=YCREY                                                    COPO     695
              TIME(L+1)=XMIN                                                   COPO     696
              TIME(L+2)=XDT                                                    COPO     697
          211 KD=K                                                             COPO     698
              IF(K.GT.10) KD=K-10                                              COPO     699
              CALL LINE (TIME,VP,L,1,-1,ORDER(KD))                             COPO     700
          190 CONTINUE                                                         COPO     701
         1901 CONTINUE                                                         COPO     702
              IF(IDIG.NE.1) GO TO 1900                                         COPO     703
              IF (KLP.NE.NNKP) GO TO 1900                                      COPO     704
              IF (NOKP.EQ.1) GO TO 1900                                        COPO     705
              CALL PLOT (17.*CM,2.*CM,-3)                                      COPO     706
              CALL FACTOR(.7874)                                               COPO     707
              CALL AXIS(0.,0.,16HZ-PLANE EXPANDED,-16,10.,90.,.92,.01)         COPO     708
              CALL AXIS(0.,8.,9HIMAG AXIS,-9,6.,180.,0.,.01)                   COPO     709
              CALL SYMBOL(-7.5,0.,.14,TITLE(1),90.,80)                         COPO     710
              CALL SYMBOL (-7.,.5,.14,"GAIN INCREMENT ORDER ---",90.,24)       COPO     711
              YY=4.55                                                          COPO     712
              DO 2044 J=1,10                                                   COPO     713
              CALL SYMBOL (-7.1,YY,.14,ORDER(J),0.,-1)                         COPO     714
         2044 YY=YY+.425                                                       COPO     715
              CALL FACTOR(1.0)                                                 COPO     716
              XMIN=.91                                                         COPO     717
              XMAX=1.03                                                        COPO     718
              YMIN=0.                                                          COPO     719
              YMAX=.07                                                         COPO     720
              XDT=.0127                                                        COPO     721
              YCREY=.0127                                                      COPO     722
              KLP=KLP+1                                                        COPO     723
              WRITE (3,1111)                                                   COPO     724
         1111 FORMAT(20X,"EXPANDED ROOT LOCUS COMPLETED")                      COPO     725
              GO TO 207                                                        COPO     726
         1900 CONTINUE                                                         COPO     727
        C                                                                      COPO     728
        C--------------------------------------------------------------------- COPO     729
        C                                                                      COPO     730
        C     CALL SUBROUTINE TO PLOT ZEROES                                   COPO     731
        C                                                                      COPO     732
              CALL READO(YMAX,YMIN,XMAX,XMIN,XDT,YCREY,IDIG ,TIME,VP,NOKP)     COPO     733
              IF(IDIG.NE.1) WRITE(3,1118)                                      COPO     734
              IF(IDIG.EQ.1) WRITE(3,1117)                                      COPO     735
         1117 FORMAT (20X,"Z PLANE ROOT LOCUS PLOT COMPLETED")                 COPO     736
        C                                                                      COPO     737
        C    SPACE TO NEXT PLOT PAGE                                           COPO     738
              IF(IDIG.NE.1) CALL PLOT (30.4950*CM,0.,-3)                       COPO     739
              IF(IDIG.EQ.1) CALL PLOT (25.4950*CM,-2.*CM,-3)                   COPO     740
              NOPLOTS=NOPLOTS+1                                                COPO     741
              IF (NOKP.EQ.2) CALL PLOT (-12.*CM,0.,-3)                         COPO     742
```

```
   94 GO TO 999                                                         COPO       743
C                                                                       COPO       744
C---------------------------------------------------------------------- COPO       745
C*****ROOT CONTOUR PLOT SECTION**********                               COPO       746
C                                                                       COPO       747
   62 IFS=0                                                             COPO       748
      GO TO 67                                                          COPO       749
   66 IFS=1                                                             COPO       750
   67 READ(7)I,(TIME(L),VP(L),L=1,I)                                    COPO       751
      IF((TIME(1).EQ.-1.).AND.(VP(1).EQ.-1.)) GO TO 63                  COPO       752
      LL=0                                                              COPO       753
      DO 260 L=1,I                                                      COPO       754
      IF(((VP(L)).GT.YMAX).OR.((VP(L)).LT.YMIN).OR.((TIME(L)).GT.XMAX).O COPO       755
     XR.((TIME(L)).LT.XMIN)) GO TO 260                                  COPO       756
      IF(((VP(L)).EQ.0.).AND.((TIME(L)).EQ.0.)) GO TO 260               COPO       757
      XN=(TIME(L)-XMIN)/XDT                                             COPO       758
      YN=(VP(L)-YMIN)/YCREY                                             COPO       759
      IF(IFS.GT.0) GO TO 65                                             COPO       760
      CALL SYMBOL (XN,YN,.30*CM,4,0.,-1)                                COPO       761
      GO TO 260                                                         COPO       762
   65 CALL SYMBOL (XN,YN,.10*CM,1,0.,-1)                                COPO       763
  260 CONTINUE                                                          COPO       764
      IF(IFS.EQ.0) GO TO 66                                             COPO       765
      GO TO 67                                                          COPO       766
   63 IDIG=0                                                            COPO       767
      WRITE (3,1119)                                                    COPO       768
 1118 FORMAT (20X,"ROOT LOCUS PLOT COMPLETED")                          COPO       769
 1119 FORMAT (20X,"ROOT CONTOUR PLOT COMPLETED")                        COPO       770
C                                                                       COPO       771
C---------------------------------------------------------------------- COPO       772
C                                                                       COPO       773
C     SPACE TO NEXT PLOT PAGE                                           COPO       774
      CALL PLOT (30.4950*CM,0,-3)                                       COPO       775
      NOPLOTS=NOPLOTS+1                                                 COPO       776
      GO TO 999                                                         COPO       777
  991 WRITE (3,1000)                                                    COPO       778
 1000 FORMAT (/10X,"N O  P L O T S  R E Q U E S T E D")                 COPO       779
      GO TO 99                                                          COPO       780
C                                                                       COPO       781
C     MUST BE A LIMIT CARD FOR EACH REGULAR ROOT LOCUS OR ROOT CONTOUR  COPO       782
C                                                                       COPO       783
  998 WRITE(3,503)                                                      COPO       784
  503 FORMAT (10X,"E R R O R - - NO DATA CARD FOR ROOT LOCUS OR ROOT CON COPO       785
     XTOUR PLOT")                                                       COPO       786
      GO TO 999                                                         COPO       787
   99 CONTINUE                                                          COPO       788
      CALL PLOT (0.,0.,999)                                             COPO       789
C                                                                       COPO       790
C     GENERATE AUTOMATIC PLOT REQUEST                                   COPO       791
C                                                                       COPO       792
      CALL PLOTREC(SNAME,SUBTASK,0,0,0,0,0,NOTA,NOTA,0,0,0,             COPO       793
     XPTITLE,NOPLOTS,0,IVSN,COMMENT)                                    COPO       794
C    XPTITLE,NOPLOTS,0,0,COMMENT)                                       COPO       795
      END                                                               COPO       796
```

```
      SUBROUTINE READO(YMAX,YMIN,XMAX,XMIN,YCREX,YCREY,IDIG,TIME,VP,NOKP
     1)
C
C-----------------------------------------------------------------------
C      THIS SUBROUTINE READS AND PLOTS THE ZEROES OF THE TRANSFER
C      FUNCTION FOR A ROOT LOCUS
C-----------------------------------------------------------------------
C
      DIMENSION TIME(1),VP(1)
      IF  ((NOKP.NE.2) .OR.(IDIG.NE.1)) GO TO 1234
      READ (9) I,(TIME(L),VP(L),L=1,I)
      IF(EOF(9).NE.0) GO TO 161
      IF((NOKP.EQ.2).AND.(IDIG.EQ.1)) GO TO 11
 1234 CONTINUE
      READ(7) I,(TIME(L),VP(L),L=1,I)
      IF (IDIG.EQ.1      ) WRITE (9) I,(TIME(L),VP(L),L=1,I)
   11 CONTINUE
      DO 160 L=1,I
      IF((VP(L).GT.YMAX).OR.(VP(L).LT.YMIN).OR.(TIME(L).GT.XMAX).OR.(TIM
     -E(L).LT.XMIN)) GO TO 160
      XN=(TIME(L)-XMIN)/YCREX-.039
      YN=(VP(L)-YMIN)/YCREY-.049
      IF(IDIG.NE.1) GO TO 159
      XN=(1.2-VP(L))/YCREX-.039
      YN=(1.+TIME(L))/YCREY-.049
      IF (YCREY.EQ..0127) XN=-VP(L)/YCREY -.039
      IF (YCREY.EQ..0127) YN=(16.+(TIME(L)-1.)/.005)/2.54 -.049
  159 CALL SYMBOL(XN,YN,.140   ,54,0.,-1)
  160 CONTINUE
  161 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CSTAR                                                  CSTAR       2
C                                                                       CSTAR       3
C---------------------------------------------------------------------- CSTAR       4
C     THIS SUBROUTINE CONTAINS THE ENVELOPE CURVES FOR CSTAR OPTIONS    CSTAR       5
C     CRUISE AND POWER APPROACH                                         CSTAR       6
C---------------------------------------------------------------------- CSTAR       7
C                                                                       CSTAR       8
      COMMON/CSTARC/CTML,CTMU,CENVL,CENVU/CSTARP/PTML,PTMU,PENVL,PENVU  CSTAR       9
      DIMENSION CTML(20),CTMU(20),CENVL(20),CENVU(20),                  CSTAR      10
     X          PTML(12),PTMU(23),PENVL(12),FENVU(23)                   CSTAR      11
      DATA CTML/  0.00,   .20,   .40,   .50,   .60,   .70,   .80,   .90, CSTAR      12
     X            1.00,  1.10,  1.20,  1.30,  1.50,  1.70,  1.80,  1.90, CSTAR      13
     X            2.00,  5.00,  0.00,  0.00/,                           CSTAR      14
     XCENVL/,00,  .000,  .775,  .855,   .890, .890,  .885,  .855,  .800, CSTAR      15
     X            .785,  .795,  .810,   .855, .905,  .926,  .940,  .950, CSTAR      16
     X            .950,  .000,  .000/,                                  CSTAR      17
     XCTMU/0.0,  .020,  .075,  .113,  .180,  .200,  .300,  .340,  .400,  CSTAR      18
     X           .600,  .800, 1.000, 1.100, 1.200, 1.300, 1.400, 1.500, CSTAR      19
     X          5.000, 0.000, 0.000/,                                   CSTAR      20
     XCENVU/0.0,  .800, 1.200, 1.400, 1.600, 1.645, 1.730, 1.735, 1.720, CSTAR      21
     X           1.555, 1.355, 1.200, 1.145, 1.105, 1.070, 1.055, 1.050, CSTAR      22
     X           1.050, 0.000, 0.000/                                   CSTAR      23
      DATA PTML/  0.00,   .20,   .60,  1.40,  2.25,  2.50,  2.75,  3.00, CSTAR      24
     X            3.25,  3.50,  0.00,  0.00/,                           CSTAR      25
     XPENVL/0.0,  0.00,   .55,   .89,   .89,   .76,   .69,   .59,   .73, CSTAR      26
     X             .85,   .00,   .00/,                                  CSTAR      27
     XPTMU/0.0,   .01,   .06,   .10,   .16,   .20,   .30,   .42,   .50,  CSTAR      28
     X      .60,   .70,   .80,   .90,  1.00,  1.20,  1.33,  1.50,  2.00, CSTAR      29
     X     2.50,  3.00,  3.50,  0.00,  0.00/,                           CSTAR      30
     XPENVU/0.0,   .50,  1.00,  1.20,  1.50,  1.62,  1.84,  2.00,  2.08, CSTAR      31
     X     2.16,  2.21,  2.23,  2.24,  2.21,  2.08,  2.00,  1.92,  1.71,  CSTAR      32
     X     1.55,  1.39,  1.17,  0.00,  0.00/                            CSTAR      33
      END                                                               CSTAR      34
```

```
      SUBROUTINE SUBSCL(XMAX,XMIN,S,KFAC,DIV,SCALE)                     SUBSCL       2
C                                                                       SUBSCL       3
C---------------------------------------------------------------------- SUBSCL       4
C     THIS SUBROUTINE COMPUTES THE SCALING FACTOR FOR ROOT LOCUS PLOTS  SUBSCL       5
C---------------------------------------------------------------------- SUBSCL       6
C                                                                       SUBSCL       7
      DIMENSION X(1)                                                    SUBSCL       8
      REAL FAC(4)                                                       SUBSCL       9
      DATA FAC/1.,2.,5.,10./                                            SUBSCL      10
      A=XMAX-XMIN                                                       SUBSCL      11
      IF(A.EQ.0.) GO TO 17                                              SUBSCL      12
      SLEN=S*DIV/10.                                                    SUBSCL      13
      B=A/SLEN                                                          SUBSCL      14
      IF(B.GT..5.AND.B.LE.1.) GO TO 401                                 SUBSCL      15
      IF(B.LE.5.) GO TO 38                                              SUBSCL      16
      GO TO 41                                                          SUBSCL      17
   17 WRITE(3,20)                                                       SUBSCL      18
   20 FORMAT("  DATUM MAXIMUM AND MINIMUM ARE THE SAME"  )              SUBSCL      19
   40 SCALE=.01                                                         SUBSCL      20
      GO TO 5                                                           SUBSCL      21
  401 SCALE=1.                                                          SUBSCL      22
      GO TO 5                                                           SUBSCL      23
   41 DO 4 I=1,999                                                      SUBSCL      24
      DO 4 J=1,3                                                        SUBSCL      25
      IF(B.GE.FAC(J)*10.**(I-1)) GO TO 4                                SUBSCL      26
      SCALE=FAC(J)*10.**(I-1)                                           SUBSCL      27
      GO TO 5                                                           SUBSCL      28
    4 CONTINUE                                                          SUBSCL      29
      GO TO 5                                                           SUBSCL      30
   38 DO 39 I=1,999                                                     SUBSCL      31
      DO 39 J=1,3                                                       SUBSCL      32
      IF(B.LE.FAC(4-J)*10.**(-I+1))    GO TO39                          SUBSCL      33
      SCALE=FAC(5-J)*10.**(-I+1)                                        SUBSCL      34
      GO TO 5                                                           SUBSCL      35
   39 CONTINUE                                                          SUBSCL      36
    5 A=AMOD(XMIN,SCALE)                                                SUBSCL      37
   19 IF(XMIN.GT.0.)XMIN=XMIN-A                                         SUBSCL      38
      IF(XMIN.LT.0.)XMIN=XMIN-SCALE-A                                   SUBSCL      39
      IF(KFAC.GT.0) GOT 050                                             SUBSCL      40
      XMIN=XMIN+SCALE*SLEN                                              SUBSCL      41
      SCALE=-SCALE                                                      SUBSCL      42
   50 CONTINUE                                                          SUBSCL      43
      RETURN                                                            SUBSCL      44
      END                                                               SUBSCL      45
```

APPENDIX 2

NAMELIST CODE

THE CONDITION CODES AND INPUT DATA ARE CONTAINED IN THE NAMELIST CODE AND ARE LISTED BELOW. ALL OF THE CODES AND DATA ARE INITIALIZED TO ZERO AT THE START OF EACH CASE UNLESS THE SAV OPTION IS SET

CONDITION CODES (INTEGER VARIABLES)
READ,SYSTEM,OUTPUT,MIXED,DIGITL,FRPS,NUMERS,TRESP,NX,NY,NU,NXC,NUC,
ZOH,N1,N2,CONTUR,MULTRT,MODEL,NSCALE,CMAT,NK2,FORM,IPT,IGO,SAV,
IFLAG,READ3

INPUT DATA (REAL VARIABLES)
DELT,FINALT,IFREQ,FFREQ,DELFRQ,M,GAIN1,GAIN2

CONDITION CODE DESCRIPTION (INTEGER VARIABLES)

READ	1	DATA MATRICES INPUT THROUGH LOAD SUBROUTINE
	2	DATA MATRICES CONSTRUCTED IN USER WRITTEN MATRIX SUBROUTINE
	3	DATA FROM PREVIOUS CASE ALTERED IN USER WRITTEN CHANGE SUBROUTINE
	4	DATA MATRICES CONSTRUCTED FROM BLOCK DIAGRAM INFORMATION IN CLASS SUBROUTINE
SYSTEM	1	OPEN LOOP SYSTEM ANALYSIS
	2	CLOSED LOOP SYSTEM ANALYSIS
	3	ROOT LOCUS ANALYSIS
OUTPUT	1	$y = Hx$
	2	$y = Hx + G\dot{x}$
	3	$y = Hx + Fu$
	4	$y = Hx + G\dot{x} + Fu$
MIXED	0	NO ACTION
	1	MIXED SYSTEM ANALYSIS (SEE TABLE V.) SYSTEM MATRICES ARE CONSTRUCTED IN A TWO-STEP PROCESS, STEP 1 SPECIFIES OPEN LOOP PLANT (I.E. SPECIFY A,B, C,H,G,F REGARDLESS OF VALUE OF SYSTEM). STEP 2 AUGMENTS PLANT WITH CONTROL SYSTEM DESCRIBED BY BLOCK DIAGRAM USING CLASS. SYSTEM SPECIFIES THE TYPE OF ANALYSIS FOR THE AUGMENTED SYSTEM.
DIGITL	0	CONTINUOUS SYSTEM ANALYSIS
	1	SAMPLED-DATA SYSTEM ANALYSIS
	2	DISCRETE SYSTEM ANALYSIS

IF DIGITL $\neq 0$, DELT SPECIFIES THE SAMPLE PERIOD OF THE DISCRETE OR SAMPLED-DATA SYSTEM.

FRPS	0	NOT APPLICABLE
	1	FREQUENCY RESPONSE CALCULATED FOR EACH TRANSFER FUNTION S-PLANE IF DIGITL = 0 W-PLANE IF DIGITL = 1,2 (DELT REQUIRED)
	-1	S-PLANE FREQUENCY RESPONSES CALCULATED FROM Z-TRANSFER FUNCTIONS WITH DIGITL = 1,2 (DELT REQUIRED)
	2	S-PLANE POWER SPECTRA CALCULATED (DIGITL= 0)
NUMERS	0	NUMERATOR ZEROES OF S- OR Z-TRANSFER FUNCTIONS CALCULATED
	1	NUMERATOR ZEROES NOT CALCULATED
		CONTROL WILL COMPUTE TRANSFER FUNCTION NUMERATOR ZEROES FOR ALL INPUT-OUTPUT PAIRS DEFINED BY THE INPUT AND OUTPUT VECTORS. FOR MIXED SYSTEM ANALYSIS, THE ITHINU AND ITHINY OPTIONS ALLOW UNWANTED TRANSFER FUNCTIONS TO BE ELIMINATED.
TRESP	0	NO ACTION
	N	N TRANSIENT RESPONSES CALCULATED. 'DELT SPECIFIES INTEGRATION STEP SIZE. IF DISC INPUT ROUTINE IS USED, THERE MUST BE N INPUT CARDS AT THE END OF THE DATA CASE GIVING THE INPUT STEP FUNCTION.
NX,NY,NU		DIMENSIONS OF X, Y, AND U VECTORS. IF MIXED = 1, NX, NY, AND NU SPECIFY DIMENSIONS OF THE OPEN LOOP PLANT (STEP 1). STATES ADDED IN STEP 2 OF THE MIXED OPTION AUTOMATICALLY INCREMENT NX, NY, AND NU.
NXC,NUC		DIMENSIONS OF STATE AND INPUT VECTORS CORRESPONDING TO THE CONTINUOUS SUBSYSTEM (PLANT) OF A SAMPLED-DATA SYSTEM. THE PLANT MUST BE PARTITIONED IN THE UPPER LEFT POSITION OF THE SYSTEM MATRICES (A,B,H,F,ETC.) $NXC \leq NX$, $NUC \leq NU$
ZOH		FOR SAMPLED-DATA SYSTEMS, THE NUMBER OF INPUTS TO THE PLANT WHICH ARE OUTPUTS OF ZERO-ORDER-HOLD DEVICES. THESE MUST BE THE FIRST ZOH COMPONENTS OF THE INPUT VECTOR, U.
N1,N2		THE ROOT LOCUS OPTION ALLOWS TWO FEEDBACK GAINS TO BE SPECIFIED. N1 IS THE NUMBER OF ITERATIONS OF THE FIRST VARIABLE (K1,K2) AND N2 IS THE NUMBER OF ITERATIONS OF THE SECOND VARIABLE (K3,K4). (COMMONLY, N2 =0). IF N1 > 0, GAIN INCREMENTS ARE ARITHMETIC (0,1,2,3,..) IF N1 < 0, GAIN INCREMENTS ARE GEOMETRIC (0,1,2,4,8,.) Gain increments of second variable are the same as the first; N2 must be >0.
CONTUR	0	NOT APPLICABLE
	1	ROOT CONTOUR OPTION FOR PARAMETER VARIATION STUDIES CONTROL DETERMINES ONLY SYSTEM EIGENVALUES AND RETURNS TO TOP OF PROGRAM FOR NEXT VARIATION. CONTINUES UNTIL CONTUR SET TO ZERO. (USED WITH IFLAG,READ3,SAV,AND CHANGE)
MULTRT		FOR SAMPLED-DATA SYSTEMS, COMPUTES MULTRT TRANSIENT RESPONSE POINTS FOR EACH SAMPLE PERIOD SO THAT INTERSAMPLE RESPONSE MAY BE INVESTIGATED. ONLY TRANSIENT RESPONSES ARE CALCULATED IF MULTRT IS SET.

MODEL	0	NOT APPLICABLE
	1	MODEL FOLLOWING ON CONSECUTIVE FREQUENCY RESPONSES
NSCALE	0	NOT APPLICABLE
	1	STATE VECTOR TRANSFORMED TO IMPROVE NUMERICAL CONDITIONING IN DETERMINATION OF EIGENVALUES. A MATRIX SCALED BY A DIAGONAL SIMILARITY TRANSFORMATION.
CMAT	0	C MATRIX IS THE IDENTITY MATRIX (C NOT REQUIRED)
	1	$C \neq I$ (C REQUIRED)
NK2	0	$K2 = 0$, $K4 = 0$ (K2,K4 NOT REQUIRED)
	1	$K2 \neq 0$ OR $K4 \neq 0$ (K2,K4 REQUIRED)
FORM	0	PRINT ONLY FOR OUTPUT
	1	PRINT AND PLOT OUTPUT
	2	PLOT ONLY FOR OUTPUT
		THE CONTROL PLOTTER PROGRAM AUTOMATICALLY SCALES ALL PLOTS EXCEPT ROOT LOCUS PLOTS (WHICH REQUIRE AN EXTRA DATA CARD).
IPT		CODE FOR EXTRA PRINTOUT FOR DEBUGGING
	0	NO EXTRA PRINTING
	1,2	EXTRA PRINTING
IGO		CODE FOR DATA REQUIRED BY CLASS SUBROUTINE
	0	INPUT DATA REQUIRED BY CLASS (TABLE V, STEP 2)
	1	CLASS USES DATA FROM PREVIOUS CASE
SAV	0	DATA MATRICES NOT SAVED
	1	DATA MATRICES SAVED FOR SUBSEQUENT CASES. IF MIXED = 1, CONTROL SAVES MATRICES DEFINED IN STEP 1 (CLASS INPUT DATA, STEP 2, IS NOT DESTROYED AND IS AVAILABLE FOR SUBSEQUENT CASES).
IFLAG	0	ON SUBSEQUENT CASE THE CONDITION CODES AND INPUT DATA ARE ZEROED BEFORE THE CALL TO CARD. CARD READS TITLE,NAMELIST, OUTPUT LABEL,AND INPUT LABEL CARDS
	1	ON SUBSEQUENT CASES THE CONDITION CODES AND INPUT DATA OF THE PRESENT CASE WILL BE USED. CARD READS ONLY A TITLE CARD FOR ALL SUBSEQUENT CASES. (THE OPTION MAY BE CANCELED BY SETTING IFLAG = 0 OR BY END OF DATA DECK).
READ3	0	NO ACTION
	1	ON SUBSEQUENT CASES, READ DEFAULTS TO 3 TO FORCE PROGRAM TO THE CHANGE SUBROUTINE. THE OPTION IS USED WITH IFLAG FOR PARAMETER VARIATION STUDIES.

INPUT_DATA_DESCRIPTION (REAL VARIABLES)

DELT	TIME INCREMENT FOR TRANSIENT RESPONSES AND/OR SAMPLE PERIOD FOR SAMPLED-DATA SYSTEMS, SECONDS
FINALT	FINAL TIME FOR TRANSIENT RESPONSES, SECONDS
IFREQ,FFREQ,DELFRQ	INITIAL,FINAL, AND INCREMENTAL FREQUENCIES FOR FREQUENCY RESPONSES OR POWER SPECTRA. DELFRQ = 1.1 IS GOOD FOR MOST APPLICATIONS. FREQUENCIES MUST BE SPECIFIED IN (DELFRQ CANNOT EQUAL 1.0) RADIANS/SEC. (S-PLANE) EVEN FOR DISCRETE AND SAMPLED-DATA SYSTEMS. IF IFREQ = 0., PROGRAM DEFAULTS TO AN INTERNAL SET OF FREQUENCY POINTS SPACED BETWEEN .1 AND 150. RAD/SEC. FOR SAMPLED-DATA FREQUENCY RESPONSES CONTROL DEFAULTS IN THE FOLLOWING MANNER, IF DIGITL$\neq$0 AND FRPS $\neq$-1 AND IFREQ=0 IFREQ = TAN (.1*DELT*.5) FFREQ = TAN (.9*3.14*.5) IF DIGITL$\neq$0 AND FRPS $\neq$-1 AND IFREQ$\neq$0 IFREQ= TAN (IFREQ*DELT*.5) FFREQ = TAN (FFREQ*DELT*.5)
M	CODE FOR MODIFIED Z-TRANSFER FUNCTION COMPUTATION FOR SAMPLED-DATA SYSTEMS. M IS THE FRACTIONAL SAMPLE PERIOD DELAY AND IS IN THE RANGE $0. \le M \le 1.$ M = 1. GIVES THE STANDARD Z-TRANSFORM IF THE SIGNAL HAS NO JUMP DISCONTINUITY AT THE SAMPLE INSTANT. M = 0. GIVES THE Z-TRANSFORM WITH A ONE SAMPLE PERIOD DELAY. HOWEVER, NUMERICAL ERRORS LIMIT M TO $M \ge .2$. THEREFORE, IF M=0., THE PROGRAM DEFAULTS TO STANDARD Z-TRANSFORM ANALYSIS. ONLY OPEN LOOP CALCULATIONS (MODIFIED Z-TRANSFER FUNCTIONS AND FREQUENCY RESPONSES) MAY BE PERFORMED WITH THIS OPTION.
GAIN1,GAIN2	ROOT LOCUS GAIN INCREMENTS FOR THE TWO FEEDBACK GAIN VARIABLES ALLOWED WITH THE ROOT LOCUS OPTION. IF NOT SET, PROGRAM DEFAULTS TO GAIN1= 1.0, GAIN2= 1.0.

C.3

APPENDIX 3

INPUT AND OUTPUT LISTINGS OF EXAMPLE PROBLEM

A. INPUT LISTING

```
   EXAMPLE PROBLEM   LATERAL-DIRECTIONAL AIRPLANE & CONTROL SYSTEM
$CODE READ=1,MIXED=1,SYSTEM=1,OUTPUT=2,NX=4,NY=5,NU=2,NSCALE=0,SAV=0,
CMAT=1,IPT=0,FRPS=1,IFLAG=0,READ3=0,DELT=.05,FINALT=3.,
TRESP=1,FORM=C,IFREQ=.1,FFREQ=20.,DELFRQ=1.11,$END
ROLLRATE YAWRATE   BETA      PHI       AY
DELTAAC DELTARC
         4         4
-5.9      1.7       -15.
-.4       -1.       10.
-.004     -1.       -.25      .11
1.
         4         2
14.       3.
-.6       -6.
          .07

         4         4
1.        -.02
-.02      1.
                    1.
                              1.
         5         4
1.
          1.
                    1.
                              1.
-.0348    8.7                 -1.
         5         4



                    8.7
    5    1
 1  1                    3      -2.
 2  4                    1      1.         25.
 3  8    4    5          2      1.         25.       .7
 4  6                    4       1.        1.
 5  5                    5      -.1       10.       2.
    1    2    3    4     5
    3    4
    3    2    1
    1    3
    2    4
    5    5
    2    1
    3    2
    6    3
          1.
```

B. OUTPUT LISTING

```
EXAMPLE PROBLEM  LATERAL-DIRECTIONAL AIRPLANE AND CONTROL SYSTEM
        CONTINUOUS SYSTEM
        MIXED OPTION
        OPEN LOOP
        LOAD ROUTINE INPUT
        TRANSFER FUNCTIONS
        FREQUENCY RESPONSES
        TRANSIENT RESPONSES


        NX  =   4           READ   =   1         TRESP  =   1         CMAT   =   1         DELT   =   .050
        NY  =   5           SYSTEM =   1         FRPS   =   1         NK2    =   0         FINALT =  3.000
        NU  =   2           MIXED  =   1         NUMERS =   0         IFLAG  =   0         IFREQ  =   .100
        NXC =   0           OUTPUT =   2         FORM   =   0         IGC    =   0         DELFRQ =  1.110
        NUC =   0           DIGITL =   0         CONTUP =   0         READ3  =   0         FFREQ  = 20.000
        ZOH =   0           IPT    =   0         MULTRT =   0         SAV    =   0         GAIN1  =  0.000
        N1  =   0           KOUNT  =   1         MODEL  =   0         NSCALE =   0         GAIN2  =  0.000
        N2  =   0                                                                          M      =  0.000


        THE A MATRIX IS

       4          4

 -.5800E+01   .1700E+01  -.1500E+02 -0.
 -.4000E+00  -.1000E+01   .1000E+02 -0.
 -.4000E-02  -.1000E+01  -.2500E+00   .1100E+00
  .1000E+01 -0.          -0.          -0.


        THE B MATRIX IS

       4          2

  .1400E+02   .3000E+01
 -.6000E+00  -.6000E+01
-0.           .7000E-01
-0.          -0.


        THE C MATRIX IS

       4          4

  .1000E+01  -.2000E-01 -0.          -0.
 -.2000E-01   .1000E+01 -0.          -0.
-0.          -0.          .1000E+01 -0.
-0.          -0.         -0.           .1000E+01


        THE H MATRIX IS

       5          4

  .1000E+01 -0.          -0.          -0.
-0.           .1000E+01 -0.          -0.
-0.          -0.          .1000E+01 -0.
-0.          -0.         -0.           .1000E+01
 -.3480E-01   .8700E+01 -0.          -.1000E+01
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
          THE G MATRIX IS

         5           4

-0.          -0.          -0.          -0.
-0.          -0.          -0.          -0.
-0.          -0.          -0.          -0.
-0.          -0.          -0.          -0.
-0.          -0.          -0.            .8700E+01

BLOCK DIAGRAM INPUT PARAMETERS

NO. TYPE         CONNEC      MOD                     PARAM

   1    1   -0   -0   -0    3   -0   -2.0000   -0.0000   -0.0000   -0.0000   -0.0000
   2    4   -0   -0   -0    1   -0    1.0000   25.0000   -0.0000   -0.0000   -0.0000
   3    8    4    5   -0    2   -0    1.0000   25.0000     .7000   -0.0000   -0.0000
   4    6   -0   -0   -0    4   -0    1.0000    1.0000   -0.0000   -0.0000   -0.0000
   5    5   -0   -0   -0    5   -0   -1.0000   10.0000    2.0000   -0.0000   -0.0000

          ITHINY

   1    2    3    4    5   -0   -0   -0   -0   -0

          ITHINU

   3    4   -0   -0   -0   -0   -0

          YTOV

   1    3
   2    4
   5    5

          ZTOU

   2    1
   3    2

          YZTOK

   5    3
```

```
          THE FINAL REDUCED SYSTEM IS


        THE A MATRIX IS

          9           9

-.5910E+01   .1681E+01  -.1441E+02  0.           .3498E+03   .1801E+04  0.          0.          0.
-.5183E+00  -.9664E+00   .9704E+01  0.          -.8003E+01  -.7714E+04  0.          0.          0.
-.4000E-02  -.1000E+01  -.2500E+00   .1100E+00  0.           .4375E+02  0.          0.          0.
 .1000E+01  0.          0.          0.          0.          0.          0.          0.          0.
-.2000E+01  0.          0.          0.          -.2500E+02  0.          0.          0.          0.
0.          0.          0.          0.          0.          0.           .1000E+01  0.          0.
-.4333E+02  -.4250E+02  0.           .5000E+01  0.          -.6250E+03  -.3500E+02  -.1000E+01   .4000E+02
0.           .1000E+01  0.          0.          0.          0.          0.          -.1000E+01  0.
 .8665E+01   .9700E+01  0.          -.1000E+01  0.          0.          0.          0.          -.1000E+02

        THE B MATRIX IS

          9           2

0.          0.
0.          0.
0.          0.
0.          0.
 .1000E+01  0.
0.          0.
0.           .1000E+01
0.          0.
0.          0.

        THE H MATRIX IS

          5           9

 .1000E+01  0.          0.          0.          0.          0.          0.          0.          0.
0.           .1000E+01  0.          0.          0.          0.          0.          0.          0.
0.          0.           .1000E+01  0.          0.          0.          0.          0.          0.
0.          0.          0.           .1000E+01  0.          0.          0.          0.          0.
 .8665E+01   .8700E+01  0.          -.1000E+01  0.          0.          0.          0.          0.

        THE F MATRIX IS

          5           2

0.          0.
0.          0.
0.          0.
0.          0.
0.          0.
```

```
          THE EIGEN VALUES OF THE SYSTEM ARE

                    REAL PART                  IMAGINARY PART


                -.38510438E+02               -.34711706E+02
                -.38510438E+02                .34711706E+02
                -.14860487E+02               -.36751121E+02
                -.14860487E+02                .36751121E+02
                 .31861414E+02               0.
                -.13418070E+01               0.
                -.16901737E+01               0.
                -.22354784E+00               0.
                 .92094333E-02               0.

          THE COEFFICIENTS OF THE CHARACTERISTIC EQUATION ORDERED FROM THE LOWEST POWER OF S


      .62837635E+06
     -.64570633E+08
     -.39345278E+09
     -.44325784E+09
     -.14167525E+09
     -.21935520E+07
      .27156306E+04
      .33936257E+04
      .78126751E+02
      .10000000E+01

          THE  ROLL RATE/ DELTA AC  NUMERATOR GAIN IS   .3498E+03

  THE ZEROES OF THE TRANSFER FUNCTION ARE

                    REAL PART                  IMAGINARY PART


                -.40837930E+02               -.48392089E+02
                -.40837930E+02                .48392089E+02
                 .37439375E+02               0.
                -.12398563E+01               0.
                -.17439393E+01               0.
                 .48342766E-01               0.
                 .53467247E-11               0.

            THE COEFFICIENTS OF THE NUMERATOR  POLYNOMIAL ORDERED FROM THE LOWEST POWER OF S


     -.84165208E-07
      .15745192E+05
     -.30410208E+06
     -.43963165E+06
     -.14722565E+06
      .10837823E+04
      .47177938E+02
      .10000000E+01

      ROLL RATE/ DELTA AC FREQUENCY RESPONSE  S-PLANE

  FREQUENCY         AMPLITUDE RATIO          PHASE ANGLE
   RAD/SEC              DB                  DEGREES

     .1000          -.15460E+02            .86549E+02
     .1110          -.14868E+02            .82507E+02
     .1232          -.14283E+02            .78420E+02
     .1368          -.13710E+02            .74304E+02
     .1518          -.13156E+02            .70173E+02
```

```
     .1685            -.12628E+02          .66047E+02
     .1870            -.12131E+02          .61946E+02
     .2076            -.11658E+02          .57894E+02
     .2305            -.11243E+02          .53915E+02
     .2558            -.10857E+02          .50038E+02
     .2839            -.10510E+02          .46286E+02
     .3152            -.10202E+02          .42681E+02
     .3498            -.99311E+01          .39244E+02
     .3883            -.96946E+01          .35989E+02
     .4310            -.94893E+01          .32925E+02
     .4785            -.93118E+01          .30057E+02
     .5311            -.91586E+01          .27384E+02
     .5895            -.90264E+01          .24902E+02
     .6544            -.89120E+01          .22603E+02
     .7263            -.88124E+01          .20474E+02
     .8062            -.87254E+01          .18504E+02
     .8949            -.86488E+01          .16679E+02
     .9934            -.85811E+01          .14982E+02
    1.1026            -.85211E+01          .13401E+02
    1.2239            -.84677E+01          .11921E+02
    1.3585            -.84205E+01          .10531E+02
    1.5080            -.83788E+01          .92195E+01
    1.6739            -.83422E+01          .79763E+01
    1.8580            -.83104E+01          .67930E+01
    2.0624            -.82828E+01          .56612E+01
    2.2892            -.82590E+01          .45732E+01
    2.5410            -.82383E+01          .35210E+01
    2.8206            -.82203E+01          .24961E+01
    3.1308            -.82042E+01          .14899E+01
    3.4752            -.81896E+01          .49278E+00
    3.8575            -.81757E+01         -.50526E+00
    4.2818            -.81619E+01         -.15151E+01
    4.7528            -.81477E+01         -.25485E+01
    5.2756            -.81324E+01         -.36183E+01
    5.8559            -.81153E+01         -.47387E+01
    6.5001            -.80955E+01         -.59255E+01
    7.2151            -.80722E+01         -.71969E+01
    8.0088            -.80443E+01         -.85739E+01
    8.8897            -.80105E+01         -.10081E+02
    9.8676            -.79691E+01         -.11750E+02
   10.9530            -.79182E+01         -.13616E+02
   12.1579            -.78556E+01         -.15726E+02
   13.4952            -.77713E+01         -.18139E+02
   14.9797            -.76832E+01         -.20932E+02
   16.6275            -.75667E+01         -.24207E+02
   18.4565            -.74255E+01         -.28099E+02
   20.4867            -.72589E+01         -.32793E+02

          THE  YAW RATE / DELTA AC  NUMERATOR GAIN IS  -.8003E+01

THE ZEROES OF THE TRANSFER FUNCTION ARE

                        REAL PART                IMAGINARY PART


                      -.11898556E+03            -.16589430E+03
                      -.11898556E+03             .16589430E+03
                       .16616361E+03            0.
                      -.10000000E+01            0.
                      -.19927576E+01            0.
                      -.61133543E-02            -.65198013E-01
                      -.61133543E-02             .65198013E-01
```

```
          THE COEFFICIENTS OF THE NUMERATOR  POLYNOMIAL ORDERED FROM THE LOWEST POWER OF S

     -.59179687E+05
     -.25759641E+06
     -.14083768E+08
     -.20806513E+08
     -.59188814E+07
      .23541531E+04
      .74812500E+02
      .10000000E+01

     YAW RATE / DELTA AC FREQUENCY RESPONSE  S-PLANE

 FREQUENCY           AMPLITUDE RATIO           PHASE ANGLE
  RAD/SEC                 DB                  DEGREES

      .1000           -.21300E+02           -.13061E+03
      .1110           -.19472E+02           -.12982E+03
      .1232           -.17959E+02           -.12999E+03
      .1368           -.16674E+02           -.13080E+03
      .1518           -.15566E+02           -.13205E+03
      .1685           -.14600E+02           -.13364E+03
      .1870           -.13754E+02           -.13548E+03
      .2076           -.13010E+02           -.13749E+03
      .2305           -.12357E+02           -.13963E+03
      .2558           -.11783E+02           -.14183E+03
      .2839           -.11281E+02           -.14405E+03
      .3152           -.10842E+02           -.14626E+03
      .3498           -.10458E+02           -.14841E+03
      .3883           -.10121E+02           -.15050E+03
      .4310           -.98262E+01           -.15249E+03
      .4785           -.95655E+01           -.15438E+03
      .5311           -.93335E+01           -.15616E+03
      .5895           -.91250E+01           -.15785E+03
      .6544           -.89360E+01           -.15944E+03
      .7263           -.87630E+01           -.16095E+03
      .8062           -.86039E+01           -.16240E+03
      .8949           -.84573E+01           -.16380E+03
      .9934           -.83228E+01           -.16516E+03
     1.1026           -.82006E+01           -.16649E+03
     1.2239           -.80913E+01           -.16780E+03
     1.3585           -.79956E+01           -.16908E+03
     1.5080           -.79139E+01           -.17034E+03
     1.6739           -.78463E+01           -.17156E+03
     1.8580           -.77923E+01           -.17274E+03
     2.0624           -.77510E+01           -.17388E+03
     2.2892           -.77209E+01           -.17497E+03
     2.5410           -.77004E+01           -.17601E+03
     2.8206           -.76876E+01           -.17700E+03
     3.1308           -.76808E+01           -.17795E+03
     3.4752           -.76785E+01           -.17886E+03
     3.8575           -.76795E+01           -.17975E+03
     4.2818           -.76826E+01           -.18059E+03
     4.7528           -.76873E+01           -.18147E+03
     5.2756           -.76929E+01           -.18236E+03
     5.8559           -.76991E+01           -.18328E+03
     6.5001           -.77058E+01           -.18425E+03
     7.2151           -.77129E+01           -.18530E+03
     8.0088           -.77202E+01           -.18643E+03
     8.8897           -.77278E+01           -.18770E+03
     9.8676           -.77355E+01           -.18912E+03
    10.9530           -.77433E+01           -.19074E+03
    12.1579           -.77509E+01           -.19262E+03
    13.4952           -.77580E+01           -.19484E+03
    14.9797           -.77640E+01           -.19750E+03
```

```
   16.6275         -.77689E+01          -.20070E+03
   18.4565         -.77730E+01          -.20465E+03
   20.4867         -.77790E+01          -.20955E+03

         THE    BETA     / DELTA AC   NUMERATOR GAIN IS    .6604E+01

THE ZEROES OF THE TRANSFER FUNCTION ARE

                    REAL PART                 IMAGINARY PART


                 -.30601356E+03              0.
                  .31081002E+03              0.
                 -.88070229E+02              0.
                 -.97666335E+00              0.
                 -.20036076E+01              0.
                  .84457968E-02              0.

          THE COEFFICIENTS OF THE NUMERATOR  POLYNOMIAL ORDERED FROM THE LOWEST POWER OF S


      .13896621E+06
     -.15160948E+08
     -.25077612E+08
     -.86602900E+07
     -.35285104E+05
      .86245561E+02
      .10000000E+01

      BETA     / DELTA AC FREQUENCY RESPONSE  S-PLANE

FREQUENCY          AMPLITUDE RATIO          PHASE ANGLE
 RAD/SEC                DB                 DEGREES

    .1000           .32188E+01          -.23476E+02
    .1110           .30589E+01          -.25631E+02
    .1232           .28696E+01          -.27927E+02
    .1368           .26474E+01          -.30356E+02
    .1518           .23889E+01          -.32901E+02
    .1685           .20911E+01          -.35542E+02
    .1870           .17515E+01          -.38253E+02
    .2076           .13684E+01          -.41006E+02
    .2305           .94098E+00          -.43769E+02
    .2558           .46964E+00          -.46510E+02
    .2839          -.44257E-01          -.49198E+02
    .3152          -.59833E+00          -.51805E+02
    .3498          -.11894E+01          -.54308E+02
    .3883          -.18135E+01          -.56698E+02
    .4310          -.24666E+01          -.58936E+02
    .4785          -.31443E+01          -.61046E+02
    .5311          -.38426E+01          -.63021E+02
    .5895          -.45578E+01          -.64869E+02
    .6544          -.52868E+01          -.66601E+02
    .7263          -.60275E+01          -.68235E+02
    .8062          -.67765E+01          -.69785E+02
    .8949          -.75392E+01          -.71269E+02
    .9934          -.83099E+01          -.72701E+02
   1.1026          -.90915E+01          -.74091E+02
   1.2239          -.98851E+01          -.75443E+02
   1.3585          -.10692E+02          -.76758E+02
   1.5080          -.11513E+02          -.78032E+02
   1.6739          -.12348E+02          -.79257E+02
   1.8580          -.13197E+02          -.80425E+02
```

```
    2.0624          -.14059E+02          -.81527E+02
    2.2892          -.14933E+02          -.82558E+02
    2.5410          -.15816E+02          -.83515E+02
    2.8206          -.16708E+02          -.84397E+02
    3.1306          -.17606E+02          -.85208E+02
    3.4752          -.18508E+02          -.85954E+02
    3.8575          -.19414E+02          -.86643E+02
    4.2818          -.20321E+02          -.87286E+02
    4.7528          -.21230E+02          -.87893E+02
    5.2756          -.22139E+02          -.88476E+02
    5.8559          -.23048E+02          -.89048E+02
    6.5001          -.23957E+02          -.89624E+02
    7.2151          -.24865E+02          -.90221E+02
    8.0088          -.25772E+02          -.90857E+02
    8.8897          -.26678E+02          -.91558E+02
    9.8676          -.27582E+02          -.92353E+02
   10.9530          -.28484E+02          -.93281E+02
   12.1579          -.29383E+02          -.94392E+02
   13.4952          -.30277E+02          -.95753E+02
   14.9797          -.31167E+02          -.97454E+02
   16.6275          -.32050E+02          -.99615E+02
   18.4565          -.32925E+02          -.10240E+03
   20.4867          -.33796E+02          -.10603E+03

          THE  PHI       / DELTA AC  NUMERATOR GAIN IS   .3498E+03

THE ZEROES OF THE TRANSFER FUNCTION ARE

                    REAL PART                IMAGINARY PART


                -.40837930E+02               -.48392089E+02
                -.40837930E+02                .48392089E+02
                 .37439375E+02               0.
                -.12398563E+01               0.
                -.17499393E+01               0.
                 .48342747E-01               0.

           THE COEFFICIENTS OF THE NUMERATOR  POLYNOMIAL ORDERED FROM THE LOWEST POWER OF S


      .15745187E+05
     -.30410210E+06
     -.43963166E+06
     -.14722565E+06
      .10837823E+04
      .47177938E+02
      .10000000E+01

      PHI        / DELTA AC FREQUENCY RESPONSE  S-PLANE

FREQUENCY         AMPLITUDE RATIO          PHASE ANGLE
 RAD/SEC                DB                DEGREES

     .1000          .45403E+01          -.34575E+01
     .1110          .42252E+01          -.74998E+01
     .1232          .39043E+01          -.11586E+02
     .1368          .35707E+01          -.15703E+02
     .1518          .32177E+01          -.19833E+02
     .1685          .28393E+01          -.23960E+02
     .1870          .24304E+01          -.28061E+02
     .2076          .19868E+01          -.32113E+02
     .2305          .15057E+01          -.36091E+02
```

```
    .2556            .98536E+00          -.39968E+02
    .2839            .42551E+00          -.43721E+02
    .3152           -.17309E+00          -.47325E+02
    .3499           -.80864E+00          -.50762E+02
    .3883           -.14785E+01          -.54018E+02
    .4310           -.21797E+01          -.57082E+02
    .4785           -.29087E+01          -.59950E+02
    .5311           -.36620E+01          -.62622E+02
    .5895           -.44362E+01          -.65104E+02
    .6544           -.52282E+01          -.67404E+02
    .7263           -.60352E+01          -.69532E+02
    .8062           -.68546E+01          -.71502E+02
    .8949           -.76845E+01          -.73328E+02
    .9934           -.85232E+01          -.75024E+02
   1.1026           -.93696E+01          -.76606E+02
   1.2239           -.10223E+02          -.78085E+02
   1.3585           -.11082E+02          -.79475E+02
   1.5080           -.11947E+02          -.80787E+02
   1.6739           -.12817E+02          -.82030E+02
   1.8580           -.13691E+02          -.83214E+02
   2.0624           -.14570E+02          -.84345E+02
   2.2892           -.15453E+02          -.85433E+02
   2.5410           -.16339E+02          -.86486E+02
   2.8206           -.17227E+02          -.87510E+02
   3.1308           -.18117E+02          -.88517E+02
   3.4752           -.19009E+02          -.89514E+02
   3.8575           -.19902E+02          -.90512E+02
   4.2818           -.20794E+02          -.91522E+02
   4.7528           -.21687E+02          -.92555E+02
   5.2756           -.22578E+02          -.93625E+02
   5.8559           -.23467E+02          -.94745E+02
   6.5001           -.24354E+02          -.95932E+02
   7.2151           -.25237E+02          -.97204E+02
   8.0088           -.26116E+02          -.98580E+02
   8.8897           -.26988E+02          -.10009E+03
   9.8676           -.27853E+02          -.10176E+03
  10.9530           -.28709E+02          -.10362E+03
  12.1579           -.29553E+02          -.10573E+03
  13.4952           -.30382E+02          -.10815E+03
  14.9797           -.31193E+02          -.11094E+03
  16.6275           -.31983E+02          -.11421E+03
  18.4565           -.32749E+02          -.11811E+03
  20.4867           -.33487E+02          -.12280E+03


        THE  AY        / DELTA AC  NUMERATOR GAIN IS   .2962E+04

THE ZEROES OF THE TRANSFER FUNCTION ARE

                      REAL PART                 IMAGINARY PART


                 -.12023271E+02                -.15019465E+02
                 -.12023271E+02                 .15019465E+02
                 -.10962181E+02                0.
                 -.70471725E+00                -.97429378E+00
                 -.70471725E+00                 .97429378E+00
                  .79854692E-02                0.
                 -.10000000E+02                0.


         THE COEFFICIENTS OF THE NUMERATOR  POLYNOMIAL ORDERED FROM THE LOWEST POWER OF S


    -.46854623E+03
     .58090733E+05
```

```
      .71766408E+05
      .86354798E+05
      .11838388E+05
      .10483452E+04
      .46410172E+02
      .10000000E+01

     AY          / DELTA AC FREQUENCY RESPONSE   S-PLANE

 FREQUENCY            AMPLITUDE RATIO              PHASE ANGLE
  RAD/SEC                  DB                     DEGREES

     .1000              .72610E+01               -.20545E+03
     .1110              .70846E+01               -.20775E+03
     .1232              .68748E+01               -.21022E+03
     .1368              .66273E+01               -.21283E+03
     .1518              .63376E+01               -.21558E+03
     .1685              .60013E+01               -.21844E+03
     .1870              .56145E+01               -.22138E+03
     .2076              .51737E+01               -.22437E+03
     .2305              .46758E+01               -.22738E+03
     .2558              .41187E+01               -.23036E+03
     .2839              .35008E+01               -.23327E+03
     .3152              .28210E+01               -.23606E+03
     .3498              .20790E+01               -.23868E+03
     .3883              .12745E+01               -.24108E+03
     .4310              .40787E+00               -.24319E+03
     .4785             -.51997E+00               -.24496E+03
     .5311             -.15070E+01               -.24629E+03
     .5895             -.25488E+01               -.24708E+03
     .6544             -.36373E+01               -.24722E+03
     .7263             -.47574E+01               -.24658E+03
     .8062             -.58846E+01               -.24502E+03
     .8949             -.69816E+01               -.24244E+03
     .9934             -.80000E+01               -.23886E+03
    1.1026             -.88874E+01               -.23442E+03
    1.2239             -.96040E+01               -.22947E+03
    1.3585             -.10140E+02               -.22444E+03
    1.5080             -.10520E+02               -.21976E+03
    1.6739             -.10792E+02               -.21566E+03
    1.8580             -.11012E+02               -.21222E+03
    2.0624             -.11222E+02               -.20934E+03
    2.2892             -.11454E+02               -.20687E+03
    2.5410             -.11723E+02               -.20461E+03
    2.8206             -.12035E+02               -.20238E+03
    3.1308             -.12387E+02               -.20002E+03
    3.4752             -.12772E+02               -.19740E+03
    3.8575             -.13182E+02               -.19442E+03
    4.2818             -.13603E+02               -.19098E+03
    4.7528             -.14023E+02               -.18703E+03
    5.2756             -.14430E+02               -.18251E+03
    5.8559             -.14810E+02               -.17742E+03
    6.5001             -.15150E+02               -.17168E+03
    7.2151             -.15436E+02               -.16529E+03
    8.0088             -.15656E+02               -.15827E+03
    8.8897             -.15795E+02               -.15059E+03
    9.8676             -.15838E+02               -.14227E+03
   10.9530             -.15765E+02               -.13331E+03
   12.1579             -.15552E+02               -.12375E+03
   13.4952             -.15169E+02               -.11365E+03
   14.9797             -.14577E+02               -.10317E+03
   16.6275             -.13734E+02               -.92598E+02
   18.4565             -.12605E+02               -.82409E+02
   20.4867             -.11175E+02               -.73240E+02
```

```
        THE  ROLL RATE/DELTA RC   NUMERATOR GAIN IS   .1801E+04

THE ZEROES OF THE TRANSFER FUNCTION ARE

                   REAL PART                  IMAGINARY PART


                -.10000000E+01                 0.
                -.25000000E+02                 0.
                 .61123953E+01                 0.
                -.10000000E+02                 0.
                -.35026731E+01                 0.
                -.14030276E-12                 0.


        THE COEFFICIENTS OF THE NUMERATOR  POLYNOMIAL ORDERED FROM THE LOWEST POWER OF S


    -.75096080E-09
    -.53524306E+04
    -.67542014E+04
    -.12645209E+04
     .16964028E+03
     .33390278E+02
     .10000000E+01


    ROLL RATE/DELTA RC  FREQUENCY RESPONSE  S-PLANE

FREQUENCY        AMPLITUDE RATIO         PHASE ANGLE
 RAD/SEC              DB                DEGREES

    .1000         -.17819E+02          -.29895E+02
    .1110         -.17975E+02          -.31742E+02
    .1232         -.18162E+02          -.33797E+02
    .1368         -.18383E+02          -.36047E+02
    .1518         -.18642E+02          -.38473E+02
    .1685         -.18941E+02          -.41056E+02
    .1870         -.19284E+02          -.43767E+02
    .2076         -.19672E+02          -.46577E+02
    .2305         -.20105E+02          -.49456E+02
    .2558         -.20585E+02          -.52371E+02
    .2839         -.21109E+02          -.55293E+02
    .3152         -.21676E+02          -.58193E+02
    .3498         -.22283E+02          -.61051E+02
    .3883         -.22927E+02          -.63849E+02
    .4310         -.23603E+02          -.66577E+02
    .4785         -.24310E+02          -.69232E+02
    .5311         -.25042E+02          -.71816E+02
    .5895         -.25798E+02          -.74337E+02
    .6544         -.26574E+02          -.76805E+02
    .7263         -.27371E+02          -.79232E+02
    .8062         -.28187E+02          -.81631E+02
    .8949         -.29022E+02          -.84010E+02
    .9934         -.29876E+02          -.86374E+02
   1.1026         -.30752E+02          -.88719E+02
   1.2239         -.31649E+02          -.91035E+02
   1.3585         -.32568E+02          -.93302E+02
   1.5080         -.33507E+02          -.95492E+02
   1.6739         -.34463E+02          -.97574E+02
   1.8580         -.35431E+02          -.99511E+02
   2.0624         -.36404E+02          -.10127E+03
   2.2892         -.37373E+02          -.10282E+03
   2.5410         -.38327E+02          -.10413E+03
   2.8206         -.39253E+02          -.10520E+03
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
    3.1308              -.40140E+02            -.10609E+03
    3.4752              -.40974E+02            -.10653E+03
    3.8575              -.41742E+02            -.10680E+03
    4.2818              -.42433E+02            -.10682E+03
    4.7528              -.43036E+02            -.10658E+03
    5.2756              -.43542E+02            -.10611E+03
    5.8559              -.43942E+02            -.10541E+03
    6.5001              -.44232E+02            -.10449E+03
    7.2151              -.44407E+02            -.10337E+03
    8.0088              -.44463E+02            -.10208E+03
    8.8897              -.44409E+02            -.10063E+03
    9.8676              -.44217E+02            -.99091E+02
   10.9530              -.43915E+02            -.97508E+02
   12.1579              -.43498E+02            -.95962E+02
   13.4952              -.42966E+02            -.94555E+02
   14.9797              -.42324E+02            -.93413E+02
   16.6275              -.41575E+02            -.92700E+02
   18.4565              -.40725E+02            -.92624E+02
   20.4867              -.39780E+02            -.93459E+02

          THE  YAW RATE /DELTA RC   NUMERATOR GAIN IS  -.3714E+04

THE ZEROES OF THE TRANSFER FUNCTION ARE

                    REAL PART                IMAGINARY PART


                -.10000000E+01             0.
                -.15581129E+02            -.24557545E+02
                -.15581129E+02             .24557545E+02
                -.10000000E+02             0.
                -.67523717E-01            -.16816906E+00
                -.67523717E-01             .16816906E+00

           THE COEFFICIENTS OF THE NUMERATOR  POLYNOMIAL ORDERED FROM THE LOWEST POWER OF S


       .27777778E+03
       .14580808E+04
       .97964141E+04
       .97791700E+04
       .12043562E+04
       .42297306E+02
       .10000000E+01

       YAW RATE /DELTA RC  FREQUENCY RESPONSE  S-PLANE

FREQUENCY          AMPLITUDE RATIO          PHASE ANGLE
 RAD/SEC               DB                  DEFREES

     .1000           -.19086E+02           -.90021E+02
     .1110           -.20524E+02           -.86391E+02
     .1232           -.22021E+02           -.81386E+02
     .1368           -.23517E+02           -.74458E+02
     .1518           -.24888E+02           -.65099E+02
     .1685           -.25920E+02           -.53317E+02
     .1870           -.26397E+02           -.40252E+02
     .2076           -.26274E+02           -.27954E+02
     .2305           -.25723E+02           -.18026E+02
     .2558           -.24978E+02           -.10859E+02
     .2839           -.24205E+02           -.60623E+01
     .3152           -.23485E+02           -.30594E+01
     .3498           -.22848E+02           -.13528E+01
```

```
     .3883            -.22302E+02          -.57444E+00
     .4316            -.21841E+02          -.46687E+00
     .4785            -.21457E+02          -.85425E+00
     .5311            -.21140E+02          -.16181E+01
     .5895            -.20883E+02          -.26800E+01
     .6544            -.20680E+02          -.39895E+01
     .7263            -.20526E+02          -.55104E+01
     .8062            -.20418E+02          -.72233E+01
     .8949            -.20356E+02          -.91100E+01
     .9934            -.20341E+02          -.11153E+02
    1.1026            -.20374E+02          -.13332E+02
    1.2239            -.20457E+02          -.15620E+02
    1.3585            -.20595E+02          -.17992E+02
    1.5080            -.20788E+02          -.20377E+02
    1.6739            -.21039E+02          -.22753E+02
    1.8580            -.21347E+02          -.25057E+02
    2.0624            -.21712E+02          -.27234E+02
    2.2892            -.22130E+02          -.29228E+02
    2.5410            -.22597E+02          -.30986E+02
    2.8206            -.23108E+02          -.32463E+02
    3.1308            -.23657E+02          -.33620E+02
    3.4752            -.24237E+02          -.34423E+02
    3.8575            -.24839E+02          -.34851E+02
    4.2818            -.25457E+02          -.34885E+02
    4.7528            -.26083E+02          -.34518E+02
    5.2756            -.26710E+02          -.33748E+02
    5.8559            -.27331E+02          -.32580E+02
    6.5001            -.27941E+02          -.31025E+02
    7.2151            -.28533E+02          -.29099E+02
    8.0088            -.29104E+02          -.26822E+02
    8.8897            -.29649E+02          -.24217E+02
    9.8676            -.30168E+02          -.21307E+02
   10.9530            -.30657E+02          -.18114E+02
   12.1579            -.31115E+02          -.14655E+02
   13.4952            -.31540E+02          -.10941E+02
   14.9797            -.31927E+02          -.69774E+01
   16.6275            -.32265E+02          -.27709E+01
   18.4565            -.32535E+02           .16525E+01
   20.4867            -.32705E+02           .62035E+01

          THE    BETA    /DELTA RC    NUMERATOR GAIN IS    .4375E+02

THE ZEROES OF THE TRANSFER FUNCTION ARE

                    REAL PART                  IMAGINARY PART


                 -.10000000E+01             0.
                 -.95276345E+02             0.
                 -.15664379E+02             -.24652880E+02
                 -.15664379E+02              .24652880E+02
                 -.10000000E+02             0.
                  .38894055E-02             0.

          THE COEFFICIENTS OF THE NUMERATOR  POLYNOMIAL ORDERED FROM THE LOWEST POWER OF S


   -.28297033E+04
    .72429157E+06
    .83510518E+06
    .11267429E+06
    .48169837E+04
    .12760321E+03
    .10000000E+01
```

```
       BETA      /DELTA RC   FREQUENCY RESPONSE   S-PLANE

FREQUENCY        AMPLITUDE RATIO           PHASE ANGLE
 RAD/SEC              DB                  DEGREES

    .1000          -.74410E+01           -.20830E+03
    .1110          -.75992E+01           -.21044E+03
    .1232          -.77884E+01           -.21278E+03
    .1368          -.80119E+01           -.21529E+03
    .1518          -.82734E+01           -.21798E+03
    .1685          -.85758E+01           -.22081E+03
    .1870          -.89220E+01           -.22378E+03
    .2076          -.93139E+01           -.22684E+03
    .2305          -.97526E+01           -.22997E+03
    .2558          -.10238E+02           -.23314E+03
    .2839          -.10770E+02           -.23633E+03
    .3152          -.11346E+02           -.23951E+03
    .3498          -.11964E+02           -.24265E+03
    .3883          -.12622E+02           -.24576E+03
    .4310          -.13315E+02           -.24882E+03
    .4785          -.14042E+02           -.25182E+03
    .5311          -.14799E+02           -.25478E+03
    .5895          -.15586E+02           -.25770E+03
    .6544          -.16401E+02           -.26061E+03
    .7263          -.17243E+02           -.26350E+03
    .8062          -.18116E+02           -.26640E+03
    .8949          -.19020E+02           -.26932E+03
    .9934          -.19959E+02            .87744E+02
   1.1026          -.20938E+02            .84785E+02
   1.2239          -.21960E+02            .81819E+02
   1.3585          -.23029E+02            .78871E+02
   1.5080          -.24150E+02            .75973E+02
   1.6739          -.25323E+02            .73167E+02
   1.8580          -.26552E+02            .70502E+02
   2.0624          -.27834E+02            .68023E+02
   2.2892          -.29167E+02            .65796E+02
   2.5410          -.30547E+02            .63855E+02
   2.8206          -.31970E+02            .62250E+02
   3.1308          -.33429E+02            .61017E+02
   3.4752          -.34917E+02            .60183E+02
   3.8575          -.36428E+02            .59786E+02
   4.2818          -.37952E+02            .59827E+02
   4.7528          -.39484E+02            .60321E+02
   5.2756          -.41016E+02            .61270E+02
   5.8559          -.42541E+02            .62671E+02
   6.5001          -.44053E+02            .64516E+02
   7.2151          -.45546E+02            .66791E+02
   8.0088          -.47017E+02            .69479E+02
   8.8897          -.48460E+02            .72562E+02
   9.8676          -.49874E+02            .76020E+02
  10.9530          -.51256E+02            .79836E+02
  12.1579          -.52604E+02            .83999E+02
  13.4952          -.53915E+02            .88504E+02
  14.9797          -.55183E+02           -.26665E+03
  16.6275          -.56398E+02           -.26146E+03
  18.4565          -.57538E+02           -.25595E+03
  20.4867          -.58573E+02           -.25020E+03

          THE   PHI       /DELTA RC   NUMERATOR  GAIN  IS    .1801E+04

THE ZEROES OF THE TRANSFER FUNCTION ARE
```

```
                   REAL PART                  IMAGINARY PART

                 -.10000000E+01                 0.
                  .61123953E+01                 0.
                 -.25000000E+02                 0.
                 -.35026731E+01                 0.
                 -.10000000E+02                 0.

       THE COEFFICIENTS OF THE NUMERATOR  POLYNOMIAL ORDERED FROM THE LOWEST POWER OF S

  -.53524306E+04
  -.57542014E+04
  -.12645208E+04
   .16964028E+03
   .33390278E+02
   .10000000E+01

      PHI       /DELTA RC   FREQUENCY RESPONSE   S-PLANE

  FREQUENCY        AMPLITUDE RATIO          PHASE ANGLE
   RAD/SEC               DB                 DEGREES

      .1000          .21805E+01           -.11990E+03
      .1110          .11181E+01           -.12175E+03
      .1232          .24909E-01           -.12380E+03
      .1368         -.11026E+01           -.12605E+03
      .1518         -.22677E+01           -.12848E+03
      .1685         -.34736E+01           -.13106E+03
      .1870         -.47226E+01           -.13377E+03
      .2076         -.60167E+01           -.13658E+03
      .2305         -.73568E+01           -.13946E+03
      .2558         -.87427E+01           -.14238E+03
      .2839         -.10173E+02           -.14530E+03
      .3152         -.11647E+02           -.14820E+03
      .3498         -.13160E+02           -.15106E+03
      .3883         -.14711E+02           -.15386E+03
      .4310         -.16294E+02           -.15658E+03
      .4785         -.17906E+02           -.15924E+03
      .5311         -.19545E+02           -.16182E+03
      .5895         -.21207E+02           -.16434E+03
      .6544         -.22891E+02           -.16681E+03
      .7263         -.24594E+02           -.16924E+03
      .8062         -.26316E+02           -.17164E+03
      .8949         -.28057E+02           -.17402E+03
      .9934         -.29819E+02           -.17638E+03
     1.1026         -.31601E+02           -.17873E+03
     1.2239         -.33404E+02           -.18102E+03
     1.3585         -.35230E+02           -.18328E+03
     1.5080         -.37075E+02           -.18547E+03
     1.6739         -.38938E+02           -.18755E+03
     1.8580         -.40812E+02           -.18949E+03
     2.0624         -.42692E+02           -.19125E+03
     2.2892         -.44567E+02           -.19280E+03
     2.5410         -.46427E+02           -.19411E+03
     2.8206         -.48260E+02           -.19518E+03
     3.1308         -.50053E+02           -.19598E+03
     3.4752         -.51793E+02           -.19651E+03
     3.8575         -.53468E+02           -.19678E+03
     4.2818         -.55065E+02           -.19680E+03
     4.7528         -.56575E+02           -.19656E+03
     5.2756         -.57987E+02           -.19609E+03
     5.8559         -.59294E+02           -.19539E+03
     6.5001         -.60491E+02           -.19447E+03
```

```
      7.2151            -.61571E+02            -.19335E+03
      8.0098            -.62534E+02            -.19206E+03
      8.8897            -.63377E+02            -.19061E+03
      9.8676            -.64101E+02            -.18907E+03
     10.9530            -.64706E+02            -.18749E+03
     12.1579            -.65195E+02            -.18594E+03
     13.4952            -.65570E+02            -.18453E+03
     14.9797            -.65834E+02            -.18339E+03
     16.6275            -.65992E+02            -.18268E+03
     18.4565            -.66048E+02            -.18260E+03
     20.4867            -.66009E+02            -.18344E+03

           THE  AY        /DELTA RC   NUMERATOR GAIN IS  -.1671E+05

THE ZEROES OF THE TRANSFER FUNCTION ARE

                      REAL PART                   IMAGINARY PART


                  -.10000000E+01                0.
                  -.19653147E+02                -.36396151E+02
                  -.19653147E+02                 .36396151E+02
                  -.42251286E+00                0.
                   .54973431E-02                0.
                  -.10000000E+02                0.

            THE COEFFICIENTS OF THE NUMERATOR  POLYNOMIAL ORDERED FROM THE LOWEST POWER OF S


     -.39667357E+02
      .70904524E+04
      .25116682E+05
      .20111111E+05
      .21742719E+04
      .50723320E+02
      .10000000E+01

      AY        /DELTA RC  FREQUENCY RESPONSE  S-PLANE

FREQUENCY        AMPLITUDE RATIO          PHASE ANGLE
 RAD/SEC              DB                DEGREES

    .1000          .43855E+01           -.14732E+02
    .1110          .42791E+01           -.15074E+02
    .1232          .41536E+01           -.15968E+02
    .1368          .40075E+01           -.16899E+02
    .1518          .38400E+01           -.17846E+02
    .1685          .36508E+01           -.18785E+02
    .1870          .34403E+01           -.19692E+02
    .2076          .32102E+01           -.20542E+02
    .2305          .29630E+01           -.21310E+02
    .2558          .27023E+01           -.21978E+02
    .2839          .24328E+01           -.22533E+02
    .3152          .21593E+01           -.22969E+02
    .3498          .18873E+01           -.23293E+02
    .3883          .16216E+01           -.23520E+02
    .4310          .13667E+01           -.23674E+02
    .4785          .11258E+01           -.23787E+02
    .5311          .90055E+00           -.23899E+02
    .5895          .69117E+00           -.24049E+02
    .6544          .49611E+00           -.24280E+02
    .7263          .31211E+00           -.24629E+02
    .8062          .13440E+00           -.25128E+02
```

```
     .8949          -.43125E-01         -.25803E+02
     .9934          -.22755E+00         -.26665E+02
    1.1026          -.42652E+00         -.27717E+02
    1.2239          -.64773E+00         -.28947E+02
    1.3585          -.83843E+00         -.30331E+02
    1.5080          -.11848E+01         -.31830E+02
    1.6739          -.15118E+01         -.33400E+02
    1.8580          -.18822E+01         -.34986E+02
    2.0624          -.22972E+01         -.36532E+02
    2.2892          -.27561E+01         -.37982E+02
    2.5410          -.32561E+01         -.39281E+02
    2.8205          -.37332E+01         -.40383E+02
    3.1308          -.43621E+01         -.41246E+02
    3.4752          -.49566E+01         -.41838E+02
    3.8575          -.55699E+01         -.42134E+02
    4.2818          -.61951E+01         -.42119E+02
    4.7528          -.68250E+01         -.41787E+02
    5.2756          -.74527E+01         -.41140E+02
    5.8559          -.80718E+01         -.40187E+02
    6.5001          -.86761E+01         -.38947E+02
    7.2151          -.92694E+01         -.37447E+02
    8.0088          -.98205E+01         -.35721E+02
    8.8897          -.10353E+02         -.33810E+02
    9.8676          -.10857E+02         -.31762E+02
   10.9530          -.11332E+02         -.29631E+02
   12.1579          -.11778E+02         -.27476E+02
   13.4952          -.12198E+02         -.25362E+02
   14.9797          -.12596E+02         -.23361E+02
   16.6275          -.12975E+02         -.21552E+02
   18.4565          -.13340E+02         -.20031E+02
   20.4867          -.13697E+02         -.18914E+02

  THE TRANSITION MATRIX     11  TERMS

       9          9

-.3585E+00 -.7306E+00 -.4430E+00  .9217E-01  .3458E+01  .6567E+02  .1036E+01 -.2057E-01  .7287E+00
 .1734E+01  .2963E+01  .3737E+00 -.2360E+00  .6289E+01 -.2055E+03 -.2887E+01  .5190E-01 -.1947E+01
-.4516E-01 -.9990E-01  .5730E+00  .1148E-01 -.1452E+00  .7171E+01  .8679E-01 -.1306E-02  .4717E-01
 .2428E-01 -.3584E-02 -.1412E-01  .1001E+01  .2046E+00  .1766E+01  .2086E-01 -.2898E-03  .1055E-01
-.1942E-01  .1528E-01  .1868E-01 -.2052E-02  .3643E-01 -.2376E+01 -.3029E-01  .4516E-03 -.1638E-01
-.2322E-01 -.2360E-01  .1553E-02  .3461E-02 -.1172E+00  .1706E+01  .2902E-01 -.7784E-03  .2666E-01
-.6350E+00 -.1314E+01  .5618E-01  .1186E+00 -.4957E+01  .5005E+02  .6899E+00 -.2824E-01  .8943E+00
 .2423E-01  .7467E-01  .1060E-01 -.3172E-02  .6630E-01 -.4648E+01 -.5180E-01  .9519E+00 -.2501E-01
 .3349E+00  .4729E+00 -.2312E-01 -.5424E-01  .1974E+01 -.2226E+02 -.2441E+00  .3181E-02  .4903E+00


  TIME         ROLL RATE    YAW RATE     BETA         PHI          AY           DELTA AC     DELTA RD
 0.            0.           0.           0.           0.           0.           -0.           .100E+01
  .500E-01      .209E-01    -.525E-01     .132E-02     .293E-03    -.276E+00    -0.           .100E+01
  .100E+00      .129E+00    -.415E+00     .150E-01     .346E-02    -.250E+01    -0.           .100E+01
  .150E+00      .619E+00    -.219E+01     .901E-01     .191E-01    -.137E+02    -0.           .100E+01
  .200E+00      .303E+01    -.110E+02     .468E+00     .950E-01    -.692E+02    -0.           .100E+01
  .250E+00      .149E+02    -.542E+02     .234E+01     .468E+00    -.343E+03    -0.           .100E+01
  .300E+00      .733E+02    -.267E+03     .115E+02     .230E+01    -.169E+04    -0.           .100E+01
  .350E+00      .361E+03    -.131E+04     .568E+02     .113E+02    -.830E+04    -0.           .100E+01
  .400E+00      .177E+04    -.645E+04     .280E+03     .557E+02    -.408E+05    -0.           .100E+01
  .450E+00      .873E+04    -.317E+05     .137E+04     .274E+03    -.201E+06    -0.           .100E+01
  .500E+00      .429E+05    -.156E+06     .676E+04     .135E+04    -.988E+06    -0.           .100E+01
  .550E+00      .211E+06    -.768E+06     .333E+05     .663E+04    -.486E+07    -0.           .100E+01
  .600E+00      .104E+07    -.378E+07     .164E+06     .326E+05    -.239E+08    -0.           .100E+01
  .650E+00      .511E+07    -.186E+08     .805E+06     .160E+06    -.118E+09    -0.           .100E+01
  .700E+00      .251E+08    -.914E+08     .396E+07     .789E+06    -.578E+09    -0.           .100E+01
  .750E+00      .124E+09    -.450E+09     .195E+08     .388E+07    -.284E+10    -0.           .100E+01
  .800E+00      .608E+09    -.221E+10     .958E+08     .191E+08    -.140E+11    -0.           .100E+01
  .850E+00      .299E+10    -.109E+11     .471E+09     .939E+08    -.688E+11    -0.           .100E+01
```

```
.900E+00    .147E+11    -.535E+11    .232E+10    .462E+09    -.339E+12   -0.          .100E+01
.950E+00    .724E+11    -.263E+12    .114E+11    .227E+10    -.167E+13   -0.          .100E+01
.100E+01    .356E+12    -.129E+13    .561E+11    .112E+11    -.819E+13   -0.          .100E+01
.105E+01    .175E+13    -.637E+13    .276E+12    .550E+11    -.403E+14   -0.          .100E+01
.110E+01    .861E+13    -.313E+14    .136E+13    .270E+12    -.198E+15   -0.          .100E+01
.115E+01    .424E+14    -.154E+15    .667E+13    .133E+13    -.975E+15   -0.          .100E+01
.120E+01    .208E+15    -.758E+15    .328E+14    .654E+13    -.480E+16   -0.          .100E+01
.125E+01    .102E+16    -.373E+16    .161E+15    .322E+14    -.236E+17   -0.          .100E+01
.130E+01    .504E+16    -.183E+17    .794E+15    .158E+15    -.116E+18   -0.          .100E+01
.135E+01    .248E+17    -.902E+17    .391E+16    .778E+15    -.571E+18   -0.          .100E+01
.140E+01    .122E+18    -.444E+18    .192E+17    .383E+16    -.281E+19   -0.          .100E+01
.145E+01    .600E+18    -.218E+19    .945E+17    .188E+17    -.138E+20   -0.          .100E+01
.150E+01    .295E+19    -.107E+20    .465E+18    .926E+17    -.679E+20   -0.          .100E+01
.155E+01    .145E+20    -.528E+20    .229E+19    .456E+18    -.334E+21   -0.          .100E+01
.160E+01    .714E+20    -.260E+21    .112E+20    .224E+19    -.164E+22   -0.          .100E+01
.165E+01    .351E+21    -.128E+22    .553E+20    .110E+20    -.808E+22   -0.          .100E+01
.170E+01    .173E+22    -.628E+22    .272E+21    .542E+20    -.398E+23   -0.          .100E+01
.175E+01    .850E+22    -.309E+23    .134E+22    .267E+21    -.196E+24   -0.          .100E+01
.180E+01    .418E+23    -.152E+24    .659E+22    .131E+22    -.962E+24   -0.          .100E+01
.185E+01    .206E+24    -.748E+24    .324E+23    .645E+22    -.473E+25   -0.          .100E+01
.190E+01    .101E+25    -.368E+25    .159E+24    .317E+23    -.233E+26   -0.          .100E+01
.195E+01    .497E+25    -.181E+26    .784E+24    .156E+24    -.114E+27   -0.          .100E+01
.200E+01    .245E+26    -.890E+26    .385E+25    .768E+24    -.563E+27   -0.          .100E+01
.205E+01    .120E+27    -.438E+27    .190E+26    .378E+25    -.277E+28   -0.          .100E+01
.210E+01    .592E+27    -.215E+28    .933E+26    .186E+26    -.136E+29   -0.          .100E+01
.215E+01    .291E+28    -.106E+29    .459E+27    .914E+26    -.670E+29   -0.          .100E+01
.220E+01    .143E+29    -.521E+29    .226E+28    .450E+27    -.330E+30   -0.          .100E+01
.225E+01    .705E+29    -.256E+30    .111E+29    .221E+28    -.162E+31   -0.          .100E+01
.230E+01    .347E+30    -.126E+31    .546E+29    .109E+29    -.798E+31   -0.          .100E+01
.235E+01    .170E+31    -.620E+31    .269E+30    .535E+29    -.392E+32   -0.          .100E+01
.240E+01    .839E+31    -.305E+32    .132E+31    .263E+30    -.193E+33   -0.          .100E+01
.245E+01    .412E+32    -.150E+33    .650E+31    .129E+31    -.949E+33   -0.          .100E+01
.250E+01    .203E+33    -.738E+33    .320E+32    .637E+31    -.467E+34   -0.          .100E+01
.255E+01    .998E+33    -.363E+34    .157E+33    .313E+32    -.230E+35   -0.          .100E+01
.260E+01    .491E+34    -.179E+35    .773E+33    .154E+33    -.113E+36   -0.          .100E+01
.265E+01    .241E+35    -.878E+35    .380E+34    .758E+33    -.556E+36   -0.          .100E+01
.270E+01    .119E+36    -.432E+36    .187E+35    .373E+34    -.273E+37   -0.          .100E+01
.275E+01    .584E+36    -.213E+37    .920E+35    .183E+35    -.134E+38   -0.          .100E+01
.280E+01    .287E+37    -.105E+38    .453E+36    .902E+35    -.661E+38   -0.          .100E+01
.285E+01    .141E+38    -.514E+38    .223E+37    .444E+36    -.325E+39   -0.          .100E+01
.290E+01    .695E+38    -.253E+39    .110E+38    .218E+37    -.160E+40   -0.          .100E+01
.295E+01    .342E+39    -.124E+40    .539E+38    .107E+38    -.787E+40   -0.          .100E+01
.300E+01    .168E+40    -.612E+40    .265E+39    .528E+38    -.387E+41   -0.          .100E+01
```

TABLE I. CONTROL PROGRAM ANALYSIS OPTIONS

```
CONTINUOUS SYSTEMS
   OPEN AND CLOSED LOOP SYSTEMS
        TRANSFER FUNCTIONS                                   (S-PLANE)
        FREQUENCY RESPONSES                                  (S-PLANE)
        POWER SPECTRA                                        (S-PLANE)
        TRANSIENT RESPONSES
   ROOT LOCUS                                                (S-PLANE)


DISCRETE SYSTEMS
  OPEN AND CLOSED LOOP SYSTEMS
        TRANSFER FUNCTIONS                                   (Z- OR W-PLANE)
        FREQUENCY RESPONSES                                  (W- OR S-PLANE)
        TRANSIENT RESPONSES
  ROOT LOCUS                                                 (Z-PLANE)


SAMPLED-DATA SYSTEMS
  OPEN LOOP SYSTEMS
        STANDARD OR MODIFIED Z-TRANSFER FUNCTIONS(Z- OR W-PLANE)
        FREQUENCY RESPONSES                                  (W- OR S-PLANE)
        TRANSIENT RESPONSES
  OPEN AND CLOSED-LOOP SYSTEMS
        STANDARD Z-TRANSFER FUNCTIONS                        (Z- OR W-PLANE)
        FREQUENCY RESPONSES                                  (W- OR S-PLANE)
        TRANSIENT RESPONSES
  ROOT LOCUS                                                 (Z-PLANE)


SYSTEMS MAY BE DEFINED BY,
  1. MATRICES                                 (LOAD)
  2. PARAMETERS                               (MATRIX)
  3. BLOCK DIAGRAM                            (CLASS)
  4. COMBINATION OF (1.,2.) AND (3.)          (MIXED)
```

TABLE II CONTROL PROGRAM JOB CONTROL LANGUAGE AS OF 11-1-74

```
JOB CONTROL CARDS FOR
  STANDARD CONTROL RUN
  WITHOUT SOURCE DECK
  WITHOUT PLOT TAPE

JOBN,CM72000,T4777.
ATTACH(JWELIB,ID=JWE,PW=ADDY,MR=1)
ATTACH(OVRLY,ID=JWE)
SEGLOAD(I=OVRLY)
LOAD(JWELIB)
EXECUTE.
789 PUNCH
   DATA
6789 PUNCH     (YELLOW CARD)
---------------------------------------------------------------------------------
JOB CONTROL CARDS FOR
  STANDARD CONTROL RUN
  WITH SOURCE DECK
  WITHOUT PLOT TAPE

JOBN,CM72000,T4777.
ATTACH(JWELIB,ID=JWE,PW=ADDY,MR=1)
FTN(LR=OUTPUT)
REWIND(LGO)
COPYL(JWELIB,LGO,JWE)
ATTACH(OVRLY,ID=JWE)
SEGLOAD(I=OVRLY)
LOAD(JWE)
EXECUTE.
789 PUNCH
   SOURCE DECKS
789 PUNCH
   DATA
6789 PUNCH     (YELLOW CARD)
---------------------------------------------------------------------------------
```

```
JOB CONTROL CARDS FOR
  STANDARD CONTROL RUN
  WITHOUT SOURCE DECK
  WITH PLOT TAPE

JOBN,CM72000,T4777,NT1.
ATTACH(JWELIB,ID=JWE,PW=ADDY,MR=1)
LABEL (CARD)  SEE DESCRIPTION
ATTACH(OVRLY,ID=JWE)
SEGLOAD(I=OVRLY)
LOAD(JWELIB)
EXECUTE.
EXIT.
BKSP(INPUT)
ATTACH(PLOTRC,ID=JWE,PW=ADDY,MR=1)
LOAD(PLOTRC)
REDUCE.
EXECUTE.
789 PUNCH
   DATA
PLOT CARDS    (SEE DESCRIPTION )
6789 PUNCH    (YELLOW CARD)
--------------------------------------------------------------------------------

JOB CONTROL CARDS FOR
  STANDARD CONTROL RUN
  WITH SOURCE DECK
  WITH PLOT TAPE

JOBN,CM72000,T4777,NT1.
ATTACH(JWELIB,ID=JWE,PW=ADDY,MR=1)
LABEL (CARD)  SEE DESCRIPTION
FTN(LR=OUTPUT)
REWIND(LGO)
COPYL(JWELIB,LGO,JWE)
ATTACH(OVRLY,ID=JWE)
SEGLOAD(I=OVRLY)
LOAD(JWE)
EXECUTE.
EXIT.
BKSP(INPUT)
ATTACH(PLOTRC,ID=JWE,PW=ADDY,MR=1)
LOAD(PLOTRC)
REDUCE.
EXECUTE.
789 PUNCH
   SOURCE DECKS
789 PUNCH
   DATA
PLOT CARDS    (SEE DESCRIPTION )
6789 PUNCH    (YELLOW CARD)
--------------------------------------------------------------------------------
```

JOB CONTROL CARDS USING UPDATE FILE WITHOUT PLOT TAPE

```
JOBN,CM72000,T4777.
ATTACH(CONUPF,ID=JWE,PW=ADDY,MR=1)
ATTACH(OVRLY,ID=JWE,MR=1)
UPDATE(P=CONUPF)
FTN(A,I=COMPILE,LR=0)
SEGLOAD(I=OVRLY)
LOAD(LGO)
EXECUTE.
789 PUNCH
*IDENT NAME
   UPDATES
*COMPILE  CONTROL.SUBSCL
*END
789 PUNCH
   DATA
6789 PUNCH    (YELLOW CARD)
------------------------------------------------------------------
```

JOB CONTROL CARDS USING UPDATE FILE WITH PLOT TAPE

```
JOBN,CM72000,T4777,NT1.
ATTACH(CONUPF,ID=JWE,PW=ADDY,MR=1)
ATTACH(OVRLY,ID=JWE,MR=1)
LABEL (CARD) SEE DESCRIFTION
UPDATE(P=CONUPF)
FTN(A,I=COMPILE,LR=0)
SEGLOAD(I=OVRLY)
LOAD(LGO)
EXECUTE.
EXIT.
BKSP(INPUT)
ATTACH(PLOTRC,ID=JWE,PW=ADDY,MR=1)
LOAD(PLOTRC)
REDUCE.
EXECUTE.
789 PUNCH
*IDENT NAME
   UPDATES
*COMPILE  CONTROL.SUBSCL
*END
789 PUNCH
   DATA
PLOT CARDS    (SEE DESCRIPTION )
6789 PUNCH    (YELLOW CARD)
------------------------------------------------------------------
```

THE UPDATE FILE CAN BE USED TO MODIFY SOURCE ROUTINES IN THE CONTROL PROGRAM. UPDATES ARE USED TO INSERT OR DELETE CARDS OF A SPECIFIED SUBROUTINE. EXAMPLES OF THE UPDATE DIRECTIVES CAN BE FOUND IN THE UPDATE REFERENCE MANUAL. THE SOURCE LISTING FOR THE CONTROL PROGRAM IS CONTAINED IN ROCM 2115.

```
-------------------------------------------------------------------------------


   DESCRIPTION OF LABEL CARD
CC1
LABEL(TAPE6,W,D=HD,L=XXXXXXXXXX,VSN=YYYY)*RING IN*ZZZ**
         XXXXXXXXXX=YOUR LABEL
         YYYY=VOLUME NO. OBTAINED FROM COMPUTER ROOM
         ZZZ=YOUR PAYROLL NUMBER




   DESCRIPTION OF PLOT CARDS
  PLOT CARDS
CARD ONE
CC1
PLOT
CARD TWO
CC1-4       CC 26-35                 CC 41-44
NNNN           NAME OF                  SUBTASK
               SUBMITTER                (4 DIGIT NUMBER)
CARD THREE-N
CC1-10  CC11-20
    Y         Y
MINIMUM   MAXIMUM
(FLOATING POINT)
(ROOT LOCUS OR ROOT CONTOUR ONLY)
NNNN-VSN NUMBER
```

TABLE III. CONTROL: SYSTEM MODELS

A. CONTINUOUS SYSTEM MODELS

1. OPEN LOOP

$$C\dot{x} = Ax + Bu$$
$$y = Hx + G\dot{x} + Fu$$

2. CLOSED LOOP

$$C\dot{x} = Ax + Bu$$
$$u = K1x + K2\dot{x} + Du_{com}$$
$$y = Hx + G\dot{x} + Fu$$

3. ROOT LOCUS

$$C\dot{x} = Ax + Bu$$
$$u = (K1x + K2\dot{x}) + (K3x + K4\dot{x})$$

B. DISCRETE SYSTEM MODELS

1. OPEN LOOP

$$x_{n+1} = Ax_n + Bu_n$$
$$y_n = Hx_n + Fu_n$$

2. CLOSED LOOP

$$x_{n+1} = Ax_n + Bu_n$$
$$u_n = K1x_n + Du_{com\,n}$$
$$y_n = Hx_n + Fu_n$$

3. ROOT LOCUS

$$x_{n+1} = Ax_n + Bu_n$$
$$u_n = (K1x_n) + (K3x_n)$$

C. SAMPLED-DATA SYSTEM MODELS

1. ORIGINAL SYSTEM

$$\left[\begin{array}{c} \dot{x}^c \\ \hline x^d_{n+1} \end{array}\right] = \left[\begin{array}{c:c} A_c & 0 \\ \hdashline 0^* & A_d \end{array}\right] \left[\begin{array}{c} x^c \\ \hline x^d_n \end{array}\right] + \left[\begin{array}{c:c} B_c & 0 \\ \hdashline 0 & B_d \end{array}\right] \left[\begin{array}{c} u^c \\ \hline u^d_n \end{array}\right]$$

$$\left[\begin{array}{c} y^c \\ \hline y^d_n \end{array}\right] = \left[\begin{array}{c:c} H_c & 0 \\ \hdashline 0^* & H_d \end{array}\right] \left[\begin{array}{c} x^c \\ \hline x^d_n \end{array}\right] + \left[\begin{array}{c:c} F_c & 0 \\ \hdashline 0 & F_d \end{array}\right] \left[\begin{array}{c} u^c \\ u^d_n \end{array}\right]$$

2. CONNECTIONS PRIOR TO DISCRETIZATION

$$\left[\begin{array}{c} u^c \\ \hline u^d_n \end{array}\right] = \left[C \right] \left[\begin{array}{c} y^c \\ \hline y^d_n \end{array}\right]$$

WHERE C IS DEFINED BY YTOV AND ZTOU AND DEFINES CONNECTIONS MADE BEFORE THE PLANT IS DISCRETIZED (SEE TABLE VII B.)

3. DISCRETIZED SYSTEM

$$\left[\begin{array}{c} x^c_{n+1} \\ \hline x^d_{n+1} \end{array}\right] = \left[\begin{array}{c:c} \phi(T) & 0 \\ \hdashline 0^* & A_d \end{array}\right] \left[\begin{array}{c} x^c_n \\ x^d_n \end{array}\right] + \left[\begin{array}{c:c:c} [\oplus(T)B_c] & [\phi(T)B_c] & 0 \\ \hdashline 0 & 0 & B_d \end{array}\right] \left[\begin{array}{c} u^c_n \\ \hline u^d_n \end{array}\right]$$

$$\left[\begin{array}{c} y^c_n \\ \hline y^d_n \end{array}\right] = \left[\begin{array}{c:c} H_c & 0 \\ \hdashline 0^* & H_d \end{array}\right] \left[\begin{array}{c} x^c_n \\ x^d_n \end{array}\right] + \left[\begin{array}{c:c:c} [F_c] & [H_c B_c] & 0 \\ \hdashline 0 & 0 & F_d \end{array}\right] \left[\begin{array}{c} u^c_n \\ \hline u^d_n \end{array}\right]$$

4. CONNECTIONS AFTER DISCRETIZATION

$$\left[\begin{array}{c} u^c_n \\ \hline u^d_n \end{array}\right] = \left[R \right] \left[\begin{array}{c} y^c_n \\ \hline y^d_n \end{array}\right] = \left[K1 \right] \left[\begin{array}{c} x^c_n \\ \hline x^d_n \end{array}\right]$$

WHERE R IS DEFINED BY YZTOK AND DEFINES CONNECTIONS MADE AFTER THE PLANT IS DISCRETIZED (SEE TABLE VII B.). FOR ROOT LOCUS. THE SECOND CONNECTION SPECIFIED BY YZTOK (IF ANY) WILL GENERATE K3 (SIMILAR TO K1) DEFINING A SECOND FEEDBACK VARIABLE.

5. FINAL SAMPLED-DATA SYSTEM

$$\left.\begin{array}{l} x_{n+1} = Ax_n + Bu_n \\ y_n = Hx_n + Fu_n \end{array}\right\} \quad \text{FROM 3.}$$

$$u_n = K1\,x_n + D u_{com_n} \quad \text{FROM 4. } (D=I)$$

* THESE SUBMATRICES MAY CONTAIN NON-ZERO ELEMENTS.

TABLE IV CONTROL PROGRAM DATA DECK FORMAT

CARD 1 TITLE CARD FORMAT (10A8)
ANY LITERAL DATA DESIRED TO LABEL ALL PRINTOUTS AND PLOTS

CARD(S) 2 TO N NAMELIST CODE (SEE APPENDIX 2)
INTEGER VARIABLES
READ,SYSTEM,OUTPUT,MIXED,DIGITL,FRPS,NUMERS,TRESP,NX,NY,NU,NXC,
NUC,ZOH,N1,N2,CONTUR,MULTRT,MODEL,NSCALE,CMAT,NK2,FORM,IPT,IGO,
SAV,IFLAG,READ3

REAL VARIABLES
DELT,FINALT,IFREQ,FFREQ,DELFRQ,GAIN1,GAIN2,M

CARD N+1 OUTPUT LABELS FORMAT (8A10)
LITERAL OUTPUT VARIABLE LABELS USED TO LABEL PRINTOUTS AND PLOTS. ORDERED IN SEQUENCE CORRESPONDING TO NY OUTPUTS, Y. LEAVE BLANK IF SYSTEM = 3. Will read the greater of 8 or NY outputs.

CARD N+2 INPUT LABELS FORMAT (8A10)
LITERAL INPUT VARIABLE LABELS USED TO LABEL PRINTOUTS AND PLOTS. ORDERED IN SEQUENCE CORRESPONDING TO NU INPUTS, U. LEAVE BLANK IF SYSTEM = 3. Will read the greater of 8 or NU inputs

CARD(S) (N+3) TO (M) SYSTEM DATA (SEE TABLE V AND APPENDIX 2)
SYSTEM DATA AS SPECIFIED BY LOAD,MATRIX,CHANGE,OR CLASS. IF READ =1, EACH DATA MATRIX IS READ ROW-WISE WITH FORMAT (8F10.4) AND EACH MATRIX MUST BE PRECEDED BY A DIMENSION CARD (FORMAT (2I10)) GIVING THE NUMBER OF ROWS AND COLUMNS OF THE MATRIX.

CARD M+1 TRANSIENT RESPONSE INPUT DATA
INPUT IS CALLED BY THIST TO GENERATE THE TRANSIENT RESPONSE INPUT VECTOR. THE INPUT SUBROUTINE ON THE DISC READS ONE DATA CARD FOR EACH RESPONSE DEFINING A STEP INPUT ON THE NU COMPONENTS OF THE augmented and thinned input vector, u. (format(**7F10.4**))

THE ABOVE CARDS DEFINE ONE CASE. AS MANY CASES AS DESIRED MAY BE STACKED TOGETHER FOR A SINGLE COMPUTER RUN.

IF A PLOT IS REQUESTED THE PLOT CARDS AS DESCRIBED IN TABLE II ARE REQUIRED

TABLE V. DATA REQUIRED BY CONDITION CODES

STEP 1 READ = 1,2,3

SYSTEM		REQUIRED MATRICES
1	OPEN LOOP	A,B,C,H,G,F
	IF MIXED =1, THIS IS STEP 1 OF THE MIXED LOADING OPTION	
2	CLOSED LOOP	A,B,C,H,G,F,K1,K2,D
3	ROOT LOCUS	A,B,C,K1,K2,K3,K4

IF CMAT = 0 C MATRIX NOT REQUIRED
IF NK2 = 0 K2,K4 MATRIX NOT REQUIRED
IF OUTPUT = 1 G,F MATRIX NOT REQUIRED
IF OUTPUT = 2 F MATRIX NOT REQUIRED
IF OUTPUT = 3 G MATRIX NOT REQUIRED
IF N2=0 K3,K4 MATRIX NOT REQUIRED

STEP 2 READ = 4 ; OR,
IF MIXED =1, THIS IS STEP 2 OF THE MIXED LOADING OPTION

CARD 1 NBLOCK,NIT FORMAT (2I5)
IF NIT = 0 GO TO (*)

CARDS 2-(NBLOCK+1) ONE CARD PER BLOCK FORMAT (I2,I3,5I5,5F10.4)

NUM,TYPE,(CONNEC(I),I=1,4),MOD,(PARAM(I),I=1,5)

NUM = BLOCK NUMBER
TYPE = BLOCK TYPE (1-10) SEE TABLE VI
CONNEC = SPECIFIES INPUTS TO BLOCK. FIRST THREE ELEMENTS MAY SPECIFY CONNECTIONS FROM OTHER BLOCKS (±). FOURTH ELEMENT MAY SPECIFY EXTERNAL INPUT (±).
MOD = SPECIFIES THAT G(P) IS S-,Z-, OR W- TRANSFORM As specified in Table VI
PARAM = PARAMETERS DEFINING BLOCKS AS SPECIFIED IN TABLE VI
GO TO (**)

	NAME	TYPE	DIMENSION	FORMAT
(*)	GRAPH	INTEGER	NBLOCK X 5	5I5
	BLOCK	INTEGER	NBLOCK X 3	3I5
	NUMER	REAL	NBLOCK X 5	5F10.4
	DENOM	REAL	NBLOCK X 5	5F10.4
	GAIN	REAL	NBLOCK	8F10.4
(**)	ITHINY	INTEGER	≤ # OF OUTPUTS	16I5

STOP IF MIXED =0 UNLESS READ=4 and SYSTEM =3

ITHINU	INTEGER	≤ # OF INPUTS	16I5
NYTOV,NZTOU,NYZTOK	INTEGER		3I5
YTOV	INTEGER	NYTOV X 2	2I5
ZTOU	INTEGER	NZTOU X 2	2I5
YZTOK	INTEGER	NYZTOK X 2	2I5

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE VI   TRANSFER FUNCTION STANDARD FORMS   (NIT = 1)

FOR READ = 4 OR MIXED =1 STEP 2, THE CLASS SUBROUTINE ACCEPTS BLOCK DIAGRAM TRANSFER FUNCTIONS OF THE FOLLOWING FORMS. (SEE TABLE V. STEP2)

TYPE	G(P)	PARAM(I)				
		I=1	2	3	4	5
1	K	K				
2	Kp	K				
3	$K\frac{1}{p}$	K				
4	$\frac{K}{(1+p/a)}$	K	a			
5	$\frac{K(1+p/b)}{(1+p/a)}$	K	a	b		
6	$\frac{Kp}{(p+a)}$	K	a			
7	$\frac{K}{(1+p/a)(1+p/b)}$	K	a	b		
8	$\frac{K}{1+\frac{2\zeta}{\omega}p+p^2/\omega^2}$	K	ω	ζ		
9	$\frac{K(1+\frac{2\zeta_2}{\omega_2}p+p^2/\omega_2^2)}{(1+\frac{2\zeta_1}{\omega_1}p+p^2/\omega_1^2)}$	K	ω_1	ζ_1	ω_2	ζ_2
10	$\frac{K(1+p/a)}{(1+\frac{2\zeta}{\omega}p+p^2/\omega^2)}$	K	ω	ζ	a	
11	$\frac{Kp}{1+2\frac{\zeta}{\omega}p+p^2/\omega^2}$ *	K	w	ζ		

*Can consist of two real roots

TABLE VI (CONT.)

THE STANDARD FORM, G(P), IS INTERPRETED AS AN S-,W-, OR Z-PLANE TRANSFER FUNCTION AS SPECIFIED BELOW

MOD / DIGITL	0	1	2
0	G(S)	---	---
1	G(S),G(Z)	G(S)	G(W)
2	G(Z)	G(S)	G(W)

IF MOD = 0 THE TRANSFER FUNCTION COEFFICIENTS ARE NOT MODIFIED

IF MOD = 1 THE TRANSFER FUNCTION CRITICAL FREQUENCIES ARE PREWARPED TO APPROXIMATE THE S-PLANE FILTER IN THE W- PLANE AND THEN TRANSFORMED TO THE Z-PLANE AS DESCRIBED BELOW. FIRST ORDER POLES AND ZEROES ARE WARPED AS

$$a = \tanh\left(\frac{aT}{2}\right)$$

COMPLEX POLES AND ZEROES ARE WARPED AS

$$\omega_w = \sqrt{u_w^2 + \nu_w^2} \qquad \zeta_w = -\frac{u_w}{\omega_w}$$

where $$u_w = \frac{\sinh(\alpha T)}{\cosh(\alpha T) + \cos(\beta T)}$$

$$\nu_w = \frac{\sin(\beta T)}{\cosh(\alpha T) + \cos(\beta T)}$$

with $$\omega_s^2 = \alpha^2 + \beta^2$$

$$\zeta_s = -\frac{\alpha}{\omega_s}$$

IF MOD = 2 THE TRANSFER FUNCTION IS REGARDED AS A W-PLANE TRANSFER FUNCTION AND TRANSFORMED TO THE Z-PLANE BY THE BILINEAR TRANSFORMATION

W = (Z-1)/(Z+1)

IF DIGITL = 1 AND MOD ≠ 0 THE TRANSFER FUNCTION CANNOT BE PART OF THE CONTINUOUS PLANT (I.E. THE INPUT TO THE FILTER MUST BE NUMBERED HIGHER THAN NUC AND THE FILTER STATES MUST BE NUMBERED HIGHER THAN NXC)

IF DIGITL = 1 AND MOD = 0 THE TRANSFER FUNCTION COEFFICIENTS ARE NOT MODIFIED. THE FUNCTION IS INTERPRETED AS AN S- OR Z- PLANE FUNCTION DEPENDING ON THE RELATION OF THE FILTER STATES TO NXC. IF THE FILTER STATES ARE NUMBERED LESS THEN NXC, THEN THE FILTER IS TREATED AS PART OF THE CONTINUOUS PLANT. IF THE FILTER STATES ARE NUMBERED GREATER THAN NXC THEN THE FILTER IS TREATED AS PART OF DIGITAL CONTROLLER.

TABLE VII DISCRETIZED PLANT MODELS FOR VARIOUS INPUT AND OUTPUT DEFINITIONS

	A. (*) $y_n = y_n^- \equiv y(nT-\epsilon)$	B. (**) $y_n = y_n^+ \equiv y(nT+\epsilon)$
CONTINUOUS MODEL	$\dot{x} = Ax + Bu$ $y = Hx + Fu$	
SAMPLED INPUT	$x_{n+1} = \phi(T)x_n + \phi(T)Bu_n$ $y_n = Hx_n$	$x_{n+1} = \phi(T)x_n + \phi(T)Bu_n$ $y_n = Hx_n + HBu_n \quad (F \equiv 0)$
ZOH INPUT	$x_{n+1} = \phi(T)x_n + \oplus(T)Bu_n$ $y = Hx_n$	$x_{n+1} = \phi(T)x_n + \oplus(T)Bu_n$ $y_n = Hx_n + Fu_n$

	C. (***) TIME DELAY $y_n(m) \equiv y[(n+m-1)T]$ $\quad 0 < M \leq 1$
CONTINUOUS MODEL	$\dot{x} = Ax + Bu$ $y = Hx + Fu$
SAMPLED INPUT $(F \equiv 0)$	$x_{n+1} = \phi(T)x_n + \phi(T)Bu_n$ $y_n(m) = H\phi(mT)x_{n-1} + H\phi(mT)Bu_{n-1}$
ZOH INPUT	$x_{n+1} = \phi(T)x_n + \oplus(T)Bu_n$ $y_n(m) = H\phi(mT)x_{n-1} + [H\oplus(mT)B + F]u_{n-1}$

$$\phi(t) = \int_0^t e^{-A(t-\tau)}d\tau \; ; \quad \oplus(t) = \int_0^t \phi(t-\tau)d\tau \; ; \quad x_n = x_n^- \equiv x(nT-\epsilon)$$

(*) CONTROL ASSUMES $y_n = y_n^-$ FOR ANY CONNECTIONS FROM THE PLANT TO THE DIGITAL CONTROLLER (I.E. PLANT FEEDBACK).

(**) CONTROL ASSUMES $y_n = y_n^+$ FOR FINAL SYSTEM OUTPUT CALCULATIONS (TRANSIENT RESPONSES, TRANSFER FUNCTIONS).

(***) MODIFIED Z-TRANSFORM ANALYSIS CAN ONLY BE USED WITH OPEN LOOP SAMPLED-DATA SYSTEMS.

TABLE VIII VECTOR ORDERING AND SYSTEM CONNECTION CONVENTIONS FOR SAMPLED-DATA SYSTEMS

A. VECTOR ORDERING CONVENTION

THE COMPONENTS OF THE AUGMENTED VECTORS U, X, AND Y MUST BE ORDERED IN THE FOLLOWING SEQUENCES, (IN STEP 2 OF THE MIXED OPTION, NUMBER THE EXTERNAL INPUTS AND BLOCK OUTPUTS IN THE INDICATED ORDER)

INPUT VECTOR, U

1. INPUTS TO PLANT FROM ZERO-ORDER-HOLD ELEMENTS (ZOH)
2. INPUTS TO PLANT FROM SAMPLERS (NUC-ZOH)
3. INPUTS TO DIGITAL CONTROLLER (NU-NUC)

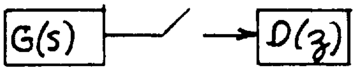
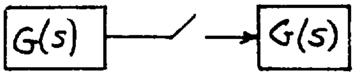
STATE VECTOR, X

1. CONTINUOUS STATES (PLANT, NXC)
2. DISCRETE STATES (DIGITAL CONTROLLER, NX-NXC)

OUTPUT VECTOR, Y

1. PLANT OUTPUTS
2. DIGITAL CONTROLLER OUTPUTS

B. SYSTEM CONNECTION CONVENTION

<u>CONNECTION</u>	<u>DEFINED IN</u>
	YTCV,ZTOU,GRAPH,A
	"
	"
	YZTOK
	"

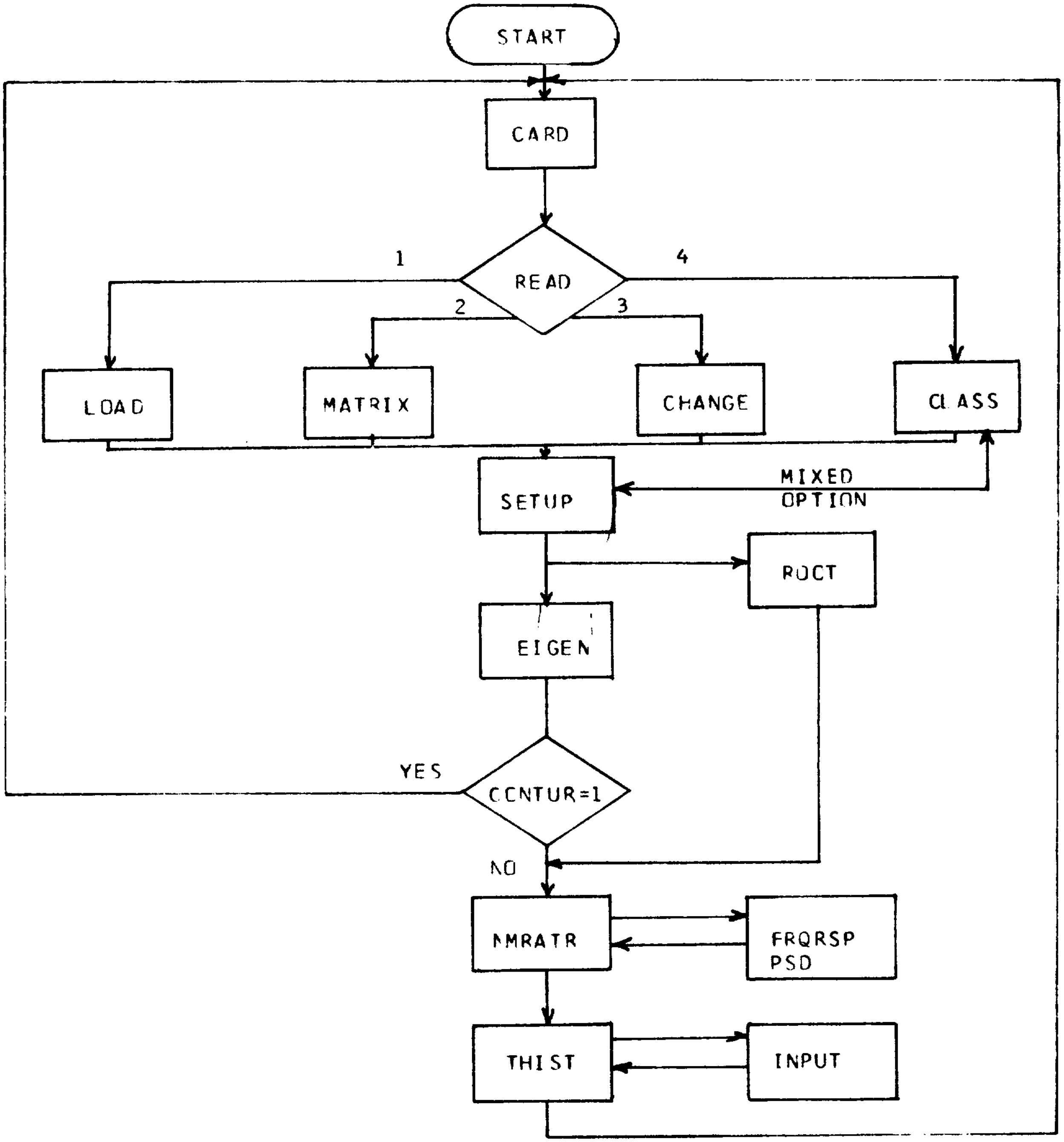
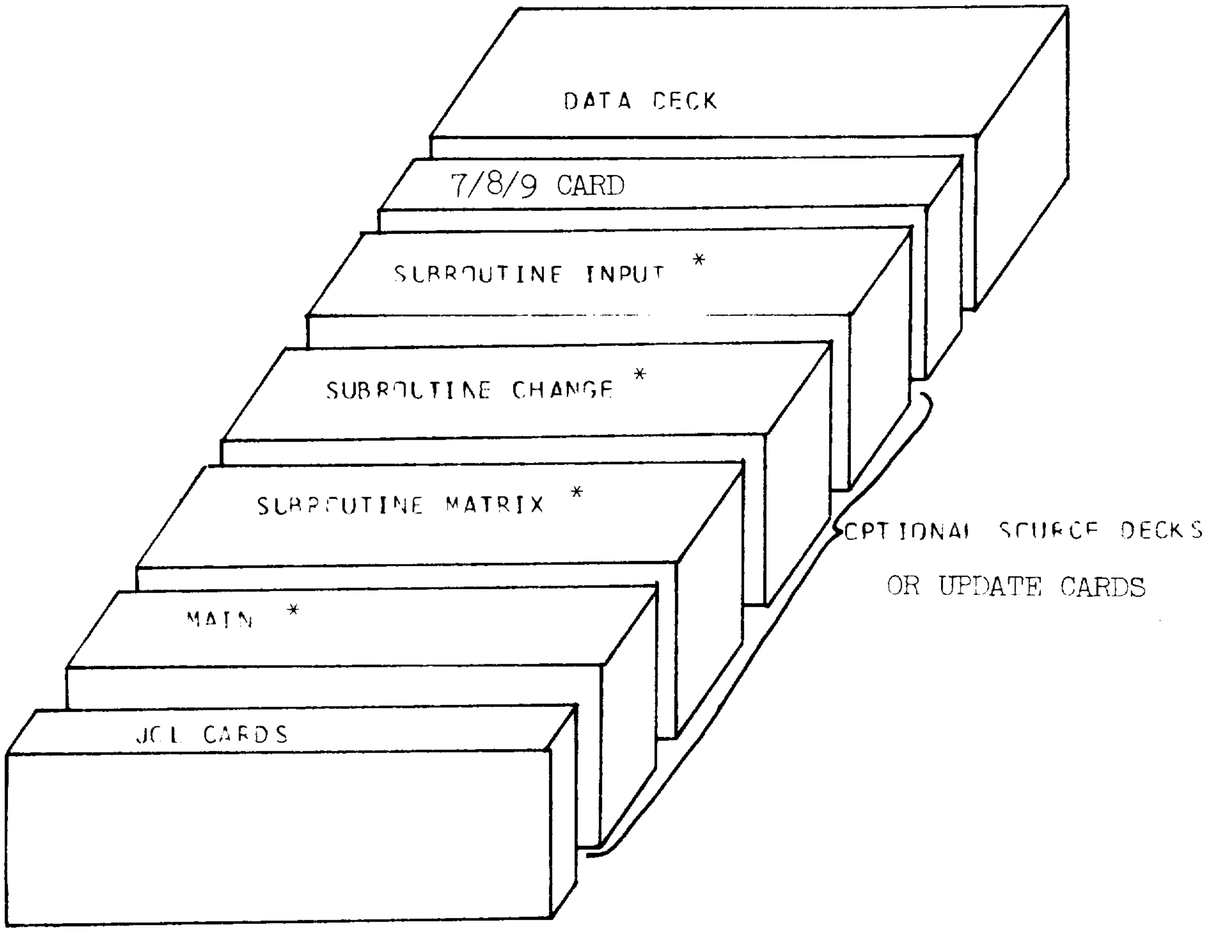


FIGURE 1 FLOW CHART OF CNTRLR SUBROUTINE



*OTHER SUBROUTINES AS DESIRED

FIGURE 2. STRUCTURE OF CONTROL DECK

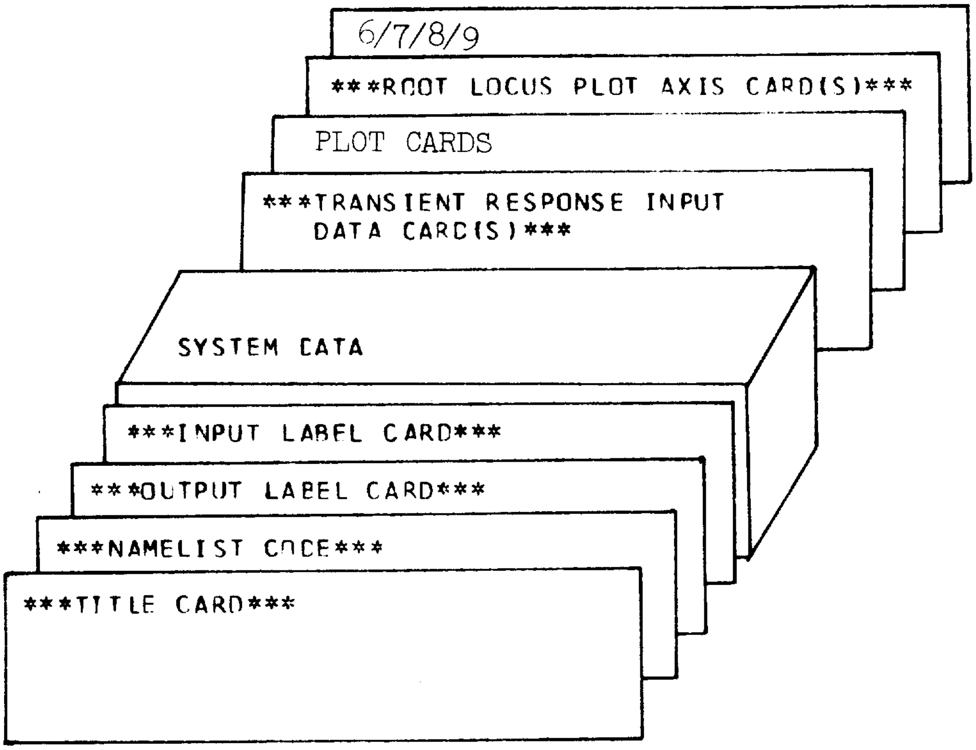


FIGURE 3. CONTROL PROGRAM DATA DECK STRUCTURE

```
CC2
 |
 ↓
 $CODE           READ=1,     SYSTEM=  1,OUTPUT=3,NX=4,NY=5,NU=2,
     DELT= .05 ,FINALT=10., IFREQ=.5 ,FFREQ= 100.,DELFRQ=1.11,
 IPT=2,$END
```

FIGURE 4 EXAMPLE OF DATA ENTRY USING A NAMELIST FORMAT

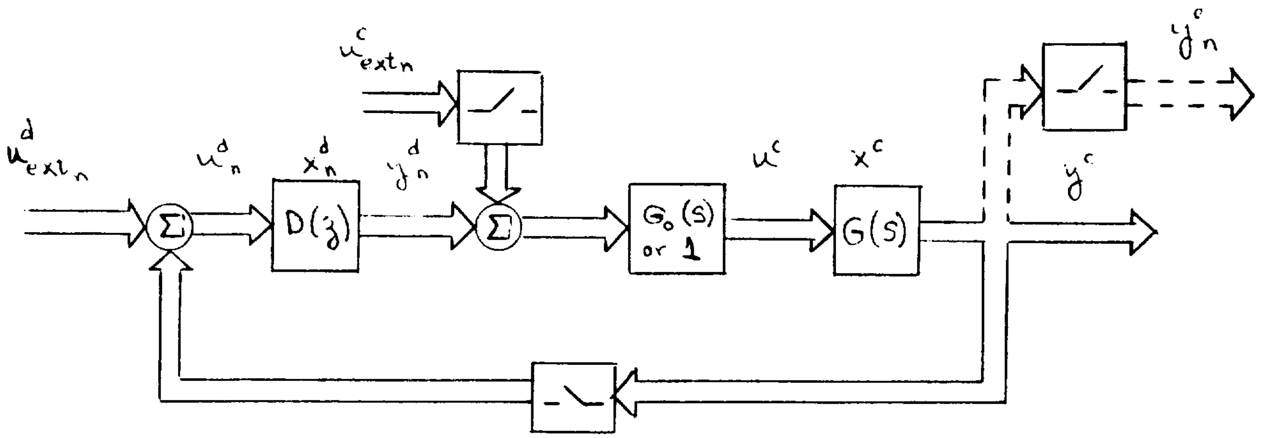
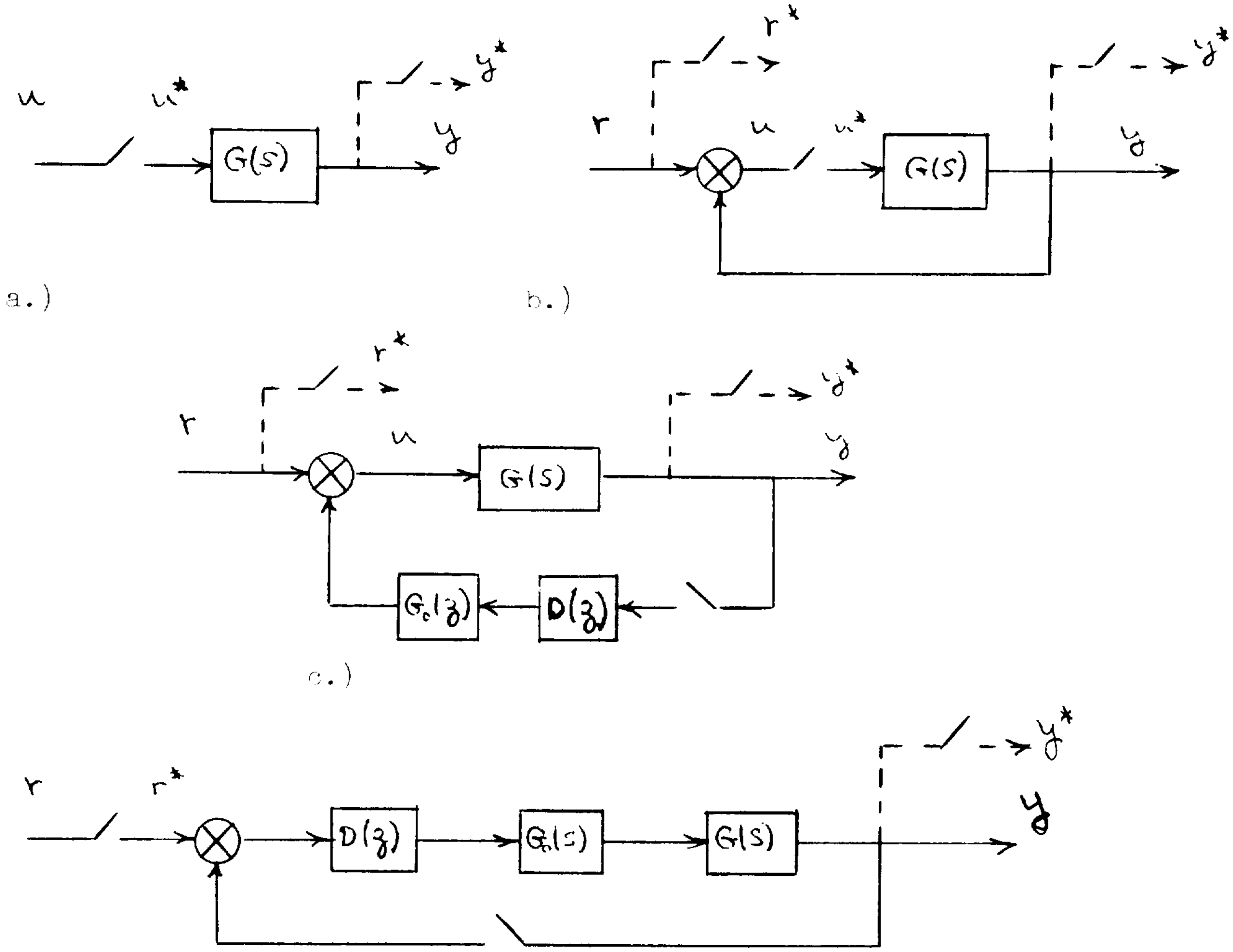


Figure 5.- Sampled-data system block diagram.



a.)

b.)

c.)

d.)

Figure 6.- Typical sampled-data systems analyzed by CONTROL.

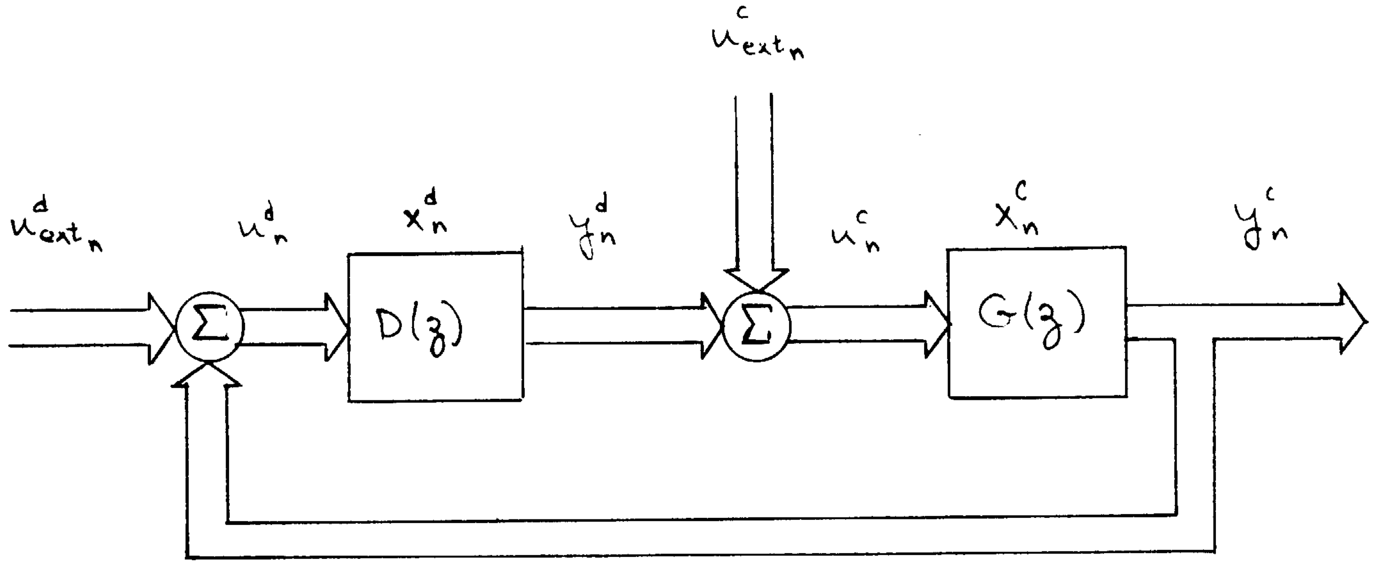
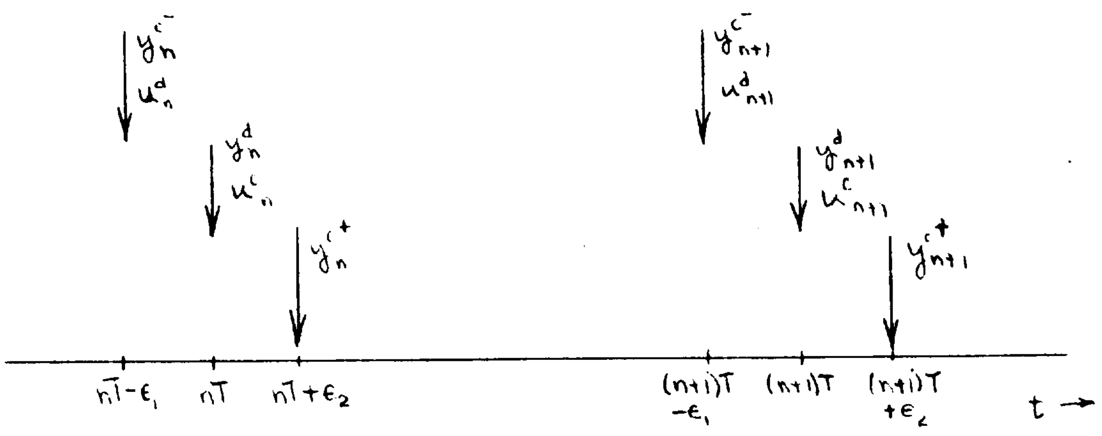


Figure 7.- Discretized sampled-data system block diagram.



a.) Time sequence of digital controller and plant.

b.) Idealized time sequence.

Figure 8.- Time sequence models for sampled-data systems.

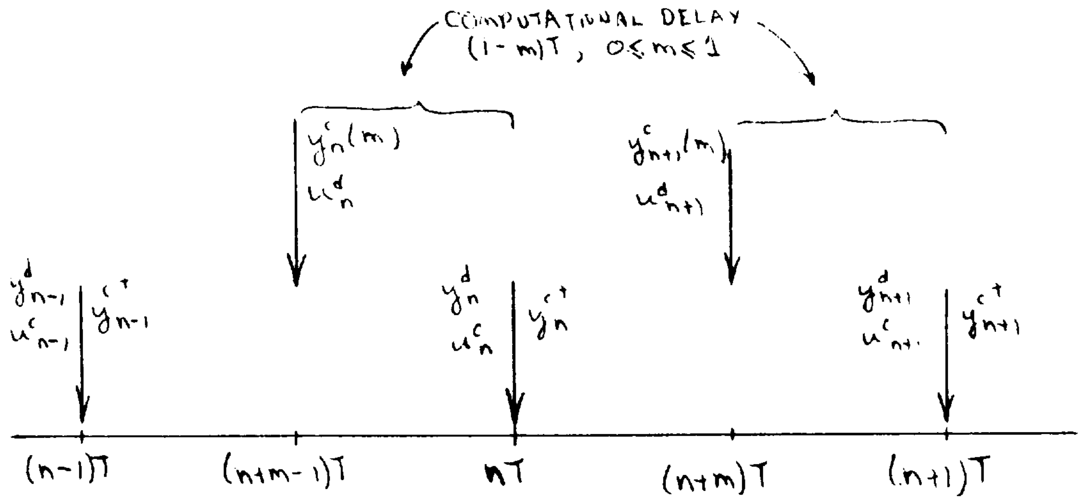


Figure 9.- Time sequence model for sampled-data systems with computational time delay.