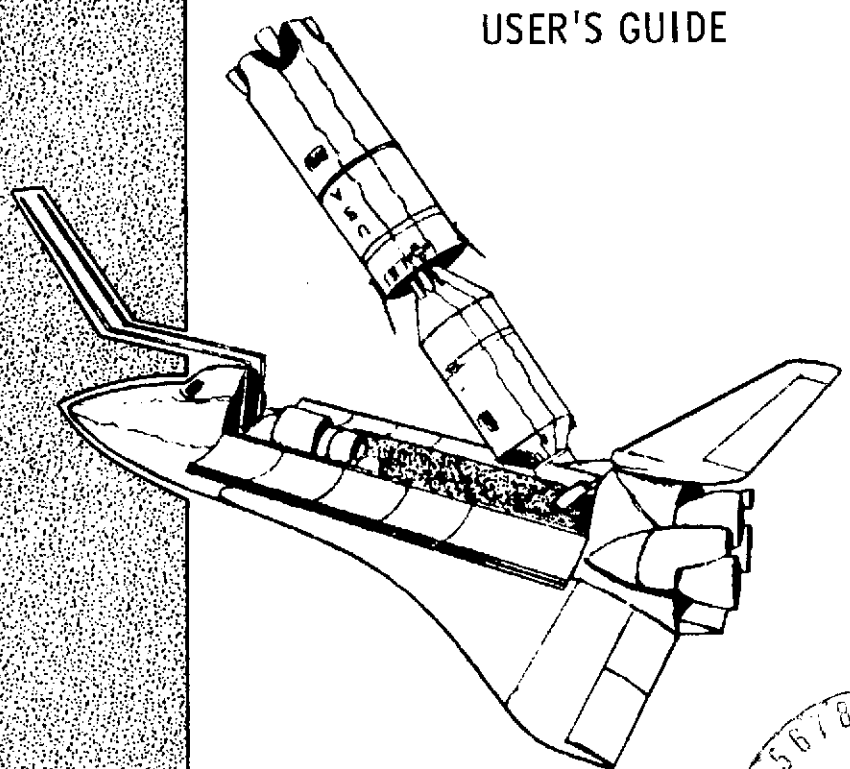
NASA CR-140190

FINAL REPORT
CONTRACT NAS9-13568
JUNE 28, 1974

ASYMMETRICAL BOOSTER ASCENT GUIDANCE AND CONTROL SYSTEM DESIGN STUDY

VOLUME IV

SAMPLED DATA STABILITY ANALYSIS PROGRAM (SADSAP) USER'S GUIDE



REPORT
5-2581-HOU-154

BOEING aerospace company
HOUSTON, TEXAS

5-2581-HOU-154

CONTRACT NAS9-13568

ASYMMETRICAL BOOSTER ASCENT GUIDANCE AND CONTROL SYSTEM DESIGN STUDY

VOLUME IV

SAMPLED DATA STABILITY ANALYSIS PROGRAM (SADSAP) USERS GUIDE

JUNE 28, 1974

PREPARED BY J. L. WILSON

APPROVED BY W. G. RYALS
PROGRAM MANAGER

PREFACE

Final report of Asymmetrical Booster Ascent Guidance and Control System Design Studies performed under Contract NAS9-13568 are contained in five separate volumes identified as follows:

Volume I - Summary
Volume II - SSFS Math Models - Ascent
Volume III - Space Shuttle Vehicle SRB Actuator Failure Study
Volume IV - Sampled Data Stability Analysis Program (SADSAP) - Users Guide
Volume V - Space Shuttle Powered Explicit Guidance

ABSTRACT AND ACKNOWLEDGMENTS

Volume IV provides a users guide to the Sampled Data Stability Analysis Program (SADSAP). This program is a general purpose sampled data Stability Analysis Program capable of providing frequency response on root locus data.

Acknowledgments are given to George Paulchak of Boeing Computer Services, Huntsville, Alabama, and Emmit Fisher of NASA/JSC for their programming support.

KEY WORDS

Stability Analysis
Sample Data
Nichols Plot
Frequency Response
Root Locus

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

REFERENCES

1. 'CONTROL SYSTEM ROOT LOCUS' by G. E. Paulchak, Boeing computer program number BHA0272; Huntsville, Alabama; September 1969.

2. 'GENERAL FREQUENCY RESPONSE' by P. N. Smith, Boeing computer program number BHA0091; Huntsville, Alabama; April 1969.

3. B. C. Kuo, Analysis and Synthesis of Sampled Data Control Systems; Prentice-Hall Publishing, Inc.; 1963.

1.0 INTRODUCTION & SUMMARY

The SAmpled Data Stability Analysis Program (SADSAP), formerly known as The Boeing-Huntsville Program BHA-369, was converted to the SBU 1110 EXECT 8 system. This program is a general purpose sampled data stability analysis program capable of providing frequency response and root locus data. The continuous system open loop and closed loop poles along with open loop zero's are also provided.

The program accepts a general matrix format with polynomials of S (Laplace Operator) in each element location. The element location of the sampler is specified and the frequency options selected. Output includes frequency response data as a printout and/or a Nichols plot.

A detailed description of program input requirements is provided in section 2.0; a discussion of technical methods in section 3.0; flow charts in section 4.0; a sample problem in section 5.0; and a complete program listing in section 6.0.

1.1 Purpose

The Sampled Data Stability Analysis Program, BHA-369, provides a computer tool for studying a digital control system which has a sampled data signal in the system. Conventional stability analysis tools assume continuous signals throughout the system and therefore are not applicable.

BHA-369 computes R domain open loop frequency response and Z domain gain and phase root locus of a sampled data closed loop control system. The control system definition includes the S-domain characteristic matrix, sampling device location, sample period, optional zero order hold circuit, and optional transport lag. Output features of the program are Nichols plots, digital print of the frequency response and root locus analysis, and digital print of the partial fraction expansion of the open loop transfer function in the Z domain and R domain.

1.2 Assumptions

Sampled data open loop poles at the origin in the Z domain are roots which show little movement and affect insignificantly the movement of other roots. As such, these roots can be removed from the system prior to root locus analysis. Refer to section 3.5 for further discussion.

The zero order hold circuit has the Laplace transformation $\frac{1-e^{-ST}}{T}$ where T equals the sample period.

Transport lag is implemented into the system by the Laplace Transformation $e^{-T_D S}$ where T_D is the transport lag.

1.3 Limitations

Only one sampling device can exist in the system.

Analysis is restricted to systems which have the sampling device, zero order hold circuit, and transport lag in the same loop of the system block diagram.

Frequency response analysis is computed only for the system opened at the sampling device location.

Root locus analysis is computed only for the system with the gain factor implemented at the sampling device location.

Number of continuous system open loop poles at the origin less the number of open loop zeros at the origin must be less than or equal to two. Implementation of a zero order hold circuit permits a maximum of three.

Multiple non-zero continuous system open loop poles are not permitted. This is a restriction of the partial fraction expansion of the continuous system open loop transfer function. However, the rooting procedure in the program causes multiple non-zero roots to appear as clustered roots thereby circumventing the problem. Refer to section 3.4.6 for additional information.

Frequency response of sampled data open loop transfer functions cycles every $\frac{1}{2T}$ Hertz where T equals the sample period. Frequency intervals have been arbitrarily restricted to a maximum of $\frac{1}{T}$ Hertz.

2.0 INPUT DATA PREPARATION

Input to the SADSAP Program consists of seven input sections which are sequentially input then repeated for multiple cases. Certain capabilities which are not now provided but might be built later if desired are provided for and labeled as "Dummy Routines". These routines were and are available in the original BHA-369 version, which is an IBM 360 version.

SECTION	CONTENTS	STATUS
I	Basic Input Data	Required
II	Nyquist Data	Optional
III	Root Locus Data	Optional
IV	Continuous System Root Estimates	Optional
V	Continuous System Matrix	Required
VI	Parameter Variations	Optional
VII	Key Word	Optional

SECTION I - BASIC INPUT DATA - REQUIRED INPUT

CARD	COLUMNS	FORMAT	CONTENTS	EXPLANATION
I-1	1-80	20A4	TITLE	80 character case title.
I-2	1-40	10A4	TITLE 1	40 character case title
	41-80	10A4	TITLE 2	40 character case title
I-3	1-12	E12.5	TD	Time delay
	21-28	2A4	NAME	8 character name given to the open loop computation
	31-35	I5	IOPEN	Row location of sampling device
	36-40	I5	JOPEN	Column location of sampling device.
	41-45	I5	NZT	Number of sample periods. $1 \leq NZT \leq 50$
	46-50	I5	NRPOLE	Maximum number of continuous system open loop poles to find - if input as zero, all poles will be found.
	51-55	I5	NRZERO	Maximum number of continuous system closed loop zeros to find - if input as zero, all zeros will be found.
	56-60	I5	NRCLPL	Maximum number of continuous system closed loop poles to find - if input as zero, all poles will be found.
I-4	1	A1	OPTZ	Non-blank yields Z transformation without zero order hold device.
	11	A1	OPTZOH	Non-blank yields Z transformation with zero order hold device.
	21	A1	OPTTRL	Non-blank yields sampled data root locus.
	31	A1	OPTTP	Non-blank yields sampled data root locus plots.
	41	A1	OPTPCH	Non-blank generates punched cards of the continuous case matrix generated from raw data.
	51	A1	OPTPNT	Non-blank yields printout of the continuous system characteristic matrix and open loop transfer function.
	61	A1	OPTBUG	Non-blank yields debug printout in the rooting routines
I-5	1-12	E12.5	$ZTVAL_1$	First sample period
	13-24	E12.5	$ZTVAL_2$	Second sample period
	25-36	E12.5	$ZTVAL_3$	Third sample period
	37-48	E12.5	$ZTVAL_4$	Fourth sample period
	49-60	E12.5	$ZTVAL_5$	Fifth sample period
	61-72	E12.5	$ZTVAL_6$	Sixth sample period

Data card format I-5 is repeated until NZT sample rates have been input.

SECTION II - NYQUIST DATA - OPTIONAL

(omit section II if Nyquist analysis not desired)

NOTE: Frequency response of sampled data systems is cyclic in frequency intervals of $\frac{1}{2T}$ where T equals the sample period.

CARD	COLUMNS	FORMAT	CONTENTS	EXPLANATION
II-1	1-4	A4	REQEST	Input the control word NYQUIST
	21-24	A4	OPTINP	Nyquist input option selected by user: RETAIN → Retain Nyquist data from the previous case; omit data cards 2 and 3 of section II. Dummy Routine STANDARD → Implement standard Nyquist data; see Appendix E; omit data cards 2 and 3 of section II. Dummy Routine NEW → Input new Nyquist data as required by data cards 2 and 3 of section II.
	31	A1	PN	Non-blank yields Nyquist plot. Dummy Routine
	41	A1	PB	Non-blank yields Bode plot. Dummy Routine
	51	A1	P180	Non-blank yields 180 degree phase shift on all computed Nyquist gains.
	61	A1	NIC	Non-blank yields Nichols plot.
II-2	1-5	I5	NFI	Number of Nyquist frequency intervals
II-3	1-12	E12.5	STR_i	S domain start frequency in hertz of the i-th interval
	13-24	E12.5	STP_i	S domain stop frequency in hertz of the i-th interval
	25-36	E12.5	PCT_i	Percent frequency increment within the i-th interval
	37-48	E12.5	MIN_i	Minimum acceptable phase shift in degrees within the i-th interval
	49-60	E12.5	MAX_i	Maximum acceptable phase shift in degrees within the i-th interval
	65	A1	DP_i	Non-blank yields detail print within the i-th interval

Repeat data card II-3 for i=1, 2, ..., NFI

SECTION III - ROOT LOCUS DATA - OPTIONAL

(omit section III if root locus analysis not desired)

CARD	COLUMNS	FORMAT	CONTENTS	EXPLANATION
III-1	1-4	A4	REQEST	Input the control word ROOT▽LOCUS
	21-24	A4	OPTINP	Root locus input option selected by user: RETAIN → Retain root locus data from the previous case; omit data cards 2 thru 8 of section III. STANDARD → Implement standard root locus data; see Appendix E; omit data cards 2 thru 8 of section III. NEW → Input new root locus data as required by data cards 2 thru 8 of section III.
	31	A1	PG	Non-blank yields gain root locus; gain variations must exist or be input as new data.
	41	A1	PP	Non-blank yields phase root locus; phase variations must exist or be input as new data.
	51	A1	PPLT	Non-blank yields root locus plot; plot specifications must exist or be input as new data. Dummy Routine.

Omit data cards 2 and 3 of section III if PG input as blank.

III-2	1-5	I5	NGAIN	Number of gain variations $1 \leq NGAIN \leq 50$
III-3	1-12	E12.5	$GAIN_1$	First gain value.
	13-24	E12.5	$GAIN_2$	Second gain value.
	25-36	E12.5	$GAIN_3$	Third gain value.
	37-48	E12.5	$GAIN_4$	Fourth gain value.
	49-60	E12.5	$GAIN_5$	Fifth gain value.
	61-72	E12.5	$GAIN_6$	Sixth gain value.

Repeat the format of data card III-3 until NGAIN gain values have been input.

III-4	1-5	I5	NPHASE	Number of phase variations $1 \leq NPHASE \leq 50$

CARD	COLUMNS	FORMAT	CONTENTS	EXPLANATION
III-5	1-12	E12.5	$PHASE_1$	First phase value in degrees.
	13-24	E12.5	$PHASE_2$	Second phase value in degrees.
	25-36	E12.5	$PHASE_3$	Third phase value in degrees.
	37-48	E12.5	$PHASE_4$	Fourth phase value in degrees.
	49-60	E12.5	$PHASE_5$	Fifth phase value in degrees.
	61-72	E12.5	$PHASE_6$	Sixth phase value in degrees.

Repeat the format of data card III-5 until ||PHASE phase values have been input.

Omit data cards 6 thru 8 of section III of PPLT input as blanks. Dummy Routine.

III-6	1	A1	GSYM	Symbol used to represent results from gain variations on the root locus plots; refer to Appendix D for possible selections.
	6	A1	PSYM	Symbol used to represent results from phase variations on the root locus plots; refer to Appendix D for possible selections.
	11-15	I5	NRLFR	Number of plot frames to use for displaying root locus results $1 \leq NRLFR \leq 10$
III-7	1-5	I5	NGR_1	Number of grids on the first plot frame.
	6-10	I5	NGR_2	Number of grids on the second plot frame.
	11-15	I5	NGR_3	Number of grids on the third plot frame.
	16-20	I5	NGR_4	Number of grids on the fourth plot frame.
	21-25	I5	NGR_5	Number of grids on the fifth plot frame.
	26-30	I5	NGR_6	Number of grids on the sixth plot frame.
	31-35	I5	NGR_7	Number of grids on the seventh plot frame.
	36-40	I5	NGR_8	Number of grids on the eighth plot frame.
	41-45	I5	NGR_9	Number of grids on the ninth plot frame.
	46-50	I5	NGR_{10}	Number of grids on the tenth plot frame. $1 \leq NGR_i \leq 4$
				Input NGR_i $i = 1, 2, \ldots, NRLFR$

CARD	COLUMNS	FORMAT	CONTENTS	EXPLANATION
III-8	1-12	E12.5	DD_i	Delta between grid marks on the i-th plot frame.
	13-24	E12.5	RX_i	Maximum X value on the i-th plot frame.
	25-36	E12.5	$BY_{i,1}$	Minimum Y value of the first grid of i-th plot frame.
	37-48	E12.5	$BY_{i,2}$	Minimum Y value of the second grid of i-th plot frame.
	49-60	E12.5	$BY_{i,3}$	Minimum Y value of the third grid of i-th plot frame.
	61-72	E12.5	$BY_{i,4}$	Minimum Y value of the fourth grid of i-th plot frame.

Input $BY_{i,J}$ for J=1, 2, ..., NGR_i where i represents the i-th plot frame.

Repeat data card III-8 for i = 1, 2, ..., HRLFR

SECTION IV - CONTINUOUS SYSTEM ROOT ESTIMATES - OPTIONAL

(Omit section IV if estimates used to compute the continuous system open loop zeros and poles and closed loop poles are not to be input).

CARD	COLUMNS	FORMAT	CONTENTS	EXPLANATION
IV-1	1-4	A4	REQEST	Input the control word ESTIMATES
	21-24	A4	OPTINP	Estimates input option selected by user: RETAIN → Retain estimates from the previous case; omit data cards 2 and 3 of section IV. Dummy Routine. NEW → Input new set of estimates as required by data cards 2 and 3 of section IV.
IV-2	1-5	I5	NA	Number of eigenvalue estimates $0 \leq NA \leq 75$
	Omit data card IV-3 if NA=0			
IV-3	1-12 13-24	2E12.5	EA_1	Real part then imaginary part of the first complex estimate.
	25-36 37-48	2E12.5	EA_2	Real part then imaginary part of second complex estimate.
	49-60 61-72	2E12.5	EA_3	Real part then imaginary part of first complex estimate.

Repeat the format of data card IV-3 until NA complex estimates have been input.

NOTE: Do not input both a complex number and its complex conjugate as estimates; only one need be input to the program in order to obtain the pair of roots.

SECTION V - CONTINUOUS SYSTEM MATRIX - REQUIRED

CARD	COLUMNS	FORMAT	CONTENTS	EXPLANATION
V-1	1-4	A4	REQEST	Input the control word MATRIX
	21-24	A4	OPTINP	Matrix input option selected by user: NOMINAL → Retain the nominal matrix defined in the previous case; omit data cards 2 thru 44 of section V. Dummy Routine. GENERAL → Create the nominal matrix using the general matrix input format input data cards 2 and 3 of section V; omit data cards 4 thru 44 of section V. RAW▽DATA → Create the nominal matrix using raw vehicle data; omit data cards 2 and 3 of section V; input data cards 4 thru 44 of section V. Dummy Routine.
	31	A1	PVAR	Non-blank indicates parameter variations of either the general input format type or raw data type.
	41	A1	PNOM	Non-blank indicates not to execute the nominal case.

NOTE: It is invalid to input PVAR as blank and PNOM as non-blank.

Data cards 2 and 3 of section V are only input if OPTINP = GENERAL

V-2	1-2	I2	II_1	Row location of first matrix coefficient value.
	3-4	I2	JJ_1	Column location of first matrix coefficient value.
	5-6	I2	KK_1	Power of S to which the first value is a coefficient.
	7-20	E14.5	VAL_1	Value of first matrix coefficient.
	21-22	I2	II_2	Same data as in columns 1-20 but for second coefficients.
	23-24	I2	JJ_2	
	25-26	I2	KK_2	
	27-40	E14.5	VAL_2	
	41-42	I2	II_3	Same data as in columns 1-20 but for third coefficients.
	43-44	I2	JJ_3	
	45-46	I_2	KK_3	
	47-60	E14.5	VAL_3	

CARD	COLUMNS	FORMAT	CONTENTS	EXPLANATION
V-2	61-62	I2	II_4	
	63-64	I2	JJ_4	Same data as in columns 1-20 but for fourth coefficient.
	65-66	I2	KK_4	
	67-80	E14.5	VAL_4	

NOTES: At least one coefficient must be defined per data card using any of the allotted fields; multiple definitions of a coefficient results in the last definition being used; all matrix coefficients are initialized to zero.

Repeat data card V-2 until all matrix coefficients have been defined.

V-3	1-80	20A4	CARD	A completely blank data card indicates termination of the nominal matrix definition using general input format.

Data cards V-4 thru V-44 are only input if OPTINP = RAW▽DATA. Dummy Routine.

SECTION VI - PARAMETER VARIATIONS - OPTIONAL INPUT

(Omit section VI if parameter PVAR = blank [card V-1])

Omit data cards VI-1 thru VI-2 if the characteristic matrix was generated by raw data vehicle parameter values.

GENERAL MATRIX PARAMETER VARIATION

CARD	COLUMNS	FORMAT	CONTENTS	EXPLANATION
VI-1	1-5	I5	NV	Number of matrix coefficients to be varied simultaneously. $1 \leq NV \leq 100$
VI-2	1-2	I2	II_1	Row location of first matrix coefficient being varied.
	3-4	I2	JJ_1	Column location of first matrix coefficient being varied.
	5-6	I2	KK_1	Power of S first matrix coefficient being varied.
	7-20	E14.5[1]	VAL_1	Varied value of the matrix coefficient.
	21-22	I2	II_2	Same data as in columns 1-20 but for second coefficient.
	23-24	I2	JJ_2	
	25-26	I2	KK_2	
	27-40	E14.5	VAL_2	
	41-42	I2	II_3	Same data as in columns 1-20 but for third coefficient.
	43-44	I2	JJ_3	
	45-46	I2	KK_3	
	47-60	E14.5	VAL_3	
	61-62	I2	II_4	Same data as in columns in 1-20 but for fourth coefficient.
	63-64	I2	JJ_4	
	65-66	I2	KK_4	
	67-80	E14.5	VAL_4	

Repeat data card VI-2 until all matrix coefficients being varied simultaneously have been specified.

SECTION VII - KEYWORD INPUT - OPTIONAL INPUT

Section VII is an optional input section which when input performs two functions:

(1) Recovery point to which the program goes if an input or execution error occurs in a previous case.

(2) Resets Nyquist data, root locus data, characteristic matrix data and user estimates to program initial conditions.

If this section is not input, then all specifications input in the preceding data case remains intact for reference in the following data case.

CARD	COLUMNS	FORMAT	CONTENTS	EXPLANATION
VII-1	1-4	A4	CARD	The data word KEY punched in columns 1-4 causes the program to identify this card as an error recovery point and to return the program to initial conditions.

3.0 METHOD

3.1 Control Systems

A control system is an arrangement of physical components connected in such a manner as to command, direct or regulate itself or another system. There are two general classifications of control systems - open loop and closed loop. An open loop control system is one in which the control action is independent of the output. A closed loop control system is one in which the control action is dependent on the output. System feedback is the property of closed loop control systems which enables the output to be compared with the system input, so that appropriate control action can be initiated as some function of the input and output. It is customary to refer to a closed loop control system as a feedback control system. Figures 3-1 and 3-2 illustrate typical block diagrams of open loop control systems and feedback control systems, respectively.

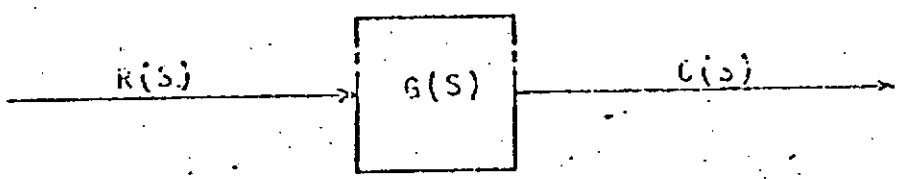


Figure 3-1 Open Loop Control System

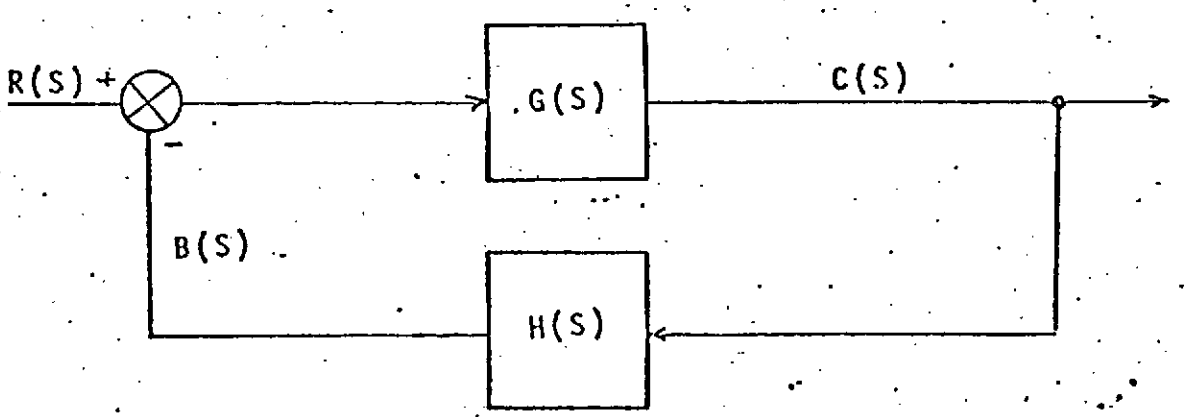


Figure 2 Feedback Control System

3.2 Representation of Feedback Control Systems

Conventional feedback control systems are continuous data systems in which all loops of the control system receive a continuous signal as input.

A linear feedback control system can be represented in several different ways: a system of differential equations, Laplace transformation of the differential equations, characteristic matrix, or block diagram. As an example, consider the following:

system of differential equations (zero initial conditions and no input signals)

$$\begin{cases} x_1 + x_3 + x_4 = 0 \\ -x_1 + \dfrac{dx_2}{dt} = 0 \\ x_2 - \dfrac{dx_3}{dt} - x_3 = 0 \\ x_2 - \dfrac{dx_4}{dt} - 2x_4 = 0 \end{cases}$$

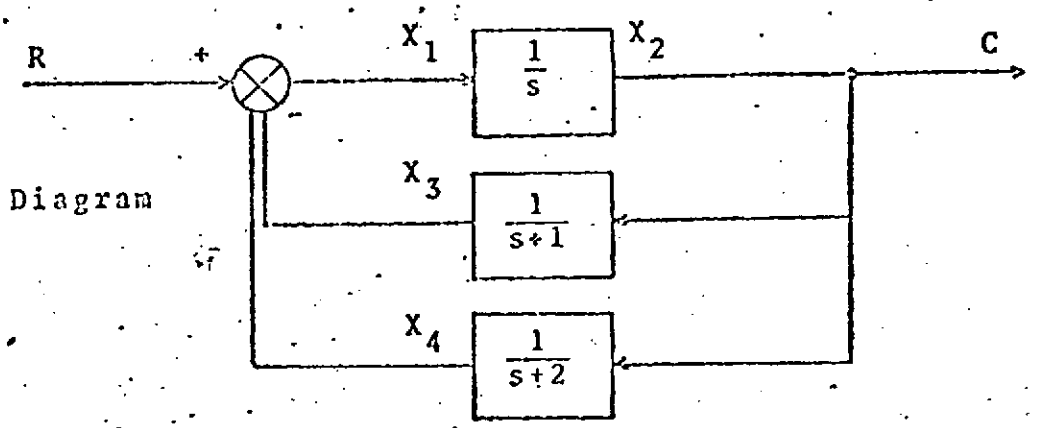
LaPlace Transformation

$$\begin{cases} X_1 + X_3 + X_4 = 0 \\ -X_1 + sX_2 = 0 \\ X_2 - (s+1)X_3 = 0 \\ X_2 - (s+2)X_4 = 0 \end{cases}$$

Characteristic Matrix

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ -1 & s & 0 & 0 \\ 0 & 1 & -(s+1) & 0 \\ 0 & 1 & 0 & -(s+2) \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Block Diagram



All are equivalent definitions of the same linear feedback control system.

A relationship fundamental to control system analysis exists between the characteristic matrix and the closed loop transfer function of a linear feedback control system. Assume the existence of a linear feedback control system as illustrated by figure 3-3.

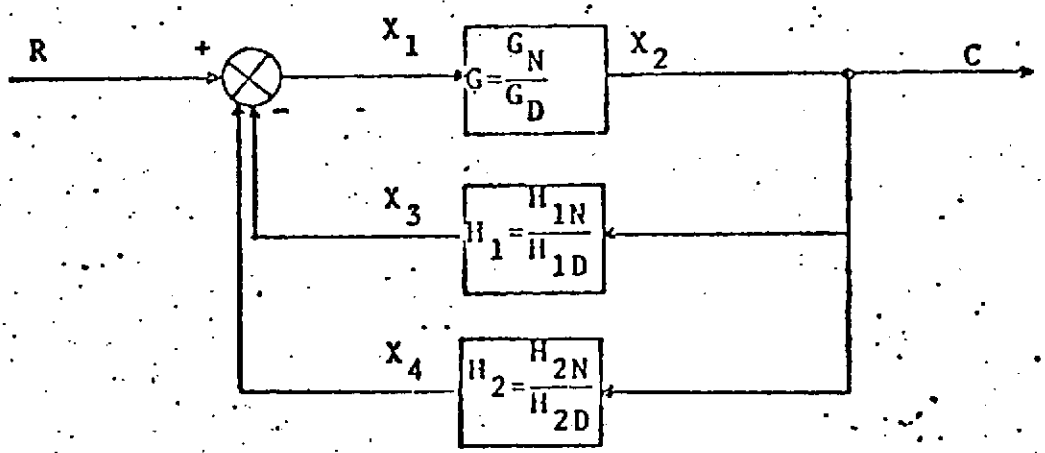


Figure 3-3 Linear Feedback Control System

3.2 Representation of Feedback Control Systems (Continued)

In reference to the block diagram, G is the forward transfer function and H_1 and H_2 are the feedback transfer functions. The quantities G_N, G_D, H_{1N}, H_{1D}, H_{2N}, and H_{2D} are polynomials in the Laplacian variable S. The closed loop transfer function, $\frac{C}{R}$, is defined as the ratio of the output signal to the input signal. The equations

$$X_1 = R - X_3 - X_4$$

$$X_2 = \frac{G_N}{G_D} X_1$$

$$X_3 = \frac{H_{1N}}{H_{1D}} X_2$$

$$X_4 = \frac{H_{2N}}{H_{2D}} X_2$$

$$C = X_2$$

of the block diagram can be used to derive the closed loop transfer function.

$$C = \frac{G_N}{G_D} (R - X_3 - X_4)$$

$$C = \frac{G_N}{G_D} \left(R - \frac{H_{1N}}{H_{1D}} C - \frac{H_{2N}}{H_{2D}} C\right)$$

$$\frac{G_D}{G_N} C = R - \frac{H_{1N}}{H_{1D}} C - \frac{H_{2N}}{H_{2D}} C$$

$$\left(\frac{G_D}{G_N} + \frac{H_{1N}}{H_{1D}} + \frac{H_{2N}}{H_{2D}} \right) C = R$$

$$\frac{C}{R} = \frac{1}{\frac{G_D}{G_N} + \frac{H_{1N}}{H_{1D}} + \frac{H_{2N}}{H_{2D}}}$$

$$= \frac{1}{\frac{G_D H_{1D} H_{2D} + G_N H_{1N} H_{2D} + G_N H_{2N} H_{1D}}{G_N H_{1D} H_{2D}}}$$

$$= \frac{G_N H_{1D} H_{2D}}{G_D H_{1D} H_{2D} + G_N (H_{1N} H_{2D} + H_{2N} H_{1D})}$$

Hence,

$$\frac{C}{R} = \frac{G_N H_{1D} H_{2D}}{G_D H_{1D} H_{2D} + G_N (H_{1N} H_{2D} + H_{2N} H_{1D})}$$

is the closed loop transfer function. Of particular importance in the study of absolute and relative stability of a closed loop control system is the location of the closed loop poles. The closed loop poles are the roots of the denominator polynomial of the closed loop transfer function and hence the roots of the equation:

$$G_D H_{1D} H_{2D} + G_N (H_{1N} H_{2D} + H_{2N} H_{1D}) = 0$$

3.2 Representation of Feedback Control Systems (Continued)

In generating the system characteristic matrix, all signals external to the closed loop are ignored. The closed loop system of equations is given by:

$$X_1 = -X_3 - X_4$$

$$X_2 = \frac{G_N}{G_D} X_1$$

$$X_3 = \frac{H_{1N}}{H_{1D}} X_2$$

$$X_4 = \frac{H_{2N}}{H_{2D}} X_2$$

Rewriting the equations yields

$$X_1 + X_3 + X_4 = 0$$

$$-G_N X_1 + G_D X_2 = 0$$

$$H_{1N} X_2 - H_{1D} X_3 = 0$$

$$H_{2N} X_2 - H_{2D} X_4 = 0$$

which can be expressed in matrix format.

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ -G_N & G_D & 0 & 0 \\ 0 & H_{1N} & -H_{1D} & 0 \\ 0 & H_{2N} & 0 & -H_{2D} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad ; \; A(s)X=0$$

The matrix polynomial A(s) is the characteristic matrix corresponding to the block diagram illustrated in figure 1. The characteristic roots of A(s) are the values of s which cause the determinant of A(s) to vanish.

$$|A(s)| = \begin{vmatrix} 1 & 0 & 1 & 1 \\ -G_N & G_D & 0 & 0 \\ 0 & H_{1N} & -H_{1D} & 0 \\ 0 & H_{2N} & 0 & -H_{2D} \end{vmatrix}$$

$$= 1 \begin{vmatrix} G_D & 0 & 0 \\ H_{1N} & -H_{1D} & 0 \\ H_{2N} & 0 & -H_{2D} \end{vmatrix} + G_N \begin{vmatrix} 0 & 1 & 1 \\ H_{1N} & -H_{1D} & 0 \\ H_{2N} & 0 & -H_{2D} \end{vmatrix}$$

$$= G_D H_{1D} H_{2D} + G_N (H_{1N} H_{2D} + H_{2N} H_{1D})$$

Note that the characteristic roots are the closed loop poles. In general it can be stated:

> The poles of the closed loop transfer function are the roots of the system characteristic matrix.

3.3 Sampled Data Control Systems

Conventional feedback control systems are continuous data systems in which all loops of the control system receive a continuous signal as input. Another classification of feedback control systems is the sampled data control system, in which one or more loops of the system receive pulsed data as the input signal. Sampled data systems originate primarily due to inherent sampling in the system or when desired system results can be obtained with intentional sampling. Radar tracking systems and time sharing systems are examples of sampled data systems with inherent sampling. Improved sensitivity and behavior and the ability to save and retransmit digitally coded signals are often the reasons for converting a previously continuous system to a sampled data system.

In sampled data systems, a continuous signal is sampled by a sampling device which outputs a sequence of pulses.

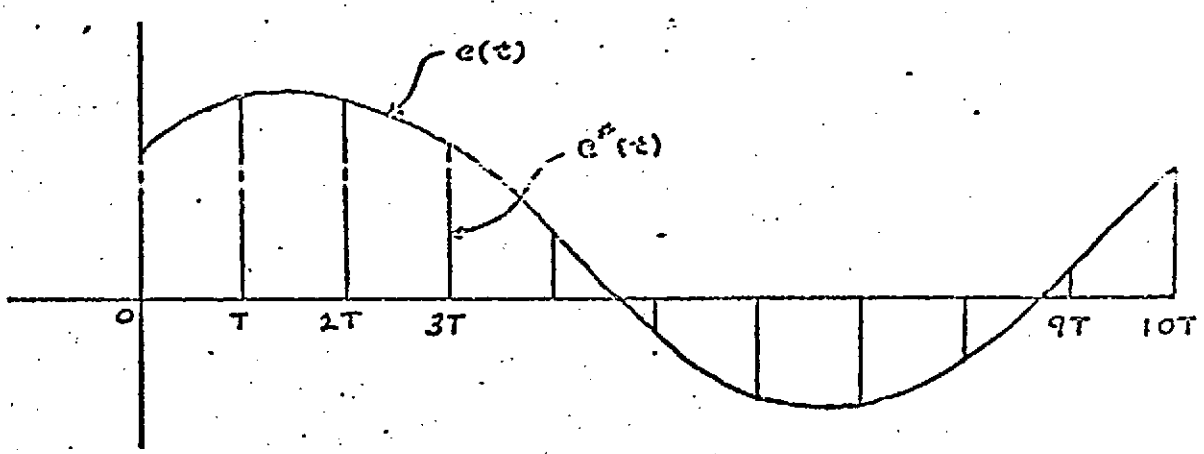


Figure 3-4 Sampled Data Signal e*(t)

The continuous signal e(t) is sampled by the sampling device, and the sampled signal e*(t) is output as a sequence of pulses (refer to Figure 3-4). In practice, the sampling device closes for a very short interval at periodic instants (t = o, T, 2T, 3T, ...) in order to sample the continuous signal. The time between two consecutive pulses is defined as the sampling period, denoted by T.

Many sampled data control systems incorporate a zero order hold circuit after the sampling device. This circuit serves to generate a step function from the pulses in which each step is of width T (see Figure 3-5).

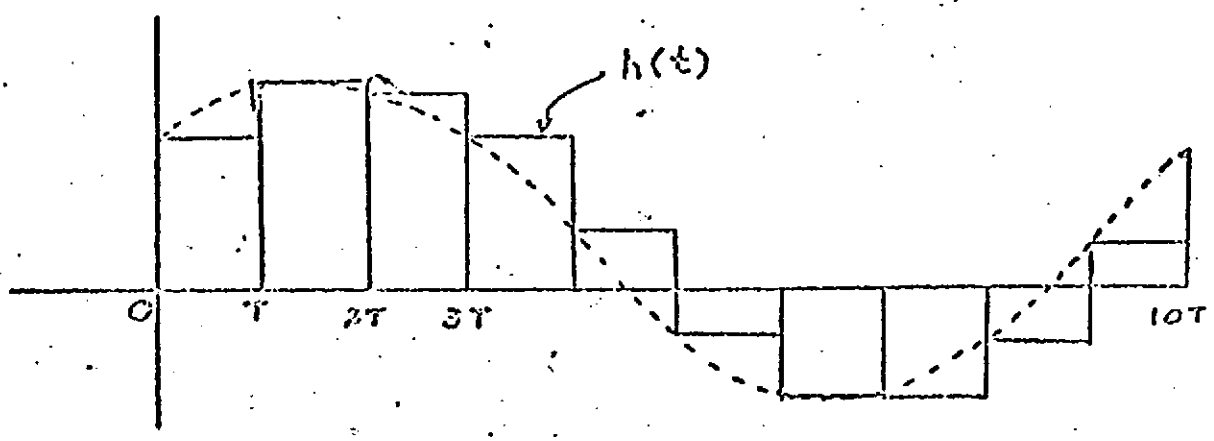


Figure 3-5 Sampled data signal h(t) with zero order hold circuit

In practice the actual waveform of the zero order hold circuit output is a series of exponential decays with large time constants. A typical block diagram of a sampled data system with a zero order hold circuit is illustrated by Figure 3-6.

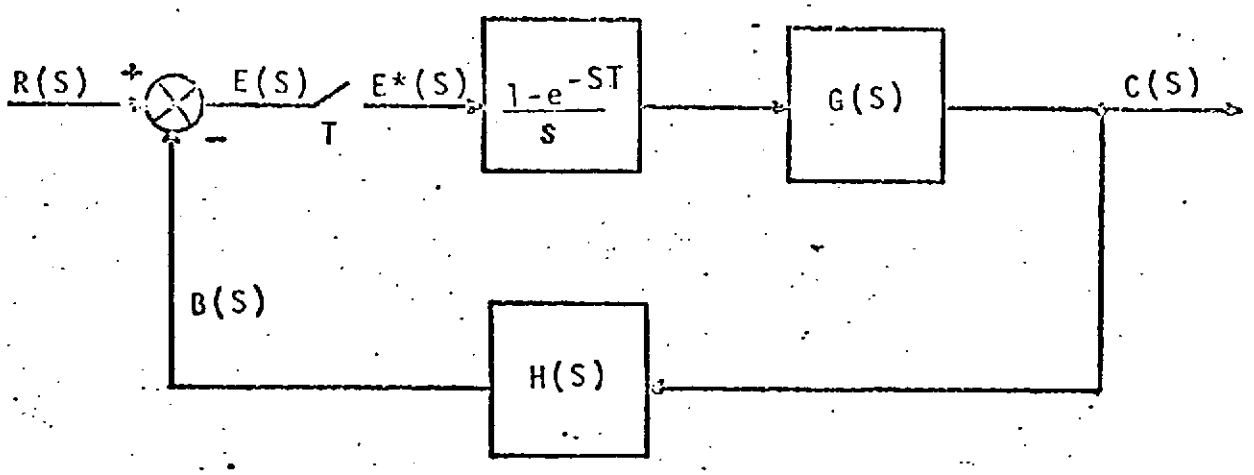


Figure 3-6 Sampled data feedback control system with zero order hold

The Laplace transformation of a zero order hold circuit is $\frac{1-e^{-ST}}{S}$.

3.3 Sampled Data Control Systems (Continued)

Stability and design analysis of continuous system feedback control systems is performed in the Laplacian variable S domain. Analysis of sampled data feedback control systems would be expected to be performed in the Laplacian variable S domain also. However, the presence of the pulsed sampled signal e*(t) makes the S domain analysis impractical. The Laplace transformation of the sampled data time function e*(t) is of the form

$$\mathcal{L}\,[e^*(t)] = E^*(s) = \sum_{n=0}^{\infty} e(nt)e^{-nTS}$$

where e(nt) denotes the continuous function e(t) at the sampling instants. Unfortunately, the factor e^{-nTS} makes E*(s) a nonalgebraic equation to which the inverse Laplace transformation and partial fraction expansion techniques are difficult to apply. The change of variable which eliminates these problems is the Z transformation:

$$Z = e^{TS} \quad \text{or} \quad S = \frac{1}{T} \ln Z$$

Substituting into the expression E*(S) yields

$$E^*\left(S=\frac{1}{T}\ln Z\right) = E(Z) = \sum_{n=0}^{\infty} e(nT)\, Z^{-n}$$

which transforms the sampled time function e*(t) into a function of Z. E(Z) is referred to as the Z-transformation of the general time function e(t):

$$E(Z) = \text{Z-transformation of } e(t) = \mathcal{Z}[e(t)].$$

The Z transformation has the limitation that the inverse Z transformation only yields correct information at the sampling instants. For this reason the Z transformation can not accurately be applied to sampled data control systems which contain a transport lag (time delay). The Saturn V flight control system has a transport lag due to computation time and analog-digital conversion time requirements within the system. This type of system can, however, be solved by a modified Z transform method. The Laplace transformation of a transport lag is

$$e^{-T_D S}$$

where T_D is the transport lag expressed in seconds. consider the sampled data control system with transport lag illustrated in Figure 3-7.

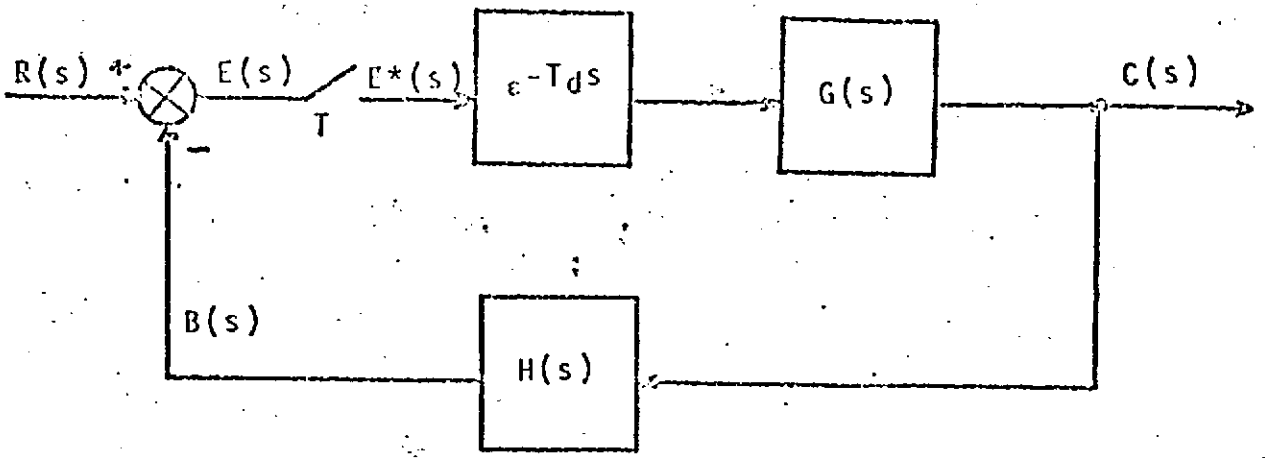


Figure 3-7 Sampled data control system with a transport lag.

The open loop transfer function of the continuous part of the illustrated control system including the transport lag is:

$$GH(S) = gh(S)\, e^{-T_D S}$$

Defining the parameter M by

$$M = 1 - \frac{T_D}{T}$$

and substituting into GH(S) yields a new expression of the continuous system open loop transfer function:

$$GH(S) = gh(S)\, e^{(M-1)\,TS}$$

The modified Z transformation of GH(S) denoted by GH(Z,M), is defined by:

$$GH(Z,M) = \mathcal{Z}_M[GH(S)]_{M=1-\frac{T_D}{T}} = Z^{-1} \sum_{}^{\infty} gh(kT+MT)\, Z^{-k}$$

3.3 Sampled Data Control Systems (Continued)

Using the modified Z transformation, design and stability analysis of sampled data control systems with transport lag can be accurately performed. To avoid confusion, the terms 'basic Z transformation' and 'modified Z transformation' will be used to distinguish between the Z transformations of sampled data control systems without transport lag and those with transport lag.

3.4 Basic and Modified Z Transformations

To further illustrate the basic Z transformation and the modified Z transformation, consider a feedback control system which has incorporated a sampling device into one of its loops. The Z transformation method requires that the system be opened at the location of the sampling device, and the Z transformation be applied to the resultant continuous system open loop transfer function. Define the following nomenclature list:

T	Sampling period
T_D	Transport lag
$GH(S)$	Continuous part of the open loop transfer function
$-(a_j+b_j i)$	Poles of GH(S)
$(c_j+d_j i)$	Residues of GH(S)
K	Leading coefficient of GH(S)

and let the continuous system open loop transfer function GH(S) be expressed by:

$$GH(S) = k\left[\frac{C_1}{S} + \frac{C_2}{S^2} + \frac{C_3}{S+a_3} + \frac{C_4+d_4 i}{S+(a_4+b_4 i)} + \frac{C_4-d_4 i}{S+(a_4-b_4 i)}\right]$$

NOTE: Due to the limitations imposed by the program (Section 1.3) and the restriction to matrix polynomials with real coefficients, all continuous system open loop transfer functions generated by the program will have terms in the partial fraction expansion only of the types expressed above.

Application of the theorem

$$\mathcal{Z}[af(S)+bg(S)] = a\,\mathcal{Z}[f(S)] + b\,\mathcal{Z}[g(S)]$$

where a, b are constants and f(S), g(S) are functions in the Laplacian variable S

to GH(S) enables the Z transformation of GH(S) to be expressed as the summation of the Z transformation of the individual terms of the partial fraction expansion of GH(S).

$$\mathcal{Z}[GH(S)] = \mathcal{Z}\, k \left[\frac{c_1}{S^2} + \frac{c_2}{S} + \frac{c_3}{S+a_3} + \frac{c_4+d_4 i}{S+(a_4+b_4 i)} + \frac{c_4-d_4 i}{S+(a_4-d_4 i)} \right]$$

$$= \mathcal{Z}\left[\frac{kc_1}{S^2}\right] + \mathcal{Z}\left[\frac{kc_2}{S}\right] + \mathcal{Z}\left[\frac{kc_3}{S+a_3}\right] + \mathcal{Z}\left[\frac{k(c_4+d_4 i)}{S+(a_4+b_4)} + \frac{k(c_4-d_4 i)}{S+(a_4-b_4)}\right]$$

For systems which do not contain a transport lag, the basic Z transformation can be applied:

$$\mathcal{Z}\left[\frac{kc}{S^2}\right] = kc\,\mathcal{Z}\left[\frac{1}{S^2}\right] = kc\left[\frac{TZ}{(Z-1)^2}\right] = kcT\left[\frac{Z}{Z^2-2Z+1}\right].$$

$$\mathcal{Z}\left[\frac{kc}{S}\right] = kc\,\mathcal{Z}\left[\frac{1}{S}\right] = kc\left[\frac{Z}{Z-1}\right]$$

$$\mathcal{Z}\left[\frac{kc}{S+a}\right] = kc\,\mathcal{Z}\left[\frac{1}{S+a}\right] = kc\left[\frac{Z}{Z-e^{-aT}}\right]$$

$$\mathcal{Z}\left[\frac{k(c+di)}{S+(a+bi)} + \frac{k(c-di)}{S+(a-bi)}\right] = \mathcal{Z}\left[\frac{2kc(S+a) + 2kbd}{(S+a)^2 + b^2}\right]$$

$$= 2kc\ \mathcal{Z}\left[\frac{S+a}{(S+a)^2 + b^2}\right] + 2kbd\ \mathcal{Z}\left[\frac{1}{(S+a)^2+b^2}\right]$$

$$= 2kc\left[\frac{Z^2 - (e^{-aT}\cos(bT))Z}{Z^2 - (2e^{-aT}\cos(bT))Z + e^{-2aT}}\right] +$$

$$2kbd\left[\frac{1}{b}\ \frac{(e^{-aT}\sin(bT))Z}{Z^2 - (2e^{-aT}\cos(bT))Z + e^{-2aT}}\right]$$

$$= \frac{(2kc)Z^2 + (2k(d\sin(bT) - c\cos(bT))e^{-aT})Z}{Z^2 - (2e^{-aT}\cos(bT))Z + e^{-2aT}}$$

For systems which include a transport lag, the modified Z transformation is applied:

$$\mathcal{Z}_M\left[\frac{kc}{S^2}\right] = kc\,\mathcal{Z}_M\left[\frac{1}{S^2}\right] = kc\left[\frac{MT}{Z-1} + \frac{T}{(Z-1)^2}\right]$$

$$= kc\left[\frac{MT\,(Z-1)+T}{(Z-1)^2}\right]$$

$$= kc\left[\frac{(MT)Z + (T-MT)}{Z^2-2Z+1}\right]$$

$$\mathcal{Z}_M\left[\frac{kc}{S}\right] = kc\,\mathcal{Z}_M\left[\frac{1}{S}\right] = kc\left[\frac{1}{Z-1}\right] = \frac{kc}{Z-1}$$

$$\mathcal{Z}_M\left[\frac{kc}{S-a}\right] = kc\,\mathcal{Z}_M\left[\frac{1}{S-a}\right] = kc\left[\frac{e^{-aMT}}{Z-e^{-aT}}\right] = \frac{kce^{-aMT}}{Z-e^{-aT}}$$

$$\mathcal{Z}_M\left[\frac{k(c+di)}{S+(a+bi)} + \frac{k(c-di)}{S+(a-bi)}\right] = \mathcal{Z}_M\left[\frac{2kc(S+a) + 2kbd}{(S+a)^2 + b^2}\right]$$

$$= 2kc\ \mathcal{Z}_M\left[\frac{S+a}{(S+a)^2+b^2}\right] + 2kbd\ \mathcal{Z}_M\left[\frac{1}{(S+a)^2+b^2}\right]$$

$$= \frac{2kc\ e^{-aMT}\left[\cos(MbT)Z - e^{-aT}\cos((1-M)bT)\right]}{Z^2 - (2e^{-aT}\cos(bT))Z + e^{-2aT}} +$$

$$2kbd\left[\frac{e^{-aMT}}{b}\ \frac{\left[\sin(MbT)\ Z + e^{-aT}\sin((1-M)bT)\right]}{Z^2 - (2e^{-aT}\cos(bT))Z + e^{-2aT}}\right]$$

$$= \frac{\left[2k\ e^{-aMT}(c\cos(MbT) + d\sin(MbT))\right]Z}{Z^2 - (2e^{-aT}\cos(bT))Z + e^{-2aT}} +$$

$$\frac{2ke^{-aMT}e^{-aT}(c\cos((1-M)bT) + d\sin((1-M)bT)}{Z^2 - (2e^{-aT}\cos(bT))Z + e^{-2aT}}$$

Assume a zero order hold device has been incorporated into the system at the point of the open loop specification. The open loop transfer function of the continuous system with zero order hold is denoted by $G_{ho}GH(S)$.

$$\left[G_{ho}GH(S)\right] = \left[\frac{1-e^{-TS}}{S}\,GH(S)\right]$$

$$= \left\{\left[(1-e^{-TS})\right](k)\left[\frac{c_1}{S^3} + \frac{c_2}{S^2} + \frac{c_3}{S} + \frac{c_4}{S+a_4} + \frac{c_5+d_5 i}{S+(a_5+b_5 i)} + \frac{c_5-d_5 i}{S+(a_5-b_5 i)}\right]\right\}$$

$$\mathcal{Z}\left[G_{ho}GH(S)\right] = \mathcal{Z}\left\{\left[(1-e^{-TS})\right](k)\left[\frac{c_1}{S^3} + \frac{c_2}{S^2} + \frac{c_3}{S} + \frac{c_4}{S+a_4} + \frac{c_5 d_5 i}{S+(a_5+b_5 i)} + \frac{c_5-d_5 i}{S+(a_5-b_5 i)}\right]\right\}$$

$$= (1-Z^{-1})\,\mathcal{Z}\left[\frac{kc_1}{S^3} + \frac{kc_2}{S^2} + \frac{kc_3}{S} + \frac{kc_4}{S+a_4} + k\frac{c_5+d_5 i}{S+(A_5+b_5 i)} + k\frac{c_5-d_5 i}{S+(a_5-b_5 i)}\right]$$

$$= \left(\frac{Z-1}{Z}\right)\left\{\mathcal{Z}\left[\frac{kc_1}{S^3}\right] + \mathcal{Z}\left[\frac{kc_2}{S^2}\right] + \mathcal{Z}\left[\frac{kc_3}{S}\right] + \mathcal{Z}\left[\frac{kc_4}{S+a_4}\right] + \mathcal{Z}\left[k\frac{c_5+d_5 i}{S+(a_5+b_5 i)} + k\frac{c_5-d_5 i}{S+(a_5-b_5 i)}\right]\right\}$$

For systems which do not contain a transport lag, the basic Z transformation is applied:

$$\frac{(Z-1)}{Z}\,\mathcal{Z}\left[\frac{kc}{S^3}\right] = \frac{(Z-1)}{Z}\,kc\,\mathcal{Z}\left[\frac{1}{S^3}\right]$$

$$= \frac{(Z-1)}{Z}\,kc\left[\frac{T^2}{2}\,\frac{Z\,(Z+1)}{(Z-1)^3}\right]$$

$$= \frac{kc\,T^2}{2}\cdot\frac{(Z+1)}{(Z-1)^2}$$

$$= \frac{kc\,T^2}{2}\,\frac{(Z+1)}{Z^2-2Z+1}$$

$$\frac{(Z-1)}{Z}\,\mathcal{Z}\left[\frac{kc}{S^2}\right] = \frac{(Z-1)}{Z}\left[kc\,T\,\frac{Z}{(Z-1)^2}\right] = \frac{kc\,T}{Z-1}$$

$$\frac{(Z-1)}{Z}\,\mathcal{Z}\left[\frac{kc}{S}\right] = \frac{(Z-1)}{Z}\cdot\left[kc\,\frac{Z}{Z-1}\right] = kc$$

$$\frac{(Z-1)}{Z}\,\mathcal{Z}\left[\frac{kc}{(S+a)}\right] = \frac{(Z-1)}{Z}\left[\frac{kc\,Z}{(Z-e^{-aT})}\right] = kc\,\frac{(Z-1)}{(Z-e^{-aT})}$$

$$\frac{(Z-1)}{Z}\mathcal{Z}\left[\frac{k(c+di)}{S+(a+bi)}+\frac{k(c-di)}{S+(a-bi)}\right]=\frac{(Z-1)}{Z}\left[\frac{(2kc)Z^2+[2k\ (d\sin(bT)-c\cos(bT))\ e^{-aT}]Z}{Z^2-(2\ e^{-aT}\cos(b\ T)\ )Z+e^{-2aT}}\right]$$

$$=\frac{(2kc)Z^2+[2\ k\ (d\sin(bT)-c\cos(bT))\ e^{-aT}-2\ kc]Z}{Z^2-(2\ e^{-aT}\cos\ (bT))Z+e^{-2aT}}+$$

$$\frac{-2\ k\ (d\sin(bT)-c\cos(bT))\ c^{-aT}}{Z^2-(2\ e^{-aT}\cos(bT))Z+e^{-2aT}}$$

For systems which include a transport lag, the modified Z transformation is applied:

$$\frac{(Z-1)}{Z}\mathcal{Z}\,M\left[\frac{kc}{S^3}\right]=kc\ \frac{(Z-1)}{Z}\mathcal{Z}\,M\left[\frac{1}{S^3}\right]$$

$$=\frac{kc\ (Z-1)\ T^2}{Z}\left[\frac{M}{(Z-1)}+\frac{2M+1}{(Z-1)^2}+\frac{2}{(Z-1)^3}\right]$$

$$=\frac{kc\ T^2}{Z}\left[\frac{M^2(Z-1)^2+(2M+1)\ (Z-1)+2}{(Z-1)^2}\right]$$

$$=kc\ T^2\left[M^2\ \frac{Z^2+(-2\ M^2+2\ M+1)Z+(M^2-2\ M+1)}{Z^3-2Z^2+Z}\right]$$

$$\frac{(Z-1)}{Z}\ \mathcal{Z}_M\left[\frac{kc}{S^2}\right] = \frac{(Z-1)}{Z}\left[\frac{(MT)Z + (T-MT)}{Z^2-2Z+1}\right]$$

$$= \frac{(MT)Z + (T-MT)}{Z^2 - Z}$$

$$\frac{(Z-1)}{Z}\ \mathcal{Z}_M\left[\frac{kc}{S}\right] = \frac{(Z-1)}{Z}\left[\frac{kc}{(Z-1)}\right]$$

$$= \frac{kc}{Z}$$

$$\frac{(Z-1)}{Z}\ \mathcal{Z}_M\left[\frac{kc}{S-a}\right] = \frac{(Z-1)}{Z}\left[\frac{kc\ e^{-aMT}}{(Z-e^{-aMT})}\right]$$

$$= \frac{kc\ e^{-aMT}\ Z - kc\ e^{-aMT}}{Z^2 - e^{-aT}Z}$$

$$\frac{(Z-1)}{Z}\mathcal{Z}_M\left[\frac{k\,(c+di)}{S+(a+bi)} + \frac{k\,(c-di)}{S+(a-bi)}\right] = \frac{(Z-1)}{Z}\left[\frac{2\,k\,e^{-aMT}\,(c\cos(M\,b\,T))Z + }{Z^2-2e^{-aT}\cos(bT)\,Z + e^{-2aT}}\right.$$

$$\left.\frac{2\,k\,e^{-aMT}\,e^{-aT}(-c\cos((1-M)\,b\,T)+d\sin((1-M)\,b\,T))}{Z^2-2e^{-aT}\cos(bT)\,Z + e^{-2aT}}\right]$$

$$= \frac{\begin{aligned}&2\,k\,e^{-aMT}(c\cos(M\,b\,T)+d\sin(M\,b\,T))\,Z^2\\ &+2\,k\,e^{-aMT}[e^{-aT}(-c\cos(1-M)\,b\,T)+d\sin((1-M)b\,T)]Z\\ &-2\,k\,e^{-aMT}[(c\cos(M\,b\,T)+d\sin(M\,b\,T))]\,Z\\ &-2\,k\,e^{-aMT}\,e^{-aT}(-c\cos((1-M)\,b\,T)+d\sin((1-M)\,b\,T))\end{aligned}}{Z^3-(2\,e^{-aT}\cos(b\,T))Z^2+e^{-2aT}\,Z}$$

3.5 Root Locus

Design and stability analysis of continuous data control systems is based on the pole-zero configuration of the system transfer function in the S plane. For a stable system, all the closed loop poles must lie in the left-half S plane. An analogous situation arises for sampled data control systems. The Z transformation maps all points in the left-hand S plane into the interior of the unit circle in the Z plane and all the points in the right-hand S plane into the exterior of the unit circle in the Z plane. As a consequence, design and stability analysis of sampled data systems is based on the pole-zero configuration of the system transfer function with respect to the unit circle. For a stable system all closed loop poles must be within the unit circle.

Sampled data root locus in BHA0369 is restricted in application to only those sample data systems which incorporate a zero order hold circuit. Gain root locus is based upon the introduction of a gain factor K at the sampling device location in the closed loop block diagram. The locus of the closed loop poles are plotted as a function of the gain factor as K varies from zero to infinity; the resultant plot being the gain root locus plot. By replacing the gain factor K by $e^{j\theta}$ and varying θ from 0 to 360 degrees, phase root locus plots can be generated. A special root locus called sample period root locus can also be obtained. For each sample period specified, the program computes the sampled data system nominal closed loop poles. The locus of these poles plotted as a function of the sample period constitute the sample period root locus.

Computational techniques used in obtaining sampled data root locus requires the analysis be performed on a modified system. Isolated and extremely stable roots appear as roots at the origin in the Z plane. Such roots show little movement and affect insignificantly the movement of other system roots. Because of the computational difficulties involved in obtaining multiple roots which frequently occur at the origin in sampled data root locus, the BHA0369 program removes from the system all the roots at the origin. Working with the partial fraction expansion of the system, transfer function, the sampled data closed loop poles are inspected and those which have a magnitude less than .0001 are removed from the system. More specifically, the program deletes the term associated with the indicated pole from the system Z domain partial fraction expansion. In addition, rather than obtaining root locus results from the open loop sampled data transfer function

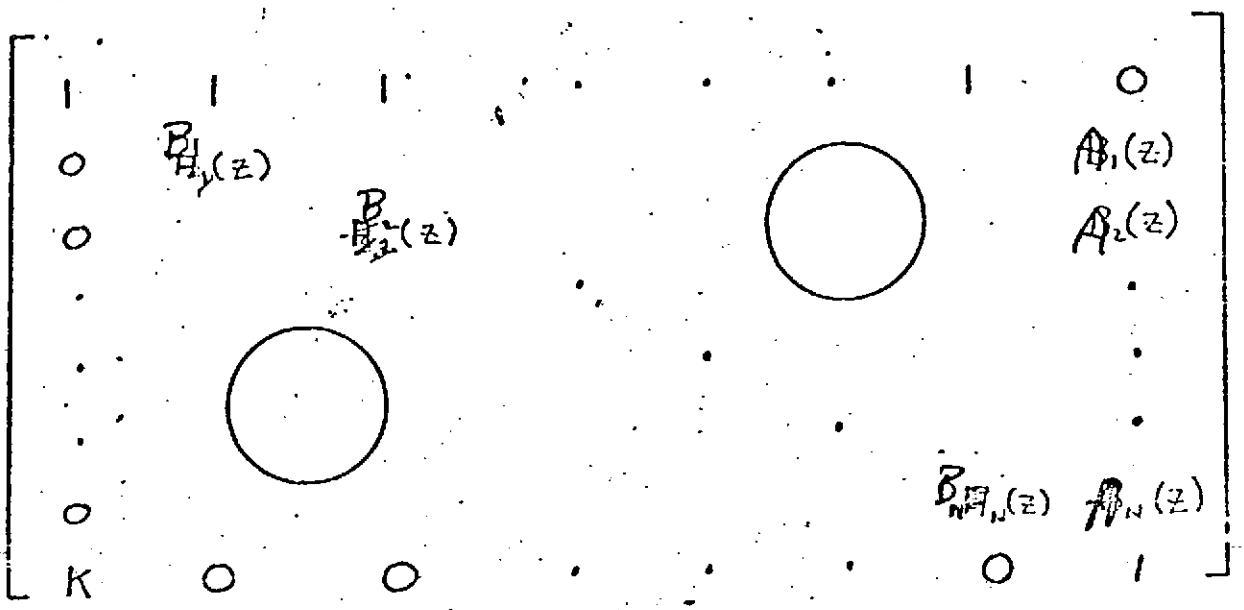
3.5 Root Locus (Continued)

i.e.

$$GH(Z) = \frac{A_1(Z)}{B_1(Z)} + \frac{A_2(Z)}{B_2(Z)} + \ldots + \frac{A_N(Z)}{B_N(Z)} = \frac{P(Z)}{Q(Z)}$$

the program uses the companion matrix.

i.e.



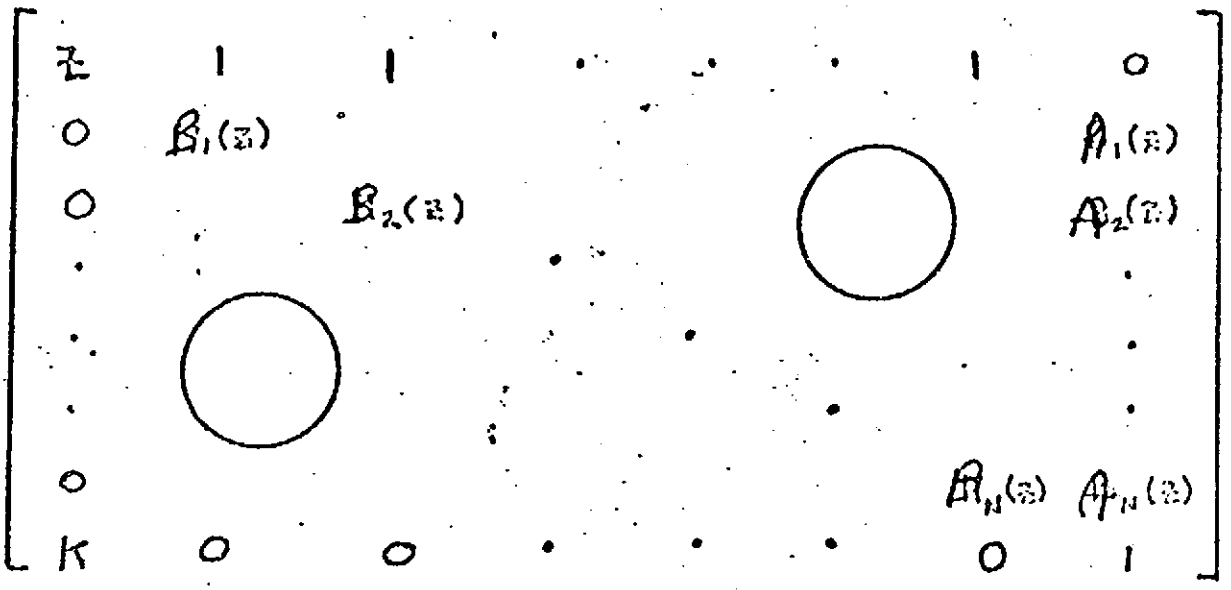
The characteristic polynomial of the companion matrix yields the expression necessary for root locus analysis.

$$D_{eT}(C) = B_1(Z)B_2(Z)\ldots B_N(Z)$$

$$+ K[A_1(Z)B_2(Z)\ldots B_N(Z) + \ldots + B_1(Z)B_{N-1}(Z)A_N(Z)]$$

$$= Q(Z) + K\,P(Z)$$

For systems which have a transport lag, a slight modification must be made to the companion matrix.



Referring to the modified Z-transformation of sampled data systems with a zero order hold circuit (section 2.2.4), it can be seen that $B_1(Z)$, $B_2(Z)$,...,$B_N(Z)$ would each have a root at the origin thereby generating n open loop poles at the origin. As discussed previously in this section, such roots are removed from the system in root locus computations. Modification of the (1,1) element in the companion effectively factors out the undesired roots at the origin.

3.6 Nyquist Frequency Response

Another analysis tool of the program is open loop frequency response employing the Nyquist criteria.

Nyquist analysis is a graphical procedure for determining absolute and relative stability of a closed loop control system in terms of frequency response of open loop transfer functions. In the BHA0369 program, the system is opened at the sampling device location.

In order to apply conventional nyquist techniques to sampled data control systems, another change of variable is required. The bi-linear transformation to the R domain

$$Z = \frac{R+1}{R-1}$$

is the desired transformation. The R transformation maps the interior of the Z-plane unit circle into the left half R-plane and the exterior of the Z-plane unit circle into the right half R-plane, thus enabling conventional Nyquist techniques to be applied. As an example of the transformation, consider the open loop transfer function

$$GH(Z) = \frac{AZ+b}{CZ^2+dZ+e}$$

where a, b, c, d, and e are constants.

$$GH(Z=\frac{R+1}{R-1}) = \frac{a\left(\frac{R+1}{R-1}\right) + b}{c\left(\frac{R+1}{R-1}\right)^2 + d\left(\frac{R+1}{R-1}\right) + e}$$

$$= \frac{a\,(R+1)(R-1) + b\,(R-1)^2}{c\,(R+1)^2 + d\,(R+1)(R-1) + e\,(R-1)^2}$$

$$= \frac{a\,(R^2-1) + b\,(R^2-2R+1)}{c(R^2+2R+1) + d\,(R^2-1) + e\,(R^2-2R+1)}$$

$$= \frac{(a+b)\,R^2 + (-2b)\,R + (b-a)}{(c+d+e)R^2 + (2c-2e)\,R + (c-d-e)}$$

Conventional Nyquist techniques can now be applied to the sampled data open loop transfer function.

3.6 Nyquist Frequency Response (Continued)

A few definitions are prerequisite to a discussion of the Nyquist computations. As it is a function of a complex variable, a transfer function GH(R) is a complex number and may be represented in polar form $GH(R) = re^{i\theta}$. The number r is referred to as the amplitude of the function; the number θ is the phase angle or phase; the combination $re^{i\theta}$ is the gain. There is an associated decibel value; it is 20 times the common (base 10) logarithm of the amplitude.

The essential computation involved is the evaluation of the function GH(R) over an interval of values of the independent variable. An obvious procedure is to calculate $R_1 = i\omega_1$, $GH_1 = GH(R_1)$, then using some frequency increment $\delta\omega$, obtain $\omega_2 = \omega_1 + \delta\omega$, $R_2 = i\omega_2$, $GH_2 = GH(R_2)$, and so on. The constant increment is objectionable for two reasons: A proper choice presupposes some knowledge of the behavior of GH(R); and no single increment will generally suffice over the desired range, a large value possibly causing regions of sudden variation to be bypassed, a small value necessitating excessive computer processing. Both accuracy and speed are achieved by enabling the program to assign the frequency increment; initially it is set to a certain (input) percentage of the first frequency. When GH(R) is changing rapidly, a small increment is used; when GH(R) is varying slowly, a large increment is used. The increment is decreased by halving its value until it becomes less than a certain constant (.0002 or .00002) times the current value of the frequency. The increment is increased by doubling its value until it becomes greater than the abovementioned percentage of the current frequency. These limits set on the minimum and maximum increments ensure that the calculations will neither require excessive computer time nor omit points of interest. A measure of the variation of GH(R) is available in the variation of its phase angle. The phase shift of a function is the difference between two phase angles corresponding to two successive values of the frequency; the results of certain phase shift tests determine how the frequency increment is modified. Two input quantities, a minimum and maximum phase shift, define acceptable variations. When the phase shift tested is less than the maximum and greater than the minimum, the computation continues with the current value of the frequency increment. If the phase shift is greater than the maximum, the increment is decreased when possible (that is, when it is greater than .0002 or .00002 of the current frequency value). If the phase shift is less than the minimum, the increment

is increased when possible (that is, when it is less than the certain percentage of the current frequency value). In such a manner, computation proceeds until a frequency is processed which is equal to or greater than some terminal value.

The term frequency interval or interval is used to represent the range of values which can be assumed by the variable ω. Such an interval is defined by specifying its end points, some starting and stopping frequencies. The program can process more than one interval. The intervals are completely independent; the stopping frequency of a particular one is not required to be less than the starting frequency of the next. The quantities used in determining the increment, the maximum percentage and the allowable phase shifts, are unique for each interval and may be varied by specifying more than one interval.

3.7 Program Computation Steps

The following is the step-by-step analysis procedure employed in the Sampled Data Analysis Program.

1. Input to the computer the S-plane system characteristic equations.

2. The S-plane matrix is opened for example in the AO channel (OP-AO), and S-plane roots (numerator and denominator of (OP-AO) are found.

   i.e., $OP\text{-}AO(S) = \dfrac{K(S+N_1)(S+N_2)\ldots}{(S+D_1)(S+D_2)\ldots}$

3. The S-plane roots of OP-AO are expanded by series expansion.

   i.e., $OP\text{-}AO = K\left[\dfrac{A}{S+D_1} + \dfrac{BS+C}{S^2+D_2S+D_3} + \ldots\right]$

4. From the series expansion the Z-transform is taken of each term.

   i.e., $OP\text{-}AO(Z) = K\mathcal{Z}\left[\dfrac{A}{S+D_1}\right] + K\mathcal{Z}\left[\dfrac{BS+C}{S^2+D_2S+D_3}\right] + \ldots$

5. The zero-order-hold transfer function is $\dfrac{1-e^{-ST}}{S}$.

   Thus, OP-AO(S) with zero-order-hold becomes:

   $$\overline{OP\text{-}AO(S)} = \frac{1-e^{-ST}}{S}\ OP\text{-}AO(S).$$

6. Taking the modified or basic Z transformation depending on the transport lag option;

   $$\overline{OP\text{-}AO(Z)} = (1-Z^{-1})\,\mathcal{Z}\left[\frac{OP\text{-}AO(S)}{S}\right]$$

7. Next, the bilinear transformation of $z = \dfrac{r+1}{r-1}$ is substituted for z in $\overline{OP\text{-}AO(Z)}$

   i.e., $\overline{OP\text{-}AO(Z)}\ \Big|_{Z=\frac{r+1}{r-1}} = \overline{OP\text{-}AO(r)}$

8. For freq. response, r is set equal to $j\omega_r$

   i.e., $\overline{OP\text{-}AO(r)}\ \Big|_{r=j\omega_r} = \overline{OP\text{-}AO(j\omega_r)}$

3.7 Program Computation Steps (Continued)

9. Gain and phase are plotted for values of ω_r.

10. ω_r is related to (S-plane freq.) by

$$\omega_r = -\cot\left(\frac{\omega T}{2}\right)$$

11. $\overline{1+KOP-AO(Z)}$ is put into matrix form and rooted to give a root locus as a function of system gain K.

3.8 Continuous System Open Loop Transfer Function

An intermediate step in the BHA0369 program requires the computation of the continuous system open loop transfer function. Given the continuous system characteristic matrix polynomial and the matrix location at which to open the loop, the program must compute the open loop zeros, open loop poles, and associated leading coefficients.

$$\text{i.e.,} \quad GH(S) = \frac{P(S)}{Q(S)} = K \frac{\prod_{i=1}^{p} (S-a_i)}{\prod_{j=1}^{q} (S-b_j)} = \frac{K_p \prod_{i=1}^{p} (S-a_i)}{K_q \prod_{j=1}^{q} (S-b_j)}$$

Associated with each computation is a specific matrix polynomial.

Consider the open loop transfer function illustrated by figure 3-8.

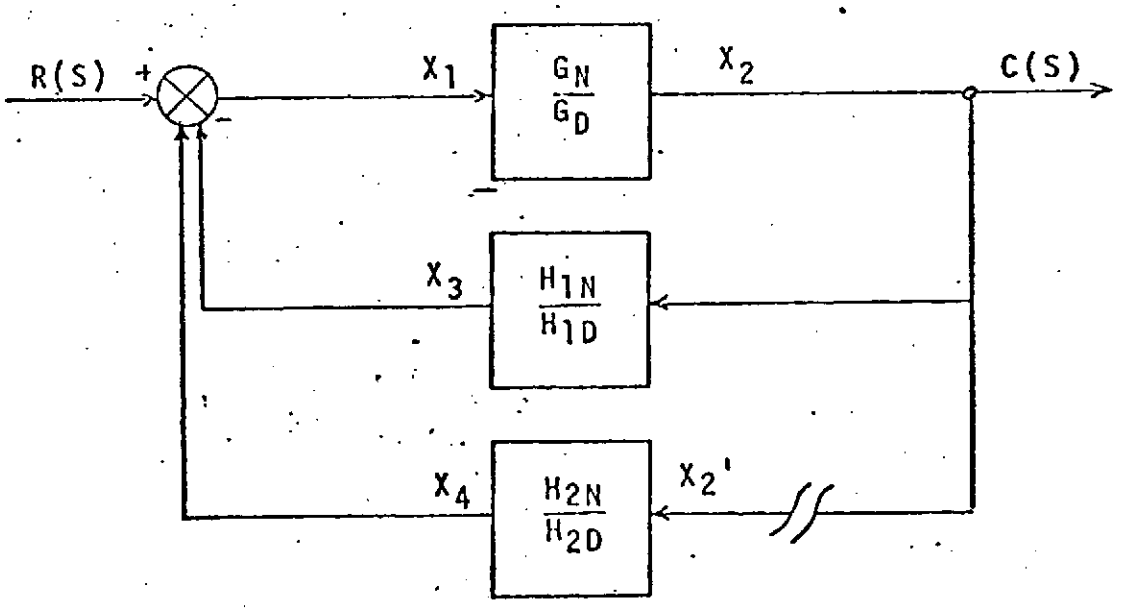


Figure 3-8 Open Loop Transfer Function

3.8 Continuous System Open Loop Transfer Function (Continued)

The open loop transfer function corresponding to the block diagram is $\frac{X_2}{X_2'}$.

The following system equations can be derived for the open loop transfer function illustrated in Figure 3-8.

$$X_1 = -X_3 - X_4 \qquad X_3 = \frac{H_{1N}}{H_{1D}} X_2$$

$$X_2 = \frac{G_N}{G_D} X_1 \qquad X_4 = \frac{H_{2N}}{H_{2D}} X_2'$$

which can be rewritten:

3.8 Continuous System Open Loop Transfer Function (Continued)

$$X_1 + X_3 + X_4 = 0$$

$$-G_N X_1 + G_D X_2 = 0$$

$$H_{1N} X_2 - H_{1D} X_3 = 0$$

$$H_{2D} X_4 = H_{2N} X_2'$$

The corresponding matrix representation is

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ -G_N & G_D & 0 & 0 \\ 0 & H_{1N} & -H_{1D} & 0 \\ 0 & 0 & 0 & H_{2D} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ H_{2N} \end{bmatrix} X_2'$$

which can be rewritten:

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ -G_N & G_D & 0 & 0 \\ 0 & H_{1N} & -H_{1D} & 0 \\ 0 & 0 & 0 & H_{2D} \end{bmatrix} \begin{bmatrix} X_1/X_2' \\ X_2/X_2' \\ X_3/X_2' \\ X_4/X_2' \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ H_{2N} \end{bmatrix}$$

Applying Cramer's rule and solving for the open loop transfer function $\frac{X_2}{X_2'}$:

$$\frac{X_2}{X_2'} = \frac{\begin{vmatrix} 1 & 0 & 1 & 1 \\ -G_N & 0 & 0 & 0 \\ 0 & 0 & -H_{1D} & 0 \\ 0 & H_{2N} & 0 & H_{2D} \end{vmatrix}}{\begin{vmatrix} 1 & 0 & 1 & 1 \\ -G_N & G_D & 0 & 0 \\ 0 & H_{1N} & -H_{1D} & 0 \\ 0 & 0 & 0 & H_{2D} \end{vmatrix}}$$

$$= \frac{G_N H_{2N} H_{1D}}{-G_D H_{1D} H_{2D} - G_N H_{1N} H_{2D}}$$

$$= - \frac{G_N H_{2N} H_{1D}}{G_D H_{1D} H_{2D} + G_N H_{1N} H_{2D}}$$

Because the transfer function illustrated in Figure3-8 employs negative feedback, then by convention, the transfer function desired is the negative of the above. Hence,

$$\frac{X_2}{X_2'} = \frac{G_N H_{2N} H_{1D}}{G_D H_{1D} H_{2D} + G_N H_{1N} H_{2D}} \qquad \text{(Equation 1)}$$

For the computation of $\frac{X_2}{X_2'}$ as shown by figure3-8, a gain factor K is incorporated in the location at which the system is opened.

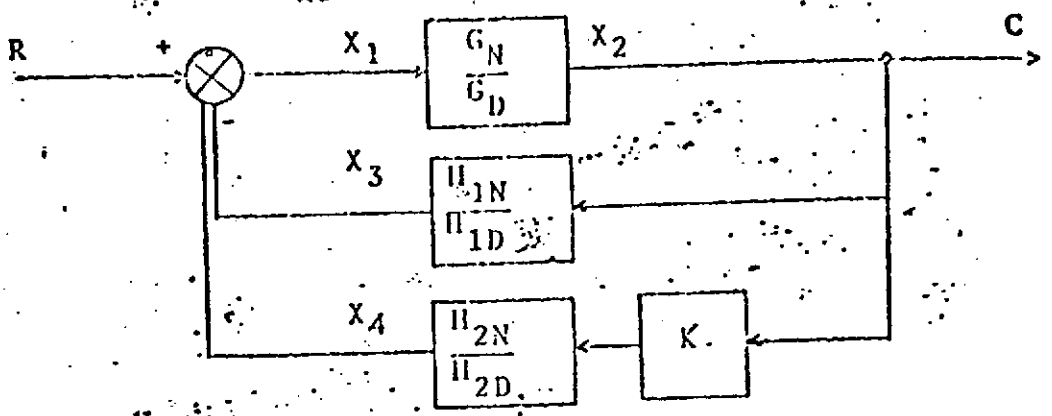


Figure 3-9 Closed Loop System

For the closed loop system illustrated in Figure 3-9, the closed loop transfer function is

$$\frac{C}{R} = \frac{G_N H_{1N} H_{2N}}{G_D H_{1D} H_{2D} + G_N (H_{1N} H_{2D} + K H_{2N} H_{1D})}$$

where the closed loop poles are the roots of the expression:

$$G_D H_{1D} H_{2D} + G_N (H_{1N} H_{2D} + K H_{2N} H_{1D}) = 0$$

The system characteristic matrix can be formed from the following equations:

$$X_1 = -X_3 - X_4 \qquad X_3 = \frac{H_{1N}}{H_{1D}} X_2$$

$$X_2 = \frac{G_N}{G_D} X_1 \qquad X_4 = K \frac{H_{2N}}{H_{2D}} X_2$$

3.8 Continuous System Open Loop Transfer Function (Continued)

Rewriting the equations yields

$$X_1 + X_3 + X_4 = 0$$

$$-G_N X_1 + G_D X_2 = 0$$

$$H_{1N} X_2 - H_{1D} X_3 = 0$$

$$KH_{2N} X_2 - H_{2D} X_4 = 0$$

which gives the characteristic matrix:

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ -G_N & G_D & 0 & 0 \\ 0 & H_{1N} & -H_{1D} & 0 \\ 0 & KH_{2N} & 0 & -H_{2D} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The characteristic polynomial is

$$\begin{vmatrix} 1 & 0 & 1 & 1 \\ -G_N & G_D & 0 & 0 \\ 0 & H_{1N} & -H_{1D} & 0 \\ 0 & KH_{2N} & 0 & -H_{2D} \end{vmatrix} = G_D H_{1D} H_{2D} + G_N (H_{1N} H_{2D} + KH_{2N} H_{1D})$$

making the characteristic equation

$$G_D H_{1D} H_{2D} + G_N (H_{1N} H_{2D} + K H_{2N} H_{1D}) = 0$$

which yields the closed loop poles as its roots. The characteristic equation can be rewritten as follows:

$$(G_D H_{1D} H_{2D} + G_N H_{1N} H_{2D}) + K G_N H_{2N} H_{1D} = 0$$

Note that when K = 0 the characteristic equation becomes

$$G_D H_{1D} H_{2D} + G_N H_{1N} H_{2D} = 0$$

which is the equation of the open loop poles. The corresponding matrix polynomial is

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ -G_N & G_D & 0 & 0 \\ 0 & H_{1N} & -H_{1D} & 0 \\ 0 & 0 & 0 & H_{2D} \end{bmatrix}$$

which has the characteristic polynomial

$$Q(S) = \begin{vmatrix} 1 & 0 & 1 & 1 \\ -G_N & G_D & 0 & 0 \\ 0 & H_{1N} & -H_{1D} & 0 \\ 0 & 0 & 0 & -H_{2D} \end{vmatrix}$$

3.8 Continuous System Open Loop Transfer Function (Continued)

$$= G_D H_{1D} H_{2D} + G_N H_{1N} H_{2N}$$

$$= K_D \prod_{J=1}^{q} (S - b_J)$$

which is the polynomial of the open loop poles.

Manipulating the characteristic polynomial, the following relationship is found,

$$\lim_{K \to \infty} \frac{1}{K} \left(G_D H_{1D} H_{2D} + G_N H_{1N} H_{2D} + K G_N H_{2N} H_{1D} \right) =$$

$$= \lim_{K \to \infty} \frac{1}{K} \left(G_D H_{1D} H_{2D} + G_N H_{1N} H_{2D} \right) + G_N H_{2N} H_{1D}$$

$$= 0 + G_N H_{2N} H_{1D}$$

$$= G_N H_{2N} H_{1D}$$

which is the polynomial of the open loop zeros. The above expression is equal to the limit as K approaches infinity of $\frac{1}{K}$ times the characteristic polynomial. Working with the characteristic matrix, this is equivalent to

$$P(S) = \lim_{K \to \infty} \frac{1}{K} \begin{vmatrix} 1 & 0 & 1 & 1 \\ -G_N & G_D & 0 & 0 \\ 0 & H_{1N} & -H_{1D} & 0 \\ 0 & KH_{2N} & 0 & -H_{2D} \end{vmatrix}$$

3.8 Continuous System Open Loop Transfer Function (Continued)

$$= \lim_{K\to\infty} \begin{vmatrix} 1 & 0 & -1 & 1 \\ -G_N & \frac{1}{K}G_D & 0 & 0 \\ 0 & \frac{1}{K}H_{1N} & -H_{1D} & 0 \\ 0 & H_{2N} & 0 & -H_{2D} \end{vmatrix}$$

$$= \begin{vmatrix} 1 & 0 & 1 & 1 \\ -G_N & 0 & 0 & 0 \\ 0 & 0 & -H_{1D} & 0 \\ 0 & H_{2N} & 0 & -H_{2D} \end{vmatrix}$$

$$= G_N H_{2N} H_{1D}$$

$$= K_N \prod_{i=1}^{p} (S-a_i)$$

which is the polynomial of the open loop zeros.

Hence,

$$\frac{P(S)}{Q(S)} = \frac{K_N \prod_{i=1}^{p} (S-a_i)}{K_D \prod_{i=1}^{q} (S-b_i)} = \frac{G_N H_{2N} H_{1D}}{G_D H_{1D} H_{2D} + G_N H_{1N} H_2} = \frac{X_2}{X_2}$$

is the desired open loop transfer function.

3.8 Continuous System Open Loop Transfer Function (Continued)

Thus, the matrices in question are the following:

Computation	Matrix Polynomial
Open loop poles	System characteristic matrix but with the element at which the system is opened being zeroed.
Open loop zeros	System characteristic matrix but with every element in the column at which the system is opened being zeroed except for the element at which the system is opened.

The computation in each case reduces to finding the characteristic roots and the leading coefficients of the associated matrix polynomials.

3.9 Computation of Characteristic Roots

As indicated in sections 2.2.5 and 2.2.7, the program must compute the continuous system open loop zeros and open loop poles and the sampled data system characteristic roots. The computational steps involved in these calculations are described as follows.

Let $A(\lambda)$ be a complex square matrix polynomial of degree n whose characteristic roots are to be computed. $A(\lambda)$, a matrix whose elements are polynomials with complex coefficients in λ, is represented mathematically by

$$A(\lambda) = A_0\lambda^n + A_1\lambda^{n-1} + \ldots + A_{n-1}\lambda + A_n$$

where the A_i are constant complex matrices and A_0 is not the zero matrix. The characteristic polynomial of $A(\lambda)$ is the determinant of $A(\lambda)$, denoted $|A(\lambda)|$. The characteristic roots (eigenvalues) of $A(\lambda)$ are defined as the complex values of λ for which the characteristic polynomial vanishes.

The method used by the program to compute the characteristic roots is a modification of an iterative technique originally employed for polynomials not having matrices as coefficients. Based on an estimate of a characteristic root, three starting iterants are formulated. The corresponding functional value for each iterant is computed where a functional value is defined as the evaluation of the characteristic polynomial at the specified iterant value. A quadratic fit by Lagrange's method is made of the three iterants and functional values. The root of the quadratic equation closest in modulus to the last iterant is selected as the next iterant. If the iterants have converged, it is hypothesized that they have converged to a characteristic root which is saved by the program. The rooting routine restarts with a new estimate. Otherwise the functional value for the new iterant is evaluated and the rooting procedure repeats the previous step with the last three iterant and functional values. Theoretically, the iterants generated by the above procedure will eventually converge to a characteristic root. Roots are factored from the characteristic polynomial to avoid repeated convergence to the previously found roots.

3.9.1 Scaling

Certain arithmetic operations involving functional values are subject to exponent overflow and require special data representation. Each functional value has a base value, b, and a scale factor, s, which are

3.9.1 Scaling (Continued)

used to construct the true value.

ie. $x \equiv (b,s)$ where $x = b * 10^s$

As an example, consider the value $x = 4.2 * 10^{84}$ which is too large for usual data representation.

$$x \equiv (b,s) \equiv (4.2 * 10^{54}, 30)$$

where $$x = (4.2 * 10^{54}) * 10^{30} = 4.2 * 10^{84}$$

The rules for scaling a functional value are that the base value cannot exceed 10^{60} and the scale factor is a multiple of 15. When the base value exceeds 10^{60} it is divided by 10^{15} and 15 is added to the scale factor s.

3.9.2 Initial Three Iterants

The iterative technique used to compute the characteristic roots of a matrix polynomial requires the formation of three initial iterants based on an estimate. Let μ represent the estimate and β the constant 0.1 (special cases require that β be scaled down in multiples of 100). The first three iterants are formed as follows:

If $|\mu| < 1.0 * 10^{-6}$ then $\lambda_0 = -\beta(1+\beta)$

$\lambda_1 = -\beta(1-\beta)$

$\lambda_2 = -\beta$

If $|\mu| \geq 1.0 * 10^{-6}$ then $\lambda_0 = \mu(1+\beta)$

$\lambda_1 = \mu(1-\beta)$

$\lambda_2 = \mu$

λ_0, λ_1, and λ_2 represent the first three iterants, respectively.

3.9.3 Functional Values

The functional value, $F_p(\lambda)$, corresponding to the iterant, λ, is defined as the evaluation at the iterant of the characteristic polynomial after previously found roots have been factored out.

$$F_0(\lambda) = |A(\lambda)|$$

$$F_p(\lambda) = \frac{|A(\lambda)|}{\prod_{i=1}^{p}(\lambda-\lambda_i)} \qquad P = 1, 2, \ldots, n \text{ where } n \text{ is the number of characteristic roots and the } \lambda_i \text{ are the roots found by the program.}$$

Because the characteristic polynomial can not be constructed directly from the matrix polynomial, an actual factoring out of previously found roots can not be done. An artificial technique of evaluating the numerator and denominator separately and then dividing is used by the program. This method works well for simple roots (non-repeated roots) but fails when converging to the multiple roots. To resolve this situation, iterants converging to previously found roots are modified in such a way as to cause multiple roots to appear as clustered roots. (Refer to section 2.2.1.5 for further discussion on multiple roots)

3.9.4 Calculate New Iterant

Mathematically, the program obtains a quadratic fit of the last three iterants (λ_i, λ_{i+1}, λ_{i+2}) and functional values ($f_p(\lambda_i)$, $f_p(\lambda_{i+1})$, $f_p(\lambda_{i+2})$), roots the quadratic, and accepts as the next iterant (λ_{i+3}) the root closest to λ_{i+2}. The mathematics have been reduced to the following computations.

$$D_2 = (\lambda_{i+2}-\lambda_{i+1})/(\lambda_{i+1}-\lambda_i)$$

$$B_2 = D_2^2 * f_p(\lambda_i) - (1+D_2)^2 * f_p(\lambda_{i+1}) + (1+2*D_2) * f_p(\lambda_{i+2})$$

3.9.4 Calculate New Iterant (Continued)

$$B_3 = D_2 * f_p(\lambda_i) - (1+D_2) * f_p(\lambda_{i+1}) + f_p(\lambda_{i+2})$$

$$DENOM = B_2 \pm \sqrt{B_2^2 - 4 * D_2 * (1+D_2) * f_p(\lambda_{i+2}) * B_3}$$

In the above expression + or - is selected according to which sign yields the largest modulus of parameter DENOM

$$D_3 = \frac{-2 * (1+D_2) * f_p(\lambda_{i+2})}{DENOM}$$

If DENOM = 0.0 then D3 = 1.0 (only imposed to avoid division by zero).

$$\lambda_{i+3} = \lambda_{i+2} + D_3 * (\lambda_{i+2} - \lambda_{i+1})$$

3.9.5 Multiple Roots

Multiple roots can not be calculated exactly because of a breakdown in the method of determining functional values (section 2.2.9.3). If an iterant is permitted to be equal to a previously found root, then the functional value evaluation would involve division by zero, thereby, causing unpredictable results. Corrective measures by the program include recognition of troublesome iterants and their modification.

if $|\lambda| < 1.0 * 10^{-6}$ and $|\lambda - \lambda_i| < 1.0 * 10^{-6}$

where λ_i is a previously found root,

then subtract 0.01 from the real part of λ.

3.9.5 Multiple Roots (Continued)

if $|\lambda| \geq 1.0 \times 10^{-6}$ and $\left|\frac{\lambda-\lambda_i}{\lambda}\right| < 1.0 \times 10^{-6}$

where λ_i is a previously found root, then subtract 0.01 from the real part of λ.

The modified value of λ is similarly checked against the previously found roots. Since the rooting technique depends on the convergence of iterants and the iterants are modified if they too closely approach a previously found root, then multiple roots must necessarily appear as clustered roots.

3.9.6 Convergence Tests

Three independent tests are performed to check for convergence of the iterants to a characteristic root. The passing of one test is sufficient for acceptance of the last iterant as a root. Suppose λ_i is the last iterant and $f_p(\lambda_i)$ is the corresponding functional value.

Resultant Convergence Code in Printout	Convergence Test passed
1	$\lvert f_p(\lambda_i)\rvert = 0.0$
2	$\lvert\lambda_i\rvert \leq 1.0 \times 10^{-4}$
3	$\left\lvert\frac{\lambda_i-\lambda_{i-1}}{\lambda_i}\right\rvert < 1.0 \times 10^{-4}$
4	Complex conjugate of a previously calculated root

In reference to convergence code four, a program option exists which instructs the program to automatically accept the conjugate of a strictly complex root as a characteristic root. Assume

3.9.6 Convergence Tests (Continued)

that $\lambda = x+iy$ has been accepted as a root and the conjugate option has been selected. The complex conjugate $\bar{\lambda}=x-iy$ is itself accepted as a characteristic root if

$$|y| > .1 \quad \text{and} \quad |y| \geq 1.0*10^{-3}*|x|.$$

3.9.7 Accuracy of the Iterant Convergence Criteria

Normally, the rooting procedure causes the iterants to converge to an accurate characteristic root where the corresponding functional values do not demonstrate a wild variance of magnitude between succeeding iterations. If the estimate of a characteristic root causes immediate convergence (one iteration) to a root, the iteration procedure restarts with the same estimate but with a scaled down value of β (ie. $\beta=.01*\beta$)(see section 2.2.5.2). This is done to insure more accurate convergence to the actual characteristic root. The value to which the restart converges is accepted as being the actual root. It is also possible that the iterants will converge but the corresponding functional values will decline in magnitude quite rapidly. This combination of events indicates that the last iterant is close to a characteristic root with respect to the other iterants although it may not be the actual root. When such a condition is recognized, the iteration procedure restarts by taking the last iterant as the new estimate and scales down the value of β(ie. $\beta = .01*\beta$). The value to which the restart converges is accepted as being the actual root. To recognize the rapid decline of the functional values, the following test is applied to each successive pair of functional values denoted $f_p(\lambda_{i-1})$ and $f_p(\lambda_i)$, respectively:

If $|f_p(\lambda_{i-1})|>|f_p(\lambda_i)|*10^5$, then restart the iterative procedure with λ_i as the estimate.

By using these special techniques, more confidence can be expressed in accepting the convergence of the iterants as valid criteria for being a characteristic root.

3.9.8 Divergence of the Rooting Procedure

Increases in magnitude of functional values corresponding to successive iterants generally implies divergence of the rooting procedure. To insure non-divergence, successive functional values are not permitted to increase in magnitude by more than a factor of 10. If

$$\left| \frac{f_p(\lambda_i)}{f_p(\lambda_{i-1})} \right| > 10.0$$

where $f_p(\lambda_{i-1})$ and $f_p(\lambda_i)$ denote successive functional values, then proceed with one of the following two solutions:

(1) If this is the first successive functional value increase, then permit the iterative procedure to continue.

(2) If this is the second or later successive functional increase then let $\lambda_i = (1/2)\lambda_i$ and recompute $f_p(\lambda_i)$. Test the new value of $f_p(\lambda_i)$ as before. Repeat this procedure at most 10 times or until an acceptable value of $f_p(\lambda_i)$ has been found. If after 10 attempts the functional value $f_p(\lambda_i)$ is still unacceptable, then restart the rooting technique with a new estimate.

End of Rooting Test

The program continues to find the characteristic roots of the matrix polynomial until either the number of roots as specified by the user or all the roots possessed by the matrix polynomial have been computed. Two rooting completion tests must be satisfied for the rooting procedure to terminate on the latter condition. After each characteristic root has been computed, the program assumes there is at least one more characteristic root to find and determines the starting three iterants (R_0, R_1, and R_2). The conditions required for the termination of the rooting technique are:

$$\left| \frac{f_p(R_0) - f_p(R_1)}{f_p(R_0)} \right| \leq 1.0*10^{-4}$$

3.9.9 End of Rooting Test (Continued)

$$\left| \frac{f_p(-100) - f_p(R_0)}{f_p(R_2)} \right| \leq 1.0 * 10^{-4}$$

The reliability of the tests is that when all of the characteristic roots have been computed the polynomial formed by the characteristic roots differs from the characteristic polynomial by a constant.

ie. $|\Lambda(\lambda)| = c \prod_{i=1}^{n} (\lambda - \lambda_i)$ where the matrix polynomial possesses n characteristic roots.

$$f_n(\lambda) = \frac{|\Lambda(\lambda)|}{\prod_{i=1}^{n} (\lambda - \lambda_i)} = \frac{c \prod_{i=1}^{n} (\lambda - \lambda_i)}{\prod_{i=1}^{n} (\lambda - \lambda_i)} = c$$

Therefore, each iterant value will have a corresponding functional value of c and the above two tests will be passed. If all of the characteristic roots have not been found, then at least one and usually both of the above tests will fail and cause the rooting routine to continue.

3.9.10 Restarts

The iterative procedure described in this document can occasionally diverge from the characteristic roots, converge too rapidly and lose accuracy, or not converge within the forty iteration limit. Logic is incorporated into the program to recognize the above adverse conditions and restart the iterative technique with a new estimate. If ten successive restarts are attempted without convergence, an error message is printed and the rooting procedure terminates for the current matrix polynomial.

3.9.11 Estimates

With few exceptions due to slow convergence and iteration limits, the success of the rooting technique does not depend on accurate estimates of the characteristic roots. The input of accurate estimates can, however, minimize the number of iterations required for convergence to the characteristic roots and reduce computer run time. The program has five sources of estimates.

- A user supplied estimate
- B modification of previously computed root of the current matrix polynomial
- C complex conjugate of previously computed complex root of the current matrix polynomial
- D modification of a selected iterant value
- E one of four program default estimates 0, -50i, -50, -75 -75i

The user supplied estimates can be input to the program either directly or indirectly by the user. Data card specification of the estimates comprises the direct user control of estimates. Through the manipulation of program control cards, the user can cause the program to utilize characteristic roots generated in the previous case as estimates for the characteristic roots of the current case. In this way the user has an indirect control of estimates. Program logic determines which estimate source is most suitable for supplying an estimate to the next root sought by the program. The logic is patterned after the following table:

Table 3-I PROGRAM CONDITIONS

Normal Restart	User Estimate Available	Complex Conjugate of Last Root Available	Last Root Available	Iterants Converging Too Fast	Rooting Technique Diverging	ESTIMATE SOURCE
NO		YES		NO	NO	C
NO	YES			NO	NO	A
NO	NO	NO	YES	NO	NO	B
NO	NO	NO	NO	NO	NO	E
YES	YES			NO	NO	A
YES	NO			NO	NO	E
				YES		D
	YES				YES	A
	NO				YES	E

NOTE: A blank slot in the table implies the contents has no effect on the program selection of the estimate.

3.9.12 Default Estimates

When all other estimate sources are exhausted, one of the program default estimates is selected as the estimate for the next characteristic root. There are four default estimates.

(1) 0+0i
(2) 0-50i
(3) -50+0i
(4) -75-75i

Normally, the first default estimate used by the program is 0+0i. If convergence to a characteristic root does not occur within the required iteration limit then the next default estimate is used. This procedure is continued until either a root is found or a complete cycle of four default estimates are selected yielding no results. If a root is found using a default estimate and the iteration count for that estimate exceeds twenty five, then the next time default estimates are needed the cycle starts with the following default estimate. Otherwise, it uses the default estimate which was used to find the last characteristic root. For clarification, consider the following example:

Default Estimate	Iteration Count*
0+0i	50 (non convergence)
0-50i	50 (non convergence)
-50+0i	n (root found)

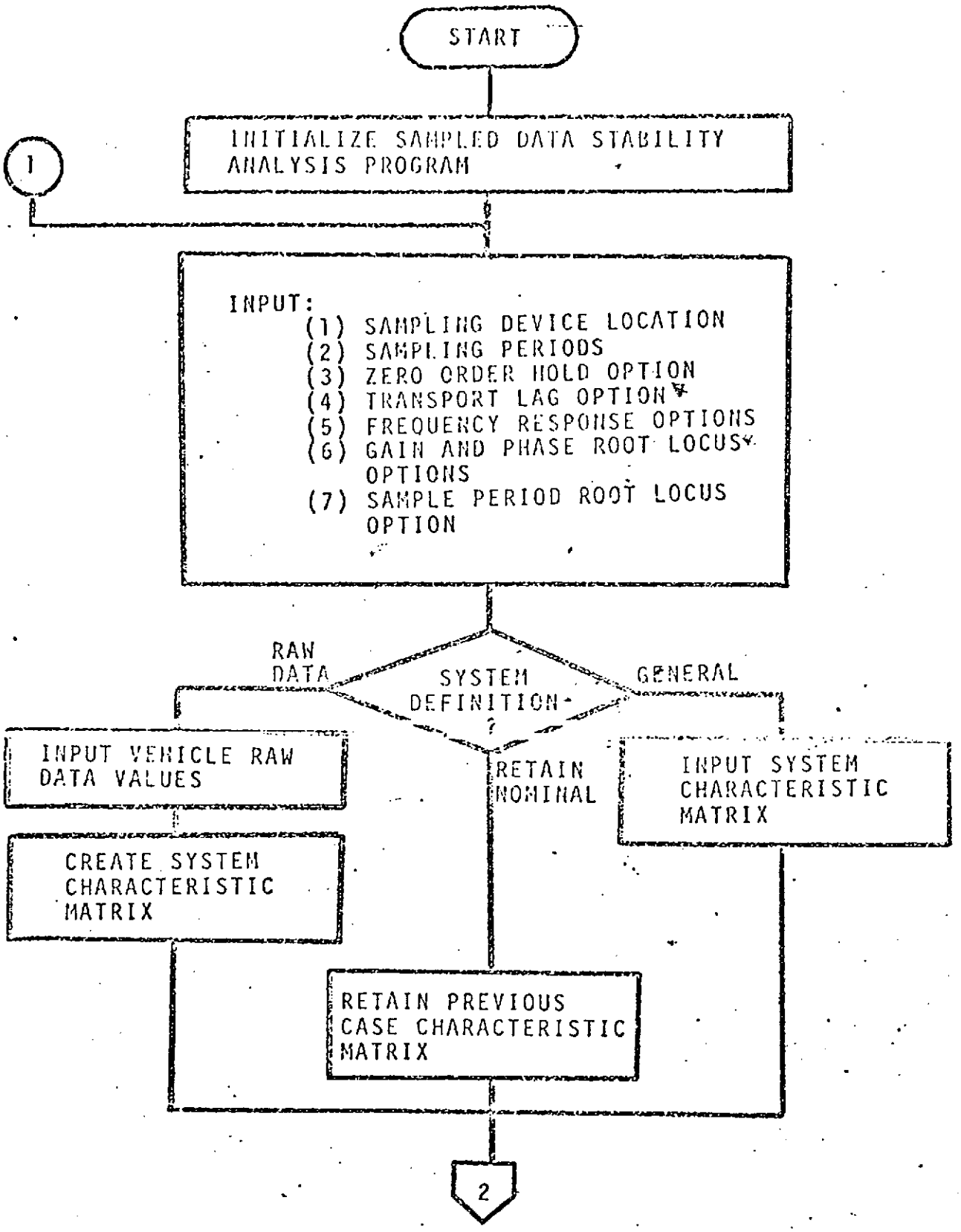
If n≤25, then the next default estimate cycle begins with -50+0i as the first selection.

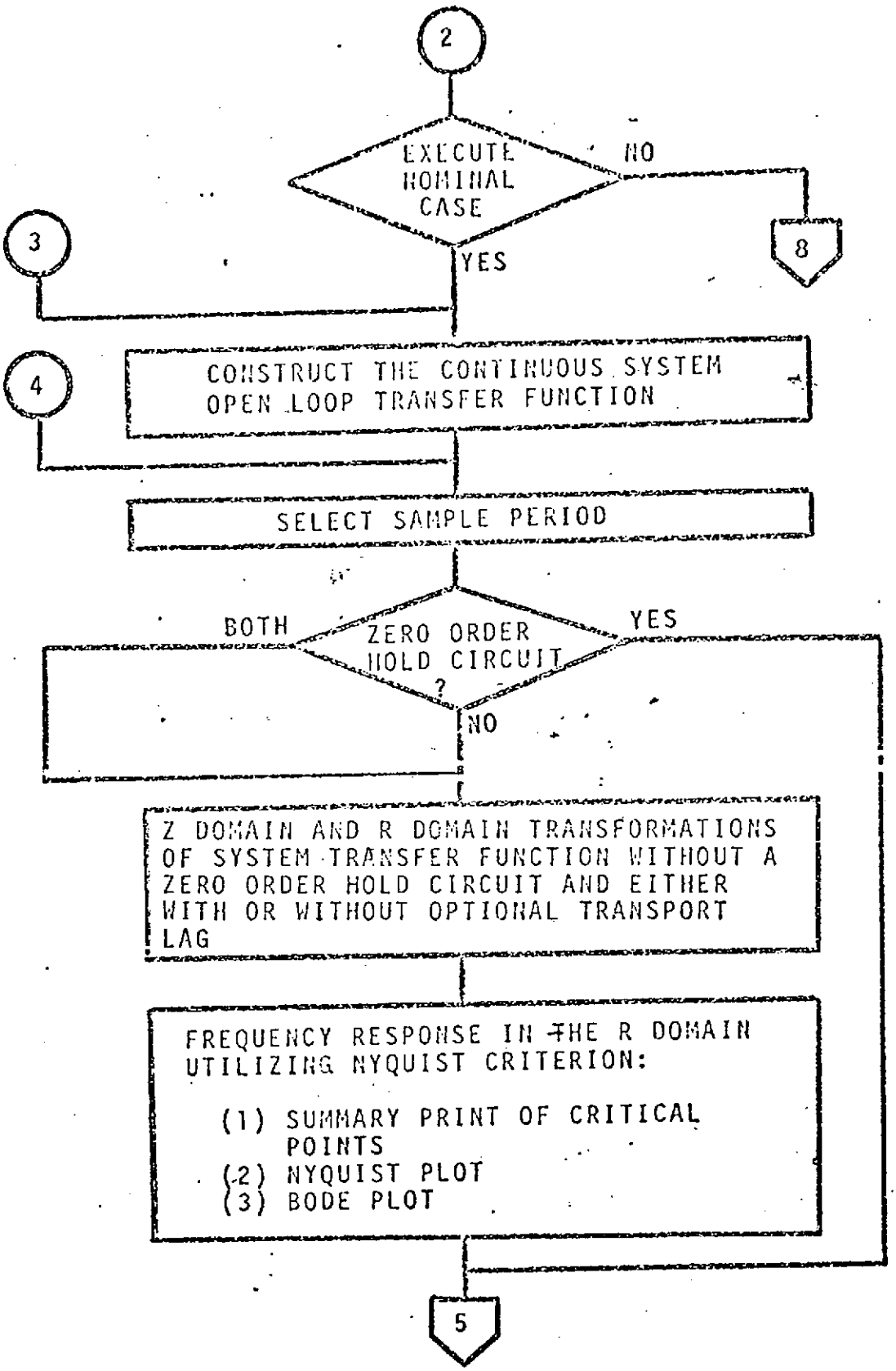
If n>25, then the next default estimate cycle begins with -75-75i as the first selection.

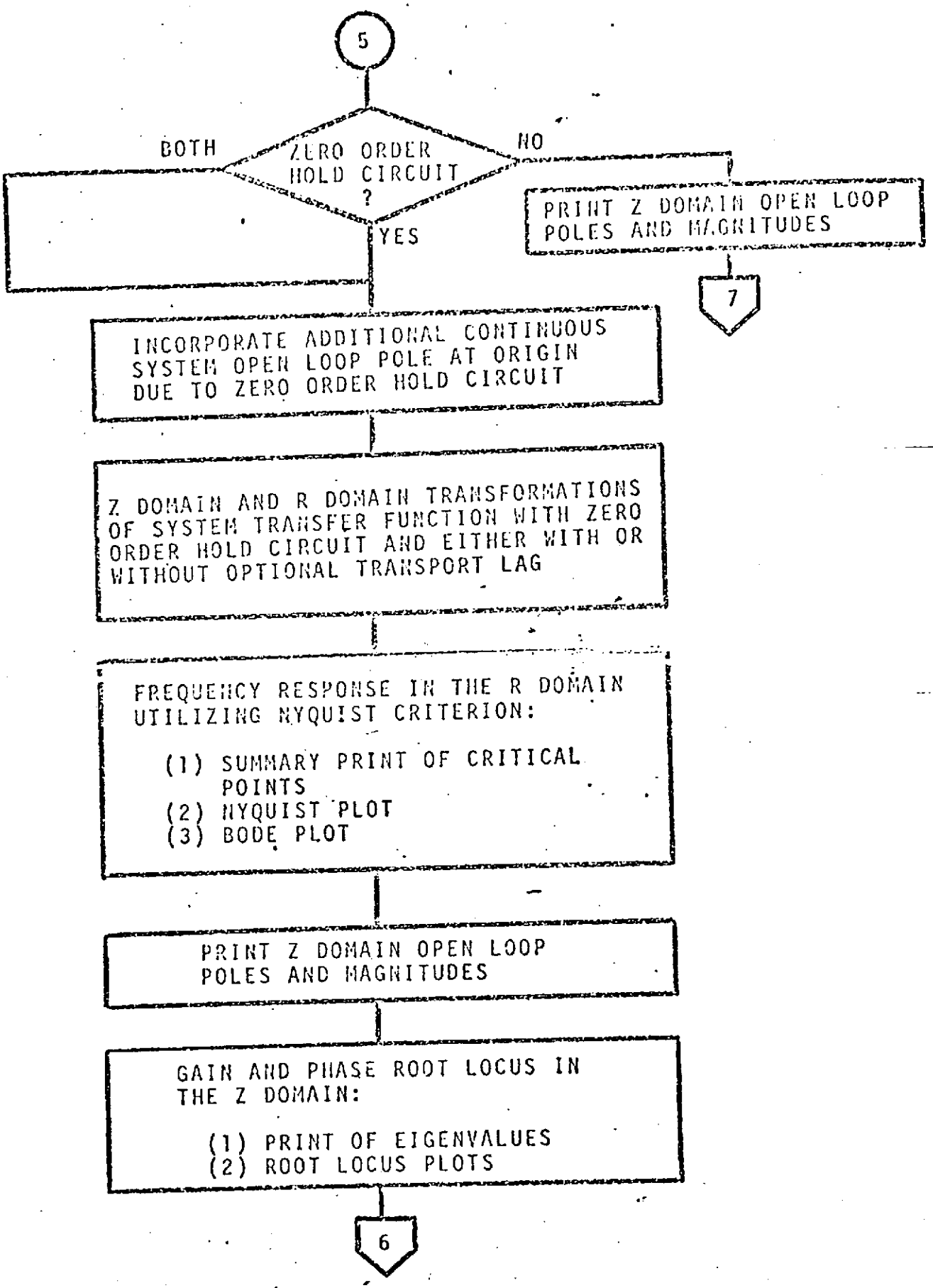
*(maximum of 50 iterations per estimate)

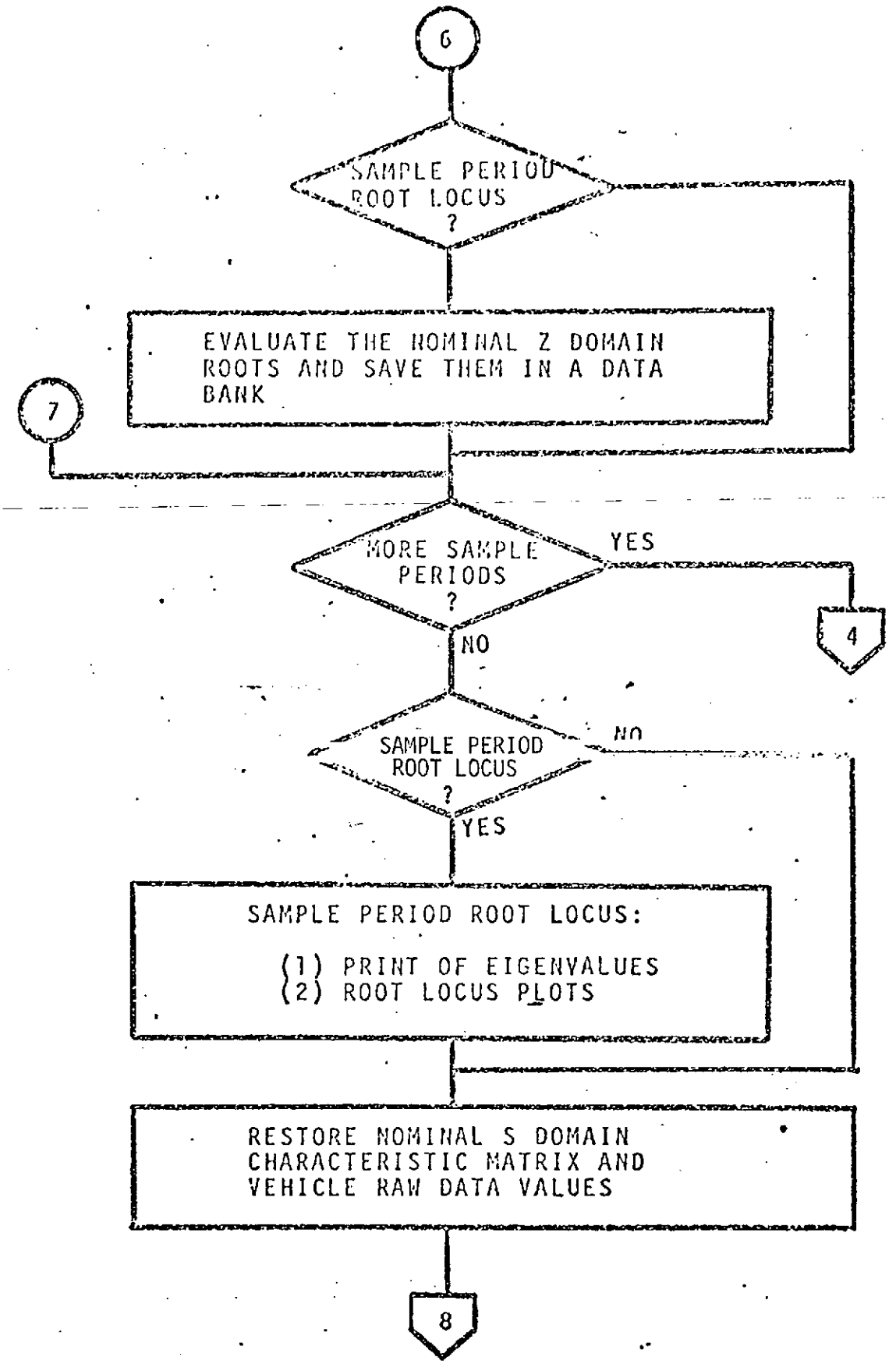
4.0 FLOWCHARTS

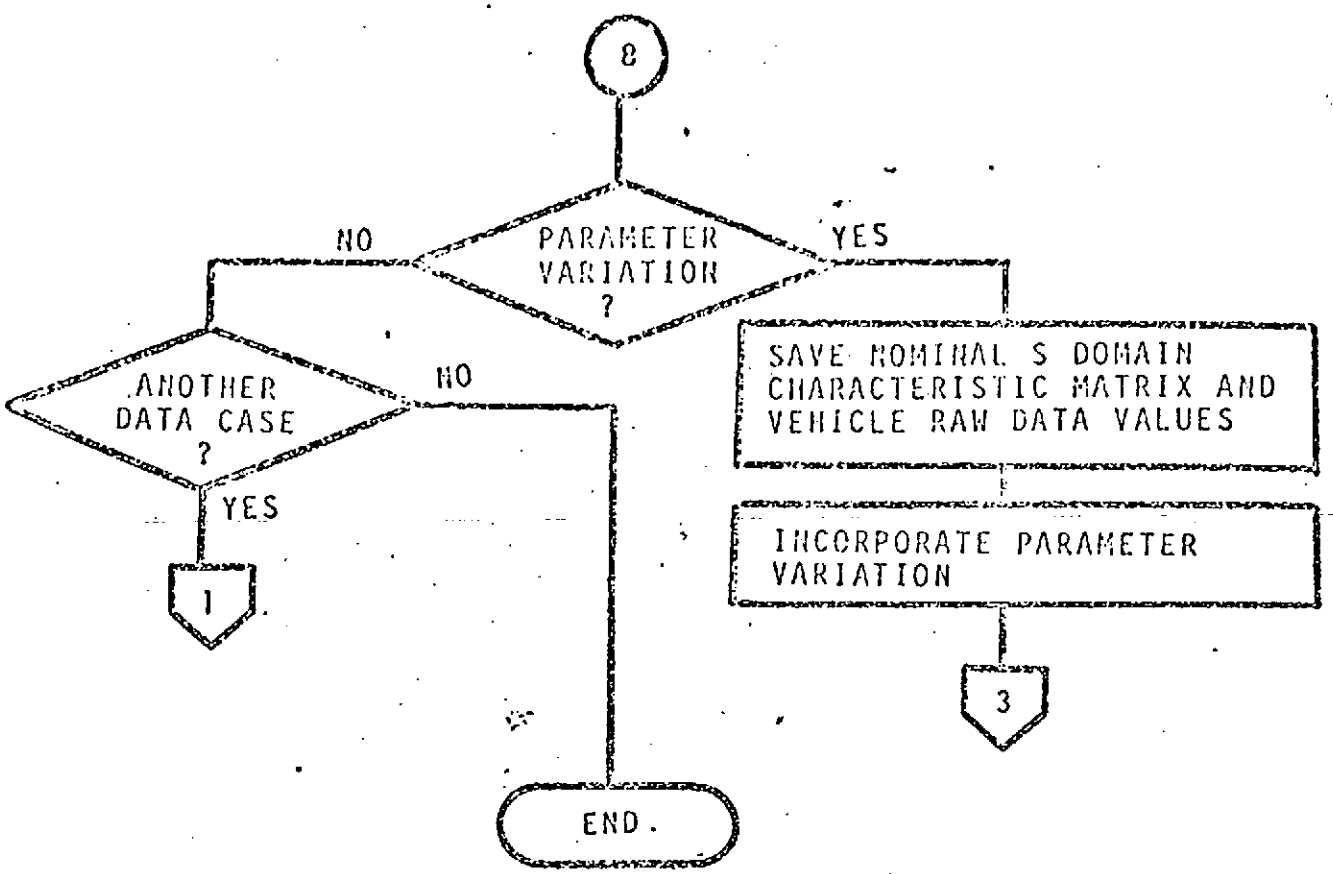
4.1 General Program Logic


START
1
INITIALIZE SAMPLED DATA STABILITY ANALYSIS PROGRAM
INPUT:
(1) SAMPLING DEVICE LOCATION
(2) SAMPLING PERIODS
(3) ZERO ORDER HOLD OPTION
(4) TRANSPORT LAG OPTION
(5) FREQUENCY RESPONSE OPTIONS
(6) GAIN AND PHASE ROOT LOCUS OPTIONS
(7) SAMPLE PERIOD ROOT LOCUS OPTION
SYSTEM DEFINITION
RAW DATA
GENERAL
RETAIN NOMINAL
INPUT VEHICLE RAW DATA VALUES
CREATE SYSTEM CHARACTERISTIC MATRIX
INPUT SYSTEM CHARACTERISTIC MATRIX
RETAIN PREVIOUS CASE CHARACTERISTIC MATRIX
2

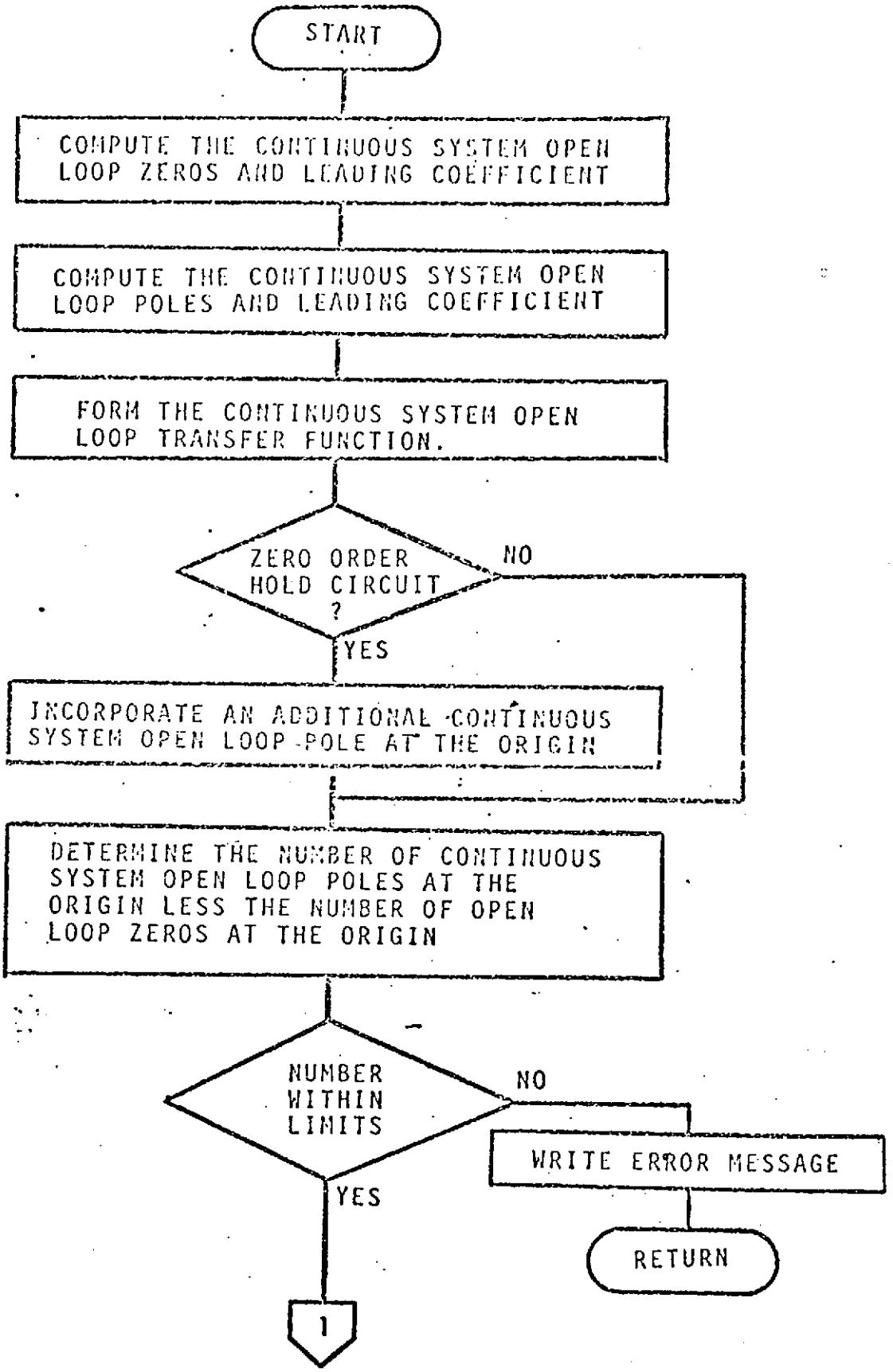

2
EXECUTE NOMINAL CASE
NO
YES
3
8
4
CONSTRUCT THE CONTINUOUS SYSTEM OPEN LOOP TRANSFER FUNCTION
SELECT SAMPLE PERIOD
BOTH
ZERO ORDER HOLD CIRCUIT ?
YES
NO
Z DOMAIN AND R DOMAIN TRANSFORMATIONS OF SYSTEM TRANSFER FUNCTION WITHOUT A ZERO ORDER HOLD CIRCUIT AND EITHER WITH OR WITHOUT OPTIONAL TRANSPORT LAG
FREQUENCY RESPONSE IN THE R DOMAIN UTILIZING NYQUIST CRITERION:
(1) SUMMARY PRINT OF CRITICAL POINTS
(2) NYQUIST PLOT
(3) BODE PLOT
5

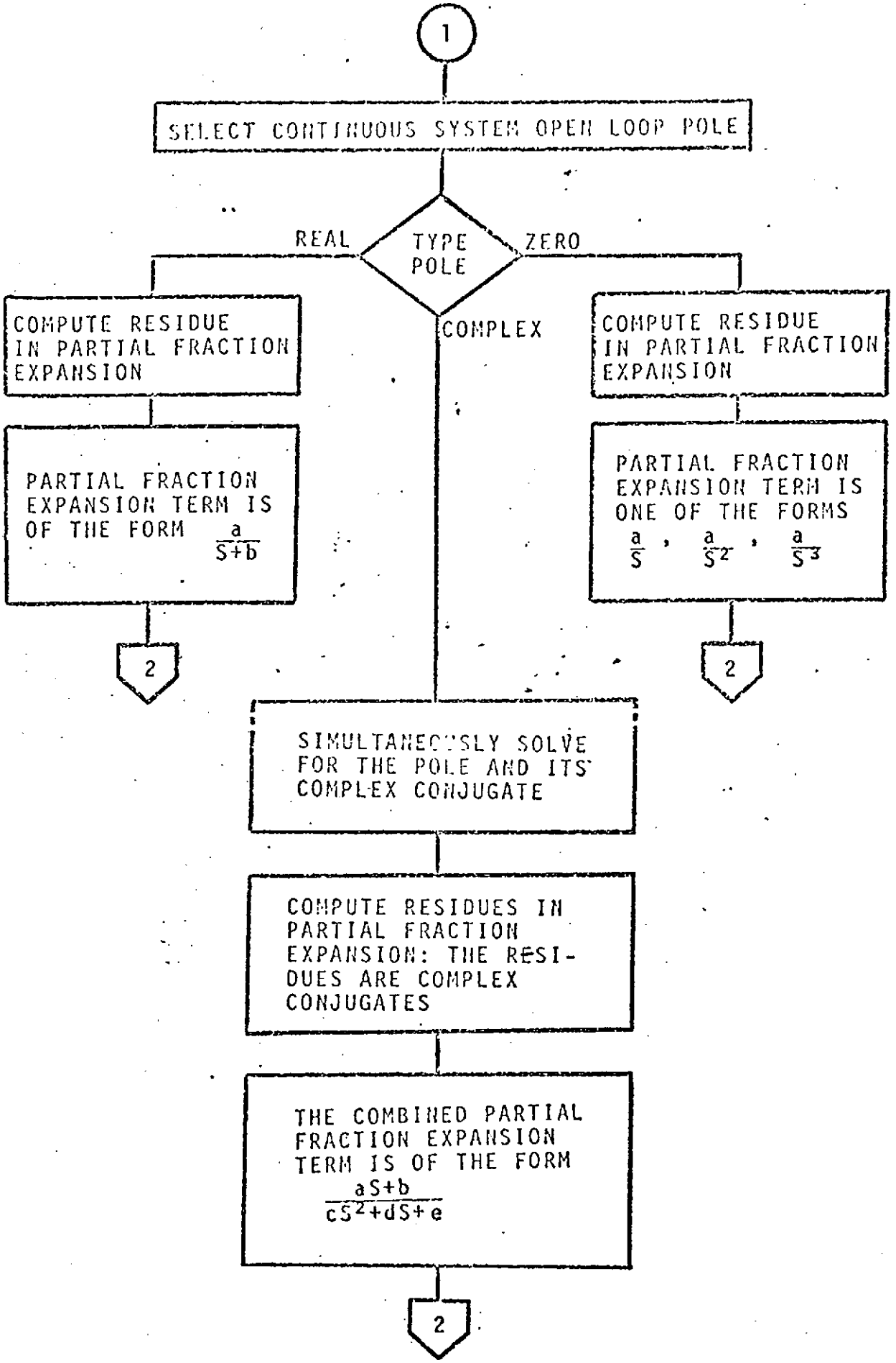

5
BOTH
ZERO ORDER HOLD CIRCUIT ?
NO
YES
PRINT Z DOMAIN OPEN LOOP POLES AND MAGNITUDES
7
INCORPORATE ADDITIONAL CONTINUOUS SYSTEM OPEN LOOP POLE AT ORIGIN DUE TO ZERO ORDER HOLD CIRCUIT
Z DOMAIN AND R DOMAIN TRANSFORMATIONS OF SYSTEM TRANSFER FUNCTION WITH ZERO ORDER HOLD CIRCUIT AND EITHER WITH OR WITHOUT OPTIONAL TRANSPORT LAG
FREQUENCY RESPONSE IN THE R DOMAIN UTILIZING NYQUIST CRITERION:
(1) SUMMARY PRINT OF CRITICAL POINTS
(2) NYQUIST PLOT
(3) BODE PLOT
PRINT Z DOMAIN OPEN LOOP POLES AND MAGNITUDES
GAIN AND PHASE ROOT LOCUS IN THE Z DOMAIN:
(1) PRINT OF EIGENVALUES
(2) ROOT LOCUS PLOTS
6

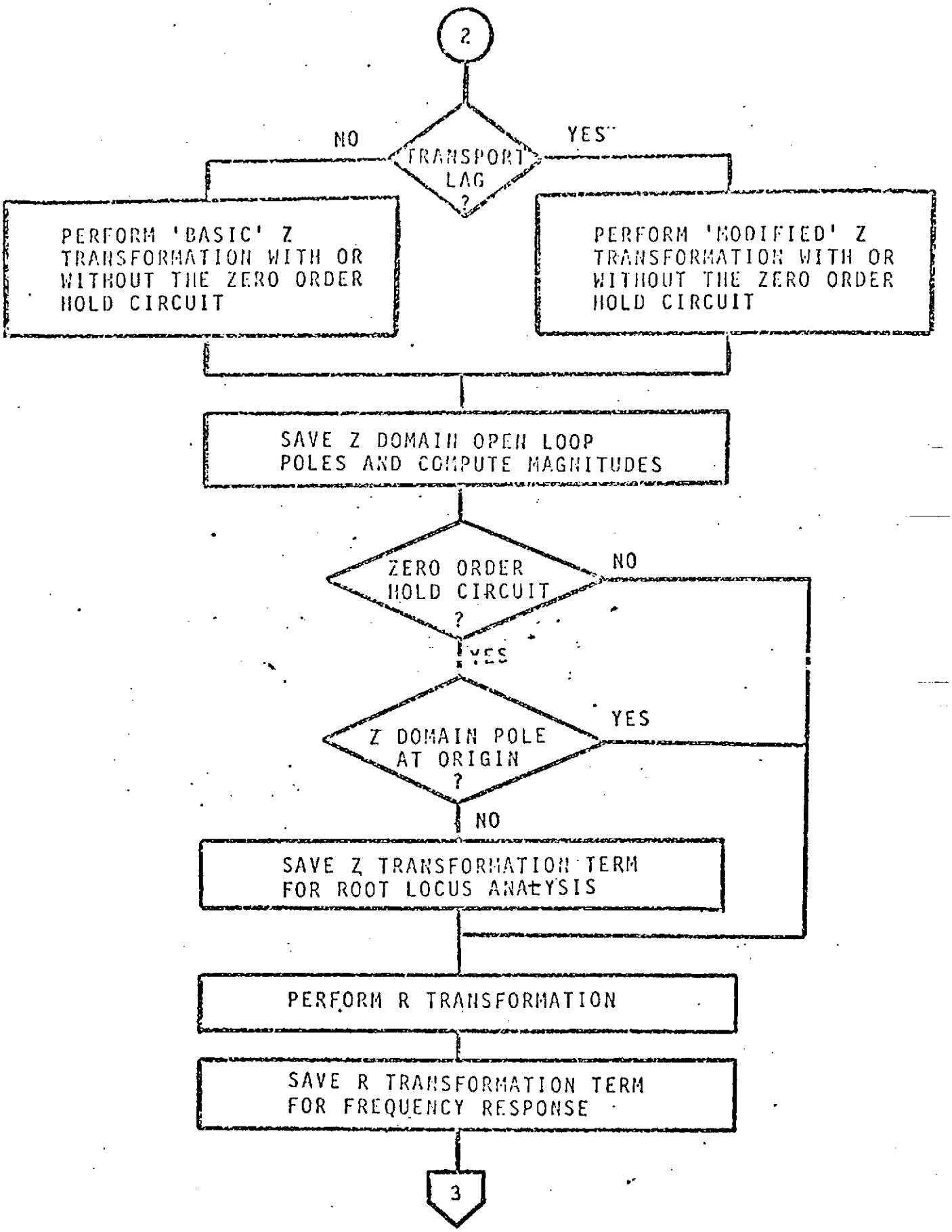

6
SAMPLE PERIOD ROOT LOCUS ?
EVALUATE THE NOMINAL Z DOMAIN ROOTS AND SAVE THEM IN A DATA BANK
7
MORE SAMPLE PERIODS ?
YES
4
NO
SAMPLE PERIOD ROOT LOCUS ?
NO
YES
SAMPLE PERIOD ROOT LOCUS:
(1) PRINT OF EIGENVALUES
(2) ROOT LOCUS PLOTS
RESTORE NOMINAL S DOMAIN CHARACTERISTIC MATRIX AND VEHICLE RAW DATA VALUES
8

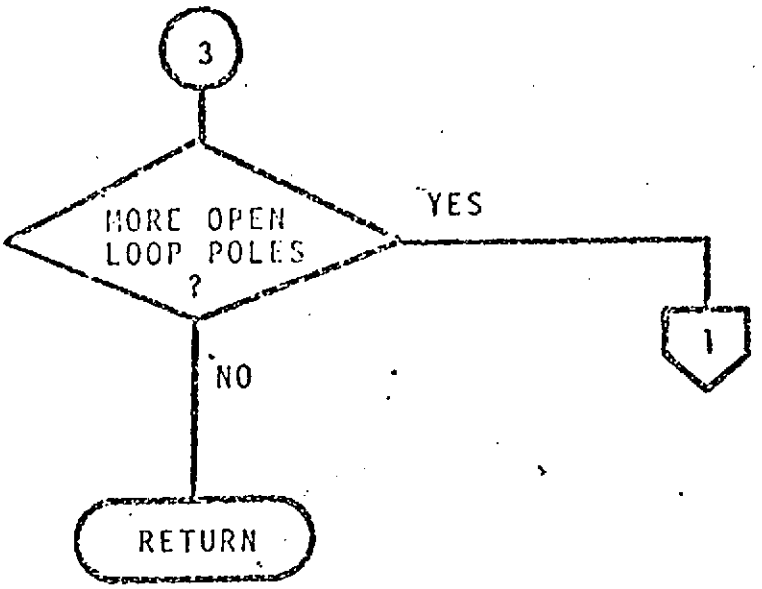

8
PARAMETER VARIATION ?
NO
YES
ANOTHER DATA CASE ?
NO
YES
1
END.
SAVE NOMINAL S DOMAIN CHARACTERISTIC MATRIX AND VEHICLE RAW DATA VALUES
INCORPORATE PARAMETER VARIATION
3

4.2 Sampled Data Open Loop Transfer Function

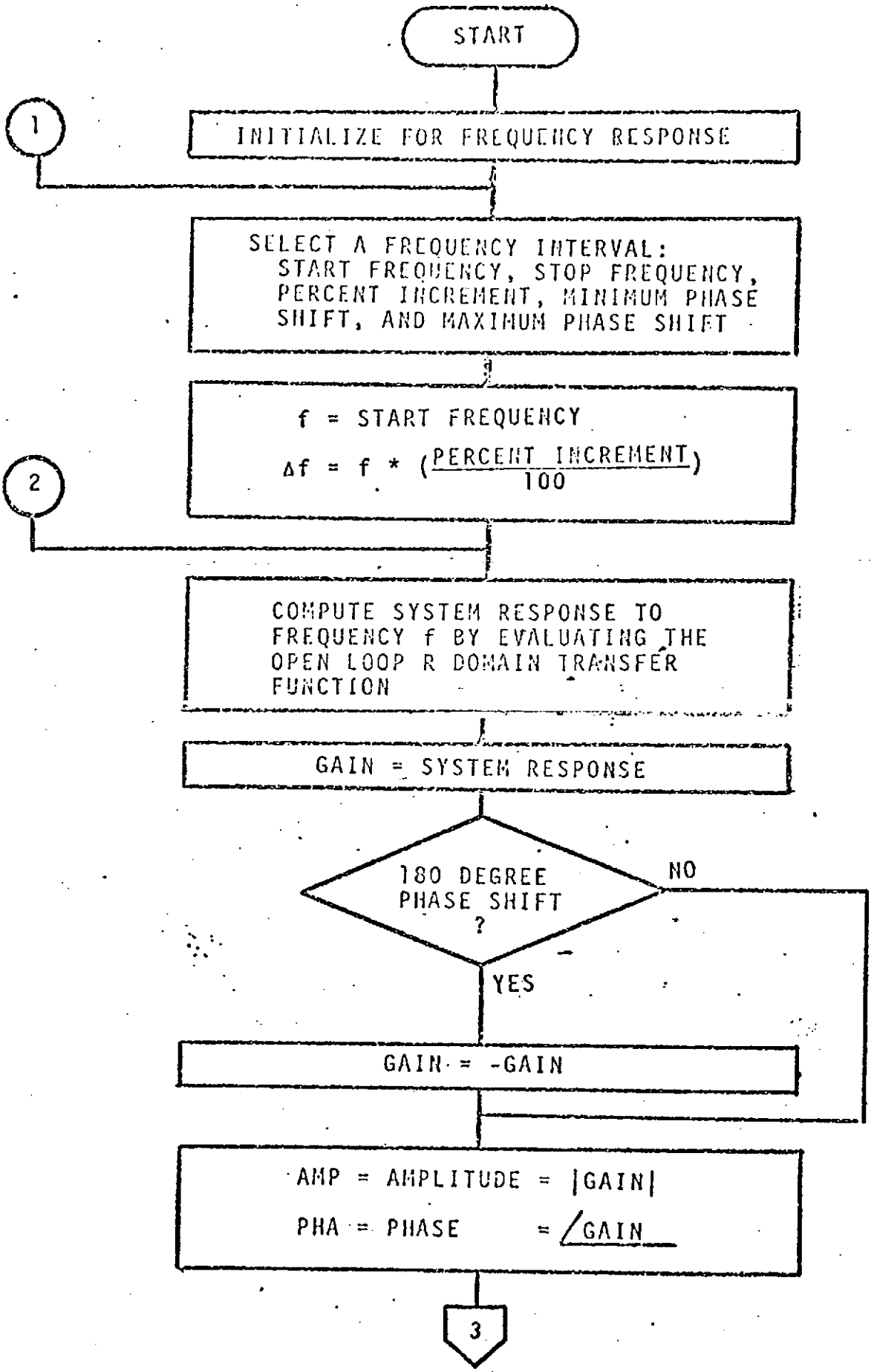

START
COMPUTE THE CONTINUOUS SYSTEM OPEN LOOP ZEROS AND LEADING COEFFICIENT
COMPUTE THE CONTINUOUS SYSTEM OPEN LOOP POLES AND LEADING COEFFICIENT
FORM THE CONTINUOUS SYSTEM OPEN LOOP TRANSFER FUNCTION.
ZERO ORDER HOLD CIRCUIT ?
NO
YES
INCORPORATE AN ADDITIONAL CONTINUOUS SYSTEM OPEN LOOP POLE AT THE ORIGIN
DETERMINE THE NUMBER OF CONTINUOUS SYSTEM OPEN LOOP POLES AT THE ORIGIN LESS THE NUMBER OF OPEN LOOP ZEROS AT THE ORIGIN
NUMBER WITHIN LIMITS
NO
YES
WRITE ERROR MESSAGE
RETURN
1

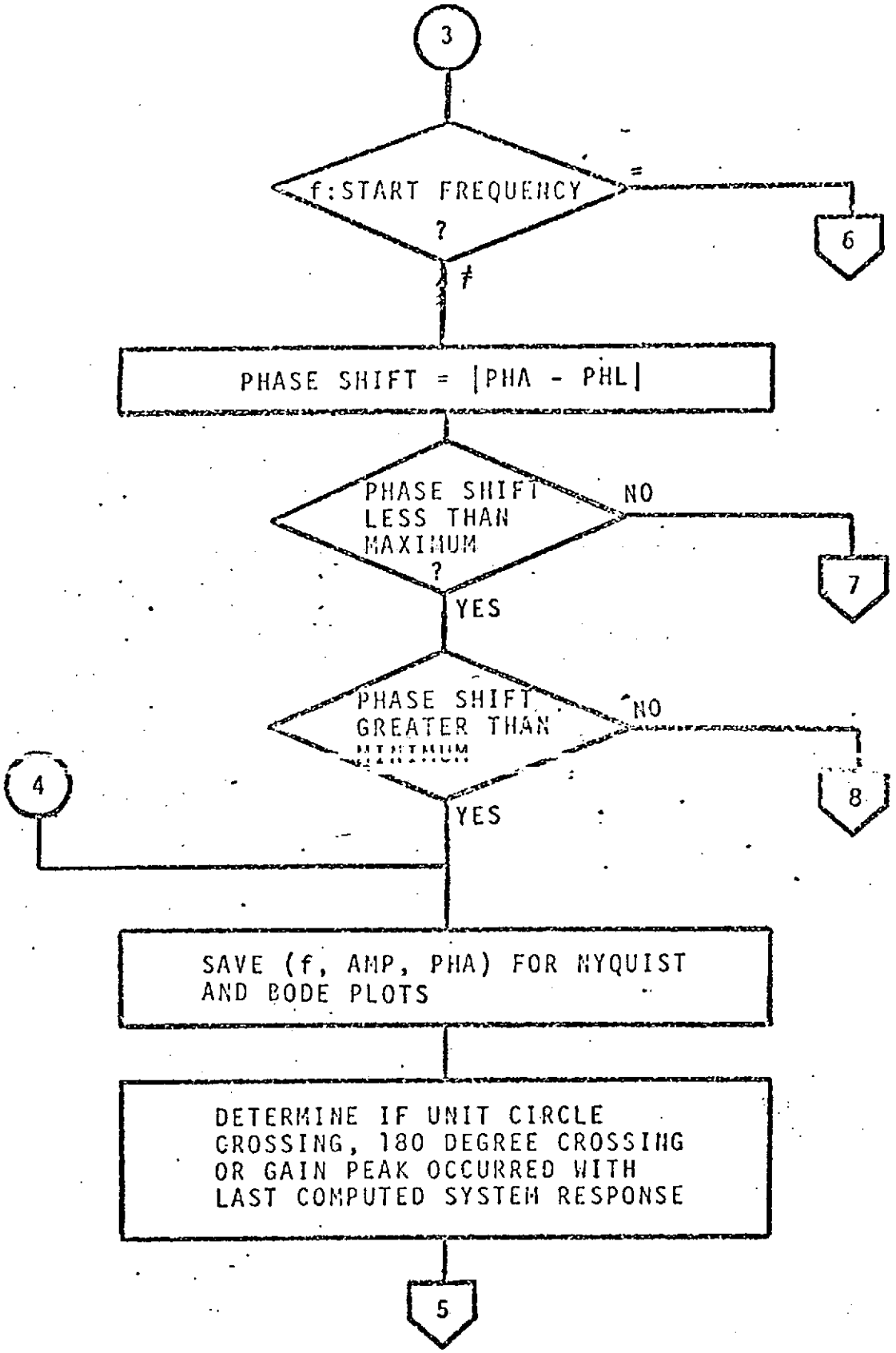

1
SELECT CONTINUOUS SYSTEM OPEN LOOP POLE
TYPE POLE
REAL
ZERO
COMPLEX
COMPUTE RESIDUE IN PARTIAL FRACTION EXPANSION
PARTIAL FRACTION EXPANSION TERM IS OF THE FORM a/(S+b)
COMPUTE RESIDUE IN PARTIAL FRACTION EXPANSION
PARTIAL FRACTION EXPANSION TERM IS ONE OF THE FORMS a/S , a/S² , a/S³
2
2
SIMULTANEOUSLY SOLVE FOR THE POLE AND ITS COMPLEX CONJUGATE
COMPUTE RESIDUES IN PARTIAL FRACTION EXPANSION: THE RESIDUES ARE COMPLEX CONJUGATES
THE COMBINED PARTIAL FRACTION EXPANSION TERM IS OF THE FORM (aS+b)/(cS²+dS+e)
2

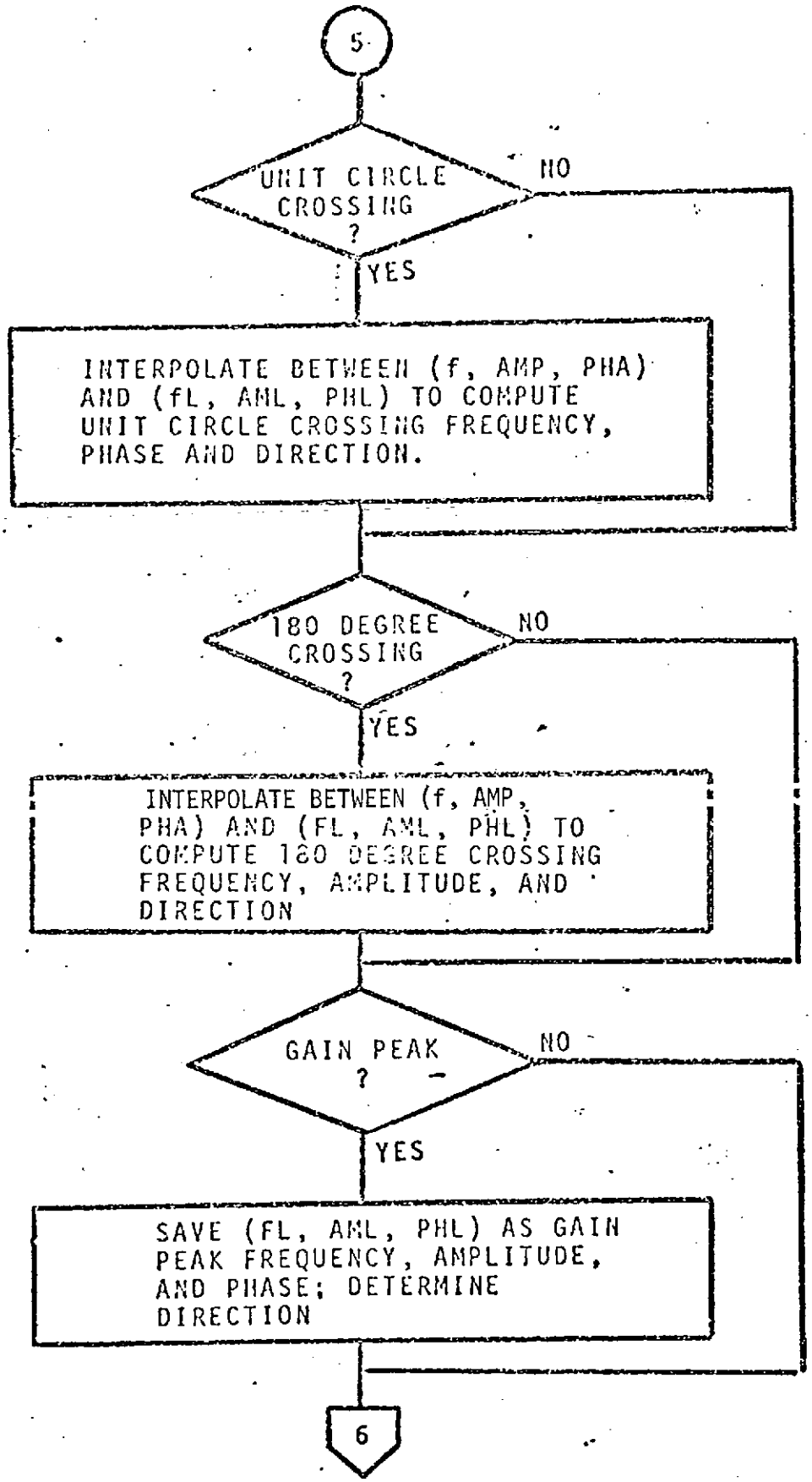

2
TRANSPORT LAG ?
NO
YES
PERFORM 'BASIC' Z TRANSFORMATION WITH OR WITHOUT THE ZERO ORDER HOLD CIRCUIT
PERFORM 'MODIFIED' Z TRANSFORMATION WITH OR WITHOUT THE ZERO ORDER HOLD CIRCUIT
SAVE Z DOMAIN OPEN LOOP POLES AND COMPUTE MAGNITUDES
ZERO ORDER HOLD CIRCUIT ?
NO
YES
Z DOMAIN POLE AT ORIGIN ?
YES
NO
SAVE Z TRANSFORMATION TERM FOR ROOT LOCUS ANALYSIS
PERFORM R TRANSFORMATION
SAVE R TRANSFORMATION TERM FOR FREQUENCY RESPONSE
3

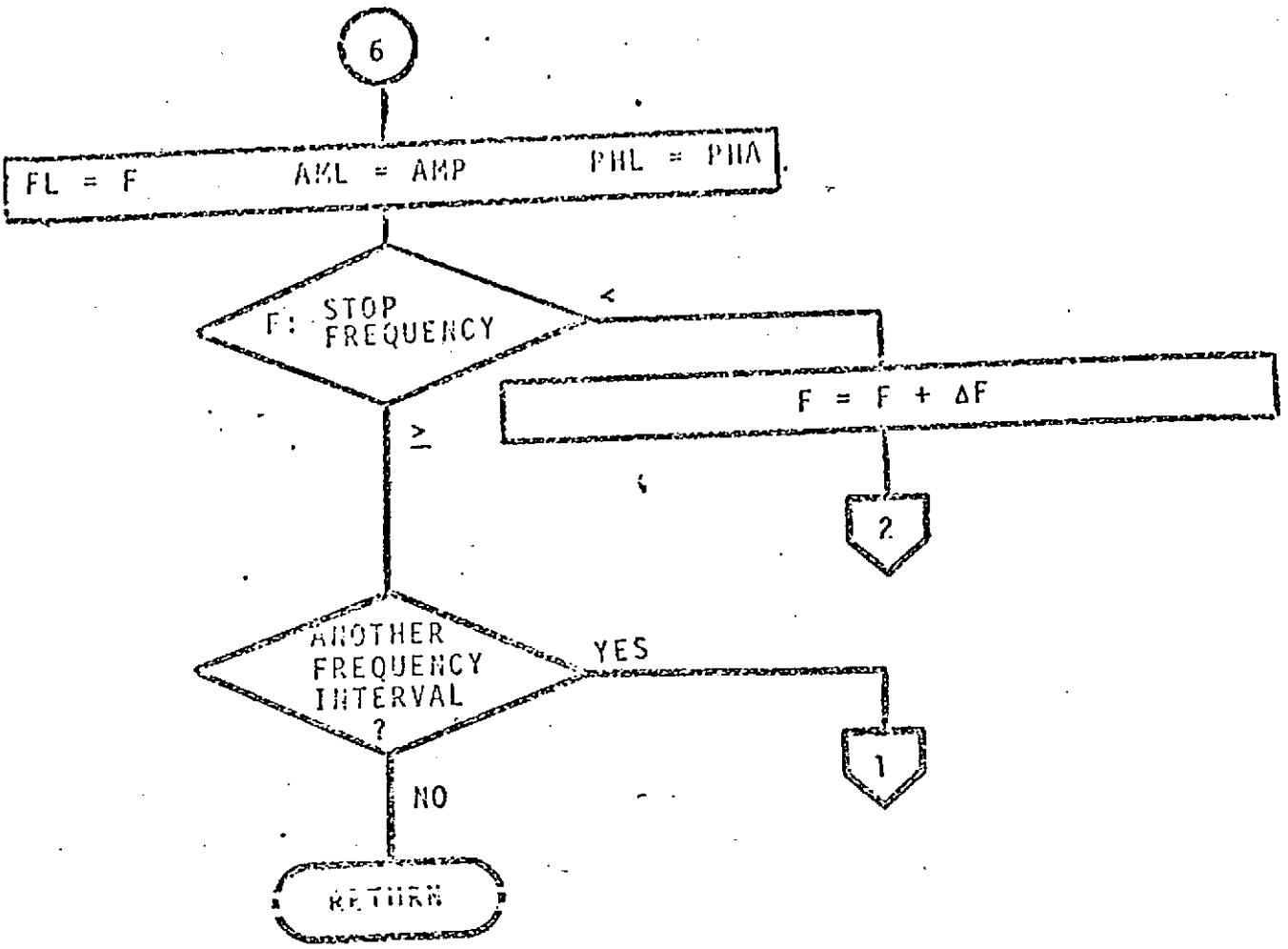

3
MORE OPEN LOOP POLES ?
YES
1
NO
RETURN

4.3 Frequency Response

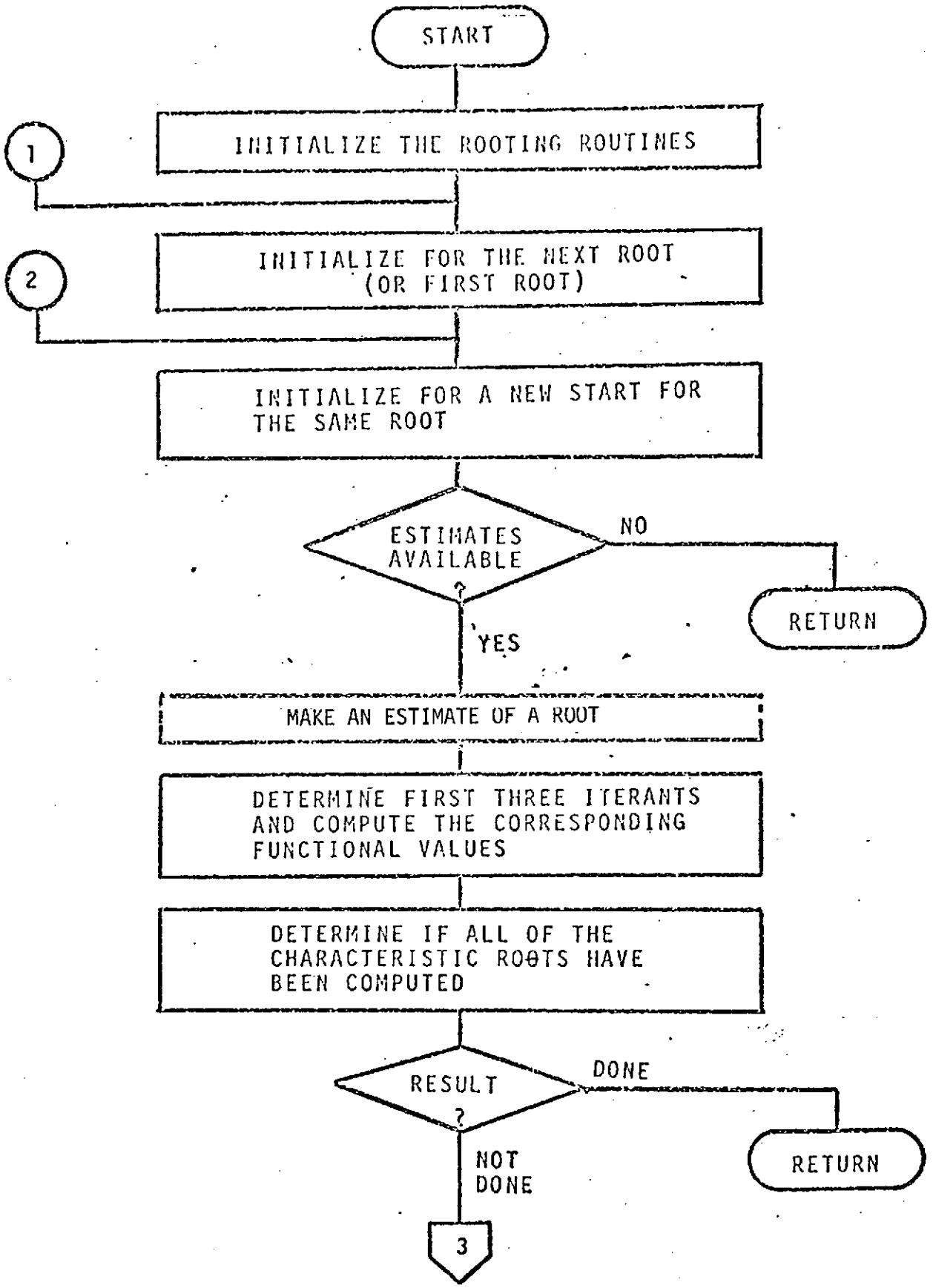

START
1
INITIALIZE FOR FREQUENCY RESPONSE
SELECT A FREQUENCY INTERVAL:
START FREQUENCY, STOP FREQUENCY,
PERCENT INCREMENT, MINIMUM PHASE
SHIFT, AND MAXIMUM PHASE SHIFT
f = START FREQUENCY
Δf = f * (PERCENT INCREMENT / 100)
2
COMPUTE SYSTEM RESPONSE TO
FREQUENCY f BY EVALUATING THE
OPEN LOOP R DOMAIN TRANSFER
FUNCTION
GAIN = SYSTEM RESPONSE
180 DEGREE
PHASE SHIFT
?
NO
YES
GAIN = -GAIN
AMP = AMPLITUDE = |GAIN|
PHA = PHASE = /GAIN
3

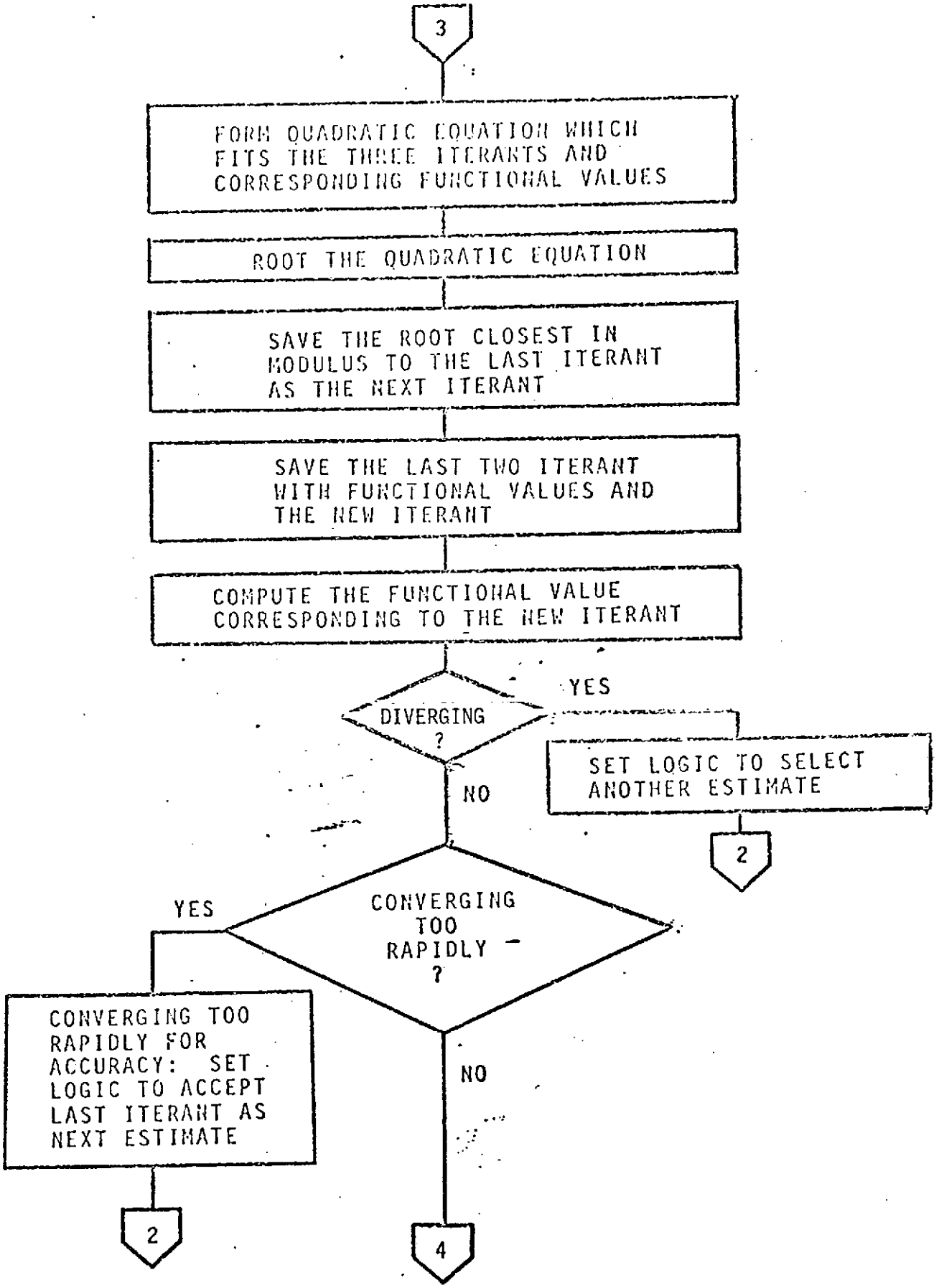

3
f:START FREQUENCY
?
=
≠
6
PHASE SHIFT = |PHA - PHL|
PHASE SHIFT
LESS THAN
MAXIMUM
?
NO
7
YES
PHASE SHIFT
GREATER THAN
MINIMUM
NO
8
4
YES
SAVE (f, AMP, PHA) FOR NYQUIST
AND BODE PLOTS
DETERMINE IF UNIT CIRCLE
CROSSING, 180 DEGREE CROSSING
OR GAIN PEAK OCCURRED WITH
LAST COMPUTED SYSTEM RESPONSE
5

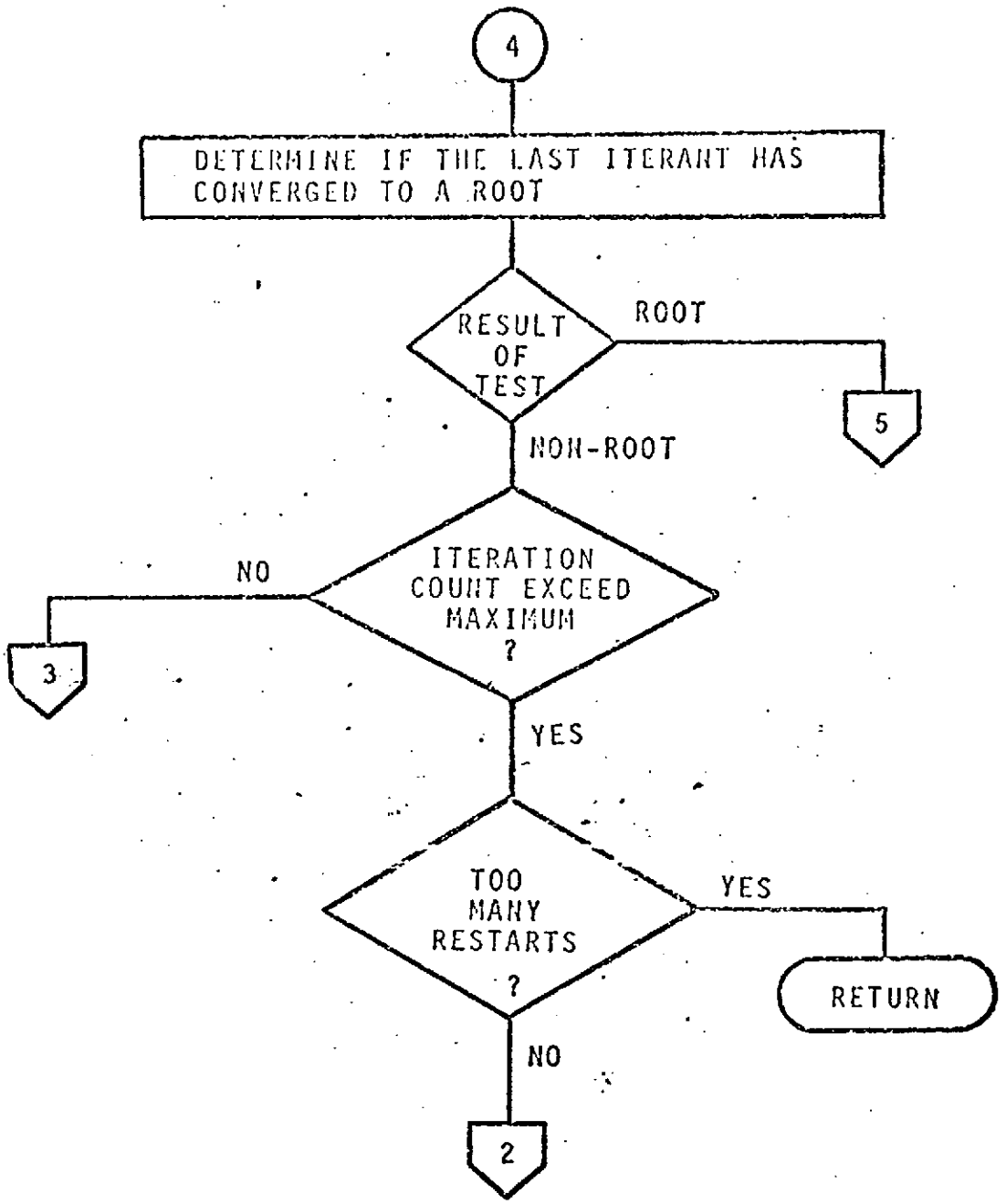

5
UNIT CIRCLE CROSSING ?
NO
YES
INTERPOLATE BETWEEN (f, AMP, PHA) AND (fL, AML, PHL) TO COMPUTE UNIT CIRCLE CROSSING FREQUENCY, PHASE AND DIRECTION.
180 DEGREE CROSSING ?
NO
YES
INTERPOLATE BETWEEN (f, AMP, PHA) AND (FL, AML, PHL) TO COMPUTE 180 DEGREE CROSSING FREQUENCY, AMPLITUDE, AND DIRECTION
GAIN PEAK ?
NO
YES
SAVE (FL, AML, PHL) AS GAIN PEAK FREQUENCY, AMPLITUDE, AND PHASE; DETERMINE DIRECTION
6

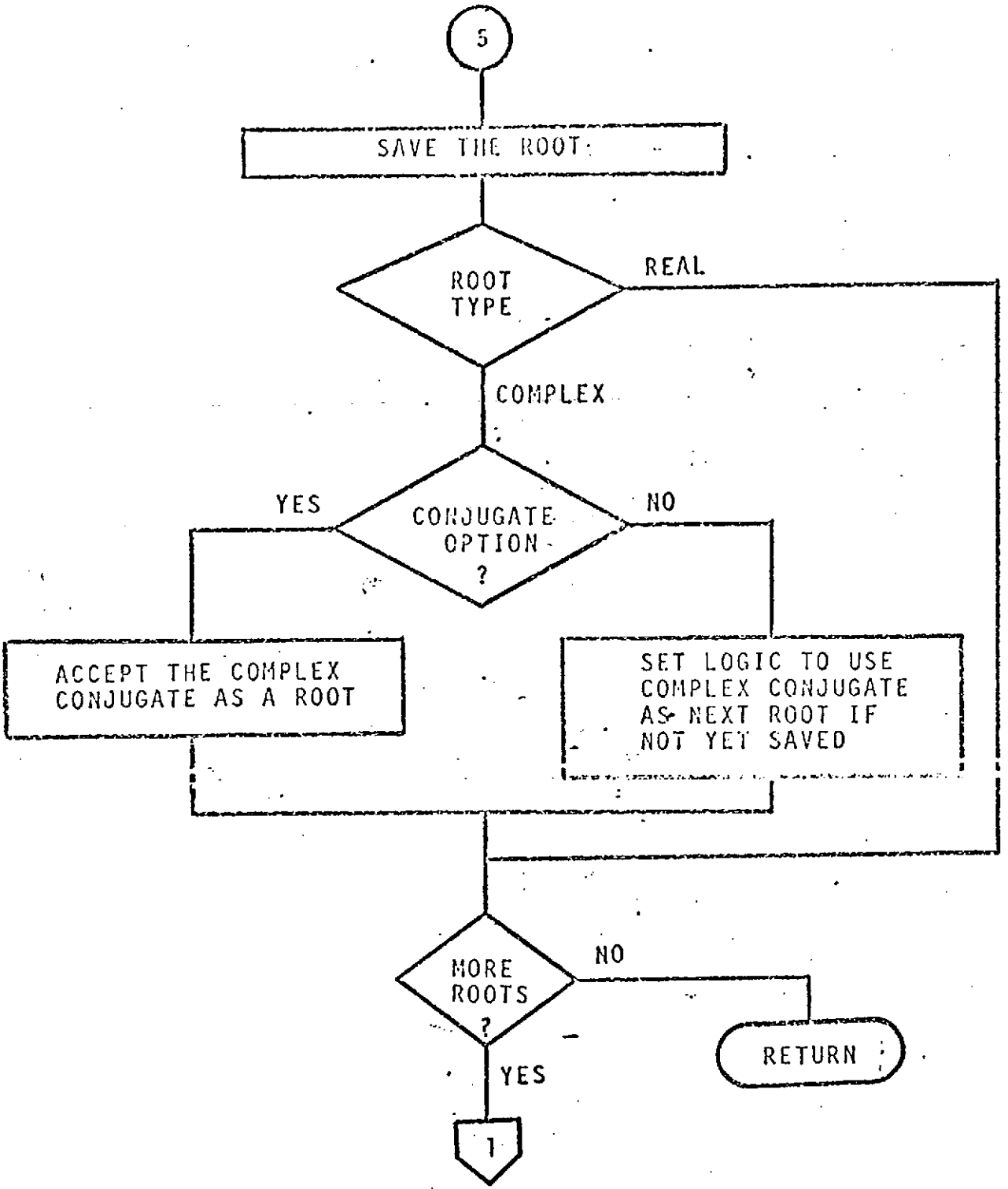

6
FL = F
AML = AMP
PHL = PHA
F: STOP FREQUENCY
<
F = F + ΔF
≥
2
ANOTHER FREQUENCY INTERVAL ?
YES
1
NO
RETURN

4.4 Characteristic Roots of a Matrix Polynomial


START
1
INITIALIZE THE ROOTING ROUTINES
2
INITIALIZE FOR THE NEXT ROOT (OR FIRST ROOT)
INITIALIZE FOR A NEW START FOR THE SAME ROOT
ESTIMATES AVAILABLE ?
NO
RETURN
YES
MAKE AN ESTIMATE OF A ROOT
DETERMINE FIRST THREE ITERANTS AND COMPUTE THE CORRESPONDING FUNCTIONAL VALUES
DETERMINE IF ALL OF THE CHARACTERISTIC ROOTS HAVE BEEN COMPUTED
RESULT ?
DONE
RETURN
NOT DONE
3


3
FORM QUADRATIC EQUATION WHICH FITS THE THREE ITERANTS AND CORRESPONDING FUNCTIONAL VALUES
ROOT THE QUADRATIC EQUATION
SAVE THE ROOT CLOSEST IN MODULUS TO THE LAST ITERANT AS THE NEXT ITERANT
SAVE THE LAST TWO ITERANT WITH FUNCTIONAL VALUES AND THE NEW ITERANT
COMPUTE THE FUNCTIONAL VALUE CORRESPONDING TO THE NEW ITERANT
DIVERGING ?
YES
SET LOGIC TO SELECT ANOTHER ESTIMATE
2
NO
CONVERGING TOO RAPIDLY ?
YES
CONVERGING TOO RAPIDLY FOR ACCURACY: SET LOGIC TO ACCEPT LAST ITERANT AS NEXT ESTIMATE
2
NO
4


4
DETERMINE IF THE LAST ITERANT HAS CONVERGED TO A ROOT
RESULT OF TEST
ROOT
5
NON-ROOT
ITERATION COUNT EXCEED MAXIMUM ?
NO
3
YES
TOO MANY RESTARTS ?
YES
RETURN
NO
2


5
SAVE THE ROOT
ROOT TYPE
REAL
COMPLEX
CONJUGATE OPTION ?
YES
NO
ACCEPT THE COMPLEX CONJUGATE AS A ROOT
SET LOGIC TO USE COMPLEX CONJUGATE AS NEXT ROOT IF NOT YET SAVED
MORE ROOTS ?
NO
YES
RETURN
1

5.0 Sample Problem

THE SAMPLED DATA STABILITY ANALYSIS PROGRAM

PITCH SSV FREQUENCY RESPONCE NO BENDING T= 61
CONTINUOUS SYSTEM MATRIX POLYNOMIAL

SAMPLING DEVICE (ROW 17 COLUMN 19)

NONZERO MATRIX ELEMENTS . . .

1	1	0	1.083n0-02	1	1	2	1.00000n+00	1	2	0	-2.66993-05	1	3	0	-3.39193-03		
1	3	2	-3.54395-03	1	15	0	1.3990n+00	1	15	2	1.64800-03	1	16	0	1.33400+00		
1	16	2	7.407n0-03	2	1	0	1.5552n+01	2	2	0	1.02557-03	2	2	1	-1.00000+00		
2	3	2	-4.89072-02	2	15	0	5.0137n+00	2	15	2	1.43600-02	2	16	0	1.8339n+01		
2	16	2	2.315n0-02	3	1	0	-1.5968n+01	3	1	2	-1.6680n+01	3	2	1	1.00000+00		
3	3	0	3.41252+02	3	3	1	1.4778n+00	3	3	2	1.00000+00	4	4	0	1.00000+00		
5	1	0	-1.000n0+n0	5	5	0	1.00000+00	6	1	1	-1.00000+00	6	6	0	1.57910+04		
6	6	1	2.01062+02	6	6	2	1.00000+00	7	1	0	1.5968n+01	7	1	2	2.92120+00		
7	2	1	-1.00000+00	7	7	0	3.19775+03	7	7	1	1.13097+02	7	7	2	1.00000+00		
8	5	0	-1.00000+00	8	8	0	1.00000+00	9	6	0	-1.57910+04	9	9	0	1.00000+00		
10	7	0	-3.19775+03	10	10	0	1.00000+00	10	10	1	1.00000+00	11	8	0	1.00000+00		
11	11	0	-1.00000+00	12	9	0	1.00000+00	12	12	0	-1.00000+00	13	10	0	1.00000+00		
13	13	0	-1.00000+00	14	11	0	9.0000n-01	14	12	0	1.50000+00	14	13	0	4.50000-02		
14	14	0	-1.00000+00	14	14	1	-5.4990n-01	14	14	2	-2.57840-02	14	14	3	-6.98600-04		
14	14	4	-1.065n0-05	14	14	5	-8.7299n-08	15	15	0	-1.00000+00	15	17	0	1.00000+00		
16	16	0	-1.00000+00	16	17	0	1.0000n+00	17	14	0	1.00000+00	17	14	1	5.00000-01		
17	17	0	-1.00000+00	18	14	0	1.0000n+00	18	14	1	5.00000-01	18	18	0	-1.00000+00		

PITCH SSV FREQUENCY RESPONCE NO BENDING T= 61
CALCULATION OF THE CONTINUOUS SYSTEM CLOSED LOOP POLES

E I G E N V A L U E S

ROOT NUMBER	SEQUENCE	CODE	ITERATION COUNT	REAL PART	IMAGINARY PART	DAMPING RATIO	FREQUENCY RAD/SEC	FREQUENCY HZ
1	14	3	2	-9.8408734+01	-7.9045897+01	7.7963-01	1.2622+02	2.0089+01
2	15	4	0	-9.8408734+01	7.9045897+01	7.7963-01	1.2622+02	2.0089+01
3	12	3	8	-7.5924552+01	1.6407054+01	9.7744-01	7.7677+01	1.2363+01
4	13	4	0	-7.5924552+01	-1.6407054+01	9.7744-01	7.7677+01	1.2363+01
5	10	3	6	-1.5369791+01	5.0099914+01	2.9329-01	5.2405+01	8.3404+00
6	11	4	0	-1.5369791+01	-5.0099914+01	2.9329-01	5.2405+01	8.3404+00
7	6	3	9	-2.5957020+01	1.6490638+01	8.4407-01	3.0752+01	4.8944+00
8	7	4	0	-2.5957020+01	-1.6490638+01	8.4407-01	3.0752+01	4.8944+00
9	8	3	7	-8.5658466-01	-1.9593204+01	4.3677-02	1.9612+01	3.1213+00
10	9	4	0	-8.5658466-01	1.9593204+01	4.3677-02	1.9612+01	3.1213+00
11	3	3	7	-1.6537968+00	-1.7687352+00	6.8298-01	2.4215+00	3.8539-01
12	4	4	0	-1.6537968+00	1.7687352+00	6.8298-01	2.4215+00	3.8539-01
13	5	3	4	-1.9956566+00	0.0000000			
14	2	3	6	-4.7309338-01	0.0000000			
15	1	3	3	4.3618746-04	0.0000000			

PITCH SSY FREQUENCY RESPONCE NO BENDING T= 61

OP-RD OP-RD T= 61

SYSTEM OPENED AT VARIABLE OP-RD

FREQUENCY INTERVALS . . .

START (CPS)	STOP (CPS)	PER CENT MAX. INCR.	MINIMUM PHASE SHIFT	MAXIMUM PHASE SHIFT	DETAIL PRINT
1.00000-02	1.00000+00	2.50000+01	2.00000+00	5.00000+00	*
1.00000+00	1.00000+01	2.50000+01	2.00000+00	5.00000+00	*

OPEN LOOP GAIN	ROW	COL	PHASE SHIFT TEST	NYQUIST PLOT	BODE PLOT
OP-RD	17	14	*		

PITCH SSV FREQUENCY RESPONCE NO BENDING T= 61

CPS	GAIN	AMPLITUDE	DECIBELS	PHASE	CPS	GAIN	AMPLITUDE	DECIBELS	PHASE
.01000	OP-RD	2.5063+02	4.7981+01	321.902	.01250	OP-RD	3.4135+02	5.0669+01	318.728
(126.65141)					(101.32110)				
.01312	OP-RD	3.7371+02	5.1451+01	316.401	.01375	OP-RD	4.1146+02	5.2287+01	313.155
(96.49628)					(92.11008)				
.01437	OP-RD	4.5497+02	5.3160+01	308.739	.01469	OP-RD	4.7873+02	5.3602+01	305.998
(88.10528)					(86.23070)				
.01500	OP-RD	5.0347+02	5.4039+01	302.846	.01531	OP-RD	5.2868+02	5.4464+01	299.244
(84.43422)					(82.71107)				
.01562	OP-RD	5.5354+02	5.4863+01	295.162	.01594	OP-RD	5.7691+02	5.5222+01	290.583
(81.05684)					(79.46749)				
.01609	OP-RD	5.8758+02	5.5381+01	288.111	.01625	OP-RD	5.9730+02	5.5524+01	285.521
(78.69596)					(77.93927)				
.01641	OP-RD	6.0586+02	5.5647+01	282.822	.01656	OP-RD	6.1304+02	5.5750+01	280.024
(77.19699)					(76.46871)				
.01672	OP-RD	6.1863+02	5.5829+01	277.139	.01687	OP-RD	6.2247+02	5.5882+01	274.183
(75.75405)					(75.05262)				
.01703	OP-RD	6.2441+02	5.5909+01	271.175	.01719	OP-RD	6.2436+02	5.5909+01	268.133
(74.36906)					(73.68803)				
.01734	OP-RD	6.2228+02	5.5880+01	265.079	.01750	OP-RD	6.1820+02	5.5823+01	262.034
(73.02417)					(72.37217)				
.01766	OP-RD	6.1219+02	5.5738+01	259.019	.01781	OP-RD	6.0439+02	5.5626+01	256.054
(71.73170)					(71.10248)				
.01797	OP-RD	5.9497+02	5.5490+01	253.156	.01812	OP-RD	5.8414+02	5.5330+01	250.341
(70.48419)					(69.87657)				
.01828	OP-RD	5.7212+02	5.5150+01	247.622	.01844	OP-RD	5.5914+02	5.4950+01	245.008

(69.27933)

(68.69221)

.01859	OP-RD	5.4543+02	5.4735+01	242.506	.01875	OP-RD	5.3119+02	5.4505+01	240.121
(68.11497)					(67.54734)				
.01891	OP-RD	5.1662+02	5.4263+01	237.855	.01906	OP-RD	5.0189+02	5.4012+01	235.709
(66.98910)					(66.44001)				
.01922	OP-RD	4.8714+02	5.3753+01	233.680	.01953	OP-RD	4.5807+02	5.3219+01	229.965
(65.89984)					(64.84544)				

PITCH SSV FREQUENCY RESPONCE NO BENDING T= 61

OP-RD OP-RD T= 61

CPS	GAIN	AMPLITUDE	DECIBELS	PHASE	CPS	GAIN	AMPLITUDE	DECIBELS	PHASE
.01984 (63.82425)	OP-RD	4.3013+02	5.2672+01	226.676	.02016 (62.83472)	OP-RD	4.0374+02	5.2122+01	223.776
.02047 (61.47541)	OP-RD	3.7910+02	5.1575+01	221.221	.02078 (60.94495)	OP-RD	3.5629+02	5.1036+01	218.971
.02141 (59.16552)	OP-RD	3.1594+02	4.9992+01	215.237	.02203 (57.48706)	OP-RD	2.8193+02	4.9003+01	212.320
.02266 (55.90120)	OP-RD	2.5325+02	4.8071+01	210.025	.02391 (52.97825)	OP-RD	2.0827+02	4.6373+01	206.757
.02516 (50.34577)	OP-RD	1.7511+02	4.4866+01	204.668	.02766 (45.79471)	OP-RD	1.3037+02	4.2303+01	202.477
.03266 (38.78302)	OP-RD	8.2867+01	3.8368+01	201.616	.03766 (33.63334)	OP-RD	5.8805+01	3.5388+01	202.444
.04766 (26.57573)	OP-RD	3.5446+01	3.0991+01	205.329	.05766 (21.96627)	OP-RD	2.4489+01	2.7779+01	208.330
.06766 (18.71940)	OP-RD	1.8327+01	2.5262+01	211.006	.07766 (16.30873)	OP-RD	1.4452+01	2.3198+01	213.298
.09766 (12.96846)	OP-RD	9.9282+00	1.9937+01	216.880	.11766 (10.76375)	OP-RD	7.9125+00	1.7399+01	219.482
.13766 (9.19964)	OP-RD	5.8298+00	1.5313+01	221.456	.15766 (8.03234)	OP-RD	4.7520+00	1.3538+01	223.032
.19766 (6.40635)	OP-RD	3.3977+00	1.0624+01	225.505	.23766 (5.32760)	OP-RD	2.5985+00	8.2944+00	227.467
.27766 (4.55960)	OP-RD	2.0821+00	6.3702+00	229.105	.31766 (3.98494)	OP-RD	1.7266+00	4.7439+00	230.485
.39766	OP-RD	1.2768+00	2.1222+00	232.579	.47766	OP-RD	1.0084+00	7.2675-02	233.912

(3.[illegible]00)					(2.64833)				
.55766	OP-RD	8.3174-01	-1.6002+00	234.628	.63766	OP-RD	7.0696-01	-3.0121+00	234.854
(2.26742)					(1.98195)				
.79766	OP-RD	5.4214-01	-5.3179+00	234.229	.95766	OP-RD	4.3745-01	-7.1814+00	232.610
(1.58247)					(1.31612)				
1.11766	OP-RD	3.6434-01	-8.7698+00	230.328					
(1.12573)									

PITCH SSV FREQUENCY RESPONCE NO BENDING T= 61

CPS	GAIN	AMPLITUDE	DECIBELS	PHASE	CPS	GAIN	AMPLITUDE	DECIBELS	PHASE
OP-RD					OP-RD T= 61				
1.00000	OP-RD	9.1573-01	-7.6238+00	232.060	1.25000	OP-RD	3.1825-01	-9.9446+00	228.082
(1.25984)					(1.00486)				
1.50000	OP-RD	2.5278-01	-1.1945+01	223.197	1.62500	OP-RD	2.2704-01	-1.2878+01	220.511
(.83432)					(.76853)				
1.75000	OP-RD	2.0448-01	-1.3787+01	217.692	1.87500	OP-RD	1.8438-01	-1.4686+01	214.759
(.71202)					(.66293)				
2.00000	OP-RD	1.6618-01	-1.5588+01	211.727	2.12500	OP-RD	1.4945-01	-1.6510+01	208.614
(.61987)					(.58177)				
2.25000	OP-RD	1.3381-01	-1.7470+01	205.443	2.37500	OP-RD	1.1890-01	-1.8496+01	202.265
(.54781)					(.51734)				
2.50000	OP-RD	1.0436-01	-1.9629+01	199.188	2.62500	OP-RD	8.9690-02	-2.0945+01	196.511
(.48983)					(.46485)				
2.87500	OP-RD	5.7810-02	-2.4760+01	199.440	2.90625	OP-RD	5.4109-02	-2.5335+01	202.878
(.42119)					(.41624)				
2.92187	OP-RD	5.2548-02	-2.5589+01	205.219	2.93750	OP-RD	5.1304-02	-2.5797+01	208.028
(.41380)					(.41139)				
2.95312	OP-RD	5.0513-02	-2.5932+01	211.303	2.96875	OP-RD	5.0339-02	-2.5962+01	214.966
(.40900)					(.40664)				
2.98437	OP-RD	5.0951-02	-2.5857+01	218.848	3.00000	OP-RD	5.2508-02	-2.5595+01	222.674
(.40430)					(.40198)				
3.01562	OP-RD	5.5126-02	-2.5173+01	226.085	3.03125	OP-RD	5.8817-02	-2.4610+01	228.731
(.39968)					(.39741)				
3.09375	OP-RD	8.0331-02	-2.1902+01	228.035	3.10937	OP-RD	8.5482-02	-2.1363+01	225.280
(.38854)					(.38638)				
3.12500	OP-RD	8.9717-02	-2.0943+01	221.915	3.14062	OP-RD	9.2799-02	-2.0649+01	218.206

(.38231)					(.38211)				
3.15625	OP-RD	9.4664-02	-2.0476+01	214.388	3.17187	OP-RD	9.5396-02	-2.0409+01	210.637
(.38001)					(.37792)				
3.18750	OP-RD	9.5174-02	-2.0430+01	207.072	3.20313	OP-RD	9.4209-02	-2.0518+01	203.757
(.37586)					(.37381)				
3.21875	OP-RD	9.2703-02	-2.0658+01	200.717	3.23437	OP-RD	9.0828-02	-2.0836+01	197.950
(.37178)					(.36978)				

PITCH SSV FREQUENCY RESPONCE NO BENDING T= 61

CPS	GAIN	AMPLITUDE	DECIBELS	PHASE	CPS	GAIN	AMPLITUDE	DECIBELS	PHASE
	OP-RD					OP-RD	T= 61		
3.25000 (.36779)	OP-RD	8.8718-02	-2.1040+01	195.439	3.26562 (.36581)	OP-RD	8.6471-02	-2.1263+01	193.161
3.28125 (.36386)	OP-RD	8.4161-02	-2.1498+01	191.087	3.31250 (.36000)	OP-RD	7.9523-02	-2.1990+01	187.456
3.34375 (.35621)	OP-RD	7.5020-02	-2.2496+01	184.361	3.37500 (.35249)	OP-RD	7.0736-02	-2.3007+01	181.663
3.40625 (.34883)	OP-RD	6.6681-02	-2.3520+01	179.256	3.43750 (.34523)	OP-RD	6.2852-02	-2.4034+01	177.063
3.46875 (.34170)	OP-RD	5.9222-02	-2.4550+01	175.026	3.53125 (.33480)	OP-RD	5.2476-02	-2.5601+01	171.262
3.59375 (.32813)	OP-RD	4.6281-02	-2.6692+01	167.720	3.65625 (.32167)	OP-RD	4.0510-02	-2.7849+01	164.236
3.71875 (.31542)	OP-RD	3.5079-02	-2.9099+01	160.675	3.78125 (.30935)	OP-RD	2.9916-02	-3.0482+01	156.885
3.84375 (.30347)	OP-RD	2.4984-02	-3.2047+01	152.664	3.90625 (.29776)	OP-RD	2.0261-02	-3.3867+01	147.674
3.93750 (.29497)	OP-RD	1.7978-02	-3.4906+01	144.704	3.96875 (.29221)	OP-RD	1.5749-02	-3.6055+01	141.249
4.00000 (.28950)	OP-RD	1.3582-02	-3.7341+01	137.075	4.01562 (.28816)	OP-RD	1.2527-02	-3.8043+01	134.618
4.03125 (.28683)	OP-RD	1.1496-02	-3.8789+01	131.836	4.04687 (.28551)	OP-RD	1.0490-02	-3.9584+01	128.627
4.06250 (.28419)	OP-RD	9.5179-03	-4.0430+01	124.890	4.07812 (.28289)	OP-RD	8.5853-03	-4.1325+01	120.466
4.08594	OP-RD	8.1376-03	-4.1790+01	117.928	4.09375	OP-RD	7.7061-03	-4.2263+01	115.153

([redacted]24)					(.28159)				
4.10156	OP-RD	7.2921-03	-4.2743+01	112.098	4.10937	OP-RD	6.8977-03	-4.3226+01	108.713
(.28095)					(.28031)				
4.11719	OP-RD	6.5271-03	-4.3706+01	104.974	4.12500	OP-RD	6.1843-03	-4.4174+01	100.866
(.27967)					(.27903)				
4.13281	OP-RD	5.8727-03	-4.4623+01	96.307	4.14063	OP-RD	5.5980-03	-4.5039+01	91.332
(.27839)					(.27776)				

PITCH SSV FREQUENCY RESPONCE NO BENDING T= 61

OP-RD OP-RD T= 61

CPS	GAIN	AMPLITUDE	DECIBELS	PHASE	CPS	GAIN	AMPLITUDE	DECIBELS	PHASE
4.14453 (.27744)	OP-RD	5.4761-03	-4.5231+01	88.687	4.14844 (.27713)	OP-RD	5.3654-03	-4.5408+01	85.907
4.15234 (.27681)	OP-RD	5.2665-03	-4.5570+01	83.036	4.15625 (.27650)	OP-RD	5.1794-03	-4.5714+01	80.123
4.16016 (.27619)	OP-RD	5.1059-03	-4.5839+01	77.075	4.16406 (.27587)	OP-RD	5.0457-03	-4.5942+01	73.964
4.16797 (.27556)	OP-RD	4.9997-03	-4.6021+01	70.783	4.17187 (.27525)	OP-RD	4.9677-03	-4.6077+01	67.570
4.17578 (.27494)	OP-RD	4.9498-03	-4.6108+01	64.340	4.17969 (.27463)	OP-RD	4.9486-03	-4.6110+01	61.052
4.18359 (.27432)	OP-RD	4.9597-03	-4.6091+01	57.823	4.18750 (.27401)	OP-RD	4.9857-03	-4.6046+01	54.612
4.19141 (.27370)	OP-RD	5.0275-03	-4.5973+01	51.419	4.19531 (.27339)	OP-RD	5.0814-03	-4.5880+01	48.313
4.19922 (.27308)	OP-RD	5.1477-03	-4.5768+01	45.288	4.20312 (.27277)	OP-RD	5.2274-03	-4.5634+01	42.340
4.20703 (.27246)	OP-RD	5.3213-03	-4.5480+01	39.466	4.21094 (.27216)	OP-RD	5.4244-03	-4.5313+01	36.714
4.21484 (.27185)	OP-RD	5.5390-03	-4.5131+01	34.062	4.21875 (.27154)	OP-RD	5.6644-03	-4.4937+01	31.514
4.22266 (.27124)	OP-RD	5.7967-03	-4.4736+01	29.091	4.22656 (.27093)	OP-RD	5.9402-03	-4.4524+01	26.763
4.23047 (.27063)	OP-RD	6.0909-03	-4.4306+01	24.597	4.23437 (.27033)	OP-RD	6.2516-03	-4.4080+01	22.426
4.24219	OP-RD	6.5850-03	-4.3629+01	18.514	4.25000	OP-RD	6.9451-03	-4.3166+01	14.962

([redacted]972)					(.26912)				
4.25781	OP-RD	7.3246-03	-4.2704+01	11.748	4.26562	OP-RD	7.7206-03	-4.2247+01	8.835
(.26851)					(.26791)				
4.27344	OP-RD	8.1303-03	-4.1798+01	6.187	4.28125	OP-RD	8.5497-03	-4.1361+01	3.775
(.26732)					(.26672)				
4.28906	OP-RD	8.9808-03	-4.0934+01	1.568	4.29688	OP-RD	9.4203-03	-4.0519+01	359.543
(.26613)					(.26553)				

PITCH SSV FREQUENCY RESPONCE NO BENDING T= 61

OP-RD OP-RD T= 61

CPS	GAIN	AMPLITUDE	DECIBELS	PHASE	CPS	GAIN	AMPLITUDE	DECIBELS	PHASE
4.31250	OP-RD	1.0319-02	-3.9727+01	355.954	4.32812	OP-RD	1.1291-02	-3.8984+01	352.866
(.26436)					(.26318)				
4.34375	OP-RD	1.2174-02	-3.8292+01	350.169	4.35937	OP-RD	1.3120-02	-3.7642+01	347.787
(.26202)					(.26087)				
4.37500	OP-RD	1.4072-02	-3.7033+01	345.654	4.40625	OP-RD	1.5993-02	-3.5922+01	341.966
(.25972)					(.25744)				
4.43750	OP-RD	1.7926-02	-3.4930+01	338.843	4.46875	OP-RD	1.9862-02	-3.4040+01	336.117
(.25520)					(.25298)				
4.50000	OP-RD	2.1800-02	-3.3231+01	333.684	4.53125	OP-RD	2.3730-02	-3.2494+01	331.463
(.25079)					(.24862)				
4.56250	OP-RD	2.5657-02	-3.1816+01	329.411	4.62500	OP-RD	2.9490-02	-3.0607+01	325.670
(.24649)					(.24229)				
4.68750	OP-RD	3.3286-02	-2.9555+01	322.260	4.75000	OP-RD	3.7045-02	-2.8625+01	319.074
(.23819)					(.23419)				
4.81250	OP-RD	4.0764-02	-2.7794+01	316.042	4.87500	OP-RD	4.9440-02	-2.7044+01	313.116
(.23028)					(.22645)				
4.93750	OP-RD	4.8074-02	-2.6362+01	310.269	5.00000	OP-RD	5.1662-02	-2.5737+01	307.478
(.22272)					(.21906)				
5.06250	OP-RD	5.5205-02	-2.5160+01	304.728	5.12500	OP-RD	5.8699-02	-2.4627+01	302.005
(.21548)					(.21197)				
5.18750	OP-RD	6.2145-02	-2.4132+01	299.304	5.25000	OP-RD	6.5539-02	-2.3670+01	296.615
(.20854)					(.20518)				
5.31250	OP-RD	6.8879-02	-2.3238+01	293.935	5.37500	OP-RD	7.2163-02	-2.2834+01	291.258
(.20189)					(.19866)				
5.43750	OP-RD	7.5389-02	-2.2454+01	288.581	5.50000	OP-RD	7.8554-02	-2.2097+01	285.905

(.[illegible]49)					(.19239)				
5.56250	OP-RD	8.1654-02	-2.1760+01	283.222	5.62500	OP-RD	8.4688-02	-2.1444+01	280.535
(.18934)					(.18635)				
5.68750	OP-RD	8.7651-02	-2.1145+01	277.839	5.75000	OP-RD	9.0539-02	-2.0863+01	275.134
(.18341)					(.18053)				
5.81250	OP-RD	9.3351-02	-2.0598+01	272.420	5.87500	OP-RD	9.6081-02	-2.0347+01	269.698
(.17769)					(.17491)				

PITCH SSV FREQUENCY RESPONCE NO BENDING T= 61

OP-RD OP-RD T= 61

CPS	GAIN	AMPLITUDE	DECIBELS	PHASE	CPS	GAIN	AMPLITUDE	DECIBELS	PHASE
5.93750 (.17217)	OP-RD	9.8726-02	-2.0111+01	266.965	6.00000 (.16948)	OP-RD	1.0128-01	-1.9889+01	264.222
6.06250 (.16684)	OP-RD	1.0375-01	-1.9681+01	261.468	6.12500 (.16424)	OP-RD	1.0611-01	-1.9485+01	258.704
6.18750 (.16167)	OP-RD	1.0838-01	-1.9301+01	255.930	6.25000 (.15915)	OP-RD	1.1054-01	-1.9129+01	253.146
6.31250 (.15667)	OP-RD	1.1260-01	-1.8969+01	250.353	6.37500 (.15423)	OP-RD	1.1455-01	-1.8820+01	247.550
6.43750 (.15183)	OP-RD	1.1638-01	-1.8683+01	244.741	6.50000 (.14946)	OP-RD	1.1809-01	-1.8556+01	241.922
6.56250 (.14712)	OP-RD	1.1969-01	-1.8439+01	239.099	6.62500 (.14482)	OP-RD	1.2116-01	-1.8333+01	236.269
6.68750 (.14255)	OP-RD	1.2251-01	-1.8237+01	233.435	6.75000 (.14031)	OP-RD	1.2373-01	-1.8150+01	230.597
6.81250 (.13811)	OP-RD	1.2483-01	-1.8074+01	227.755	6.87500 (.13593)	OP-RD	1.2580-01	-1.8007+01	224.913
6.93750 (.13378)	OP-RD	1.2669-01	-1.7949+01	222.070	7.00000 (.13166)	OP-RD	1.2735-01	-1.7900+01	219.228
7.06250 (.12957)	OP-RD	1.2794-01	-1.7860+01	216.387	7.12500 (.12751)	OP-RD	1.2839-01	-1.7829+01	213.551
7.18750 (.12547)	OP-RD	1.2873-01	-1.7807+01	210.718	7.25000 (.12345)	OP-RD	1.2894-01	-1.7792+01	207.890
7.31250 (.12146)	OP-RD	1.2903-01	-1.7786+01	205.069	7.37500 (.11950)	OP-RD	1.2900-01	-1.7788+01	202.256
7.43750	OP-RD	1.2886-01	-1.7798+01	199.451	7.50000	OP-RD	1.2861-01	-1.7815+01	196.656

([illegible]755)					(.11563)				
7.56250	OP-RD	1.2824-01	-1.7839+01	193.871	7.62500	OP-RD	1.2778-01	-1.7871+01	191.098
(.11373)					(.11186)				
7.68750	OP-RD	1.2722-01	-1.7909+01	188.338	7.75000	OP-RD	1.2657-01	-1.7954+01	185.590
(.11000)					(.10816)				
7.81250	OP-RD	1.2582-01	-1.8005+01	182.856	7.87500	OP-RD	1.2500-01	-1.8062+01	180.136
(.10634)					(.10455)				

PITCH SSV FREQUENCY RESPONCE NO BENDING T= 61

	CPS	GAIN	AMPLITUDE	DECIBELS	PHASE	CPS	GAIN	AMPLITUDE	DECIBELS	PHASE
OP-RD	7.93750	OP-RD	1.2409-01	-1.8125+01	177.432	8.00000	OP-RD	1.2311-01	-1.8194+01	174.792
	(.10277)					(.10100)				
OP-RD	8.06250	OP-RD	1.2206-01	-1.8268+01	172.069	8.12500	OP-RD	1.2095-01	-1.8348+01	169.411
	(.09926)					(.09753)				
OP-RD	8.18750	OP-RD	1.1978-01	-1.8432+01	166.770	8.25000	OP-RD	1.1856-01	-1.8521+01	164.146
	(.09582)					(.09412)				
OP-RD	8.31250	OP-RD	1.1729-01	-1.8615+01	161.537	8.37500	OP-RD	1.1598-01	-1.8712+01	158.946
	(.09244)					(.09078)				
OP-RD	8.43750	OP-RD	1.1463-01	-1.8814+01	156.370	8.50000	OP-RD	1.1324-01	-1.8920+01	153.812
	(.08913)					(.08750)				
OP-RD	8.56250	OP-RD	1.1183-01	-1.9029+01	151.270	8.62500	OP-RD	1.1039-01	-1.9141+01	148.743
	(.08588)					(.08427)				
OP-RD	8.68750	OP-RD	1.0893-01	-1.9257+01	146.232	8.75000	OP-RD	1.0795-01	-1.9376+01	143.737
	(.08267)					(.08109)				
OP-RD	8.81250	OP-RD	1.0596-01	-1.9497+01	141.257	8.87500	OP-RD	1.0446-01	-1.9621+01	138.792
	(.07953)					(.07797)				
OP-RD	8.93750	OP-RD	1.0295-01	-1.9747+01	136.341	9.00000	OP-RD	1.0194-01	-1.9876+01	133.903
	(.07643)					(.07489)				
OP-RD	9.06250	OP-RD	9.9930-02	-2.0006+01	131.478	9.12500	OP-RD	9.8418-02	-2.0138+01	129.065
	(.07337)					(.07186)				
OP-RD	9.18750	OP-RD	9.6913-02	-2.0272+01	126.666	9.25000	OP-RD	9.5414-02	-2.0408+01	124.277
	(.07036)					(.06887)				
OP-RD	9.31250	OP-RD	9.3924-02	-2.0544+01	121.898	9.37500	OP-RD	9.2445-02	-2.0682+01	119.529
	(.06739)					(.06592)				
OP-RD	9.43750	OP-RD	9.0979-02	-2.0821+01	117.170	9.50000	OP-RD	8.9528-02	-2.0961+01	114.820

([redacted]46)					(.06301)				
9.56250	OP-RD	8.8093-02	-2.1101+01	112.478	9.62500	OP-RD	8.6675-02	-2.1242+01	110.141
(.06157)					(.06014)				
9.68750	OP-RD	8.5277-02	-2.1383+01	107.813	9.75000	OP-RD	8.3899-02	-2.1525+01	105.489
(.05872)					(.05730)				
9.81250	OP-RD	8.2543-02	-2.1666+01	103.172	9.87500	OP-RD	8.1208-02	-2.1808+01	100.855
(.05589)					(.05449)				

PITCH SSV FREQUENCY RESPONCE NO BENDING T= 61

OP-RD OP-RD T= 61

CPS	GAIN	AMPLITUDE	DECIBELS	PHASE	CPS	GAIN	AMPLITUDE	DECIBELS	PHASE
9.93750 (.05310)	OP-RD	7.9897-02	-2.1949+01	98.544	10.00000 (.05171)	OP-RD	7.8611-02	-2.2090+01	96.236

TCH SSV FREQUENCY RESPONCE NO BENDING T= 61

OP-RD OP-RD T= 61

SYSTEM OPENED AT VARIABLE OP-RD

	UNIT AMPLITUDES			180 DEGREE PHASES				GAIN PEAKS			
	CPS	AMPLITUDES	PHASES	PHASE MARGIN	CPS	AMPLITUDES	DECIBELS	PHASES	CPS	AMPLITUDES	DECIBELS	PHASES
OP-RD	.4815	1.0000 DEC	233.95	53.95	3.3966	6.7934-02	-23.36	180.00 CW	.0170	6.2441+02	55.91	271.17 CW
	(2.6274)				(.3509)				(74.3641)			
					7.878)	1.2495-01	-18.07	180.00 CW	3.1719	9.5396-02	-20.41	210.64 CW
					(.1045)				(.3779)			
									7.3125	1.2903-01	-17.79	205.07 CW
									(.1215)			

PLOT ARRAY FOR NICHOLS PLOT - - NO. OF POINTS = 256

PHASE VS. DB

-38.0984	47.9806	-41.2718	50.6640	-43.5986	51.4507	-46.8449	52.2866	-51.2608	53.1597
-54.0022	53.6018	-57.1541	54.0395	-60.7558	54.4638	-64.8384	54.8630	-69.4166	55.2222
-71.8893	55.3814	-74.4788	55.5239	-77.1778	55.6475	-79.9762	55.7498	-82.8610	55.8287
-85.8168	55.8824	-88.8255	55.9094	-91.9673	55.9087	-94.9213	55.8797	-97.9661	55.8225
-100.9810	55.7377	-103.9461	55.6264	-106.8439	55.4900	-109.6588	55.3304	-112.3783	55.1498
-114.9924	54.9505	-117.4941	54.7347	-119.8789	54.5050	-122.1446	54.2634	-124.2911	54.0121
-126.3197	53.7531	-130.0354	53.2187	-133.3235	52.6720	-136.2240	52.1219	-138.7790	51.5751
-141.0292	51.0360	-144.7627	49.9920	-147.6798	49.0027	-149.9750	48.0712	-153.2425	46.3725
-155.3318	44.8663	-157.5227	42.3032	-158.3843	38.3676	-157.5564	35.3883	-154.6707	30.9913
-151.6704	27.7793	-148.9943	25.2619	-146.7022	23.1983	-143.1201	19.9374	-140.5178	17.3993
-138.5444	15.3130	-136.9677	13.5376	-134.4947	10.6237	-132.5333	8.2944	-130.8949	6.3702
-129.5146	4.7439	-127.4208	2.1222	-126.0882	.0727	-125.3724	-1.6002	-125.1458	-3.0121
-125.7709	-5.3179	-127.3903	-7.1814	-129.6723	-8.7698	-127.9396	-7.6238	-131.9184	-9.9446
-136.8030	-11.9450	-139.4892	-12.8781	-142.3077	-13.7871	-145.2405	-14.6859	-148.2729	-15.5884
-151.3864	-16.5103	-154.5572	-17.4705	-157.7351	-18.4962	-160.8116	-19.6293	-163.4889	-20.9451
-160.5597	-24.7599	-157.1219	-25.3345	-154.7815	-25.5890	-151.9716	-25.7970	-148.6969	-25.9319
-145.0341	-25.9619	-141.1517	-25.8570	-137.3262	-25.5955	-133.9153	-25.1729	-131.2693	-24.6099
-131.9646	-21.9023	-134.7199	-21.3625	-138.0850	-20.9425	-141.7938	-20.6492	-145.6124	-20.4763
-149.3627	-20.4094	-152.9278	-20.4297	-156.2429	-20.5181	-159.2830	-20.6581	-162.0502	-20.8356
-164.5610	-21.0398	-166.8395	-21.2625	-168.9126	-21.4977	-172.5444	-21.9902	-175.6386	-22.4965
-178.3368	-23.0072	-180.7439	-23.5199	-182.9372	-24.0338	-184.9738	-24.5504	-188.7384	-25.6008
-192.2802	-26.6920	-195.7641	-27.8487	-199.3252	-29.0991	-203.1155	-30.4820	-207.3355	-32.0467
-212.3259	-33.8669	-215.2959	-34.9061	-218.7511	-36.0552	-222.9247	-37.3408	-225.3817	-38.0428
-228.1643	-38.7889	-231.3735	-39.5844	-235.1097	-40.4297	-239.5339	-41.3249	-242.0722	-41.7901
-244.8471	-42.2634	-247.9022	-42.7430	-251.2873	-43.2259	-255.0255	-43.7056	-259.1340	-44.1742
-263.6933	-44.6233	-268.6680	-45.0393	-271.3127	-45.2306	-274.0929	-45.4080	-276.9638	-45.5695
-279.8765	-45.7145	-282.9251	-45.8386	-286.0362	-45.9416	-289.2167	-46.0211	-292.4303	-46.0769
-295.6604	-46.1083	-298.9482	-46.1104	-302.1772	-46.0909	-305.3884	-46.0455	-308.5809	-45.9729
-311.6867	-45.8804	-314.7116	-45.7677	-317.6598	-45.6343	-320.5336	-45.4797	-323.2856	-45.3130
-325.9378	-45.1314	-328.4857	-44.9369	-330.9092	-44.7363	-333.2373	-44.5239	-335.4531	-44.3064

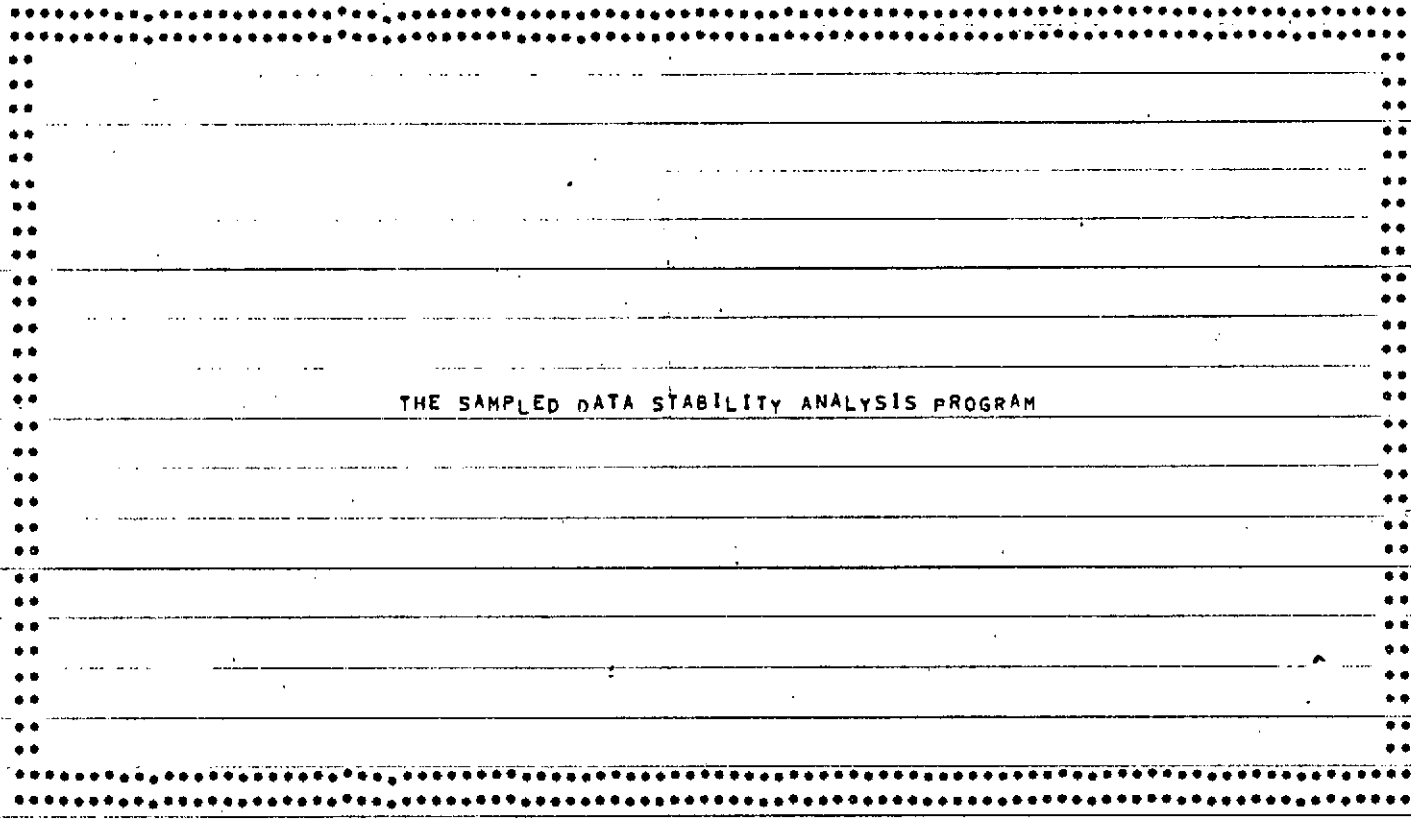
-337.5	-44.0802	-341.4861	-43.6289	-345.0376	-43.1664	-348.2519	-42.7043	-351.1654	-42.2470
-353.8129	-41.7979	-356.2247	-41.3610	-358.4316	-40.9337	-360.4568	-40.5187	-364.0462	-39.7270
-367.1342	-38.9841	-369.8310	-38.2916	-372.2134	-37.6416	-374.3459	-37.0328	-378.0340	-35.9217
-381.1569	-34.9305	-383.8827	-34.0395	-386.3160	-33.2310	-388.5371	-32.4941	-390.5888	-31.8159
-394.3304	-30.6066	-397.7397	-29.5547	-400.9259	-28.6253	-403.9582	-27.7944	-406.8838	-27.0444
-409.7306	-26.3618	-412.5217	-25.7365	-415.2716	-25.1604	-417.9948	-24.6274	-420.6962	-24.1319
-423.3845	-23.6701	-426.0650	-23.2383	-428.7416	-22.8337	-431.4187	-22.4539	-434.0946	-22.0966
-436.7780	-21.7604	-439.4654	-21.4436	-442.1609	-21.1449	-444.8660	-20.8633	-447.5798	-20.5977
-450.3017	-20.3473	-453.0347	-20.1114	-455.7784	-19.8893	-458.5324	-19.6806	-461.2958	-19.4846
-464.0702	-19.3010	-466.8535	-19.1293	-469.6472	-18.9692	-472.4495	-18.8204	-475.2594	-18.6826
-478.0777	-18.5555	-480.9013	-18.4391	-483.7313	-18.3328	-486.5654	-18.2367	-489.4032	-18.1504
-492.2453	-18.0736	-495.0871	-18.0066	-497.9299	-17.9488	-500.7723	-17.9000	-503.6126	-17.8602
-506.4494	-17.8292	-509.2821	-17.8067	-512.1097	-17.7924	-514.9309	-17.7863	-517.7444	-17.7881
-520.5488	-17.7979	-523.3442	-17.8148	-526.1286	-17.8392	-528.9017	-17.8707	-531.6625	-17.9089
-534.4102	-17.9537	-537.1441	-18.0048	-539.8637	-18.0620	-542.5684	-18.1253	-545.2577	-18.1942
-547.9312	-18.2684	-550.5887	-18.3478	-553.2299	-18.4322	-555.8543	-18.5211	-558.4627	-18.6147
-561.0541	-18.7124	-563.6296	-18.8143	-566.1880	-18.9198	-568.7305	-19.0290	-571.2567	-19.1415
-573.7677	-19.2572	-576.2626	-19.3757	-578.7427	-19.4971	-581.2077	-19.6209	-583.6593	-19.7472
-586.0972	-19.8757	-588.5216	-20.0061	-590.9348	-20.1385	-593.3344	-20.2723	-595.7235	-20.4077
-598.1024	-20.5445	-600.4707	-20.6823	-602.8297	-20.8212	-605.1800	-20.9609	-607.5220	-21.1012
-609.8591	-21.2422	-612.1870	-21.3834	-614.5106	-21.5249	-616.8284	-21.6664	-619.1449	-21.8080
-621.4563	-21.9494	-623.7636	-22.0903	.0000	.0000	.0000	.0000	.0000	.0000

OPEN LOOP POLES IN THE Z DOMAIN

(ZERO ORDER HOLD NEGLECTED **• STAR INDICATES MAGNITUDE OUTSIDE UNITY CIRCLE)

(PLUS INDICATES MAGNITUDE WITHIN 0.0001 AND TERM DELETED FROM Z DOMAIN TRANSFER FUNCTION)

POLE		MAGNITUDE
REAL	IMAGINARY	
-1.77891-02	2.24908-03	1.79307-02
-1.77891-02	-2.24908-03	1.79307-02
1.04147-01	5.45595-04	1.04148-01
1.04147-01	-5.45595-04	1.04148-01
8.34500-02	1.61272-01	1.81584-01
8.34500-02	-1.61272-01	1.81584-01
-9.39792-02	4.90683-01	4.99602-01
-9.39792-02	-4.90683-01	4.99602-01
6.86642-01	6.81469-01	9.67407-01
6.86642-01	-6.81469-01	9.67407-01
9.23286-01	0.00000	9.23286-01
9.60789-01	0.00000	9.60789-01
9.99304-01	4.33479-03	9.99313-01
9.99304-01	-4.33479-03	9.99313-01
1.00142+00	0.00000	1.00142+00•




```
@PMD,E

PMD 0029-05/20- 11:03:23
@FIN
```

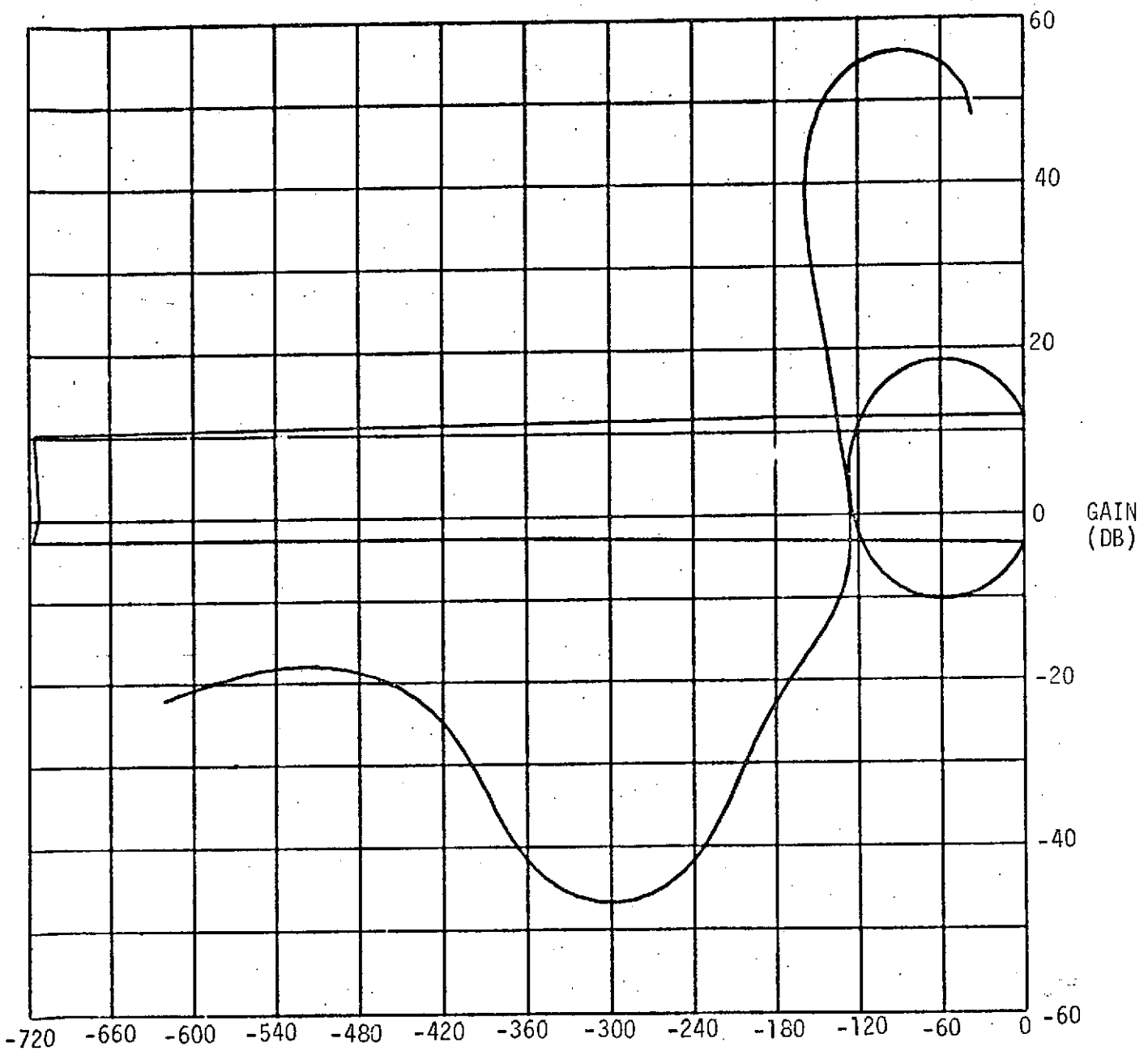


FIGURE 5-1 SPACE SHUTTLE PITCH AXES NICHOLS PLOT

T = 61 Sec. - 1 Slosh Mode - No Bending

6.0 Program Listing

```
$ASG,AX B.
FAC WARNING          04000000n200

$ASG,T A.,F2///500


$COPY B.,A.
FURPUR 0026-05/23-08:25
     176 BLOCKS COPIED
$USE F.,A.


$PRT,T F.
FURPUR 0026-05/23-08:25

FD36-001121*A      ELEMENT TABLE

  D  NAME      VERSION      TYPE          DATE        TIME       SEQ #  SIZE-PRE*TEXT (CYCLE WORD) PSRMODE  LOCATION
     DET                   FOR SYMB      05 FEB 74   06:04:00       1          3     5    0    1            1792
     DET                   RELOCATABLE   05 FEB 74   06:04:04       2    1     3                            1795
     I!PRL                 FOR SYMB      05 FEB 74   06:05:27       3         84     5    0    1            1799
     I!PRL                 RELOCATABLE   05 FEB 74   06:05:32       4    2    18                            1883
     I!ATA                 ELT SYMB      05 FEB 74   06:10:15       5         43     5    0    1            1903
     ADDZOH                FOR SYMB      14 FEB 74   11:43:08       6         18     5    2    3            1946
     ADDZOH                RELOCATABLE   14 FEB 74   11:43:14       7    2     4                            1964
     AFTVAR                FOR SYMB      14 FEB 74   11:43:20       8          8     5    2    3            1970
     AFTVAR                RELOCATABLE   14 FEB 74   11:43:25       9    1     2                            1978
     AGAIN                 FOR SYMB      14 FEB 74   11:43:27      10         42     5    2    3            1981
     AGAIN                 RELOCATABLE   14 FEB 74   11:43:36      11    2    12                            2023
     ALGIN                 FOR SYMB      14 FEB 74   11:43:39      12         28     5    2    3            2037
     ALGIN                 RELOCATABLE   14 FEB 74   11:43:41      13    2     4                            2065
     BIA091                FOR SYMB      14 FEB 74   11:43:43      14         22     5    2    3            2071
     BIA091                RELOCATABLE   14 FEB 74   11:43:44      15    2     4                            2093
     BHA272                FOR SYMB      14 FEB 74   11:43:54      16         59     5    3    4            2099
     BHA272                RELOCATABLE   14 FEB 74   11:43:58      17    4    13                            2158
     BIA341                FOR SYMB      14 FEB 74   11:44:02      18         71     5    2    3            2175
     BIA341                RELOCATABLE   14 FEB 74   11:44:05      19    3    14                            2246
     BLOCK                 FOR SYMB      14 FEB 74   11:44:12      20         38     5    4    5            2263
     BLOCK                 RELOCATABLE   14 FEB 74   11:44:14      21    2    11                            2301
     BODE                  FOR SYMB      14 FEB 74   11:44:20      22          1     5    2    3            2314
     BODE                  RELOCATABLE   14 FEB 74   11:44:24      23    1     1                            2315
     CIABV                 FOR SYMB      14 FEB 74   11:44:27      24          9     5    2    3            2317
     CIABV                 RELOCATABLE   14 FEB 74   11:44:31      25    1     4                            2326
     CIPOLE                FOR SYMB      14 FEB 74   11:44:37      26         17     5    2    3            2331
     CIPOLE                RELOCATABLE   14 FEB 74   11:44:39      27    2     3                            2348
     COMPUT                FOR SYMB      14 FEB 74   11:44:45      28         46     5    2    3            2353
     COMPUT                RELOCATABLE   14 FEB 74   11:44:50      29    3     7                            2399
     CSMTRX                FOR SYMB      14 FEB 74   11:44:54      30         16     5    2    3            2409
     CSMTRX                RELOCATABLE   14 FEB 74   11:44:57      31    2     9                            2425
     CSOLTF                FOR SYMB      14 FEB 74   11:44:59      32         47     5    7    5            2436
```

CSOLTF	RELOCATABLE	14 FEB 74	11:45:13	33	3	11					2483
DF	FOR SYMB	14 FEB 74	11:45:29	34		4	5	2	3		2497
DF	RELOCATABLE	14 FEB 74	11:45:30	35	1	2					2501
DFLZOH	FOR SYMB	14 FEB 74	11:45:48	36		9	5	2	3		2504
DFLZOH	RELOCATABLE	14 FEB 74	11:45:49	37	1	3					2513
DFTCS	FOR SYMB	14 FEB 74	11:45:53	38		40	5	2	3		2517
DFTCS	RELOCATABLE	14 FEB 74	11:45:56	39	2	15					2563
EFROR	FOR SYMB	14 FEB 74	11:46:18	40		7	5	2	3		2580
EFROR	RELOCATABLE	14 FEB 74	11:46:19	41	1	2					2587
EIXIT	FOR SYMB	14 FEB 74	11:46:38	42		8	5	2	3		2590
EIXIT	RELOCATABLE	14 FEB 74	11:46:46	43	1	3					2598
FORM	FOR SYMB	14 FEB 74	11:46:52	44		34	5	2	3		2602
FORM	RELOCATABLE	14 FEB 74	11:47:06	45	1	13					2636
FRMTX	FOR SYMB	14 FEB 74	11:47:11	46		25	5	3	4		2650
FRMTX	RELOCATABLE	14 FEB 74	11:47:27	47	2	8					2675
GENMTX	FOR SYMB	14 FEB 74	11:47:32	48		94	5	2	3		2685
GENMTX	RELOCATABLE	14 FEB 74	11:47:50	49	2	21					2779
GITEST	FOR SYMB	14 FEB 74	11:47:53	50		6	5	2	3		2802
GITEST	RELOCATABLE	14 FEB 74	11:47:58	51	1	3					2800
IHITAL	FOR SYMB	14 FEB 74	11:48:26	52		33	5	2	3		2812
IHITAL	RELOCATABLE	14 FEB 74	11:48:34	53	2	4					2845
IHPEST	FOR SYMB	14 FEB 74	11:48:47	54		35	5	3	4		2851
IHPEST	RELOCATABLE	14 FEB 74	11:48:50	55	2	6					2886
IHPMTX	FOR SYMB	14 FEB 74	11:48:53	56		37	5	2	3		2894
IHPMTX	RELOCATABLE	14 FEB 74	11:48:57	57	2	4					2931
IHPNYQ	FOR SYMB	14 FEB 74	11:49:00	58		44	5	2	3		2937
IHPNYQ	RELOCATABLE	14 FEB 74	11:49:03	59	2	7					2981
IHPUT	FOR SYMB	14 FEB 74	11:49:08	60		24	5	2	3		2990
IHPUT	RELOCATABLE	14 FEB 74	11:49:17	61	2	3					3014
IHT1	FOR SYMB	14 FEB 74	11:49:20	62		4	5	2	3		3019
IHT1	RELOCATABLE	14 FEB 74	11:49:22	63	1	2					3023
IHT2	FOR SYMB	14 FEB 74	11:49:33	64		3	5	2	3		3026
IHT2	RELOCATABLE	14 FEB 74	11:49:35	65	1	1					3029
KCALC	FOR SYMB	14 FEB 74	11:49:37	66		16	5	2	3		3031
KCALC	RELOCATABLE	14 FEB 74	11:49:38	67	2	6					3047
LIMIT	FOR SYMB	14 FEB 74	11:49:40	68		15	5	2	3		3055
LIMIT	RELOCATABLE	14 FEB 74	11:49:42	69	1	3					3070
MLTZRO	FOR SYMB	14 FEB 74	11:49:46	70		142	5	2	3		3074
MLTZRO	RELOCATABLE	14 FEB 74	11:50:06	71	3	34					3216
MODZRO	FOR SYMB	14 FEB 74	11:50:13	72		155	5	2	3		3253
MODZRO	RELOCATABLE	14 FEB 74	11:50:51	73	3	40					3408
MZTRAN	FOR SYMB	14 FEB 74	11:50:56	74		136	5	2	3		3451
MZTRAN	RELOCATABLE	14 FEB 74	11:51:18	75	3	44					3587
NORMTX	FOR SYMB	14 FEB 74	11:51:21	76		16	5	2	3		3634
NORMTX	RELOCATABLE	14 FEB 74	11:51:23	77	2	3					3650
NQNIST	FOR SYMB	14 FEB 74	11:51:25	78		1	5	2	3		3655
NQNIST	RELOCATABLE	14 FEB 74	11:51:26	79	1	1					3656
PFE	FOR SYMB	14 FEB 74	11:52:09	80		32	5	3	4		3658
PFE	RELOCATABLE	14 FEB 74	11:52:13	81	2	14					3690
PFEZRO	FOR SYMB	14 FEB 74	11:52:15	82		50	5	3	4		3706
PFEZRO	RELOCATABLE	14 FEB 74	11:52:20	83	2	11					3756
POINT	FOR SYMB	14 FEB 74	11:52:25	84		89	5	2	3		3769
POINT	RELOCATABLE	14 FEB 74	11:52:44	85	3	29					3858
POLES	FOR SYMB	14 FEB 74	11:52:46	86		17	5	2	3		3890
POLES	RELOCATABLE	14 FEB 74	11:52:48	87	2	3					3907
POLVAL	FOR SYMB	14 FEB 74	11:52:50	88		20	5	2	3		3912
POLVAL	RELOCATABLE	14 FEB 74	11:52:52	89	2	5					3932

PREVAR	FOR SYMB	14 FEB 74	11:52:53	90		11	5	2	3	3939
PREVAR	RELOCATABLE	14 FEB 74	11:52:55	91	2	3				3950
PRINTT	FOR SYMB	14 FEB 74	11:52:59	92		34	5	5	5	3955
PRINTT	RELOCATABLE	14 FEB 74	11:53:03	93	2	12				3989
PJTOUT	FOR SYMB	14 FEB 74	11:53:06	94		54	5	2	3	4003
PJTOUT	RELOCATABLE	14 FEB 74	11:53:11	95	3	19				4057
RAMNTX	FOR SYMB	14 FEB 74	11:53:13	96		5	5	2	3	4079
RAMNTX	RELOCATABLE	14 FEB 74	11:53:14	97	1	2				4084
RLOCUS	FOR SYMB	14 FEB 74	11:53:29	98		25	5	2	3	4087
RLOCUS	RELOCATABLE	14 FEB 74	11:53:32	99	2	8				4112
RLPLOT	FOR SYMB	14 FEB 74	11:53:35	100		5	5	2	3	4122
RLPLOT	RELOCATABLE	14 FEB 74	11:53:37	101	1	1				4127
ROOTER	FOR SYMB	14 FEB 74	11:53:55	102		91	5	2	3	4129
ROOTER	RELOCATABLE	14 FEB 74	11:54:02	103	2	15				4220
RJTER	FOR SYMB	14 FEB 74	11:54:06	104		44	5	3	4	4237
RJTER	RELOCATABLE	14 FEB 74	11:54:09	105	3	8				4281
SAVE	FOR SYMB	14 FEB 74	11:54:13	106		36	5	1	2	4292
SAVE	RELOCATABLE	14 FEB 74	11:54:15	107	2	8				4328
SAVRUT	FOR SYMB	14 FEB 74	11:54:20	108		21	5	1	2	4338
SAVRUT	RELOCATABLE	14 FEB 74	11:54:22	109	2	4				4359
SEPDPC	FOR SYMB	14 FEB 74	11:54:30	110		10	5	1	2	4365
SEPDPC	RELOCATABLE	14 FEB 74	11:54:33	111	1	2				4375
SOLVE	FOR SYMB	14 FEB 74	11:54:36	112		24	5	1	2	4378
SOLVE	RELOCATABLE	14 FEB 74	11:54:38	113	2	21				4402
START	FOR SYMB	14 FEB 74	11:55:00	114		29	5	1	2	4425
START	RELOCATABLE	14 FEB 74	11:55:03	115	2	15				4454
STNNYQ	FOR SYMB	14 FEB 74	11:55:06	116		15	5	1	2	4471
STNNYQ	RELOCATABLE	14 FEB 74	11:55:08	117	2	3				4486
STNRL	FOR SYMB	14 FEB 74	11:55:11	118		30	5	1	2	4491
STNRL	RELOCATABLE	14 FEB 74	11:55:19	119	1	6				4521
SUMMRY	FOR SYMB	14 FEB 74	11:55:26	120		99	5	1	2	4528
SUMMRY	RELOCATABLE	14 FEB 74	11:55:39	121	2	44				4627
SYSERQ	FOR SYMB	14 FEB 74	11:55:44	122		7	5	1	2	4673
SYSERQ	RELOCATABLE	14 FEB 74	11:55:46	123	1	2				4680
TEST	FOR SYMB	14 FEB 74	11:55:50	124		38	5	1	2	4683
TEST	RELOCATABLE	14 FEB 74	11:55:56	125	2	9				4721
TRNSFR	FOR SYMB	14 FEB 74	11:55:59	126		21	5	1	2	4732
TRNSFR	RELOCATABLE	14 FEB 74	11:56:10	127	2	4				4753
VRYGEN	FOR SYMB	14 FEB 74	11:56:13	128		42	5	1	2	4759
VRYGEN	RELOCATABLE	14 FEB 74	11:56:16	129	2	10				4801
WRITE	FOR SYMB	14 FEB 74	11:56:20	130		36	5	1	2	4813
WRITE	RELOCATABLE	14 FEB 74	11:56:23	131	2	10				4849
XCNG	FOR SYMB	14 FEB 74	11:56:24	132		6	5	1	2	4861
XCNG	RELOCATABLE	14 FEB 74	11:56:26	133	1	2				4867
ZEEPLS	FOR SYMB	14 FEB 74	11:56:30	134		17	5	2	3	4870
ZEEPLS	RELOCATABLE	14 FEB 74	11:56:32	135	2	7				4887
ZEROS	FOR SYMB	14 FEB 74	11:56:40	136		17	5	1	2	4896
ZEROS	RELOCATABLE	14 FEB 74	11:56:43	137	2	3				4913
ZTRAN	FOR SYMB	14 FEB 74	11:56:48	138		121	5	1	2	4918
ZTRAN	RELOCATABLE	14 FEB 74	11:57:01	139	3	40				5039
SAVEST	FOR SYMB	15 FEB 74	01:52:32	140		5	5	0	1	5082
SAVEST	RELOCATABLE	15 FEB 74	01:52:36	141	1	3				5087
DECIDE	FOR SYMB	20 FEB 74	23:26:46	142		17	5	3	4	5091
DECIDE	RELOCATABLE	20 FEB 74	23:26:48	143	2	6				5108
VRYRAN	FOR SYMB	20 FEB 74	23:27:15	144		5	5	1	2	5116
VRYRAN	RELOCATABLE	20 FEB 74	23:27:16	145	1	2				5121
ESTMAT	FOR SYMB	28 FEB 74	03:44:16	146		52	5	4	5	5124

ESTMAT	RELOCATABLE	28	FEB	74	03:44:24	147	2	12				5176
PEVAL	FOR SYMB	28	FEB	74	03:44:41	148		31	5	3	4	5190
PEVAL	RELOCATABLE	28	FEB	74	03:44:48	149	2	13				5221
DETSD	FOR SYMB	02	MAR	74	08:29:49	150		24	5	5	5	5236
DETSD	RELOCATABLE	02	MAR	74	08:29:56	151	2	24				5260
SRRL	FOR SYMB	02	MAR	74	08:29:59	152		47	5	4	5	5296
SRRL	RELOCATABLE	02	MAR	74	08:30:05	153	3	10				5333
RLPRNT	FOR SYMB	05	MAR	74	03:24:37	154		71	5	6	5	5346
RLPRNT	RELOCATABLE	05	MAR	74	03:24:48	155	3	28				5417
SRRLPP	FOR SYMB	05	MAR	74	03:32:08	156		60	5	3	4	5448
SRRLPP	RELOCATABLE	05	MAR	74	03:32:14	157	3	12				5508
SCALE	FOR SYMB	05	MAR	74	22:11:45	158		12	5	2	3	5523
SCALE	RELOCATABLE	05	MAR	74	22:11:47	159	1	8				5535
AERO	MAP SYMB	13	MAY	74	13:23:25	160		8	5	3	4	5544
MAIN	FOR SYMB	20	MAY	74	10:21:49	161		08	5	1	2	5552
MAIN	RELOCATABLE	20	MAY	74	10:21:53	162	4	15				5640
DATA	FOR SYMB	20	MAY	74	10:25:21	163		114	5	2	3	5659
DATA	RELOCATABLE	20	MAY	74	10:27:06	164	4	22				5773
GRAPHS	FOR SYMB	20	MAY	74	10:29:41	165		20	5	1	2	5799
GRAPHS	RELOCATABLE	20	MAY	74	10:29:43	166	2	2				5819
NICHOL	FOR SYMB	20	MAY	74	10:29:52	167		8	5	7	5	5823
NICHOL	RELOCATABLE	20	MAY	74	10:30:06	168	2	7				5831
OUTPUT	FOR SYMB	20	MAY	74	10:30:20	169		63	5	4	5	5840
OUTPUT	RELOCATABLE	20	MAY	74	10:41:12	170	3	21				5903
RESET	FOR SYMB	20	MAY	74	10:43:16	171		64	5	1	2	5927
RESET	RELOCATABLE	20	MAY	74	10:48:46	172	4	5				5991
• AERO	ABSOLUTE	20	MAY	74	10:55:09	173		1165			SET	6000
AERO	ABSOLUTE	21	MAY	74	21:50:44	174		1165			SET	7165
NEXT AVAILABLE LOCATION=												8330

ASSEMBLER PROCEDURE TABLE EMPTY

COBOL PROCEDURE TABLE EMPTY

FORTRAN PROCEDURE TABLE EMPTY

ENTRY POINT TABLE

D NAME	LINK	D NAME	LINK	D NAME	LINK	D NAME	LINK	D NAME	LINK
ADDZOH	7	AETVAR	9	AGAIN	11	BEGIN	13	BHA091	15
BHA272	17	BHA341	19	BODE	23	CDABV	25	CLPOLE	27
COMPUT	29	CSMTRX	31	CSOLTF	33	DATA	164	DB	35
DECIDE	143	DELZOH	37	DET	2	DETCS	39	DETSD	151
ERROR	41	ESTMAT	147	FIXIT	43	FORM	45	FORHAINS	162
FRMTX	47	GENMTX	49	GETEST	51	GRAPHS	166	INITAL	53
INPEST	55	INPUTX	57	INPSYQ	59	INPRL	4	INPUT	61
INT1	63	INT2	65	KCALC	67	LIMIT	69	MLTZRO	71
MODZRO	73	MZTRAN	75	NICHOL	168	NONMTX	77	NYQIST	79
OUTPUT	170	PEVAL	149	PFE	81	PFEZRO	83	POINT	85
POLES	87	POLVAL	89	PREVAR	91	PRINTT	93	PUTOUT	95
RAMMTX	97	RESET	172	RLOCUS	99	RLPLOT	101	RLPRNT	155
ROOTER	103	PUTER	105	SAVE	107	SAVEST	141	SAVRUT	109
SCALE	159	SEPOPC	111	SOLVE	113	SRRL	153	SRRLPP	157
START	115	STNDYQ	117	STNRL	119	SUMMRY	121	SYSERQ	123
TEST	125	TRNSFR	127	VRYGEN	129	VRYRAW	145	WRITE	131
XCNG	133	ZEEPLS	135	ZEROS	137	ZTRAN	139		

```
$FOR,USR F,MAIN,F,MAIN
FOR,SE1X=05/23/74=08:25:40 (1,2)

    MAIN PROGRAM

  STORAGE USED: CODE(1) 000271 DATA(0) 000125 BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   KEEP2   000047
    0004   KEEP3   000102
    0005   KEEP4   000263
    0006   KEEP5   000074
    0007   KEEP6   000134
    0010   KEEP11  007723
    0011   KEEP16  000031
    0012   PLT     000012

  EXTERNAL REFERENCES (BLOCK, NAME)

    0013   LIMIT
    0014   RESET
    0015   INPUT
    0016   CSOLTF
    0017   CSMTRX
    0020   BHA341
    0021   BHAD91
    0022   ADDZOH
    0023   DELZOH
    0024   ZEEPLS
    0025   BHA272
    0026   SRRL
    0027   SRRLPP
    0030   NONMTX
    0031   ERROR
    0032   NINTRS
    0033   NRDUS
    0034   NIO2S
    0035   NRDUS
    0036   NIO3S
    0037   NSTOPS

  STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001  000006 10L      0001  000176 1000L    0001  000205 1010L    0001  000230 1020L    0001  000042 146G
    0001  000241 2000L    0001  000122 25L      0001  000036 30L      0001  000265 3000L    0001  000147 50L
    0000  000054 500F     0000  000002 600F     0000  000042 601F     0000  000055 602F     0000  000057 603F
    0006 L 000071 BOTH     0005  000212 BY       0007 R 000069 CARD     0005  000166 DD     0011 L 000022 DEBUG
    0003 D 000002 DHFILT   0012 R 000007 DIF1     0012 R 000010 DIF2     0004  000063 DP     0005  000004 GAIN
    0006 L 000073 GPRINT   0005  000151 GSYM     0003  000037 HACC     0003  000025 HAST     0003  000035 HATT
    0003 D 000000 HBLANK   0003  000004 HBLK     0003  000034 HCCW     0003  000033 HCW      0003  000032 HDEC
    0003   000027 HDOT     0003  000006 HESTI    0003  000043 HFGD     0003  000044 HFGN     0003  000040 HFPD
    0003   000045 HFPDD    0003  000042 HFPDN    0003  000041 HFPN     0003  000010 HGENE    0003  000031 HINC
    0003 R 000005 HKEY     0003  000007 HMATR    0003  000014 HNEW     0003  000046 HNOMI    0003  000015 HOYQU
    0003   000023 HO       0003  000026 HPLUS    0003  000036 HRATE    0003  000011 HRAW     0003  000012 HRETA
```

```
0003  000017 HROLL   0003  000016 HROOT   0003  000013 HSTAN   0003  000030 HSTAR   0003  000020 HSIC
0003  000021 HS2     0003  000022 HS4B    0003  000024 HX      0000 I 000000 I       0012  000011 ICK
0012 I 000004 ICT    0011  000010 IOPEN   0012 I 000003 ISW     0006 I 000067 ITHZT   0011  000011 JOPEN
0010  005670 KDS     0011  000001 KODE    0007  000110 LABEL   0011 L 000023 LFLT    0004 R 000051 MAX
0004 R 000037 MIN    0006 L 000072 MODIFY  0007 R 000062 NAME    0010 I 007722 NEIGZT  0004  000000 NFI
0005  000003 NGAIN   0005  000154 NGR     0012 L 000001 NICPLT   0010  003720 NIS     0011 L 000027 NOMNAL
0011 L 000030 NOTYET 0012 I 000002 NP      0005  000066 NPHASE   0011  000002 NRCLPL  0005  000153 NRLFR
0011  000003 NRPOLE  0011  000004 NRZERO   0010  007640 NSEIG   0011 I 000005 NXB     0011 I 000006 NXN
0011 I 000007 NXR    0000 I 000001 NXT     0006 I 000004 NZT     0004  000076 PB      0011  000014 PCPL
0004  000025 PCT     0011  000015 PEAC    0005  000000 PG      0005  000067 PHASE   0004  000075 PN
0012  000000 PNI     0011  000013 PNOM    0005  000001 PP      0005  000002 PPLT    0011  000016 PSLOSH
0005  000152 PSYM    0011  000012 PVAR    0004  000077 PI80    0005  000200 RX      0010 C 000000 SEIG
0011  000000 STAGE   0004 L 000101 STNDRD  0004  000013 STP     0004  000001 STR     0006  000000 SUPERK
0012 R 000006 S360   0006 R 000003 TD      0007  000000 TITLE   0007  000024 TITLE1  0007  000036 TITLE2
0007  000050 TITLE3  0012 R 000005 T360    0011 L 000017 YESMTX  0004 L 000100 YESNYQ  0011 L 000024 YESPCH
0011 L 000020 YESRAW 0005 L 000262 YESRL   0011 L 000025 YESRLP  0011 L 000026 YESSRL  0011 L 000021 YESSRP
0006 L 000070 YESZOH 0006 R 000002 ZM      0006 R 000001 ZT      0006 R 000005 ZTVAL
```

```
00100    1•      C                                                                    10    000000
00101    2•            COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,  20    000000
00101    3•      1           HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS4B,        30    000000
00101    4•      2           HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCC,          40    000000
00101    5•      3           HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,         50    000000
00101    6•      4           HNOMI                                                    60    000000
00103    7•            DOUBLE PRECISION HBLANK,DHFILT                                      000001
00104    8•            COMMON/KEEP3/ NFI,STR(10),STP(10),PCT(10),MIN(10),MAX(10),DP(10),  80    000001
00104    9•      1           PN,PB,PI80,YESNYQ,STNDRD                                 90    000001
00105   10•            REAL        MIN,MAX                                             100   000001
00106   11•            LOGICAL     YESNYQ,STNDRD                                       110   000001
00107   12•            COMMON/KEEP4/ PG,PP,PPLT,NGAIN,GAIN(50),NPHASE,PHASE(50),GSYM,  120   000001
00107   13•      1           PSYM,NRLFR,NGR(10),DD(10),RX(10),DY(4,10),YESRL          130   000001
00110   14•            LOGICAL     YESRL                                               140   000001
00111   15•            COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,  150   000001
00111   16•      1           MODIFY,GPRINT                                            160   000001
00112   17•            LOGICAL     YESZOH,BOTH,MODIFY,GPRINT                           180   000001
00113   18•            COMMON/KEEP6/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),  190   000001
00113   19•      1           CARD(20),LABEL(20)                                       200   000001
00114   20•            REAL        NAME                                                210   000001
00115   21•            COMMON/KEEP11/SEIG(1000),NIS(1000),KDS(1000),NSEIG(50),NEIGZT   220   000001
00115   22•            COMPLEX     SEIG                                                230   000001
00117   23•            COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,      240   000001
00117   24•      1           IOPEN,JOPEN,PVAR,PNOM,PCPL,PEAC,PSLOSH,                  250   000001
00117   25•      2           YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,           260   000001
00117   26•      3           YESSRL,NOMNAL,NOTYET                                     270   000001
00120   27•            LOGICAL     YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      280   000001
00120   28•      1           YESSRL,NOMNAL,NOTYET                                     290   000001
00121   29•            LOGICAL NICPLT                                                      000001
00122   30•            COMMON /PLT/ PNI,NICPLT,NP,ISW,ICT,T360,S360,DIF1,DIF2,ICK          000001
00122   31•      C                                                                    300   000001
00122   32•      C  INITIALIZE PROGRAM                                                310   000001
00123   33•      C                                                                    320   000001
00123   34•            CALL LIMIT                                                     340   000001
00124   35•            CALL RESET                                                     350   000003
00124   36•      C                                                                    360   000003
```

```
00124    37*     C                                                          370    000003
00124    38*     C  INPUT THE NEXT DATA CASE                                380    000003
00124    39*     C                                                          390    000003
00125    40*       10 CONTINUE                                              400    000006
00125    41*          WRITE(6,600)                                          410    000006
00130    42*       600 FORMAT('1',10(/),2(17X,94('*')/),14(17X,2('*'),90X,2('*')/),  420    000012
00133    43*          117X,2('*'),24X,'THE SAMPLED DATA STABILITY ANALYSIS PROGRAM',  430    000012
00133    44*          223X,2('*')/14(17X,2('*'),90X,2('*')/),2(17X,94('*')/))  440    000012
00131    45*          CALL INPUT($1000,$2000)                                      000012
00131    46*     C                                                          460    000012
00131    47*     C                                                          470    000012
00131    48*     C  CONSTRUCT THE CONTINUOUS SYSTEM OPEN-LOOP TRANSFER FUNCTION  480    000012
00131    49*     C                                                          490    000012
00132    50*          CALL CSOLTF($1000)                                           000016
00132    51*     C                                                          510    000016
00132    52*     C                                                          520    000016
00132    53*     C  LOOP ON THE SAMPLING RATE                               530    000016
00132    54*     C                                                          540    000016
00133    55*          NEIGZT = 0                                            550    000021
00134    56*          IF (.NOT. NICPLT) GO TO 30                                   000022
00136    57*          DIF1 = 0                                                     000024
00137    58*          DIF2 = 0.                                                    000025
00140    59*          ISW = 0                                                      000026
00141    60*          ICT = 0                                                      000027
00142    61*          T360 = -360.                                                 000030
00143    62*          S360 = -360.                                                 000032
00144    63*          NP = 1                                                       000033
00145    64*       30 DO 100 I=1,NZT                                               000036
00150    65*          ITHZT = I                                             570    000042
00151    66*          ZT = ZTVAL(I)                                         580    000044
00152    67*          ZM = 1.0 - TD/ZT                                      590    000046
00152    68*     C                                                          600    000046
00152    69*     C                                                          610    000046
00152    70*     C  DETERMINE REQUEST FOR THE ZERO ORDER HOLD DEVICE        620    000046
00152    71*     C                                                          630    000046
00153    72*          IF (.NOT.BOTH) GO TO 25                               640    000052
00153    73*     C                                                          650    000052
00153    74*     C                                                          660    000052
00153    75*     C  CONSTRUCT WITHOUT ZERO ORDER HOLD, THEN WITH ZERO ORDER HOLD  670    000052
00153    76*     C  PERFORM Z-R TRANSFORMATION THEN NYQUIST                 680    000052
00153    77*     C                                                          690    000052
00155    78*          CALL CSNTRX                                           700    000054
00156    79*          CALL BHA341($1000)                                           000056
00157    80*          IF (YESNYQ) CALL BHA091($1000)                               000061
00161    81*          YESZOH = .TRUE.                                       730    000066
00162    82*          BOTH = .FALSE.                                        740    000070
00163    83*          CALL ADDZOH($1000)                                           000071
00164    84*          CALL CSNTRX                                           760    000074
00165    85*          CALL BHA341($1000)                                           000076
00165    86*          IF (YESNYQ) CALL BHA091($1000)                               000101
00170    87*          YESZOH = .FALSE.                                      790    000106
00171    88*          BOTH = .TRUE.                                         800    000107
00172    89*          IF (I.LT.NZT) CALL DELZOH                             810    000111
00174    90*          GO TO 50                                              820    000120
00174    91*     C                                                          830    000120
00174    92*     C                                                          840    000120
00174    93*     C  CONSTRUCT EITHER WITH OR WITHOUT ZERO ORDER HOLD        850    000120
```

```
00174    95*    C
00175    96*       25 CONTINUE                                                    880  000122
00176    97*          IF (YESZOH .AND. I.EQ.1) CALL ANDZOH($1000)                      000122
00201    98*          CALL CSMTRX                                                 900  000134
00201    99*          CALL BHA391($1000)                                               000136
00203   100*          IF (YESDYO) CALL BHA091($1000)                                   000141
00203   101*    C                                                                 930  000141
00203   102*    C                                                                 940  000141
00203   103*    C  PRINT THE SAMPLED DATA SYSTEM OPEN LOOP POLES IN THE Z DOMAIN  950  000141
00203   104*    C                                                                 960  000141
00204   105*       50 CONTINUE                                                    970  000147
00205   106*          CALL ZEEPLS                                                 980  000147
00205   107*    C                                                                 990  000147
00205   108*    C                                                                1000  000147
00205   109*    C  PERFORM ROOT LOCUS ANALYSIS                                   1010  000147
00205   110*    C                                                                1020  000147
00205   111*          NOTYET = .TRUE.                                            1030  000150
00207   112*          IF (YESRL) CALL BHA272($1000)                                    000152
00207   113*    C                                                                1050  000152
00207   114*    C                                                                1060  000152
00207   115*    C  SAVE NOMINAL ROOTS FOR SAMPLE RATE                            1070  000152
00207   116*    C                                                                1080  000152
00211   117*          IF (YESSRL) CALL SRRL($1000)                                     000157
00211   118*    C                                                                1100  000157
00211   119*    C                                                                1110  000157
00211   120*    C  END OF ANALYSIS FOR THE SPECIFIED SAMPLING RATE               1120  000157
00211   121*    C                                                                1130  000157
00213   122*       100 CONTINUE                                                  1140  000166
00213   123*    C                                                                1150  000166
00213   124*    C                                                                1160  000166
00213   125*    C  GENERATE SAMPLE RATE ROOT LOCUS PRINT AND PLOT                1170  000166
00213   126*    C                                                                1180  000166
00215   127*          IF (YESSRL) CALL SRRLPP                                          000166
00215   128*    C                                                                1200  000166
00215   129*    C                                                                1210  000166
00215   130*    C  RETURN NOMINAL S - DOMAIN MATRIX                              1220  000166
00215   131*    C                                                                1230  000166
00217   132*          CALL NORMTX                                                1240  000172
00217   133*    C                                                                1250  000172
00217   134*    C                                                                1260  000172
00217   135*    C  GO TO THE START OF THE NEXT CASE                              1270  000172
00217   136*    C                                                                1280  000172
00220   137*          GO TO 10                                                   1290  000174
00220   138*    C                                                                1300  000174
00220   139*    C                                                                1310  000174
00220   140*    C  AN ERROR WAS ENCOUNTERED                                      1320  000174
00220   141*    C                                                                1330  000174
00221   142*       1000 CONTINUE                                                 1340  000176
00222   143*          CALL ERROR                                                 1350  000176
00222   144*    C                                                                1360  000176
00222   145*    C                                                                1370  000176
00222   146*    C SEARCH FOR THE KEY WORD AND RESTART                            1380  000176
00222   147*    C                                                                1390  000176
00223   148*          WRITE(6,601)                                               1400  000177
00225   149*       601 FORMAT(//////' SKIPPING DATA CARDS UNTIL KEY WORD IS FOUND')  1410  000205
00226   150*       1010 READ(5,500,END=2000) CARD                                1420  000205
```

```
00231     151*        500 FORMAT(20A4)                                                  143D    000215
00232    *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00233     152*        IF (CARD(1).NE.HKEY) GO TO 1020                                   144D    000215
00234     153*        WRITE(6,600)                                                      145D    000221
00236     154*        GO TO 10                                                          146D    000226
00237     155*     1020 CONTINUE                                                        147D    000230
00240     156*        WRITE(6,602) CARD                                                 148D    000230
00243     157*     602 FORMAT(20X,20A4)                                                 149D    000237
00244     158*        GO TO 1010                                                        150D    000237
00244     159*     C                                                                    151D    000237
00244     160*     C                                                                    152D    000237
00245     161*     C  PRINT THE NUMBER OF PLOT FRAMES                                   153D    000237
00246     162*     C                                                                    154D    000237
00246     163*     2000 CONTINUE                                                        155D    000241
00246     164*        NXN = 2*NXN                                                       156D    000241
00247     165*        NXT = NXN + NXB + NXR + 2                                         157D    000243
00251     166*        IF (NXT.LE.2) GO TO 3000                                          158D    000247
00257     167*        WRITE(6,603) NXN,NXB,NXR,NXT                                      159D    000253
00260     168*     603 FORMAT('1'///17X,'PLOTTING INFORMATION'///10X,I5,2X,'NYQUIST PLOTS  160D    000265
00260     169*       1'//10X,I5,2X,'BODE PLOTS'//10X,I5,2X,'ROOT LOCUS PLOTS'//10X,I5,    161D    000265
00260     170*       22X,'TOTAL FRAMES ON THE BENSON - LEHNER PLOT TAPE')                 162D    000265
00261     171*     C                                                                    164D    000265
00261     172*     C                                                                    165D    000265
00261     173*     C  END OF PROGRAM                                                    166D    000265
00261     174*     C                                                                    167D    000265
00261     175*     3000 CONTINUE                                                        168D    000265
00262     176*        STOP                                                              169D    000265
00263     177*        END                                                               170D    000270

   END OF COMPILATION:          1 DIAGNOSTICS*
```

```
?FOR,USW R=ADDZOH,P=ADDZOH
FOR 78UX-05/23/79-08:25:50 02.3)

    SUBROUTINE ADDZOH    ENTRY POINT 000064

    STORAGE USED: CODE(1) 000073; DATA(0) 000012; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   KEEP1  000026
    0004   KEEP7  000712
    0005   KEEP16 000031

    EXTERNAL REFERENCES (BLOCK, NAME)

    0006   NERR4$
    0007   NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000043 IOOOL     0001   000017 I15G      0005 L 000022 DEBUG     0004 R 000570 FDPOL     0004   000454 FNPOL
    0000 I 000000 I         0000   000002 INJP$     0005   000010 IOPEN     0005   000011 JOPEN     0005 I 000001 KODE
    0005 L 000023 LFLT      0003   000000 MAXIT     0003 I 000001 MXEIG     0003   000024 MXEIGT    0003   000023 MXEST
    0003   000002 MXERM     0003   000003 MXNBN     0003   000004 MXNCOF    0003   000005 MXNCT     0003   000025 MXNCV
    0003   000006 MXNE      0003   000007 MXNEQ     0003   000010 MXNFI     0003   000011 MXNG      0003   000012 MXNPH
    0003   000013 MXNPP     0003   000014 MXNQPT    0003   000015 MXNSM     0003   000016 MXNSP     0003   000017 MXNTM
    0003   000020 MXNV      0003   000021 MXNZT     0003   000022 MXPOLY    0000 I 000001 N         0004 I 000707 NDCOEF
    0004 I 000705 NDGD      0004   000704 NDGN      0004   000706 NNCOEF    0005 L 000027 NOMNAL    0005 L 000030 NOTYET
    0005   000002 NRCLPL    0005   000003 NRPOLE    0005   000004 NRZERO    0005   000005 NXB       0005   000006 NXN
    0005   000007 NXR       0004   000711 NZD       0004   000710 NZN       0005   000014 PCPL      0005   000015 PFAC
    0005   000013 PNOM      0005   000016 PSLOSH    0005   000012 PVAR      0004 R 000341 RID       0004   000113 RIN
    0004 R 000226 RRD       0004   000000 RRN       0005   000000 STAGE     0005 L 000017 YESMTX    0005 L 000024 YESPCH
    0005 L 000020 YESRAW    0005 L 000025 YESRLP    0005 L 000026 YESSRL    0005 L 000021 YESSRP

00101     1*          SUBROUTINE ADDZOH(*)                                                    264₀  000000
00102     2*          COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBN,MXNCOF,MXNCT,MXNE,MXNEQ,          265₀  000000
00103     3*         1 MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,                      266₀  000000
00103     4*         2 MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV                                  267₀  000000
00104     5*          COMMON/KEEP7/ RRN(75),RIN(75),RRD(75),RID(75),FNPOL(76),                268₀  000000
00104     6*         1 FDPOL(76),NDGN,NDGD,NNCOEF,NDCOEF,NZN,NZD                              269₀  000000
00105     7*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,              271₀  000000
00105     8*         1 IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                                272₀  000000
00105     9*         2 YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,                         273₀  000000
00105    10*         3 YESSRL,NOMNAL,NOTYET                                                   274₀  000000
00106    11*          LOGICAL YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,                  275₀  000000
00106    12*         1 YESSRL,NOMNAL,NOTYET                                                   276₀  000000
00107    13*     C                                                                            277₀  000000
00108    14*     C                                                                            278₀  000000
00109    15*     C  ADD THE ZERO ORDER HOLD DEVICE                                            279₀  000000
00109    16*     C                                                                            280₀  000000
```

```
00107     17*           IF (NDGD.GE.MXEIG) GO TO 1000                    291₀    000000
00111     18*           NDGD = NDGD + 1                                  282₀    000004
00112     19*           RRD(NDGD) = 0.0                                          000007
00113     20*           RID(NDGD) = 0.0                                          000011
00111     21*           DO 10 I=1,NDCOEF                                 285₀    000017
00117     22*           N = NDCOEF - I + 1                               286₀    000017
00121     23*        10 FDPOL(N+1) = FDPOL(N)                            287₀    000023
00122     24*           FDPOL(1) = 0.0                                           000030
00123     25*           NDCOEF = NDCOEF + 1                              289₀    000031
00124     26*           NZD = NZD + 1                                    290₀    000034
00125     27*           RETURN                                           291₀    000037
00126     28*     C                                                      292₀    000037
00125     29*     C                                                      293₀    000037
00126     30*     C  CAN NOT INCORPORATE THE ZERO-ORDER HOLD DEVICE      294₀    000037
00126     31*     C                                                      295₀    000037
00126     32*     1000 CONTINUE                                          296₀    000043
00127     33*           KODE = 2                                         297₀    000043
00130     34*           RETURN 1                                         298₀    000044
00131     35*           END                                              299₀    000072

      END OF COMPILATION:      NO DIAGNOSTICS.
```

```
;FOR,USW F,AFTVAR,F,AFTVAR
FOR.SEIX-05/23/74-08:25:55 (2,3)

     SUBROUTINE AFTVAR    ENTRY POINT 000022

     STORAGE USED: CODE(1) 000026; DATA(0) 000004; BLANK COMMON(2) 000000

     COMMON BLOCKS:

     0003  KEEP16 00003)

     EXTERNAL REFERENCES (BLOCK, NAME)

     0004  NERR4$
     0005  NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0003 L 000022 DEBUG    0003    000000 INJP$    0003    000010 IOPEN    0003    000011 JOPEN    0003 I 000001 KODE
     0003 L 000023 LFLT     0003 L  000027 NOMNAL   0003 L  000030 NOTYET   0003    000002 NRCLPL   0003    000003 NRPOLE
     0003   000004 NRZERO   0003    000005 NXB      0003    000006 NXN      0003    000007 NXR      0003    000014 PCPL
     0003   000015 PFAC     0003    000013 PNOM     0003    000016 PSLOSH   0003    000012 PVAR     0003    000000 STAGE
     0003 L 000017 YESMTX   0003 L  000024 YESPCH   0003 L  000020 YESRAW   0003 L  000025 YESRLP   0003 L  000026 YESSRL
     0003 L 000021 YESSRP

1-9
000101     1*          SUBROUTINE AFTVAR(*)                                          171₀  000000
000103     2*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,    178₀  000000
000103     3*        1           IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,             179₀  000000
000103     4*        2           YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      180₀  000000
000103     5*        3           YESSRL,NOMNAL,NOTYET                                181₀  000000
000104     6*          LOGICAL     YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,    182₀  000000
000104     7*        1           YESSRL,NOMNAL,NOTYET                                183₀  000000
000104     8*    C                                                                   194₀  000000
000104     9*    C                                                                   195₀  000000
000104     10*   C  RESTORE NOMINAL RAW DATA PARAMETER VALUES                        196₀  000000
000104     11*   C                                                                   197₀  000000
000105     12*         IF (.NOT.YESRAW) RETURN                                       198₀  000000
000105     13*   C                                                                         000000
000106     14*   C                                                                         000000
000106     15*   C  ONLY A GENERAL MATRIX DEFINITION IS PERMITTED IN THIS VERSION          000000
000106     16*   C                                                                         000000
000107     17*         KODE = 101                                                          000004
000110     18*         RETURN 1                                                            000006
000111     19*         END                                                           263₀  000025

     END OF COMPILATION:     NO  DIAGNOSTICS.
```

```
@FOR,USR F,AGAIN,F,AGAIN
FOR SEIX-05/23/74-08:26:00 (2,3)

      SUBROUTINE AGAIN      ENTRY POINT 000244

      STORAGE USED: CODE(I) 000253; DATA(D) 000063; BLANK COMMON(2) 000030

      COMMON BLOCKS:

      0003  KEEP1  000026
      0004  KEEP14 000031
      0005  KEEP16 000031
      0006  CRUD3  016115

      EXTERNAL REFERENCES (BLOCK, NAME)

      0007  PEVAL
      0010  DET
      0011  COABV
      0012  SCALE
      0013  COV$
      0014  MADU$
      0015  NIO2$
      0016  NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001  000024 1L     0001  000167 2L     0001  000174 3L     0000  000004 600E   0006 L 016113 AUTO
0006 C 016072 B     0011 R 000000 COABV 0006 L 016112 CONJ   0006 C 000000 CU     0005 L 000022 DEBUG
0004  000025 DEG    0006 L 016110 DONE  0004  000012 D0     0004  000013 D1     0004  000014 D2
0004  000015 D3     0004  000016 D4     0000 C 000000 SP     0004 C 000010 FIFTEN  0004  000017 FIFTY
0004 C 000006 FOUR  0006 C 016040 FPR0  0006 C 016042 FPR1   0006 C 016044 FPR2   0006 C 016046 FR0
0006 C 016050 FR1   0004 C 016052 FR2   0000 C 000000 HALF   0000  000043 INJP$  0005  000010 IOPEN
0005  000011 JOPEN  0005  000001 KODE  0005 L 000023 LFLT   0003  000000 MAXIT  0003  000001 MXEIG
0003  000024 MXEIGT 0003  000023 MXEST 0003  000002 MXFRM   0003  000003 MXNBM  0003  000004 MXNCOF
0003  000005 MXNCT  0003  000025 MXNCV 0003  000006 MXNE   0003  000007 MXNEQ  0003  000010 MXNFI
0003  000011 MXNG   0003  000012 MXNPH 0003  000013 MXNPP  0003  000014 MXNQPT 0003  000015 MXNSM
0003  000016 MXNSP  0003  000017 MXNTM 0003  000020 MXNV   0003  000021 MXNZT  0003  000022 MXPOLY
0006 I 016102 NCT   0006  016107 NEST  0006 I 016076 NFP0  0006 I 016077 NFP1  0006 I 016100 NFP2
0000 I 000003 NF2   0006 I 016101 NITER 0006  016104 NKODE 0005 L 000027 NOMNAL 0005 L 000030 NOTYET
0000 I 000002 NP2   0005  000002 NRCLPL 0006  016103 NREG   0005  000003 NRPOLE 0005  000004 NRZERO
0004  016105 NSTART 0006  016106 NTIME 0005  000005 NXA    0005  000006 NXN    0005  000007 NXR
0004  000026 N1    0004  000027 N2     0004  000030 N3     0004 C 000002 ONE    0005  000014 PCPL
0005  000015 PFAC  0004  000020 PI     0004  000021 PI2    0005  000013 PNOM   0006 C 016054 PR0
0006 C 016056 PR1   0006 C 016060 PR2   0005  000016 PSLOSH 0005  000012 PVAR   0004  000024 RADDEG
0006 L 016114 REGSEL 0006 L 016111 RESTRT 0004  000023 RP1   0006 C 016062 R0     0006 C 016064 R1
0006 C 016066 R2    0006 C 016070 R3    0004  000022 SMALL  0005  000000 STAGE  0004 C 000004 TWO
0006 C 016074 U     0005 L 000017 YESMIX 0005 L 000024 YESPCH 0005 L 000020 YESRAW 0005 L 000025 YESRLP
0005 L 000026 YESSRL 0005 L 000021 YESSRP
```

```
00101    1*            SUBROUTINE AGAIN                                                    300₀  000000
00102    2*            COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,     301₀  000000
00103    3*           1              MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,     302₀  000000
00104    4*           2              MXNV,MXNZI,MXPOLY,MXEST,MXEIGT,MXNCV                 303₀  000000
00104    5*            COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,     304₀  000000
00104    6*           1              PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                     305₀  000000
00105    7*            COMPLEX        HALF,ONE,TWO,FOUR,FIFTEN                                 000000
00105    8*            COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,         308₀  000000
00105    9*           1              IOPEN,JOPEN,PVAR,PNO1,PCPL,PFAC,PSLOSH,               309₀  000000
00105   10*           2              YESMTX,YESRAW,YESSRP,DEBUG,LELT,YESPCH,YESRLP,        310₀  000000
00105   11*           3              YESSRL,NOMNAL,NOTYET                                  311₀  000000
00107   12*            LOGICAL        YESMTX,YESRAW,YESSRP,DEBUG,LELT,YESPCH,YESRLP,        312₀  000000
00107   13*           1              YESSRL,NOMNAL,NOTYET                                  313₀  000000
00111   14*            COMMON/CRUD3/ CU(60,60),FPRO,FPR1,FPR2,FRO,FR1,FR2,PRO,PR1,PR2,     314₀  000000
00111   15*           1              RO,R1,R2,R3,B,U,NFPO,NFP1,NFP2,NITER,NCT,NREG,NKODE,  315₀  000000
00111   16*           2              NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL        316₀  000000
00111   17*            COMPLEX        CU,FPRO,FPR1,FPR2,FRO,FR1,FR2,PRO,PR1,PR2,               000000
00111   18*           1              RO,R1,R2,R3,B,U                                       318₀  000000
00112   19*            LOGICAL        DONE,RESTRT,CONJ,AUTO,REGSEL                          319₀  000000
00113   20*            COMPLEX        EP                                                       000000
00113   21*       C                                                                         321₀  000000
00113   22*       C                                                                         322₀  000000
00113   23*       C*****    PERFORM ITERATION                                               323₀  000000
00113   24*       C*****    SAVE PREVIOUS RESULTS                                           324₀  000000
00113   25*       C                                                                         325₀  000000
00114   26*             EP = (1.0E-8,0.0)                                                      000000
00115   27*             RESTRT=.FALSE.                                                    327₀  000001
00116   28*             RO=R1                                                             328₀  000002
00117   29*             R1=R2                                                             329₀  000004
00120   30*             R2=R3                                                             330₀  000006
00121   31*             FPRO=FPR1                                                         331₀  000010
00122   32*             NFPO=NFP1                                                         332₀  000012
00123   33*             FPR1=FPR2                                                         333₀  000014
00124   34*             NFP1=NFP2                                                         334₀  000016
00125   35*             NITER=NITER+1                                                     335₀  000020
00125   36*       C                                                                         336₀  000020
00125   37*       C                                                                         337₀  000020
00125   38*       C*****    DETERMINE FUNCTIONAL EVALUATION FOR NEW ITERANT                 338₀  000020
00125   39*       C                                                                         339₀  000020
00125   40*         1 CONTINUE                                                            340₀  000024
00127   41*             CALL PEVAL(R2,PR2,NP2)                                            341₀  000024
00130   42*             CALL DET(R2,FR2,NF2)                                              342₀  000030
00131   43*             NFP2=NF2-NP2                                                      343₀  000035
00132   44*             FPR2=FR2/PR2                                                      344₀  000040
00133  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00133   45*             IF (CDABV(FPR2) .EQ. 0.0) NFP2=0                                      000045
00135   46*             CALL SCALE(FPRO,NFPO,FPR1,NFP1,FPR2,NFP2)                         346₀  000053
00136   47*             IF(DEBUG) WRITE(6,600) R2,FPR2,NFP2,FR2,NF2,PR2,NP2,NCT           347₀  000063
00151   48*         600 FORMAT('0 AGAIN',6X,'R2  ',2E14.5,5X,'FPR2  ',2E14.5,5X,'NFP2',       000102
00151   49*            1I4/13X,'FR2  ',2E14.5,5X,'NF2',I4,5X,'PR2  ',2E14.5,5X,'NP2',I4)      000102
00151   50*            2 13X,'NCT',I4)                                                        000102
00152   51*             IF (CDABV(FPR2/FPR1) .LE. 1.0) GO TO 3                            352₀  000102
00152   52*       C                                                                         353₀  000102
00152   53*       C                                                                         354₀  000102
00152   54*       C*****    FUNCTIONAL VALUE INCREASED TOO MUCH, TRY AGAIN                  355₀  000102
00152   55*       C                                                                         356₀  000120
00154   56*             NCT=NCT+1
```

```
00153     57*           IF(NCT.EQ.1) RETURN                                  357D    000123
00157     58*           IF (NCT.GT.MXNCT) GO TO 2                            358D    000132
00161     59*           R2=HALF*R2                                           359D    000137
00162 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00162     60*           IF (CDABV(R2) .EQ. 0.0) R2=EP                                000156
00163     61*           GO TO 1                                              361D    000165
00164     62*     C                                                          362D    000165
00164     63*     C                                                          363D    000165
00164     64*     C*****     SET RESTART BECAUSE OF SUCCESSIVE FUNCTION VALUE INCREASES  364D  000165
00164     65*     C                                                          365D    000165
00165     66*           2 CONTINUE                                           366D    000167
00165     67*           RESTRT=.TRUE.                                        367D    000167
00167     68*           RETURN                                               368D    000170
00167     69*     C                                                          369D    000170
00167     70*     C                                                          370D    000170
00167     71*     C*****     TEST FOR RAPID DECLINE OF FUNCTIONAL VALUE       371D    000170
00167     72*     C                                                          372D    000170
00170     73*           3 CONTINUE                                           373D    000174
00171     74*           NCT=0                                                374D    000174
00172 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00172     75*           IF (CDABV(FPR2) .EQ. 0.0) RETURN                             000174
00174     76*           IF (CDABV(FPR1/FPR2) .LE. 1.0E-5) RETURN                     000204
00174     77*     C                                                          377D    000204
00174     78*     C                                                          378D    000204
00174     79*     C*****     TOO RAPID DECLINE IN FUNCTIONAL VALUE            379D    000204
00174     80*     C                                                          380D    000204
00175     81*           RESTRT=.TRUE.                                        381D    000225
00177     82*           RETURN                                               382D    000227
00200     83*           END                                                  383D    000252

      END OF COMPILATION:     3 DIAGNOSTICS.
```

```
$FOR,USW F,BEGIN,F,BEGIN
FOR.3E1X-05/23/74-08:26:10 (2,3)

     SUBROUTINE BEGIN      ENTRY POINT 000067

     STORAGE USED; CODE(1) 000074; DATA(0) 000013; BLANK COMMON(2) 000000

     COMMON BLOCKS;

     0003   KEEP2  000047
     0004   KEEP3  000102
     0005   KEEPS  000074
     0006   CRU03  011650

     EXTERNAL REFERENCES (BLOCK, NAME)

     0007   NERR35

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0001   000042 IL        0001   000045 2L        0006   000070 AOIR      0006   000006 AFRQ      0006   011616 AML
     0006   011617 AMP       0006   000152 APHA      0004 R 011620 BIG       0005 L 000071 BOTH      0006 L 011642 DECR
     0006 R 011621 DF        0003 D 000002 DHFILT     0004 R 000063 DP        0006 L 011643 DPI       0006 L 011645 ERAM
     0006 L 011647 ERGP      0006 L 011644 ERR        0006 L 011646 ERPH      0006   011622 FL        0006 R 011623 FR
     0006 C 000000 GAINS     0005 L 000073 GPRINT     0003   000037 HACC      0003   000025 HAST      0003   000035 HATT
     0003 D 000000 HBLANK    0003 R 000004 HBLK       0003   000034 HCCW      0003   000033 HCW       0003   000032 HDEC
     0003   000027 HDOT      0003   000006 HESTI      0003   000043 HFGD      0003   000044 HFGN      0003   000040 HFPD
     0003   000045 HFPDD     0003   000042 HFPDN      0003   000041 HFPN      0003   000010 HGENE     0003   000031 HINC
     0003   000005 HKEY      0003   000007 HMATR      0003   000014 HNEW      0003   000046 HNOMI     0003   000015 HNYQU
     0003   000023 HQ        0003   000026 HPLUS      0003   000036 HRATE     0003   000011 HRAW      0003   000012 HRETA
     0003   000017 HROLL     0003   000016 HROOT      0003   000013 HSTAN     0003   000030 HSTAR     0003   000020 HSIC
     0003   000021 HS2       0003   000022 HS4B       0003   000024 HX        0006 R 011624 IM        0000   000002 INJPS
     0006 I 011634 INT       0005   000067 ITHZT      0006 I 011641 LHX       0006   011635 LRPR      0004 R 000051 MAX
     0004 R 000037 MIN       0005 L 000072 MODIFY     0006 I 011636 MPPP      0006   011637 NEXT      0004   000000 NFI
     0006   000003 NGNPK     0006 I 011640 NPPP       0006   000004 NP180     0006   000005 NYQPTS    0005   000004 NZT
     0006   000002 NIAMP     0006   000316 PAMP       0004   000076 PD        0004 R 000025 PCT       0006   000400 PDIR
     0006 R 011625 PER       0006   000234 PFRQ       0006   011626 PHA       0006   000626 PHAMP     0006   000710 PHDIR
     0006   000544 PHFRQ     0006   011627 PHL        0004   000075 PN        0006   000462 PPHA      0004   000077 P180
     0006   011630 RE        0006   003726 SAVAMP     0006   000772 SAVFRQ    0006   006662 SAVPHA    0006 R 011631 SMA
     0006 R 011632 STA       0004 L 000101 STNDRD     0004 R 011633 STO       0004 R 000013 STP       0004 R 000001 STR
     0005   000001 SUPERK    0005   000003 TD         0004 L 000100 YESNYQ     0005 L 000070 YESZOH     0005   000002 ZM
     0005 R 000001 ZT        0005   000005 ZTVAL

00101    1*          SUBROUTINE BEGIN                                                    3840  000000
00101    2*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,       3850  000000
00103    3*     1    HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS4B,                   3860  000000
00103    4*     2    HQ,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,                     3870  000000
00103    5*     3    HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,                    3880  000000
00103    6*     4    HNOMI                                                               3890  000000
00104    7*          DOUBLE PRECISION HBLANK,DHFILT                                            000000
```

```
00105     8*          COMMON/KEEP3/ NFI,STR(10),STP(10),PCT(10),MIN(10),MAX(10),DP(10),    391D  000000
00106     9*         1            PN,PR,P180,YESNYQ,STNDRD                                 392D  000000
00107     10*         REAL         MIN,MAX                                                  393D  000000
00108     11*         LOGICAL      YESNYQ,STNDRD                                            394D  000000
00109     12*         COMMON/KEEP5/ SUPERK,ZT,ZM,YD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,        395D  000000
00110     13*        1            MODIFY,GPRINT                                            396D  000000
00111     14*         LOGICAL      YESZOH,BOTH,MODIFY,GPRINT                                398D  000000
00112     15*         COMMON/GRUB3/ GAINS,NIAMP,NGNPK,NP180,NYQPTS,AFRQ(50),ADIR(50),       399D  000000
00113     16*        1            APHA(50),PFRQ(50),PAMP(50),PDIR(50),PPHA(50),            400D  000000
00114     17*        2            PHFRQ(50),PHAMP(50),PHDIR(50),SAVFRQ(1500),              401D  000000
00115     18*        3            SAVAMP(1500),SAVPHA(1500),AML,AMP,BIG,DF,FL,FR,IM,       402D  000000
00116     19*        4            PER,PHA,PHL,RE,SHA,STA,STO,                              403D  000000
00117     20*        5            INT,LRPR,MPPP,NEXT,NPPP,LMX,                             404D  000000
00118     21*        6            DECR,DPI,ERP,ERAM,ERPH,ERGP                              405D  000000
00119     22*         COMPLEX      GAINS                                                    406D  000000
00120     23*         REAL         IN                                                       407D  000000
00121     24*         LOGICAL      DECR,DPI,ERP,ERAM,ERPH,ERGP                              408D  000000
00122     25*    C                                                                         407D  000000
00123     26*    C  PROGRAM CODING                                                         410D  000000
00124     27*    C                                                                         411D  000000
00125     28*         LMX = 0                                                              412D  000000
00126     29*         DECR = .FALSE.                                                       413D  000001
00127     30*         LRPR = 1                                                             414D  000002
00128     31*         MPPP = 16                                                            415D  000004
00129     32*         MPPP = NPPP                                                          416D  000006
00130     33*         DPI = .FALSE.                                                        417D  000007
00131   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00132     34*         IF (DP(INT).NE.HBLK) DPI=.TRUE.                                      418D  000010
00133     35*         STA = STR(INT)                                                       419D  000016
00134     36*         STO = STP(INT)                                                       420D  000020
00135     37*         PER = PCT(INT)/100.                                                  421D  000022
00136     38*         SHA = MIN(INT)                                                       422D  000025
00137     39*         BIG = MAX(INT)                                                       423D  000027
00138     40*         FR = STA                                                             424D  000031
00139     41*         IF(STO.LT.STA) GO TO 1                                               425D  000032
00140     42*         DF = FR*PER                                                          426D  000036
00141     43*         GO TO 2                                                              427D  000040
00142     44*        1 DF = -STA*PER                                                       428D  000042
00143     45*        2 CONTINUE                                                            429D  000045
00144     46*    C                                                                         430D  000045
00145     47*    C  GO TO NEXT INCREMENTED FREQUENCY TO AVOID UNITY PRODUCT                431D  000045
00146     48*    C  OF SAMPLING PERIOD AND SYSTEM FREQUENCY                                432D  000045
00147     49*    C                                                                         433D  000045
00148   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00149     50*         IF(FR*ZT.EQ.1.0) FR = FR + DF                                        434D  000045
00150     51*         RETURN                                                               435D  000054
00151     52*         END                                                                  436D  000073

      END OF COMPILATION:      2 DIAGNOSTICS.
```

```
$FOR,US F,BHA091,F,BHA091
FOR SEIX=05/23/74=08:26:17 (2.3)

      SUBROUTINE BHA091    ENTRY POINT 000071

      STORAGE USED: CODE(1) 000077; DATA(0) 000007; BLANK COMMON(2) 000000

      COMMON BLOCKS:

       0003   KEEP3  000102
       0004   CRUD3  011650

      EXTERNAL REFERENCES (BLOCK, NAME)

       0005   INITAL
       0006   WRITE
       0007   BEGIN
       0010   TRNSFR
       0011   TEST
       0012   OUTPUT
       0013   POINT
       0014   SUMRY
       0015   GRAPHS
       0016   NERR2$
       0017   NERR4$
       0020   NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0001  000053 I0001L    0001  000014 I1L      0001  000011 I1SG     0001  000027 I2L      0001  000042 I20L
       0004  000070 ADIR      0004  000006 AFRQ     0004  011616 AML     0004  011617 AMP      0004  000152 APHA
       0004  011620 BIG       0004 L 011642 DECR     0004  011621 DF      0003  000063 DP       0004 L 011643 DPI
       0004 L 011645 ERAM      0004 L 011647 ERGP     0004 L 011644 ERP     0004 L 011646 ERPH     0004  011622 FL
       0004  011623 FR        0004 C 000000 GAINS    0004 R 011624 IN      0000  000000 INJP$     0004 I 011634 INT
       0004  011641 LMX       0004  011635 LRPR     0003 R 000051 MAX     0003 R 000037 MIN      0004  011636 MPPP
       0004 I 011637 NEXT      0003 I 000000 NF1      0004  000003 NGUPK    0004  011640 NPPP     0004  000004 NP180
       0004  000005 NYQPTS    0004  000002 NIAMP    0004  000316 PAMP    0003  000076 PB       0003  000025 PCT
       0004  000400 PDIR      0004  011625 PER      0004  000234 PFRQ     0004  011626 PHA      0004  000626 PHAMP
       0004  000710 PHDIR     0004  000544 PHFRQ    0004  011627 PHL     0003  000075 PN       0004  000462 PPHA
       0003  000077 P180      0004  011630 RE       0004  003726 SAVAMP   0004  000772 SAVFRQ   0004  006662 SAVPHA
       0004  011631 SMA       0004  011632 STA      0003 L 000101 STNDRD   0004  011633 STO      0003  000013 STP
       0003  000001 STR       0003 L 000100 YESNYQ

00101    1*          SUBROUTINE BHA091(*)                                                    4370  000000
00103    2*          COMMON/KEEP3/ NF1,STR(10),STP(10),PCT(10),MIN(10),MAX(10),DP(10),       4380  000000
00103    3*         1            PN,PB,P180,YESNYQ,STNDRD                                  4390  000000
00104    4*          REAL         MIN,MAX                                                   4400  000000
00105    5*          LOGICAL      YESNYQ,STNDRD                                             4410  000000
00106    6*          COMMON/CRUD3/ GAINS,NIAMP,NGUPK,NP180,NYQPTS,AFRQ(50),ADIR(50),        4420  000000
00106    7*         1            APHA(50),PFRQ(50),PAMP(50),PDIR(50),PPHA(50),             4430  000000
```

```
00105     8*          2       PHFRQ(50),PHAMP(50),PHDIR(50),SAVFRQ(1500),
00106     9*          3       SAVAMP(1500),SAVPHA(1500),AML,AMP,BIG,DF,FL,FR,IM,
00106    10*          4       PER,PHA,PHL,RE,SMA,STA,STO,
00106    11*          5       INT,LRPR,MRPP,NEXT,NPPP,LMX,
00106    12*          6       DECR,DPI,ERP,ERAM,ERPH,ERGP
00107    13*            COMPLEX GAINS
00111    14*            REAL    IM
00111    15*            LOGICAL DECR,DPI,ERP,ERAM,ERPH,ERGP
00111    16*      C
00111    17*      C
00111    18*      C  PERFORM SAMPLED DATA NYQUIST ANALYSIS
00111    19*      C
00112    20*            CALL INITAL(51000)
00113    21*            CALL WRITE
00114    22*            DO 20 INT=1,NFI
00117    23*            CALL BEGIN
00120    24*         11 CALL TRNSFR
00121    25*            CALL TEST
00122    26*            GO TO (11,12), NEXT
00123    27*         12 CALL OUTPUT
00124    28*            CALL POINT
00125    29*            GO TO (11,20), NEXT
00126    30*         20 CONTINUE
00131    31*            CALL SUMMRY
00131    32*            CALL GRAPHS
00132    33*            RETURN
00132    34*      C
00132    35*      C
00132    36*      C  ERROR IN NYQUIST INITIALIZATION
00132    37*      C
00133    38*         1000 CONTINUE
00134    39*            RETURN 1
00135    40*            END
```

```
4440  000000
4450  000000
4460  000000
4470  000000
4480  000000
4490  000000
4500  000000
4510  000000
4520  000000
4530  000000
4540  000000
4550  000000
      000000
4570  000002
4580  000004
4590  000011
4600  000014
4610  000015
4620  000017
4630  000027
4640  000030
4650  000032
4660  000043
4670  000043
4680  000045
4690  000047
4700  000047
4710  000047
4720  000047
4730  000047
4740  000053
4750  000053
4760  000076
```

```
 END OF COMPILATION:  NO DIAGNOSTICS.
```

```
?FOR,US F.BHA272,F.BHA272
FOR SEIX-05/23/74-08:26:26 (3.4)

SUBROUTINE BHA272     ENTRY POINT 000311

STORAGE USED: CODE(1) 000325; DATA(0) 000037; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003   KEEP1   000026
0004   KEEP2   000047
0005   KEEP4   000263
0006   KEEP5   000074
0007   KEEP9   000705
0010   KEEP10  021420
0011   KEEP15  000260
0012   KEEP16  000031
0013   KEEP19  000005
0014   KEEP21  001133
0015   CRUD2   003737
0016   CRUD3   016115

EXTERNAL REFERENCES (BLOCK, NAME)

0017   SAVEST
0020   GETEST
0021   RUTER
0022   SAVRUT
0023   RLPLOT
0024   NERR45
0025   NERR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001   000251 100#L     0001   000052 156#G     0001   000164 200L      0001   000260 2000L     0001   000137 204#G
0001   000206 234#G     0001   000157 25L       0001   000162 26L       0001   000235 400L      0016 L 016113 AUTO
0016 C 016072 B         0006 E 000071 BOTH      0005   000212 BY        0016 L 016112 CONJ      0011   000116 CPSLBL
0016 C 000000 CU        0011   000024 DA        0011   000130 DCBLBL    0005   000166 DD        0012 L 000022 DEBUG
0011   000131 DEGLBL    0015   003734 DELTA     0004 D 000002 DHFILT    0016 L 016110 DONE      0007 C 000226 EA
0015   001754 EIP       0015   000004 ERP       0010   016664 EV        0011   000146 FMAT      0011   000132 FMT
0016 C 016040 FPRO      0016 C 016042 FPR1      0016 C 016044 FPR2      0016 C 016046 FRO       0016 C 016050 FR1
0016 C 016052 FR2       0011   000162 F6        0011   000163 F7        0005 R 000004 GAIN      0015 R 000002 GAINV
0006 L 000073 GPRINT    0005   000151 GSYM      0004   000037 HACC      0004   000025 HAST      0004   000035 HATT
0004 D 000000 HBLANK    0004 R 000004 HBLK      0004   000034 HCCW      0004   000033 HCW       0004   000032 HDEC
0004   000027 HDOT      0004   000006 HESTI     0004   000043 HFGD      0004   000044 HFGN      0004   000040 HFPD
0004   000045 HFPOD     0004   000042 HFPON     0004   000041 HFPN      0004   000010 HGENE     0004   000031 HINC
0004   000005 HKEY      0004   000007 HMATR     0004   000014 HNEW      0004   000046 HNOMI     0004   000015 HNYQU
0004   000023 HO        0004   000026 HPLUS     0004   000036 HRATE     0004   000011 HRAW      0004   000012 HRETA
0004   000017 HROLL     0004   000016 HROOT     0004   000013 HSTAN     0004   000030 HSTAR     0004   000020 HSIC
0004   000021 HS2       0004   000022 HS#B      0004   000024 Hx        0000 I 000003 I         0011   000100 IDB
0011   000107 IDG       0000   000020 INJP#     0012   000010 IOPEN     0010   000004 IR        0000 I 000002 ITH
0006   000067 ITH#T     0010   001754 JC        0012   000011 JOPEN     0007 I 000572 KD        0012 I 000001 KODE
0011   000164 LABEL1    0011   000210 LABEL2    0011   000234 LABEL3    0012 L 000023 LFLT      0013 L 000000 LGAIN
```

```
0010  000674 LL          0014 I 000704 LOCD2          0014  000703 LOCNZ          0010  007644 LOCPOL        0018 L 000001 LPHASE
0013  000002 LPOLES       0013 L 000003 LSORL          0013 L 000004 LZEROS        0003  000000 MAXIT         0004 L 000072 MDIFF
0003  000001 MXEIG        0003  000024 MXEIGT         0003 I 000023 MXEST         0003  000002 MXFRM         0003  000003 MXNBM
0003  000004* MXNCOF      0003  000005 MXNCT          0003  000025 MXNCV          0003  000006 MXNE          0003  000007 MXNEQ
0003  000010 MXNFI        0003  000011 MXNG           0003  000012 MXNPH          0003 I 000013 MXNPP         0003  000014 MXNQPT
0003  000015 MXNSM        0003  000016 MXNSP          0003  000017 MXNTM         0003  000020 MXNV          0003  000021 MXNZT
0003  000022 MXPOLY       0007 I 000456 NA            0016  000003 NCOF           0016  016102 NCT           0010  003724 ND
0014  001020 NDC2         0010  000000 NDEG           0010  000002 NE             0007 I 000454 NEIG          0010  000001 NEQ
0016  016107 NEST         0016  016076 NFP0           0016  016077 NFP1           0016  016100 NFP2          0005 I 000003 NGAIN
0005  000154 NGR          0015  003735 NGSYM          0007  000457 NI             0016  016101 NITER         0016  016104 NKODE
0014  000705 NNCZ         0012 L 000027 NOMNAL         0012 L 000030 NOTYET        0015 I 003724 NPG           0005 I 000066 NPHASE
0015  003725 NPP          0015  003736 NPSYM          0007 I 000456 NQ            0012  000002 NRCLPL        0016  016103 NREG
0005  000153 NPLFR        0000 I 000000 NRP           0012  000003 NRPOLE         0000 I 000001 NRZ           0012  000004 NRZERO
0015  003726 NSHIFT       0016  016105 NSTART         0014  016106 NTIME          0014 I 000702 NUMZ          0012  000005 NXB
0012  000006 NXN          0012  000007 NXR            0004  000004 NZT            0012  000014 PCPL          0014  000341 PDZ
0012  000015 PEAC         0005 R 000000 PG            0005 R 000067 PHASE         0015 C 000000 PHAVAR        0012  000013 PNOM
0014  000000 PNZ          0005 R 000001 PP            0005 R 000002 PPLT          0016 C 016054 PRD           0016 C 016056 PRI
0016 C 016060 PRZ         0012  000016 PSLOSH         0005  000152 PSYM          0012  000012 PVAR          0011  000034 P36
0011 C 000000 REGION      0016 L 016114 REGSEL        0016 L 016111 RESTRT        0011  000030 RID,          0007 C 000000 ROOT
0005  000200 RX           0016 C 016062 RO            0016 C 016064 R1            0016 C 016066 R2            0016 C 016070 R3
0015 R 000003 SHIFT       0011  000014 SPACE         0012  000000 STAGE          0006  000000 SUPERK         0006  000003 TO
0016 C 016074 U           0011  000010 WIDTH         0015  003727 XR             0015  003730 YB            0012 L 000017 YESMTX
0012 L 000024 YESPCH      0012 L 000020 YESRAW        0005 L 000262 YESRL         0012 L 000025 YESRLP        0012 L 000026 YESSRL
0012 L 000021 YESSRP      0006 L 000070 YESZOH        0011  000020 YINC           0006  000002 ZH            0006  000001 ZT
0006  000005 ZTVAL
```

```
00101   1*        SUBROUTINE BHA272(*)                                                    477₀  000000
00103   2*          COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,       478₀  000000
00103   3*        1           MXNF1,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,          479₀  000000
00103   4*        2           MXNV,MXNZT,MAPOLY,MXEST,MXEIGT,MXNCV                      480₀  000000
00104   5*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,       481₀  000000
00104   6*        1           HRETA,HSTAN,HNEW,HNYOU,HROOT,HROLL,HS1C,HS2,HS4B,          482₀  000000
00104   7*        2           HD,HX,HAST,HPLUS,HDOT,HSTAR,HIEC,HDEC,HCW,HCC*,            483₀  000000
00104   8*        3           HATT,HRATE,HACC,HFPD,HFPN,HFPDC,HFGD,HFGN,HFPDD,           484₀  000000
00105   9*        4           HMONI                                                      485₀  000000
00105  10*          DOUBLE PRECISION HBLANK,DHFILT                                            000000
00106  11*          COMMON/KEEP3/ PG,PP,PPLT,NGAIN,GAIN(50),NPHASE,PHASE(50),GSYM,       487₀  000000
00106  12*        1           PSYM,NRLFR,NGR(10),DD(10),RX(10),BY(4,10),YESRL            488₀  000000
00107  13*          LOGICAL     YESRL                                                    489₀  000000
00110  14*          COMMON/KEEP5/ SUPERK,ZT,ZH,TO,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,        15₀  000000
00110  15*        1           MODIFY,GPRINT                                              160  000000
00111  16*          LOGICAL     YESZOH,BOTH,MODIFY,GPRINT                                18₀  000000
00112  17*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)             4900  000000
00113  18*          COMPLEX     ROOT,EA                                                      000000
00114  19*          COMMON/KEEP10/ DEG,NEQ,NE,NCOF,IR(1000),JC(1000),ND(1000),         493₀  000000
00114  20*        1           LL(1000),LOCPOL(60,60),EV(150)                             494₀  000000
00115  21*          COMMON/KEEP15/ REGION(4),WIDTH(4),SPACE(4),YINC(4),                      000000
00115  22*        1           DA(4),RID(4),P3A(36),IRS(7),IPS(7),CPSLBL(10),             749₀  000000
00115  23*        2           DCBLBL,DEGLBL,FNT(12),FMAT(12),F6,F7,LABEL1(20),           750₀  000000
00115  24*        3           LABEL2(20),LABEL3(20)                                      751₀  000000
00116  25*          COMPLEX     REGION                                                   752₀  000000
00117  26*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,         495₀  000000
00117  27*        1           IOPEN,JOPEN,PVAR,PNOM,PCPL,PEAC,PSLOSH,                  4960  000000
00117  28*        2           YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,            497₀  000000
```

```
00117    27*          3     YESSRL,NOMNAL,NOTYET                                  498₀  000000
00120    28*          LOGICAL  YESHTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,     499₀  000000
00120    29*          1     YESSRL,NOMNAL,NOTYET                                  500₀  000000
00121    32*          COMMON/KEEP19/LGAIN,LPHASE,LPOLES,LSDRL,LZEROS              ----  000000
00122    33*          LOGICAL  LGAIN,LPHASE,LPOLES,LSDRL,LZEROS                   ----  000000
00123    34*          COMMON/KEEP21/PNZ(225),PDZ(225),NUMZ,LOCNZ,LOCDZ,NNCZ(75),NDCZ(75)  ----  000000
00124    35*          COMMON/CPUD7/ PHAVAR,GAINV,SHIFT,ERP(1000),PIP(1000),NPG,NPP,  501₀  000000
00124    36*          1     NSHIFT,XP,YP(4),DELTA,NGSYN,NPSYM                     502₀  000000
00125    37*          COMPLEX  PHAVAR                                             503₀  000000
00126    38*          COMMON/CPUD3/ CU(60,60),FPRO,FPR1,FPR2,FRO,FR1,FR2,PRO,PR1,PR2,  504₀  000000
00126    39*          1     R0,R1,R2,R3,B,U,NFPO,NFP1,NFP2,NITER,NCT,NREG,NKODE,  505₀  000000
00126    40*          2     NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL        506₀  000000
00127    41*          COMPLEX  CU,FPRO,FPR1,FPR2,FRO,FR1,FR2,PRO,PR1,PR2,         ----  000000
00127    42*          1     R0,R1,R2,R3,B,U                                       508₀  000000
00130    43*          LOGICAL  DONE,RESTRT,CONJ,AUTO,REGSEL                       509₀  000000
00130    44*     C                                                                515₀  000000
00130    45*     C                                                                516₀  000000
00130    46*     C  INITIALIZE FOR SAMPLED DATA ROOT LOCUS                        517₀  000000
00130    47*     C                                                                518₀  000000
00131    48*          CALL SAVEST(1)                                              ----  000000
00132    49*          LSDRL = .TRUE.                                              ----  000002
00133    50*          REGION(1) = (0.,1.)                                         ----  000004
00134    51*          REGION(2) = (-1.,0.)                                        ----  000006
00135    52*          REGION(3) = (1.,0.)                                         ----  000010
00136    53*          REGION(4) = (0.,0.)                                         ----  000012
00137    54*          IF (NUMZ.EQ.0) GO TO 2000                                   ----  000014
00141    55*          NRP = LOCDZ - NUMZ                                          ----  000016
00142    56*          NRZ = NRP                                                   ----  000021
00143    57*          IF (MODIFY) NRP=NRP+1                                       ----  000022
00145    58*          NPG = 0                                                     520₀  000027
00146    59*          NPP = MXNPP + 1                                             521₀  000030
00146    60*     C                                                                526₀  000030
00146    61*     C                                                                527₀  000030
00146    62*     C  START OF GAIN ROOT LOCUS                                      528₀  000030
00146    63*     C                                                                529₀  000030
00147  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00147    64*          IF (PG.EQ.HALK) GO TO 200                                   530₀  000033
00151    65*          LGAIN = .TRUE.                                              ----  000037
00152    66*          LPHASE = .FALSE.                                            ----  000041
00153    67*          AUTO = .TRUE.                                               531₀  000042
00154    68*          CALL GETEST                                                 532₀  000043
00155    69*          DO 100 ITH=1,NGAIN                                          533₀  000045
00160    70*          GAINV = GAIN(ITH)                                           534₀  000052
00161    71*          LPOLES = .FALSE.                                            ----  000054
00162    72*          NOMNAL = .FALSE.                                            ----  000055
00163    73*          LZEROS = .FALSE.                                            ----  000056
00164  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00164    74*          IF (GAINV.EQ.0.0) LPOLES=.TRUE.                             ----  000057
00166  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00164    75*          IF (GAINV.EQ.1.0) NOMNAL=.TRUE.                             ----  000063
00170  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00170    76*          IF (GAINV.EQ.12345.) LZEROS=.TRUE.                          ----  000071
00172    77*          NR = NRP                                                    ----  000077
00173    78*          IF (LZEROS) NR=NRZ                                          ----  000101
00175    79*          CALL ROTEC(1000)                                            ----  000105
00176  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00176    80*          IF (GAINV.EQ.1.0 .AND. YESSRL .AND. NOTYET) CALL SAVRUT     539₀  000110
```

```
00176     81*     C                                                          540₀   000110
00176     82*     C  SAVE ROOTS AS ESTIMATES TO NEXT CASE                    541₀   000110
00176     83*     C                                                          542₀   000110
00200     84*           NA = 0                                               543₀   000124
00201     85*           IF (ITH.EQ.NGAIN) GO TO 26                           544₀   000127
00203     86*           DO 25 I=1,NEIG                                       545₀   000137
00206     87*           IF (KD(I).EQ.4) GO TO 25                             546₀   000137
00211     88*           IF (NA.GE.MXEST) GO TO 26                            547₀   000143
00213     89*           NA = NA + 1                                          548₀   000150
00214     90*           EA(NA) = ROOT(I)                                     549₀   000154
00217     91*        25 CONTINUE                                             550₀   000164
00216     92*        26 CONTINUE                                             551₀   000164
00217     93*       100 CONTINUE                                             552₀   000164
00217     94*     C                                                          553₀   000164
00217     95*     C                                                          554₀   000164
00217     96*     C  START OF PHASE ROOT LOCUS                               555₀   000164
00217     97*     C                                                          556₀   000164
00221     98*       200 CONTINUE                                             557₀   000164
00222  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00222     99*           IF (PP.EQ.HBLK) GO TO 400                            559₀   000164
00224    100*           LGAIN = .FALSE.                                              000167
00225    101*           LPHASE = .TRUE.                                              000170
00225    102*           LPOLES = .FALSE.                                             000172
00227    103*           LZEROS = .FALSE.                                             000173
00231    104*           AUTO = .FALSE.                                       561₀    000174
00231    105*           NR = NRP                                                     000175
00232    106*           CALL GETEST                                          562₀    000177
00233    107*           DO 300 ITH=1,NPHASE                                  563₀    000201
00234    108*           SHIFT = PHASE(ITH)                                   564₀    000206
00237    109*           NOMNAL = .FALSE.                                             000210
00241  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00241    110*           IF (SHIFT.EQ.0.0) NOMNAL=.TRUE.                              000211
00242    111*           CALL RUTSR$(1000)                                            000215
00243  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00243    112*           IF (SHIFT.EQ.0.0 .AND. YESSAL .AND. NOTYET) CALL SAVRUT  570₀   000220
00246    113*       300 CONTINUE                                             571₀    000235
00247    114*       400 CONTINUE                                             573₀    000235
00247    115*     C                                                          574₀    000235
00247    116*     C                                                          575₀    000235
00247    117*     C  GENERATE ROOT LOCUS PLOT                                576₀    000235
00247    118*     C                                                          577₀    000235
00250    119*           CALL SAVEST(2)                                               000235
00251  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00251    120*           IF (PPLT.NE.HBLK) CALL RLPLOT                        578₀    000237
00253    121*           RETURN                                               579₀    000245
00253    122*     C                                                          580₀    000245
00253    123*     C  ERROR IN ROOTING                                        581₀    000245
00253    124*     C                                                          582₀    000245
00253    125*     C                                                          583₀    000245
00254    126*      1000 CONTINUE                                             584₀    000251
00255    127*           CALL SAVEST(2)                                               000251
00256    128*           RETURN 1                                                     000253
00256    129*     C                                                                  000253
00256    130*     C                                                                  000253
00256    131*     C  NO Z DOMAIN TRANSFER FUNCTION HAS BEEN STORED                   000253
00256    132*     C                                                                  000253
00257    133*      2000 CONTINUE                                                     000260
```

```
00260    134*          KODE = 102                                          000260
00261    135*          CALL SAVEST(2)                                      000261
00262    136*          RETURN 1                                            000264
00261    137*          END                                      595        000324

        END OF COMPILATION:        9 DIAGNOSTICS.
```

```
@FOR,US F,8HA341,F,8HA341
FOR. SEQ=05/23/74-08:26:36 (2,3)

    SUBROUTINE 8HA341    ENTRY POINT 000261

    STORAGE USED: CODE(1) 002701 DATA(0) 000103 BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003  KEEP4  000263
    0004  KEEP5  000074
    0005  KEEP6  000134
    0006  KEEP7  000712
    0007  KEEP8  000342
    0010  KEEP16 000031
    0011  KEEP20 000227
    0012  KEEP21 001133
    0013  CRUD2  001215
    0014  CRUD4  000002

    EXTERNAL REFERENCES (BLOCK, NAME)

    0015  PFE
    0016  PFEZRO
    0017  NWDUS
    0020  NIO2$
    0021  NIO3$
    0022  NIO1$
    0023  NERR4$
    0024  NERR3$

    STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001  000176 100L      0001  000234 1000L     0001  000121 170G      0001  000141 20L       0001  000213 200L
    0001  000240 2000L     0001  000151 30L       0001  000222 300L      0001  000036 5L        0000  000011 600F
    0000  000022 601F      0000  000027 602F      0000  000041 603F      0004 L 000071 80TH      0003  000212 BY
    0005  000064 CARD      0003  000166 DD        0010 L 000022 DEBUG     0011 C 000000 ESTZ      0006  000570 FDPOL
    0006  000494 FNPOL     0003  000004 GAIN      0004 L 000073 GPRINT    0003  000151 GSYM      0000 I 000002 I
    0000  000072 INJP$     0010  000010 IOPEN     0004  000067 ITHZT     0010  000011 JOPEN     0000 I 000001 K
    0010 I 000001 KODE      0005  000110 LABEL     0010 L 000023 LFLT      0013 I 001214 LOCD      0012 I 000704 LOCDZ
    0013 I 001213 LOCN      0012 I 000703 LOCNZ     0004 L 000072 MODIFY    0000 I 000037 N         0005 R 000062 NAME
    0000 I 000003 NCOUNT    0006  000707 NDCOEF     0013  001077 NDCPER     0012  001020 NDCZ      0006 I 000705 NDGD
    0005 I 000704 NDGN      0000 I 000006 NDIFF     0010  000000 NEGZ      0011 I 000226 NESTZ     0000 I 000004 NFINAL
    0003  000003 NGAIN      0003  000154 NGR       0000 I 000010 NINC      0006  000706 NNCOEF    0013  000764 NNCPER
    0012  000705 NNCZ      0010 L 000027 NOMNAL    0010 L 000030 NOTYET    0003  000066 NPHASE    0010  000002 NRCLPL
    0003  000153 NRLFR     0010  000003 NRPOLE    0010  000004 NRZERO    0000 I 000005 NTEST     0013 I 001212 NUMPOL
    0012 I 000702 NUMZ      0010  000005 NXB       0010  000006 NXN       0010  000007 NXR       0006 I 000711 NZD
    0000 I 000000 NZEROS    0006 I 000710 NZN       0003 I 000341 NZPOLE    0004  000004 NZT       0010  000014 PCPL
    0013  000310 PD        0012  000341 PDZ       0010  000015 PFAC      0003  000000 PG        0003  000067 PHASE
    0013  000000 PIN       0010  000013 PNOM      0012  000000 PNZ       0003  000001 PP        0003  000002 PPLT
    0010  000014 PSLOSH     0003  000152 PSYM      0010  000012 PVAR      0006  000341 RID       0006  000113 RIN
    0006  000226 RRD       0005  000000 RRN       0003  000200 RX        0010  000000 STAGE     0004  000000 SUPERK
    0004 R 000003 TD        0005 R 000000 TITLE     0005  000024 TITLE1    0005  000036 TITLE2    0005  000050 TITLE3
```

```
@FOR,US [redacted]HA341,F,BHA341
FOR. SEIX-05/23/74-08:26:36 (2.3)

      SUBROUTINE BHA341     ENTRY POINT 000261

     STORAGE USED: CODE(1) 000270; DATA(0) 000103; BLANK COMMON(2) 000000

     COMMON BLOCKS:

     0003  KEEP4  000263
     0004  KEEP5  000074
     0005  KEEP6  000134
     0006  KEEP7  000712
     0007  KEEP8  000342
     0010  KEEP16 000031
     0011  KEEP20 000227
     0012  KEEP21 001133
     0013  CRUD2  001215
     0014  CRUD4  000002

     EXTERNAL REFERENCES (BLOCK, NAME)

     0015  PFE
     0016  PFEZRO
     0017  N@DUS
     0020  NIO2$
     0021  NIO3$
     0022  NIO1$
     0023  NERR4$
     0024  NERR3$

     STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000176 100L    0001 000234 1000L   0001 000121 17GG    0001 000141 20L     0001 000213 200L
0001 000240 2000L   0001 000151 30L     0001 000222 300L    0001 000036 5L      0000 000011 600F
0000 000022 601F    0000 000027 602F    0000 000041 603F    0004 L 000071 80TH  0003 000212 BY
0005 000064 CARD    0003 000166 DD      0010 L 000022 DEBUG 0011 C 000000 ESTZ  0006 000570 FDPOL
0006 000454 FNPOL   0003 000004 GAIN    0003 L 000073 GPRINT 0003 000151 GSYM   0000 I 000002 I
0000 000072 INJP$   0010 000010 IOPEN   0004 000067 ITHZT   0010 000011 JOPEN  0000 I 000001 K
0010 I 000001 KODE  0005 000110 LABEL   0010 L 000023 LFLT  0013 I 001214 LOCO  0012 I 000704 LOCOZ
0013 I 001213 LOCN  0012 I 000703 LOCNZ 0004 L 000072 MODIFY 0000 I 000007 N   0005 R 000062 NAME
0000 I 000003 NCOUNT 0006 I 000707 NDCOEF 0013 001077 NDEPER 0012 001020 NDCZ  0006 I 000705 NDGD
0005 I 003704 NDGN  0000 I 000006 NDIFF 0014 000000 NE,Z    0011 I 000226 NESTZ 0000 I 000004 NFINAL
0003 000003 NGAIN   0003 000154 NGR     0000 I 000010 NINC  0006 000706 NNCOEF  0013 000764 NNCPER
0012 000705 NNCZ    0010 L 000027 NOMNAL 0010 L 000030 NOTYET 0003 000066 NPHASE 0010 000002 NRCLPL
0003 000153 NRLER   0010 000003 NRPOLE  0010 000004 NRZERO  0000 I 000005 NTEST  0013 I 001212 NUMPOL
0012 I 000702 NUMZ  0010 000005 NX8     0010 000006 NXN     0010 000007 NXR     0006 I 000711 NZD
0000 I 000000 NZEROS 0006 I 000710 NZN  0005 I 000341 NZPOLE 0004 000004 NZT   0010 000014 PCPL
0013 000310 PD      0012 000341 PDZ     0010 000015 PFAC    0003 000000 PG      0003 000067 PHASE
0013 000000 PNN     0010 000013 PNOM    0012 000000 PNZ     0003 000001 PP      0003 000002 PPLT
0010 000016 PSLASH  0003 000152 PSYM    0010 000012 PVAR    0006 000341 RID     0006 000113 RIN
0006 000226 RRD     0005 000000 RRN     0003 000200 RX      0010 000000 STAGE   0004 000000 SUPERK
0004 R 000003 TD    0005 R 000000 TITLE 0005 000024 TITLE1  0005 000036 TITLE2  0005 000050 TITLE3
```

```
0010 L 000017 YESMTX     0010 L 000024 YESPCH     0010 L 000020 YESRAW     0003 L 000262 YESRL      0010 L 000025 YESRLP
0010 L 000026 YESSRL     0010 L 000021 YESSRP     0010 L 000001 YESZM      0004 L 000070 YESZOH     0004   000002 ZM
0007 R 000226 ZMAG       0007 C 000000 ZPOLE      0004 R 000001 ZT         0004   000005 ZTVAL
```

```
00101    1*          SUBROUTINE SHA391(*)                                                    596  000000
00102    2*          COMMON/KEEP4/ PG,PP,PPLT,NGAIN,GAIN(50),NPHASE,PHASE(50),GSYM,          597  000000
00103    3*         1          PSYM,NRLFR,NGR(10),DD(10),RX(10),BY(4,10),YESRL              598  000000
00104    4*          LOGICAL    YESRL                                                        599  000000
00105    5*          COMMON/KEEP5/ SUPERK,ZI,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,          600  000000
00106    6*         1          MODIFY,GPRINT                                                601  000000
00106    7*          LOGICAL    YESZOH,BOTH,MODIFY,GPRINT                                    603  000000
00107    8*          COMMON/KEEP6/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),       604  000000
00107    9*         1          CARD(20),LABEL(20)                                           605  000000
00110   10*          REAL       NAME                                                         606  000000
00111   11*          COMMON/KEEP7/ RRN(75),RIN(75),RRD(75),RID(75),ENPOL(76),               607  000000
00111   12*         1          EDPOL(76),NDGN,NDGD,NNCOEF,NDCOEF,NZN,NZD                    608  000000
00112   13*          COMMON/KEEP8/ ZPOLE(75),ZMAG(75),NZPOLE                                 610  000000
00113   14*          COMPLEX    ZPOLE                                                        000000
00114   15*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,             613  000000
00114   16*         1          IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                      614  000000
00114   17*         2          YESMTX,YESRAW,YESSRP,DEBUG,LELT,YESPCH,YESRLP,               615  000000
00114   18*         3          YESSRL,NOMNAL,NOTYET                                         616  000000
00115   19*          LOGICAL    YESMTX,YESRAW,YESSRP,DEBUG,LELT,YESPCH,YESRLP,               617  000000
00115   20*         1          YESSRL,NOMNAL,NOTYET                                         618  000000
00115   21*          COMMON/KEEP20/ESTZ(75),NESTZ                                            000000
00117   22*          COMPLEX    ESTZ                                                         000000
00121   23*          COMMON/KEEP21/PNZ(225),PDZ(225),NUMZ,LOCNZ,LOCDZ,NNCZ(75),NDCZ(75)      000000
00121   24*          COMMON/CRUD2/ PNN(200),PD(300),NNCPER(75),NDCPER(75),                   000000
00121   25*         1          NUMPOL,LOCN,LOCD                                             620  000000
00122   26*          COMMON/CRUD4/ NEQZ,YESZM                                                622  000000
00123   27*          LOGICAL    YESZM                                                        623  000000
00123   28*     C                                                                           624  000000
00123   29*     C                                                                           625  000000
00123   30*     C  PRINT HEADINGS PRIOR TO THE Z-R TRANSFORMATION                           626  000000
00123   31*     C                                                                           627  000000
00124   32*           IF (.NOT.GPRINT) GO TO 5                                              628  000000
00125   33*           WRITE(6,600)                                                          629  000001
00131   34*       600 FORMAT('1',41X,'Z - R   T R A N S F O R M A T I O N')                  630  000006
00131   35*           IF (YESZOH) WRITE(6,601)                                              631  000006
00131   36*       601 FORMAT(/50X,'(ZERO ORDER HOLD)')                                      632  000015
00133   37*           WRITE(6,602) TITLE,TD,ZT                                              633  000015
00142   38*       602 FORMAT(///20X,20A4//35X,'TD = ',1PE12.5,30X,'T = ',E12.5///)          634  000030
00143   39*           WRITE(6,603)                                                          635  000030
00145   40*       603 FORMAT('0   ',18X,' R O O T ',31X,'RESIDUE FOLLOWED BY COEFFICIENTS   636  000036
00145   41*          * IN DESCENDING ORDER'//14X,'REAL',11X,'IMAGINARY'//)                  637  000036
00145   42*         5 CONTINUE                                                              638  000036
00145   43*     C                                                                           639  000036
00145   44*     C                                                                           640  000036
00145   45*     C  TEST THE CONTINUOUS SYSTEM OPEN LOOP TRANSFER FUNCTION                   641  000036
00145   46*     C                                                                           642  000036
00147   47*           IF (NDGD.LE.NDGN) GO TO 1000                                          643  000036
00147   48*     C                                                                           644  000036
00147   49*     C                                                                           645  000036
00147   50*     C  DETERMINE THE NUMBER OF ZERO POLES IN THE PARTIAL FRACTION EXPANSION     646  000036
```

```
00147    51*     C                                                                  647₀    000036
00150    52*           NZEROS = NZD - NZN                                           648₀    000044
00152    53*           IF (NZEROS.LT.0) NZEROS=0                                    649₀    000046
00154    54*           IF ((NZEROS.GT.3 .AND. YESZOH) .OR.                          650₀    000053
00156    55*          *    (NZEROS.GT.2 .AND. .NOT.YESZOH)) GO TO 2000              651₀    000053
00164    56*     C                                                                  652₀    000053
00156    57*     C                                                                  653₀    000053
00151    58*     C  DETERMINE THE OPEN LOOP POLES IN THE Z DOMAIN AT 1.0            654₀    000053
00156    59*     C                                                                  655₀    000053
00154    60*           NZPOLE = 0                                                   656₀    000076
00157    61*           NESTZ = 0                                                            000077
00160    62*           IF (BOTH) GO TO 20                                           657₀    000100
00162    63*           K = NZD - NZN                                                658₀    000102
00163    64*           IF (YESZOH) K=K-1                                            659₀    000104
00165    65*           IF (K.LE.0) GO TO 20                                         660₀    000111
00167    66*           DO 10 I=1,K                                                  661₀    000115
00172    67*           NZPOLE = NZPOLE + 1                                          662₀    000121
00173    68*           ZPOLE(NZPOLE) = (1.0,0.0)                                            000125
00174    69*           ZMAG(NZPOLE) = 1.0                                                   000127
00175    70*           NESTZ = NESTZ + 1                                                    000131
00176    71*           ESTZ(NESTZ) = (1.,0.)                                                000135
00177    72*        10 CONTINUE                                                     665₀    000141
00201    73*           20 CONTINUE                                                  666₀    000141
00201    74*     C                                                                  667₀    000141
00201    75*     C                                                                  668₀    000141
00201    76*     C  INITIALIZE FOR THE Z-R TRANSFORMATION                           669₀    000141
00201    77*     C                                                                  670₀    000141
00202    78*           YESZM = .FALSE.                                              671₀    000141
00203    79*           IF (.NOT.YESZOH) GO TO 30                                    672₀    000141
00205    80*           IF (.NOT.YESRL .AND. .NOT.YESSRL) GO TO 30                   673₀    000143
00207    81*           YESZM = .TRUE.                                               675₀    000146
00210    82*        30 CONTINUE                                                     676₀    000151
00211    83*           NUMPOL = 0                                                   677₀    000151
00212    84*           LOCN = 0                                                     678₀    000151
00213    85*           LOCD = 0                                                     679₀    000152
00214    86*           NUMZ = 0                                                             000153
00215    87*           LOCNZ = 0                                                            000154
00216    88*           LOCDZ = 0                                                            000155
00217    89*           NCOUNT = NZEROS + 1                                          681₀    000156
00220    90*           NFINAL = NOGO = NZD + NZEROS                                 682₀    000161
00221    91*           NTEST = NFINAL - NZEROS + 1                                  683₀    000165
00222    92*           NDIFF = NOGO - NFINAL                                        684₀    000170
00223    93*           N = 1                                                        685₀    000173
00223    94*     C                                                                  686₀    000173
00223    95*     C                                                                  687₀    000173
00223    96*     C  EVALUATE THE PARTIAL FRACTION EXPANSION OF THE TRANSFER FUNCTION 688₀    000173
00223    97*     C  COMPUTE THE RESIDUE OF THE N-TH POLE THEN APPLY Z-R TRANSFORMATION 689₀  000173
00223    98*     C  PERFORM THE Z-R TRANSFORMATION                                  690₀    000173
00223    99*     C                                                                  691₀    000173
00225   100*        100 CONTINUE                                                    692₀    000176
00225   101*           NINC = 1                                                     693₀    000176
00226   102*           IF (N.GE.NTEST) GO TO 200                                    695₀    000177
00226   103*     C                                                                  696₀    000177
00226   104*     C                                                                  697₀    000177
00226   105*     C  A NON ZERO ROOT                                                 698₀    000177
00226   106*     C                                                                  699₀    000177
00230   107*           CALL PFE(N,NINC,NFINAL)                                      700₀    000204
```

```
00230   108*          GO TO 300                                                  7010   000211
00231   109*    C                                                                7020   000211
00231   110*    C                                                                7030   000211
00231   111*    C  A ZERO ROOT                                                   7040   000211
00231   112*    C                                                                7050   000211
00232   113*      200 CONTINUE                                                   7060   000213
00233   114*          CALL PFEZRO(S1000,N,NCOUNT,NZEROS,NDIFF)                          000213
00233   115*    C                                                                7080   000213
00233   116*    C                                                                7090   000213
00233   117*    C  SELECT NEXT POLE TO CONSIDER                                  7100   000213
00233   118*    C                                                                7110   000213
00234   119*      300 CONTINUE                                                   7120   000222
00235   120*          N = N + NINC                                               7130   000222
00236   121*          IF (N.LE.NFINAL) GO TO 100                                 7140   000224
00236   122*    C                                                                7150   000224
00236   123*    C                                                                7160   000224
00236   124*    C END OF Z-R TRANSFORMATION                                      7170   000224
00237   125*    C                                                                7180   000224
00240   126*          RETURN                                                     7190   000230
00240   127*    C                                                                7200   000230
00240   128*    C                                                                7210   000230
00240   129*    C  ERROR IN Z-R TRANSFORMATION                                   7220   000230
00240   130*    C                                                                7230   000230
00241   131*     1000 CONTINUE                                                   7240   000234
00242   132*          RETURN 1                                                   7250   000234
00242   133*    C                                                                7260   000234
00242   134*    C                                                                7270   000234
00242   135*    C  TOO MANY ZERO POLES IN THE PARTIAL FRACTION EXPANSION         7280   000234
00242   136*    C                                                                7290   000234
00243   137*      2000 CONTINUE                                                  7300   000240
00244   138*          KODE = 4                                                   7310   000240
00245   139*          RETURN 1                                                   7320   000241
00246   140*          END                                                        7330   000267

      END OF COMPILATION:  NO DIAGNOSTICS.
```

```
BFOR:V64 F+BLOCK,F+BLOCK
FOR 3[1%-05/23/74-09:27:03-64.5]

      BLOCK DATA

      STORAGE USED: CODE(1) 000001  DATA(0) 000001  BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP2  000047
      0004   KEEP6  000134
      0005   KEEP14 000031
      0006   KEEP15 000260
      0007   KEEP16 000031


      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0004   000064 CARD       0006 R 000116 CPSLBL     0004 R 000024 DA         0006 R 000130 DCBLBL     0007 L 000022 DEBUG
      0005 R 000025 DEG        0006 R 000131 DEGLBL     0003 D 000002 DHFILT     0005 R 000012 D0         0005 R 000013 D1
      0005 R 000014 D2         0005 R 000015 D3         0005 R 000016 D4         0005 C 000010 FIFTEN     0005 R 000017 FIFTY
      0006   000146 FMAT       0004   000132 FMT        0005 C 000006 FOUR       0006   000162 F6         0006   000163 F7
      0003 R 000037 HACC       0005 C 000000 HALF       0003 R 000025 HAST       0003 R 000035 HATT       0003 D 000000 HBLANK
      0003 R 000009 HBLK       0003 R 000034 HCCW       0003 R 000033 HCW        0003 R 000032 HDEC       0003 R 000027 HDOT
      0003 R 000006 HESTI      0003 R 000043 HFGD       0003 R 000044 HFGN       0003 R 000040 HFPD       0003 R 000045 HFPDD
      0003 R 000042 HFPDN      0003 R 000041 HFPN       0003 R 000010 HGENE      0003 R 000031 HINC       0003 R 000005 HKEY
      0003 R 000007 HMATR      0003 R 000014 HNEW       0003 R 000046 HNOMI      0003 R 000015 HNYQU      0003 R 000023 HO
      0003 R 000026 HPLUS      0003 R 000036 HRATE      0003 R 000011 HRAW       0003 R 000012 HRETA      0003 R 000017 HROLL
      0003 R 000016 HROOT      0003 R 000013 HSTAN      0003 R 000030 HSTAR      0003 R 000020 HSIC       0003 R 000021 HS2
      0003 R 000022 HS4B       0003 R 000024 HX         0006 I 000100 IDA        0006 I 000107 IDB        0007   000010 IOPEN
      0007   000011 IOPEN      0007   000001 KODE       0004   000110 LABEL      0006 I 000164 LABELI     0006 I 000210 LABEL2
      0006 I 000234 LABEL3     0007 L 000023 LFLT       0004 R 000062 NAME       0007 L 000027 NOMNAL     0007 L 000030 NOTYET
      0007   000002 NRCLPL     0007   000003 NRPOLE     0007   000004 NRZERO     0007 I 000005 NXB        0007 I 000006 NXN
      0007 I 000007 NXR        0005 I 000026 N1         0005 I 000027 N2         0005 I 000030 N3         0005 C 000002 ONE
      0007   000014 PCPL       0007   000015 PFAC       0005 R 000020 PI         0005 R 000021 PI2        0007   000013 PNOM
      0007   000016 PSLOSH     0007   000012 PVAR       0006 R 000034 P36        0005 R 000024 RADDEG     0006 C 000000 REGION
      0006 R 000033 RID        0005 R 000023 RPI        0005 R 000022 SMALL      0006 R 000014 SPACE      0007   000000 STAGE
      0004   000000 TITLE      0004   000024 TITLE1     0004   000036 TITLE2     0004 R 000050 TITLE3     0005 C 000004 TWO
      0006 R 000010 WIDTH      0007 L 000017 YESMTX     0007 L 000024 YESPCH     0007 L 000020 YESRAW     0007 L 000025 YESRLP
      0007 L 000026 YESSRL     0007 L 000021 YESSRP     0004 R 000020 YINC

      00101    1*          BLOCK DATA                                                                    734B   000000
      00102    2*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,                 735B   000000
      00102    3*        1             HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS4B,               736B   000000
      00102    4*        2             HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,                 737B   000000
      00102    5*        3             HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,                738B   000000
      00102    6*        4             HNOMI                                                           739B   000000
      00103    7*          DOUBLE PRECISION HBLANK,DHFILT                                                       000000
      00104    8*          COMMON/KEEP6/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),           741B   000000
      00104    9*        1             CARD(20),LABEL(20)                                              742B   000000
      00104   10*          REAL          NAME                                                          743B   000000
      00104   11*          COMMON/KEEP14/ HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,            744B   000000
      00104   12*        1             PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                               745B   000000
      00104   13*          COMPLEX       HALF,ONE,TWO,FOUR,FIFTEN                                             000000
```

```
00110   14*         COMMON/KEEP15/REGION(4),WIDTH(4),SPACE(4),YINC(4),                  000000
00110   15*       1 DA(4),RID(4),P36(36),IDB(7),IDG(7),CPSLBL(10),          749₀  000000
00110   16*       2 DCBLBL,DEGLBL,FMT(12),FMAT(12),F6,F7,LABEL1(20),        750₀  000000
00110   17*       3 LABEL2(20),LABEL3(20)                                   751₀  000000
00111   18*     COMPLEX   REGION                                            752₀  000000
00112   19*     COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,  754₀  000000
00112   20*       1 IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                 755₀  000000
00112   21*       2 YESMTX,YESRAW,YESCRP,DEBUG,LFLT,YESPCH,YESRLP,          756₀  000000
00112   22*       3 YESSRL,NOMNAL,NOTYET                                    757₀  000000
00113   23*     LOGICAL   YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,    758₀  000000
00113   24*       1 YESSRL,NOMNAL,NOTYET                                    759₀  000000
00114   25*     DATA      HBLANK,DHEILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,   800₀  000000
00114   26*       1 HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HS1C,HS2,HS4B,       801₀  000000
00114   27*       2 HD,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,         802₀  000000
00114   28*       3 HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD         803₀  000000
00114   29*       4 /    ,    ,    ,'FILTER',    ,    ,    ,'KEY ',        804₀  000000
00114   30*       5 'ESTI',  'MATR',  'GENE',  'RAW ',  'RETA',             805₀  000000
00114   31*       6 'STAN',  'NEW ',  'NYQU',  'ROOT',  'ROLL',             806₀  000000
00114   32*       7 'S-1C',  'S-2 ',  'S-4B',  'D   ',  'X   ',             807₀  000000
00114   33*       8 '*   ',  '+   ',  '.   ',  '*   ',  'INC ',             808₀  000000
00114   34*       9 'DEC ',  'CW  ',  'CCW ',  ' ATT',  'RATE',             809₀  000000
00114   35*       A ' ACC',  'FPD ',  'FPN ',  'FPDN',  'FGD ',             810₀  000000
00114   36*       B 'FGN ',  'FPDD' /                                        811₀  000000
00161   37*     DATA      HNOMI / 'NOMI' /                                  812₀  000000
00163   38*     DATA      HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,  813₀  000000
00163   39*       1 PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                       814₀  000000
00163   40*       2 / 0.5,0.0) , (1.0,0.0) , (2.0,0.0) ,                          000000
00163   41*       3 (4.0,0.0) , (1.0E+5,0.0) , 0.0 , 1.0 ,                        000000
00163   42*       4 2.0 , 3.0 , 4.0 , 1.0E35 , 3.14159 , 6.283185 ,               000000
00163   43*       5 1.0E-6 , 136.5 , 0.0174533 , 57.29578 , 1 , 2 , 3 /           000000
00211   44*     DATA DA /10., 10., 5., 5./                                  820₀  000000
00212   45*     DATA RID/' 1', ' 1', ' 10', ' 100'/                         821₀  000000
00214   46*     DATA P36/' 0', ' 10', ' 20', ' 30', ' 40', ' 50', ' 60',    822₀  000000
00214   47*       1 ' 70', ' 80', ' 90', ' 100', ' 110', ' 120', ' 130', ' 140',  823₀  000000
00214   48*       2 ' 150', ' 160', ' 170', ' 180', ' 190', ' 200', ' 210', ' 220',  824₀  000000
00214   49*       3 ' 230', ' 240', ' 250', ' 260', ' 270', ' 280', ' 290', ' 300',  825₀  000000
00214   50*       4 ' 310', ' 320', ' 330', ' 340', ' 350'/                   826₀  000000
00216   51*     DATA      NXB,NXN,NXR / 0,0,0 /                             827₀  000000
00223   52*     DATA DCBLBL,DEGLBL /'DB  ','DEG '/                          828₀  000000
00225   53*     DATA IDB /' 60', ' 40', ' 20', '  0', '-20', '-40', '-60'/   829₀  000000
00227   54*     DATA IDG /'  0', '-60', '-120', '-180', '-240', '-300', '-360'/  830₀  000000
00231   55*     DATA CPSLBL /' .01', 'HZ  ', ' .1', 'HZ  ', ' 1.', 'HZ  ',  831₀  000000
00231   56*       1 ' 10.', ' HZ ', '100.', ' HZ '/                         832₀  000000
00233   57*     DATA WIDTH/ 6.25,2.8125,1.875,1.25 /                        833₀  000000
00235   58*     DATA SPACE/ 0.0,.625,.3125,.416666 /                        834₀  000000
00237   59*     DATA YINC/20.0,9.0,6.0,4.0/                                 835₀  000000
00241   60*     DATA LABEL1 / 'CALC', 'ULAT', 'ION ', 'OF T', 'HE C',       843₀  000000
00241   61*       1 'ONTI', 'NUOU', 'S SY', 'STEM', ' OPE',                 844₀  000000
00241   62*       2 'N LO', 'OP Z', 'EROS', 7*' ' /                         845₀  000000
00243   63*     DATA LABEL2 / 'CALC', 'ULAT', 'ION ', 'OF T', 'HE C',       846₀  000000
00243   64*       1 'ONTI', 'NUOU', 'S SY', 'STEM', ' OPE',                 847₀  000000
00243   65*       2 'N LO', 'OP P', 'OLES', 7*' ' /                         848₀  000000
00245   66*     DATA LABEL3 / 'CALC', 'ULAT', 'ION ', 'OF T', 'HE C',       849₀  000000
00245   67*       1 'ONTI', 'NUOU', 'S SY', 'STEM', ' CLO',                 850₀  000000
00245   68*       2 'SED ', 'LOOP', ' POL', 'ES  ', 6*' ' /                 851₀  000000
00247   69*     DATA TITLE3 / 'SYST', 'EM O', 'PENE', 'D AT', ' VAR',       852₀  000000
00247   70*       1 'LABL', 'E   ', 3*'    ' /                            853₀  000000
```

```
00251                    END                                    8540    000000

END OF COMPILATION:      NO  DIAGNOSTICS.
```

```
FFOR,USW F.BODE,F.BODE
FOR SEIX-05/23/74-08:27:09 (2.3)

      SUBROUTINE BODE      ENTRY POINT 000006

      STORAGE USED: CODE(1) 000010; DATA(0) 000004; BLANK COMMON(2) 000000

      EXTERNAL REFERENCES (BLOCK, NAME)

          0003    NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

          0000    000000 IN0P$

00101!    1*            SUBROUTINE BODE                                  000000
00101     2*        C                                                    000000
00101     3*        C                                                    000000
00101     4*        C  DUMMY SUBROUTINE FOR BODE PLOT                    000000
00101     5*        C                                                    000000
00103     6*              RETURN                                         000000
00104     7*              END                                            000007

      END OF COMPILATION:      NO  DIAGNOSTICS.
```

```
?FOR,US F,CDABV,F,CDABV
FOR-SE1X-05/23/74-08:27:15 (2,3)

      FUNCTION CDABV        ENTRY POINT 000062

      STORAGE USED: CODE(1) 0n0066; DATA(0) 0n0016; BLANK COMMON(2) 000000

      EXTERNAL REFERENCES (BLOCK, NAME)

        0003    SQRF
        0004    NERR3S

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

        0001    000040 IL       0000 C 000004 ARG       0000 R 000000 CDABV     0000 R 000004 E         0000   000007 INJPS
        0000 R-000001 X         0000 R-000002 Y         0000 R-000003 Z

00101    1*             FUNCTION CDABV(CMOD)                                                          000000
00103    2*             COMPLEX        CMOD,ARG                                                       000000
00104    3*             DIMENSION      E(2)                                                           000000
00105    4*             EQUIVALENCE    (ARG,E(1))                                                     000000
00105    5*       C                                                                          1117(0)  000000
00106    6*       C                                                                          1118(0)  000000
00105    7*       C***** CALCULATE MODULUS OF COMPLEX ARGUMENT                               1119(0)  000000
00105    8*       C                                                                          1120(0)  000000
00106    9*             ARG = CMOD                                                           1121(0)  000000
00107   10*             X = ABS(E(1))                                                                 000001
00110   11*             Y = ABS(E(2))                                                                 000003
00111   12*             Z=X+Y                                                                1124(0)  000005
00112   13*             CDABV = 0.0                                                                   000007
00113    *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00113   14*             IF(Z.EQ.0.0) RETURN                                                  1126(0)  000010
00115   15*             IF(X.LT.Y) GO TO 1                                                   1127(0)  000015
00117   16*             CDABV=X*SQRT(1.0+(Y/X)**2)                                           1128(0)  000022
00120   17*             RETURN                                                               1129(0)  000034
00121   18*       1     CONTINUE                                                             1130(0)  000040
00122   19*             CDABV=Y*SQRT(1.0+(X/Y)**2)                                           1131(0)  000040
00123   20*             RETURN                                                               1132(0)  000051
00124   21*             END                                                                  1133(0)  000065

      END OF COMPILATION:       1 DIAGNOSTICS.
```

```
@FOR,US F.CLPOLE,F.CLPOLE
FOR SEIx-05/23/74-08:27:20 (2,3)

      SUBROUTINE CLPOLE    ENTRY POINT 000033

      STORAGE USED: CODE(1) 000041; DATA(D) 000010; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP6  000134
      0004   KEEP9  000705
      0005   KEEP15 000260
      0006   KEEP16 000031

      EXTERNAL REFERENCES (BLOCK, NAME)

      0007   COMPUT
      0010   NERR45
      0011   NERR35

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000015 1000L      0001   000004 115G       0003   000064 CARD       0005   000116 CPSLBL     0005   000024 DA
      0005   000130 DCALBL     0006 L 000022 DEBUG      0005   000131 DEGLBL     0004 C 000226 EA         0005   000146 FMAT
      0005   000132 FMT        0005   000162 F6         0005   000163 F7         0000 I 000000 I          0005   000100 IDB
      0005   000107 IDG        0000   000002 INJP5      0006   000010 IOPEN      0006   000011 JOPEN      0004   000572 KD
      0006   000001 KODE       0003 I 000110 LABEL      0005   000164 LABEL1     0005   000210 LABEL2     0005 I 000234 LABEL3
      0004   000023 LFLT       0004   000455 NA         0003 R 000062 NAME       0004   000454 NEIG       0004   000457 NI
      0004 L 000027 NORMAL     0006 L 000030 NOTYET     0004 I 000456 NP         0006 I 000002 NRCLPL     0006   000003 NRPOLE
      0004   000004 NRZERO     0006   000006 NXB        0006   000006 NXN        0006   000007 NXR        0006   000014 PCPL
      0006   000015 PFAC       0006   000013 PNOM       0006   000016 PSLOSH     0006   000012 PVAR       0005   000034 P36
      0005 C 000000 REGION     0005   000030 RID        0004 C 000000 ROOT       0005   000014 SPACE      0006   000000 STAGE
      0003   000000 TITLE      0003   000024 TITLE1     0003   000036 TITLE2     0003   000050 TITLE3     0005   000010 WIDTH
      0006 L 000017 YESMTX     0006 L 000024 YESPCH     0006 L 000020 YESRAW     0006 L 000025 YESRLP     0006 L 000026 YESSRL
      0006 L 000021 YESSRP     0005   000020 YINC

00101    1*            SUBROUTINE CLPOLE(*)                                                    1134D   000000
00103    2*            COMMON/KEEP6/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),       1135D   000000
00103    3*           1              CARD(20),LABEL(20)                                        1136D   000000
00104    4*            REAL          NAME                                                      1137D   000000
00105    5*            COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)                  1138D   000000
00106    6*            COMPLEX       ROOT,EA                                                           000000
00107    7*            COMMON/KEEP15/REGION(4),WIDTH(4),SPACE(4),YINC(4),                              000000
00107    8*           1              DA(4),RID(4),P36(36),IDB(7),IDG(7),CPSLBL(10),            1142D   000000
00107    9*           2              DCPLBL,DEGLBL,FMT(12),FMAT(12),F6,F7,LABEL1(20),          1143D   000000
00107   10*           3              LABEL2(20),LABEL3(20)                                     1144D   000000
00110   11*            COMPLEX       REGION                                                    1145D   000000
00111   12*            COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,              1147D   000000
00111   13*           1              IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                   1148D   000000
00111   14*           2              YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,            1149D   000000
```

```
00111    15*          3              YESSRL,NOMNAL,NOTYET                          11500   000000
00112    16*                LOGICAL  YESMTX,YESRAN,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,  11510   000000
00112    17*          1              YESSRL,NOMNAL,NOTYET                          11520   000000
00112    18*     C                                                                11590   000000
00112    19*     C                                                                11600   000000
00112    20*     C  CALCULATE THE CLOSED LOOP POLES                               11610   000000
00112    21*     C                                                                11620   000000
00113    22*          NR=NPCLPL                                                   11660   000000
00114    23*          DO 10 I=1,20                                                11670   000004
00117    24*       10 LABEL(I) = LABEL3(I)                                        11660   000004
00121    25*          CALL COMPUT(*1000)                                                  000006
00122    26*          RETURN                                                      11700   000011
00122    27*     C                                                                11710   000011
00122    28*     C                                                                11720   000011
00122    29*     C  ERROR IN COMPUTING THE CLOSED LOOP POLES                      11730   000011
00122    30*     C                                                                11740   000011
00123    31*      1000 CONTINUE                                                   11750   000015
00124    32*          RETURN 1                                                    11760   000015
00125    33*          END                                                         11770   000040

          END OF COMPILATION:      NO  DIAGNOSTICS.
```

```
@FOR,US F.COMPUT,F.COMPUT
FOR SCIX=05/23/74-08:27:29 (2,3)

     SUBROUTINE COMPUT    ENTRY POINT 000157

     STORAGE USED: CODE(1) 000163; DATA(0) 000014; BLANK COMMON(2) 000000

     COMMON BLOCKS:

     0003  KEEP1  000026
     0004  KEEP5  000074
     0005  KEEP9  000705
     0006  KEEP10 021620
     0007  KEEP16 000031
     0010  KEEP19 000005
     0011  CRUD3  016115
     0012  CRUD5  000010

     EXTERNAL REFERENCES (BLOCK, NAME)

     0013  RLOCUS
     0014  KCALC
     0015  N.DUS$
     0016  N102$
     0017  NERR4$
     0020  NERR3$

     STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001  000052 10L     0001  000127 1000L   0001  000101 40L     0001  000133 5000L   0001  000117 60L
0000  000000 600F    0001  000141 6000L   0011 L 016113 AUTO    0011 C 016072 B       0004 L 000071 BOTH
0011 L 016112 CONJ    0011 C 000000 CU      0007 L 000022 DEBUG   0012 I 000005 DEXPO    0011 L 016110 DONE
0012 C 000002 DOAN    0005 C 000226 EA      0006   016664 EV      0011 C 016040 FPRO     0011 C 016042 FPR1
0011 C 016044 FPR2    0011 C 016046 FRD     0011 C 016050 FR1     0011 C 016052 FR2      0004 L 000073 GPRINT
0000  000010 INJP$    0007  000016 IOPEN    0006  000004 IR      0004  000067 ITHZT    0006  001754 JC
0007  000011 JOPEN    0005  000572 KD      0007 I 000001 KODE    0007 L 000023 LFLT     0010 L 000000 LGAIN
0006  005674 LL      0006  007644 LOCPOL   0010 L 000001 LPHASE  0010 L 000002 LPOLES   0010 L 000003 LSORL
0010 L 000004 LZEROS  0003  000000 MAXIT    0006 L 000072 MODIFY  0003 I 000001 MXEIG    0003  000024 MXEIGT
0003  000023 MXEST   0003  000002 MXFRM    0003  000003 MXNBM   0003  000004 MXNCOF   0003  000005 MXNCT
0003  000025 MXNCV   0003  000006 MXNE     0003  000007 MXNEQ   0003  000010 MXNF1    0003  000011 MXNG
0003  000012 MXNPH   0003  000013 MXNPP    0003  000014 MXNQPT  0003  000015 MXNSM    0003  000016 MXNSP
0003  000017 MXNTM   0003  000020 MXNV     0003  000021 MXNZT   0003  000022 MXPOLY   0005  000455 NA
0006 I 000003 NCOF    0011  016102 NCT     0006  003724 ND      0006 I 000000 NDEG     0006 I 000002 NE
0105 I 000454 NEIG    0006 I 000001 NEQ     0011  016107 NEST    0012 I 000004 NEXPO    0011  016076 NFPO
0011  016077 NFP1    0012  000006 NG      0005  000457 NI      0011  016101 NITER    0011  016104 NKODE
0007 L 000027 NORMAL  0007 L 000030 NOTYET   0011  016100 NPF2    0005 I 000456 NR       0007  000002 NRCLPL
0011  016103 NREG    0007  000003 NRPOLE   0007  000004 NRZERO  0011  016105 NSTART   0011  016106 NTIME
0007  000005 NXB     0007  000006 NXN     0007  000007 NXR     0012  000007 NZ      0004  000004 NZT
0007  000014 PCPL    0007  000015 PFAC    0007  000013 PNOM    0011 C 016054 PRO     0011 C 016056 PR1
0011 C 016060 PR2     0007  000016 PSLOSH   0007  000012 PVAR    0011 L 016114 REGSEL   0011 L 016111 RESTRT
0005 C 000000 ROOT    0011 C 016062 RP      0011 C 016064 RI      0011 C 016066 R2       0011 C 016070 R3
0007  000000 STAGE   0004  000000 SUPERK   0005  000003 TE      0011 C 016074 U       0012 C 000000 UP
0007 L 000017 YESNTX  0007 L 000024 YESPCH  0007 L 000020 YESRAW  0007 L 000025 YESRLP  0007 L 000026 YESSRL
```

```
0007 L 000021 YESSRP    0004 L 000070 YESZOH    0004   000002 ZM        0004   000001 ZT        0004   000005 ZTVAL

00101    1*          SUBROUTINE COMPUT(*)                                          1178₀  000000
00103    2*          COMMON/KEEP1/ MAXIT,MXEIG,NXFRM,MXNBH,MXNCOF,MXNCT,MXNE,MXNEQ,  1179₀  000000
00103    3*        1               MXNFI,MXNG,MXNFH,MXNPP,MXNGPT,MXNSM,MXNSP,MXNTM,  1180₀  000000
00103    4*        2               MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV              1181₀  000000
00104    5*          COMMON/KEEP5/ SUPERK,ZT,ZM,TO,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,   1182₀  000000
00104    6*        1               MODIFY,GPRINT                                     1183₀  000000
00105    7*          LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                         1185₀  000000
00106    8*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)          1186₀  000000
00107    9*          COMPLEX       ROOT,EA                                                  000000
00110   10*          COMMON/KEEP10/NDEG,NEQ,NE,NCOF,PR(1000),JC(1000),ND(1000),       1189₀  000000
00110   11*        1               LL(1000),LOCPOL(60,60),EV(1500)                   1190₀  000000
00111   12*          COMMON/KEEP14/STAGE,KODE,NRCPL,NRPOLE,NRZERO,NXB,NXN,NXR,       1191₀  000000
00111   13*        1               IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,          1192₀  000000
00111   14*        2               YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,   1193₀  000000
00111   15*        3               YESSRL,NOMNAL,NOTYET                              1194₀  000000
00112   16*          LOGICAL       YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,   1195₀  000000
00112   17*        1               YESSRL,NOMNAL,NOTYET                              1196₀  000000
00113   18*          COMMON/KEEP19/GAIN,LPHASE,LPOLES,LSDRL,LZEROS                           000000
00114   19*          LOGICAL       GAIN,LPHASE,LPOLES,LSDRL,LZEROS                           000000
00115   20*          COMMON/CRUD3Z/CV(60,60),FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,  1197₀  000000
00115   21*        1               R0,R1,R2,R3,U,V,NFP0,NFP1,NPF2,NITER,NCT,NREG,NKODE,  1198₀  000000
00115   22*        2               NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL    1199₀  000000
00116   23*          COMPLEX       CV,FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,               000000
00116   24*        1               R0,R1,R2,R3,B,U                                   1201₀  000000
00117   25*          LOGICAL       DONE,RESTRT,CONJ,AUTO,REGSEL                      1202₀  000000
00120   26*          COMMON/CRUD5/ UP,DOWN,NEXPO,DEXPO,NG,NZ                         1203₀  000000
00121   27*          INTEGER       DEXPO                                             1204₀  000000
00122   28*          COMPLEX       UP,DOWN                                           1205₀  000000
00122   29*    C                                                                     1207₀  000000
00122   30*    C                                                                     1208₀  000000
00122   31*    C  INITIALIZE AND CHECK WHETHER A MATRIX HAS BEEN DEFINED             1209₀  000000
00122   32*    C                                                                     1210₀  000000
00123   33*          AUTO = .TRUE.                                                   1211₀  000000
00124   34*          NEIG = 0                                                        1212₀  000001
00125   35*          IF (NR.LE.0 .OR. NR.GT.MXEIG) NR = MIN0(MXEIG,NEQ*(NDEG-1))    1213₀  000002
00127   36*          IF (NE.GT.0 .AND. NCOF.GT.0) GO TO 10                           1214₀  000027
00127   37*    C                                                                     1215₀  000027
00127   38*    C                                                                     1216₀  000027
00127   39*    C  NO MATRIX HAS BEEN DEFINED                                         1217₀  000027
00127   40*    C                                                                     1218₀  000027
00131   41*          KODE = 5                                                        1219₀  000043
00132   42*          RETURN 1                                                        1220₀  000045
00132   43*    C                                                                     1221₀  000045
00132   44*    C                                                                     1222₀  000045
00132   45*    C     O P E N    L O O P    Z E R O S                                1223₀  000045
00132   46*    C                                                                     1224₀  000045
00133   47*       10 IF (.NOT.LZEROS) GO TO 40                                             000052
00135   48*          CALL RLOCUS(S,1000)                                                   000053
00136   49*          IF (NEIG.EQ.0 .AND. GPRINT) WRITE(6,600)                        1250₀  000056
00141   50*      600 FORMAT(///20X,'NO OPEN LOOP ZEROS EXIST')                       1251₀  000071
00142   51*          CALL KCALC (UP,NEXPO)                                           1255₀  000071
00143   52*          RETURN                                                          1273₀  000075
```

```
00143     53*     C                                                    12740     000075
00143     54*     C                                                    12750     000075
00143     55*     C     O P E N   L O O P   P O L E S                  12760     000075
00143     56*     C                                                    12770     000075
00144     57*        40 IF (.NOT.LPOLES) GO TO 60                                 000101
00146     58*           CALL RLOCUS($1000)                                        000102
00147     59*           IF (NEIG.EQ.0) GO TO 5000                      13000     000105
00151     60*           CALL KCALC (DOWN,DEXPO)                        13040     000107
00152     61*           RETURN                                         13180     000113
00152     62*     C                                                    13190     000113
00152     63*     C                                                    13200     000113
00152     64*     C     C L O S E D   L O O P   P O L E S              13210     000113
00152     65*     C                                                    13220     000113
00153     66*        60 CONTINUE                                       13260     000117
00154     67*           CALL RLOCUS($1000)                                        000117
00155     68*           IF (NEIG.EQ.0) GO TO 6000                      13280     000121
00157     69*           RETURN                                         13290     000123
00157     70*     C                                                    13300     000123
00157     71*     C                                                    13310     000123
00157     72*     C ERROR IN ROOT LOCUS CALCULATIONS                   13320     000123
00157     73*     C                                                    13330     000123
00160     74*     1000 CONTINUE                                        13340     000127
00161     75*           RETURN 1                                       13350     000127
00161     76*     C                                                    13570     000127
00161     77*     C                                                    13580     000127
00161     78*     C  NO CONTINUOUS SYSTEM OPEN LOOP POLES              13590     000127
00161     79*     C                                                    13600     000127
00162     80*     5000 CONTINUE                                        13610     000133
00163     81*           KODE = 9                                       13620     000133
00164     82*           RETURN 1                                       13630     000134
00164     83*     C                                                    13640     000134
00164     84*     C                                                    13650     000134
00164     85*     C  NO CONTINUOUS SYSTEM CLOSED LOOP POLES            13660     000134
00164     86*     C                                                    13670     000139
00165     87*     6000 CONTINUE                                        13680     000141
00166     88*           KODE = 10                                      13690     000141
00167     89*           RETURN 1                                       13700     000142
00170     90*           END                                            13710     000162

          END OF COMPILATION:      NO  DIAGNOSTICS.
```

```
@FOR,JSN F.CSMTRX,F.CSMTRX
FOR.SEIX=05/23/74=08:27:44 +2,3)

      SUBROUTINE CSMTRX     ENTRY POINT 000074

      STORAGE USED: CODE(I) 000100; DATA(D) 000141; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP5   000074
      0004   KEEP7   000712
      0005   KEEP18  000002

      EXTERNAL REFERENCES (BLOCK, NAME)

      0006   N4DU$
      0007   N102$
      0010   N101$
      0011   NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001  000025 I20G      0001  000040 I30G      0001  000063 5HL       0000  000001 600F      0000  000032 601F
      0000  000052 602F      0000  000073 603F      0003 L 000071 BOTH      0004 R 000570 FDPOL     0004 R 000454 FNPOL
      0003 L 000073 GPRINT    0000 I 000000 I        0000  000131 INJP$     0003  000067 ITHZT     0003 L 000072 MODIFY
      0004 I 000707 NDCOEF    0004 I 000705 NDGD     0004 I 000704 NDGN      0005 I 000001 NDSCL     0004 I 000706 NNCOEF
      0005 I 000000 NNSCL     0004  000711 NZD       0004  000710 NZN       0003  000004 NZT       0004  000341 RID
      0004  000113 RIN       0004  000226 RRD       0001  000000 RRN       0003 R 000000 SUPERK    0003  000003 TD
      0003 L 000070 YESZOH    0003  000002 ZM        0003  000001 ZT        0003  000005 ZTVAL

      SUBROUTINE CSMTRX                                                                    1372D  000005
      COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,                       1373D  000005
     1          MODIFY,GPRINT                                                              1374D  000005
      LOGICAL     YESZOH,BOTH,MODIFY,GPRINT                                              1376D  000005
      COMMON/KEEP7/ RRD(75),RIN(75),RRD(75),RID(75),FNPOL(76),                           1377D  000005
     1          FDPOL(76),NDGN,NDGD,NNCOEF,NDCOEF,NZN,NZD                                1378D  000005
      COMMON/KEEP18/ NNSCL,NDSCL                                                         1380D  000005
C                                                                                        1381D  000005
C                                                                                        1382D  000005
C  PRINT THE CONTINUOUS SYSTEM OPEN LOOP TRANSFER FUNCTION                               1383D  000005
C                                                                                        1384D  000005
      IF (.NOT.GPRINT) GO TO 50                                                          1385D  000005
      WRITE(6,600) SUPERK                                                                1386D  000007
  600 FORMAT('1',15X,'C O N T I N U O U S   S Y S T E M   O P E N   L O                  1387D  000015
     1OP  T R A N S F E R  F U N C T I O N'////' LEADING COEFFICIENT                      1388D  000015
     2 K = ',1PE12.5)                                                                    1389D  000015
      WRITE(6,601) NDGD,(FDPOL(I),I=1,NNCOEF)                                            1390D  000015
  601 FORMAT(///' NUMERATOR DEGREE',I5,10X,'COEFFICIENTS IN ASCENDING PO                 1391D  000030
     1WERS OF S'//(10X,6(1PE12.5,5X)))                                                   1392D  000030
      WRITE(6,602) NDGN,(FDPOL(I),I=1,NDCOEF)                                            1393D  000030
```

```
00134    21*      602 FORMAT(///' DENOMINATOR DEGREE',I5,10X,'COEFFICIENTS IN ASCENDING P   13940   000043
00134    22*        *OWERS OF S'//(10X,6(1PE12.5,5X)))                                     13950   000043
00135    23*        IF (NNSCL.NE.0 .OR. NDSCL.NE.0) WRITE(6,603) NNSCL,NDSCL                13960   000043
00142    24*      603 FORMAT(////' SCALE FACTORS WERE REQUIRED IN EXPANDING THE OPEN LOO    13970   000063
00142    25*        1P ZEROS AND POLES INTO POLYNOMIALS'//30X,'OPEN LOOP ZERO SCALE =',     13980   000063
00142    26*        2 I3/30X,'OPEN LOOP POLE SCALE =',I3)                                   13990   000063
00143    27*       50 CONTINUE                                                              14000   000063
00144    28*          RETURN                                                               14010   000063
00145    29*          END                                                                  14020   000077
```

END OF COMPILATION: NO DIAGNOSTICS.

```
?FOR,US" F,CSOLTF,F,CSOLTF
FOR SEGX-05/23/74-08:28:07-47.81

    SUBROUTINE CSOLTF     ENTRY POINT 000260

    STORAGE USED: CODE(1) 00027D; DATA(0) 000D30; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003  KEEP5  000074
    0004  KEEP7  000712
    0005  KEEP10 021620
    0006  KEEP15 000260
    0007  KEEP16 006031
    0010  KEEP18 000002
    0011  KEEP19 000005
    0012  CRUD5  000010

    EXTERNAL REFERENCES (BLOCK, NAME)

    0013  PRINTT
    0014  POLES
    0015  FORM
    0016  ZEROS
    0017  CLPOLE
    0020  XPR1
    0021  COV$
    0022  NERR4$
    0023  NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001  000216 100DL    0001  000101 12L     0001  000105 13L     0001  000070 143G    0001  000222 2000L
    0001  000230 3000L    0001  000236 4000L   0003 L 000071 80TH    0006  000116 CPSLBL  0006  000024 DA
    0006  000130 DCBLBL   0007 L 000022 DEBUG   0006  000131 DEGLBL  0012 I 000005 DEXPO   0012 C 000002 DOWN.
    0006  000030 DUM      0005  016664 EV      0004 R 000570 FDPOL   0006  000146 FMAT    0006  000132 FMT
    0004 R 000454 FNPOL   0006  000162 F6      0006  000163 F7      0003 L 000073 GPRINT   0000 I 000000 I
    0006  000100 IDB      0006  000107 IDG     0000  000015 INJPS   0007 I 000010 IOPEN    0005  000004 IR
    0003  000067 ITH7T    0005  001754 JC      0007 I 000011 JOPEN   0007 I 000001 KODE    0006  000164 LABEL1
    0006  000210 LABEL2   0006  000234 LABEL3  0007 L 000023 LFLT    0011 L 000000 LGAIN   0005  005674 LL
    0005 I 007644 LOCPOL  0011 L 000001 LPHASE  0011 L 000002 LPOLES  0011 L 000003 LSDRL   0011 L 000004 LZEROS
    0003 L 000022 MODIFY  0005  000003 NCOF    0005  003724 NL      0004 I 000707 NDCOEF   0005  000000 NDEG
    0004 I 000705 NDGD    0004 I 000704 NDGN    0010 I 000001 NESCL   0005  000002 NE      0005 I 000001 NEQ
    0012 I 000004 NEXPO   0012  000006 NG      0004 I 000706 NECOEF   0010 I 000000 NNSCL   0007 L 000027 NOMNAL
    0007 L 003030 NOTYET  0000 I 000001 NPT     0007  000002 NRCLPL   0007  000003 NRPOLE   0007  000004 NRZERO
    0007  000005 NXB      0007  000006 NXN     0005  000007 NXR     0012  000007 NZ      0004 I 000711 NZO
    0004 I 000710 NZN     0003  000004 NZT     0007  000014 PCPL    0007  000015 PFAC    0007  000013 PNOM
    0000 R 000002 POWER   0007  000016 PSLOSH  0005  000012 RVAR    0006  000034 P36     0006 C 000000 REGION
    0004 R 000341 RID     0004 R 000113 RIN     0004 R 000226 RRD     0004 R 000000 RRN     0006  000014 SPACE
    0007  000000 STAGE    0003 R 000000 SUPERK  0003  000003 TO      0012 C 000000 UP      0006  000010 WIDTH
    0007 L 000017 YESMTX  0007 L 000024 YESPCH  0007 L 000020 YESRAW  0007 L 000025 YESPLP  0007 L 000026 YESSRL
    0007 L 000021 YESSRP  0003 L 000070 YFSZOH  0006  000020 YTRC    0003  000002 ZM      0003  000001 ZT
    0003  000005 ZTVAL

6-43
```

```
00104                 SUBROUTINE CSOLVE(*)                                    1403    000000
00105                 COMMON/KEEP5/ SUPERK,ZT,ZM,YD,NZT,XTVAL(50),ITHZT,YESZOH,BOTH,  1404    000000
00106            3    1            MODIFY,GPRINT                              1405    000000
00107            4     LOGICAL      YESZOH,BOTH,MODIFY,GPRINT                  1407    000000
00108            5     COMMON/KEEP7/ RRN(75),RIN(75),RND(75),RID(75),FNPOL(76),  1408    000000
00109            6    1            FDPOL(76),NDGN,NDGD,NNCOEF,NDCOEF,NZN,NZD  1409    000000
00110            7     COMMON/KEEP10/NEQ,NE,NCOF,R(1000),JC(1000),ND(1000),      493    000000
00111            8    1            LL(1000),LOCPOL(60,60),EV(150),            494    000000
00112            9     COMMON/KEEP15/REGION(4),WIDTH(4),SPACE(4),YINC(4),               000000
00113           10    1            DA(4),DUM(4),P36(36),IDR(7),IDG(7),CPSLBL(10),     749    000000
00114           11    2            DCBLBL,DEGLBL,EMT(12),ENAT(12),E6,E7,LABEL1(20),   750    000000
00115           12    3            LABEL2(20),LABEL3(20)                      7510    000000
00116           13     COMPLEX      REGION                                     7520    000000
00117           14     COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,  2    000000
00118           15    1            IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,      25    000000
00119           16    2            YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,  26    000000
00120           17    3            YESSRL,NOMNAL,NOTYET                         27    000000
00121           18     LOGICAL      YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,  28    000000
00122           19    1            YESSRL,NOMNAL,NOTYET                         29    000000
00123           20     COMMON/KEEP18/ NNSCL,NDSCL                              1411    000000
00124           21     COMMON/KEEP19/LGAIN,LPHASE,LPOLFS,LSDRL,LZEROS                  000000
00125           22     LOGICAL      LGAIN,LPHASE,LPOLFS,LSDRL,LZEROS                   000000
00126           23     COMMON/CRUDS/ UP,DOWN,NEXPO,DEXPO,NG,NZ                 1412    000000
00127           24     INTEGER      DEXPO                                      1413    000000
00128           25     COMPLEX      UP,DOWN                                    1414    000000
00129           26 C                                                                   000000
00130           27 C                                                                   000000
00131           28 C  INITIALIZE FOR CONTINUOUS SYSTEM ANALYSIS                        000000
00132           29 C                                                                   000000
00133           30      LSDRL = .FALSE.                                                000000
00134           31      LGAIN = .FALSE.                                                000000
00135           32      LPHASE = .FALSE.                                               000001
00136           33      LPOLFS = .FALSE.                                               000002
00137           34      NOMNAL = .FALSE.                                               000003
00138           35      LZEROS = .FALSE.                                               000004
00139           36      REGION(1) = (0.,0.)                                            000005
00140           37      REGION(2) = (-10.,5.)                                          000007
00141           38      REGION(3) = (-50.,0.)                                          000011
00142           39      REGION(4) = (0.,-50.)                                          000013
00143           40 C                                                           1415    000013
00144           41 C                                                           1416    000013
00145           42 C  PRINT THE CONTINUOUS SYSTEM CHARACTERISTIC MATRIX         1417    000013
00146           43 C                                                           1418    000013
00147           44      CALL PRINTT($1000)                                             000015
00148           45 C                                                                   000015
00149           46 C                                                                   000015
00150           47 C  DETERMINE IF CONTINUOUS SYSTEM ANALYSIS CAN BE PERFORMED         000015
00151           48 C                                                                   000015
00152           49      IF (IOPEN.LE.0 .OR. IOPEN.GT.NEQ) GO TO 2000                   000020
00153           50      IF (JOPEN.LE.0 .OR. JOPEN.GT.NEQ) GO TO 2000                   000035
00154           51      IF (LOCPOL(IOPEN,JOPEN) .EQ. 0) GO TO 3000                     000052
00155           52      DO 12 I=1,NEQ                                                  000070
00156           53      IF (I.EQ.IOPEN) GO TO 12                                       000070
00157           54      NPT = LOCPOL(I,JOPEN)                                          000074
00158           55      IF (NPT.NE.0) GO TO 13                                         000076
00159           56   12 CONTINUE                                                       000103
```

```
00154    57          GO TO 4000                                                  000103
00155    58*       13 CONTINUE                                                   000105
00156    59*     C                                                    14200     000105
00157    60*     C                                                    14210     000105
00158    61*     C  CALCULATE THE CONTINUOUS SYSTEM OPEN LOOP POLES   14220     000105
00159    62*     C                                                    14230     000105
00160    63*         LPOLES = .TRUE.                                             000105
00161    64*         CALL POLES(*1000)                                           000106
00162    65*         LPOLES = .FALSE.                                            000111
00163    66*     C                                                    14250     000111
00164    67*     C                                                    14260     000111
00165    68*     C  EXPAND THE POLES INTO A POLYNOMIAL                14270     000111
00166    69*     C                                                    14280     000111
00167    70*         CALL FORM(NDGD,RRD,RID,EDPOL,NDCOEF,NZD,NDSCL)   14290     000112
00168    71*     C                                                    14300     000112
00169    72*     C                                                    14310     000112
00170    73*     C  CALCULATE THE CONTINUOUS SYSTEM OPEN LOOP ZEROS   14320     000112
00171    74*     C                                                    14330     000112
00172    75*         LZEROS = .TRUE.                                             000123
00173    76*         CALL ZEROS(*1000)                                           000125
00174    77*         LZEROS = .FALSE.                                            000130
00175    78*     C                                                    14350     000130
00176    79*     C                                                    14360     000130
00177    80*     C  EXPAND THE ZEROS INTO A POLYNOMIAL                14370     000130
00178    81*     C                                                    14380     000130
00179    82*         CALL FORM(NDGN,RRN,RIN,ENPOL,NNCOEF,NZN,NNSCL)   14390     000131
00180    83*     C                                                    14400     000131
00181    84*     C                                                    14410     000131
00182    85*     C  CALCULATE THE CONTINUOUS SYSTEM CLOSED LOOP POLES 14420     000131
00183    86*     C                                                    14430     000131
00184    87*         NOMNAL = .TRUE.                                             000142
00185    88*         CALL CLPOLE(*1000)                                          000144
00186    89*         NOMNAL = .FALSE.                                            000147
00187    90*     C                                                    14450     000147
00188    91*     C                                                    14460     000147
00189    92*     C  CALCULATE THE OPEN LOOP TRANSFER FUNCTION LEADING COEFFICIENT 14470 000147
00190    93*     C                                                    14480     000147
00191    94*         POWER = 1.0                                      14490     000150
00192    95*         IF (NEXPO.NE.DEXPO) POWER = 10.0**(NEXPO-DEXPO)  14500     000152
00193    96*         SUPERK = (UP/DOWN)*POWER                         14510     000166
00194    97*         RETURN                                           14520     000212
00195    98*     C                                                    14530     000212
00196    99*     C                                                    14540     000212
00197   100*     C ERROR RETURN                                       14550     000212
00198   101*     C                                                    14560     000212
00199   102*  1000 CONTINUE                                           14570     000216
00200   103*         RETURN 1                                         14580     000216
00201   104*     C                                                              000216
00202   105*     C                                                              000216
00203   106*     C  SAMPLING DEVICE LOCATION IS OUT OF RANGE                    000216
00204   107*     C                                                              000216
00205   108*  2000 CONTINUE                                                     000222
00206   109*         KODE = 103                                                 000222
00207   110*         RETURN 1                                                   000223
00208   111*     C                                                              000223
00209   112*     C                                                              000223
00210   113*     C  MATRIX ILL-DEFINED FOR OPEN LOOP ZEROS                      000223
```

```
00202    114*      C                                                    000223
00203    115*    3000 CONTINUE                                          000230
00204    116*         KODE = 104                                        000230
00205    117*         RETURN 1                                          000231
00206    118*      C                                                    000231
00207    119*      C                                                    000231
00208    120*      C  MATRIX ILL-DEFINED FOR OPEN LOOP POLES            000231
00209    121*      C                                                    000231
00210    122*    4000 CONTINUE                                          000236
00211    123*         KODE = 105                                        000236
00212    124*         RETURN 1                                          000237
00213    125*         END                                    1459.0     000267
```

END OF COMPILATION: NO DIAGNOSTICS.

SUBROUTINE DATA ENTRY POINT 000554

STORAGE USED: CODE(1) 000569; DATA(0) 000411; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003	KEEP1	000026
0004	KEEP2	000047
0005	KEEP3	000102
0006	KEEP4	000263
0007	KEEP5	000074
0010	KEEP6	000134
0011	KEEP9	000705
0012	KEEP16	000031
0013	CRUD2	000115
0014	PLT	000012

EXTERNAL REFERENCES (BLOCK, NAME)

0015	RESET
0016	INPNYQ
0017	INPRL
0020	INPEST
0021	INPMTX
0022	NROUS
0023	NIO3S
0024	NIO2S
0025	NIO1S
0026	NERR4S
0027	NERR3S

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001	000000	1DL	0001	000472	1000L	0001	000121	176G	0001	000020	20L	0001	000476	2000L	
0001	000263	30L	0001	000504	3000L	0001	000316	35L	0001	000342	40L	0001	000512	4000L	
0001	000340	45L	0001	000401	5DL	0000	000010	5DDF	0001	000520	5000L	0000	000011	501F	
0000	000015	502F	0000	000017	503F	0000	000021	504F	0001	000407	55L	0001	000424	60L	
0001	000526	6000L	0001	000534	9000L	0013	000024	ANORM	0013	000000	BCD	0007 L	000071	BOTH	
0006	000212	BY	0010	000064	CARD	0006	000166	D	0012 L	000022	DEBUG	0004 D	000002	DHFILT	
0014	000007	DIF1	0014	000010	DIF2	0005	000063	DP	0011 C	000226	EA	0006	000004	GAIN	
0007	000073	GPRINT	0006	000151	GSYM	0004	000037	HACC	0004	000025	HAST	0004	000035	HATT	
0004 D	000000	HBLANK	0004 R	000004	HBLK	0004	000034	HCCW	0004	000033	HCW	0004	000032	HDEC	
0004	000027	HDOT	0004 R	000006	HESTI	0004	000043	HFGD	0004	000044	HFGN	0004	000040	HFPD	
0004	000045	HFPDD	0004	000042	HFPDN	0004	000041	HFPN	0004	000010	HGENE	0004	000031	HINC	
0004 R	000005	HKEY	0004 R	000007	HMATR	0004	000014	HNEN	0004	000046	HNOMI	0004 R	000015	HNYQU	
0004	000023	HO	0004	000026	HPLUS	0004	000036	HRATE	0004	000011	HRAW	0004	000012	HRETA	
0004	000017	HROLL	0004 R	000016	HROOT	0004	000013	HSTAN	0004	000030	HSTAR	0004	000020	HSIC	
0004	000021	HS2	0004	000022	HS4B	0004	000024	HX	0000 I	000007	I	0014	000011	ICK	
0014	000004	ICT	0013	000076	II	0004	000027	IJP3	0012 I	000010	IOPEN	0014	000003	ISW	
0007	000067	ITHZT	0013	000101	JJ	0012 I	000011	JOPEN	0011	000572	KD	0013	000105	KK	

```
0012  "  000000  KODE      0010    000010  LABEL     0012  L 000023  LFLT      0005  R 000051  MAX       0008    000000  MAXIT
0008  R  000037  MDN       0007  L 000022  MODIFY    0003    000001  MXEIG     0003    000024  MXEIGT    0003    000028  MXESF
0003     000002  MXFRM     0003    000003  MXNBM     0003    000004  MXNCOF    0003    000005  MXNCT     0003    000026  MXNCV
0003     000008  MXNE      0003    000007  MXNEQ     0003    000010  MXNFI     0003    000011  MXNG      0003    000012  MXNPH
0003     000013  MXNPP     0003    000014  MXNQPT    0003    000015  MXNSM     0003    000016  MXNSP     0003    000017  MXNTM
0003     000020  MXNV      0003  I 000021  MXNZT     0003    000022  MXPOLY    0011  I 000453  NA        0010  R 000062  NAME
0011     000454  NEIG      0005  I 000000  NFI       0013    000074  NFILT     0006  I 000003  NGAIN     0006    000154  NGR
0011     000457  NI        0014  L 000001  NICPLT    0013  L 000027  NOMNAL    0012  L 000030  NOTYET    0014    000002  NP
0006  I  000066  NPHASE    0011    000456  NR        0012  I 000002  NRCLPL    0006  I 000153  NRLFR     0012  I 000003  NRPOLE
0012  I  000004  NRZERO    0013    000071  NTMPO     0013    000072  NTMPOC    0013    000073  NTMPIC    0012    000005  NXB
0012     000006  NXN       0012    000007  NXR       0007  I 000004  NZT       0000  R 000006  OPTBUG    0013  R 000020  OPTINP
0000  R  000004  OPTPCH     0000  R 000005  OPTPNT    0000  R 000003  OPTTP     0000  R 000002  OPTTRL    0013    000003  OPTTYP
0000  R  000000  OPTZ        0000  R 000001  OPTZOH    0013  R 000021  OPT1      0013  R 000022  OPT2      0013  R 000023  OPT3
0005  R  000076  PB        0013    000111  PC        0012    000014  PCPL      0005    000025  PCT       0012    000015  PFAC
0006  R  000000  PG        0006    000067  PHASE     0005  R 000075  PN        0014  R 000000  PNI       0012  R 000013  PNOM
0006  R  000001  PP        0006  R 000002  PPLT      0012    000016  PSLOSH    0006    000152  PSYM      0012  R 000012  PVAR
0005  R  000077  P180      0013  R 000017  REQEST    0011  C 000000  ROOT      0006    000200  RX        0012    000000  STAGE
0005  L  000101  STNDRD    0005    000013  STP       0005    000001  STR       0007    000000  SUPERK    0014    000006  S360
0007  R  000003  TD        0013    000006  TEMP      0013    000025  TEMP0     0013    000045  TEMP1     0010  R 000000  TITLE
0010  R  000024  TITLE1    0010  R 000036  TITLE2    0010  R 000050  TITLE3    0014    000005  T360      0013    000065  VAL
0013     000011  VFILT     0012  L 000017  YESMTX    0005  L 000100  YESNYQ    0012  L 000024  YESPCH    0012  L 000020  YESRAW
0006  L  000262  YESRL     0012  L 000025  YESRLP    0012  L 000026  YESSRL    0012  L 000021  YESSRP    0007  L 000070  YESZOH
0007     000002  ZM        0007    000001  ZT        0007  R 000005  ZTVAL
```

```
00101    1*          SUBROUTINE DATA(*,*,*)                                                    14600    000000
00103    2*          COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,           14610    000000
00103    3*                1       MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,          14620    000000
00104    4*                2       MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV                      14630    000000
00104    5*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,           14640    000000
00104    6*                1       HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS48,         14650    000000
00104    7*                2       HD,HX,HASI,HPLUS,HDOI,HSTAR,HINC,HDEC,HCW,HCCW,           14660    000000
00104    8*                3       HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,          14670    000000
00104    9*                4       HNOM/                                                     14680    000000
00106   10*          DOUBLE PRECISION HBLANK,DHFILT                                                   000000
00106   11*          COMMON/KEEP3/ NFI,STR(10),STP(10),PCT(10),MIN(10),MAX(10),DP(10),         14700    000000
00106   12*                1       PN,PB,P180,YESNYQ,STNDRD                                  14710    000000
00107   13*          REAL          MIN,MAX                                                   14720    000000
00111   14*          LOGICAL       YESNYQ,STNDRD                                             14730    000000
00111   15*          COMMON/KEEP4/ PG,PP,PPLT,NGAIN,GAIN(50),NPHASE,PHASE(50),GSYM,          14740    000000
00111   16*                1       PSYM,NRLFR,NGR(10),DD(10),RX(10),BY(4,10),YESRL           14750    000000
00112   17*          LOGICAL       YESRL                                                     14760    000000
00113   18*          COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,          14770    000000
00113   19*                1       MODIFY,GPRINT                                             14780    000000
00114   20*          LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                                 14800    000000
00115   21*          COMMON/KEEP6/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),       14810    000000
00115   22*                1       CARD(20),LABEL(20)                                        14820    000000
00115   23*          REAL          NAME                                                      14830    000000
00117   24*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)                  14840    000000
00120   25*          COMPLEX       ROOT,EA                                                            000000
00121   26*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,              14870    000000
00121   27*                1       IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                   14880    000000
00121   28*                2       YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,            14890    000000
00121   29*                3       YESSRL,NOMNAL,NOTYET                                      14900    000000
00121   30*          LOGICAL       YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,            14910    000000
```

```
00122    31*          1                YESSRL,NOMNAL,NOTYET                                    1492D    000000
00123    32*          COMMON/CRUD2/ BCD(3),OPTTYP(3),TEMP(3),VFILT(6),REQEST,OPTINP,          1493D    000000
00124    33*          1                OPT1,OPT2,OPT3,ANORM,TEMP0(16),TEMP1(16),VAL(4),       1494D    000000
00125    34*          2                NIMPO,NIMPOC,NIMPIC,NFILT,II(4),JJ(4),KK(4),PC(4)      1495D    000000
00126    35*           COMMON /PLT/ PNI,NICPLT,NP,ISW,,ICT,T360,S360,DIFI,DIF2,ICK                     000000
00127    36*           LOGICAL NICPLT                                                                  000000
00128    37*     C                                                                            1497D    000000
00129    38*     C                                                                            1498D    000000
00130    39*     C  READ CASE TITLE AND CHECK FOR KEY WORD                                    1499D    000000
00131    40*     C                                                                            1500D    000000
00132    41*        10 READ(5,500,END=9000) TITLE                                             1501D    000000
00133    42*       500 FORMAT(20A4)                                                           1502D    000010
00134   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00134    43*           IF (TITLE(1).NE.HKEY) GO TO 20                                        1503D    000010
00135    44*           CALL RESET                                                            1504D    000014
00136    45*           GO TO 10                                                              1505D    000016
00137    46*     C                                                                           1506D    000016
00138    47*     C                                                                           1507D    000016
00139    48*     C  READ TRANSPORT LAG, OPEN LOOP NAME, LOCATION OF SAMPLING DEVICE,         1508D    000016
00140    49*     C  NUMBER OF SAMPLE RATES, MAXIMUM VALUES FOR CONTINUOUS CASE ROOTING,      1509D    000016
00141    50*     C  ZERO ORDER HOLD OPIONS, SAMPLE RATE ROOT LOCUS OPTION,                   1510D    000016
00142    51*     C  PUNCH CONTINUOUS MATRIX OPTION, PRINT OPTIONS, SAMPLE RATES              1511D    000016
00143    52*     C                                                                           1512D    000016
00144    53*        20 CONTINUE                                                              1513D    000020
00145    54*           READ(5,500) TITLE1,TITLE2                                             1514D    000020
00146    55*           READ(5,501) TO,NAME,IOPEN,JOPEN,NZT,NRPOLE,NRZERO,NRCLPL              1515D    000032
00155    56*       501 FORMAT(E12.4,8X,2A4,2X,6I5)                                           1516D    000052
00156    57*           TITLE3(8) = NAME(1)                                                   1517D    000052
00157    58*           TITLE3(9) = NAME(2)                                                   1518D    000054
00160    59*           IF (NZT.LT.1 .OR. NZT.GT.MXNZT) GO TO 2000                            1519D    000056
00162    60*           READ(5,502) OPTZ,OPTZOH,OPTTRL,OPTTP,OPTPCH,OPTPNT,OPTBUG                      000074
00173    61*       502 FORMAT (7(A1,9X))                                                             000112
00174    62*           READ(5,503) (ZTVAL(I),I=1,NZT)                                        1522D    000112
00202    63*       503 FORMAT(6E12.5)                                                        1523D    000124
00202    64*     C                                                                           1524D    000124
00202    65*     C                                                                           1525D    000124
00202    66*     C  PROCESS THE OPTIONS                                                      1526D    000124
00202    67*     C                                                                           1527D    000124
00203    68*           MODIFY = .FALSE.                                                      1528D    000124
00204   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00204    69*           IF (TO.NE.0.0) MODIFY=.TRUE.                                          1529D    000125
00206    70*           DEBUG = .FALSE.                                                       1530D    000131
00207   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00207    71*           IF (OPTBUG.NE.HBLK) DEBUG=.TRUE.                                      1531D    000132
00211    72*           GPRINT = .FALSE.                                                      1532D    000140
00212   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00212    73*           IF (OPTBUG.NE.HBLK .OR. OPTPNT.NE.HBLK) GPRINT=.TRUE.                 1533D    000141
00214    74*           YESPCH = .FALSE.                                                      1534D    000157
00215   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00215    75*           IF (OPTPCH.NE.HBLK) YESPCH=.TRUE.                                     1535D    000160
00217    76*           YESZOH = .FALSE.                                                      1536D    000166
00220    77*           BOTH = .FALSE.                                                        1537D    000167
00221   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00221    78*           IF (OPTZ.EQ.HBLK .AND. OPTZOH.EQ.HBLK) GO TO 3000                     1538D    000170
00223   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00223    79*           IF (OPTZ.EQ.HBLK .AND. OPTZOH.NE.HBLK) YESZOH=.TRUE.                  1539D    000204
00225   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
```

```
00225      *                IF (OPTZ.NE.HBLK .AND. OPTZOH.NE.HBLK) BOTH=.TRUE.                1540  000222
00227     81*               YESSRL = .FALSE.                                                  1541  000240
00230     82*               YESSRP = .FALSE.                                                  1542  000241
00231    *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00231     83*               IF (OPTTRL.EQ.HBLK) GO TO 30                                      1543  000242
00233    *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00233     84*               IF (OPTZOH.EQ.HBLK) GO TO 6000                                          000246
00235     85*               YESSRL = .TRUE.                                                   1544  000252
00236    *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00236     86*               IF (OPITP.NE.HBLK) YESSRP=.TRUE.                                  1545  000254
00240     87*            30 CONTINUE                                                          1546  000263
00240     88*     C                                                                           1547  000263
00240     89*     C                                                                           1548  000263
00240     90*     C  CONSTRUCT THE THIRD PLOT TITLE                                           1549  000263
00240     91*     C                                                                           1550  000263
00241     92*               TITLE3(15) = NAME(1)                                              1551  000263
00242     93*               TITLE3(16) = NAME(2)                                              1552  000264
00242     94*     C                                                                           1553  000264
00242     95*     C                                                                           1554  000264
00242     96*     C  READ NEXT REQUEST                                                        1555  000264
00242     97*     C                                                                           1556  000264
00243     98*               READ(5,504) REQEST,OPTINP,OPT1,OPT2,OPT3,PNI                            000266
00253     99*            504 FORMAT(A4,16X,A4,6X,4(A1,9X))                                          000301
00253    100*     C                                                                           1559  000301
00253    101*     C                                                                           1560  000301
00253    102*     C  INCORPORATE NYQUIST REQUEST                                              1561  000301
00253    103*     C                                                                           1562  000301
00254    *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00254    104*               IF (REQEST.EQ.HNYQU) GO TO 35                                     1563  000301
00256    105*               YESNYQ = .FALSE.                                                  1564  000305
00257    106*               STNDRD = .FALSE.                                                  1565  000306
00260    107*               PN = HBLK                                                         1566  000307
00261    108*               PB = HBLK                                                         1567  000311
00262    109*               PIBD = HBLK                                                       1568  000312
00263    110*               NFI = 0                                                           1569  000313
00264    111*               GO TO 40                                                          1570  000314
00265    112*            35 CONTINUE                                                          1571  000316
00266    113*               NICPLT = .FALSE.                                                        000316
00267    *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00267    114*               IF (PNI .NE. HBLK) NICPLT = .TRUE.                                      000316
00271    115*               CALL INPNYQ(*1000)                                                      000324
00271    116*     C                                                                           1573  000324
00271    117*     C                                                                           1574  000324
00271    118*     C  READ NEXT REQUEST                                                        1575  000324
00271    119*     C                                                                           1576  000324
00272    120*               READ(5,504) REQEST,OPTINP,OPT1,OPT2,OPT3                          1577  000327
00272    121*     C                                                                           1578  000327
00272    122*     C                                                                           1579  000327
00272    123*     C  INCORPORATE ROOT LOCUS REQUEST                                           1580  000327
00272    124*     C                                                                           1581  000327
00301    125*            40 CONTINUE                                                          1582  000342
00302    *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00302    126*               IF (REQEST.EQ.HROOT) GO TO 45                                     1583  000342
00304    127*               YESRL = .FALSE.                                                   1584  000345
00305    128*               YESRLP = .FALSE.                                                  1585  000346
00306    129*               PG = HBLK                                                         1586  000347
00307    130*               PP = HBLK                                                         1587  000351
```

```
00310     131*            PPLT = HBLK                                          1588₀    000352
00311     132*            NGAIN = 0                                            1589₀    000353
00312     133*            NPHASE = 0                                           1590₀    000354
00313     134*            NRLFR = 0                                            1591₀    000355
00314     135*            GO TO 50                                             1592₀    000356
00315     136*        45 CONTINUE                                              1593₀    000360
00316  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00316     137*            IF (OPTZOH.EQ.HBLK) GO TO 6000                               000360
00320     138*            CALL INPRL($1000)                                            000363
00320     139*      C                                                          1595₀    000363
00320     140*      C                                                          1596₀    000363
00320     141*      C  READ NEXT REQUEST                                       1597₀    000363
00320     142*      C                                                          1598₀    000363
00321     143*            READ(5,504) REQEST,OPTINP,OPT1,OPT2,OPT3             1599₀    000366
00321     144*      C                                                          1600₀    000366
00321     145*      C                                                          1601₀    000366
00321     146*      C  INCORPORATE CONTINUOUS CASE ESTIMATES                   1602₀    000366
00321     147*      C                                                          1603₀    000366
00330     148*           50 CONTINUE                                           1604₀    000401
00331  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00331     149*            IF (REQEST.EQ.HEST) GO TO 55                         1605₀    000401
00333     150*            NA = 0                                               1606₀    000404
00334     151*            GO TO 60                                             1607₀    000405
00335     152*        55 CONTINUE                                              1608₀    000407
00336     153*            CALL INPEST($1000)                                           000407
00337     154*      C                                                          1610₀    000407
00336     155*      C                                                          1611₀    000407
00336     156*      C  READ NEXT REQUEST                                       1612₀    000407
00336     157*      C                                                          1613₀    000407
00337     158*            READ(5,504) REQEST,OPTINP,OPT1,OPT2,OPT3             1614₀    000411
00337     159*      C                                                          1615₀    000411
00337     160*      C                                                          1616₀    000411
00337     161*      C  INCORPORATE CONTINUOUS SYSTEM CHARACTERISTIC MATRIX     1617₀    000411
00337     162*      C                                                          1618₀    000411
00346     163*           60 CONTINUE                                           1619₀    000424
00347  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00347     164*            IF (REQEST.NE.HMATR) GO TO 4000                      1620₀    000424
00351     165*            CALL INPMTX($1000)                                           000427
00351     166*      C                                                          1622₀    000427
00351     167*      C                                                          1623₀    000427
00351     168*      C  DETERMINE WHETHER TO EXECUTE THE NOMINAL MATRIX         1624₀    000427
00351     169*      C                                                          1625₀    000427
00352  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00352     170*            IF (PNOM.NE.HBLK .AND. PVAR.NE.HBLK) RETURN 3        1626₀    000432
00353  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00354     171*            IF (PNOM.NE.HBLK .AND. PVAR.EQ.HBLK) GO TO 5000      1627₀    000452
00356     172*            RETURN                                               1628₀    000466
00356     173*      C                                                          1629₀    000466
00356     174*      C                                                          1630₀    000466
00356     175*      C  DATA INPUT ERROR                                        1631₀    000466
00356     176*      C                                                          1632₀    000466
00357     177*       1000 CONTINUE                                             1633₀    000472
00360     178*            RETURN 1                                             1634₀    000472
00360     179*      C                                                          1635₀    000472
00360     180*      C                                                          1636₀    000472
00360     181*      C  NUMBER OF SAMPLE RATES IS OUT OF RANGE                  1637₀    000472
00361     182*      C                                                          1638₀    000472
```

```
00361   183*      2000 CONTINUE                                          1639₀   000476
00362   184*           KODE = 11                                         1640₀   000476
00363   185*           RETURN 1                                          1641₀   000477
00363   186*      C                                                      1642₀   000477
00363   187*      C                                                      1643₀   000477
00363   188*      C  NEITHER Z TRANSFORMATION WITH OR WITHOUT ZERO ORDER HOLD REQUESTED  1644₀   000477
00363   189*      C                                                      1645₀   000477
00364   190*      3000 CONTINUE                                          1646₀   000504
00365   191*           KODE = 12                                         1647₀   000504
00366   192*           RETURN 1                                          1648₀   000505
00366   193*      C                                                      1649₀   000505
00366   194*      C                                                      1650₀   000505
00366   195*      C  ILLEGAL REQUEST                                     1651₀   000505
00367   196*      C                                                      1652₀   000505
00367   197*      4000 CONTINUE                                          1653₀   000512
00370   198*           KODE = 13                                         1654₀   000512
00371   199*           RETURN 1                                          1655₀   000513
00371   200*      C                                                      1656₀   000513
00371   201*      C                                                      1657₀   000513
00371   202*      C  USER SELECTED NOT TO EXECUTE NOMINAL MATRIX NOR INPUT VARIATIONS  1658₀   000513
00371   203*      C                                                      1659₀   000513
00372   204*      5000 CONTINUE                                          1660₀   000520
00373   205*           KODE = 14                                         1661₀   000520
00374   206*           RETURN 1                                          1662₀   000521
00374   207*      C                                                              000521
00374   208*      C                                                              000521
00374   209*      C  CAN NOT REQUEST ROOT LOCUS FOR SYSTEMS WITHOUT A ZERO ORDER HOLD  000521
00374   210*      C                                                              000521
00375   211*      6000 CONTINUE                                                  000526
00376   212*           KODE = 108                                                000526
00377   213*           RETURN 1                                                  000527
00377   214*      C                                                      1663₀   000527
00377   215*      C                                                      1664₀   000527
00377   216*      C  NO MORE DATA CARDS                                  1665₀   000527
00377   217*      C                                                      1666₀   000527
00400   218*      9000 CONTINUE                                          1667₀   000534
00401   219*           RETURN 2                                          1668₀   000534
00402   220*           END                                               1669₀   000563

      END OF COMPILATION:        19 DIAGNOSTICS.
```

```
?FOR,USM F.DB,F.DB
FOR SEIX-05/23/74-08:28:42-(2.3)

FUNCTION DB          ENTRY POINT 000025

STORAGE USED: CODE(1) 000031; DATA(0) 000007; BLANK COMMON(2) 000000

EXTERNAL REFERENCES (BLOCK, NAME)

    1003    ALOG10
    1004    NERR3$

STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    1001    000010 IL       0000 R 000000 DB        0000    000002 INJP$

00101       1*          FUNCTION DB(A)                          1670₀   000000
00102       2*          IF (A.GT.0.) GO TO 1                    1671₀   000000
00103       3*          DB = 0.                                 1672₀   000003
00104       4*          RETURN                                  1673₀   000004
00107       5*        1 DB = 20.*ALOG10(A)                      1674₀   000010
00111       6*          RETURN                                  1675₀   000014
00111       7*          END                                     1676₀   000030

      END OF COMPILATION:  NO DIAGNOSTICS.
```

```
@FOR,USW F,DECIDE,F,DECIDE
FOR SEIX-05/23/74-08:28:50 (3,4)

     SUBROUTINE DECIDE     ENTRY POINT 000132

     STORAGE USED: CODE(1) 000134; DATA(0) 000007; BLANK COMMON(2) 000000

     COMMON BLOCKS:

     1003   KEEP9  000705
     1004   CRUD3  016115

     EXTERNAL REFERENCES (BLOCK, NAME)

     1005   CDABV
     1006   COVS
     1007   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     1001   000110 IOL      0001   000117 ZOL      0004 L 016113 AUTO     0004 C 016072 B        0005 R 000000 CDABV
     1004 L 016112 CONJ     0004 C 000000 CU       0004 L 016110 DONE     0003 C 000226 EA       0004 C 016040 FPRO
     1004 C 016042 FPR1     0004 C 016044 FPR2     0004 C 016046 FRO      0004 C 016050 FR1      0004 C 016052 FR2
     1000   000001 INJ#$    0003   000572 KD       0003   000455 NA       0004   016102 NCT      0003   000454 NEIG
     1004   016107 NEST     0004   016076 NFPO     0004   016077 NFP1     0004   016100 NFP2     0003   000457 NI
     1004   016101 NITER    0004   016104 NKODE    0003   000456 NR       0004   016103 NREG     0004   016105 NSTART
     1004   016106 NTIME    0004 C 016054 PRO      0004 C 016056 PR1      0004 C 016060 PR2      0004 L 016114 REGSEL
     1004 L 016111 RESTRT   0003 C 000000 ROOT     0004 C 016062 RO       0004 C 016064 R1       0004 C 016066 R2
     1004 C 016070 R3       0004 C 016074 U

6-54

00101     1*          SUBROUTINE DECIDE                                                    1677  000000
00102     2*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)               1678  000000
00103     3*          COMPLEX  ROOT,EA                                                           000000
00104     4*          COMMON/CRUD3/ CU(60,60),FPRO,FPR1,FPR2,FRO,FR1,FR2,PRO,PR1,PR2,      1681  000000
00105     5*         1            RO,R1,R2,R3,B,U,NFPO,NFP1,NFP2,NITER,NCT,NREG,NKODE,  1682  000000
00106     6*         2            NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL        1683  000000
00107     7*          COMPLEX  CU,FPRO,FPR1,FPR2,FRO,FR1,FR2,PRO,PR1,PR2,                        000000
00108     8*         1            RO,R1,R2,R3,B,U                                       1685  000000
00109     9*          LOGICAL  DONE,RESTRT,CONJ,AUTO,REGSEL                             1686  000000
00107    10*     C                                                                      1688  000000
00107    11*     C                                                                      1689  000000
00107    12*     C***      DETERMINE IF ALL ROOTS HAVE BEEN FOUND                       1690  000000
00107    13*     C                                                                      1691  000000
00110    14*          DONE=.FALSE.                                                      1692  000000
00111    15*     C                                                                      1694  000000
00111    16*     C     TEST NUMBER ONE                                                  1695  000000
00113    17*     C                                                                      1696  000000
00111  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00111    18*          IF (CDABV(FPRO) .EQ. 0.) GO TO 10                                       000000
00113  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
```

```
00113      19*            IF (CDABV(FPR1) .EQ. 0.) GO TO 20                                    000005
00115  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00115      20*            IF (CDABV(FPR2) .EQ. 0.0) RETURN                                     000012
00117      21*            IF (CDABV((FPR0-FPR1)/FPR0) .GT. 1.E-3) RETURN                       000022
00117      22*      C                                                          17000           000022
00117      23*      C     TEST NUMBER TWO                                      17010           000022
00117      24*      C                                                          17020           000022
00121      25*            IF (CDABV((FPR0-FPR2)/FPR0) .GT. 1.E-3) RETURN                       000052
00121      26*      C                                                          17120           000052
00121      27*      C     BOTH TESTS PASSED -** ALL ROOTS HAVE BEEN FOUND      17130           000052
00121      28*      C                                                          17140           000052
00123      29*            DONE=.TRUE.                                          17150           000102
00123      30*            RETURN                                               17160           000104
00125      31*       10   CONTINUE                                                             000110
00125      32*            R2 = R0                                                              000110
00127      33*            FPR2 = FPR0                                                          000111
00131      34*            RETURN                                                               000113
00131      35*       20   CONTINUE                                                             000117
00133      36*            R2 = R1                                                              000117
00133      37*            FPR2 = FPR1                                                          000120
00131      38*            RETURN                                                               000122
00135      39*            END                                                  17170           000133

      END OF COMPILATION:        3 DIAGNOSTICS.
```

```
@FOR,USN F.DELZOH,F.DELZOH
FOR (E1X=05/23/74=08:29:01 (2.3)

SUBROUTINE DELZOH     ENTRY POINT 000033

STORAGE USED: CODE(1) 000040; DATA(0) 000011; BLANK COMMON(2) 000000

COMMON BLOCKS:

1003   KEEP7  000712

EXTERNAL REFERENCES (BLOCK, NAME)

1004   NERR3$

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

1001  000015 1106      0003 R 000570 FDPOL     0003  000454 FNPOL     0000 I 000000 I          0000  000002 INJR$
1003 I 000707 NDCOEF   0003 I 000705 NDGD      0003  000704 NDGN      0003  000706 NNCOEF      0003 I 000711 NZD
1003  000710 NZN       0003  000341 RID       0003  000113 RIN       0003  000226 RRD         0003  000000 RRN


00101     1*          SUBROUTINE DELZOH                                          1718₀  000000
00103     2*          COMMON/KEEP7/ RRN(75),RIN(75),RRD(75),RID(75),FNPOL(76),   1719₀  000000
00103     3*         1          FDPOL(76),NDGN,NDGD,NNCOEF,NDCOEF,NZN,NZD        1720₀  000000
00103     4*     C                                                              1722₀  000000
00103     5*     C                                                              1723₀  000000
00103     6*     C  DELETE THE ZERO ORDER HOLD DEVICE                           1724₀  000000
00103     7*     C                                                              1725₀  000000
00104     8*          NZD = NZD - 1                                             1726₀  000000
00105     9*          NDGD = NDGD - 1                                           1727₀  000002
00106    10*          NDCOEF = NDCOEF - 1                                       1728₀  000005
00107    11*          DO 100 I=1,NDCOEF                                         1729₀  000010
00112    12*      100 FDPOL(I) = FDPOL(I+1)                                     1730₀  000015
00114    13*          FDPOL(NDCOEF+1) = 0.0                                     1731₀  000017
00115    14*          RETURN                                                    1732₀  000020
00115    15*          END                                                       1733₀  000037

      END OF COMPILATION:     NO DIAGNOSTICS.
```

```
@FOR,USW F,DETCS,F,DETCS
FOR SEIX-05/23/74-08:29:19 (2.3)

      SUBROUTINE DETCS      ENTRY POINT 000342

      STORAGE USED: CODE(1) 000402; DATA(0) 000061; BLANK COMMON(2) 000008

      COMMON BLOCKS:

      0003   KEEP10 021620
      0004   KEEP14 000031
      0005   CRUD3  016115

      EXTERNAL REFERENCES (BLOCK, NAME)

      0006   FRMTX
      0007   CDV$
      0010   NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000065 1L       0001   000121 10L      0001   000225 1000L    0001   000022 125G     0001   000046 133G
      0001   000233 150DL    0001   000113 153G     0001   000145 164G     0001   000200 173G     0001   000265 1750L
      0001   000316 2000L    0001   000246 207G     0000 C 000016 ARG      0005 L 016113 AUTO     0005 C 016072 B
      0000 R 000014 C        0005 L 016112 CONJ     0005 C 000000 CU       0004   000025 DEG      0000 C 000000 DIV
      0005 L 016110 DONE     0004   000012 D0       0004   000013 D1       0004   000014 D2       0004   000015 D3
      0004   000016 D4       0000 R 000020 E        0003   016664 EV       0000 R 000012 E13      0004 C 000010 FIFTEN
      0004   000017 FIFTY    0004 C 000004 FOUR     0005 C 016040 FPR0     0005 C 016042 FPR1     0005 C 016044 FPR2
      0005 C 016046 FR0      0005 C 016050 FR1      0005 C 016052 FR2      0004 C 000000 HALF     0000 I 000015 I
      0000   000030 INJP$    0003   000004 IR       0000 I 000013 J        0003   001754 JC       0000 I 000007 K
      0000 I 000010 KPI      0003   005674 LL       0003   007644 LOCPOL   0003   000003 NCOF     0005   016102 NCT
      0003   003724 ND       0003   000000 NDEG     0003   000002 NE       0003 I 000001 NEQ      0005   016107 NEST
      0005   016076 NFP0     0005   016077 NFP1     0005   016100 NFP2     0005   016101 NITER    0005   016104 NKODE
      0000 I 000006 NM1      0005   016103 NREG     0000 I 000011 NROW     0005   016105 NSTART   0005   016106 NTIME
      0004   000026 N1       0004   000027 N2       0004   000030 N3       0004 C 000002 ONE      0004   000020 PI
      0000 C 000022 PIVOT    0004   000021 PI2      0000 L 000004 PLUS     0005 C 016054 PR0      0005 C 016056 PR1
      0005 C 016060 PR2      0004   000024 RADDEG   0005 L 016114 REGSEL   0005 L 016111 RESTRT   0004   000023 RPI
      0005 C 016062 R0       0005 C 016064 R1       0005 C 016066 R2       0005 C 016070 R3       0004   000022 SMALL
      0000 C 000002 TEMP     0000 R 000014 TEI3     0000 R 000005 TWNTY    0004 C 000004 TWO      0005 C 016074 U

00101     1*        SUBROUTINE DETCS(X,EVAL,NSCALE)                                                  000000
00103     2*        COMPLEX      X,EVAL                                                              000000
00105     3*        COMMON/KEEP10/NDEG,NEQ,NE,NCOF,IR(1000),JC(1000),ND(1000),                1735D  000000
00107     4*              1      LL(1000),LOCPOL(60,60),EV(1500)                                   1736D  000000
00113     5*        COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,           1737D  000000
00115     6*              1      PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                                 1738D  000000
00117     7*        COMPLEX      HALF,ONE,TWO,FOUR,FIFTEN                                            000000
00107     8*        COMMON/CRUD3/ CU(60,60),FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,           1741D  000000
00107     9*              1      R0,R1,R2,R3,B,U,NFP0,NFP1,NFP2,NITER,NCT,NREG,NKODE,       1742D  000000
00107    10*              2      NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL             1743D  000000
```

```
00111    14*          COMMON /CW/FRR0,FRR1,FRR2,FR0,FR1,FR2,PR0,PR1,PR2,
00112    13*         1     R0,R1,R2,R3,B,U                                  17460   000000
00113    13*          LOGICAL        DONE,RESTRT,CONJ,AUTO,REGSEL           17470   000000
00114    14*          COMPLEX        ARG,DIV,PIVOT,TEMP                             000000
00115    15*          LOGICAL        PLUS                                   17490   000000
00116    16*          DIMENSION      C(2),E(2)                                      000000
00117    17*          EQUIVALENCE    (ARG,C(1)),(PIVOT,E(1))                17510   000000
00118    18*     C                                                          17520   000000
00119    19*     C                                                          17530   000000
00120    20*     C    ROOTING FOR CONTINUOUS SYSTEM                                 000000
00115    21*     C***     INITIALIZATION                                    17540   000000
00115    22*     C                                                          17550   000000
00116    23*          TWNTY = 1.0E20                                                000000
00117    24*          PLUS=.TRUE.                                           17580   000001
00120    25*          NM1 = NEQ - 1                                         17590   000003
00120    26*     C                                                          17600   000003
00121    27*     C                                                          17610   000003
00120    28*     C***     FORM CONSTANT MATRIX FOR LAMDA = X                17620   000003
00120    29*     C                                                          17630   000003
00121    30*          CALL FRMTX(X)                                         17640   000005
00122    31*          IF (NEQ.EQ.1) GO TO 1500                              17650   000010
00122    32*     C                                                          17660   000010
00122    33*     C                                                          17670   000010
00122    34*     C***     TRIANGULARIZATION OF SYSTEM                       17680   000010
00122    35*     C                                                          17690   000010
00124    36*          DO 1000 K=1,NM1                                       17700   000022
00127    37*          KP1=K+1                                               17710   000034
00127    38*     C                                                          17720   000034
00127    39*     C                                                          17730   000034
00127    40*     C***     SEARCH FOR NONZERO DIVISOR                        17740   000034
00127    41*     C                                                          17750   000034
00130    42*          NROW=K                                                17760   000037
00131    43*          E13=0.0                                               17770   000041
00132    44*          DO 1 J=K,NEQ                                          17780   000046
00135    45*          PIVOT=CU(J,K)                                         17790   000046
00135    46*          TE13 = ABS(E(1)) + ABS(E(2))                                  000050
00137    47*          IF (TE13.LE.E13) GO TO 1                              17810   000054
00141    48*          E13 = TE13                                            17820   000060
00142    49*          NROW=J                                                17830   000062
00143    50*        1 CONTINUE                                              17840   000067
00145   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00145    51*          IF(E13.EQ.0.0) RETURN                                 17850   000067
00147    52*          IF(NROW.EQ.K) GO TO 10                                17860   000074
00147    53*     C                                                          17870   000074
00147    54*     C                                                          17880   000074
00147    55*     C***     ROWS INTERCHANGED TO AVOID DIVISION BY ZERO        17890   000074
00147    56*     C                                                          17900   000074
00151    57*          PLUS=.NOT.PLUS                                        17910   000100
00152    58*          DO 2 J=K,NEQ                                          17920   000102
00155    59*          TEMP=CU(K,J)                                          17930   000113
00156    60*          CU(K,J)=CU(NROW,J)                                    17940   000114
00157    61*          CU(NROW,J)=TEMP                                       17950   000116
00160    62*        2 CONTINUE                                              17960   000121
00160    63*     C                                                          17970   000121
00161    64*     C                                                          17980   000121
00161    65*     C***     REDUCTION OF MATRIX TO UPPER TRIANGULAR FORM       17990   000121
00160    66*     C                                                          18000   000121
```

```
00162     67*          10 DIV=CU(K,K)                                          18010   000121
00163     68*             DO 1000 I=KP1,NEQ                                    18020   000122
00166     69*             PIVOT=CU(I,K)/DIV                                    18030   000151
00167  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00167     70*             IF (E(1).EQ.0.0 .AND. E(2).EQ.0.0) GO TO 1000                000160
00171     71*             CU(I,K) = (0.,0.)                                            000170
00172     72*             DO 100 J=KP1,NEQ                                     18060   000200
00175     73*         100 CU(I,J)=CU(I,J)-PIVOT*CU(K,J)                        18070   000200
00177     74*        1000 CONTINUE                                             18080   000233
00177     75*          C                                                       18090   000233
00177     76*          C                                                       18100   000233
00177     77*          C**** CALCULATE THE DETERMINANT                         18110   000233
00177     78*          C                                                       18120   000233
00202     79*        1500 CONTINUE                                             18130   000233
00203     80*             EVAL = (1.,0.)                                               000233
00205     81*             IF(.NOT.PLUS) EVAL=-EVAL                             18150   000234
00206     82*             DO 2000 I=1,NEQ                                      18160   000246
00211     83*             EVAL=EVAL*CU(I,I)                                    18170   000246
00212     84*        1750 CONTINUE                                             18180   000265
00213     85*             ARG = EVAL                                           18190   000265
00215     86*             IF (ABS(C(1)).LT.TWNTY .AND. ABS(C(2)).LT.TWNTY) GO TO 2000  000266
00216     87*             EVAL=EVAL/FIFTEN                                     18210   000304
00217     88*             NSCALE=NSCALE+15                                     18220   000311
00220     89*             GO TO 1750                                           18230   000314
00221     90*        2000 CONTINUE                                             18240   000317
00223     91*             RETURN                                               18250   000317
00224     92*             END                                                  18260   000401

     END OF COMPILATION:       2 DIAGNOSTICS.
```

```
@FOR,USW F,DETSD,F,DETSD
FOR.SEIX=05/23/74-08:29:59 15,6)

      SUBROUTINE DETSD      ENTRY POINT 00064B

      STORAGE USED: CODE(1) 000664; DATA(0) 000306; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP5  000074
      0004   KEEP14 000031
      0005   KEEP19 000005
      0006   KEEP21 001133
      0007   CRUD2  003737

      EXTERNAL REFERENCES (BLOCK, NAME)

      0010   CDABV
      0011   XPR1
      0012   CDV$
      0013   CEXP
      0014   NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001  000020 127G     0001  000037 137G     0001  000107 155G     0001  000222 203G     0001  000243 214G
      0001  000302 224G     0001  000071 310L     0001  000155 350L     0001  000211 399L     0001  000213 400L
      0001  000275 410L     0001  000355 415L     0001  000411 420L     0001  000541 500L     0000 R 000241 A
      0003 L 000071 BOTH     0000 R 000243 C       0010 R 000000 CDABV    0004  000025 DEG      0007  003734 DELTA
      0004  000012 D0       0004  000013 D1       0004  000014 D2       0004  000015 D3       0004  000016 D4
      0007  001754 EIP      0007  000004 ERP      0004 C 000010 FIFTEN   0004  000017 FIFTY    0004 C 000006 FOUR
      0007 R 000002 GAINV    0003 L 000073 GPRINT    0006 C 000000 HALF     0000 I 000234 I        0000  000261 INJP$
      0003  000067 ITHZT    0000 I 000236 J        0000 I 000233 K        0005 L 000000 LGAIN    0006  000704 LOCDZ
      0006  000703 LOCNZ    0005 L 000001 LPHASE    0005 L 000002 LPOLES    0005 L 000003 LSDRL    0005 L 000004 LZEROS
      0003 L 000072 MODIFY   0000 I 000235 N        0006 I 001020 NOCZ     0007  003735 NGSYM    0000 I 000240 NMAX
      0004 I 000705 NNCZ     0007  003724 NPG      0007  003725 NPP      0007  003736 NPSYM    0000 I 000237 NP2
      0007  003726 NSHIFT   0000 I 000231 NS1      0000 I 000232 NS2      0006 I 000702 NUMZ     0003  000004 NZT
      0004  000026 N1       0004  000027 N2       0004  000030 N3       0004 C 000002 ONE      0000 C 000241 P
      0000 C 000243 PART1    0000 C 000000 PART2    0006 R 000341 PDZ      0007 C 000000 PHAVAR   0004  000020 PI
      0004  000021 PI2      0006 R 000000 PNZ      0004 R 000024 RADDEG   0004  000023 RP1      0007 R 000003 SHIFT
      0004  000022 SMALL    0003  000000 SUPERK    0003  000003 TD       0000 R 000230 TWNTY    0004 C 000004 TWO
      0007  003727 XR       0000 C 000002 Y        0005  003730 YB       0003 L 000070 YESZOH   0003  000002 ZM
      0003  000001 ZT       0003  000005 ZTVAL

00101    1*          SUBROUTINE DETSD(X,EVAL,NSCALE)                                                    000000
00103    2*          COMPLEX      X,EVAL                                                                000000
00104    3*          COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,          150          000000
00104    4*          1             MODIFY,GPRINT                                             160          000000
00105    5*          LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                                 180          000000
00106    6*          COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,         3040          000000
```

```
                      1     P1Z,SMALL,RPI,RADDEG,DEG,N1,N2,N3                    $05@    @@@@@@
                      8*    COMPLEX     HALF,ONE,TWO,FOUR,FIFTEN                         000000
                      9*    COMMON/KEEP19/LGAIN,LPHASE,LPOLES,LSORL,LZEROS               000000
                      10*   LOGICAL     LGAIN,LPHASE,LPOLES,LSORL,LZEROS                 000000
                      11*   COMMON/KEEP21/NZ(25),PDZ(25),NUMZ,LOCNZ,LOCDZ,NNCZ(75),NDCZ(75)  000000
                      12*   COMMON/CRUDZ/ PHAVAR,GAIN,ISHIFT,ERR(1000),EIP(1000),NPS,NPP     $01@    000000
                      13*   1             ISHIFT,XR,YB(4),DELTA,NGSYM,NPSYM            $02@    000000
                      14*   COMPLEX     PHAVAR                                         $03@    000000
                      15*   COMPLEX     P,PART1,PART2,Y(75)                                    000000
                      16*   DIMENSION   A(2),C(2)                                              000000
                      17*   EQUIVALENCE (P,A(1)),(PART1,C(1))                                  000000
                      18*C                                                                     000000
                      19*C                                                                     000000
                      20*C  ROOTING FOR SAMPLED DATA SYSTEM                                    000000
                      21*C                                                                     000000
                      22*   TWENTY = 1.0E20                                                    000000
                      23*   PART1 = (0.,0.)                                                    000003
                      24*   PART2 = (0.,0.)                                                    000005
                      25*   NS1 = 0                                                            000007
                      26*   NS2 = 0                                                            000010
                      27*   K = 0                                                              000011
                      28*   DO 310 I=1,NUMZ                                                    000020
                      29*   K = K + 1                                                          000020
                      30*   Y(I) = CMPLX(PDZ(K),0.)                                            000022
                      31*   N = NDCZ(I)                                                        000026
                      32*   IF (N.EQ.1) GO TO 310                                              000030
                      33*   DO 305 J=2,N                                                       000033
                      34*   K = K + 1                                                          000037
                      35*   Y(I) = Y(I)*X + CMPLX(PDZ(K),0.)                                   000041
                      36*305 CONTINUE                                                          000074
                      37*310 CONTINUE                                                          000074
                      38*   IF (LZEROS) GO TO 400                                              000074
                      39*   PART1 = (1.,0.)                                                    000076
                      40*   IF (MODIFY) PART1=X                                                000100
                      41*   DO 399 I=1,NUMZ                                                    000107
                      42*   PART1 = Y(I)*PART1                                                 000107
                      43*   IF (ABS(C(1)).LT.TWENTY .AND. ABS(C(2)).LT.TWENTY) GO TO 350       000125
                      44*   NS1 = NS1 + 10                                                     000143
                      45*   PART1 = PART1/1.0E10                                               000146
                      46*   GO TO 399                                                          000153
                      47*350 CONTINUE                                                          000155
*DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
                      48*   IF (CDABV(PART1) .EQ. 0.0) GO TO 399                               000155
                      49*   IF (ABS(C(1)).GE.1.E-20 .OR. ABS(C(2)).GE.1.E-20) GO TO 399        000161
                      50*   PART1 = PART1*1.E10                                                000177
                      51*   NS1 = NS1 - 10                                                     000205
                      52*399 CONTINUE                                                          000213
                      53*400 CONTINUE                                                          000213
                      54*   IF (LPOLES) GO TO 500                                              000213
                      55*   K = 0                                                              000214
                      56*   DO 450 I=1,NUMZ                                                    000222
                      57*   NP2 = 0                                                            000222
                      58*   K = K + 1                                                          000223
                      59*   P = CMPLX(PNZ(K),0.)                                               000227
                      60*   N = NDCZ(I)                                                        000232
                      61*   IF (N.EQ.1) GO TO 410                                              000234
                      62*   DO 405 J=2,N                                                       000237
```

```
00226   65*                K = K * 10                                          000248
00227   66*                P = P*Z * CMPLX(PHZ(K),0.)                          000246
00227   67*            415 CONTINUE                                            000275
00227   68*            410 CONTINUE                                            000275
00227   69*                DO 420 J=1,NUMZ                                     000275
00227   70*                IF (I.EQ.J) GO TO 420                               000302
00230   71*                P = P*Y(J)                                          000306
00231   72*                IF (ABS(A(1)).LT.TWNTY .AND. ABS(A(2)).LT.TWNTY) GO TO 415   000325
00233   73*                NP2 = NP2 + 10                                      000343
00234   74*                P = P/1.0E10                                        000346
00235   75*                GO TO 420                                           000353
00236   76*            415 CONTINUE                                            000355
00237  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00237   77*                IF (CDABV(P) .EQ. 0.0) GO TO 420                    000355
00241   78*                IF (ABS(A(1)).GE.1.E-20 .OR. ABS(A(2)).GE.1.E-20) GO TO 420   000361
00243   79*                P = P*1.E10                                         000377
00244   80*                NP2 = NP2 - 10                                      000405
00245   81*            420 CONTINUE                                            000413
00247   82*                NMAX = MAXO(NS2,NP2)                                000413
00250   83*                PART2 = PART2/10.0**(NMAX-NS2) + P/10.0**(NMAX-NP2)   000421
00251   84*                NS2 = NMAX                                          000460
00252   85*            450 CONTINUE                                            000464
00254   86*                IF (LZEROS) GO TO 500                               000464
00257   87*                IF (LGAIN) PART2=GAINV*PART2                        000466
00260   88*                IF (LPHASE) PART2=CEXP(CMPLX(0.0,RADDEG*SHIFT))*PART2   000510
00262   89*            500 CONTINUE                                            000541
00263   90*                NSCALE = MAXO(NS1,NS2)                              000541
00264   91*                IF (NS1.NE.NSCALE)PART1=PART1/10.0**(NSCALE-NS1)    000546
00266   92*                IF (NS2.NE.NSCALE)PART2=PART2/10.0**(NSCALE-NS2)    000570
00270   93*                EVAL = PART1 + PART2                                000612
00271   94*                RETURN                                              000620
00272   95*                END                                                 000663

29-9

      END OF COMPILATION:        2 DIAGNOSTICS.
```

```
?FOR,US# ,ERROR,F,ERROR
FOR SEIX-05/23/74-08:30:24 (2.3)

      SUBROUTINE ERROR      ENTRY POINT 000017

      STORAGE USED: CODE(1) 000021; DATA(0) 000011; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP16 000031

      EXTERNAL REFERENCES (BLOCK, NAME)

      0004   RESET
      0005   NYDUS
      0006   NIOZS
      0007   NERR3S

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0000   000000 600F      0003 L 000022 DEBUG      0000   000005 INJPS      0003   000010 IOPEN      0003   000011 JOPEN
      0003   000001 KODE      0003 L 000023 LFLT       0003 L 000027 NOMNAL     0003 L 000030 NOTYET     0003   000002 NRCLPL
      0003   000003 NRPOLE    0003   000004 NRZERO     0003   000005 NXB        0003   000006 NXN        0003   000007 NXR
      0003   000014 PCPL      0003   000015 PFAC       0003   000013 PNOM       0003   000016 PSLOSH     0003   000012 PVAR
      0003   000030 STAGE     0003 L 000017 YESMTX     0003 L 000024 YESPCH     0003 L 000020 YESRAW     0003 L 000025 YESRLP
      0003 L 000026 YESSRL     0003 L 000021 YESSRP

      END OF COMPILATION:       NO  DIAGNOSTICS.
```



```
00101     1*            SUBROUTINE ERROR                                          1827B  000000
00102     2*            COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,  1828B  000000
00103     3*          1               IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,     1829B  000000
00103     4*          2               YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,  1830B  000000
00103     5*          3               YESSRL,NOMNAL,NOTYET                        1831B  000000
00104     6*            LOGICAL       YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,  1832B  000000
00105     7*          1               YESSRL,NOMNAL,NOTYET                        1833B  000000
00105     8*            WRITE(6,600) KODE                                         1834B  000000
00110     9*        600 FORMAT('1ERROR CODE = ',I3)                               1835B  000005
00111    10*            CALL RESET                                                1836B  000005
00112    11*            RETURN                                                    1837B  000007
00113    12*            END                                                       1838B  000020
```

```
@FOR,USN F.ESTMAT,F.ESTMAT
FOR SEIX-05/23/74-08:30:33 (4.5)

      SUBROUTINE ESTMAT     ENTRY POINT 000262

      STORAGE USED: CODE(1) 000266; DATA(0) 000401; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP1   000026
      0004   KEEP9   000705
      0005   KEEP15  000260
      0006   CRUD3   016115

      EXTERNAL REFERENCES (BLOCK, NAME)

      0007   CDABV
      0010   NWDUS
      0011   NIO2S
      0012   NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001  000034 10L     0001  000041 15L     0001  000077 20L     0001  000144 25L     0001  000151 30L
      0001  000160 35L     0001  000222 40L     0001  000242 50L     0000  000000 600F    0006 L 016113 AUTO
      0006 C 016072 B       0007 R 000000 CDABV    0006 L 016112 CONJ    0005  000116 CPSLBL    0006 C 000000 CU
      0005  000024 DA      0005  000130 DCBLBL    0005  000131 DEGLBL    0006 L 016110 DONE    0004 C 000226 EA
      0005  000146 EMAT    0005  000132 EMT       0006 C 016040 FPRD      0006 C 016042 FPR1    0006 C 016044 FPR2
      0006 C 016046 FR0     0006 C 016050 FR1     0006 C 016052 FR2     0005  000162 F6    0005  000163 F7
      0005  000100 ID3     0005  000107 IDG     0003  000030 IHJPS     0004  000572 KD    0005  000164 LABEL1
      0005  000210 LABEL2   0005  000234 LABEL3   0003  000000 MAXIT     0003  000001 MXEIG    0003  000024 MXEIGT
      0003  000023 MXEST    0003  000002 MXFRM    0003  000003 MXNBM    0003  000004 MXNCOF    0003 I 000005 MXNCT
      0003  000025 MXNCV     0003  000006 MXNE      0003  000007 MXNEQ     0003  000010 MXNFI    0003  000011 MXNG
      0003  000012 MXNPH     0003  000013 MXNPP     0003  000014 MXNQPT    0003  000015 MXNSM    0003  000016 MXNSP
      0003  000017 MXNTM     0003  000020 MXNV      0003  000021 MXNZT     0003  000022 MXPOLY    0004 I 000455 NA
      0006 I 016102 NCT     0004 I 000454 NEIG      0006 I 016107 NEST    0006  016076 NFPO    0006  016077 NFP1
      0006  016100 NFP2    0004  000457 NI        0006  016101 NITER    0006  016104 NKODE    0004  000456 NR
      0006 I 016103 NREG    0006 I 016105 NSTART    0006 I 016106 NTIME     0006 C 016054 PRO     0006 C 016056 PR1
      0006 C 016060 PR2     0005  000034 P36       0005 C 000000 REGION    0006 L 016114 REGSEL    0006 L 016111 RESTRT
      0005  000030 RID     0004 C 000000 ROOT      0006 C 016062 RU      0006 C 016064 R1      0006 C 016066 R2
      0006 C 016070 R3      0005  000014 SPACE     0006 C 016074 U       0005  000010 WIDTH    0005  000020 YINC

00101    1*          SUBROUTINE ESTMAT                                                    18390   000000
00103    2*          COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,       18400   000000
00103    3*     1                  MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,     18410   000000
00103    4*     2                  MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV                 18420   000000
00104    5*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)              18430   000000
00105    6*          COMPLEX       ROOT,EA                                                       000000
00107    7*          COMMON/KEEP15/REGION(4),WIDTH(4),SPACE(4),YINC(4),                          000000
00106    8*     1                  DA(4),RID(4),P36(36),IDB(7),IDC(7),CPSLBL(10),        18470   000000
```

```
00106     9*          2     DOUBLE,DEGLBL,FMT(12),FMAT(12),F6,F7,LABEL1(20),     18480   000000
00106    10*          3     LABEL2(20),LABEL3(20)                                18490   000000
00107    11*          COMPLEX REGION                                             18500   000000
00110    12*          COMMON/CBUF3/ CU(60,60),FPRO,FPR1,FPR2,FRO,FR1,FR2,PRO,PR1,PR2,  18520   000000
00110    13*          1     RO,R1,R2,R3,B,U,NFPO,NFP1,NFP2,NITER,NCT,NREG,NKODE,  18530   000000
00110    14*          2     NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL       18540   000000
00111    15*          COMPLEX CU,FPRO,FPR1,FPR2,FRO,FR1,FR2,PRO,PR1,PR2,                   000000
00111    16*          1     RO,R1,R2,R3,B,U                                      18560   000000
00112    17*          LOGICAL DONE,RESTRT,CONJ,AUTO,REGSEL                       18570   000000
00112    18*     C                                                               18580   000000
00112    19*     C  PROGRAM CODING                                               18590   000000
00112    20*     C                                                               18600   000000
00113    21*          DONE=.FALSE.                                               18610   000000
00114    22*          IF(RESTRT) GO TO 40                                        18620   000000
00115    23*     C                                                               18630   000000
00115    24*     C                                                               18640   000000
00115    25*     C*****    FIRST ESTIMATE OR NORMAL RESTART                      18650   000000
00115    26*     C                                                               18660   000000
00116    27*          B = (.1,0.)                                                        000002
00117    28*          IF(.NOT.CONJ).OR.(NSTART.GT.1)) GO TO 10                   18680   000004
00117    29*     C                                                               18690   000004
00117    30*     C                                                               18700   000004
00117    31*     C*****    SELECT CONJUGATE OF PREVIOUS ROOT AS ESTIMATE         18710   000004
00117    32*     C                                                               18720   000004
00121    33*          U = 1.05*CONJG(R3) + (.1,.1)                                       000015
00122    34*          RETURN                                                     18740   000030
00122    35*     C                                                               18750   000030
00122    36*     C                                                               18760   000030
00122    37*       10 CONTINUE                                                   18770   000034
00123    38*          IF(NEST.GE.NA) GO TO 20                                    18780   000034
00124    39*     C                                                               18790   000034
00124    40*     C                                                               18800   000034
00124    41*     C*****    SELECT USER INPUT ESTIMATE                            18810   000034
00124    42*     C                                                               18820   000034
00126    43*       15 CONTINUE                                                   18830   000041
00127    44*          NEST=NEST+1                                                18840   000041
00130    45*          U = 1.05*EA(NEST) + (.01,.01)                                      000043
00131    *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00131    46*          IF (CDABS(EA(NEST)) .EQ. 0.0) U=(0.,0.)                            000056
00132    47*          RETURN                                                     18860   000073
00133    48*     C                                                               18870   000073
00133    49*     C                                                               18880   000073
00134    50*       20 CONTINUE                                                   18890   000077
00135    51*          IF((NEIG.EQ.0).OR.(NSTART.GT.1)) GO TO 35                  18900   000077
00135    52*     C                                                               18910   000077
00135    53*     C                                                               18920   000077
00135    54*     C****     ACCEPT PREVIOUS ROOT AS ESTIMATE                      18930   000077
00135    55*     C                                                               18940   000077
00137    56*          IF (CDABS(ROOT(NEIG)) .LE. .0001) GO TO 25                         000111
00141    57*          U = 1.05*ROOT(NEIG) + (.1,.1)                                      000124
00142    58*          RETURN                                                             000140
00143    59*       25 CONTINUE                                                           000144
00144    60*          U = (0.,0.)                                                18960   000145
00145    61*          RETURN                                                     18970   000145
00145    62*     C                                                               18980   000145
00145    63*     C                                                               18990   000145
00145    64*     C*****    MODULUS OF FUNCTIONAL VALUES INCREASING
```

```
00145    65*      C                                                          19000    000145
00146    66*           30 CONTINUE                                           19010    000151
00147    67*              B = (.1,0.)                                                   000151
00150    68*              IF(NEST.LT.NA) GO TO 15                            19030    000152
00150    69*      C                                                          19040    000152
00150    70*      C                                                          19050    000152
00150    71*      C*****       TAKE ESTIMATE FROM NEXT REGION                19060    000152
00150    72*      C                                                          19070    000152
00152    73*           35 CONTINUE                                           19080    000160
00153    74*              IF(REGSEL) NREG=NREG+1                             19090    000160
00155    75*              NTIME=NTIME+1                                      19100    000164
00156    76*              IF(NTIME.GT.9) GO TO 50                            19110    000167
00160    77*              IF((NREG.GT.4).OR.(NREG.EQ.0)) NREG=1              19120    000173
00162    78*              REGSEL=.TRUE.                                      19130    000210
00163    79*              U=REGION(NREG)                                     19140    000212
00165    80*              RETURN                                             19150    000216
00164    81*      C                                                          19160    000216
00164    82*      C                                                          19170    000216
00164    83*      C*****       SPECIAL RESTART                               19180    000216
00164    84*      C                                                          19190    000216
00165    85*           40 CONTINUE                                           19200    000222
00166    86*              IF (NCT.GT.MXNCT) GO TO 30                         19210    000222
00166    87*      C                                                          19220    000222
00166    88*      C                                                          19230    000222
00166    89*      C*****       RAPID DECLINE IN FUNCTIONAL VALUES            19240    000222
00166    90*      C*****       TAKE PREVIOUS ITERANT AS ESTIMATE AND REDUCE BETA 19250    000222
00166    91*      C                                                          19260    000222
00170    92*              B = (.01,0.)*B                                              000226
00171    93*              U=R2                                               19280    000234
00172    94*              RETURN                                             19290    000236
00172    95*      C                                                          19300    000236
00172    96*      C*****       ONLY FOUR REGIONS ARE AVAILABLE               19310    000236
00172    97*      C                                                          19320    000236
00173    98*           50 CONTINUE                                           19330    000242
00174    99*              DONE=.TRUE.                                        19340    000242
00175   100*              WRITE(6,600)                                       19350    000243
00177   101*          600 FORMAT(' ** NO MORE ESTIMATES AVAILABLE *** CAN NOT CONVERGE')  19360  000250
00200   102*              RETURN                                             19370    000250
00201   103*              END                                                19380    000265

      END OF COMPILATION:      1 DIAGNOSTICS.
```

```
$FOR,US# *.FIXIT,F.FIXIT
FOR SEIX-05/23/74-08:30:46 (2.3)

      SUBROUTINE FIXIT      ENTRY POINT 00003S

   STORAGE USED: CODE(1) 0n0047; DATA(0) 0n0012; BLANK COMMON(2) 000000

      EXTERNAL REFERENCES (BLOCK, NAME)

         0003    XPRI
         0004    CDVS
         0005    NERR3S

   STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0000  000005 INJPS      0000 I 000002 NDIF      0000 C 000000 SC      0000 R 000003 SCL

00101     1*          SUBROUTINE FIXIT(X,NX,N)                          20670   000000
00103     2*          COMPLEX        X,SC                                       000000
00103     3*    C                                                       20710   000000
00103     4*    C
                                                                        20720   000000
00103     5*    C  RESCALE THE PARAMETER VALUE                          20730   000000
00103     6*    C
                                                                        20740   000000
00104     7*          NDIF=N-NX                                         20750   000000
00105     8*          IF(NDIF.EQ.0) RETURN                              20760   000002
00107     9*          SCL=10.0**NDIF                                    20770   000007
00110    10*          SC=SCL                                            20780   000014
00111    11*          X=X/SC                                            20790   000016
00112    12*          NX=N                                              20800   000023
00113    13*          RETURN                                            20810   000025
00114    14*          END                                               20820   000046

   END OF COMPILATION:      NO DIAGNOSTICS.
```

```
?FOR:US# F.FORM,F.FORM
FOR SEIx=05/23/74-08:30:53 (2,3)

      SUBROUTINE FORM      ENTRY POINT 000316

      STORAGE USED: CODE(I) 000355; DATA(D) 000044; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003  KEEP9  000705
      0004  CRUD1  000234

      EXTERNAL REFERENCES (BLOCK, NAME)

      0005  NERR36

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001  000047 10L      0001  000022 12G      0001  000070 14DG     0001  000103 146G     0001  000167 160G
      0001  000229 173G     0001  000243 204G     0001  000163 205L     0001  000215 22L      0001  000234 30L
      0001  000260 60L      0001  000267 70L      0004 C 000000 COEF     0003 C 000226 EA       0004 R 000000 ECOEF
      0003 R 000000 EROOT   0000 I 000000 I        0000 I 000015 INJP      0000 I 000001 J        0000 I 000002 JEND
      0000 I 000003 JJ      0004 I 000232 JMAX     0000 I 000004 JTH      0003  000572 KD       0004 I 000233 N
      0003  000556 NA       0003 I 000454 NEIG     0003  000457 NI       0003  000456 NR       0003 C 000000 ROOT
      0004 C 000230 SMROOT

00101     1*          SUBROUTINE FORM(NDG,RR,RI,FPOL,NFPOL,NZZ,NSCALE)                    23190   000001
00103     2*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)              23200   000001
00104     3*          COMPLEX       ROOT,EA                                                       000001
00105     4*          COMMON/CRUD1/ COEF(76),SMROOT,JMAX,N                              23230   000001
00106     5*          COMPLEX       COEF,SMROOT                                                 000001
00107     6*          DIMENSION     EROOT(150),ECOEF(152)                                       000001
00110     7*          DIMENSION     RR(75),RI(75),FPOL(76)                                      000001
00111     8*          EQUIVALENCE   (ROOT(1),EROOT(1)),(COEF(1),ECOEF(1))               23270   000001
00111     9*     C                                                                       23280   000001
00111    10*     C                                                                       23290   000001
00111    11*     C  STORE REAL AND IMAGINARY PARTS OF COMPUTED ROOTS                     23300   000001
00111    12*     C                                                                       23310   000001
00112    13*          NZZ=0                                                              23320   000001
00113    14*          NSCALE = 0                                                         23330   000002
00114    15*          NDG = NEIG                                                         23340   000003
00115    16*          NFPOL = NEIG + 1                                                   23350   000005
00117    17*          IF (NEIG.EQ.0) GO TO 60                                            23360   000007
00120    18*          DO 10 I=1,NEIG                                                     23370   000012
00123    19*          N=2*I-1                                                            23380   000022
00124    20*          RR(I)=EROOT(N)                                                     23390   000026
00125    21*          RI(I)=EROOT(N+1)                                                   23400   000031
00126  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00126    22*          IF (FR(I).NE.0.0 .OR. RI(I).NE.0.0) GO TO 10                               000033
00130    23*          NZZ = NZZ + 1                                                              000043
```

```
00131    24*        10 CONTINUE                                          23460     000052
00131    25*     C                                                       23470     000052
00131    26*     C                                                       23480     000052
00131    27*     C  FORM POLYNOMIAL FROM THE COMPUTED ROOTS              23490     000052
00131    28*     C                                                       23500     000052
00133    29*           IF (NEIG.EQ.1) GO TO 70                                     000052
00135    30*           COEF(1) = -ROOT(1)                                          000056
00136    31*           COEF(2) = (1.0,0.0)                                         000060
00137    32*           DO 30 I=2,NEIG                                       23540  000070
00142    33*           SMROOT=ROOT(I)                                       23550  000070
00143    34*           COEF(I+1) = COEF(I)                                  23560  000072
00144    35*           JMAX=I-1                                             23570  000074
00145    36*           DO 20 J=1,JMAX                                       23580  000077
00150    37*           N=I-J+1                                              23590  000103
00151    38*           COEF(N)=COEF(N-1)-COEF(N)*SMROOT                     23600  000107
00152    39*        20 CONTINUE                                             23610  000137
00154    40*           COEF(1)=-SMROOT*COEF(1)                              23620  000137
00155    41*           JEND = I + 1                                         23630  000157
00156    42*     205 CONTINUE                                                       000163
00157    43*           DO 21 JJ=1,JEND                                      23640  000163
00162    44*           JTH = 2*JJ                                                  000167
00163    45*           IF (ABS(ECOEF(JTH-1)).GT.1.0E20 .OR.                        000172
00163    46*          1    ABS(ECOEF(JTH)).GT.1.0E20) GO TO 22                    000172
00165    47*        21 CONTINUE                                             23670  000213
00167    48*           GO TO 30                                             23680  000213
00170    49*        22 CONTINUE                                             23690  000215
00171    50*           NSCALE = NSCALE + 10                                 23700  000215
00172    51*           DO 25 JJ=1,JEND                                      23710  000224
00175    52*           COEF(JJ) = COEF(JJ)/1.0E10                           23720  000224
00176    53*        25 CONTINUE                                             23730  000232
00200    54*           GO TO 205                                                   000232
00201    55*        30 CONTINUE                                             23740  000243
00203    56*           DO 50 I=1,NFPOL                                      23760  000243
00206    57*           N=2*I-1                                              23770  000243
00207    58*           FPOL(I)=ECOEF(N)                                     23780  000250
00210    59*        50 CONTINUE                                             23790  000254
00212    60*           RETURN                                               23800  000254
00212    61*     C                                                          23810  000254
00212    62*     C                                                          23820  000254
00212    63*     C  NO ROOTS HAVE BEEN COMPUTED                             23830  000254
00212    64*     C                                                          23840  000254
00213    65*        60 CONTINUE                                                     000260
00214    66*           FPOL(1) = 1.0                                               000260
00215    67*           FPOL(2) = 0.0                                               000261
00216    68*           FPOL(3) = 0.0                                               000262
00217    69*           RETURN                                                      000263
00217    70*     C                                                                 000263
00217    71*     C                                                                 000263
00217    72*     C  ONE ROOT COMPUTED  ***  IT MUST BE REAL                        000263
00217    73*     C                                                                 000263
00220    74*        70 CONTINUE                                                    000267
00221    75*           FPOL(1) = -RR(1)                                            000267
00222    76*           FPOL(2) = 1.0                                               000270
00223    77*           FPOL(3) = 0.0                                               000272
00224    78*           RETURN                                               23880  000273
00225    79*           END                                                  23890  000354
```

```
FOR,JSW [illegible]RMTX,F,FRMTX
FOR SEIX-05/23/74-08:31:07-43,4)

      SUBROUTINE FRMTX      ENTRY POINT 00017A

 STORAGE USED: CODE(1) 000210; DATA(0) 000043; BLANK COMMON(2) 000000

 COMMON BLOCKS:

 0003  KEEP10 021620
 0004  KEEP19 000005
 0005  KEEP16 000031
 0006  CR002  003737
 0007  CR003  016115

 EXTERNAL REFERENCES (BLOCK, NAME)

 0010  NERR3S

 STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

 0001  000113 11L     0001  000021 117G    0001  000022 122G    0001  000033 130G    0001  000057 141G
 0001  000132 157G    0001  000147 40L     0007 L 016113 AUTO    0007 C 016072 B       0007 L 016112 CONJ
 0007 C 000000 CU      0005 L 000022 DEBUG   0006  003734 DELTA   0007 L 016110 DONE    0006  001754 EIP
 0006  000004 ERP     0003 R 016664 EV      0007 C 016040 FPR0    0007 C 016042 FPR1    0007 C 016044 FPR2
 0007 C 016046 FR0     0007 C 016050 FR1     0007 C 016052 FR2     0006  000002 GAINV   0000 I 000002 J
 0000 I 000011 II      0000  000020 INJP$    0005 I 000010 IOPEN   0003 I 000004 IR      0000 I 000003 J
 0003 I 001754 JC      0000 I 000012 JJ      0005 I 000011 JOPEN   0000 I 000005 K       0005  000001 KODE
 0000 I 000006 L       0005 L 000023 LFLT    0004 L 000000 LGAIN   0003 I 005674 LL      0003  007644 LOCPOL
 0004 L 001001 LPHASE  0004 L 000002 LPOLES  0004 L 000003 LSDRL   0004 L 000004 LZEROS   0000 I 000007 M
 0000 I 001010 MMJ     0000 I 000004 N       0003  000003 NCOF    0007  016102 NCT     0003 I 003724 ND
 0003  001000 NDEG    0003 I 000002 NE      0003 I 000001 NEQ     0007  016107 NEST    0007  016076 NFP0
 0007  016077 NFP1    0007  016100 NFP2    0006  003735 NGSYM   0007  016101 NITER   0007  016104 NKODE
 0005 L 000027 NOMNAL  0005 L 000030 NOTYET  0006  003724 NPG     0006  003725 NPP     0006  003736 NPSYM
 0005  000002 NRCLPL   0007  016103 NREG    0005  000003 NRPOLE  0005  000004 NRZERO  0006  003726 NSHIFT
 0007  016105 NSTART   0007  016106 NTIME   0005  000005 NXB     0005  000006 NXN     0005  000007 NXR
 0005  000014 PCPL     0005  000015 PFAC    0006 C 000000 PHAVAR  0005  000013 PNOM    0000 C 000000 POLY
 0007 C 016054 PR0     0007 C 016056 PR1     0007 C 016060 PR2     0005  000016 PSLOSH  0005  000012 PVAR
 0007 L 016114 REGSEL  0007 L 016111 RESTRT  0007 C 016062 R0      0007 C 016064 R1      0007 C 016066 R2
 0007 C 016070 R3      0006  003730 SHIFT    0005  000000 STAGE   0007 C 016074 U       0006  003727 XR
 0006  003730 YR      0005 L 000017 YESMTX  0005 L 000024 YESPCH  0005 L 000020 YESRAW  0005 L 000025 YESRLP
 0005 L 000026 YESSRL  0005 L 000021 YESSRP

 00101    1*        SUBROUTINE FRMTX(VAL)                                          23900    000022
 00103    2*        COMMON/KEEP10/NDEG,NEQ,NE,NCOF,IR(1000),JC(1000),ND(1000),     23910    000022
 00103    3*       1          LL(1000),LOCPOL(60,60),EV(1500)                     23920    000022
 00104    4*        COMMON/KEEP19/LGAIN,LPHASE,LPOLES,LSDRL,LZEROS                          000022
 00105    5*        LOGICAL    LGAIN,LPHASE,LPOLES,LSDRL,LZEROS                             000022
 00107    6*        COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,      240    000022
 00107    7*       1          IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,             250    000022
```

```
00104     8*          2                   YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      260     000022
00106     9*          3                   YESSRL,NONNAL,NOTYET                                270     000022
00107    10*          LOGICAL             YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      280     000022
00107    11*          1                   YESSRL,NOMNAL,NOTYET                                290     000022
00110    12*          COMMON/CRUD2/ PHAVAR,GAINV,SHIFT,ERP(1000),EIP(1000),NPG,NPP,           23930   000022
00110    13*          1                   NSHIFT,XR,YG(4),DELTA,NGSYM,NPSYM                   23940   000022
00111    14*          COMPLEX             PHAVAR                                              23950   000022
00112    15*          COMMON/CRUD3/ CU(60,60),EPRO,EPR1,EPR2,FRO,FR1,FR2,PRO,PR1,PR2,         23960   000022
00112    16*          1                   R0,R1,R2,R3,B,U,NEPO,NEP1,NEP2,NITER,NCT,NREG,NKODE,23970   000022
00112    17*          2                   NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL      23980   000022
00113    18*          COMPLEX             CU,EPRO,EPR1,EPR2,FRO,FR1,FR2,PRO,PR1,PR2,                  000022
00113    19*          1                   R0,R1,R2,R3,B,U                                     24000   000022
00114    20*          LOGICAL             DONE,RESTRT,CONJ,AUTO,REGSEL                        24010   000022
00115    21*          COMPLEX             VAL,POLY                                                    000022
00115    22*     C                                                                           24040   000022
00115    23*     C                                                                           24050   000022
00115    24*     C  FORM THE CHARACTERISTIC MATRIX                                           24060   000022
00115    25*     C                                                                           24070   000022
00116    26*          DO 5 I=1,NEQ                                                           24080   000022
00121    27*          DO 5 J=1,NEQ                                                           24090   000022
00124    28*        5 CU(I,J) = (0.,0.)                                                      24100   000022
00127    29*          DO 20 N=1,NE                                                           24110   000033
00132    30*          K = NO(N)                                                              24120   000033
00133    31*          L = LL(N) + K - 1                                                      24130   000034
00134    32*          POLY = CMPLX(EV(L),0.0)                                                24140   000040
00135    33*          IF (K.LT.2) GO TO 11                                                   24150   000043
00137    34*          M = L + 1                                                              24160   000050
00140    35*          DO 10 J=2,K                                                            24170   000052
00143    36*          MMJ = M - J                                                                    000057
00146    37*       10 POLY = POLY*VAL + EV(MMJ)                                                      000063
00146    38*       11 CONTINUE                                                               24190   000113
00147    39*          II = IR(N)                                                             24200   000113
00150    40*          JJ = JC(N)                                                             24210   000114
00151    41*          CU(II,JJ) = POLY                                                       24220   000121
00152    42*       20 CONTINUE                                                               24230   000124
00152    43*     C                                                                                  000124
00152    44*     C                                                                                  000124
00152    45*     C  MODIFY MATRIX IF OPEN LOOP ZEROS                                                000124
00152    46*     C                                                                                  000124
00154    47*          IF (.NOT.LZEROS) GO TO 40                                                     000124
00156    48*          DO 30 I=1,NEQ                                                                 000132
00161    49*          IF (I.NE.IOPEN) CU(I,JOPEN)=(0.,0.)                                           000132
00163    50*       30 CONTINUE                                                                      000143
00165    51*          RETURN                                                                        000143
00165    52*     C                                                                                  000143
00165    53*     C                                                                                  000143
00165    54*     C  MODIFY MATRIX IF OPEN LOOP POLES                                                000143
00165    55*     C                                                                                  000143
00166    56*       40 IF (.NOT.LPOLES) RETURN                                                       000147
00170    57*          CU(IOPEN,JOPEN) = (0.,0.)                                                     000153
00171    58*          RETURN                                                                        000160
00172    59*          END                                                                   24330   000207

     END OF COMPILATION:        0 DIAGNOSTICS.
```

SUBROUTINE GENMTX ENTRY POINT 000540

STORAGE USED: CODE(1) 000555; DATA(0) 000052; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003	KEEP1	000026
0004	KEEP10	021420
0005	KEEP16	000031
0006	CR002	000115

EXTERNAL REFERENCES (BLOCK, NAME)

0007	NR00$
0010	NI01$
0011	NI02$
0012	NER34$
0013	NER33$

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001		000467	1000L	0001		000372	110L	0001		000025	117G	0001		000031	124G
0001		000042	136G	0001		000410	150L	0001		000056	152G	0001		000067	163G
0001		000462	1900L	0001		000475	2000L	0001		000331	253G	0001		000370	266G
0001		000503	3000L	0001		000421	311G	0001		000430	314G	0001		000443	326G
0001		000511	4000L	0001		000047	50L	0000		000019	SNOF	0001		000240	60L
0001		000317	80L	0004		000024	ANORM	0004		000000	BCD	0005	L	000022	DEBUG
0000	I	000002	I	0004	I	000075	II	0007		000024	INJPS	0005		000010	IOPEN
0000	I	000003	J	0004	I	001754	JC	0004	I	000101	JJ	0005		000011	JOPEN
0006	I	000105	KK	0005	I	000001	KODE	0000	I	000001	L	0000	I	000006	LE
0004	I	005674	LL	0004	I	007644	LOCPOL	0000	I	000013	LI	0000	I	000004	M
0000	L	000000	MORE	0003		000001	MXEIG	0003		000024	MXEIGT	0003		000023	MXEST
0003		000003	MXNBM	0003	I	000004	MXNCOF	0003		000005	MXNCT	0003		000025	MXNCV
0003	I	000007	MXNEQ	0003		000010	MXNFI	0003		000011	MXNG	0003		000012	MXNPH
0003		000014	MXNPPT	0003		000015	MXNSH	0003		000016	MXNSP	0003		000017	MXNTM
0003		000021	MXNZT	0003		000022	MXPOLY	0000	I	000011	N	0004	I	000003	NCOF
0004	I	000000	NDEG	0004	I	000002	NE	0004	I	000001	NEQ	0006		000074	NFILT
0005	L	000027	NORMAL	0005	L	000030	NOTYET	0005		000002	NPCLPL	0005		000003	NRPOLE
0000	I	000007	NSTEP	0006		000071	NTMP0	0006		000072	NTMP0C	0006		000073	NTMP1C
0005		000006	NXN	0005		000007	NXR	0000	I	000012	N1	0006		000020	OPTINP
0004		000021	OPT1	0006		000022	OPT2	0006		000023	OPT3	0006		000111	PC
0005		000015	PFAC	0005		000013	PNOM	0005		000016	PSLOSH	0005		000012	PVAR
0005		000000	STAGE	0006		000006	TEMP	0006		000025	TEMP0	0006		000045	TEMP1
0006		000011	VFILT	0005	L	000017	YESMTX	0005	L	000024	YESPCH	0005	L	000020	YESRAW
0005		000026	YESSRL	0005	L	000021	YESSRP								

```
00103          COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNSM,MXNCOF,MXNCT,MXNE,MXNEQ,    2435   000014
00103     3*   1             MXNFI,MXNG,MXNPH,MXNPP,MXNOPT,MXNSM,MXNSP,MXNTM,    2436   000014
00103     4*   2             MXNV,MXNZT,MAPOLY,MXEST,MXEIGT,MXNCV                2437   000014
00105     5*                 COMMON/KEEP10/NDEG,NEQ,NE,NCOF,IR(1000),JC(1000),ND(1000),  2438   000014
00109     6*   1             LL(1000),LOCPOL(60,60),FV(1500)                     2439   000014
00105     7*                 COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,  2440   000014
00105     8*   1             IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,             2441   000014
00105     9*   2             YESMTX,YESRAN,YESCRP,DEBUG,LFLT,YESPCH,YESRLP,      2442   000014
00105    10*   3             YESSOL,NOMNAL,NOTYET                                2443   000014
00106    11*                 LOGICAL       YESMTX,YESRAN,YESCRP,DEBUG,LFLT,YESPCH,YESRLP,  2444   000014
00106    12*   1             YESSOL,NOMNAL,NOTYET                                2445   000014
00107    13*                 COMMON/CRUD2/ BCD(3),OPTTYP(3),TEMP(3),VFILT(6),REQEST,OPTINP,  2446   000014
00107    14*   1             OPT1,OPT2,OPT3,ANORM,TEMP0(16),TEMP1(16),VAL(4),    2447   000014
00107    15*   2             NTMP0,NTMP0C,NTMP1C,NFILT,II(4),JJ(4),KK(4),PC(4)   2448   000014
00110    16*                 LOGICAL       MORE                                  2450   000014
00110    17*       C                                                             2451   000014
00110    18*       C                                                             2452   000014
00110    19*       C  INITIALIZE PRIOR TO CREATION OF MATRIX                     2453   000014
00110    20*       C                                                             2454   000014
00111    21*             L = 1                                                   2455   000014
00112    22*             NE = 0                                                  2456   000016
00113    23*             NEQ = 0                                                 2457   000017
00114    24*             NDEG = 0                                                2458   000020
00115    25*             NCOF = 0                                                2459   000021
00116    26*             DO 10 I=1,MXNCOF                                        2460   000025
00121    27*          10 FV(I) = 0.0                                             2461   000025
00123    28*             DO 20 I=1,MXNEQ                                         2462   000032
00126    29*             DO 20 J=1,MXNEQ                                         2463   000032
00131    30*             LOCPOL(I,J) = 0                                         2464   000032
00132    31*          20 CONTINUE                                                2465   000042
00135    32*             DO 30 I=1,MXNE                                          2466   000042
00140    33*             IR(I) = 0                                               2467   000042
00141    34*             JC(I) = 0                                               2468   000042
00142    35*             ND(I) = 0                                               2469   000043
00143    36*             LL(I) = 0                                               2470   000044
00144    37*          30 CONTINUE                                                2471   000047
00144    38*       C                                                             2472   000047
00144    39*       C                                                             2473   000047
00144    40*       C  READ MATRIX POLYNOMIAL IN GENERAL FORM                     2474   000047
00144    41*       C  PROCESS A MATRIX DATA CARD AT A TIME                       2475   000047
00144    42*       C                                                             2476   000047
00146    43*          50 CONTINUE                                                2477   000047
00147    44*             MORE = .FALSE.                                          2478   000047
00150    45*             READ(5,500) (II(M),JJ(M),KK(M),VAL(M),M=1,4)            2479   000047
00161    46*        500 FORMAT(4(3I2,E14.6))                                     2480   000067
00162    47*             DO 150 M=1,4                                            2481   000067
00165    48*             I = II(M)                                               2482   000067
00166    49*             J = JJ(M)                                               2483   000071
00167    50*             K = KK(M) + 1                                           2484   000073
00170    51*             IF (I.EQ.0 .AND. J.EQ.0) GO TO 150                      2485   000076
00172    52*             MORE = .TRUE.                                           2486   000106
00173    53*             IF (I.LT.1 .OR. I.GT.MXNEQ) GO TO 1000                  2487   000110
00175    54*             IF (J.LT.1 .OR. J.GT.MXNEQ) GO TO 1000                  2488   000124
00177    55*             L = LOCPOL(I,J)                                         2489   000141
00200    56*             IF (L.NE.0) GO TO 60                                    2490   000146
00203    57*       C                                                             2491   000146
00203    58*       C                                                             2492   000146
```

```
00227    59*      C  ADD NEW ELEMENT
00227    60*      C
00227    61*            IF (J.GT.NEQ) NEQ=J
00228    62*            IF (J.GT.NEQ) NEQ=J
00228    63*            IF (K.GT.NDEG) NDEG=K
00210    64*            NE = NE + 1
00211    65*            IF (NE.GT.MXNE) GO TO 2000
00213    66*            IR(NE) = I
00214    67*            JC(NE) = J
00215    68*            ND(NE) = K
00216    69*            LL(NE) = NCOF + 1
00217    70*            LOCPOL(I,J) = NE
00220    71*            NCOF = NCOF + K
00221    72*            IF (NCOF.GT.MXNCOF) GO TO 3000
00223    73*            EV(NCOF) = VAL(M)
00224    74*            GO TO 150
00224    75*      C
00225    76*      C
00226    77*      C  OLD ELEMENT
00226    78*      C
00225    79*           60 CONTINUE
00226    80*            IF (K.GT.ND(L)) GO TO 70
00226    81*      C
00226    82*      C
00226    83*      C  DEGREE OK
00226    84*      C
00230    85*            LE = LL(L) + K - 1
00231    86*            EV(LE) = VAL(M)
00232    87*            GO TO 150
00232    88*      C
00232    89*      C
00232    90*      C  DEGREE OF ELEMENT MUST BE INCREASED
00232    91*      C
00233    92*           70 CONTINUE
00234    93*            IF (K.GT.NDEG) NDEG=K
00236    94*            NSTUFF = K - ND(L)
00237    95*            IF (NCOF+NSTUFF .GT. MXNCOF) GO TO 3000
00241    96*            IF (L.LT.NE) GO TO 80
00241    97*      C
00241    98*      C
00241    99*      C  LAST ELEMENT
00241   100*      C
00243   101*            ND(L) = K
00244   102*            NCOF = LL(L) + K - 1
00245   103*            EV(NCOF) = VAL(M)
00246   104*            GO TO 150
00246   105*      C
00246   106*      C
00246   107*      C  NOT LAST ELEMENT
00246   108*      C
00247   109*           80 CONTINUE
00250   110*            ND(L) = K
00251   111*            NFIX = NCOF - LL(L+1) + 1
00252   112*            DO 90 N=1,NFIX
00255   113*            N1 = NCOF - N + 1
00256   114*         90 EV(N1+NSTUFF) = EV(N1)
00260   115*            N = LL(L) + K - 1
```

```
2493₀   000146
2499₀   000146
2995₀   000150
2496₀   000157
2497₀   000166
2498₀   000201
2499₀   000204
2500₀   000210
2501₀   000213
2502₀   000215
2503₀   000217
2504₀   000222
2505₀   000225
2506₀   000227
2507₀   000233
2508₀   000236
2509₀   000236
2510₀   000236
2511₀   000236
2512₀   000236
2513₀   000240
2514₀   000240
2515₀   000240
2516₀   000240
2517₀   000240
2518₀   000240
2519₀   000245
2520₀   000252
2521₀   000254
2522₀   000254
2523₀   000254
2524₀   000254
2525₀   000254
2526₀   000256
2527₀   000257
2528₀   000266
2529₀   000272
2530₀   000277
2531₀   000277
2532₀   000277
2533₀   000277
2534₀   000277
2535₀   000304
2536₀   000307
2537₀   000313
2538₀   000315
2539₀   000315
2540₀   000315
2541₀   000315
2542₀   000315
2543₀   000317
2544₀   000317
2545₀   000321
2546₀   000325
2547₀   000331
2548₀   000335
2549₀   000344
```

```
00261   116*          EV(N) = VAL(M)                                    25500   000352
00262   117*          IF (NSTUFF.EQ.1) GO TO 110                        25510   000354
00264   118*          NFIX = NSTUFF - 1                                 25520   000360
00265   119*          DO 100 N1=1,NFIX                                  25530   000363
00270   120*          EV(N-N1) = 0.0                                    25540   000370
00271   121*      100 CONTINUE                                          25550   000372
00273   122*      110 CONTINUE                                          25560   000372
00274   123*          L1 = L + 1                                        25570   000372
00275   124*          DO 120 N=L1,NE                                    25580   000374
00300   125*          LL(N) = LL(N) + NSTUFF                            25590   000401
00301   126*      120 CONTINUE                                          25600   000404
00303   127*          NCOF = NCOF + NSTUFF                              25610   000404
00304   128*      150 CONTINUE                                          25620   000411
00306   129*          IF (MORE) GO TO 50                                25630   000411
00306   130*      C                                                     25640   000411
00306   131*      C                                                     25650   000411
00306   132*      C  CHECK MATRIX FOR SINGULARITY                       25660   000411
00306   133*      C                                                     25670   000411
00310   134*          DO 170 I=1,NEQ                                    25680   000413
00313   135*          DO 160 J=1,NEQ                                    25690   000430
00316   136*          IF (LOCPOL(I,J).NE.0) GO TO 170                   25700   000430
00320   137*      160 CONTINUE                                          25710   000435
00322   138*          GO TO 4000                                        25720   000435
00323   139*      170 CONTINUE                                          25730   000443
00325   140*          DO 190 J=1,NEQ                                    25740   000443
00330   141*          DO 180 I=1,NEQ                                    25750   000453
00333   142*          IF (LOCPOL(I,J).NE.0) GO TO 190                   25760   000453
00335   143*      180 CONTINUE                                          25770   000460
00337   144*          GO TO 4000                                        25780   000460
00340   145*      190 CONTINUE                                          25790   000463
00342   146*          RETURN                                            25800   000463
00342   147*      C                                                     25810   000463
00342   148*      C                                                     25820   000463
00342   149*      C  ROW OR COLUMN SPECIFIED IS OUT OF RANGE            25830   000463
00342   150*      C                                                     25840   000463
00343   151*      1000 CONTINUE                                         25850   000467
00344   152*          KODE = 20                                         25860   000467
00345   153*          RETURN 1                                          25870   000470
00345   154*      C                                                     25880   000470
00345   155*      C                                                     25890   000470
00345   156*      C  TOO MANY POLYNOMIAL ELEMENTS                       25900   000470
00345   157*      C                                                     25910   000470
00346   158*      2000 CONTINUE                                         25920   000475
00347   159*          KODE = 21                                         25930   000475
00350   160*          RETURN 1                                          25940   000476
00350   161*      C                                                     25950   000476
00350   162*      C                                                     25960   000476
00350   163*      C  TOO MANY POLYNOMIAL COEFFICIENTS                   25970   000476
00350   164*      C                                                     25980   000476
00351   165*      3000 CONTINUE                                         25990   000503
00352   166*          KODE = 22                                         26000   000503
00353   167*          RETURN 1                                          26010   000504
00353   168*      C                                                     26020   000504
00353   169*      C                                                     26030   000504
00353   170*      C  ZERO ROW OR ZERO COLUMN EXISTS                     26040   000504
00353   171*      C                                                     26050   000504
00354   172*      4000 CONTINUE                                         26060   000511
```

```
00355    173*          KODE = 23                                  26070    000511
00356    174*          RETURN 1                                   26080    000512
00357    175*          END                                        26090    000554

 END OF COMPILATION:  NO DIAGNOSTICS.
```

```
@FOR,USN F*GETEST,F*GETEST
FOR SE1X-05/23/79-08:32:22 (2.3)

      SUBROUTINE GETEST    ENTRY POINT 000031

      STORAGE USED: CODE(1) 000035; DATA(0) 000010; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003  KEEP9  000705
      0004  KEEP20 000227

      EXTERNAL REFERENCES (BLOCK, NAME)

      0005  NEPR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001  000015 113G      0003 C 000226 EA       0004 C 000000 ESTZ     0000 I 000000 I          0000   000002 INJP$
      0003  000572 KD        0003 I 000455 NA       0003   000454 NEIG     0004 I 000226 NESTZ      0003   000457 NI
      0003  000456 NR        0003 C 000000 ROOT

00101     1*          SUBROUTINE GETEST                                                    26100   000000
00102     2*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)               26170   000000
00104     3*          COMPLEX ROOT,EA                                                              000000
00105     4*          COMMON/KEEP20/ESTZ(75),NESTZ                                                 000000
00106     5*          COMPLEX ESTZ                                                                 000000
00106     6*    C                                                                          26220   000000
00106     7*    C                                                                          26230   000000
00106     8*    C  SELECT THE Z - DOMAIN POLES AS THE ESTIMATES                            26240   000000
00106     9*    C                                                                          26250   000000
00107    10*          NA = NESTZ                                                                   000000
00110    11*          IF (NESTZ.LE.0) RETURN                                                       000001
00112    12*          DO 10 I=1,NESTZ                                                              000010
00115    13*            EA(I) = ESTZ(I)                                                            000015
00116    14*       10 CONTINUE                                                                     000017
00120    15*          RETURN                                                             26420   000017
00121    16*          END                                                                26430   000034

      END OF COMPILATION:  NO DIAGNOSTICS.
```

```
PROPLUS - GRAPHS/REFGRAPHS
FOR SEQX=05/23/74=08:32:33-(1,2)

SUBROUTINE GRAPHS     ENTRY POINT 000035

STORAGE USED: CODE(I): 000037  DATA(D): 000041  BLANK COMMON(2): 000000

COMMON BLOCKS:

  0003   KEEP2   000047
  0004   KEEP3   000102
  0005   KEEP16  000031
  0006   PLT     000012

EXTERNAL REFERENCES (BLOCK, NAME)

  0007   NYQIST
  0010   BODE
  0011   NICHOL
  0012   NERR3$

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

  0001  000011 IOL      0001  000022 ZOL      0001  000026 3OL      0005 L 000022 DEBUG    0003 D 000002 DHFILT
  0006  000007 DIF1     0006  000010 DIF2     0004  000063 DP       0003  000037 HACC     0003  000025 HAST
  0003  000035 HATT     0003 D 000000 HBLANK   0003 R 000004 HBLK     0003  000034 HCCW     0003  000033 HCW
  0003  000032 HDEC     0003  000027 HDOT     0003  000006 HESTI    0003  000043 HFGD     0003  000044 HFGN
  0003  000040 HFPD     0003  000045 HFPDD    0003  000042 HFPDN    0003  000041 HFPN     0003  000010 HGENE
  0003  000031 HINC     0003  000005 HKEY     0003  000007 HMATR    0003  000014 HNEW     0003  000046 HNOMI
  0003  000015 HNYQU    0003  000023 HO       0003  000026 HPLUS    0003  000036 HRATE    0003  000011 HRAW
  0003  000012 HRETA    0003  000017 HROLL    0003  000016 HROOT    0003  000013 HSTAN    0003  000030 HSTAR
  0003  000020 HSIC     0003  000021 HS2      0003  000022 HS4B     0003  000024 HX       0006  000011 ICK
  0004  000004 ICT      0000  000000 INJP5    0005  000010 IOPEN    0006  000003 ISW      0005  000011 JOPEN
  0005  000001 KODE     0005 L 000023 LFLT     0004 R 000051 MAX      0004 R 000037 MIN      0004  000000 NFI
  0006  000001 NICPLT   0005 L 000027 NOMNAL   0005 L 000030 NOTYET   0006  000002 NP       0005  000002 NRCLPL
  0005  000003 NRPOLE   0005  000004 NRZERO   0005 I 000005 NXB      0005 I 000006 NXN      0005  000007 NXR
  0004 R 000076 PB      0005  000014 PCPL     0004  000025 PCT      0005  000015 PFAC     0004 R 000075 PN
  0006  000000 PNI      0005  000013 PNOM     0005  000016 PSLOSH   0005  000012 PVAR     0004  000077 P180
  0005  000000 STAGE    0004 L 000101 STNDRD   0004  000013 STP      0004  000001 STR      0006  000006 S360
  0006  000005 T360     0005 L 000017 YESMTX   0004 L 000100 YESNYQ   0005 L 000024 YESPCH   0005 L 000020 YESRAW
  0005 L 000025 YESPLP   0005 L 000026 YESSRL   0005 L 000021 YESSRP

00101   1*          SUBROUTINE GRAPHS                                                    37500   000000
00103   2*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HREY,HESTI,HMATR,HGENE,HRAW,        37510   000000
00103   3*      1       HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS4B,                37520   000000
00103   4*      2       HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,                  37530   000000
00103   5*      3       HATT,HRATE,HACC,HFPD,HFPDD,HFPDN,HFGD,HFGN,HFPDD,                37540   000000
00103   6*      4       HNOMI                                                            37550   000000
00107   7*          DOUBLE PRECISION HBLANK,DHFILT                                               000000
00105   8*          COMMON/KEEP3/ HFI,STR(10),STP(10),PCT(10),HMIN(10),HMAX(10),DP(10),  37570   000000
```

```
00105      9*          1                 PN,PA,P180,YESNYQ,STNDRD                    3758₀  000000
00106     10*               REAL         MIN,MAX                                     3759₀  000000
00107     11*               LOGICAL      YESNYQ,STNDRD                               3760₀  000000
00110     12*               COMMON/XFSP16/STAGE,KODE,NRPLPL,NRPOLE,NRZERO,NXB,NXN,NXR, 3761₀  000000
00110     13*          1                 IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,     3762₀  000000
00110     14*          2                 YESTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP, 3763₀  000000
00110     15*          3                 YESSRL,NOMNAL,NOTYET                        3764₀  000000
00111     16*               LOGICAL      YESTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP, 3765₀  000000
00111     17*          1                 YESSRL,NOMNAL,NOTYET                        3766₀  000000
00112     18*               COMMON /PLT/ PNT,NICPLT,NPTS,ICI,T360,S360,DIFL,DIFZ,ICK         000000
00113     19*               LOGICAL NICPLT                                                   000000
00113     20*     C                                                                  3767₀  000000
00113     21*     C                                                                  3768₀  000000
00113     22*     C    NYQUIST PLOTTING                                              3769₀  000000
00113     23*     C                                                                  3770₀  000000
*DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00114     24*               IF (PN.EQ.HBLK) GO TO 10                                 3771₀  000000
00115     25*               NXN = NXN + 1                                            3772₀  000003
00117     26*               CALL NYNIST                                              3773₀  000006
00120     27*          10   CONTINUE                                                 3774₀  000011
00120     28*     C                                                                  3775₀  000011
00120     29*     C    BODE PLOTTING                                                 3776₀  000011
00120     30*     C                                                                  3777₀  000011
*DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00121     31*               IF (PB.EQ.HBLK) GO TO 20                                 3778₀  000011
00122     32*               NXB = NXB + 1                                            3779₀  000014
00124     33*               CALL BODE                                                3780₀  000017
00125     34*          20   CONTINUE                                                 3781₀  000022
00125     35*               IF (.NOT. NICPLT) GO TO 30                                       000022
00130     36*               CALL NICHOL                                                      000023
00131     37*          30   CONTINUE                                                        000026
00132     38*               RETURN                                                   3782₀  000026
00133     39*               END                                                      3783₀  000036

      END OF COMPILATION:      2 DIAGNOSTICS.
```

```
@FOR,USN F.INITAL,F.INITAL
FOR SEIX-05/23/74-08:32:46 (2.3)

      SUBROUTINE INITAL     ENTRY POINT 000071

      STORAGE USED: CODE(1) 000227  DATA(0) 000101  BLANK COMMON(2) 000000

      COMMON BLOCKS:

 0003  KEEP2  000047
 0004  KEEP3  000102
 0005  KEEP5  000074
 0006  KEEP16 000031
 0007  CR003  011450

      EXTERNAL REFERENCES (BLOCK, NAME)

 0010  STNNYQ
 0011  NERR45
 0012  NERR36

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

 0001  000051 1000L  0001  000023 I35G   0007  000070 ADDR   0007  000006 AFRQ   0007  011616 AML
 0007  011617 AMP    0007  000152 APHA   0007  011620 BIG    0005 L 000071 BOTH   0006 L 000022 DEBUG
 0007 L 011642 DECR  0007  011621 DF     0003  000002 DHFILT 0004  000063 DP     0007 L 011643 DPI
 0007 L 011645 ERAM  0007 L 011647 ERGP  0007 L 011644 ERP    0007 L 011646 ERPH   0007  011622 FL
 0007  011623 FR     0007  000000 GAINS  0005 L 000073 GPRINT 0003  000037 HACC   0003  000025 HAST
 0003  000035 HATT   0003  000000 HBLANK 0003  000004 HBLK   0003  000034 HCCW   0003  000033 HCW
 0003  000032 HDEC   0003  000027 HDOT   0003  000006 HESTI  0003  000043 HFGD   0003  000044 HFGN
 0003  000040 HFPD   0003  000045 HFPDO  0003  000042 HFPDN  0003  000041 HFPN   0003  000010 HGENE
 0003  000031 HINC   0003  000005 HKEY   0003  000007 HMATR  0003  000014 HNEW   0003  000046 HNOMI
 0003  000015 HNYQU  0003  000023 HO     0003  000026 HPLUS  0003  000036 HRATE  0003  000011 HRAW
 0003  000012 HRETA  0003  000017 HROLL  0003  000016 HROOT  0003  000013 HSTAN  0003  000030 HSTAR
 0003  000020 HSIC   0003  000021 HS2    0003  000022 HS48   0003  000024 HX     0000 I 000000 I
 0007 L 011624 IM    0000  000002 INJPS  0007  011634 INT    0006  000010 IOPEN  0005  000067 ITHZT
 0006  000001 JOPEN  0006 I 000001 KODE  0004 L 000023 LFLT   0007  011641 LMX    0007  011635 LRPR
 0004 R 000051 MAX   0004 R 000037 MIN   0005 L 000072 MODIFY 0007  011636 MPPP   0007 I 011637 NEXT
 0004 I 000000 NFI   0007 I 000003 NGNPK 0006 L 000027 NOMNAL 0006 L 000030 NOTYET 0007 I 011640 NPPP
 0007 I 000004 NP180 0006  000002 NRCLPL 0006  000003 NRPOLE 0006  000004 NRZERO 0006  000005 NXB
 0006  000006 NXN    0006  000007 NXR    0007 I 000005 NYQPTS 0005  000004 NZT    0007 I 000002 N1AMP
 0007  000316 PAMP   0004  000026 PB     0007  000014 PCPL   0004  000025 PCT    0007  000400 PDIR
 0007  011625 PER    0006  000015 PFAC   0007  000234 PFRQ   0007  011626 PHA    0007  000626 PHAMP
 0007  000710 PHDIR  0007  000544 PNFRQ  0007  011627 PHL    0004  000075 PN     0006  000013 PNOM
 0007  000962 PPHA   0006  000016 PSLOSH 0006  000012 PVAR   0004  000077 P180   0007  011630 RE
 0007  003726 SAVAMP 0007  000772 SAVFRQ 0007  006662 SAVPHA 0007  011631 SMA    0007  011632 STA
 0006  000000 STAGE  0004 L 000101 STNORD 0007  011633 STU    0004 R 000013 STP    0004 R 000001 STR
 0005  000000 SUPERK 0005  000003 TD     0006 L 000017 YESMTX 0004 L 000100 YESNYQ 0006 L 000024 YESPCH
 0006  000020 YESRAW 0006 L 000025 YESRLP 0006 L 000026 YESSRL 0006 L 000021 YESSRP 0005 L 000070 YESZOH
 0005  000002 ZM     0005 R 000001 ZT     0005  000005 ZTVAL


```

```
00101     1*          SUBROUTINE INITAL(*)                                                3784    000000
00102     2*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,       3785    000000
00103     3*         1            HBETA,HSTAN,HNEW,HNOQU,HROOT,HROLL,HSIC,HS2,HS4D,      3786    000000
00103     4*         2            HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,        3787    000000
00103     5*         3            HATT,HRATE,HACC,HEPO,HEPN,HEPD,,HEGO,HEGN,HEPDO,       3788    000000
00103     6*         4            HNOM1                                                  3789    000000
00104     7*          DOUBLE PRECISION HBLANK,DHFILT                                      3790    000000
00105     8*          COMMON/KEEP3/ NFI,STR(10),STP(10),PCT(10),MIN(10),MAX(10),DP(10),   3791    000000
00105     9*         1            PN,PO,P180,YESNYQ,STNORD                               3792    000000
00106    10*          REAL          MIN,MAX                                               3793    000000
00107    11*          LOGICAL       YESNYQ,STNORD                                         3794    000000
00110    12*          COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,      3795    000000
00110    13*         1            MODIFY,GPRINT                                          3796    000000
00111    14*          LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                             3798    000000
00112    15*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,          3799    000000
00112    16*         1            IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                3800    000000
00113    17*         2            YESHTX,YESRAM,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,         3801    000000
00113    18*         3            YESSRL,NOMNAL,NOTYET                                   3802    000000
00113    19*          LOGICAL       YESHTX,YESRAM,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      3803    000000
00114    20*         1            YESSRL,NOMNAL,NOTYET                                   3804    000000
00114    21*          COMMON/CROUD3/ GAINS,NIAMP,NGNPK,NP180,NYQPTS,AFRQ(50),ADIR(50),    3805    000000
00115    22*         1            APHA(50),PFRQ(50),PAMP(50),PDIR(50),PPHA(50),          3806    000000
00115    23*         2            RHFRQ(50),RHAMP(50),RHDIR(50),SAVERQ(1500),            3807    000000
00116    24*         3            SAVAMP(1500),SAVPHA(1500),AML,AMP,BIG,DF,FL,FR,IM,     3808    000000
00116    25*         4            PER,PHA,PHL,RE,SHA,STA,STO,                            3809    000000
00117    26*         5            INT,LRPR,MPPP,NEXT,NPPP,LMX,                           3810    000000
00117    27*         6            DECR,DPI,ERP,ERAM,ERPH,ERGP                            3811    000000
00115    28*          COMPLEX       GAINS                                                 3812    000000
00115    29*          REAL          IM                                                    3813    000000
00117    30*          LOGICAL       DECR,DPI,ERP,ERAM,ERPH,ERGP                           3814    000000
00117    31*     C                                                                        3815    000000
00117    32*     C                                                                        3816    000000
00117    33*     C    INITIALIZATION OF PROGRAM CONSTANTS                                 3817    000000
00117    34*     C                                                                        3818    000000
00120    35*          NEXT = 0                                                            3819    000000
00121    36*          NPPP = 1                                                            3820    000000
00122    37*          NYQPTS = 0                                                          3821    000002
00123    38*          NIAMP = 0                                                           3822    000003
00124    39*          NP180 = 0                                                           3823    000004
00125    40*          NGNPK = 0                                                           3824    000005
00126    41*          ERAM = .FALSE.                                                      3825    000006
00127    42*          ERPH = .FALSE.                                                      3826    000007
00130    43*          ERGP = .FALSE.                                                      3827    000010
00131    44*          ERP  = .FALSE.                                                      3828    000011
00132    45*          IF (STNORD) CALL STNNYQ                                             3829    000012
00132    46*     C                                                                        3830    000012
00132    47*     C                                                                        3831    000012
00132    48*     C  TEST FREQUENCY INTERVALS                                              3832    000012
00136    49*          DO 100 I=1,NFI                                                      3833    000016
00136    50*     C                                                                        3834    000016
00137    51*          IF ((STR(I)*ZT .GT. 1.0) .OR. (STP(I)*ZT .GT. 1.0)) GO TO 1000      3835    000023
00141    52*        100 CONTINUE                                                          3836    000045
00143    53*          RETURN                                                              3837    000045
00143    54*     C                                                                        3838    000045
00143    55*     C                                                                        3839    000045
00143    56*     C  INVALID FREQUENCY INTERVAL FOR THE SAMPLE RATE                        3840    000045
```

```
00143     57*     C                                                    38410     000045
00144     58*     1000 CONTINUE                                        38420     000051
00145     59*          KODE = 24                                       38430     000051
00146     60*          RETURN 1                                        38440     000052
00147     61*          END                                             38450     000076

 END OF COMPILATION:  NO  DIAGNOSTICS.
```

```
)FOR*US# INPEST,F*INPEST
 FOR SEIX-05/23/74-08:33:16 (3,4)

      SUBROUTINE INPEST    ENTRY POINT 000116

      STORAGE USED: CODE(1) 000124; DATA(0) 000016; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003  KEEP1  000026
      0004  KEEP2  000047
      0005  KEEP9  000705
      0006  KEEP16 000031
      0007  CRUD2  000115

      EXTERNAL REFERENCES (BLOCK, NAME)

      0010  NRDU$
      0011  NIO2$
      0012  NIO1$
      0013  NERR4$
      0014  NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001  000016 IOL      0001  000066 IOOOL    0001  000057 I36G     0001  000023 ZOL      0001  000074 ZOOOL
0000  000001 500F     0000  000002 501F     0007  000024 ANORM    0007  000000 BCD      0006 L 000022 DEBUG
0004 D 000002 DHFILT  0005 C 000226 EA       0004  000037 HACC     0004  000025 HAST     0004  000035 HATT
0004 D 000000 HBLANK  0004  000004 HBLK     0004  000034 HCCW     0004  000033 HCW      0004  000032 HDEC
0004  000027 HDOT     0004  000006 HESTI    0004  000043 HFGD     0004  000044 HFGN     0004  000040 HFPD
0004  000045 HFPDD    0004  000042 HFPDN    0004  000041 HFPN     0004  000010 HGENE    0004  000031 HINC
0004  000005 HKEY     0004  000007 HMATR    0004 R 000014 HNEW2    0004  000046 HNOMI    0004  000015 HNYQU
0004  000023 HO       0004  000026 HPLUS    0004  000036 HRATE    0004  000011 HRAW     0004 R 000012 HRETA
0004  000017 HROLL    0004  000016 HROOT    0004  000013 HSTAN    0004  000030 HSTAR    0004  000020 HSIC
0004  000021 HS2      0004  000022 HS48     0004  000024 HX       0000 I 000000 I        0007  000075 II
0001  000006 IMPS     0006  000010 IOPEN    0007  000101 JJ       0006  000011 JOPEN    0005  000572 KD
0007  000105 KK       0006 I 000001 KODE     0006 L 000023 LFLT     0003  000000 MAXIT    0003  000001 MXEIG
0003  000024 MXEIGT   0003 I 000023 MXEST    0003  000002 MXFRM    0003  000003 MXNBM    0003  000004 MXNCOF
0003  000005 MXNCT    0003  000025 MXNCV    0003  000006 MXNE     0003  000007 MXNEQ    0003  000010 MXNF1
0003  000011 MXNG     0003  000012 MXNPH    0003  000013 MXNPP    0003  000014 MXNQPT   0003  000015 MXNSM
0003  000016 MXNSP    0003  000017 MXNTM    0003  000020 MXNV     0003  000021 MXNZT    0003  000022 MXPOLY
0005  000455 NA       0005  000454 NEIG     0007  000074 NFILT    0005  000457 NI       0006 L 000027 NOMNAL
0006 L 000035 NOTYET   0005  000456 NR      0006  000002 NRCLPL   0006  000003 NRPOLE   0006  000004 NRZERO
0007  000071 NTMP0    0007  000072 NTMP0C   0007  000073 NTMP1C   0006  000005 NXB      0006  000006 NXN
0006  000007 NXR      0007 R 000020 OPTINP   0007  000003 OPTTYP   0007  000021 OPT1     0007  000022 OPT2
0007  000023 OPT3     0007  000011 PC       0006  000014 PCPL     0006  000015 PFAC     0006  000013 PNOM
0006  000016 PSLOSH   0006  000012 PVAR     0007  000017 REQEST   0005 C 000000 ROOT     0006  000000 STAGE
0007  000006 TEMP     0007  000025 TEMP0    0005  000045 TEMP1    0007  000045 VAL      0007  000011 VFILT
0006 L 000017 YESMTX  0006 L 000024 YESPCH   0006 L 000020 YESRAW   0006 L 000025 YESRLP   0006 L 000026 YESSRL
0006 L 000021 YESSRP
```

```
00101                 SUBROUTINE INPEST(*)                                          3846   000000
00103                 COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,  3847   000000
00104                1              MXNF1,MXNG,MXNPH,MXNPP,MXNOPT,MXNSM,MXNSP,MXNTM,  3848   000000
00105                2              MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV              3849   000000
00106                 COMMON/KEEP2/ HBLANK,DHFILT,HBL,HKEY,HESTI,HMATR,HGENE,HRA,     3850   000000
00107                1              HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HS1C,HS2,HS4B,  3851   000000
00108                2              HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,    3852   000000
00109                3              HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,   3853   000000
00110                4              HNO4                                               3854   000000
00111                 COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)           3856   000000
00112                 COMPLEX       ROOT,EA                                          000000
00113                 COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,       3859   000000
00114                1              IUPEN,JOPEN,PVAR,PNO4,PCPL,PFAC,PSLOSH,            3860   000000
00115                2              YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,     3861   000000
00116                3              YESSRL,NOMNAL,NOTYET                               3862   000000
00117                 LOGICAL       YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,     3863   000000
00118                1              YESSRL,NOMNAL,NOTYET                               3864   000000
00119                 COMMON/CRUD2/ HCD(3),OPTTYP(3),TEMP(3),VFILT(6),REQEST,OPTINP,   3865   000000
00120                1              OPT1,OPT2,OPT3,ANORM,TEMP0(16),TEMP1(16),VAL(4),   3866   000000
00121                2              NTMP0,NTMP0C,NTMP1C,NFILT,II(4),JJ(4),KK(4),PC(4)  3867   000000
00122                 REAL*8 HBLANK,DHFILT                                            000000
00123                C                                                                3869   000000
00124                C                                                                3870   000000
00125                C  INPUT ESTIMATES TO THE CONTINUOUS MATRIX                       3871   000000
00126                C                                                                3872   000000
00127 *DIAGNOSTIC*   THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00128                      IF (OPTINP.EQ.HRETA) GO TO 10                              3873   000000
00129 *DIAGNOSTIC*   THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00130                      IF (OPTINP.EQ.HNEW) GO TO 20                               3874   000003
00131                C                                                                3875   000003
00132                C                                                                3876   000003
00133                C  CAN NOT INTERPRET THE EXTIMATE REQUEST                        3877   000003
00134                C                                                                3878   000003
00135                      KODE = 25                                                  3879   000007
00136                      RETURN 1                                                   3880   000011
00137                C                                                                3881   000011
00138                C                                                                3882   000011
00139                C  RETAIN PREVIOUS CASE ESTIMATES                                3883   000011
00140                C                                                                3884   000011
00141                 10 CONTINUE                                                     3885   000016
00142                      IF (NA.EQ.0) GO TO 1000                                    3886   000016
00143                      RETURN                                                     3887   000017
00144                C                                                                3888   000017
00145                C                                                                3889   000017
00146                C  INPUT NEW SET OF ESTIMATES                                    3890   000017
00147                C                                                                3891   000017
00148                 20 CONTINUE                                                     3892   000023
00149                      READ(5,500) NA                                             3893   000023
00150                 500  FORMAT(I5)                                                3894   000030
00151                      IF (NA.LT.1 .OR. NA.GT.MXEST) GO TO 2000                   3895   000030
00152                      READ(5,501) (EA(I),I=1,NA)                                 3896   000046
00153                 501  FORMAT(6E12.5)                                            3897   000062
00154                      RETURN                                                     3898   000062
00155                C                                                                3899   000062
00156                C                                                                3900   000062
00157                C  NO ESTIMATES IN PREVIOUS CASE                                 3901   000062
```

```
00141    55*     C
00144    56*       1000 CONTINUE
00145    57*            KODE = 26
00146    58*            RETURN 1
00146    59*     C
00146    60*     C
00146    61*     C  NUMBER OF ESTIMATES OUT OF RANGE
00146    62*     C
00147    63*       2000 CONTINUE
00151    64*            KODE = 27
00151    65*            RETURN 1
00152    66*            END
```

```
39020    000062
39030    000066
39040    000066
39050    000067
39060    000067
39070    000067
39080    000067
39090    000067
39100    000074
39110    000074
39120    000075
39130    000123
```

END OF COMPILATION: 2 DIAGNOSTICS.

SUBROUTINE INPMTX ENTRY POINT 000077

STORAGE USED: CODE(1) 000103; DATA(0) 000004; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003	KEEP2	000047
0004	KEEP16	000031
0005	CR002	000115

EXTERNAL REFERENCES (BLOCK, NAME)

0006	GENMTX
0007	RANMTX
0010	NERR45
0011	NERR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001		000026	10L	0001		000056	1000L	0001		000033	20L	0001		000061	2000L
0001		000044	30L	0005		000024	ANORM	0005		000000	BCO	0004	L	000022	DEBUG
0003	D	000002	DHFILT	0003		000037	HACC	0003		000025	HAST	0003		000035	HATT
0003	D	000000	HBLANK	0003		000004	HBLK	0003		000034	HCCW	0003		000033	HCW
0003		000032	HDEC	0003		000027	HDOT	0003		000006	HESTI	0003		000043	HFGD
0003		000040	HFPD	0003		000045	HFPDD	0003		000042	HFPDN	0003		000041	HFPN
0003		000031	HINC	0003		000005	HKEY	0003		000007	HMATR	0003		000014	HNEW
0003	R	000015	HNYQU	0003		000023	HO	0003		000026	HPLUS	0003		000036	HRATE
0003	R	000012	HRETA	0003		000017	HROLL	0003		000016	HROOT	0003		000013	HSTAN
0003		000020	HSTAR	0003		000021	HS1C	0003		000022	HS2	0003		000024	HX
0005		000101	JJ	0004		000011	JOPEN	0005		000375	II	0000		000000	INJPS
0005		000105	KK	0004		000001	KODE	0004	L	000023	LFLT	0005		000074	NFILT
0004	L	000027	NOMNAL	0004		000030	NOTYET	0004		000002	NRCLPL	0004		000003	NRPOLE
0004		000004	NRZERO	0005		000071	NTMPD	0005		000072	NTMPDC	0005		000073	NTMPIC
0004		000005	NXB	0004		000006	NXN	0004		000007	NXR	0005	R	000020	OPTINP
0005	R	000021	OPTT	0005	R	000022	OPTZ	0005		000023	OPT3	0005		000111	PC
0004		000014	PCPL	0004		000015	PFAC	0004	R	000013	PNOM	0004		000016	PSLOSH
0004	R	000012	PVAR	0005		000017	REQEST	0004		000000	STAGE	0005		000006	TEMP
0005		000025	TEMPO	0005		000045	TEMP1	0005		000005	VAL	0005		000011	VFILT
0004	L	000017	YESMTX	0004	L	000024	YESPCH	0004	L	000025	YESRAW	0004	L	000026	YESRLP
0004	L	000021	YESSRL	0004	L	000020	YESSRP								

```
00101    1*          SUBROUTINE INPMTX(*)                                          43250  000000
00101    2*           COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,  43260  000000
00101    3*     1     HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HS1C,HS2,HS4B,              43270  000000
00101    4*     2     HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,               43280  000000
00101    5*     3     HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,               43290  000000
00101    6*     4     HNOM1                                                         43300  000000
00101    7*           DOUBLE PRECISION HBLANK,DHFILT                                       000000
00105    8*           COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,    43320  000000
00105    9*     1     ICPEN,JOPEN,PVAR,NOM1,PCPL,PFAC,PSLOSH                        43330  000000
```

```
00105    10*          2                YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      43340   000000
00105    11*          3                YESSRL,NOMNAL,NOTYET                                43350   000000
00106    12*          LOGICAL          YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      43360   000000
00106    13*          1                YESSRL,NOMNAL,NOTYET                                43370   000000
00107    14*          COMMON/CRUD2/ BCD(3),OPTTYP(3),TEMP(3),VFILT(6),REQEST,OPTINP,       43380   000000
00107    15*          1                OPT1,OPT2,OPT3,ANORM,TEMPD(16),TEMPI(16),VAL(4),    43390   000000
00107    16*          2                NTMPD,NTMPDC,NTMPIC,NFILT,II(4),JJ(4),KK(4),PC(4)   43400   000000
00107    17*     C                                                                        43420   000000
00107    18*     C                                                                        43430   000000
00107    19*     C  DETERMINE MATRIX INPUT METHOD                                         43440   000000
00107    20*     C                                                                        43450   000000
00110    21*          PVAR = OPT1                                                         43460   000000
00111    22*          PNOM = OPT2                                                         43470   000001
00112  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00112    23*          IF (OPTINP.EQ.HNOMI) GO TO 10                                       43480   000003
00114  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00114    24*          IF (OPTINP.EQ.HGENE) GO TO 20                                       43490   000007
00116  *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00116    25*          IF (OPTINP.EQ.HRAW) GO TO 30                                        43500   000013
00116    26*     C                                                                        43510   000013
00116    27*     C                                                                        43520   000013
00116    28*     C  CAN NOT INTERPRET MATRIX INPUT METHOD                                 43530   000013
00116    29*     C                                                                        43540   000013
00120    30*          KODE = 32                                                           43550   000017
00121    31*          RETURN 1                                                            43560   000021
00121    32*     C                                                                        43570   000021
00121    33*     C                                                                        43580   000021
00121    34*     C  USE THE NOMINAL MATRIX INPUT IN A PREVIOUS CASE                       43590   000021
00121    35*     C                                                                        43600   000021
00122    36*       10 CONTINUE                                                           43610   000026
00123    37*          IF (.NOT.YESMTX) GO TO 2000                                         43620   000026
00125    38*          RETURN                                                              43630   000027
00125    39*     C                                                                        43640   000027
00125    40*     C                                                                        43650   000027
00125    41*     C  INPUT THE NOMINAL MATRIX IN GENERAL FORM                              43660   000027
00125    42*     C                                                                        43670   000027
00127    43*       20 CONTINUE                                                           43680   000033
00127    44*          YESMTX = .TRUE.                                                     43690   000033
00130    45*          YESRAW = .FALSE.                                                    43700   000034
00131    46*          CALL GENMTX(*1000)                                                  000035
00132    47*          RETURN                                                              43720   000040
00132    48*     C                                                                        43730   000040
00132    49*     C                                                                        43740   000040
00132    50*     C  CREATE NOMINAL MATRIX FROM RAW DATA                                   43750   000040
00132    51*     C                                                                        43760   000040
00132    52*       30 CONTINUE                                                           43770   000044
00134    53*          YESMTX = .TRUE.                                                     43780   000044
00135    54*          YESRAW = .TRUE.                                                     43790   000045
00136    55*          CALL RAWMTX(*1000)                                                  000046
00137    56*          RETURN                                                              43810   000051
00137    57*     C                                                                        43820   000051
00137    58*     C                                                                        43830   000051
00137    59*     C  ERROR IN CREATING THE NOMINAL MATRIX                                  43840   000051
00137    60*     C                                                                        43850   000051
00140    61*     1000 CONTINUE                                                           43860   000055
00141    62*          RETURN 1                                                            43870   000055
00141    63*     C                                                                        43880   000055
```

```
00141          C
00141    165*  C  PREVIOUS MATRIX REQUESTED BUT NONE EXISTS
00141    166*  C
00142    167*  2000 CONTINUE
00143    168*       KODE = 33
00144    169*       RETURN 1
00145    170*       END
```

```
43890  000055
43900  000055
43910  000055
43920  000061
43930  000061
43940  000062
43950  000102
```

END OF COMPILATION: 1 DIAGNOSTICS.

```
RFOR,USN F,INPNYQ,F.INPNYQ
FOR SE1x-05/23/74-08:33:35 (2.3)

      SUBROUTINE INPNYQ    ENTRY POINT 000145

      STORAGE USED; CODE(1) 000154; DATA(D) 000017; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003  KEEP1  000026
      0004  KEEP2  000047
      0005  KEEP3  000102
      0006  KEEP16 000031
      0007  CRUD2  000115

      EXTERNAL REFERENCES (BLOCK, NAME)

      0010  NROU$
      0011  NLO2$
      0012  N101$
      0013  NERR4$
      0014  NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001  000030 10L      0001  000116 1000L    0001  000102 152G     0001  000035 20L      0001  000124 2000L
0001  000043 30L      0000  000001 500F     0000  000002 501F     0007  000024 ANORM    0007  000000 BCD
0004 L 000022 DEBUG    0004 D 000002 DHFILT  0005 R 000063 DP      0004  000037 HACC     0004  000025 HAST
0004  000035 HATT     0004 D 000000 HBLANK   0004  000004 HBLK     0004  000034 HCCW     0004  000033 HCW
0004  000032 HDEC     0004  000027 HDOT     0004  000006 HESTJ    0004  000043 HFGD     0004  000044 HFGN
0004  000040 HFPD     0004  000045 HFPDD    0004  000042 HFPDN    0004  000041 HFPN     0004  000010 HGENE
0004  000031 HINC     0004  000005 HKEY     0004  000007 HMATR    0004 R 000014 HNEW     0004  000046 HNOM1
0004  000015 HNYQU    0004  000023 HO       0004  000026 HPLUS    0004  000036 HRATE    0004  000011 HRAW
0004 R 000012 HRETA    0004  000017 HROLL    0004  000016 HROOT    0004 R 000013 HSTAN    0004  000030 HSTAR
0004  000020 HS1C     0004  000021 HS2      0004  000022 HS4B     0004  000024 HX       0000 I 000000 I
0007  000075 II       0000  000007 INJP$    0006  000010 IOPEN    0007  000101 JJ       0006  000011 JOPEN
0007  000105 KK       0004 I 000001 KODE     0004 L 000023 LFLT     0005 R 000051 MAX      0003  000000 MAXIT
0005 R 000037 MID      0003  000001 MXEIG    0003  000024 MXEIGT   0003  000023 MXEST    0003  000002 MXFRM
0003  000003 MXNDM    0003  000004 MXNCOF   0003  000005 MXNCT    0003  000025 MXNCV    0003  000006 MXNE
0003  000007 MXNEQ    0003 I 000010 MXNFI    0003  000011 MXNG     0003  000012 MXNPH    0003  000013 MXNPP
0003  000014 MXNOPT   0003  000015 MXNSM    0003  000016 MXNSP    0003  000017 MXNTM    0003  000020 MXNV
0003  000021 MXNZT    0003  000022 MXPOLY   0005 I 000000 NEI      0007  000074 NEILT    0006 L 000027 NOMVAL
0004 I 000030 NOTYET   0006  000002 NPCLPL   0006  000003 NPPOLE   0006  000004 NRZERO   0007  000071 NTMP0
0007  000072 NTMP0C   0007  000073 NTMP1C   0006  000005 NXN      0006  000006 NXN      0006  000007 NXR
0007 R 000020 OPTINP   0007  000003 OPTTYP   0007 R 000021 OPT1     0007 R 000022 OPT2     0007 R 000023 OPT3
0005 R 000076 PB       0007  000111 PC       0006  000014 PCPL     0005 R 000025 PCT      0006  000015 PFAC
0005 R 000075 PN       0006  000013 PNOM     0006  000016 PSLOSH   0006  000012 PVAR     0005 R 000077 P180
0007  000017 REQEST   0006  000000 STAGE    0005 I 000101 STNDRD   0005 R 000013 STP      0005 R 000001 STR
0007  000006 TEMP     0007  000025 TEMP0    0007  000045 TEMP1    0007  000065 VAL      0007  000011 VFILT
0006 L 000017 YESNTX   0006 L 000100 YESNYQ   0006 L 000024 YESPCH   0006 L 000020 YESRAW   0006 L 000025 YESRLP
0006 L 000034 YESSRL   0006 L 000021 YESSRP
```

```
00101          SUBROUTINE INPNYQ(*)                                          443960 000000
00102     2*   COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,  43970 000000
00103     3*   1             MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,  43980 000000
00104     4*   2             MXNV,MXNZT,NXPOLY,NXEST,NXEIGT,MXNCV              43990 000000
00105     5*   COMMON/KEEP2/ BLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAC,    44000 000000
00106     6*   1             HRETA,HSTAN,HUNEW,HNYQU,HROOT,HRDLL,HS1C,HS2,HS4B, 44010 000000
00107     7*   2             HO,HX,HAST,HPLUS,HDOT,HSTAR,HIPC,HDEC,HCW,HCCW,    44020 000000
00108     8*   3             HATT,HRATE,HACC,HFPO,HFPH,HFPDC,HFGD,HFGN,HFPDO,   44030 000000
00109     9*   4             HNOX)                                             44040 000000
00110    10*   DOUBLE PRECISION BLANK,DHFILT                                   44050 000000
00111    11*   COMMON/KEEP3/ HFI,STR(10),STP(10),PC(10),MIN(10),MAX(10),DP(10), 44060 000000
00112    12*   1             PN,PA,P180,YESNYQ,STNDRD                          44070 000000
00113    13*   REAL          MIN,MAX                                           44080 000000
00114    14*   LOGICAL       YESNYQ,STNDRD                                     44090 000000
00115    15*   COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,      44100 000000
00116    16*   1             IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,           44110 000000
00117    17*   2             YESMTX,YESRAC,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,    44120 000000
00118    18*   3             YESSRL,NOMNAL,NOTYET                              44130 000000
00119    19*   LOGICAL       YESMTX,YESRAC,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,    44140 000000
00120    20*   1             YESSRL,NOMNAL,NOTYET                              44150 000000
00121    21*   COMMON/CRUD2/ ACC(3),OPTTYP(3),TEMP(3),VFILT(6),REQEST,OPTINP,  44160 000000
00122    22*   1             OPT1,OPT2,OPT3,ANORM,TEMP0(16),TEMP1(16),VAL(4),  44170 000000
00123    23*   2             NTMP0,NTMP0C,NTMP1C,NFILT,II(4),JJ(4),KK(4),PC(4) 44180 000000
00124    24*   C                                                               44190 000000
00125    25*   C                                                               44210 000000
00126    26*   C  INPUT NYQUIST DATA                                           44220 000000
00127    27*   C                                                               44230 000000
00128    28*         PN = OPT1                                                 44240 000000
00129    29*         PA = OPT2                                                 44250 000001
00130    30*         P180 = OPT3                                               44260 000003
00131    31*   *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00131          IF (OPTINP.EQ.HRETA) GO TO 10                                   44270 000005
00132    32*   *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00132          IF (OPTINP.EQ.HSTAN) GO TO 20                                   44280 000011
00133    33*   *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00133          IF (OPTINP.EQ.HUNEW) GO TO 30                                   44290 000015
00134    34*   C                                                               44300 000015
00135    35*   C  CAN NOT INTERPRET NYQUIST INPUT OPTION                       44310 000015
00136    36*   C                                                               44320 000015
00137    37*         KODE = 34                                                 44330 000021
00138    38*         RETURN 1                                                  44340 000023
00139    39*   C                                                               44350 000023
00140    40*   C                                                               44360 000023
00141    41*   C                                                               44370 000023
00142    42*   C - RETAIN PREVIOUS CASE NYQUIST DATA                           44380 000023
00143    43*   C                                                               44390 000023
00144    44*   10 CONTINUE                                                     44400 000030
00145    45*         IF (.NOT.YESNYQ) GO TO 1000                               44410 000030
00146    46*         RETURN                                                    44420 000031
00147    47*   C                                                               44430 000031
00148    48*   C                                                               44440 000031
00149    49*   C  EMPLOY STANDARD NYQUIST DATA                                 44450 000031
00150    50*   C                                                               44460 000031
00151    51*   20 CONTINUE                                                     44470 000035
00152    52*         YESNYQ = .TRUE.                                           44480 000035
00153    53*         STNDRD = .TRUE.                                           44490 000036
```

```
00136    54*          RETURN                                          44500    000037
00136    55*    C                                                     44510    000037
00136    56*    C                                                     44520    000037
00136    57*    C  READ NEW NYQUIST DATA                              44530    000037
00136    58*    C                                                     44540    000037
00137    59*       30 CONTINUE                                        44550    000043
00140    60*          YESNYQ = .TRUE.                                 44560    000043
00141    61*          STNDRD = .FALSE.                                44570    000044
00142    62*          READ(5,500) NFI                                 44580    000045
00145    63*      500 FORMAT(I5)                                      44590    000053
00146    64*          IF (NFI.LT.1 .OR. NFI.GT.MXNFI) GO TO 2000      44600    000053
00150    65*          READ(5,501) (STR(I),STP(I),RCT(I),MIN(I),MAX(I),DP(I),I=1,NFI)  44610    000071
00163    66*      501 FORMAT(5E12.5,4X,A1)                            44620    000112
00164    67*          RETURN                                          44630    000112
00164    68*    C                                                     44640    000112
00164    69*    C                                                     44650    000112
00164    70*    C  PREVIOUS NYQUIST DATA REQUESTED BUT NONE EXISTS    44660    000112
00164    71*    C                                                     44670    000112
00165    72*     1000 CONTINUE                                        44680    000116
00165    73*          KODE = 35                                       44690    000116
00167    74*          RETURN 1                                        44700    000117
00167    75*    C                                                     44710    000117
00167    76*    C                                                     44720    000117
00167    77*    C  NUMBER OF NYQUIST FREQUENCY INTERVALS IS OUT OF RANGE  44730    000117
00167    78*    C                                                     44740    000117
00170    79*     2000 CONTINUE                                        44750    000124
00171    80*          KODE = 36                                       44760    000124
00172    81*          RETURN 1                                        44770    000125
00173    82*          END                                             44780    000153

      END OF COMPILATION:      3 DIAGNOSTICS.
```

SUBROUTINE INPUT ENTRY POINT 000046

STORAGE USED: CODE(1) 000052 DATA(0) 000041 BLANK COMMON(2) 000000

COMMON BLOCKS:

0003 KEEP2 000047
0004 KEEP16 000031

EXTERNAL REFERENCES (BLOCK, NAME)

0005 AFTVAR
0006 PREVAR
0007 DATA
0010 NERR45
0011 NERR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001		000007	IOL	0001	000016	IOOL	0001	000032	IOOOL	0001	000026	ZOOL	0004	L	000022	DEBUG
0003	D	000002	DHFILT	0003	000037	HACC	0003	000025	HAST	0003	000035	HATT	0003	D	000000	HBLANK
0003	R	000009	HBLK	0003	000034	HCCN	0003	000033	HCN	0003	000032	HDEC	0003		000027	HDOT
0003		000006	HESTI	0003	000043	HFGD	0003	000044	HFGN	0003	000040	HFPD	0003		000045	HFPDD
0003		000042	HFPDN	0003	000041	HFPN	0003	000010	HGENE	0003	000031	HINC	0003		000005	HKEY
0003		000007	HMATR	0003	000014	HNEW	0003	000046	HNOMI	0003	000015	HNYQU	0003		000023	HO
0003		000026	HPLUS	0003	000036	HRATE	0003	000011	HRAW	0003	000012	HRETA	0003		000017	HROLL
0003		000016	HROOT	0003	000013	HSTAN	0003	000030	HSTAR	0003	000020	HSIC	0003		000021	HS2
0003		000022	HS4B	0003	000024	HX	0000	000000	INJPS	0004	000010	IOPEN	0004		000011	JOPEN
0004		000001	KODE	0004	L 000023	LFLT	0004	L 000027	NOMNAL	0004	L 000030	NOTYET	0004		000002	NRCLPL
0004		000003	NRPOLE	0004		000004	NRZERO	0004	000005	NXB	0004	000006	NXN	0004		000007	NXR
0004		000014	PCPL	0004		000015	PFAC	0004	000013	PNOM	0004	000016	PSLOSH	0004	R	000012	PVAR
0004		000000	STAGE	0004	L 000017	YESHTX	0004	L 000024	YESPCH	0004	L 00002D	YESRAW	0004	L	000025	YESRLP
0004		000026	YESSRL	0004	L 000021	YESSRP										

```
00101    1*          SUBROUTINE INPUT(*,*)                                          465400  000000
00102    2*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,  465500  000000
00103    3*         1      HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS4B,        465600  000000
00104    4*         2      HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,          465700  000000
00105    5*         3      HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,         465800  000000
00106    6*         4      HNOMI                                                    465900  000000
00107    7*          DOUBLE PRECISION HBLANK,DHFILT                                         000000
00108    8*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,     466100  000000
00109    9*         1      IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                  466200  000000
00110   10*         2      YESHTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,           466300  000000
00111   11*         3      YESSRL,NOMNAL,NOTYET                                     466400  000000
00112   12*          LOGICAL YESHTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,         466500  000000
00113   13*         1      YESSRL,NOMNAL,NOTYET                                     466600  000000
```

```
00104    14*      C                                                          4667₀  000000
00104    15*      C                                                          4668₀  000000
00104    16*      C  RETURN PARAMETER VALUES AFTER VARIATION                 4669₀  000000
00104    17*      C                                                          4670₀  000000
00107  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00137    18*            IF (PVAR.EQ.HBLK) GO TO 100                          4671₀  000000
00111    19*               CALL AFTVAR(*1000)                                       000003
00111    20*      C                                                          4673₀  000003
00111    21*      C                                                          4674₀  000003
00111    22*      C  INCORPORATE NEXT VARIATION                              4675₀  000003
00111    23*      C                                                          4676₀  000003
00113    24*         10 CONTINUE                                             4677₀  000007
00113    25*               CALL PREVAR(*1000,*100)                                  000007
00114    26*               RETURN                                           4679₀  000012
00114    27*      C                                                          4680₀  000012
00114    28*      C                                                          4681₀  000012
00114    29*      C  INPUT DATA FOR THE NEXT CASE                            4682₀  000012
00114    30*      C                                                          4683₀  000012
00115    31*         100 CONTINUE                                            4684₀  000016
00116    32*               CALL DATA(*1000,*200,$10)                                000016
00117    33*               RETURN                                           4686₀  000022
00117    34*      C                                                          4687₀  000022
00117    35*      C                                                          4688₀  000022
00117    36*      C  NO MORE DATA CARDS                                      4689₀  000022
00117    37*      C                                                          4690₀  000022
00121    38*         200 CONTINUE                                            4691₀  000026
00121    39*               RETURN 2                                         4692₀  000026
00121    40*      C                                                          4693₀  000026
00121    41*      C                                                          4694₀  000026
00121    42*      C  INPUT ERROR                                             4695₀  000026
00121    43*      C                                                          4696₀  000026
00123    44*         1000 CONTINUE                                           4697₀  000032
00123    45*               RETURN 1                                         4698₀  000032
00123    46*               END                                              4699₀  000051
```

END OF COMPILATION: 1 DIAGNOSTICS.

```
BFOR,US F.INTI,F.INTI
FOR SEIX-05/23/74-08:33:57 (2,3)

      FUNCTION INTI          ENTRY POINT 000026

     STORAGE USED: CODE(1) 000036; DATA(C) 000010; BLANK COMMON(2) 000000

      EXTERNAL REFERENCES (BLOCK, NAME)

        0003    NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

        0001    000010 IL       0000    000001 INJPS    0000 R 000000 INTI

00101      1*            REAL FUNCTION INTI(Y1,Y,Y2)                          47000   000000
00101     *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00101      2*              IF(Y2.EQ.Y1) GO TO 1                               47010   000000
00105      3*              INTI = 0.                                          47020   000003
00106      4*              RETURN                                             47030   000004
00107      5*            1 INTI = (Y - Y1)/(Y2 - Y1)                          47040   000010
00111      6*              RETURN                                             47050   000015
00111      7*              END                                                47060   000035

      END OF COMPILATION:       1 DIAGNOSTICS.
```

```
?FOR,US F.INT2,F.INT2
FOR.SE1x-05/23/Z4-08:34:04 (2,3)

      FUNCTION INT2          ENTRY POINT 000014

      STORAGE USED: CODE(1) 0o00016; DATA(0) 0o00005; BLANK COMMON(2) 000000

      EXTERNAL REFERENCES (BLOCK, NAME)

          0003    NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

          0000  000001 IN/P$      0000 R 000000 INT2


 00101      1*          REAL FUNCTION INT2(X1,X2,D)                    4707o  000000
 00102      2*          INT2 = X1 + D*(X2-X1)                          4708o  000000
 00103      3*          RETURN                                         4709o  000004
 00104      4*          END                                            4710o  000015

      END OF COMPILATION:      NO DIAGNOSTICS.
```

96-9

```
BFOR,US F.KCALC,F*KCALC
FOR-GEIX-05/23/74-08:34:13 (2.3)

SUBROUTINE KCALC      ENTRY POINT 000064

 STORAGE USED: CODE(1) 000076; DATA(0) 000060; BLANK COMMON(2) 000000

 COMMON BLOCKS:

 1003  KEEP5  000074
 1004  KEEP19 000005

 EXTERNAL REFERENCES (BLOCK, NAME)

 1005  PEVAL
 1006  DET
 1007  CDV$
 1010  N=DUS
 1011  NIO2$
 1012  NERR3$

 STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

 1001  000047 IL      0001  000055 4L      0000  000010 600F    0000  000030 601F    0003 L 000071 BOTH
 1000 C 000004 FSF     0000 C 000002 FSP     0003 L 000073 GPRINT  0000  000054 INJP$   0003  000067 ITHZT
 1004 L 000000 LGAIN   0004 L 000001 LPHASE  0004 L 000002 LPOLES  0004 L 000003 LSDRL   0004 L 000004 LZEROS
 1003 L 000072 MODIFY  0000 I 000007 NSF     0000 I 000006 NSP     0003  000004 NZT     0003  000000 SUPERK
 1003   000003 TD      0003 L 000070 YESZOH  0000 C 000000 Z       0003  000002 ZM      0003  000001 ZT
 1003   000005 ZTVAL

00101     1*        SUBROUTINE KCALC(BASE,NPOWER)                                  471110  000000
00101     2*        COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,  471210  000000
00103     3*       1          MODIFY,GPRINT                                        471310  000000
00104     4*        LOGICAL     YESZOH,BOTH,MODIFY,GPRINT                          471510  000000
00105     5*        COMMON/KEEP19/LGAIN,LPHASE,LPOLES,LSDRL,LZEROS                         000000
00106     6*        LOGICAL     LGAIN,LPHASE,LPOLES,LSDRL,LZEROS                           000000
00107     7*        COMPLEX     BASE                                               471610  000000
00111     8*        COMPLEX     Z,FSP,FSF                                                  000000
00111     9*      C                                                                471810  000000
00111    10*      C                                                                471910  000000
00111    11*      C  PROGRAM CODING                                                472010  000000
00111    12*      C                                                                472110  000000
00111    13*        Z = (0.,0.)                                                            000000
00112    14*        IF (LSDRL) Z=(1.,0.)                                                   000001
00114    15*        CALL PEVAL(Z,FSP,NSP)                                          472310  000005
00115    16*        CALL DET(Z,FSF,NSF)                                            472410  000012
00116    17*        NPOWER = NSF - NSP                                             472510  000017
00117    18*        BASE = FSF/FSP                                                 472610  000022
00121    19*        IF (.NOT.GPRINT) RETURN                                        472710  000027
00122    20*        IF(NPOWER.EQ.0)  GO TO 1                                       473610  000034
```

```
00129      21*          WRITE(6,600) BASE,NPOWER                          47290   000036
00130      22*      600 FORMAT('2(/),20X,'CHARACTERISTIC'/20X,'POLYNOMIAL'/20X,   47300   000045
00130      23*         1'LEADING COEFFICIENT',1PE15.7,E18.7,5X,I5)        47310   000045
00131      24*          GO TO 4                                           47320   000045
00132      25*        1 CONTINUE                                          47330   000047
00133      26*          WRITE(6,601) BASE                                 47340   000047
00135      27*      601 FORMAT('2(/),20X,'CHARACTERISTIC'/20X,'POLYNOMIAL'/20X,   47350   000055
00135      28*         1'LEADING COEFFICIENT',1PE15.7,E18.7)              47360   000055
00137      29*        4 CONTINUE                                          47370   000055
00140      30*          RETURN                                            47380   000055
00141      31*          END                                               47390   000075

      END OF COMPILATION:  NO DIAGNOSTICS.
```

```
PRORUS F[illegible]MIT,F*LIMIT
FOR SE]x-05/23/74-09:34:23 [2,3]

      SUBROUTINE LIMIT      ENTRY POINT 000042

      STORAGE USED: CODE(1) 000501 DATA(0) 000194 BLANK COMMON(2) 000000

      COMMON BLOCKS:

      3003   KEEP1  000026

      EXTERNAL REFERENCES (BLOCK, NAME)

      0004   NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0000 000000 INJPS     0003 I 000000 MAXIT     0003 I 000001 MXEIG     0003 I 000024 MXEIGT    0003 I 000023 MXEST
      0003 I 000002 MXFRM   0003   000003 MXNBM     0003 I 000004 MXNCOF    0003 I 000005 MXNCT     0003   000025 MXNCV
      0003 I 000006 MXNE    0003 I 000007 MXNEQ     0003 I 000010 MXNFI     0003 I 000011 MXNG      0003 I 000012 MXNPH
      0003 I 000013 MXNPP   0003 I 000014 MXNQPT    0003   000015 MXNSM     0003 I 000016 MXNSP     0003   000017 MXNTM
      0003 I 000020 MXNV    0003 I 000021 MXNZT     0003   000022 MXPOLY
```

```
00101    1*          SUBROUTINE LIMIT                                    47400   000000
00103    2*          COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,   47410   000000
00103    3*         1                MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,   47420   000000
00103    4*         2                MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV              47430   000000
00103    5*      C                                                       47440   000000
00103    6*      C                                                       47450   000000
00103    7*      C  SET PROGRAM LIMITS                                   47460   000000
00103    8*      C                                                       47470   000000
00104    9*          MXNEQ = 60                                          47480   000000
00106   10*          MAXIT = 50                                          47490   000001
00106   11*          MXNZT = 50                                          47500   000003
00107   12*          MXNCT = 10                                          47510   000004
00111   13*          MXEIG = 75                                          47520   000006
00111   14*          MXNPH = 50                                          47530   000010
00112   15*          MXFRM = 10                                          47540   000011
00113   16*          MXNG = 50                                           47550   000012
00114   17*          MXNFI = 10                                          47560   000013
00115   18*          MXNQPT = 1500                                       47570   000014
00115   19*          MXNE = 1000                                         47630   000016
00117   20*          MXNCOF = 1500                                       47640   000020
00121   21*          MXEIGT = 1000                                       47650   000021
00121   22*          MXNPP = 1000                                        47660   000022
00121   23*          MXNSP = 50                                          47670   000023
00121   24*          MXNV = 100                                          47680   000024
00121   25*          MXNST = 75                                          47690   000026
00125   26*          RETURN                                              47700   000027
00125   27*          END                                                 47710   000047
```

END OF COMPILATION: NO DIAGNOSTICS.

```
@FOR,USW C*,L*TZRO,F*MLTZRO
FOR (EIX=05/23/74=08:34:30 (2,3)

      SUBROUTINE MLTZRO     ENTRY POINT 000705

      STORAGE USED: CODE(1) 000733; DATA(6) 000217; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003  KEEP5   000074
      0004  KEEP14  000031
      0005  KEEP16  000031
      0006  KEEP21  001133
      0007  CRUD1   000025
      0010  CRUD2   001215
      0011  CRUD4   000002

      EXTERNAL REFERENCES (BLOCK, NAME)

      0012  NWDU$
      0013  NIO2$
      0014  NERR4$
      0015  NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001  000657 101OL     0001  000120 150L     0001  000223 200L     0001  000335 250L     0001  000461 300L
0000  000000 600F     0000  000011 611F     0000  000026 612F     0000  000040 613F     0000  000054 614F
0000  000065 621F     0000  000102 622F     0000  000113 623F     0000  000130 624F     0000  000141 631F
0000  000157 632F     0007 R 000024 AT2      0003 L 000071 BOTH     0005 L 000022 DEBUG    0004  000025 DEG
0004 R 000012 D0      0004 R 000013 D1       0004 R 000014 D2       0004 R 000015 D3       0004 R 000016 D4
0004 C 000010 FIFTEN  0004  000017 FIFTY     0004 C 000006 FOUR     0003 L 000073 GPRINT   0004 C 000000 HALF
0000  000171 INJP$    0005  000010 IOPEN     0003  000067 ITHZT     0005  000011 JOPEN     0005 I 000001 KODE
0005 L 000023 LFLT    0010 I 001214 LOCO     0006 I 000704 LOCO2    0010 I 001213 LOCN     0006 I 000703 LOCNZ
0003 L 000072 MODIFY  0010 I 001077 NDCPER   0004 I 001020 NDCZ     0011  000000 NEQZ     0010 I 000764 NNCPER
0006 I 000705 NNCZ    0005 L 000027 NOMNAL   0005 L 000030 NOTYET   0005  000002 NRCLPL   0005  000003 NRPOLE
0005  000004 NRZERO   0010 I 001212 NUMPOL   0006  000702 NUMZ     0005  000005 NXB      0005  000006 NXN
0005  000007 NXR      0003  000004 NZT      0004  000026 N1       0004  000027 N2       0004  000030 N3
0004 C 000002 ONE     0005  000014 PCPL     0010 R 000310 PD       0006 R 000341 PDZ      0005  000015 PFAC
0004  000020 PI       0004  000021 PI2      0010 R 000000 PNN      0005  000013 PNOM     0006 R 000000 PNZ
0005  000016 PSLOSH   0005  000012 PVAR     0004  000024 RADDEG   0007 R 000020 ROH1     0007 R 000021 ROH2
0007 R 000022 ROH3    0007 R 000023 ROH4     0004  000023 RPI      0007 R 000007 R1       0007 R 000010 R2
0007 R 000011 R3      0007 R 000012 R4       0003  000022 SMALL    0005  000000 STAGE    0003 R 000000 SUPERK
0003  000003 TD       0004 C 000004 TWO      0005 L 000017 YESMTX   0005 L 000024 YESPCH   0005 L 000020 YESRAW
0005 L 000025 YESRLP  0005 L 000026 YESSRL   0005 L 000021 YESSRP   0011 L 000001 YESZM    0003 L 000070 YESZOH
0003  000002 ZM       0007 R 000013 ZOH1     0007 R 000014 ZOH2     0007 R 000015 ZOH3     0007 R 000016 ZOH4
0007 R 000017 ZOH5    0003 R 000001 ZT       0003  000005 ZTVAL    0007 R 000000 Z1       0007 R 000001 Z2
0007 R 000002 Z3      0007 R 000003 Z4       0007 R 000004 Z5       0007 R 000005 Z6       0007 R 000006 Z7

00101      1*          SUBROUTINE MLTZRO(*,M,N,RTR,RTI,RSR,RSI)                          000000
```

```
00103     2*          COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,
00104     3*         1              MODIFY,GPRINT
00104     4*          LOGICAL       YESZOH,BOTH,MODIFY,GPRINT
00105     5*          COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,
00106     6*         1              PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3
00106     7*          COMPLEX       HALF,ONE,TWO,FOUR,FIFTEN
00107     8*          COMMON/KEEP15/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,
00107     9*         1              IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,
00107    10*         2              YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,
00107    11*         3              YESSRL,NOMNAL,NOTYET
00111    12*          LOGICAL       YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,
00111    13*         1              YESSRL,NOMNAL,NOTYET
00111    14*          COMMON/KEEP21/PNZ(225),PDZ(225),NUMZ,LOCNZ,LOCDZ,NNCZ(75),NDCZ(75)
00112    15*          COMMON/CRUD1/ Z1,Z2,Z3,Z4,Z5,Z6,Z7,R1,R2,R3,R4,
00112    16*         1              ZOH1,ZOH2,ZOH3,ZOH4,ZOH5,
00112    17*         2              ROH1,ROH2,ROH3,ROH4,AT2
00113    18*          COMMON/CRUD2/ PNN(200),PD(300),NNCPER(75),NDCPER(75),
00113    19*         1              NUMPOL,LOCN,LOCD
00114    20*          COMMON/CRUD4/ YESZ,YESZM
00115    21*          LOGICAL       YESZM
00115    22*     C
00115    23*     C
00115    24*     C  PRINT THE ZERO ROOT AND ITS RESIDUE
00115    25*     C
00116    26*               IF (GPRINT) WRITE(6,600) RTR,RTI,RSR,RSI
00120    27*           600 FORMAT('1///10X,1PE12.5,5X,E12.5,10X,'RESIDUE',2(5X,E12.5))
00120    28*     C
00120    29*     C
00120    30*     C  DETERMINE WHETHER A/S, A/S**2, OR A/S**3 IS BEING CONSIDERED
00120    31*     C
00120    32*               IF (M.EQ.2) GO TO 200
00131    33*               IF (N.EQ.3) GO TO 300
00131    34*     C
00131    35*     C
00131    36*     C  A/S TO BE CONVERTED
00131    37*     C
00132    38*               Z1=RSR*SUPERK
00133    39*               Z2 = D0
00134    40*               Z3 = D1
00135    41*               Z4 = -D1
00136    42*               IF (YESZOH) GO TO 150
00136    43*     C
00136    44*     C
00136    45*     C  PRINT Z COEFFICIENTS
00136    46*     C
00140    47*               IF (GPRINT) WRITE(6,611) Z1,Z2,Z3,Z4
00147    48*           611 FORMAT(/10X,'RESIDUE / S',3DX,'Z NUM',2(5X,1PE12.5)/53X,'DEN',
00147    49*         1              2(5X,E12.5))
00150    50*               R1=Z1
00151    51*               R2=Z1
00152    52*               R3 = D2
00152    53*     C
00152    54*     C
00152    55*     C  PRINT R COEFFICIENTS
00152    56*     C
00154    57*               IF (GPRINT) WRITE(6,612) R1,R2,R3
00161    58*           612 FORMAT(/51X,'R NUM',5X,1PE12.5,5X,E12.5/53X,'DEN',5X,E12.5)
```

```
5429₀  000000
5430₀  000000
5432₀  000000
5433₀  000000
5434₀  000000
       000000
5437₀  000000
5438₀  000000
5439₀  000000
5440₀  000000
5441₀  000000
5442₀  000000
       000000
5443₀  000000
5444₀  000000
5445₀  000000
       000000
5450₀  000000
5452₀  000000
5453₀  000000
5454₀  000000
5455₀  000000
5456₀  000000
5457₀  000000
5458₀  000000
5459₀  000013
5460₀  000013
5461₀  000013
5462₀  000013
5463₀  000013
5464₀  000013
5465₀  000017
5466₀  000017
5467₀  000017
5468₀  000017
5469₀  000017
5470₀  000023
5471₀  000026
5472₀  000030
5473₀  000032
5474₀  000033
5475₀  000033
5476₀  000033
5477₀  000033
5478₀  000033
5479₀  000035
5480₀  000050
5481₀  000050
5482₀  000050
5483₀  000052
5484₀  000053
5485₀  000053
5486₀  000053
5487₀  000053
5488₀  000053
5489₀  000055
5490₀  000067
```

```
00161     59*      C
00161     60*      C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS                    54910   000067
00161     61*      C                                                                54920   000067
00161     62*      C  NUMERATOR COEFFICIENTS                                        54930   000067
00162     63*            LOCN = LOCN + 1                                            54940   000067
00163     64*            PNN(LOCN) = R1                                             54950   000067
00164     65*            LOCN = LOCN + 1                                            54960   000072
00165     66*            PNN(LOCN) = R2                                             54970   000075
00165     67*      C  DENOMINATOR COEFFICIENTS                                      54980   000077
00166     68*            LOCD = LOCD + 1                                            54990   000077
00167     69*            PD (LOCD) = R3                                             55000   000102
00170     70*            NNCPER(NUMPOL) = 2                                         55010   000106
00171     71*            NDCPER(NUMPOL) = 1                                         55020   000110
00172     72*            RETURN                                                     55030   000112
00172     73*      C                                                                55040   000114
00172     74*      C                                                                55050   000114
00172     75*      C  ZERO ORDER HOLD SELECTED                                      55060   000114
00172     76*      C                                                                55070   000114
00173     77*        150 CONTINUE                                                   55080   000114
00174     78*            ZOH1=Z1                                                    55090   000120
00175     79*            ZOH2 = 0)                                                  55100   000120
00175     80*      C                                                                55110   000121
00175     81*      C                                                                55120   000121
00175     82*      C  PRINT ZOH COEFFICIENTS                                        55130   000121
00175     83*      C                                                                55140   000121
00176     84*            IF (GPRINT) WRITE(6,613) ZOH1,ZOH2                         55150   000121
00203     85*        613 FORMAT('1',10X,'RESIDUE / S',2AX,'ZOH NUM',5X,1PE12.5/53X,  55160   000123
00203     86*           1       'DEN',5X,E12.5)                                     55170   000134
00204     87*            ROH1=ZOH1                                                  55180   000134
00205     88*            ROH2=ZOH2                                                  55190   000134
00205     89*      C                                                                55200   000136
00205     90*      C                                                                55210   000136
00205     91*      C  PRINT ROH COEFFICIENTS                                        55220   000136
00205     92*      C                                                                55230   000136
00206     93*            IF (GPRINT) WRITE(6,614) ROH1,ROH2                         55240   000136
00213     94*        614 FORMAT('1',49X,'ROH NUM',5X,1PE12.5/53X,'DEN',5X,E12.5)    55250   000140
00213     95*      C                                                                55260   000151
00213     96*      C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS                    55270   000151
00213     97*      C                                                                55280   000151
00213     98*      C  NUMERATOR COEFFICIENTS                                        55290   000151
00214     99*            LOCN = LOCN + 1                                            55300   000151
00215    100*            PNN(LOCN) = ROH1                                           55310   000151
00215    101*      C  DENOMINATOR COEFFICIENTS                                      55320   000154
00216    102*            LOCD = LOCD + 1                                            55330   000154
00217    103*            PD (LOCD) = ROH2                                           55340   000157
00221    104*            NNCPER(NUMPOL) = 1                                         55350   000162
00221    105*            NDCPER(NUMPOL) = 1                                         55360   000165
00222    106*            IF (.NOT.YESZH) RETURN                                     55370   000167
00224    107*            NUMZ = NUMZ + 1                                            55380   000170
00225    108*            NNCZ(NUMZ) = 1                                                     000175
00226    109*            LOCNZ = LOCNZ + 1                                                  000200
00227    110*            PNZ(LOCNZ) = ZOH1                                                  000203
00231    111*            NDCZ(NUMZ) = 1                                                     000205
00231    112*            LOCDZ = LOCDZ + 1                                                  000210
00232    113*            PDZ(LOCDZ) = ZOH2                                                  000212
00233    114*            RETURN                                                     55410   000215
00233    115*      C                                                                55420   000217
```

```
00230*      C
00231*      C  A/S**2 TO BE CONVERTED
00232*      C
00233*        200 CONTINUE
00234*            Z1=SUPERK*RSR*ZT
00235*            Z2 = D0
00236*            Z3 = D1
00237*            Z4 = -D2
00238*            Z5 = D1
00239*            IF (Y.GE.ZOH) GO TO 250
00240*      C
00241*      C
00242*      C  PRINT Z COEFFICIENTS
00243*      C
00244*            IF (GPRINT) WRITE(6,621) Z1,Z2,Z3,Z4,Z5
00245*          621 FORMAT(/10X,'RESIDUE / S**2',,27X,'Z NUM',,2(5X,1PE12.5)/53X,
00246*         1      'DEN',3(5X,E12.5))
00247*            R1=Z1
00248*            R2 = D0
00249*            R3=-Z1
00250*            R4=Z3-Z4+Z5
00251*      C
00252*      C
00253*      C  PRINT R COEFFICIENTS
00254*      C
00255*            IF (GPRINT) WRITE(6,622) R1,R2,R3,R4
00256*          622 FORMAT(/51X,'R NUM',3(5X,1PE12.5)/53X,'DEN',5X,E12.5)
00257*      C
00258*      C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS
00259*      C
00260*      C  NUMERATOR COEFFICIENTS
00261*            LOCN = LOCN + 1
00262*            PN(LOCN) = R1
00263*            LOCN = LOCN + 1
00264*            PN(LOCN) = R2
00265*            LOCN = LOCN + 1
00266*            PN(LOCN) = R3
00267*      C  DENOMINATOR COEFFICIENTS
00268*            LOCD = LOCD + 1
00269*            PD(LOCD) = R4
00270*            NNCPER(NUMPOL) = 3
00271*            NDCPER(NUMPOL) = 1
00272*            RETURN
00273*      C
00274*      C
00275*      C  ZERO ORDER HOLD SELECTED
00276*      C
00277*        250 CONTINUE
00278*            ZOH1=Z1
00279*            ZOH2 = D1
00280*            ZOH3 = -D1
00281*      C
00282*      C
00283*      C  PRINT ZOH COEFFICIENTS
00284*      C
00285*            IF (GPRINT) WRITE(6,623) ZOH1,ZOH2,ZOH3
00286*          623 FORMAT(/10X,'RESIDUE / S**2',,25X,'ZOH NUM',5X,1PE12.5/53X,
```

```
55430   000217
55440   000217
55450   000217
55460   000223
55470   000223
55480   000226
55490   000230
55500   000232
55510   000234
55520   000235
55530   000235
55540   000235
55550   000235
55560   000235
55570   000237
55580   000253
55590   000253
56000   000253
56010   000255
56020   000257
56030   000260
56040   000260
56050   000260
56060   000260
56070   000260
56080   000264
56090   000277
56100   000277
56110   000277
56120   000277
56130   000277
56140   000277
56150   000302
56160   000305
56170   000307
56180   000312
56190   000314
56200   000314
56210   000317
56220   000323
56230   000325
56240   000327
56250   000331
56260   000331
56270   000331
56280   000331
56290   000331
56300   000335
56310   000335
56320   000336
56330   000340
56340   000340
56350   000340
56360   000340
56370   000341
56390   000353
```

```
00315   173*          1       'DEN',2(5X,E12.5))                                  56000   000353
00316   174*                  ROH1=ZOH1                                           56010   000353
00320   175*                  ROH2=-ZOH1                                          56020   000355
00324   176*                  ROH3=ZOH2-ZOH3                                      56030   000356
00328   177*          C                                                           56040   000356
00329   178*          C                                                           56050   000356
00329   179*          C  PRINT ROH COEFFICIENTS                                   56060   000356
00329   180*          C                                                           56070   000356
00329   181*                  IF (GPRINT) WRITE(6,624) ROH1,ROH2,ROH3             56080   000361
00330   182*           624  FORMAT(/40X,'ROH NUM',2(5X,1PE12.5)/53X,'DEN',5X,E12.5) 56090   000373
00330   183*          C                                                           56100   000373
00330   184*          C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS               56110   000373
00330   185*          C                                                           56120   000373
00330   186*          C  NUMERATOR COEFFICIENTS                                   56130   000373
00331   187*                  LOCN = LOCN + 1                                     56140   000373
00332   188*                  PN(LOCN) = ROH1                                     56150   000376
00333   189*                  LOCN = LOCN + 1                                     56160   000401
00334   190*                  PN(LOCN) = ROH2                                     56170   000403
00337   191*          C  DENOMINATOR COEFFICIENTS                                 56180   000403
00337   192*                  LOCD = LOCD + 1                                     56190   000406
00340   193*                  PD(LOCD) = ROH3                                     56200   000411
00343   194*                  NNCPER(NUMPOL) = 2                                  56210   000414
00343   195*                  NDCPER(NUMPOL) = 1                                  56220   000416
00343   196*                  IF(L.NOT.YESZH) RETURN                              56230   000420
00343   197*                  NUMZ = NUMZ + 1                                             000425
00344   198*                  NNCZ(NUMZ) = 1                                              000430
00345   199*                  LOCNZ = LOCNZ + 1                                           000433
00346   200*                  PNZ(LOCNZ) = ZOH1                                           000435
00347   201*                  NDCZ(NUMZ) = 2                                              000440
00350   202*                  LOCDZ = LOCDZ + 1                                           000442
00351   203*                  PDZ(LOCDZ) = ZOH2                                           000445
00352   204*                  LOCDZ = LOCDZ + 1                                           000450
00353   205*                  PDZ(LOCDZ) = ZOH3                                           000453
00354   206*                  RETURN                                              56260   000455
00354   207*          C                                                           56270   000455
00354   208*          C                                                           56280   000455
00354   209*          C  A/S**3  TO  BE  CONVERTED                                56290   000455
00354   210*          C                                                           56300   000455
00355   211*            300 CONTINUE                                              56310   000461
00356   212*                  IF(L.NOT.YESZOH) GO TO 1010                         56320   000461
00360   213*                  AT2=RSR*ZT*ZT*SUPERK                                56330   000462
00361   214*                  Z1 = AT2/D2                                         56340   000467
00362   215*                  Z2 = AT2/D3                                         56350   000471
00363   216*                  Z3 = D0                                             56360   000472
00364   217*                  Z4 = D1                                             56370   000474
00365   218*                  Z5 = -D3                                            56380   000476
00366   219*                  Z6 = D3                                             56390   000500
00367   220*                  Z7 = -D1                                            56400   000501
00367   221*          C                                                           56410   000501
00367   222*          C                                                           56420   000501
00367   223*          C  ZERO ORDER HOLD SELECTED                                 56430   000501
00367   224*          C                                                           56440   000501
00370   225*                  ZOH1=Z1                                             56450   000502
00371   226*                  ZOH2=Z2                                             56460   000503
00372   227*                  ZOH3 = D1                                           56470   000504
00373   228*                  ZOH4 = -D2                                          56480   000505
00374   229*                  ZOH5 = D1                                           56490   000507
```

```
00374   230*      C
00374   231*      C
00374   232*      C  PRINT ZOH COEFFICIENTS
00374   233*      C
00375   234*            IF (GPRINT) WRITE(6,631) ZOH1,ZOH2,ZOH3,ZOH4,ZOH5
00405   235*        631 FORMAT(1/10X,'RESIDUE / S**3',25X,'ZOH NUM',2(5X,1PE12.5)/53X,
00405   236*           1     'DEN',3(5X,E12.5))
00406   237*            ROH1=ZOH1+ZOH2
00407   238*            ROH2 = -D2*ZOH2
00410   239*            ROH3 = D0
00411   240*            ROH4 = D4
00411   241*      C
00411   242*      C
00411   243*      C  PRINT ROH COEFFICIENTS
00411   244*      C
00412   245*            IF (GPRINT) WRITE(6,632) ROH1,ROH2,ROH3,ROH4
00421   246*        632 FORMAT(1/49X,'ROH NUM',3(5X,1PE12.5)/53X,'DEN',5X,E12.5)
00421   247*      C
00421   248*      C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS
00421   249*      C
00421   250*      C  NUMERATOR COEFFICIENTS
00422   251*            LOCN = LOCN + 1
00423   252*            PNN (LOCN) = ROH1
00424   253*            LOCN = LOCN + 1
00425   254*            PNN (LOCN) = ROH2
00426   255*            LOCN = LOCN + 1
00427   256*            PNN (LOCN) = ROH3
00427   257*      C  DENOMINATOR COEFFICIENTS
00430   258*            LOCD = LOCD + 1
00431   259*            PD (LOCD) = ROH4
00432   260*            NDCPER(NUMPOL) = 3
00433   261*            LDCPER(NUMPOL) = 1
00434   262*            IF (.NOT.YESZM) RETURN
00436   263*            NUMZ = NUMZ + 1
00437   264*            NNCZ(NUMZ) = 2
00440   265*            LOCNZ = LOCNZ + 1
00441   266*            PNZ(LOCNZ) = ZOH1
00442   267*            LOCNZ = LOCNZ + 1
00443   268*            PNZ(LOCNZ) = ZOH2
00444   269*            NDCZ(NUMZ) = 3
00445   270*            LOCDZ = LOCDZ + 1
00446   271*            PDZ(LOCDZ) = ZOH3
00447   272*            LOCDZ = LOCDZ + 1
00450   273*            PDZ(LOCDZ) = ZOH4
00451   274*            LOCDZ = LOCDZ + 1
00452   275*            PDZ(LOCDZ) = ZOH5
00453   276*            RETURN
00453   277*      C
00453   278*      C
00453   279*      C  PRINT ERROR MESSAGE *** THREE ZEROS, BUT ZOH NOT REQUESTED
00453   280*      C
00454   281*       1010 CONTINUE
00455   282*            KODE = 47
00456   283*            RETURN 1
00457   284*            END
```

```
56500   000507
56510   000507
56520   000507
56530   000507
56540   000510
56550   000524
56560   000524
56570   000524
56580   000527
56590   000532
56600   000534
56610   000534
56620   000534
56630   000534
56640   000534
56650   000536
56660   000551
56670   000551
56680   000551
56690   000551
56700   000551
56710   000551
56720   000554
56730   000557
56740   000561
56750   000564
56760   000566
56770   000566
56780   000571
56790   000574
56800   000577
56810   000601
56820   000603
        000610
        000613
        000616
        000621
        000624
        000626
        000631
        000633
        000636
        000641
        000643
        000646
        000651
56850   000653
56860   000653
56870   000653
56880   000653
56890   000653
56900   000657
56910   000657
56920   000660
56930   000732
```

```
END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR.US4 F.MODZRO,F.MODZRO
FOR SEIX-05/23/74-08:35:00 (2,3)

SUBROUTINE MODZRO     ENTRY POINT 001046

STORAGE USED: CODE(1) 001074; DATA(0) 000225; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003   KEEPS  000074
0004   KEEP14 000031
0005   KEEP16 000031
0006   KEEP21 001133
0007   CRUD1  000035
0010   CRUD2  001215
0011   CRUD4  000002

EXTERNAL REFERENCES (BLOCK, NAME)

0012   NWDU$
0013   NIO2$
0014   NERR4$
0015   NERR3$

STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001  001020 101OL    0001  000115 150L    0001  000241 200L    0001  000363 250L    0001  000544 300L
0000  000000 600F    0000  000011 611F    0000  000025 612F    0000  000037 613F    0000  000054 614F
0000  000066 621F    0000  000103 622F    0000  000114 623F    0000  000132 624F    0000  000144 631F
0000  000142 632F    0007 R 000033 ANT2    0003 R 000034 AM2T2    0007 R 000032 AT2    0003 L 000071 BOTH
0005  000022 DEBUG   0004  000025 DEG     0004 R 000012 D0      0004 R 000013 D1      0004 R 000014 D2
0004 R 000015 D3     0004 R 000016 D4      0004 C 000010 FIFTEN   0004  000017 FIFTY    0004 C 000006 FOUR
0003 L 000073 GPRINT 0004 C 000000 HALF    0000  000175 IHOPS   0005  000010 IOPEN    0003  000067 ITHZT
0005  000011 JOPEN   0005 I 000001 KODE    0005 L 000023 LFLT    0010 I 001214 LOCD    0006 I 000704 LOCDZ
0010 I 001213 LOCN   0006 I 000703 LOCNZ   0003 L 000072 MODIFY   0010 I 001077 NDCPER   0006 I 001020 NDCZ
0011  000000 NEQZ    0010 I 000764 NNCPER   0006 I 000705 NNCZ    0005 L 000027 NORMAL   0005 L 000030 NOTYET
0005  000002 NRCLPL  0005  000003 NRPOLE   0005  000004 NRZERO   0010 I 001212 NUMPOL   0006 I 000702 NUMZ
0005  000005 NXB     0005  000006 NXN     0005  000007 NXR     0003  000004 NZT     0004  000026 N1
0004  000027 N2      0004  000030 N3      0004 C 000002 ONE     0005  000014 PCPL    0010 R 000310 PD
0006 R 000341 PDZ    0005  000015 PFAC    0004  000020 PI      0004  000021 PI2     0010 R 000000 PNN
0005  000013 PNOM    0006 R 000000 PNZ     0005  000016 PSLOSH   0005  000012 PVAR    0004  000024 RADDEG
0007 R 000024 ROH1   0007 R 000025 ROH2    0007 R 000026 ROH3    0007 R 000027 ROH4    0007 R 000030 ROH5
0007 R 000031 ROH6   0004  000023 RPI     0007 R 000007 RI      0007 R 000010 R2      0007 R 000011 R3
0007 R 000012 R4     0004  000022 SIALL   0005  000000 STAGE   0003 R 000010 SUPERK   0003  000003 TD
0004 C 000004 TEN    0005 L 000017 YESNTX   0005 L 000024 YESPCH   0005 L 000020 YESRAW   0005 L 000025 YESRLP
0005 L 000026 YESSRL 0005 L 000021 YESSRP   0011 L 000001 YESZH    0003 L 000070 YESZOH   0003 R 000002 ZH
0007 R 000013 ZOH1   0007 R 000014 ZOH2    0007 R 000015 ZOH3    0007 R 000016 ZOH4    0007 R 000017 ZOH5
0007 R 000020 ZOH6   0007 R 000021 ZOH7    0007  000022 ZOH8    0007  000023 ZOH9    0003 R 000001 ZT
0003  000005 ZTVAL   0007 R 000000 Z1      0007 R 000001 Z2      0007 R 000002 Z3      0007 R 000003 Z4
0007 R 000004 Z5     0007 R 000005 Z6      0007 R 000006 Z7

801-9
```

```
00100     1*          SUBROUTINE MODZRO(*,M,N,RTR,RTI,RSR,RSI)                        000000
00101     2*          COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,  000000
00102     3*         1            MODIFY,GPRINT                                      000000
00103     4*          LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                       000000
00104     5*          COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,DD,D1,D2,D3,D4,FIFTY,PI, 000000
00105     6*         1            PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                  000000
00106     7*          COMPLEX       HALF,ONE,TWO,FOUR,FIFTEN                        000000
00107     8*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,      000000
00108     9*         1            IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,            000000
00109    10*         2            YESNTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,     000000
00110    11*         3            YESSRL,NOMNAL,NOTYET                               000000
00111    12*          LOGICAL       YESNTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,  000000
00112    13*         1            YESSRL,NOMNAL,NOTYET                               000000
00113    14*          COMMON/KEEP21/PNZ(225),PDZ(225),NOMZ,LOCNZ,LOCDZ,NNCZ(75),NDCZ(75) 000000
00114    15*          COMMON/CRUD1/ Z1,Z2,Z3,Z4,Z5,Z6,Z7,R1,R2,R3,R4,                 000000
00115    16*         1            ZOH1,ZOH2,ZOH3,ZOH4,ZOH5,ZOH6,ZOH7,ZOH8,ZOH9,       000000
00116    17*         2            ROH1,ROH2,ROH3,ROH4,ROH5,ROH6,AMT2,AMT2*AMT2       000000
00117    18*          COMMON/CRUD2/ PNN(200),PD(300),NNCPER(75),NDCPER(75),           000000
00118    19*         1            NUMPOL,LOCN,LOCD                                   000000
00119    20*          COMMON/CRUD4/ NEGZ,YESZM                                        000000
00120    21*          LOGICAL       YESZM                                            000000
00121    22*      C                                                                   000000
00122    23*      C                                                                   000000
00123    24*      C  PRINT THE ZERO ROOT AND ITS RESIDUE                              000000
00124    25*      C                                                                   000000
00125    26*            IF (GPRINT) WRITE(6,600) RTR,RTI,RSR,RSI                      000013
00126    27*          600 FORMAT(////10X,1PE12.5,5X,E12.5,10X,'RESIDUE',2(5X,E12.5))  000013
00127    28*      C                                                                   000013
00128    29*      C                                                                   000013
00129    30*      C  DETERMINE WHETHER A/S, A/S**2, OR A/S**3 IS BEING CONSIDERED     000013
00130    31*      C                                                                   000013
00131    32*            IF (M.EQ.2) GO TO 200                                         000017
00132    33*            IF (M.EQ.3) GO TO 300                                         000017
00133    34*      C                                                                   000017
00134    35*      C                                                                   000017
00135    36*      C  A/S TO BE CONVERTED                                              000017
00136    37*      C                                                                   000023
00137    38*            Z1=RSR*SUPERK                                                 000026
00138    39*            Z2 = D1                                                       000030
00139    40*            Z3 = -D1                                                      000031
00140    41*            IF (YESZOH) GO TO 150                                         000031
00141    42*      C                                                                   000031
00142    43*      C                                                                   000031
00143    44*      C  PRINT Z COEFFICIENTS                                             000031
00144    45*      C                                                                   000033
00145    46*            IF (GPRINT) WRITE(6,611) Z1,Z2,Z3                             000045
00146    47*          611 FORMAT(/10X,'RESIDUE / S',3OX,'Z NUM',5X,1PE12.5/53X,'DEN', 000045
00147    48*         1        2(5X,E12.5))                                           000045
00148    49*            R1=Z1                                                         000047
00149    50*            R2=-Z1                                                        000050
00150    51*            R3 = D2                                                       000050
00151    52*      C                                                                   000050
00152    53*      C                                                                   000050
00153    54*      C  PRINT R COEFFICIENTS                                             000050
00154    55*      C                                                                   000052
00155    56*            IF (GPRINT) WRITE(6,612) R1,R2,R3
```

```
00157    57*       612 FORMAT(/5(1X,'R NUM',5X,1PE12.5,5(5X,E12.5/53X,'DEN',5X,E12.5)
00157    58*     C
00157    59*     C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS
00157    60*     C
00157    61*     C  NUMERATOR COEFFICIENTS
00161    62*           LOCN = LOCN + 1
00161    63*           PN(LOCN) = R1
00162    64*           LOCN = LOCN + 1
00163    65*           PN(LOCN) = R2
00163    66*     C  DENOMINATOR COEFFICIENTS
00164    67*           LOCD = LOCD + 1
00165    68*           PD(LOCD) = R3
00165    69*           NNCPER(NUMPOL) = 2
00167    70*           NDCPER(NUMPOL) = 1
00171    71*           RETURN
00171    72*     C
00171    73*     C
00171    74*     C  ZERO ORDER HOLD SELECTED
00171    75*     C
00171    76*       150 CONTINUE
00172    77*           ZOH1=T1
00173    78*           ZOH2 = 01
00174    79*           ZOH3 = D0
00174    80*     C
00174    81*     C
00174    82*     C  PRINT ZOH COEFFICIENTS
00174    83*     C
00175    84*           IF (GPRINT) WRITE(6,613) ZOH1,ZOH2,ZOH3
00203    85*       613 FORMAT(/10X,'RESIDUE / S',28X,'ZOH NUM',5X,1PE12.5/53X,
00203    86*          1        'DEN',2(5X,E12.5))
00204    87*           ROH1=ZOH1
00205    88*           ROH2=-ZOH1
00206    89*           ROH3 = 01
00207    90*           ROH4 = 01
00207    91*     C
00207    92*     C
00207    93*     C  PRINT ROH COEFFICIENTS
00207    94*     C
00211    95*           IF (GPRINT) WRITE(6,614) ROH1,ROH2,ROH3,ROH4
00217    96*       614 FORMAT(1/49X,'ROH NUM',2(5X,1PE12.5)/53X,'DEN',2(5X,E12.5))
00217    97*     C
00217    98*     C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS
00217    99*     C
00217   100*     C  NUMERATOR COEFFICIENTS
00221   101*           LOCN = LOCN + 1
00221   102*           PN(LOCN) = ROH1
00222   103*           LOCN = LOCN + 1
00223   104*           PN(LOCN) = ROH2
00223   105*     C  DENOMINATOR COEFFICIENTS
00224   106*           LOCD = LOCD + 1
00225   107*           PD (LOCD) = ROH3
00225   108*           LOCD = LOCD + 1
00227   109*           PD (LOCD) = ROH4
00231   110*           NNCPER(NUMPOL) = 2
00231   111*           NDCPER(NUMPOL) = 2
00232   112*           IF (.NOT.YESZM) RETURN
00234   113*           NUMZ = NUMZ + 1
```

```
57560   000064
57570   000064
57580   000064
57590   000064
57600   000064
57610   000064
57620   000067
57630   000072
57640   000074
57650   000074
57660   000077
57670   000103
57680   000105
57690   000107
57700   000111
57710   000111
57720   000111
57730   000111
57740   000111
57750   000115
57760   000115
57770   000116
57780   000120
57790   000120
57800   000120
57810   000120
57820   000120
57830   000122
57840   000134
57850   000134
57860   000134
57870   000136
57880   000137
57890   000141
57900   000141
57910   000141
57920   000141
57930   000141
57940   000142
57950   000155
57960   000155
57970   000155
57980   000155
57990   000155
58000   000155
58010   000160
58020   000163
58030   000165
58040   000165
58050   000170
58060   000173
58070   000176
58080   000200
58090   000203
58100   000205
58110   000206
        000213
```

```
00235     114*            NNCZ(NUMZ) = 1                                                  000216
00236     115*            LOCNZ = LOCNZ + 1                                               000221
00237     116*            PNZ(LOCNZ) = Z1                                                 000223
00240     117*            NDCZ(NUMZ) = 1                                                  000226
00241     118*            LOCDZ = LOCDZ + 1                                               000230
00242     119*            PDZ(LOCDZ) = D1                                                 000233
00243     120*            RETURN                                                  58140   000235
00243     121*      C                                                             58150   000235
00243     122*      C                                                             58160   000235
00243     123*      C  A/S**2 TO BE CONVERTED                                     59170   000235
00243     124*      C                                                             58180   000235
00244     125*        200 CONTINUE                                                58190   000241
00245     126*            Z1=RSR*ZM*ZT*SUPERK                                     58200   000241
00246     127*            Z2 = SUPERK*RSR*ZT*(D1-ZM)                              58210   000245
00247     128*            Z3 = D1                                                 58220   000254
00250     129*            Z4 = -D2                                                58230   000256
00251     130*            Z5 = D1                                                 58240   000260
00252     131*            IF (YESZOH) GO TO 250                                   58250   000261
00252     132*      C                                                             58260   000261
00252     133*      C                                                             58270   000261
00252     134*      C  PRINT Z COEFFICIENTS                                       58280   000261
00252     135*      C                                                             58290   000261
00253     136*            IF (GPRINT) WRITE(A,621) Z1,Z2,Z3,Z4,Z5                 58300   000263
00261     137*        621 FORMAT(/10X,'RESIDUE / S**2',27X,'Z NUM',2(5X,1PE12.5)/53X,  58310   000277
00261     138*           1       'DEN',3(5X,E12.5))                               58320   000277
00265     139*            R1=Z1+Z2                                                58330   000277
00265     140*            R2 = -D2*Z2                                             58340   000302
00267     141*            R3=Z2-Z1                                                58350   000305
00271     142*            R4 = D1                                                 58360   000310
00271     143*      C                                                             58370   000310
00271     144*      C                                                             58380   000310
00271     145*      C  PRINT R COEFFICIENTS                                       58390   000310
00271     146*      C                                                             58400   000310
00271     147*            IF (GPRINT) WRITE(A,622) R1,R2,R3,R4                    58410   000312
00301     148*        622 FORMAT(/51X,'R NUM',3(5X,1PE12.5)/53X,'DEN',5X,E12.5)   58420   000325
00303     149*      C                                                             58430   000325
00303     150*      C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS                 58440   000325
00303     151*      C                                                             58450   000325
00303     152*      C  NUMERATOR COEFFICIENTS                                     58460   000325
00303     153*            LOCN = LOCN + 1                                         58470   000325
00303     154*            PNN (LOCN) = R1                                 58480   000330
00304     155*            LOCN = LOCN + 1                                         58490   000333
00304     156*            PNN (LOCN) = R2                                 58500   000335
00305     157*            LOCN = LOCN + 1                                         58510   000340
00305     158*            PNN (LOCN) = R3                                 58520   000342
00307     159*      C  DENOMINATOR COEFFICIENTS                                   58530   000342
00307     160*            LOCD = LOCD + 1                                         58540   000345
00311     161*            PD (LOCD) = R4                                  58550   000351
00311     162*            NNCPER(NUMPOL) = 3                              58560   000353
00312     163*            NDCPER(NUMPOL) = 1                                      58570   000355
00313     164*            RETURN                                          58580   000357
00313     165*      C                                                             58590   000357
00313     166*      C                                                             58600   000357
00313     167*      C  ZERO ORDER HOLD SELECTED                                   58610   000357
00313     168*      C                                                             58620   000357
00313     169*        250 CONTINUE                                                58630   000363
00313     170*            ZOH1=Z1                                                 58640   000363
```

```
00316   171          ZOH2=Z2                                                  5865₀   000364
00317   172*         ZOH3 = D1                                                5866₀   000366
00321   173*         ZOH4 = -D1                                               5867₀   000370
00321   174*         ZOH5 = D0                                                5868₀   000371
00321   175*     C                                                            5869₀   000371
00321   176*     C                                                            5870₀   000371
00321   177*     C  PRINT ZOH COEFFICIENTS                                    5871₀   000371
00321   178*     C                                                            5872₀   000371
00322   179*         IF (GPRINT) WRITE(6,623) ZOH1,ZOH2,ZOH3,ZOH4,ZOH5        5873₀   000373
00332   180*       623 FORMAT(/10X,'RESIDUE / S**2',25X,'ZOH NUM',25X,1PE12.5/53X,  5874₀   000407
00332   181*      1        'DEN',35X,E12.5)                                    5875₀   000407
00333   182*         ROH1=ZOH1+ZOH2                                           5876₀   000407
00334   183*         ROH2 = -D2*ZOH2                                          5877₀   000412
00335   184*         ROH3=ZOH2-ZOH1                                           5878₀   000415
00336   185*         ROH4 = D2                                                5879₀   000420
00337   186*         ROH5 = D2                                                5880₀   000422
00337   187*     C                                                            5881₀   000422
00337   188*     C                                                            5882₀   000422
00337   189*     C  PRINT ROH COEFFICIENTS                                    5883₀   000422
00337   190*     C                                                            5884₀   000422
00340   191*         IF (GPRINT) WRITE(6,624) ROH1,ROH2,ROH3,ROH4,ROH5        5885₀   000423
00356   192*       624 FORMAT(/49X,'ROH NUM',35X,1PE12.5/53X,'DEN',25X,E12.6)  5886₀   000437
00356   193*     C                                                            5887₀   000437
00356   194*     C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS                5888₀   000437
00356   195*     C                                                            5889₀   000437
00356   196*     C  NUMERATOR COEFFICIENTS                                    5890₀   000437
00361   197*         LOCN = LOCN + 1                                          5891₀   000437
00362   198*         PN(LOCN) = ROH1                                          5892₀   000442
00363   199*         LOCN = LOCN + 1                                          5893₀   000445
00365   200*         PN(LOCN) = ROH2                                          5894₀   000447
00365   201*         LOCN = LOCN + 1                                          5895₀   000452
00365   202*         PN(LOCN) = ROH3                                          5896₀   000454
00365   203*     C  DENOMINATOR COEFFICIENTS                                  5897₀   000454
00365   204*         LOCD = LOCD + 1                                          5898₀   000457
00366   205*         PD (LOCD) = ROH4                                         5899₀   000462
00367   206*         LOCD = LOCD + 1                                          5900₀   000465
00367   207*         PD (LOCD) = ROH5                                         5901₀   000467
00368   208*         NNCPER(NUMPOL) = 3                                       5902₀   000472
00369   209*         NDCPER(NUMPOL) = 2                                       5903₀   000474
00365   210*         IF (.NOT.YESZN) RETURN                                   5904₀   000476
00367   211*         NUMZ = NUMZ + 1                                                  000503
00370   212*         NNCZ(NUMZ) = 2                                                   000506
00371   213*         LOCNZ = LOCNZ + 1                                                000511
00372   214*         PNZ(LOCNZ) = Z1                                                  000514
00373   215*         LOCNZ = LOCNZ + 1                                                000517
00375   216*         PNZ(LOCNZ) = Z2                                                  000521
00375   217*         NDCZ(NUMZ) = 2                                                   000524
00377   218*         LOCDZ = LOCDZ + 1                                                000525
00377   219*         PDZ(LOCDZ) = D1                                                  000530
00400   220*         LOCDZ = LOCDZ + 1                                                000533
00401   221*         PDZ(LOCDZ) = ZOH4                                                000536
00402   222*         RETURN                                                   5907₀   000540
00402   223*     C                                                            5908₀   000540
00402   224*     C                                                            5909₀   000540
00402   225*     C  5/5**3                                                    5910₀   000540
00402   226*     C                                                            5911₀   000540
00403   227*       300 CONTINUE                                               5912₀   000544
```

```
00406   228*          IF (.NOT.YESZOH) GO TO 1010
00406   229*          AT2=RSR*ZT+ZT*SUPER*K
00407   230*          ANT2=AT2*ZM
00411   231*          AM2T2=ANT2*ZM
00411   232*          Z1 = AM2T2/D2
00412   233*          Z2 = (D2*ANT2+AT2-D2*AM2T2)/D2
00413   234*          Z3 = (AT2-D2*ANT2+AM2T2)/D2
00415   235*          Z4 = D1
00415   236*          Z5 = -D3
00416   237*          Z6 = D3
00417   238*          Z7 = -D1
00417   239*     C
00417   240*     C
00417   241*     C    ZERO ORDER HOLD SELECTED
00417   242*     C
00421   243*          ZOH1=Z1
00421   244*          ZOH2=Z2
00422   245*          ZOH3=Z3
00423   246*          ZOH4 = D1
00425   247*          ZOH5 = -D2
00425   248*          ZOH6 = D1
00426   249*          ZOH7 = D0
00426   250*     C
00427   251*     C
00427   252*     C    PRINT ZOH COEFFICIENTS
00427   253*     C
00427   254*          IF (GPRINT) WRITE(6,631) ZOH1,ZOH2,ZOH3,ZOH4,ZOH5,ZOH6,ZOH7
00441   255*      631 FORMAT(/10X,'RESIDUE / S**3',25X,'ZOH NUM',3(5X,1PE12.5)/53X,
00441   256*         1        'DEN',4(5X,E12.5))
00442   257*          ROH1=ZOH1+ZOH2+ZOH3
00443   258*          ROH2=ZOH1-ZOH2-D3*ZOH3
00445   259*          ROH3=-ZOH1-ZOH2+D3*ZOH3
00445   260*          ROH4=ZOH2-ZOH1-ZOH3
00446   261*          ROH5 = D4
00447   262*          ROH6 = D4
00447   263*     C
00447   264*     C
00447   265*     C    PRINT ROH COEFFICIENTS
00447   266*     C
00451   267*          IF (GPRINT) WRITE(6,632) ROH1,ROH2,ROH3,ROH4,ROH5,ROH6
00461   268*      632 FORMAT(/49X,'ROH NUM',4(5X,1PE12.5)/53X,'DEN',2(5X,E12.5))
00461   269*     C
00461   270*     C    THE FOLLOWING IS USED FOR NYQUIST ANALYSIS
00461   271*     C
00461   272*     C    NUMERATOR COEFFICIENTS
00462   273*          LOCN = LOCN + 1
00463   274*          PNN (LOCN) = ROH1
00464   275*          LOCN = LOCN + 1
00465   276*          PNN (LOCN) = ROH2
00466   277*          LOCN = LOCN + 1
00467   278*          PNN (LOCN) = ROH3
00470   279*          LOCN = LOCN + 1
00471   280*          PNN (LOCN) = ROH4
00471   281*     C    DENOMINATOR COEFFICIENTS
00472   282*          LOCD = LOCD + 1
00473   283*          PD (LOCD) = ROH5
00474   284*          LOCD = LOCD + 1
```

```
59130   000544
59140   000545
59150   000552
59160   000554
59170   000556
59180   000560
59190   000571
59200   000576
59210   000600
59220   000602
59230   000603
59240   000603
59250   000603
59260   000603
59270   000603
59280   000604
59290   000605
59300   000606
59310   000607
59320   000610
59330   000612
59340   000613
59350   000613
59360   000613
59370   000613
59380   000613
59390   000615
59400   000633
59410   000633
59420   000633
59430   000640
59440   000647
59450   000651
59460   000654
59470   000656
59480   000656
59490   000656
59500   000656
59510   000656
59520   000657
59530   000674
59540   000674
59550   000674
59560   000674
59570   000674
59580   000674
59590   000677
59600   000702
59610   000704
59620   000707
59630   000711
59640   000714
59650   000716
59660   000716
59670   000721
59680   000724
59690   000727
```

```
00475     285*          PD (LOCD) = ROH6                                    59700   000731
00476     286*          NNCPER(NUMPOL) = 4                                  59710   000734
00477     287*          NDCPER(NUMPOL) = 2                                  59720   000736
00500     288*          IF (.NOT.YESZN) RETURN                              59730   000740
00502     289*          NUMZ = NUMZ + 1                                             000745
00503     290*          NNCZ(NUMZ) = 3                                              000750
00505     291*          LOCNZ = LOCNZ + 1                                           000753
00505     292*          PNZ(LOCNZ) = Z1                                             000756
00506     293*          LOCNZ = LOCNZ + 1                                           000761
00507     294*          PNZ(LOCNZ) = Z2                                             000763
00510     295*          LOCNZ = LOCNZ + 1                                           000766
00511     296*          PNZ(LOCNZ) = Z3                                             000770
00512     297*          NDCZ(NUMZ) = 3                                              000773
00513     298*          LOCDZ = LOCDZ + 1                                           000774
00514     299*          PDZ(LOCDZ) = ZOH4                                           000777
00515     300*          LOCDZ = LOCDZ + 1                                           001002
00516     301*          PDZ(LOCDZ) = ZOH5                                           001004
00517     302*          LOCDZ = LOCDZ + 1                                           001007
00520     303*          PDZ(LOCDZ) = ZOH6                                           001012
00521     304*          RETURN                                              59760   001014
00521     305*     C                                                        59770   001014
00521     306*     C                                                        59780   001014
00521     307*     C  PRINT ERROR MESSAGE *** THREE ZEROS, BUT ZOH NOT REQUESTED 59790   001014
00521     308*     C                                                        59800   001014
00522     309*       1010 CONTINUE                                          59810   001020
00523     310*            KODE = 48                                         59820   001020
00524     311*            RETURN 1                                          59830   001021
00525     312*            END                                               59840   001073

      END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
@FOR,USN F,MZTRAN,F,MZTRAN
FOR SEIX-05/23/74-08:35:32-L2,31

SUBROUTINE MZTRAN    ENTRY POINT 001224

STORAGE USED: CODE(1) 001250; DATA(0) 000160; BLANK COMMON(2) 000000

COMMON BLOCKS:

 0003  KEEP5  000074
 0004  KEEP8  000342
 0005  KEEP14 000031
 0006  KEEP20 000027
 0007  KEEP21 001433
 0010  CRU01  000045
 0011  CRU02  001215
 0012  CRU04  000002

EXTERNAL REFERENCES (BLOCK, NAME)

 0013  NDDU$
 0014  NIO2$
 0015  EXP
 0016  SIN
 0017  COS
 0020  CSQRT
 0021  CABS
 0022  NERR3$

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

 0001  000061 IOL      0001  000365 IOL      0001  000574 IIOL     0001  000722 ISOL     0001  000155 5OL
 0000  000000 600F     0000  000011 601F     0000  000022 602F     0000  000033 603F     0000  000045 604F
 0000  000057 605F     0000  000070 606F     0000  000101 607F     0000  000113 608F     0003 L 000071 BOTH
 0005  000025 DEG      0005 R 000012 D0       0005 R 000013 D1       0005 R 000014 D2       0005 R 000015 D3
 0005 R 000016 D4      0005 C 000000 FST7     0005 C 000010 FIFTEN   0005  000017 FIFTY    0005 C 000006 FOUR
 0003 L 000073 GPRINT  0005 C 000000 HALF     0000  000130 INJP$     0003  000067 ITHZT    0011 I 001214 LOCD
 0007 I 000704 LOCD2   0011 I 001213 LOCN     0007 I 000703 LOCN2    0003 L 000072 MODIFY   0011 I 001077 NDCPER
 0007 I 001020 NDC7    0012  000000 NEQZ     0006 I 000226 NESTZ    0011 I 000764 NNCPER   0007 I 000705 NNCZ
 0011 I 001212 NUMPOL  0007 I 000702 NUMZ     0004 I 000341 NZPOLE   0003  000004 NZT      0005  000026 N1
 0005  000027 N2       0005  000030 N3       0005 C 000002 ONE      0011 R 000310 PD       0007 R 000341 PDZ
 0005  000023 PI       0005  000021 PI2      0011 R 000000 PNH      0007 R 000000 PNZ      0005  000024 RADDEG
 0010 R 000035 ROH1    0010 R 000036 ROH2     0010 R 000037 ROH3     0010 R 000040 ROH4     0010 R 000041 ROH5
 0010 R 000042 ROH6    0010 R 000043 ROH7     0005  000023 RPI      0010 R 000020 R1       0010 R 000021 R2
 0010 R 000022 R3      0010 R 000023 R4       0010 R 000024 R5       0010 R 000025 R6       0005  000022 SMALL
 0003 R 000000 SUPERK  0003  000003 TD       0010 C 000002 TESM     0005 C 000004 TWO      0010 R 000006 T1
 0010 R 000007 T2      0010 R 000010 T3       0010 R 000011 T4       0010 R 000012 T5       0010 C 000000 VAL
 0012 L 000001 YESZM   0003 L 000070 YESZOH   0003 R 000002 ZM       0004 R 000226 ZMAG     0010 R 000026 ZOH1
 0010 R 000027 ZOH2    0010 R 000030 ZOH3     0010 R 000031 ZOH4     0010 R 000032 ZOH5     0010 R 000033 ZOH6
 0010 R 000034 ZOH7    0004 C 000000 ZPOLE    0003 R 000001 ZT       0003  000005 ZTVAL    0010 R 000013 Z1
 0010 R 000014 Z2      0010 R 000015 Z3       0010 R 000044 Z3N      0010 R 000016 Z4       0010 C 000004 Z4TWO
 0010 R 000017 Z5
```

```
00101                 SUBROUTINE MZTRAN(NINC,RTR,RTI,RSR,RSI)                    5985₀  000000
00102          2*     COMMON/KEEP57/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,   5987₀  000000
00103          3*    1             MODIFY,GPRINT                                5988₀  000000
00104          4*     LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                    5990₀  000000
00105          5*     COMMON/KEEP58/ ZPOLE(75),ZMAG(75),NZPOLE                   5991₀  000000
                                                                                       000000
00106          6*     COMPLEX       ZPOLE
00107          7*     COMMON/KEEP18/HALF,ONE,TWO,FOUR,FIFTEN,DD,D1,D2,D3,D4,FIFTY,PI,   5994₀  000000
00108          8*    1             PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3            5995₀  000000
00109          9*     COMPLEX       HALF,ONE,TWO,FOUR,FIFTEN                          000000
00111         10*     COMMON/KEEP20/ESTZ(75),NESTZ                                    000000
00112         11*     COMPLEX       ESTZ                                              000000
00113         12*     COMMON/KEEP21/PNZ(225),PDZ(225),NUMZ,LOCNZ,LOCDZ,NNCZ(75),NDCZ(75)   000000
00114         13*     COMMON/CRUD1/ VAL,TERM,Z4TWO,                             5998₀  000000
00115         14*    1             T1,T2,T3,T4,T5,Z1,Z2,Z3,Z4,Z5,R1,R2,R3,R4,R5,R6,  5999₀  000000
00116         15*    2             ZOH1,ZOH2,ZOH3,ZOH4,ZOH5,ZOH6,ZOH7,          6000₀  000000
00117         16*    3             ROH1,ROH2,ROH3,ROH4,ROH5,ROH6,ROH7,Z3N       6001₀  000000
00118         17*     COMPLEX       VAL,TERM,Z4TWO                                    000000
00119         18*     COMMON/CRUD2/ PNN(200),PD(300),NNCPER(75),NDCPER(75),           000000
00119         19*    1             NUMPOL,LOCN,LOCD                             6007₀  000000
00117         20*     COMMON/CRUD4/ NEQZ,YESZM                                   6009₀  000000
00121         21*     LOGICAL       YESZM                                        6010₀  000000
00121         22*     C                                                          6011₀  000000
00121         23*     C                                                          6012₀  000000
00121         24*     C                                                          6013₀  000000
00121         25*     C  PNN ARRAY CONTAINS ALL NUMERATOR POLYNOMIAL COEFFICIENTS 6014₀  000000
00121         26*     C                                                          6015₀  000000
00121         27*     C  NNCPER ARRAY = NUMBER OF COEFFICIENTS PER NUMERATOR POLYNOMIAL  6016₀  000000
00121         28*     C                                                          6017₀  000000
00121         29*     C  LOCNUM = START LOCATION IN PN ARRAY FOR EACH NEW NUMERATOR POLYNOMIAL  6018₀  000000
00121         30*     C                                                          6019₀  000000
00121         31*     C                                                          6020₀  000000
00121         32*     C  PD ARRAY CONTAINS ALL DENOMINATOR POLYNOMIAL COEFFICIENTS 6021₀  000000
00121         33*     C                                                          6022₀  000000
00121         34*     C  NDCPER ARRAY = NUMBER OF COEFFICIENTS PER DENOMINATOR POLYNOMIAL  6023₀  000000
00121         35*     C                                                          6024₀  000000
00121         36*     C  LOCDEN = PD ARRAY START POSITION FOR EACH NEW DENOMINATOR POLYNOMIAL  6025₀  000000
00121         37*     C                                                          6026₀  000000
00121         38*     C                                                          6027₀  000000
00121         39*           IF (GPRINT) WRITE(6,600) RTR,RTI,RSR,RSI             6028₀  000000
00131         40*       600 FORMAT(///10X,1PE12.5,5X,E12.5,10X,'RESIDUE',2(5X,E12.5))  6029₀  000013
00131         41*           IF (NINC.EQ.2) GO TO 100                             6030₀  000013
00131         42*     C                                                          6031₀  000013
00131         43*     C                                                          6032₀  000013
00131         44*     C  R E A L   N O N - Z E R O   R O O T                     6033₀  000013
00131         45*     C                                                          6034₀  000013
00133         46*           Z1 = SUPERK*RSR*EXP(RTR*ZM*ZT)                               000017
00134         47*           Z2 = D1                                              6036₀  000032
00135         48*           Z3 = -EXP(RTR*ZT)                                            000034
00136         49*           IF (BOTH) GO TO 10                                   6038₀  000043
00140         50*           NZPOLE=NZPOLE+1                                      6039₀  000045
00141         51*           Z3N=-Z3                                              6040₀  000050
00142         52*           ZPOLE(NZPOLE)=CMPLX(Z3N,0.0)                         6041₀  000051
00143         53*           ZMAG(NZPOLE) = ABS(Z3)                                       000056
00144         54*        10 CONTINUE                                             6043₀  000061
00145         55*           IF (YESZOH) GO TO 50                                 6044₀  000061
00146         56*     C                                                          6045₀  000061
```

```
00145     57*      C
00145     58*      C  PRINT Z COEFFICIENTS
00145     59*      C
00147     60*            IF (GPRINT) WRITE(6,601) Z1,Z2,Z3
00155     61*        601 FORMAT(/51X,'Z NUM',5X,1PE12.5/53X,'DEN',2(5X,E12.5))
00156     62*            P1=Z1
00157     63*            R2=-Z1
00160     64*            R3=Z2+Z3
00161     65*            R4=Z2-Z3
00161     66*      C
00161     67*      C
00161     68*      C  PRINT R COEFFICIENTS
00161     69*      C
00162     70*            IF (GPRINT) WRITE(6,602) R1,R2,R3,R4
00171     71*        602 FORMAT(/51X,'R NUM',2(5X,1PE12.5)/53X,'DEN',2(5X,E12.5))
00171     72*      C
00171     73*      C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS
00171     74*      C
00171     75*      C  NUMERATOR COEFFICIENTS
00172     76*            LOCN = LOCN + 1
00173     77*            PNN(LOCN) = R1
00174     78*            LOCN = LOCN + 1
00175     79*            PNN(LOCN) = R2
00175     80*      C  DENOMINATOR COEFFICIENTS
00175     81*            LOCD = LOCD + 1
00177     82*            PD(LOCD) = R3
00201     83*            LOCD = LOCD + 1
00201     84*            PD(LOCD) = R4
00202     85*            NNCPER(NUMPOL) = 2
00203     86*            NDCPER(NUMPOL) = 2
00204     87*            RETURN
00204     88*      C
00204     89*      C
00204     90*      C  ZERO ORDER HOLD SELECTED
00204     91*      C
00205     92*        50  CONTINUE
00205     93*            ZOH1=Z1
00207     94*            ZOH2=-Z1
00211     95*            ZOH3=Z2
00211     96*            ZOH4=Z3
00212     97*            ZOH5 = D0
00212     98*      C
00212     99*      C
00212    100*      C  PRINT ZOH COEFFICIENTS
00212    101*      C
00213    102*            IF (GPRINT) WRITE(6,603) ZOH1,ZOH2,ZOH3,ZOH4,ZOH5
00221    103*        603 FORMAT(/49X,'ZOH NUM',2(5X,1PE12.5)/53X,'DEN',3(5X,E12.5))
00224    104*            ROH1 = -D2*ZOH2
00225    105*            ROH2=ZOH2-ZOH1
00225    106*            ROH3=ZOH3*ZOH4
00227    107*            ROH4 = D2*ZOH3
00233    108*            ROH5=ZOH3-ZOH4
00233    109*      C
00233    110*      C
00233    111*      C  PRINT ROH COEFFICIENTS
00233    112*      C
00231    113*            IF (GPRINT) WRITE(6,604) ROH1,ROH2,ROH3,ROH4,ROH5

60460    000061
60470    000061
60480    000061
60490    000062
60500    000074
60510    000074
60520    000076
60530    000077
60540    000102
60550    000102
60560    000102
60570    000102
60580    000102
60590    000105
60600    000120
60610    000120
60620    000120
60630    000120
60640    000120
60650    000120
60660    000123
60670    000126
60680    000130
60690    000130
60700    000133
60710    000136
60720    000141
60730    000144
60740    000146
60750    000150
60760    000151
60770    000151
60780    000151
60790    000151
60800    000151
60810    000155
60820    000155
60830    000156
60840    000157
60850    000161
60860    000163
60870    000163
60880    000163
60890    000163
60900    000163
60910    000165
60920    000201
60930    000201
60940    000204
60950    000207
60960    000212
60970    000215
60980    000215
60990    000215
61000    000215
61010    000215
61020    000220
```

```
00241    114      604 FORMAT(/49X,'ROH NUM',2(5X,1PE12.5)/53X,'DEN',3(5X,E12.5))   610300   000241
00241    115*     C                                                                 610400   000241
00241    116*     C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS                     610500   000241
00241    117*     C                                                                 610600   000241
00241    118*     C  NUMERATOR COEFFICIENTS                                         610700   000241
00242    119*           LOCN = LOCN + 1                                             610800   000241
00243    120*           PNN (LOCN) = ROH1                                           610900   000244
00244    121*           LOCN = LOCN + 1                                             611000   000247
00245    122*           PNN (LOCN) = ROH2                                           611100   000251
00246    123*     C  DENOMINATOR COEFFICIENTS                                       611200   000251
00246    124*           LOCD = LOCD + 1                                             611300   000254
00247    125*           PD (LOCD) = ROH3                                            611400   000257
00250    126*           LOCD = LOCD + 1                                             611500   000262
00251    127*           PD (LOCD) = ROH4                                            611600   000264
00252    128*           LOCD = LOCD + 1                                             611700   000267
00253    129*           PD (LOCD) = ROH5                                            611800   000271
00254    130*           NNCPER(NUMPOL) = 2                                          611900   000274
00255    131*           NDCPER(NUMPOL) = 3                                          612000   000276
00256    132*           IF (ZMAG(NZPOLE).LT.0.0001 .OR. .NOT.YESZN) RETURN                   000300
00260    133*           NUMZ = NUMZ + 1                                                      000315
00261    134*           NNCZ(NUMZ) = 2                                                       000320
00262    135*           LOCNZ = LOCNZ + 1                                                    000323
00263    136*           PNZ(LOCNZ) = ZOH1                                                    000326
00264    137*           LOCNZ = LOCNZ + 1                                                    000331
00265    138*           PNZ(LOCNZ) = ZOH2                                                    000333
00266    139*           NDCZ(NUMZ) = 2                                                       000336
00267    140*           LOCDZ = LOCDZ + 1                                                    000337
00270    141*           PDZ(LOCDZ) = Z2                                                      000342
00271    142*           LOCDZ = LOCDZ + 1                                                    000345
00272    143*           PDZ(LOCDZ) = Z3                                                      000347
00273    144*           NESTZ = NESTZ + 1                                                    000352
00274    145*           ESTZ(NESTZ) = ZPOLE(NZPOLE)                                          000355
00275    146*           RETURN                                                    612400   000361
00275    147*     C                                                               612500   000361
00275    148*     C                                                               612600   000361
00275    149*     C  COMPLEX NON-ZERO ROOT                                        612700   000361
00275    150*     C                                                               612800   000361
00275    151*      100 CONTINUE                                                   612900   000365
00277    152*           T1 = EXP(RTR*ZT)                                                     000365
00301    153*           T2 = EXP(RTR*ZM*ZT)                                                  000373
00301    154*           T3 = ABS(RTI)*ZT                                                     000403
00302    155*           T4 = D2*RSR*T2                                            613300   000406
00303    156*           T5 = -D2*RSI*T2                                           613400   000412
00304    157*           IF (RTI .LT. 0.0) T5=-T5                                  613500   000416
00306    158*           Z1 = SUPERK*(T4*COS(ZM*T3) + T5*SIN(ZM*T3))                          000424
00307    159*           Z2 = SUPERK*(T5*T1*SIN((D1-ZM)*T3) - T4*T1*COS((D1-ZM)*T3))          000443
00310    160*           Z3 = D1                                                   613800   000467
00311    161*           Z4 = -D2*T1*COS(T3)                                               000471
00312    162*           Z5 = EXP(D2*RTR*ZT)                                               000500
00313    163*           IF (BOTH) GO TO 110                                       614100   000510
00315    164*           VAL = Z4**2/D4 - Z5                                       614200   000512
00315    165*           TERM = CSQRT(VAL)                                                 000521
00317    166*           NZPOLE=NZPOLE+1                                           614400   000525
00321    167*           Z4T=0 = Z4/D2                                             614500   000530
00321    168*           ZPOLE(NZPOLE)=-Z4T=0+TERM                                 614600   000534
00322    169*           ZMAG(NZPOLE) = CABS(ZPOLE(NZPOLE))                                000542
00323    170*           NZPOLE=NZPOLE+1                                           614800   000551
```

```
00324    171*            ZPOLE(NZPOLE)=-Z4TWO-TERM                          6149₀  000556
00325    172*            ZMAG(NZPOLE) = CABS(ZPOLE(NZPOLE))                 6150₀  000565
00326    173*          110 CONTINUE                                         6151₀  000574
00327    174*            IF (YESZOH) GO TO 150                              6152₀  000574
00327    175*      C                                                        6153₀  000574
00327    176*      C                                                        6154₀  000574
00327    177*      C  PRINT Z COEFFICIENTS                                  6155₀  000574
00327    178*      C                                                        6156₀  000574
00331    179*            IF (GPRINT) WRITE(4,605) Z1,Z2,Z3,Z4,Z5            6157₀  000575
00341    180*        605 FORMAT(/51X,'Z NUM',,2(5X,1PE12.5)/53X,'DEN',,3(5X,E12.5)) 6158₀  000611
00342    181*            R1=Z1+Z2                                           6159₀  000611
00343    182*            R2 = -D2*Z2                                        6160₀  000614
00344    183*            R3=Z2-Z1                                           6161₀  000617
00345    184*            R4=Z3+Z4+Z5                                        6162₀  000622
00346    185*            R5 = D2*(Z3-Z5)                                    6163₀  000626
00347    186*            R6=Z3-Z4+Z5                                        6164₀  000632
00347    187*      C                                                        6165₀  000632
00347    188*      C                                                        6166₀  000632
00347    189*      C  PRINT R COEFFICIENTS                                  6167₀  000632
00347    190*      C                                                        6168₀  000632
00351    191*            IF (GPRINT) WRITE(6,606) R1,R2,R3,R4,R5,R6         6169₀  000636
00361    192*        606 FORMAT(/51X,'R NUM',,3(5X,1PE12.5)/53X,'DEN',,3(5X,E12.5)) 6170₀  000653
00361    193*      C                                                        6171₀  000653
00361    194*      C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS            6172₀  000653
00361    195*      C                                                        6173₀  000653
00361    196*      C  NUMERATOR COEFFICIENTS                                6174₀  000653
00362    197*            LOCN = LOCN + 1                                    6175₀  000653
00361    198*            PNN(LOCN) = R1                                     6176₀  000656
00361    199*            LOCN = LOCN + 1                                    6177₀  000661
00365    200*            PNN(LOCN) = R2                                     6178₀  000663
00365    201*            LOCN = LOCN + 1                                    6179₀  000666
00367    202*            PNN(LOCN) = R3                                     6180₀  000670
00367    203*      C  DENOMINATOR COEFFICIENTS                              6181₀  000670
00371    204*            LOCD = LOCD + 1                                    6182₀  000673
00371    205*            PD(LOCD) = R4                                      6183₀  000676
00372    206*            LOCD = LOCD + 1                                    6184₀  000701
00373    207*            PD(LOCD) = R5                                      6185₀  000703
00374    208*            LOCD = LOCD + 1                                    6186₀  000706
00375    209*            PD(LOCD) = R6                                      6187₀  000711
00375    210*            NNCPER(NUMPOL) = 3                                 6188₀  000713
00377    211*            NDCPER(NUMPOL) = 3                                 6189₀  000715
00401    212*            RETURN                                             6190₀  000716
00400    213*      C                                                        6191₀  000716
00401    214*      C                                                        6192₀  000716
00401    215*      C  ZERO ORDER HOLD SELECTED                              6193₀  000716
00401    216*      C                                                        6194₀  000716
00401    217*          150 CONTINUE                                         6195₀  000722
00402    218*            ZOH1=Z1                                            6196₀  000722
00403    219*            ZOH2=Z2-Z1                                         6197₀  000723
00404    220*            ZOH3=-Z2                                           6198₀  000725
00405    221*            ZOH4=Z3                                            6199₀  000727
00405    222*            ZOH5=Z4                                            6200₀  000731
00407    223*            ZOH6=Z5                                            6201₀  000733
00411    224*            ZOH7 = D0                                          6202₀  000735
00411    225*      C                                                        6203₀  000735
00411    226*      C                                                        6204₀  000735
00411    227*      C  PRINT ZOH COEFFICIENTS                                6205₀  000735
```

```
00411    229*      C
00411    229*            IF (GPRINT) WRITE(6,607) ZOH1,ZOH2,ZOH3,ZOH4,ZOH5,ZOH6,ZOH7
00421    230*        607 FORMAT(/49X,'ZOH NUM',3(5X,1PE12.5)/53X,'DEN',4(5X,E12.5))
00421    231*            ROH1 = ZOH1-ZOH2*D3+ZOH3
00426    232*            ROH2 = D3*ZOH3-ZOH1-ZOH2
00425    233*            ROH3=ZOH2-ZOH1-ZOH3
00427    234*            ROH4=ZOH4+ZOH5+ZOH6
00431    235*            ROH5 = D3*ZOH4+ZOH5-ZOH6
00431    236*            ROH6 = D3*ZOH4-ZOH5-ZOH6
00437    237*            ROH7=ZOH4-ZOH5+ZOH6
00437    238*      C
00437    239*      C
00437    240*      C  PRINT ROH COEFFICIENTS
00437    241*      C
00437    242*            IF (GPRINT) WRITE(6,608) ROH1,ROH2,ROH3,ROH4,ROH5,ROH6,ROH7
00446    243*        608 FORMAT(/49X,'ROH NUM',3(5X,1PE12.5)/53X,'DEN',4(5X,E12.5))
00446    244*      C
00446    245*      C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS
00446    246*      C
00446    247*      C  NUMERATOR COEFFICIENTS
00446    248*            LOCN = LOCN + 1
00447    249*            PN(LOCN) = ROH1
00451    250*            LOCN = LOCN + 1
00451    251*            PN(LOCN) = ROH2
00452    252*            LOCN = LOCN + 1
00453    253*            PN(LOCN) = ROH3
00453    254*      C  DENOMINATOR COEFFICIENTS
00455    255*            LOCD = LOCD + 1
00456    256*            PD (LOCD) = ROH4
00456    257*            LOCD = LOCD + 1
00457    258*            PD (LOCD) = ROH5
00461    259*            LOCD = LOCD + 1
00461    260*            PD (LOCD) = ROH6
00462    261*            LOCD = LOCD + 1
00463    262*            PD (LOCD) = ROH7
00464    263*            NNCPER(NUMPOL) = 3
00466    264*            NDCPER(NUMPOL) = 4
00466    265*            IF (ZMAG(NZPOLE).LT.0.0001 .OR. .NOT.YESZM ) RETURN
00471    266*            NUMZ = NUMZ + 1
00471    267*            NNZ(NUMZ) = 3
00473    268*            LOCNZ = LOCNZ + 1
00473    269*            PNZ(LOCNZ) = ZOH1
00474    270*            LOCNZ = LOCNZ + 1
00475    271*            PNZ(LOCNZ) = ZOH2
00476    272*            LOCNZ = LOCNZ + 1
00477    273*            PNZ(LOCNZ) = ZOH3
00501    274*            NDCZ(NUMZ) = 3
00501    275*            LOCDZ = LOCDZ + 1
00502    276*            PDZ(LOCDZ) = Z3
00503    277*            LOCDZ = LOCDZ + 1
00504    278*            PDZ(LOCDZ) = Z4
00505    279*            LOCDZ = LOCDZ + 1
00506    280*            PDZ(LOCDZ) = Z5
00507    281*            DEST(DEST) = DEST + 1
00511    282*            EST(DEST) = ZPOLE(NZPOLE)
00511    283*            RETURN
00512    284*            END
```

62060	000735
62070	000737
62080	000755
62090	000755
62100	000764
62110	000767
62120	000772
62130	000776
62140	001004
62150	001010
62160	001010
62170	001010
62180	001010
62190	001010
62200	001014
62210	001037
62220	001037
62230	001037
62240	001037
62250	001037
62260	001037
62270	001042
62280	001045
62290	001047
62300	001052
62310	001054
62320	001054
62330	001057
62340	001062
62350	001065
62360	001067
62370	001072
62380	001074
62390	001077
62400	001101
62410	001104
62420	001106
	001110
	001125
	001130
	001133
	001136
	001141
	001143
	001146
	001150
	001153
	001154
	001157
	001162
	001164
	001167
	001171
	001174
	001177
62460	001204
62470	001247

```
$FOR,USR F•[redacted]HOL,F•NICHOL
FOR SEIx-05/23/74-12:18:51 (7,8)
-2,2
-6,7
      DATA ICE/0/
-10,10
      IF (ICE .NE. 0) GO TO 31
      ICE = 1
      CALL GRDSET(2,1,2)
   31 CALL GRIDGN (35.995,2.962,90.30,1,2)
      CALL PLOTI(1,1,-720.,0.,-60.,60.,PLTX,PLTY,NP,1,1H )
      CALL PRINT (470,1000,10,0.5,5HPHASE)
      CALL PRINT (10,488,0,15,2,2H08)
      CALL FILMAV(7)
```

SUBROUTINE NICHOL ENTRY POINT 000143

STORAGE USED: CODE(1) 000150; DATA(2) 000066; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003	PLT	000012
0004	PLTARY	004540

EXTERNAL REFERENCES (BLOCK, NAME)

0005	GRDSET
0006	GRIDGN
0007	PLOTI
0010	PRINT
0011	FILMAV
0012	.ABUS
0013	NIO2$
0014	NERR3$

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001	000026	120G	0000	000002	30F	0001	000057	311	0000	000020	32F	0003	000007 DIF1
0003	000010	DIF2	0000 I	000001	I	0000 I	000000	ICE	0003 I	000011	ICK	0003	000004 ICT
0000	000056	INP	0003	000003	LSW	0003 L	000001	NICPLT	0003 I	000002	NP	0004 R	000000 PLTX
0004 R	002260	PLTY	0003	000000	PNI	0003	000006	S360	0003	000005	T360		

```
00101    1•          SUBROUTINE NICHOL                                    000000
00103    2•          LOGICAL NICPLT                                      -01000000
00104    3•          COMMON /PLT/ PNI,NICPLT,NP,ISW,ICT,T360,S360,DIF1,DIF2,ICK   000000
00105    4•          COMMON /PLTARY/ PLTX(1200),PLTY(1200)                000000
00106    5•          DATA ICE/0/                                         NE.000000
00110    6•          NP = NP-2                                           -02000000
00111    7•          IF (ICK .EQ. 1) NP = NP-1                            000002
00113    8•          WRITE(6,30) NP                                       000011
```

```
00115      9*          30 FORMAT (/' PLOT ARRAY FOR NICHOLS PLOT - - NO. OF POINTS = ',I4/                    0D0026
00116     10*         1 5X,'PHASE  VS.  MAG'/)                                                               0D0026
00117     11*           DO 35 I=1,NP,5                                                                       0D0026
00122     12*           WRITE (6,32) PLTX(I),PLTY(I),PLTX(I+1),PLTY(I+1),PLTX(I+2),PLTY(I+                   0D0026
00122     13*         1 2),PLTX(I+3),PLTY(I+3),PLTX(I+4),PLTY(I+4)                                           0D0026
00135     14*          32 FORMAT (2X,10(F9.4,3X))                                                            0D0045
00137     15*          35 CONTINUE                                                                           0D0045
00141     16*           IF (ICE .NE. 0) GO TO 31                                                             NEW0D0045
00143     17*           ICE = 1                                                                              NEW0D0047
00144     18*           CALL GRDSET(2,1,2)                                                                   NEW0D0051
00146     19*          31 CALL GRIDGN (35,995,2,962,80,80,1,2)                                               NEW0D0057
00145     20*           CALL PLOTIT(1,1,-720.,0.,-60.,60.,PLTX,PLTY,NP,1,IH )                                NEW0D0070
00147     21*           CALL PRINT (470,1000,10,0,5,5HPHASE)                                                 NEW0D0105
00150     22*           CALL PRINT (10,480,0,15,2,2HDB)                                                      NEW0D0115
00151     23*           CALL FILMAV(7)                                                                       NEW0D0125
00152     24*           RETURN                                                                               NEW0D0139
00153     25*           END                                                                                   0D0147

     ... END OF COMPILATION:  NO  DIAGNOSTICS.

$PSE> F.
-- FUNPJR 0026-05/23-12:19
```

```
@FOR,USW F,NOMMTX,F,NOMMTX
FOR,SEIX-05/23/74-08:36:48 (2.3)

      SUBROUTINE NOMMTX     ENTRY POINT 000032

      STORAGE USED: CODE(1) 000037; DATA(0) 000012; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP1   000026
      0004   KEEP9   000705
      0005   KEEP10  021620
      0006   KEEP13  000313
      0007   KEEP16  000031
      0010   CRUD4   000002

      EXTERNAL REFERENCES (BLOCK, NAME)

      0014   WERR35

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000011 117G      0007 L 000022 DEBUG    0004 C 000226 EA       0005 R 016664 EV       0000 I 000000 I
      0000   000003 INJRS     0007   000010 IOPEN    0005   000004 IR       0005   001754 JC       0007   000011 JOPEN
      0004   000572 KD        0007   000001 KODE    0007 L 000023 LFLT     0005   005674 LL       0005   007644 LOCPOL
      0006   000144 LOCV      0003   000000 MAXIT   0003   000001 MXEIG    0003   000024 MXEIGT   0003   000023 MXEST
      0003   000002 MXFRM     0003   000003 MXNBM   0003   000004 MXNCOF   0003   000005 MXNCT    0003   000025 MXNCV
      0003   000006 MXNE      0003   000007 MXNEQ   0003   000010 MXNFI    0003   000011 MXNG     0003   000012 MXNPH
      0003   000013 MXNPP     0003   000014 MXNQPT  0003   000015 MXNSM    0003   000016 MXNSP    0003   000017 MXNTM
      0003   000020 MXNV      0003   000021 MXNZT   0003   000022 MXPOLY   0000 I 000001 N        0004   000455 NA
      0005   000003 NCOF      0006   000312 NCOFV   0005   003724 ND       0005   000000 NDEG     0005   000002 NE
      0004   000454 NEIG      0005   000001 NEQ     0006   000000 NEQZ     0006   000311 NEV      0004   000457 NI
      0007   000027 NOMINAL   0007 L 000030 NOTYET  0004   000456 NR       0007   000002 NRCLPL   0007   000003 NRPOLE
      0007   000004 NRZERO    0006 I 000310 NV      0007   000005 NXB      0007   000006 NXN      0007   000007 NXR
      0007   000014 PCPL      0007   000015 PFAC    0007   000013 PNOM     0007   000016 PSLOSH   0007   000012 PVAR
      0004 C 000000 ROOT      0007   000000 STAGE   0006 R 000000 SV       0007 L 000017 YESMTX   0007 L 000024 YESPCH
      0007 L 000020 YESRAW    0007 L 000025 YESRLP   0007 L 000026 YESSRL   0007 L 000021 YESSRP   0010 L 000001 YESZM

      1*          SUBROUTINE NOMMTX                                          62480   000000
      2*         1COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,  62490   000000
      3*         1              MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,  62500   000000
      4*         2              MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV              62510   000000
      5*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)        62520   000000
                                                                                        000000
      6*          COMPLEX       ROOT,EA                                         62550   000000
      7*          COMMON/KEEP10/NDEG,NEQ,NE,NCOF,IR(1000),JC(1000),ND(1000),
      8*         1              LL(1000),LOCPOL(60,60),EV(1500)                 62560   000000
      9*          COMMON/KEEP13/SV(100),LOCV(100),NV,NEV,NCOFV                  62570   000000
     10*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,    62580   000000
     11*         1              IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,         62590   000000
     12*         2              YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,  62600   000000
```

```
00110    13*          3                YESSRL,NOMNAL,NOTYET                    626100  000000
00111    14*          LOGICAL          YESMTX,YESRAW,YESRP,DEBUG,LFLT,YESPCH,YESRLP,  626200  000000
00111    15*          1                YESSRL,NOMNAL,NOTYET                    626300  000000
00112    16*          COMMON/CRUDM/ NEQ7,YESZM                                 626400  000000
00113    17*          LOGICAL          YESZM                                   626500  000000
00113    18*     C                                                             627800  000000
00114    19*     C                                                             627900  000000
00115    20*     C  RESTORE THE MATRIX COEFFICIENTS SAVED IN GENERAL FORM       628000  000000
00116    21*     C                                                             628100  000000
00117    22*          IF (NV.EQ.0) RETURN                                      628300  000000
00119    23*          DO 10 I=1,NV                                             628400  000004
00121    24*          N = LOCV(I)                                              629500  000011
00123    25*          EV(N) = SV(I)                                            629600  000013
00123    26*     10   CONTINUE                                                 628700  000016
00125    27*          NV = 0                                                   628800  000016
00126    28*          RETURN                                                   628900  000017
00127    29*          END                                                      630800  000036

     END OF COMPILATION:  NO  DIAGNOSTICS.
```

```
@FOR,US F.NYQIST,F.NYQIST
-- FOR 9E1X-05/23/74-08:37:00--(2.3)

     SUBROUTINE NYQIST    ENTRY POINT 000006

     STORAGE USED: CODE(1) 000010; DATA(0) 000004; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

         0003    NERR3$

     STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

         0000    000000 .INJP$

 00101     1*          SUBROUTINE NYQIST                                  000000
 00101     2*      C                                                      000000
 00101     3*      C                                                      000000
 00101     4*      C  DUMMY SUBROUTINE FOR NYQUIST PLOT                   000000
 00101     5*      C                                                      000000
 00103     6*          RETURN                                             000007
 00104     7*          END

     END OF COMPILATION:      NO DIAGNOSTICS.
```

```
RFOR,USR F,OUTPUT,F,OUTPUT
FOR SEIX-05/23/74-0°;37:19 (4,5)

    SUBROUTINE OUTPUT     ENTRY POINT 000502

    STORAGE USED: CODE(1) 000507; DATA(0) 000102; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   KEEP3   000102
    0004   KEEP6   000134
    0005   KEEP10  000031
    0006   CRUD3   011650
    0007   PLT     000012
    0010   PLTARY  004540

    EXTERNAL REFERENCES (BLOCK, NAME)

    0011   SYSERR
    0012   NXDUS
    0013   NIO3S
    0014   NIO2S
    0015   SQRT
    0016   ATAN2
    0017   NERR2S
    0020   ALOG10
    0021   NIO1S
    0022   NERR3S

    STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001  000116 10L     0001  000127 11L     0001  000134 12L     0001  000043 2L      0001  000235 20L
    0001  000201 2048    0001  000245 21L     0001  000252 22L     0001  000274 2^2G    0001  000065 3L
    0001  000153 30L     0001  000163 31L     0001  000102 4L      0000  000014 600F    0000  000021 601F
    0000  000037 602F    0000  000047 603F    0000  000053 604F    0001  000344 700L    0001  000406 705L
    0001  000416 710L    0001  000465 712L    0006  000070 ADIR    0000 R 000011 AD1     0000 R 000012 AD2
    000A  000006 AFRD    0006  011616 AML     0006 R 011617 AMP     0000 R 000003 AMPN    0006  000152 APHA
    0006  011620 BIG     0004  000064 CARD    0000 R 000007 DB      0006 L 011642 DECR    0005 R 000025 DEG
    0006 R 011621 DF     0007 R 000007 DIF1    0007 R 000010 DIF2    0003  000063 DP      0006 L 011643 DPI
    0005  000012 D0      0005  000013 D1      0005  000014 D2      0005  000015 D3      0005  000016 D4
    0006 L 011645 ERAM    0006 L 011647 ERGP    0006 L 011644 ERP     0006 L 011646 ERPH    0005 C 000010 FIFTEN
    0005  000017 FIFTY    0006 R 011622 FL      0005 C 000006 FOUR    0006 R 011623 FR      0000 R 000004 FRR1
    0000 R 000010 FRR2    0006 C 000000 GAINS   0005 C 000000 HALF    0000 I 000013 ICH     0007 I 000011 ICK
    0007 I 000004 ICT     0006 R 011624 IM      0000  000071 INJPS   0006  011634 INT     0007 I 000003 ISW
    0000 I 000005 J       0004  000110 LABEL   0006  011641 LMX     0006 I 011635 LRPR    0003 R 000051 MAX
    0003 R 000037 MIN     0006 L 011636 MPPP    0004 R 000062 NAME    0006  011637 NEXT    0003  000000 NFI
    0006  000003 NGNPK    0007 L 000001 NICPLT  0007 I 000002 NP      0006 I 011640 NPPP    0006  000004 NP180
    0006  000005 NYOPTS   0005  000026 N1      0006  000002 N1ANP    0005  000027 N2      0005  00003a N3
    0000 R 000000 OLD     0005 C 000002 ONE     0006  000316 PANP    0003  000076 PB      0003  000025 PCY
    0006  000400 PDIP    0006  011625 PER     0006  000234 PFRQ    0006 R 011626 PHA     0006  000626 PHAMP
    0006  000716 PHOIR   0006  000544 PHFRQ   0006  011627 PHL     0005  000020 PI      0005  000021 PI2
    0010 R 000000 PLTX    0010 R 002940 PLTY    0003  000075 PI      0007  000000 PNI     0006  000462 PPHA
    0003  000377 PI180   0005  000024 RADDEG  0006 R 011630 RE      0005  000023 RPI     0000 R 000006 SAV
```

```
0006   003726 SAVAMP      0006   000772 SAVFRQ      0006   006662 SAVPHA      0006   011631 SMA         0005   000022 SMALL
0006   011632 STA         0003 L 000101 STNDRD      0006 R 011633 STO         0003   000013 STP         0003   000001 STR
0007 R 000006 S360        0004 R 000000 TITLE       0004 R 000024 TITLE1      0004 R 000036 TITLE2      0004   000050 TITLE3
0005 C 000004 T40         0007 R 000005 T360        0003 L 000100 YESNYQ
```

```
00101     1*          SUBROUTINE OUTPUT                                                    661600  000000
00103     2*          COMMON/KEEP3/ NFI,STR(10),STP(10),PCT(10),MIN(10),MAX(10),DP(10),    661700  000000
00103     3*         1            PN,PB,P180,YESNYQ,STNDRD                                661800  000000
00104     4*          REAL         MIN,MAX                                                 661900  000000
00105     5*          LOGICAL      YESNYQ,STNDRD                                           662000  000000
00106     6*          COMMON/KEEP4/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),    662100  000000
00107     7*         1            CARD(20),LABEL(20)                                      662200  000000
00107     8*          REAL         NAME                                                    662300  000000
00110     9*          COMMON/KEEP1N/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,     662400  000000
00110    10*         1            PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                       662500  000000
00111    11*          COMPLEX      HALF,ONE,TWO,FOUR,FIFTEN                                000000
00112    12*          COMMON/CPHD3/ GAINS,NIAMP,NGNPK,NP180,NYNPTS,AFRQ(50),ADIR(50),      662800  000000
00113    13*         1            APHA(50),PFRQ(50),PAMP(50),PDIR(50),PPHA(50),           662900  000000
00113    14*         2            PHFRQ(50),PHAMP(50),PHDIR(50),SAVFRQ(1500),             663000  000000
00113    15*         3            SAVAMP(1500),SAVPHA(1500),AML,AMP,BIG,DF,FL,FR,IM,      663100  000000
00113    16*         4            PER,PHA,PHL,RF,SMA,STA,STO,                             663200  000000
00113    17*         5            INT,LRPR,MPPP,NEXT,NPPP,LMX,                            663300  000000
00113    18*         6            DECR,DPI,ERP,FRAM,ERPH,ERGP                             663400  000000
00113    19*          COMPLEX      GAINS                                                   663500  000000
00114    20*          REAL         IM                                                      663600  000000
00115    21*          LOGICAL      DECR,DPI,ERP,FRAM,ERPH,ERGP                             663700  000000
00116    22*          DIMENSION    OLD(3)                                                  663800  000000
00117    23*          COMMON /PLT/ PNI,NICPLT,NP,ISW,ICT,T360,S360,DIF1,DIF2,ICK           000000
00120    24*          COMMON /PLTARY/ PLTX(1200),PLTY(1200)                                000000
00121    25*          LOGICAL NICPLT                                                       000000
00121    26*     C                                                                         663900  000000
00121    27*     C                                                                         664000  000000
00121    28*     C    WRITE TITLE IF NECESSARY                                             664100  000000
00121    29*     C                                                                         664200  000000
00122    30*          IF (.NOT.DPI) GO TO 2                                                664300  000000
00124    31*          IF (LRPR.NE.1 .OR. MPPP.LT.NPPP) GO TO 2                             664400  000001
00126    32*          MPPP = 0                                                             664500  000016
00127    33*          WRITE(6,600) TITLE,TITLE1,TITLE2                                     664600  000017
00131    34*      600 FORMAT('1',18X,20A4//19X,20A4//)                                     664700  000035
00135    35*          WRITE(6,601)                                                         664800  000035
00137    36*      601 FORMAT('0',21X,'CPS',7X,'GAIN',8X,'AMPLITUDE',4X,'DECIBELS',5X,      664900  000043
00137    37*     *    'PHASE',2X))                                                         665000  000043
00137    38*     C                                                                         665100  000043
00137    39*     C    COMPUTE AMPLITUDES AND PHASES                                        665200  000043
00137    40*     C                                                                         665300  000043
00140    41*          DO 2 RE = REAL(GAINS)                                                665400  000043
00141    42*          IM = AIMAG(GAINS)                                                    665500  000044
00142    43*          AMP = SQRT(RE**2 + IM**2)                                            665600  000046
00143    **0,AGNOSTIC*.  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00143    44*          IF(AMP.NE.0.) GO TO 3                                                665700  000060
00145    45*          PHA = 0.                                                             665800  000062
00146    46*          GO TO 4                                                              665900  000063
00147    47*      3   PHA = DFG * ATAN2(IM,RE)                                           666000  000065
00150    48*          IF(PHA .LT. 0.) PHA = PHA + 360.                                     666100  000072
```

```
00157     49*          4 CONTINUE                                                  66620  000102
00158     50*            IF (.NOT.OPT) RETURN                                      66630  000102
00159     51*            GO TO (10,20), LRPR                                       66640  000106
00159     52*      C                                                               66650  000106
00159     53*      C     STORE OUTPUT FOR LEFT SIDE OF PAGE                        66660  000106
00159     54*      C                                                               66670  000106
00160     55*           10 LRPR = 2                                                66680  000116
00161     56*              AMPN = AMP                                              66690  000117
00162     57*              OLD(1) = AMPN                                           66700  000121
00163  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00163     58*              IF (AMPN.NE.0.)  GO TO 11                               66710  000122
00164     59*              OLD(2) = 0.                                             66720  000124
00165     60*              GO TO 12                                                66730  000125
00166     61*           11 OLD(2) = 20.* ALOG10(AMPN)                              66740  000127
00167     62*           12 OLD(3) = PHA                                            66750  000134
00167     63*              IF (DF.LT.0.) GO TO 30                                  66760  000135
00171     64*              IF (FR.LT.STO)  RETURN                                  66770  000141
00172     65*              GO TO 31                                                66780  000151
00174     66*           30 IF (FR.GT.STO) RETURN                                   66790  000153
00176     67*           31 CONTINUE                                                66800  000163
00176     68*      C                                                               66810  000163
00176     69*      C                                                               66820  000163
00176     70*      C                                                               66830  000163
00176     71*      C  ON COMPUTER PRINTOUT, THE FIRST FREQUENCY TO APPEAR IS THE REAL  66840  000163
00176     72*      C (OR SYSTEM) FREQUENCY.  UNDERNEATH IT, ENCLOSED IN PARENTHESES, IS  66850  000163
00176     73*      C  THE OMEGA SUB-R DOMAIN FREQUENCY.                            66860  000163
00176     74*      C                                                               66870  000163
00176     75*      C                                                               66880  000163
00176     76*      C                                                               66890  000163
00177     77*              CALL SYSFRQ (FR,FRR1)                                   66900  000163
00200     78*              WRITE(6,602) FR,NAME,(OLD(J),J=1,3)                     66910  000166
00210     79*          602 FORMAT(2(/),2(1X,F12.5,5X,2A4,2(1PE13.4,0PF10.3)))      66920  000204
00211     80*              WRITE(6,603) FRR1                                       66930  000204
00214     81*          603 FORMAT(2X,'(',F10.5,')')                                66940  000212
00215     82*              IF (.NOT. NICPLT) RETURN                                       000212
00217     83*              IF (ICT .GT. 0) DIF1 = SAV-OLD(3)                              000217
00221     84*              SAV = OLD(3)                                                   000226
00222     85*              ICT = 1                                                        000230
00223     86*              ICK = 1                                                        000232
00225     87*              GO TO 700                                                      000233
00224     88*      C                                                               66960  000233
00224     89*      C     PRINT OUT PAIR OF OUTPUT POINTS                           66970  000233
00224     90*      C                                                               66980  000233
00225     91*           20 LRPR = 1                                                66990  000235
00226     92*              NPPP = NPPP + 1                                         67000  000236
00227  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00227     93*              IF (AMP.NE. 0.0) GO TO 21                               67010  000240
00231     94*              DB = 0.                                                 67020  000242
00232     95*              GO TO 22                                                67030  000243
00233     96*           21 DB = 20. *ALOG10(AMP)                                   67040  000245
00238     97*           22 CALL SYSFRQ (FL,FRR1)                                   67050  000252
00235     98*              CALL SYSFRQ (FR,FRR2)                                   67060  000255
00236     99*              WRITE (6,602) FL,NAME,(OLD(J),J=1,3),FR,NAME,AMP,DB,PHA 67070  000261
00252    100*              WRITE(6,604) FRR1,FRR2                                  67080  000310
00257    101*          604 FORMAT(2X,'(',F10.5,')',50X,'(',F10.5,')')              67090  000317
00241    102*              IF (.NOT. NICPLT) RETURN                                       000317
00242    103*              IF (ICT .GT. 0) DIF1 = SAV-OLD(3)                              000324
```

```
00264   104*            DIF2 = OLD(3)-PHA                                   000333
00265   105*            ICK =0                                              000336
00266   106*            ICT = 1                                             000337
00267   107*            SAV = PHA                                           000341
00270   108*        700 AD1 = ABS(DIF1)                                     000344
00271   109*            AD2 = ABS(DIF2)                                     000345
00272   110*            IF (AD1 .LT. 300. .AND. AD2 .LT. 300.) GO TO 710    000347
00274   111*            IS = IS+1                                           000363
00275   112*            ICH = MOD(IS,2)                                     000366
00276   113*            IF (ICH .EQ. 1) GO TO 705                           000371
00300   114*            S360 =-360.                                         000374
00301   115*            IF (AD1 .LT. 300.) GO TO 710                        000376
00303   116*            T360 =-360.                                         000403
00304   117*            GO TO 710                                           000404
00305   118*        705 S360 = -720.                                        000406
00306   119*            IF (AD1 .LT. 300.) GO TO 710                        000407
00310   120*            T360 =-720.                                         000414
00311   121*        710 PLTX(NP) = OLD(3)+T360                              000416
00312   122*            PLTY(NP) = OLD(2)                                   000421
00313   123*            IF (ICK .EQ. 1) GO TO 712                           000423
00315   124*            PLTX(NP+1) = PHA+S360                               000427
00316   125*            PLTY(NP+1) = OB                                     000432
00317   126*            NP = NP+2                                           000434
00320   127*            IF (AD2 .LT. 300.) RETURN                           000437
00322   128*            IF (ICH .EQ. 0) T360 = -360.                        000447
00324   129*            IF (ICH .EQ. 1) T360 = -720.                        000453
00326   130*            RETURN                                              000461
00327   131*        712 NP = NP+1                                           000465
00330   132*            RETURN                                  67100       000467
00331   133*            END                                     67110       000506

      END OF COMPILATION:       3 DIAGNOSTICS.
```

```
$FOR,USW F,PEVAL,F,PEVAL
FOR (EIX=05/23/74=08:38:34 (3,4)

      SUBROUTINE PEVAL      ENTRY POINT 000312

      STORAGE USED: CODE(1) 000357; DATA(0) 000032; BLANK COMMON(2) 000000

      COMMON BLOCKS:

 0003   KEEP9   000705
 0004   KEEP14  000031
 0005   CRUD3   016115

      EXTERNAL REFERENCES (BLOCK, NAME)

 0006   CDABV
 0007   CDV$
 0010   NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

 0001   000014 10L      0001   000021 12$G     0001   000146 15$G     0001   000050 20L      0001   000066 30L
 0001   000137 40L      0001   000172 45L      0001   000223 46L      0001   000273 50L      0000 C 000005 ARG
 0005 L 016113 AUTO     0005 C 016072 B        0000 R 000005 C        0006 R 000000 CDABV    0005 L 016112 CONJ
 0005 C 000000 CU       0004   000025 DEG      0005 L 016110 DONE     0004   000012 DO       0004   000013 D1
 0004   000014 D2       0004   000015 D3       0004   000016 D4       0003 C 000226 EA       0004 C 000010 FIFTEN
 0004   000017 FIFTY    0004 C 000006 FOUR     0005 C 016040 FPR0     0005 C 016042 FPR1     0005 C 016044 FPR2
 0005 C 016046 FR0      0005 C 016050 FR1      0005 C 016052 FR2      0004 C 000000 HALF     0000 I 000004 I
 0003   000716 IWJP$    0003   000572 K0       0003   000455 NA       0005   016102 NCT      0003 I 000454 NEIG
 0005   016107 NEST     0005   016076 NFP0     0005   016077 NFP1     0005   016100 NFP2     0003   000457 NI
 0005   016101 NITER    0005   016104 NKODE    0003   000456 NR       0005   016103 NREG     0005   016105 NSTART
 0005   016106 NTIME    0004   000026 N1       0004   000027 N2       0004   000030 N3       0004 C 000002 ONE
 0004   000020 PI       0004   000021 PI2      0005 C 016054 PR0      0005 C 016056 PR1      0005 C 016060 PR2
 0004   000024 RADDEG   0005 L 016114 REGSEL   0005 L 016111 RESTRT   0003 C 000000 ROOT     0004   000023 RPI
 0005 C 016062 R0       0005 C 016064 R1       0005 C 016066 R2       0005 C 016070 R3       0004   000022 SMALL
 0004 C 000004 TWO      0005 C 016074 U        0000 C 000000 X        0000 C 000002 Y

 00101    1*          SUBROUTINE PEVAL(Z,EVAL,NSCALE)                                          671200  000000
 00101    2*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)                   671300  000000
                                                                                                       000000
 00101    3*          COMPLEX       ROOT,EA                                                    671600  000000
 00103    4*          COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,          671700  000000
 00103    5*         1              PI2,SMALL,RPI,RADDEG,DEG,NI,N2,N3                                000000
 00104    6*          COMPLEX       HALF,ONE,TWO,FOUR,FIFTEN                                   672000  000000
 00107    7*          COMMON/CRUD3/ CU(60,60),FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,          672100  000000
 00107    8*         1              R0,R1,R2,R3,B,U,NFP0,NFP1,NFP2,NITER,NCT,NREG,NKODE,       672200  000000
 00107    9*         2              NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL                    000000
 00111   10*          COMPLEX       CU,FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,                 672400  000000
 00111   11*         1              R0,R1,R2,R3,B,U                                            672500  000000
 00111   12*          LOGICAL       DONE,RESTRT,CONJ,AUTO,REGSEL                                      000000
 00112   13*          COMPLEX       Z,EVAL
```

```
00113     14*           COMPLEX       X,Y,ARG                                    000000
00114     15*           DIMENSION     C(2)                                       000000
00115     16*           EQUIVALENCE   (ARG,C(1))                                 000000
00116     17*     C                                                    6730₀     000000
00117     18*     C                                                    6731₀     000000
00118     19*     C****    EVALUATE POLYNOMIAL AT X                    6732₀     000000
00119     20*     C****    CHANGE X IF X IS CLOSE TO PREVIOUSLY FOUND ROOT 6733₀ 000000
00120     21*     C                                                    6734₀     000000
00121     22*           X=Z                                            6735₀     000000
00122     23*           EVAL = (1.,0.)                                           000001
00123     24*           NSCALE=0                                       6737₀     000003
00124     25*           IF (NEIG.LE.0) RETURN                          6738₀     000004
00125     26*     C                                                    6739₀     000004
00126     27*     C                                                    6740₀     000004
00127     28*     C  COMPARE ITERANT WITH PREVIOUSLY COMPUTED ROOTS    6741₀     000004
00128     29*     C  MODIFY THE ITERANT IF NECESSARY                   6742₀     000004
00129     30*     C                                                    6743₀     000004
00130     31*      10 CONTINUE                                         6744₀     000014
00131     32*           DO 40 I=1,NEIG                                 6745₀     000014
00132     33*           Y=X-ROOT(I)                                    6746₀     000021
00133     34*           IF (CDABV(X) .GE. 1.E-6) GO TO 20                        000030
00134     35*           IF (CDABV(Y) .GE. 1.E-6) GO TO 40                        000037
00135     36*           GO TO 30                                       6749₀     000046
00136     37*      20 CONTINUE                                         6750₀     000050
00137     38*           IF (CDABV(Y/X) .GE. 1.E-6) GO TO 40                      000050
00140     39*      30 CONTINUE                                         6752₀     000066
00141     40*           X = X - (.001,0.)                                        000066
00142     *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00142     41*           IF (CDABV(X-R0) .EQ. 0.) GO TO 30                        000074
00143     *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00144     42*           IF (CDABV(X-R1) .EQ. 0.) GO TO 30                        000107
00145     *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00145     43*           IF (CDABV(X-R2) .EQ. 0.) GO TO 30                        000122
00150     44*           GO TO 10                                       6757₀     000135
00151     45*      40 CONTINUE                                         6758₀     000141
00151     46*     C                                                    6759₀     000141
00151     47*     C                                                    6760₀     000141
00151     48*     C  EVALUATE THE POLYNOMIAL AT THE ITERANT            6761₀     000141
00151     49*     C                                                    6762₀     000141
00153     50*           DO 50 I=1,NEIG                                 6763₀     000141
00155     51*           EVAL=EVAL*(X-ROOT(I))                          6764₀     000146
00157     52*      45 CONTINUE                                         6765₀     000172
00160     53*           ARG = EVAL                                     6766₀     000172
00161     54*           IF (ABS(C(1)).LT.1.E20 .AND. ABS(C(2)).LT.1.E20) GO TO 46          000173
00163     55*           EVAL=EVAL/FIFTEN                               6768₀     000211
00164     56*           NSCALE=NSCALE+15                               6769₀     000216
00165     57*           GO TO 45                                       6770₀     000221
00166     58*      46 CONTINUE                                                    000223
00167     59*           ARG = EVAL                                               000223
00171     *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00171     60*           IF (CDABV(ARG) .EQ. 0.0) GO TO 50                        000224
00172     61*           IF (ABS(C(1)).GE.1.E-20 .OR. ABS(C(2)).GE.1.E-20) GO TO 50         000231
00174     62*           EVAL = EVAL*FIFTEN                                       000247
00175     63*           NSCALE = NSCALE - 15                                     000266
00176     64*           GO TO 46                                                 000271
00177     65*      50 CONTINUE                                         6771₀     000274
00201     66*           Z=X                                            6772₀     000274
```

```
00202:    67*        RETURN                                  67730    000276
00203:    68*        END                                     67740    000356

END OF COMPILATION:     4 DIAGNOSTICS.
```

```
FOR,USN F,PFE,F,PFE
FOR SEIX-05/23/74-08:38:47 (3.4)

      SUBROUTINE PFE        ENTRY POINT 000334

      STORAGE USED: CODE(IL) 000354; DATA(D) 000044; BLANK COMMON(Z) 000000

      COMMON BLOCKS:

      0003   KEEP5  000074
      0004   KEEP7  000712
      0005   CRUD2  001215

      EXTERNAL REFERENCES (BLOCK, NAME)

      0006   SEPDPC
      0007   MZTRAN
      0010   ZTRAN
      0011   CDV$
      0012   NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001  000102 10L      0001  000115 140G     0001  000204 155G     0001  000104 20L      0001  000171 40L
      0001  000226 50L      0001  000230 60L      0003 L 000071 BOTH     0004  000570 FDPOL    0004  000454 FNPOL
      0003 L 000073 GPRINT   0000 I 000011 I       0000  000021 IND$     0003  000067 ITHZT    0000 I 000010 K
      0005  001214 LOCD      0005  001213 LOCN     0003 L 000072 MODIFY   0004  000707 NDCOEF   0005  001077 NDCPER
      0004 I 000705 NDGD     0004 I 000704 NDGN     0004  000706 NNCOEF   0005  000764 NNCPER   0005 I 001212 NUMPOL
      0004  000711 NZD       0004  000710 NZN      0003  000004 NZT      0005  000310 PD       0005  000000 PNN
      0000 C 000002 RESDUE   0003 R 000013 RESING   0000 R 000012 RESREL   0004 R 000341 RID      0004 R 000113 RIN
      0000 R 000007 RNI      0000 R 000006 RNR      0000 R 000005 RPI      0000 R 000004 RPR      0000 R 000226 RRD
      0004 R 000370 RRN      0003  000000 SUPERK    0003  000003 TD       0000 C 000000 VALUE    0003 L 000070 YESZOH
      0003  000002 ZN        0003  000001 ZT       0003  000005 ZTVAL

00101    1*          SUBROUTINE PFE(N,NINC,NFINAL)                                    6775  000000
00103    2*          COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,   6776  000000
00104    3*         1              MODIFY,GPRINT                                      6777  000000
00104    4*          LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                          6779  000000
00105    5*          COMMON/KEEP7/ RRN(75),RIN(75),RRD(75),RID(75),FNPOL(76),         6780  000000
00106    6*         1              FDPOL(76),NDG,NDGD,NNCOEF,NDCOEF,NZN,NZD           6781  000000
00107    7*          COMMON/CRUD2/ PNN(200),PD(300),NNCPER(75),NDCPER(75),                  000000
00108    8*         1              NUMPOL,LOCN,LOCD                                   620  000000
00109    9*          COMPLEX       VALUE,RESDUE                                           000000
00107   10*      C                                                                    6785  000000
00107   11*      C                                                                    6786  000000
00107   12*      C  DETERMINE WHETHER A REAL OR COMPLEX POLE                          6787  000000
00107   13*      C  NINC = 1    IMPLIES REAL POLE                                     6788  000000
00107   14*      C  NINC = 2    IMPLIES COMPLEX POLE WITH THE CONJUGATE AS THE NEXT POLE 6789  000000
00107   15*      C                                                                    6790  000000
00111   16*            VALUE = CMPLX(RRD(N),RID(N))                                       000000

6-133
```

```
00111    17          IF (N.EQ.NFINAL) GO TO 20                                  6792₀  000003
00112    18*         RPR = RRD(N)                                               6793₀  000007
00113    19*         RPI = RID(N)                                               6794₀  000012
00114    20*         RNR = RRD(N+1)                                             6795₀  000014
00115    21*         RNI = RID(N+1)                                             6796₀  000016
00116    22*         IF (ABS(RPI).LE.1.E-4 .OR. ABS(RNI).LE.1.E-4) GO TO 20             000020
00120    23*         IF (ABS(RPI/RNI + 1.) .GT. 1.E-4) GO TO 20                         000036
00121    24*         IF (ABS(RPR).LE.1.E-4 .AND. ABS(RNR).LE.1.E-4) GO TO 10            000046
00125    25*         IF (ABS(RNR).LE.1.E-4) GO TO 20                                    000064
00127    26*         IF (ABS(RPR/RNR - 1.) .GT. 1.E-4) GO TO 20                         000071
00131    27*     10  NINC = 2                                                   6802₀  000102
00132    28*     20  CONTINUE                                                  6803₀  000104
00132    29*     C                                                           6804₀  000104
00132    30*     C                                                           6805₀  000104
00132    31*     C  EVALUATE THE RESIDUE                                     6806₀  000104
00132    32*     C                                                           6807₀  000104
00133    33*         RESDUE = (1.,0.)                                                000104
00134    34*         K = 0                                                       6809₀  000105
00135    35*         IF (NDGN.EQ.0) GO TO 40                                 6810₀  000106
00137    36*         DO 30 I=1,NDGN                                          6811₀  000110
00140    37*         K = K + 1                                               6812₀  000115
00141    38*         IF (K.EQ.11) K=K+1                                      6813₀  000117
00145    39*         RESDUE = RESDUE*(VALUE=CMPLX(RRN(I),RIN(I)))/                   000126
00146    40*        *          (VALUE=CMPLX(RRD(K),RID(K)))                         000126
00149    41*     30  CONTINUE                                                  6816₀  000171
00150    42*     40  CONTINUE                                                  6817₀  000171
00151    43*         K = K + 1                                               6818₀  000171
00152    44*         IF (K.GT.NDGD) GO TO 60                                 6819₀  000173
00154    45*         DO 50 I=K,NDGD                                          6820₀  000177
00157    46*         IF (I.EQ.N) GO TO 50                                    6821₀  000204
00161    47*         RESDUE = RESDUE/(VALUE=CMPLX(RRD(I),RID(I)))                    000210
00163    48*     50  CONTINUE                                                  6823₀  000230
00164    49*     60  CONTINUE                                                  6824₀  000230
00164    50*     C                                                           6825₀  000230
00164    51*     C                                                           6826₀  000230
00164    52*     C  SEPARATE RESIDUE INTO REAL AND IMAGINARY PARTS           6827₀  000230
00164    53*     C                                                           6828₀  000230
00165    54*         CALL SEPDPC(RESDUE,RESREL,RESIMG)                       6829₀  000230
00166    55*         IF (NINC.EQ.1) RESIMG=0.0                                       000234
00170  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00170    56*         IF (RESREL.EQ.0.0 .AND. RESIMG.EQ.0.0) RETURN                   000243
00170    57*     C                                                           6831₀  000243
00170    58*     C                                                           6832₀  000243
00170    59*     C  PERFORM Z-R TRANSFORMATION                               6833₀  000243
00170    60*     C                                                           6834₀  000243
00171    61*         NUMPOL = NUMPOL + 1                                             000256
00173    62*         IF (MODIFY) CALL MZTRAN(NINC,RRD(N),RID(N),RESREL,RESIMG)  6835₀  000261
00175    63*         IF (.NOT.MODIFY) CALL ZTRAN(NINC,RRD(N),RID(N),RESREL,RESIMG)  6836₀  000277
00177    64*         RETURN                                                  6837₀  000315
00200    65*         END                                                     6838₀  000353

      END OF COMPILATION:        1 DIAGNOSTICS.
```

```
GFOR USW F=PFEZRO,F=PFEZRO
FOR REIX-05/23/74-08:39:58 (3.4)

      SUBROUTINE PFEZRO    ENTRY POINT 000262

      STORAGE USED: CODE(1) 000314  DATA(0) 000251  BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003  KEEP5   000074
      0004  KEEP7   000712
      0005  KEEP1A  000031
      0006  KEEP1B  000002
      0007  CRUD2   004245

      EXTERNAL REFERENCES (BLOCK, NAME)

      0010  MODZRO
      0011  MLTZRO
      0012  XPRI
      0013  NERR45
      0014  NERR35

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001  000052 10L    0001  000234 1000L  0001  000117 200L   0001  000242 2000L  0001  000130 30L
      0001  000142 40L    0001  000155 50L    0003 L 000071 BOTH   0005 L 000022 DEBUG  0004 R 000570 FDPOL
      0004 R 000454 FNPOL  0003 L 000073 GPRINT  0000 R 000000 HOLD  0000  000014 INJP$  0005  000010 IOPEN
      0003  000067 ITHZT   0005  000011 JOPEN   0005 I 000001 KODE   0000 I 000004 L     0005 L 000023 LFLT
      0007  001214 LOCD    0007  001213 LOCN   0000 I 000007 LP1    0003 L 000072 MODIFY  0004  000707 NDCOEF
      0007  001077 NDCPER  0004  000705 NDGD   0004  000704 NDGN   0006 I 000001 NDSCL  0000 I 000010 NLI
      0004  000726 NECOEF  0007  000764 NNCPER  0004 I 000000 NNSCL  0005 L 000027 NONNAL  0005 L 000030 NOTYET
      0005  000002 NRCLPL  0005  000003 NRPOLE  0005  000004 NRZERO  0007 I 001212 NUMPOL  0005  000005 NXB
      0005  000006 NXN     0005  000007 NXR    0004  000711 NZD    0004  000710 NZN    0003  000004 NZT
      0005  000014 PCPL    0007  000310 PD     0005  000015 PFAC   0007  000000 PNN    0005  000013 PNOM
      0000 R 000006 POWER  0005  000016 PSLOSH  0005  000012 PVAR   0000 R 000005 RESIMG  0000 R 000011 RESREL
      0004 R 000341 RID    0004  000113 RIN    0004 R 000226 RRD    0004  000000 RRN    0005  000000 STAGE
      0003  000000 SUPERK  0003  000003 TD     0005 L 000017 YESMTX  0005 L 000024 YESPCH  0005 L 000020 YESRAW
      0005 L 000025 YESRLP  0005 L 000026 YESSRL  0005 L 000021 YESSRP  0003 L 000070 YESZOH  0003  000002 ZM
      0003  000001 ZT      0003  000005 ZTVAL

0135

0010)    1*          SUBROUTINE PFEZRO(*,N,NCOUNT,NZERO,NDIFF)                          000000
0010)    2*          COMMON/KEEP5/ SUPERK,ZT,ZM,TO,N2T,ZTVAL(50),ITHZT,YESZOH,BOTH,     68400  000000
0010)    3*        1             MODIFY,GPRINT                                        68410  000000
0010)    4*          LOGICAL     YESZOH,BOTH,MODIFY,GPRINT                            68430  000000
0010)    5*          COMMON/KEEP7/ RRD(75),RID(75),RZD(75),RID(75),FNPOL(76),         68440  000000
0010)    6*        1             FDPOL(76),NDGD,NDGN,NDCOEF,NDCOEF,NZN,NZD            68450  000000
0010)    7*          COMMON/KEEP1A/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,       68460  000000
0010)    8*        1             IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,              68490  000000
0010)    9*        2             YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,       68500  000000
```

```
00107     10*          3                YESSRL,NOMNAL,NOTYET                    68510   000000
00107     11*         LOGICAL           YESUTX,YESRAM,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,  68520   000000
00107     12*          1                YESSRL,NOMNAL,NOTYET                    68530   000000
00110     13*         COMMON/KEEP10/ NUSCL,NDSCL                               68540   000000
00111     14*         COMMON/CRUD2/ PNN(200),PD(300),NNCPER(75),NDCPER(75),              000000
00111     15*          1                NUMPOL,LOCN,LOCD                        62000   000000
00112     16*         DIMENSION         HOLD(4)                                 68560   000000
00112     17*    C                                                              68570   000000
00112     18*    C                                                              68580   000000
00112     19*    C  A ZERO POLE                                                 68590   000000
00112     20*    C  USE SPECIAL COMPUTATIONS TO CALCULATE THE RESIDUE           68600   000000
00112     21*    C                                                              68610   000000
00113     22*          L = NDIFF                                                68620   000000
00114     23*          NCOUNT = NCOUNT + 1                                      68630   000002
00115     24*          RESIMG = 0.0                                                     000005
00116     25*          POWER = 1.0                                              68650   000006
00117     26*          IF (NNSCL.NE.NDSCL) POWER=10.0**(NNSCL-NDSCL)            68660   000010
00120     27*          IF (NCOUNT.NE.NZEROS) GO TO 10                           68670   000024
00121     28*    C                                                              68680   000024
00121     29*    C                                                              68690   000024
00121     30*    C  NCOUNT EQUALS NZEROS   ( I.E.   1=1 , 2=2 , 3=3 )           68700   000024
00121     31*    C                                                              68710   000024
00122     32*          LP1 = L + 1                                                      000030
00123     33*          NL1 = NZEROS + L + 1                                             000033
00124     34*          RESREL = FNPOL(LP1)/FDPOL(NL1)                                   000042
00125     35*          HOLD(NZEROS) = RESREL                                    68730   000045
00126     36*          RESREL = RESREL*POWER                                    68740   000046
00127     37*          GO TO 50                                                 68750   000050
00130     38*    C                                                              68760   000050
00130     39*    C                                                              68770   000050
00130     40*    C  NCOUNT DOES NOT EQUAL NZEROS                                68780   000050
00130     41*    C  TEST NCOUNT AND NZEROS AGAIN                                68790   000050
00130     42*    C                                                              68800   000050
00131     43*     10 CONTINUE                                                   68810   000052
00132     44*          IF (NCOUNT.EQ.1 .AND. NZEROS.EQ.2) GO TO 20              68820   000052
00134     45*          IF (NCOUNT.EQ.2 .AND. NZEROS.EQ.3) GO TO 30              68830   000065
00136     46*          IF (NCOUNT.EQ.1 .AND. NZEROS.EQ.3) GO TO 40              68840   000101
00140     47*          GO TO 1000                                               68850   000115
00140     48*    C                                                              68860   000115
00140     49*    C                                                              68870   000115
00140     50*    C  NCOUNT = 1        NZEROS = 2                                68880   000115
00140     51*    C                                                              68890   000115
00141     52*         20 CONTINUE                                               68900   000117
00142     53*          RESREL = (FNPOL(L+2)-HOLD(2)*FDPOL(L+4))/FDPOL(L+3)      68910   000117
00143     54*          RESREL = RESREL*POWER                                    68920   000124
00144     55*          GO TO 50                                                 68930   000126
00146     56*    C                                                              68940   000126
00146     57*    C                                                              68950   000126
00146     58*    C  NCOUNT = 2        NZEROS = 3                                68960   000126
00146     59*    C                                                              68970   000126
00145     60*         30 CONTINUE                                               68980   000130
00146     61*          RESREL = (FNPOL(L+2)-HOLD(3)*FDPOL(L+5))/FDPOL(L+4)      68990   000130
00147     62*          HOLD(4) = RESREL                                         69000   000135
00150     63*          RESREL = RESREL*POWER                                    69010   000136
00151     64*          GO TO 50                                                 69020   000140
00151     65*    C                                                              69030   000140
00151     66*    C                                                              69040   000140
```

```
00150                C  NCOUNT = 1        NZEROS = 3                                  6905D   000140
00151      68*       C                                                                6906D   000140
00152      69*            40 CONTINUE                                                 6907D   000142
00153      70*              RESREL = (FDPOL(L+3)-HOLD(4)*FDPOL(L+5)-HOLD(3)*FDPOL(L+6)) /   6908D   000142
00154      71*             *          FDPOL(L+4)                                      6909D   000142
00155      72*              RESREL = RESREL*POWER                                      6910D   000152
00156      73*       C                                                                6911D   000152
00157      74*       C                                                                6912D   000152
00158      75*       C     PERFORM Z-R TRANSFORMATION                                 6913D   000152
00159      76*       C                                                                6914D   000152
00160      77*            50 CONTINUE                                                 6915D   000155
00161      78*       *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00162      78*              IF (RESREL.EQ.0.0 .AND. RESIMG.EQ.0.0) RETURN              000155
00163      79*              NUMPOL = NUMPOL + 1                                        000167
00164      80*              IF (MODIFY) CALL MODZRO(%200,NCOUNT,NZEROS,RRD(N),RID(N),  000172
00165      81*             *                                RESREL,RESIMG)            000172
00166      82*              IF (.NOT.MODIFY) CALL MLTZRO(%200,NCOUNT,NZEROS,RRD(N),RID(N),  000211
00167      83*             *                                RESREL,RESIMG)            000211
00168      84*              RETURN                                                     6920D   000230
00169      85*       C                                                                6921D   000230
00170      86*       C                                                                6922D   000230
00171      87*       C  TWO MANY ZEROS HAVE BEEN REQUESTED                            6923D   000230
00172      89*       C                                                                6924D   000230
00173      89*         1000 CONTINUE                                                  6925D   000234
00174      90*              KODE = 49                                                 6926D   000234
00175      91*              RETURN 1                                                  6927D   000235
00176      92*       C                                                                6928D   000235
00177      93*       C                                                                6929D   000235
00178      94*       C  ERROR IN Z - R TRANSFORMATION                                 6930D   000235
00179      95*       C                                                                6931D   000235
00180      96*         2000 CONTINUE                                                  6932D   000242
00181      97*              RETURN 1                                                  6933D   000242
00182      98*              END                                                       6934D   000313

      END OF COMPILATION:      1 DIAGNOSTICS.
```

```
OFOR,US# F,P   T,F,POINT
FOR SEIX-05/23/74-09:41:03 (2,3)

SUBROUTINE POINT      ENTRY POINT 001023

STORAGE USED: CODE(1) 001031; DATA(0) 000221 BLANK COMMON(2) 000000

COMMON BLOCKS:

 0003  KEEP1  000026
 0004  KEEP2  000047
 0005  KEEP3  000102
 0006  KEEP5  000074
 0007  KEEP14 000031
 0010  CRUD1  000012
 0011  CRUD3  011650

EXTERNAL REFERENCES (BLOCK, NAME)

 0012  INT1
 0013  INT2
 0014  COS
 0015  NERR35

STORAGE ASSIGNMENT. (BLOCK, TYPE, RELATIVE LOCATION, NAME)

 0001    000045 10L       0001    000045 20L       0001    000063 21L       0001    000312 30L       0001    000324 31L
 0001    000475 33L       0001    000244 40L       0001    000475 40L       0001    000507 41L       0001    000535 42L
 0001    000557 43L       0001    000675 44L       0001    000711 45L       0001    000724 50L       0001    000724 60L
 0001    000764 61L       0001    000742 62L       0001    001003 70L       0001    001010 71L       0011 R  000070 ADIR
 0011 R  000006 AERO      0011 R  011616 AML       0011 R  011617 AMP       0011 R  000152 APHA      0011 R  011620 BIG
 0010 R  000003 BLIVIT    0005 L  000071 BOTH      0010 R  000001 CHAR      0010 R  000010 CHECK     0011 L  011642 DECR
 0007 R  000025 DEG       0010 R  000000 DELTA     0011 R  011621 DF        0004 D  000002 DHFILT    0005    000063 DP
 0011 L  011643 DPI       0007    000012 D0        0007    000013 D1        0007    000014 D2        0007    000015 D3
 0007    000016 D4        0011 L  011645 ERAN      0011 L  011647 ERGP      0011 L  011644 ERP       0011 L  011646 ERPH
 0007 C  000010 FIFTEN    0007    000017 FIFTY     0011 R  011622 FL        0007 C  000006 FOUR      0011 R  011623 FR
 0011 C  000000 GAINS     0006 L  000073 GPRINT    0004    000037 HACC      0007 C  000000 HALF      0004    000025 HAST
 0004    000035 HATT      0004 D  000000 HBLANK    0004 R  000004 HBLK      0004 R  000034 HCCW      0004 R  000033 HCW
 0004 R  000032 HDEC      0004    000027 HDOT      0004    000006 HESTI     0004    000043 HFGD      0004    000044 HFGN
 0004    000040 HFPD      0004    000045 HFPDD     0004    000042 HFPDN     0004    000041 HFPN      0004    000010 HGENE
 0004 R  000031 HINC      0004    000005 HKEY      0004    000007 HMATR     0004    000014 HNEW      0004    000046 HNOMI
 0004    000015 HNYOU     0004    000023 HO        0004    000026 HPLUS     0004    000036 HRATE     0004    000011 HRAN
 0004    000012 HRETA     0004    000017 HROLL     0004    000016 HROOT     0004    000013 HSTAN     0004    000030 HSTAR
 0004    000020 HSIC      0004    000021 HS2       0004    000022 HS4B      0004    000024 HX        0011 R  011624 IM
 0010 L  000011 INCR      0000    000006 INJPS     0011    011634 INT       0012 R  000000 INT1      0013 R  000000 INT2
 0006    000067 ITHZT     0011 I  011641 LMX       0011    011635 LRPR      0005 R  000051 MAX       0003    000000 MAXIT
 0005 R  000037 MIN       0006 L  000072 MODIFY    0011    011636 MPPP      0003    000001 MXEIG     0003    000024 MXEIGT
 0003    000023 MXEST     0003    000002 MXFRM     0003    000003 MXNBH     0003    000004 MXNCOF    0003    000005 MXNCT
 0003    000025 MXNCV     0003    000006 MXNE      0003    000007 MXNEQ     0003    000010 MXNFI     0003    000011 MXNG
 0003    000012 MXNPH     0003    000013 MXNPP     0003 I  000014 MXNQPT    0003    000015 MXNSM     0003 I  000016 MXNSP
 0003    000017 MXNTM     0003    000020 MXNV      0003    000021 MXNZT     0003    001022 MXPOLY    0011 I  011637 NEXT
 0005    000000 NFI       0011 I  000003 NGNPK     0011    011640 NPPP      0011 I  000004 NP1BD     0011 I  000005 NYQPTS
 0006    000004 NZT       0007    000026 N1        0011 I  000002 N1AMP     0007    000027 N2        0007    000030 N3
```

```
0007  000002 ONE      0011 R 000316 PAMP     0005 R 000076 PB       0005  000025 PCT      0011 R 000400 PDIR
0011  011625 PER      0011 R 000234 PFRQ     0011 R 011626 PHA      0011 R 000626 PHAMP    0011 R 000710 PHDIR
0011 R 000544 PHFRQ   0011 R 011627 PHL      0010 R 000005 PHX      0007  000020 PI       0007  000021 PI2
0010 R 000004 PLX      0010 R 000007 PMA      0010 R 000006 PMI      0005 R 000075 PN       0011 R 000462 PPHA
0005  000077 PIAN     0007  000024 RADDEG   0011  011630 RE       0007  000023 RPI      0011 R 003726 SAVAMP
0011 R 000772 SAVFRQ   0011 R 004662 SAVPHA   0011  011631 STA      0007  000022 SMALL    0011 R 011632 STA
0005 L 000101 STNDRD   0011 R 011633 STO      0005  000013 STP      0005  000001 STR      0006  000000 SUPERK
0004  000003 TD       0010 R 000002 TEMP     0007 C 000004 TWO      0005 L 000100 YESNYQ   0006 L 000070 YESZOH
0006  000002 ZM       0006 R 000001 ZT       0006  000005 ZTVAL
```

```
00101     1*          SUBROUTINE POINT                                                    71740   000000
00103     2*          COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,     71750   000000
00103     3*        1             MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,      71760   000000
00103     4*        2             MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV                  71770   000000
00104     5*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,      71780   000000
00104     6*        1             HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HS1C,HS2,HS4B,      71790   000000
00104     7*        2             HU,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,        71800   000000
00104     8*        3             HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPOD,       71810   000000
00104     9*        4             HNOMI                                                  71820   000000
00105    10*          DOUBLE PRECISION HBLANK,DHFILT                                     71830   000000
00106    11*          COMMON/KEEP3/ NFI,STR(10),STP(10),PCT(10),MIN(10),MAX(10),DP(10),  71840   000000
00106    12*        1             PN,PB,P180,YESNYQ,STNDRD                               71850   000000
00107    13*          REAL          MIN,MAX                                              71860   000000
00111    14*          LOGICAL       YESNYQ,STNDRD                                        71870   000000
00111    15*          COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,     71880   000000
00111    16*        1             MODIFY,GPRINT                                          71890   000000
00112    17*          LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                            71910   000000
00113    18*          COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,    71920   000000
00113    19*        1             PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                      71930   000000
00114    20*          COMPLEX       HALF,ONE,TWO,FOUR,FIFTEN                             71940   000000
00115    21*          COMMON/CRUD1/ DELTA,CHAR,TEMP,BLIVIT,PLX,PHX,PMI,PMA,CHECK,INCR    71960   000000
00116    22*          LOGICAL       INCR                                                 71970   000000
00117    23*          COMMON/CRUD3/ GAINS,NIAMP,NGNPK,NP180,NYQPTS,AFRQ(50),ADIR(50),    71980   000000
00117    24*        1             APHA(50),PFRQ(50),PAMP(50),PDIR(50),PPHA(50),        71990   000000
00117    25*        2             PHFRQ(50),PHAMP(50),PHDIR(50),SAVFRQ(1500),          72000   000000
00117    26*        3             SAVAMP(1500),SAVPHA(1500),AML,AMP,BIG,DF,FL,FR,IM,   72010   000000
00117    27*        4             PER,PHA,PHL,RF,SMA,STA,STO,                          72020   000000
00117    28*        5             INT,LRPR,MPPP,NEXT,NPPP,LMX,                         72030   000000
00117    29*        6             DECR,DPI,ERP,ERAM,ERPH,ERGP                          72040   000000
00120    30*          COMPLEX       GAINS                                              72050   000000
00121    31*          REAL          IM                                                 72060   000000
00122    32*          LOGICAL       DECR,DPI,ERP,ERAM,ERPH,ERGP                        72070   000000
00123    33*          REAL          INT1,INT2                                          72080   000000
00123    34*    C                                                                      72090   000000
00123    35*    C                                                                      72100   000000
00123    36*    C     SET GAIN PEAK AND TEST PHASE INDICATORS, STEP NUMBER OF POINTS   72110   000000
00123    37*    C                                                                      72120   000000
00124    38*          CHECK = .0005                                                    72130   000000
00125    39*          LMX = LMX + 1                                                    72140   000001
00125    40*          DECR = .FALSE.                                                   72150   000004
00127    41*          IF (ERP) GO TO 20                                                72160   000005
00131    42*          IF (NYQPTS.EQ.MXNQPT) ERP=.TRUE.                                 72170   000007
00131    43*    C                                                                      72180   000007
00131    44*    C     SAVE PLOT POINTS                                                 72190   000007
```

```
00131    45*      C                                                          72200   000007
00131    46*      C                                                          72210   000007
00131  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00131    47*            IF (PN.EQ.BLK .AND. PB.EQ.BLK) GO TO 10              72220   000023
00135    48*            NYQPTS = NYQPTS + 1                                  72230   000032
00135    49*            SAVAMP(NYQPTS) = AMP                                 72240   000036
00137    50*            SAVPHA(NYQPTS) = PHA                                 72250   000040
00140    51*            SAVERQ(NYQPTS) = FR                                  72260   000042
00141    52*         10 CONTINUE                                             72270   000045
00141    53*      C                                                          72280   000045
00141    54*      C     S A V E    S U M M A R Y    I N F O R M A T I O N     72290   000045
00141    55*      C                                                          72300   000045
00142  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00142    56*         20 IF (FR.EQ.STA) GO TO 60                              72310   000045
00142    57*      C                                                          72320   000045
00142    58*      C     UNIT AMPLITUDES                                      72330   000045
00142    59*      C                                                          72340   000045
00143    60*            IF (ERAM) GO TO 30                                   72350   000050
00145    61*            IF (NIAMP .LT. MXNSP) GO TO 21                       72360   000052
00150    62*            ERAM = .TRUE.                                        72370   000057
00151    63*            GO TO 30                                             72380   000061
00152    64*         21 IF((AMP .LT. 1. .AND. AML .LT. 1.) .OR.              72390   000063
00152    65*        .      (AMP .GT. 1. .AND. AML .GT. 1.)) GO TO 30         72400   000063
00154    66*            NIAMP = NIAMP + 1                                    72410   000117
00155    67*            CHAR = HDEC                                          72420   000122
00155    68*            IF (DF .GT. 0.0) CHAR = HINC                         72430   000124
00161    69*            IF(AML .GT. AMP)  CHAR = HINC                        72440   000132
00162    70*            IF (DF .GT. 0.0 .AND. AML .GT. AMP) CHAR = HDEC      72450   000143
00164    71*            ADIR(NIAMP) = CHAR                                   72460   000162
00165    72*            DELTA = INT1(AML,1.,AMP)                             72470   000165
00165    73*            AFRQ(NIAMP) = INT2(FL,FR,DELTA)                      72480   000173
00167    74*            PMI = AMIN1(PHL,PHA)                                 72490   000202
00170    75*            PMA = AMAX1(PHL,PHA)                                 72500   000210
00171    76*            IF((PMI.LE.BIG) .AND. ((360.-PMA).LE.BIG)) GO TO 4   72510   000217
00174    77*            APHA(NIAMP) = INT2(PHL,PHA,DELTA)                    72520   000234
00174    78*            GO TO 30                                             72530   000242
00175    79*         4  PLX = PHL                                            72540   000244
00175    80*            PHX = PHA                                            72550   000245
00177    81*            IF (PLX.GT.270.)  PLX = PLX - 360.                   72560   000247
00201    82*            IF (PHX.GT.270.)  PHX = PHX - 360.                   72570   000257
00203    83*            TEMP = INT2(PLX,PHX,DELTA)                           72580   000267
00204    84*            IF (TEMP.LT.0.)  TEMP = TEMP + 360.                  72590   000277
00206    85*            APHA(NIAMP) = TEMP                                   72600   000306
00206    86*      C                                                          72610   000306
00206    87*      C     180 DEGREE PHASES                                    72620   000306
00206    88*      C                                                          72630   000306
00207    89*         30 IF (ERPH) GO TO 40                                   72640   000312
00211    90*            IF (NP180 .LT. MXNSP) GO TO 31                       72650   000313
00213    91*            ERPH = .TRUE.                                        72660   000320
00214    92*            GO TO 40                                             72670   000322
00215    93*         31 IF((PHA .LT. 180. .AND. PHL .LT. 180.) .OR.          72680   000324
00215    94*        .      (PHA .GT. 180. .AND. PHL .GT. 180.)) GO TO 40     72690   000324
00217    95*            IF (COS(PHA/DEG) .GE. 0.)  GO TO 40                  72700   000360
00221    96*            DELTA = INT1(PHL,180.,PHA)                           72710   000371
00222    97*            BLIVIT = INT2(AML,AMP,DELTA)                         72720   000377
00223    98*            IF (BLIVIT .LT. CHECK) GO TO 33                      72730   000405
00226    99*            NP180 = NP180 + 1                                    72740   000411
```

```
00228    100*            CHAR = HCCW
00229    101*            IF (DF .GT. 0.0) CHAR = HCW
00230    102*            IF(PHL .LT. PHA) CHAR = HCW
00231    103*            IF (DF .GT. 0.0 .AND. PHL .LT. PHA) CHAR = HCCW
00232    104*            PHDIR(NP180) = CHAR
00233    105*            PHFRQ(NP180) = INT2(FL,FR,DELTA)
00234    106*            PHAMP(NP180) = INT2(AML,AMP,DELTA)
00235    107*         33 CONTINUE
00236    108*      C
00237    109*      C     GAIN PEAKS
00238    110*      C
00239    111*         40 IF(ERGP) GO TO 50
00240    112*            IF(NGNPK.LT.MXNSP) GO TO 41
00241    113*            ERGP = .TRUE.
00242    114*            GO TO 50
00243    115*         41 IF (LMX.GE.3) GO TO 42
00244    116*            INCR = .FALSE.
00245    117*            IF (LMX.EQ.2 .AND. AMP .GT. AML) INCR = .TRUE.
00246    118*            GO TO 50
00247    119*         42 IF (INCR .AND. AMP .LT. AML) GO TO 43
00248    120*            INCR = .FALSE.
00249    121*            IF (AMP .GT. AML) INCR = .TRUE.
00250    122*            GO TO 50
00251    123*         43 INCR = .FALSE.
00252    124*            IF (AML .LT. CHECK) GO TO 50
00253    125*            NGNPK = NGNPK + 1
00254    126*            PFRQ(NGNPK) = FL
00255    127*            PAMP(NGNPK) = AML
00256    128*            PPHA(NGNPK) = PHL
00257    129*            IF (PHA .LT. 90. .AND. PHL .GT. 270.) GO TO 44
00258    130*            IF (PHL .LT. 90. .AND. PHA .GT. 270.) GO TO 45
00259    131*            CHAR = HCCW
00260    132*            IF (DF .GT. 0.0) CHAR = HCW
00261    133*            IF(PHL .LT. PHA) CHAR = HCW
00262    134*            IF (DF .GT. 0.0 .AND. PHL .LT. PHA) CHAR = HCCW
00263    135*            PDIR(NGNPK) = CHAR
00264    136*            GO TO 50
00265    137*         44 PDIR(NGNPK) = HCW
00266    138*            IF (DF.GT.0.0) PDIR(NGNPK) = HCCW
00267    139*            GO TO 50
00268    140*         45 PDIR(NGNPK) = HCCW
00269    141*            IF (DF.GT.0.0) PDIR(NGNPK) = HCW
00270    142*         50 CONTINUE
00271    143*      C
00272    144*      C     I N C R E M E N T    F R E Q U E N C Y
00273    145*      C
00274    146*         60 CONTINUE
00275    147*            IF (DF.LT.0.) GO TO 62
00276    148*            IF (FR.LT.STO) GO TO 61
00277    149*            NEXT = 2
00278    150*            RETURN
00279    151*         62 CONTINUE
00280    152*            IF (FR .GT. STO) GO TO 61
00281    153*            NEXT = 2
00282    154*            RETURN
00283    155*         61 NEXT = 1
00284    156*            FL = FR
```

```
72750    000414
72760    000416
72770    000424
72780    000435
72790    000454
72800    000457
72810    000465
72820    000475
72830    000475
72840    000475
72850    000475
72860    000475
72870    000476
72880    000503
72890    000505
72900    000507
72910    000513
72920    000514
72930    000533
72940    000535
72950    000545
72960    000546
72970    000555
72980    000557
72990    000557
73000    000564
73010    000567
73020    000572
73030    000574
73040    000576
73050    000613
73060    000631
73070    000634
73080    000642
73090    000651
73100    000670
73110    000673
73120    000675
73130    000700
73140    000707
73150    000711
73160    000714
73170    000724
73180    000724
73190    000724
73200    000724
73210    000724
73220    000724
73230    000727
73240    000734
73250    000736
73260    000742
73270    000742
73280    000746
73290    000750
73300    000754
73310    000755
```

```
00335     157*          FR = FR + DF
00336     158*          IF (FR*ZT. .GE. 1.0 .OR. FR .LT. 0.01 GO TO 70
00340     159*          AML = AMP
00341     160*          PHL = PHA
00342     161*          GO TO 71
00343     162*       70 NYQPTS = NYQPTS - 1
00344     163*          NEXT = 2
00345     164*       71 CONTINUE
00346     165*          RETURN
00347     166*          END
```

```
73320    000757
73330    000761
73340    000775
73350    000777
73360    001001
73370    001003
73380    001005
73390    001010
73400    001010
73410    001030
```

END OF COMPILATION: 2 DIAGNOSTICS.

```
@FOR,USN F.POLES,F.POLES
FOR F4X-05/23/74-08:41:40 (2.3)

      SUBROUTINE POLES      ENTRY POINT 000033

     STORAGE USED: CODE(1) 000041; DATA(0) 000010; BLANK COMMON(2) 000000

     COMMON BLOCKS:

      (003  KEEP6  000134
       (004  KEEP9  000705
       (005  KEEP15 000260
       (006  KEEP16 000031

     EXTERNAL REFERENCES (BLOCK, NAME)

       (007  COMPUT
       (010  NERR4$
       (011  NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       (001    000015 100DL     0001   000004 115G.     0003   000064 CARD      0005   000116 CPSLBL    0005   000024 DA
       (005    000130 DCALBL    0006 L 000022 DEBUG     0005   000131 DEGLBL    0004 C 000226 EA        0005   000146 FMAT
       (005    000132 FMT       0005   000162 F6        0005   000163 F7        0000 I 000000 I         0005   000100 ID8
       (005    000107 IDG       0000   000002 INJP$     0006   000010 IOPEN     0006   000011 JOPEN     0004   000572 KD
       (006    000001 KODE      0003 I 000110 LABEL     0005   000164 LABEL1    0005 I 000210 LABEL2    0005   000234 LABEL3
       (006 I  000023 LFLT      0004   000455 NA        0003 C 000062 NAME      0004   000454 NEIG      0004   000457 NI
       (006 L  000027 NORMAL    0006 I 000030 NOTYET    0004 I 000456 NR        0006   000002 NRCLPL    0006 I 000003 NRPOLE
       (006    000004 NRZERO    0006   000005 NXB       0006   000006 NXN       0006   000007 NXR       0006   000014 PCPL
       (006    000015 PFAC      0006   000013 PNOM      0006   000016 PSLOSH    0006   000012 PVAR      0005   000034 P36
       (005 C  000000 REGION    0005   000030 RID       0004 C 000000 ROOT      0005   000014 SPACE     0006   000000 STAGE
       (003    000000 TITLE     0003   000024 TITLE1    0003   000036 TITLE2    0003   000050 TITLE3    0005   000010 WIDTH
       (006 L  000017 YESMTX    0006 L 000024 YESPCH    0006 L 000020 YESRAW    0006 L 000025 YESRLP    0006 L 000026 YESSRL
       (006 L  000021 YESSRP    0005   000020 YINC

                                                                                                                    6-143

00101     1*          SUBROUTINE POLES(A)                                                          734200  000000
00103     2*          COMMON/KEEP6/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),            734300  000000
00103     3*         1          CARD(20),LABEL1(20)                                               734400  000000
00104     4*          REAL          NAME                                                         734500  000000
00105     5*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)                     734600  000000
00106     6*          COMPLEX       ROOT,EA                                                              000000
00107     7*          COMMON/KEEP15/REGION(4),WIDTH(4),SPACE(4),YINC(4),                                 000000
00107     8*         1          FMT(12),RID(4),P36(36),ID8(7),IDG(7),CPSLBL(10),             735000  000000
00107     9*         2          DCBLBL,DEGLBL,FMT(12),FMAT(12),F6,F7,LABEL1(20),             735100  000000
00107    10*         3          LABEL2(20),LABEL3(20)                                        735200  000000
00110    11*          EQUIVALENCE   (REGION                                                  735300  000000
00111    12*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,             735500  000000
00111    13*         1          IOPEN,JOPEN,PVAR,NOW,PCPL,PFAC,PSLOSH,                       735600  000000
00111    14*         2          YESPTX,YESRLP,YESSRL,DEBUG,LFLT,YESPCH,YESRLP,               735700  000000
```

```
00111    15*          3              YESSRL,NOMNAL,NOTYET                    7358₀  000000
00112    16*        LOGICAL          YESMTX,YESRAN,YESSRP,DEBUG,LELT,YESPCH,YESRLP,  7359₀  000000
00113    17*          1              YESSRL,NOMNAL,NOTYET                    7360₀  000000
00112    18*      C                                                         7367₀  000000
00112    19*      C                                                         7368₀  000000
00112    20*      C  PROGRAM CODING                                         7369₀  000000
00112    21*      C                                                         7370₀  000000
00113    22*            NR=NRPOLE                                           7374₀  000000
00114    23*            DO 10 I=1,20                                        7375₀  000004
00117    24*         10 LABEL1(I) = LABEL2(I)                               7376₀  000004
00121    25*            CALL COMPUT($1000)                                        000006
00122    26*            RETURN                                              7378₀  000011
00122    27*      C                                                         7379₀  000011
00122    28*      C  ERROR IN COMPUTING THE OPEN LOOP POLES                 7380₀  000011
00122    29*      C                                                         7381₀  000011
00123    30*       1000 CONTINUE                                            7382₀  000015
00124    31*            RETURN 1                                            7383₀  000015
00125    32*            END                                                 7384₀  000040

      END OF COMPILATION:      NO  DIAGNOSTICS.
```

```
?FOR,US F,F,EVAL,F,POLVAL
FOR SEIX-05/23/74-08:41:55-(2,3)

      SUBROUTINE POLVAL     ENTRY POINT 000104

      STORAGE USED: CODE(1) 000130; DATA(0) 000025; BLANK COMMON(2) 000000

      COMMON BLOCKS:

 0003   KEEP14 000031
 0004   KEEP15 000260
 0005   CRUD3 011650

      EXTERNAL REFERENCES (BLOCK, NAME)

 0006   SYSFRD
 0007   NERR34

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

 0001   000031 123C      0001   000070 2L        0005   000070 ADIR      0005   000006 AFRQ      0005   011616 AML
 0005   011617 AMP       0005   000152 APHA      0005   011620 BIG       0004   000116 CPSLBL    0004   000024 DA
 0004   000130 DCBLBL    0005 L 011642 DECR      0003   000025 DEG       0004   000131 DEGLBL    0005   011621 DF
 0005   011643 DPI       0003   000012 D0        0003   000013 D1        0003   000014 D2        0003   000015 D3
 0003   000016 D4        0005 L 011645 ERAM      0005 L 011647 ERGP      0005 L 011644 ERP       0005 L 011646 ERPH
 0003 C 000010 FIFTEN    0003   000017 FIFTY     0005   011622 FL        0004   000146 FMAT      0004   000132 FMT
 0003 C 000006 FOUR      0005 R 011623 FR        0000 R 000002 FRQ       0004   000162 F6        0004   000163 F7
 0005 C 000000 GAINS     0003 C 000000 HALF      0000 I 000004 I         0004   000100 ID8       0004   000107 IDG
 0005 R 011624 IN        0000   000007 INJPS     0005   011634 INT       0000 I 000005 IS1       0000 I 000003 J
 0004   000169 LABEL1    0004   000210 LABEL2    0004   000234 LABEL3    0005   011641 LMX       0005   011635 LRPR
 0005   011636 MPPP      0005   011637 NEXT      0005   000003 NGNPK     0005   011640 NPPP      0005   000004 NP180
 0005   000005 NYQPTS    0003   000026 N1        0005   000002 NIAMPS    0003   000027 N2        0003   000030 N3
 0003 C 000002 ONE       0005   000316 PAMP      0005   000400 PDIR      0005   011625 PER       0005   000234 PFRQ
 0005   011626 PHA       0005   000626 PHAMP     0005   000710 PHDIR     0005   000544 PHFRQ     0005   011627 PHL
 0003   000020 PI        0003 R 000021 PI2       0005   000462 PPHA      0004   000034 P36       0000 C 000000 R
 0003   000024 RADDEG    0005   011630 RE        0004 C 000000 REGION    0004   000030 RID       0003   000023 RPI
 0005   003726 SAVAMP    0005   000772 SAVFRQ    0005   006662 SAVPHA    0005   011631 SMA       0003   000022 SMALL
 0004   000014 SPACE     0005   011632 STA       0005   011633 STO       0003 C 000004 TWO       0004   000010 WIDTH
 0004   000020 YINC

00101   1*          SUBROUTINE POLVAL (NSTART,NCOEF,POL,EVAL)                                          74090  000004
00102   2*          COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,                    74100  000004
00103   3*         1          PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                                       74110  000004
00104   4*          COMPLEX    HALF,ONE,TWO,FOUR,FIFTEN                                                   000004
00105   5*          COMMON/KEEP15/REGION(4),WIDTH(4),SPACE(4),YINC(4),                                 000004
00106   6*         1          DA(4),RID(4),P36(36),IDB(7),IDG(7),CPSLBL(10),                          74150  000004
00107   7*         2          DCBLBL,DEGLBL,FMT(12),FMAT(12),F6,F7,LABEL1(20),                        74160  000004
00108   8*         3          LABEL2(20),LABEL3(20)                                                   74170  000004
00109   9*          COMPLEX    REGION                                                                  74180  000004
00110  10*          COMMON/CRUD3/ GAINS,NIAMP,NGNPK,NP180,NYQPTS,AFRQ(50),ADIR(50),                    74200  000004

                                                                                                                6-145
```

```
00107    11*          1          APHA(50),PFRQ(50),PAMP(50),PDIR(50),PPHA(50),      74210  000004
00107    12*          2          PHFRQ(50),PHAMP(50),PHDIR(50),SAVFRQ(1500),        74220  000004
00107    13*          3          SAVAMP(1500),SAVPHA(1500),AML,AMP,BIG,DF,FL,FR,IM,  74230  000004
00107    14*          4          PER,PHA,PHL,RE,SMA,STA,STO,                        74240  000004
00107    15*          5          INT,LRPR,MPPP,NEXT,NPPP,LMX,                       74250  000004
00107    16*          6          DECR,DPI,ERP,ERAM,ERPH,ERGP                        74260  000004
00110    17*                COMPLEX    GAINS                                       74270  000004
00111    18*                REAL       IR                                          74280  000004
00112    19*                LOGICAL    DECR,DPI,ERP,ERAM,ERPH,ERGP                 74290  000004
00113    20*                COMPLEX    EVAL,R                                      74300  000004
00114    21*                DIMENSION  POL(300)                                    74310  000004
00114    22*           C                                                           74320  000004
00115    23*           C                                                           74330  000004
00115    24*           C  THIS SUBROUTINE SOLVES POLYNOMIALS IN R                  74340  000004
00115    25*           C  EVAL   = POLYNOMIAL EVALUATION FOR GIVEN VALUE OF R      74350  000004
00115    26*           C  NSTART = START POINT IN POLYNOMIAL ARRAY FOR EACH NEW POLYNOMIAL  74360  000004
00115    27*           C     AND IS LOCATION OF LEADING COEFFICIENT                74370  000004
00116    28*           C  NCOEF  = NUMBER OF COEFFICIENTS FOR EACH POLYNOMIAL SELECTED  74380  000004
00116    29*           C                                                           74390  000004
00116    30*                CALL SYSFRQ (FR,FRQ)                                   74400  000004
00116    31*                R = CMPLX(0.0,-FRQ*PI2)                                74410  000010
00117    32*                EVAL = CMPLX(POL(NSTART),0.0)                                 000015
00120    33*                IF(NCOEF.EQ.1) GO TO 2                                 74430  000020
00120    34*                DO 1 J = 2,NCOEF                                       74440  000024
00120    35*                I = J + 1                                              74450  000031
00120    36*                ISI = NSTART + I                                              000034
00127    37*           1    EVAL = EVAL*R + CMPLX(POL(ISI),0.0)                           000040
00131    38*           2    CONTINUE                                               74470  000070
00132    39*                RETURN                                                 74480  000070
00133    40*                END                                                    74490  000127

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
PFOR,US F.PREVAR,F.PREVAR
FOR SEIX-05/23/74-08:43:09-(2,3)

      SUBROUTINE PREVAR     ENTRY POINT 00004

      STORAGE USED: CODE(1) 000045; DATA(0) 000004; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP2  000047
      0004   KEEP16 000031

      EXTERNAL REFERENCES (BLOCK, NAME)

      0005   VRYGEN
      0006   VRYRAW
      0007   NERR4S
      0010   NERR3S

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000017 IOOL      0001   000025 LOOOL     0001   000011 SOL       0004 L 000022 DEBUG     0003 D 000002 DHFILT
      0003   000037 HACC      0003   000025 HAST      0003   000035 HATT      0003 D 000000 HBLANK    0003 R 000004 HBLK
      0003   000034 HCCW      0003   000033 HCW       0003   000032 HDEC      0003   000027 HDOT      0003   000006 HESTI
      0003   000043 HFGD      0003   000044 HFGN      0003   000040 HFPD      0003   000045 HFPDD     0003   000042 HFPDN
      0003   000041 HFPN      0003   000010 HGENE     0003   000031 HINC      0003   000005 HKEY      0003   000007 HMATR
      0003   000014 HNEW      0003   000046 HNOMI     0003   000015 HNYQU     0003   000023 HO        0003   000026 HPLUS
      0003   000036 HRATE     0003   000011 HRAW      0003   000012 HREFA     0003   000017 HROLL     0003   000016 HROOT
      0003   000013 HSTAN     0003   000030 HSTAR     0003   000020 HSIC      0003   000021 HS2       0003   000022 HS4B
      0003   000024 HX        0003   000000 INJPS     0004   000010 IOPEN     0004   000011 JOPEN     0004   000001 KODE
      0004   000023 LFLT      0004 L 000027 NOMNAL    0004 L 000030 NOTYET    0004   000002 NRCLPL    0004   000003 NRPOLE
      0004   000004 NRZERO    0004   000005 NXB       0004   000006 NXN       0004   000007 NXR       0004   000014 PCPL
      0004   000015 PFAC      0004   000013 PNOM      0004   000016 PSLOSH    0004 R 000012 PVAR      0004   000000 STAGE
      0004 L 000017 YESNTX    0004 L 000024 YESPCH    0004 L 000020 YESRAW    0004 L 000025 YESRLP    0004 L 000026 YESSRL
      0004 L 000021 YESSRP

00101    1*          SUBROUTINE PREVAR(*,*)                                                          000000
00103    2*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,                    20   000000
00103    3*        1             HREFA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS4B,                  30   000000
00103    4*        2             HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,                    40   000000
00103    5*        3             HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,                   50   000000
00103    6*        4             HNOMI                                                              60   000000
00104    7*          DOUBLE PRECISION HBLANK,DHFILT                                                       000000
00105    8*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,                      240   000000
00105    9*        1             IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                           250   000000
00105   10*        2             YESNTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,                    260   000000
00105   11*        3             YESSRL,NOMNAL,NOTYET                                              270   000000
00105   12*          LOGICAL       YESNTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,                  280   000000
00105   13*        1             YESSRL,NOMNAL,NOTYET                                              290   000000
00105   14*    C                                                                                          000000
```

```
00105     15*      C
00105     16*      C  PERFORM VARIATIONS IN GENERAL FORM
00105     17*      C
00107     18*            IF (YESRAW) GO TO 50
00111     19*            CALL VRYGEN(S1000,$100)
00112     20*            RETURN
00112     21*      C
00112     22*      C
00112     23*      C  PERFORM VARIATIONS IN RAW PARAMETER FORM
00112     24*      C
00113     25*         50 CONTINUE
00114     26*            CALL VRYRAW(S1000)
00115     27*            RETURN
00115     28*      C
00115     29*      C
00115     30*      C  GO TO NEXT DATA CASE
00115     31*      C  PARAMETER VARIATIONS ARE COMPLETED
00115     32*      C
00116     33*        100 CONTINUE
00117     34*            PVAR = HALK
00121     35*            RETURN 2
00123     36*      C
00123     37*      C
00123     38*      C  PARAMETER VARIATION ERROR
00123     39*      C
00123     40*       1000 CONTINUE
00122     41*            RETURN 1
00123     42*            END

000000
000000
000000
000000
000001
000005
000005
000005
000005
000005
000011
000011
000013
000013
000013
000013
000013
000013
000017
000017
000020
000020
000020
000020
000020
000025
000025
000044
```

END OF COMPILATION: NO DIAGNOSTICS.

```
@FOR,SUW F.PRINTT,F.PRINTT
FOR 3EIX-05/23/74-08:43:26 (5.6)

      SUBROUTINE PRINTT    ENTRY POINT 000222

      STORAGE USED: CODE(1) 000240; DATA(0) 000103; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP5   000074
      0004   KEEP6   000134
      0005   KEEP10  021620
      0006   KEEP16  000031
      0007   GRUD2   000115

      EXTERNAL REFERENCES (BLOCK, NAME)

      0010   NYOU$
      0011   NIO3$
      0012   NIO2$
      0013   NIO1$
      0014   NERR4$
      0015   NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001  000172 10000L     0001  000134 12L        0001  000136 13L        0001  000032 131G       0001  000041 134G
0001  000073 146G       0001  000125 164G       0001  000160 205G       0000  000010 600F       0000  000023 601F
0000  000035 602F       0000  000052 603F       0007  000024 ANORM      0007  000000 BCD        0003 L 000071 BOTH
0004  000064 CARD       0006 L 000022 DEBUG      0005 R 016664 EV        0003 L 000073 GPRINT     0000 I 000001 I
0007  000075 II         0000  000060 IMJP$      0006 I 000010 IOPEN     0005 I 000004 IR         0003  000067 ITHZT
0000 I 000002 J         0005 I 001754 JC         0007 I 000101 JJ        0006 I 000011 JOPEN      0000 I 000006 K
0007 I 000105 KK        0006 I 000001 KODE       0004  000110 LABEL     0000 I 000004 LBEG       0000 I 000005 LEND
0006 L 000023 LFLT      0005 I 005674 LL         0005 I 007644 LOCPOL    0000 I 000007 M          0003 L 000072 MODIFY
0000 I 000000 N         0004 R 000062 NAME       0005  000003 NCOF      0005 I 003724 ND         0005  000000 NDEG
0005  000002 NE        0005 I 000001 NEQ        0007  000074 NFILT     0006 L 000027 NOMNAL     0006 L 000030 NOTYET
0000 I 000003 NPT       0006  000002 NRCLPL     0006  000003 NRPOLE    0006  000004 NRZERO     0007  000071 NTMP0
0007  000072 NTMP0C     0007  000073 NTMP1C     0004  000005 NXB       0006  000006 NXN        0006  000007 NXR
0003  000004 NZT        0007  000020 OPTINP     0007  000003 OPTTYP     0007  000021 OPT1       0007  000022 OPT2
0007  000023 OPT3       0007 R 000111 PC         0006  000014 PCPL      0006  000015 PFAC       0006  000013 PNOM
0006  000016 PSLOSH     0006  000012 PVAR       0007  000017 REQEST     0006  000000 STAGE      0003  000000 SUPERK
0003  000003 TD         0007  000006 TEMP       0007  000025 TEMP0      0007  000045 TEMP1      0004 R 000000 TITLE
0004  000024 TITLE1     0004  000036 TITLE2     0004  000050 TITLE3     0007  000065 VAL        0007  000011 VFILT
0006 I 000017 YESNTX    0006 L 000024 YESPCH     0006 L 000020 YESRAN    0006 L 000025 YESRLP     0006 L 000026 YESSRL
0004 L 000021 YESSRP    0003 L 000070 YESZOH     0003  000002 ZM        0003  000001 ZT         0003  000005 ZTVAL

00101    1*         SUBROUTINE PRINTT(*)                                                    000002
00103    2*         COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,    76370  000002
00103    3*                     1MODIFY,GPRINT                                        76380  000002
00104    4*         LOGICAL     YESZOH,BOTH,MODIFY,GPRINT                             76400  000002
```

```
00103     5*          COMMON/KEEP6/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2)
00104     6*         1              CARD(20),LABEL(20)
00105     7*          REAL          NAME
00106     8*          COMMON/KEEP10/NDEG,NEQ,NE,NCOF,IR(1000),JC(1000),ND(1000),
00107     9*         1              LL(1000),LOCPOL(60,60),EV(1500)
00110    10*          COMMON/KEEP14/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,
00111    11*         1              IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,
00112    12*         2              YESMTX,YESRAN,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,
00113    13*         3              YESSRL,NOMNAL,NOTYET
00114    14*          LOGICAL       YESMTX,YESRAN,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,
00115    15*         1              YESSRL,NOMNAL,NOTYET
00116    16*          COMMON/CBUD2/ BCD(3),OPTTYP(3),TEMP(3),VFILT(6),REQEST,OPTINP,
00117    17*         1              OPT1,OPT2,OPT3,ANORM,TEMPD(16),TEMPI(16),VAL(4),
00120    18*         2              NTMPD,NTMPDC,NTMPIC,NFILT,II(4),JJ(4),KK(4),PC(4)
00121    19*          C
00122    20*          C
00123    21*          C     PRINT HEADING
00124    22*          C
00125    23*                WRITE(6,600) TITLE
00126    24*            600 FORMAT('1',19X,20A4/20X,'CONTINUOUS SYSTEM MATRIX POLYNOMIAL'//)
00127    25*                WRITE(6,601) IOPEN,JOPEN
00130    26*            601 FORMAT(20X,'SAMPLING DEVICE (RO=',I3,' COLUMN',I3,')'//)
00131    27*                WRITE(6,602)
00132    28*            602 FORMAT(3(/),20X,'N O N Z E R O   M A T R I X   E L E M E N T S
00133    29*           1  .  .  .'//1H )
00134    30*          C
00135    31*          C     PRINT NONZERO POLYNOMIAL COEFFICIENTS
00136    32*          C
00137    33*                N = 0
00140    34*                DO 13 I=1,NEQ
00141    35*                DO 13 J=1,NEQ
00142    36*                NPT = LOCPOL(I,J)
00143    37*                IF (NPT.EQ.0) GO TO 13
00144    38*                IF (IR(NPT).NE.I .OR. JC(NPT).NE.J) GO TO 1000
00145    39*                LBEG = LL(NPT)
00146    40*                LEND = LBEG + ND(NPT) - 1
00147    41*                DO 12 K=LBEG,LEND
00150          *D,AGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00151    42*                IF (EV(K).EQ.0.) GO TO 12
00152    43*                N = N + 1
00153    44*                II(N) = I
00154    45*                JJ(N) = J
00155    46*                KK(N) = K - LBEG
00156    47*                PC(N) = EV(K)
00157    48*                IF (N.NE.4) GO TO 12
00160    49*                N = 0
00161    50*                WRITE (6,603) (II(M),JJ(M),KK(M),PC(M),M=1,4)
00170    51*            603 FORMAT (8X,4(3I3,1PE14.5,5X))
00171    52*             12 CONTINUE
00172    53*             13 CONTINUE
00173    54*                IF (N.EQ.0) RETURN
00174    55*                WRITE (6,603) (II(M),JJ(M),KK(M),PC(M),M=1,N)
00210    56*                RETURN
00211    57*          C
00212    58*          C
00213    59*          C  LOGIC ERROR IN STORING MATRIX
00214    60*          C
```

```
76410     000002
76420     000002
76430     000002
76440     000002
76450     000002
76460     000002
76470     000002
76480     000002
76490     000002
76500     000002
76510     000002
76520     000002
76530     000002
76540     000002
76560     000002
76570     000002
76580     000002
76590     000002
76600     000002
76610     000012
76620     000012
76630     000021
76640     000021
76650     000026
76660     000026
76670     000026
76680     000026
76690     000026
76700     000026
76710     000032
76720     000041
76730     000041
76740     000043
76750     000045
76760     000062
76770     000064
76780     000067
76790     000073
76800     000075
76810     000101
76820     000103
76830     000105
76840     000110
76850     000112
76860     000115
76870     000116
76880     000142
76890     000142
          000142
76900     000142
76910     000147
76920     000166
76930     000166
76940     000166
76950     000166
76960     000166
```

```
00215           1000 CONTINUE                                          7697B    000172
00216     62*        KODE = 52                                         7698B    000172
00217     63*        RETURN 1                                          7699B    000173
00220     64*        END                                               7700B    000237

 END OF COMPILATION:        1 DIAGNOSTICS.
```

```
;FOR,US# F.PUTOUT,F.PUTOUT
FOR SE1x-05/23/74-08:43:57 (2,3)

      SUBROUTINE PUTOUT    ENTRY POINT 000335

      STORAGE USED; CODE(1) 000344; DATA(0) 000201; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003  KEEP5  000074
      0004  KEEP6  000134
      0005  KEEP9  000705
      0006  KEEP14 000031
      0007  KEEP16 000031
      0010  KEEP19 000005
      0011  CRUD1  000463
      0012  CRUD5  000010

      EXTERNAL REFERENCES (BLOCK, NAME)

      0013  XCNG
      0014  N=DU$
      0015  N103$
      0016  N102$
      0017  N101$
      0020  CA$5
      0021  NERR35

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001  000203 11L      0001  000207 12L      0001  000252 154G     0001  000277 171G     0001  000250 21L
0001  000147 211G     0001  000156 215G     0001  000216 232G     0001  000306 275G     0001  000276 30L
0001  000316 303G     0001  000300 40L      0000  000007 600F     0000  000013 601F     0000  000036 602F
0000  000062 603F     0000  000075 604F     0000  000153 605F     0001  000063 9L       0011 R 000226 AMP
0003  000071 BOTH     0004  000064 CARD     0011 R 000345 CPS      0007 L 000022 DEBUG    0006  000025 DEG
0000 R 000001 DEL     0012  000005 DEXPD    0012 C 000002 DOGN     0011 R 000343 DPF      0006  000012 DQ
0006  000013 D1       0006  000014 D2       0006  000015 D3       0006  000016 D4       0005 C 000226 EA
0011 R 000342 EIP     0000 R 000004 EI1      0011 R 000341 ERP      0000 R 000003 ERI      0006 C 000010 FIFTEN
0006  000017 FIFTY    0006 C 000006 FOUR     0003 L 000073 GPRINT   0006 C 000000 HALF     0000 I 000000 I
0000  000166 INJP$    0007 I 000010 IOPEN    0003  000067 ITHZT     0007 I 000011 JOPEN    0005 I 000572 KD
0007  000001 KODF     0011 I 000461 L        0004 I 000110 LABEL    0007 L 000023 LFLT     0010 L 000000 LGAIN
0011 I 000346 LOC     0010 L 000001 LPHASE   0010 L 000002 LPOLES   0010 L 000003 LSDRL    0010 L 000004 LZEROS
0000 I 000306 M       0003 L 000072 MODIFY   0011 I 000462 M1       0000 I 000002 N        0005 I 000455 NA
0004 R 000042 NAME    0005 I 000454 NEIG     0000 I 000005 NEND     0012  000004 NEXPO    0012  000006 NG
0005 I 000457 NI      0007 L 000027 NOMNAL   0007 L 000030 NOTYET   0005  000456 NR       0007  000002 NRCLPL
0007  000003 NRPOLE   0007  000004 NRZERO   0007  000005 NXB      0007  000006 NXN      0007  000007 NXR
0012  000007 NZ       0003  000009 NZT      0006  000026 N1       0006  000027 N2       0006  000030 N3
0006 C 000002 ONE     0007  000014 PCPL     0007  000015 PFAC     0006  000020 PI       0006 R 000021 PI2
0007  000013 PNOM     0007  000016 PSLOSH   0007  000012 PVAR     0006  000024 RADDEG   0005 C 000000 ROOT
0006  000023 RPI      0005 R 000000 RPIP     0011 R 000344 RPS      0011 C 000000 RUTE     0006  000022 SMALL
0007  000000 STAGE    0003  000500 SUPERV   0003  000003 TD       0004 R 000000 TITLE    0004  000024 TITLE1
0000  000035 TITLE2   0004  000056 TITLE3   0004 C 000004 T_0      0012 C 000000 UP       0007 L 000017 YESMTX
0007  000024 YESPCH   0007 L 000020 YESRAN   0007 L 000025 YESRLP   0007 L 000026 YESSRL   0007 L 000021 YESSRP
```

```
0003 L 000070 YESZOH     0003    000002 ZM        0003    000001 ZT        0003    000005 ZTVAL

00101     1*            SUBROUTINE PUTOUT                                          7777B  000000
00102     2*            COMMON/KEEPS/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,   7778B  000000
00103     3*          1               MODIFY,GPRINT                                7779B  000000
00104     4*            LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                    7781B  000000
00105     5*            COMMON/KEEP6/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),  7782B  000000
00106     6*          1               CARD(20),LABEL(20)                           7783B  000000
00107     7*            REAL          NAME                                         7784B  000000
00110     8*            COMMON/KEEP9/ ROOT(75),EA(75),NFIG,NA,NR,NI(75),KD(75)     7785B  000000
00111     9*            COMPLEX       ROOT,EA                                      7786B  000000
00112    10*            COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,    7788B  000000
00113    11*          1               PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3            7789B  000000
00114    12*            COMPLEX       HALF,ONE,TWO,FOUR,FIFTEN                     7790B  000000
00115    13*            COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,  7792B  000000
00116    14*          1               JOPEN,JOPEN,PVAR,PNON,PCPL,PFAC,PSLOSH,      7793B  000000
00117    15*          2               YESNTX,YESRAW,YESRP,DEBUG,LFLT,YESPCH,YESRLP,7794B  000000
00120    16*          3               YESSRL,NONNAL,NOTYET                         7795B  000000
00121    17*            LOGICAL       YESNTX,YESRAW,YESRP,DEBUG,LFLT,YESPCH,YESRLP,7796B  000000
00122    18*          1               YESSRL,NONNAL,NOTYET                         7797B  000000
00123    19*            COMMON/KEEP19/LGAIN,LPHASE,LPOLES,LSDRL,LZEROS                    000000
00124    20*            LOGICAL       LGAIN,LPHASE,LPOLES,LSDRL,LZEROS             7798B  000000
00125    21*            COMMON/CRUD1/ ROTE(75),AMP(75),ERP,EIP,DPF,RPS,CPS,LOC(75),L,M1  000000
00126    22*            COMPLEX       ROTE                                         7801B  000000
00127    23*            DIMENSION     PRIP(150)                                    7802B  000000
00130    24*            EQUIVALENCE   (RPIP(1),ROOT(1))                            7803B  000000
00131    25*            COMMON/CRUDS/ UP,DOWN,NEXPO,DEXPO,NG,NZ                    7804B  000000
00132    26*            INTEGER       DEXPO                                        7805B  000000
00133    27*            COMPLEX       UP,DOWN                                      7806B  000000
00134    28*      C                                                                  7807B  000000
00135    29*      C                                                                  7808B  000000
00136    30*      C     PRINT TITLES                                                      000000
00137    31*      C                                                                  7810B  000002
00140    32*            IF (.NOT.NONNAL .AND. .NOT.GPRINT) GO TO 9                  7811B  000015
00141    33*            WRITE(6,600) TITLE,LABEL                                   7812B  000015
00142    34*        600 FORMAT(1H1,19X,20A4/20X,20A4/)                             7813B  000015
00143    35*      C                                                                  7814B  000015
00144    36*      C     PRINT ADDITIONAL OUTPUT FOR OPEN LOOP ZERO OR GAIN VARIATION       000015
00145    37*      C                                                                  7817B  000026
00146    38*            IF (LZEROS) WRITE(6,601) IOPEN,JOPEN                       7817B  000037
00147    39*            IF (LPOLES) WRITE(6,602) IOPEN,JOPEN                       7818B  000037
00150    40*        601 FORMAT(3(/),20X,'OPEN LOOP ZERO COMPUTATION - SYSTEM OPENED AT MA  7819B  000037
00151    41*          1 TRIX ELEMENT LOCATION'//20X,'ROW',I3,', COLUMN',I3)        7820B  000037
00152    42*        602 FORMAT(3(/),20X,'OPEN LOOP POLE COMPUTATION - SYSTEM OPENED AT MAT  7821B  000037
00153    43*          1 RIX ELEMENT LOCATION'//20X,'ROW',I3,', COLUMN',I3)         7822B  000037
00154    44*      C                                                                  7823B  000037
00155    45*      C     PRINT ESTIMATES                                              7824B  000037
00156    46*      C                                                                  7825B  000055
00157    47*            IF (NA.NE.0) WRITE(6,603) (EA(I),I=1,NA)                   7826B  000055
00160    48*        603 FORMAT(////,20X,'EIGENVALUE ESTIMATES - - - -',///,17X,1PE14.5,   7827B  000055
00161    49*          *       5E14.5))                                             7826B  000055
00162    50*      C                                                                  7829B  000055
00163    51*      C     PRINT COLUMN HEADINGS
00164    52*      C
```

```
00163          WRITE (6,604)                                                  78300   000055
00163    54*   604 FORMAT (3(/),41X,'E I G E N V A L U E S'//                  78310   000063
00163    55*       1 99X,'F R E Q U E N C Y' /2X,'ROOT',8X,'ITERATION',12X,'REAL',12X  78320   000063
00163    56*       2 'IMAGINARY',14X,'DAMPING' /1X,'NUMBER',1X,'SEQUENCE',2X,'CODE',4X  78330   000063
00163    57*       3 'COUNT',14X,'PART',14X,'PART',18X,'RATIO',9X,'RAD/SEC',8X,'HZ'  78340   000063
00163    58*       4 /1H )                                                    78350   000063
00164    59*     9 CONTINUE                                                   78360   000063
00164    60*   C                                                              78370   000063
00164    61*   C     ARRANGE EIGENVALUES IN ORDER OF DECREASING ABSOLUTE VALUE  78380   000063
00164    62*   C                                                              78390   000063
00165    63*         IF (NEIG.EQ.0) RETURN                                    78400   000063
00167    64*         DEL = .0001                                                      000067
00170    65*         DO 10 N=1,NEIG                                           78410   000071
00173    66*         ER1 = RPIP(2*N-1)                                                000077
00175    67*         EI1 = RPIP(2*N)                                                  000101
00175    68*         IF (ABS(ER1) .LT. DEL) ER1=0.0                                   000103
00177    69*         IF (ABS(EI1) .LT. DEL) EI1=0.0                                   000111
00201    70*         ROOT(N) = CMPLX(ER1,EI1)                                         000117
00202    71*         LOC(N) = N                                               78420   000122
00203    72*      10 AMP(N) = CABS(ROOT(N))                                           000124
00205    73*         IF (NEIG.EQ.1) GO TO 12                                  78440   000134
00206    74*         NEND = NEIG - 1                                          78450   000140
00210    75*         DO 11 N=1,NEND                                           78460   000143
00213    76*         M1 = N + 1                                               78470   000147
00215    77*         DO 11 M=M1,NEIG                                          78480   000152
00217    78*         IF (AMP(N).LE.AMP(M)) GO TO 11                           78490   000156
00221    79*         CALL XCNG (LOC(M),LOC(N))                                78500   000162
00222    80*         CALL XCNG (AMP(M),AMP(N))                                78510   000172
00223    81*      11 CONTINUE                                                 78520   000207
00225    82*      12 CONTINUE                                                 78530   000207
00224    83*   C                                                              78540   000207
00224    84*   C     PRINT OUTPUT DATA.                                       78550   000207
00224    85*   C                                                              78560   000207
00227    86*         IF (.NOT.NORMAL .AND. .NOT.PRINT) GO TO 40                       000207
00231    87*         DO 30 N=1,NEIG                                           78580   000216
00233    88*         L = LOC(N)                                               78590   000216
00235    89*         ERP = RPIP(2*L-1)                                        78600   000222
00236    90*         EIP = RPIP(2*L)                                          78610   000224
00237    91*         IF (ABS(EIP) .GE. 0.0001) GO TO 21                               000226
00241    92*         WRITE (6,605) N,L,KD(L),NI(L),ERP,EIP                    78630   000233
00251    93*         GO TO 30                                                 78640   000246
00252    94*      21 DPF = - ERP/AMP(N)                                       78650   000250
00254    95*         RPS = AMP(N)                                             78660   000253
00254    96*         CPS = RPS/PI2                                            78670   000255
00256    97*         WRITE (6,605) N,L,KD(L),NI(L),ERP,EIP,DPF,RPS,CPS        78680   000257
00273    98*     605 FORMAT (1X,I4,3I8,1P6E25.7,E18.7,E20.4,2X,2E13.4)        78690   000300
00271    99*      30 CONTINUE                                                 78700   000300
00271   100*   C                                                              78710   000300
00271   101*   C     REARRANGE ROOTS                                          78720   000300
00271   102*   C                                                              78730   000300
00271   103*      40 CONTINUE                                                 78740   000300
00271   104*         DO 51 N=1,NEIG                                           78750   000300
00277   105*         L = LOC(N)                                               78760   000306
00301   106*      51 ROTE(N) = ROOT(L)                                        78770   000307
00311   107*         DO 52 N=1,NEIG                                           78780   000316
00315   108*      52 ROOT(N) = ROTE(N)                                        78790   000316
00317   109*         RETURN                                                   78800   000320
```

```
00311                    END                                    78810    000343

      END OF COMPILATION:      NO  DIAGNOSTICS.
```

```
FOR,US F,RAWMTX,F,RAWMTX
FOR SEIX=05/23/79=08:49:29 (2.3)

      SUBROUTINE RAWMTX     ENTRY POINT 000015

      STORAGE USED: CODE(1) 000021; DATA(0) 000004; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003  KEEP16 000031

      EXTERNAL REFERENCES (BLOCK, NAME)

      0004  NERR45
      0005  NERR35

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0003 L 000022 DEBUG     0003   000000 INJPS     0003   000010 IOPEN     0003   000011 JOPEN     0003 I 000001 KODE
      0003 L 000023 LFLT      0003 L 000027 NOMNAL    0003 L 000030 NOTYET    0003   000002 NRCLPL    0003   000003 NRPOLE
      0003   000004 NRZERO    0003   000005 NXB       0003   000006 NXN       0003   000007 NXR       0003   000014 PCPL
      0003   000015 PFAC      0003   000013 PNOM      0003   000016 PSLOSH    0003   000012 PVAR      0003   000000 STAGE
      0003 L 000017 YESMTX    0003 L 000024 YESPCH    0003 L 000020 YESRAW    0003 L 000025 YESRLP    0003 L 000026 YESSRL
      0003 L 000021 YESSRP

6-156

00101     1*          SUBROUTINE RAWMTX(*)                                                    000000
00103     2*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,          24B 000000
00103     3*         1          IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                   25B 000000
00103     4*          2         YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,            26B 000000
00103     5*          3         YESSRL,NOMNAL,NOTYET                                      27B 000000
00104     6*          LOGICAL   YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,            28B 000000
00104     7*         1          YESSRL,NOMNAL,NOTYET                                      29B 000000
00104     8*      C                                                                          000000
00104     9*      C                                                                          000000
00104    10*      C  RAW PARAMETER MATRIX DEFINITION NOT PERMITTED                           000000
00104    11*      C                                                                          000000
00105    12*          KODE = 106                                                             000000
00105    13*          RETURN 1                                                               000001
00107    14*          END                                                                    000020

      END OF COMPILATION:  NO DIAGNOSTICS.
```

```
PROP,USR PARSET,PARESET
FOR SEDX-05/23/74-09:45:28-01.21

SUBROUTINE RESET      ENTRY POINT 000110

STORAGE USED: CODE(1) 000112; DATA(0) 000004; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003   KEEP2   000047
0004   KEEP3   000102
0005   KEEP4   000263
0006   KEEP5   000074
0007   KEEP7   000712
0010   KEEP8   000342
0011   KEEP9   000705
0012   KEEP10  021620
0013   KEEP11  007723
0014   KEEP13  000313
0015   KEEP16  000031
0016   KEEP19  000005
0017   KEEP20  000227
0020   PLT     000012

EXTERNAL REFERENCES (BLOCK, NAME)

0021  .NERR3$

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0006 L 000071 80TH    0005   000212 BY      0005   000166 DD      0015 L 000022 DEBUG   0003 D 000002 DHFILT
0020   000007 DIF1    0020   000010 DIF2    0004   000063 DP      0011 C 000226 EA      0017 C 000000 ESTZ
0012   016664 EV      0007   000570 FDPOL   0007   000454 FNPOL   0005   000004 GAIN    0006 L 000073 GPRINT
0005   000151 GSYM    0003   000037 HACC    0003   000025 HAST    0003   000035 HATT    0003 D 000000 HBLANK
0003 R 000004 HBLK    0003   000034 HCCW    0003   000033 HCA     0003   000032 HDEC    0003   000027 HDOT
0003   000006 HESTI   0003   000043 HFGD    0003   000044 HFGN    0003   000040 HFPD    0003   000045 HFPDD
0003   000042 HFPDN   0003   000041 HFPN    0003   000010 HGENE   0003   000031 HINC    0003   000005 HKEY
0003   000007 HMATR   0003   000014 HNEW    0003   000046 HNOMI   0003   000015 HNYQU   0003   000023 HO
0003   000026 HPLUS   0003   000036 HRATE   0003   000011 HRAW    0003   000012 HRETA   0003   000017 HROLL
0003   000016 HROOT   0003   000013 HSTAN   0003   000030 HSTAR   0003   000020 HS1C    0003   000021 HS2
0003   000022 HS4B    0003   000024 HX      0020   000011 ICK     0020   000004 ICT     0000   000000 INJP$
0015 I 000010 IOPEN   0012   000004 IR      0020   000003 ISW     0004   000067 ITHZT   0012   001754 JC
0015 I 000011 JOPEN   0011   000572 KD      0013   005670 KDS     0015 I 000001 KODE    0015 L 000023 LFLT
0016 L 000000 LGAIN   0012   025674 LL      0012   007644 LOCPOL  0014   000144 LOCV    0016 L 000001 LPHASE
0016 L 000002 LPOLES  0016 L 000003 LSDRL   0014 L 000004 LZEROS  0004 R 000051 MAX     0004 R 000037 MIN
0006 L 000072 MODIFY  0011 I 000455 NA      0012 I 000003 NCOF    0014 I 000312 NCOFV   0012   003724 ND
0007 I 000707 NDCOEF  0012 I 000000 NDEG    0007 I 000705 NDGD    0007 I 000704 NDGN    0012 I 000002 NE
0011 I 000454 NEIG    0013 I 007722 NEIGZT  0012 I 000001 NEG     0017 I 000226 NESTZ   0014 I 000311 NEV
0004 I 000000 NFI     0005 I 000003 NGAIN   0005   000154 NGR     0011   000457 NI      0020 L 000001 NICPLT
0013   003720 NIS     0007 I 003706 NNCOEF  0015 L 000027 NOMNAL  0015 L 000030 NOTYET  0020   000002 NP
0005 I 000046 NPHASE  0011 I 000456 NR      0015 I 000002 NRCLPL  0005 I 000153 NRLFR   0015 I 000003 NRPOLE
0015 I 007704 NRZERO  0013   007670 NSEIG   0014 I 000310 NV      0015   000005 NXB     0015   000006 NXN
0015   000007 NXR     0007 I 000711 NZD     0007 I 000710 NZT     0010 I 000341 NZPOLE  0006 I 000004 NZT
```

```
0004 R 000076 PB          0015 R 000014 PCPL        0004   000025 PCT         0015 R 000015 PFAC        0005 R 000000 PG
0005   000067 PHASE       0004 R 000075 PN          0020 R 000000 PNI         0015 R 000013 PNOM        0005 R 000001 PP
0005 R 000002 PPLT        0015 R 000016 PSLOSH      0005   000152 PSYM        0015 R 000012 PVAR        0004 R 000077 P180
0007   000341 RID         0007   000113 RIN         0011 C 000000 ROOT        0007   000226 RRD         0007   000000 RRN
0005   000200 RX          0013 C 000000 SEIG        0015 R 000000 STAGE       0004 L 000101 STNDRD      0004   000013 STP
0004   000001 STR         0006   000000 SUPERK      001A   000000 SV          0020   000006 S360        0006   000003 TD
0020   000305 T360        0015 L 000017 YESMTX      0004 L 000100 YESNYQ      0015 L 000024 YESPCH      0015 L 000020 YESRAW
0005   000242 YESRL       0015 L 000025 YESRLP      0015 L 000026 YESSRL      0015 L 000021 YESSRP      000A L 000070 YESZOH
0004   000002 ZM          0010   000226 ZMAG        0010 C 000000 ZPOLE       0006   000001 ZT          0006   000005 ZTVAL
```

```
00101     1*          SUBROUTINE RESET                                                                    79750  000000
00102     2*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,                       7976₀  000000
00103     3*         1            HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS4B,                    7977₀  000000
00103     4*         2            HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,                      7978₀  000000
00103     5*         3            HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,                     7979₀  000000
00103     6*         4            HNOMI                                                                7980₀  000000
00104     7*          DOUBLE PRECISION HBLANK,DHFILT                                                              000000
00105     8*          COMMON/KEEP3/ NFI,STR(10),STP(10),PCT(10),MIN(10),MAX(10),DP(10),                   7982₀  000000
00105     9*         1            PN,PD,P180,YESNYQ,STNDRD                                             7983₀  000000
00106    10*          REAL         MIN,MAX                                                              7984₀  000000
00107    11*          LOGICAL      YESNYQ,STNDRD                                                        7985₀  000000
00110    12*          COMMON/KEEP4/ PG,PP,PPLT,NGAIN,GAIN(50),NPHASE,PHASE(50),GSYM,                      7986₀  000000
00110    13*         1            PSYM,NRLER,NGR(10),DD(10),RX(10),BY(4,10),YESRL                      7987₀  000000
00111    14*          LOGICAL      YESRL                                                                7988₀  000000
00112    15*          COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,                      7989₀  000000
00112    16*         1            MODIFY,GPRINT                                                        7990₀  000000
00113    17*          LOGICAL      YESZOH,BOTH,MODIFY,GPRINT                                            7992₀  000000
00114    18*          COMMON/KEEP7/ RRN(75),RIN(75),RRD(75),RID(75),FNPOL(76),                          7993₀  000000
00114    19*         1            FDPOL(76),NDGN,NDGD,NNCOEF,NDCOEF,NZN,NZD                            7994₀  000000
00115    20*          COMMON/KEEP8/ ZPOLE(75),ZMAG(75),NZPOLE                                             7996₀  000000
00115    21*          COMPLEX      ZPOLE                                                                       000000
00117    22*          COMMON/KEEP9/ ROOT(75),EA(75),NFIG,NA,NR,NI(75),KD(75)                              7999₀  000000
00123    23*          COMPLEX      ROOT,EA                                                                     000000
00121    24*          COMMON/KEEP10/NDEG,NEQ,NE,NCOF,IR(1000),JC(1000),ND(1000),                          8002₀  000000
00121    25*         1            LL(1000),LOCPOL(60,60),EV(1500)                                      8003₀  000000
00122    26*          COMMON/KEEP11/SIG(1000),NIS(1000),KDS(1000),NSEIG(50),NEIGZT                        8004₀  000000
00123    27*          COMPLEX      SEIG                                                                 8005₀  000000
00124    28*          COMMON/KEEP13/SV(100),LOCV(100),NV,NEV,NCOFV                                        8006₀  000000
00125    29*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,                          8007₀  000000
00125    30*         1            IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                              8008₀  000000
00125    31*         2            YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,                       8009₀  000000
00125    32*         3            YESSRL,NOMNAL,NOTYET                                                 8010₀  000000
00126    33*          LOGICAL      YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,                       8011₀  000000
00126    34*         1            YESSRL,NOMNAL,NOTYET                                                 8012₀  000000
00127    35*          COMMON/KEEP19/LGAIN,LPHASE,LPOLES,LSORL,LZEROS                                              000000
00130    36*          LOGICAL      LGAIN,LPHASE,LPOLES,LSORL,LZEROS                                            000000
00131    37*          COMMON/KEEP20/ESTZ(75),NESTZ                                                            000000
00132    38*          COMPLEX      ESTZ                                                                       000000
00133    39*          COMMON /PLT/ PNI,NICPLT,NP,ISW,ICT,T360,S360,DIF1,DIF2,ICK                              000000
00134    40*          LOGICAL NICPLT                                                                          000000
00134    41*     C                                                                                            000000
00134    42*     C  RESET NICHOLS PLOT OPTION                                                                 000000
00134    43*     C                                                                                            000000
00135    44*          PNI = PBLK                                                                              000000
```

```
00135    45*     C
00135    46*     C
00135    47*     C  RESET Z - R TRANSFORMATION DATA
00135    48*     C
00135    49*           NZT = 0
00137    50*           NZPOLE = 0
00141    51*           NESTZ = 0
00141    52*           MODIFY = .FALSE.
00142    53*           BOTH = .FALSE.
00143    54*           YESZOH = .FALSE.
00143    55*     C
00143    56*     C
00143    57*     C  RESET NYQUIST DATA
00143    58*     C
00144    59*           NFI = 0
00146    60*           PN = HBLK
00146    61*           PR = HBLK
00147    62*           PIeD = HBLK
00150    63*           YESNYQ = .FALSE.
00151    64*           STABRD = .FALSE.
00151    65*     C
00151    66*     C
00151    67*     C  RESET ROOT LOCUS DATA
00151    68*     C
00152    69*           NGAIN = 0
00153    70*           NPHASE = 0
00154    71*           NPLFP = 0
00155    72*           PG = HBLK
00156    73*           PP = HBLK
00157    74*           PPLT = HBLK
00160    75*           YESRL = .FALSE.
00161    76*           YESRLP = .FALSE.
00161    77*     C
00161    78*     C
00161    79*     C  RESET USER ESTIMATES
00161    80*     C
00162    81*           NA = 0
00162    82*     C
00162    83*     C
00162    84*     C  RESET GENERAL PARAMETERS
00162    85*     C
00163    86*           KODE = 0
00164    87*           NRCLPL = 0
00165    88*           NRPOLE = 0
00167    89*           NRZERO = 0
00167    90*           IOPEN = 0
00170    91*           JOPEN = 0
00171    92*           NEIG = 0
00172    93*           NR = 0
00173    94*           NDEG = 0
00174    95*           NEQ = 0
00175    96*           NE = 0
00176    97*           NCOF = 0
00177    98*           NDGN = 0
00200    99*           NDGD = 0
00201   100*           NNCOEF = 0
00202   101*           NDCOEF = 0
```

```
8013₀  000000
8014₀  000000
8015₀  000000
8016₀  000000
8017₀  000001
8018₀  000002
       000003
8019₀  000004
8020₀  000005
8021₀  000006
8022₀  000006
8023₀  000006
8024₀  000006
8025₀  000006
8026₀  000007
8027₀  000010
8028₀  000011
8029₀  000012
8030₀  000013
8031₀  000014
8032₀  000014
8033₀  000014
8034₀  000014
8035₀  000014
8036₀  000015
8037₀  000016
8038₀  000017
8039₀  000020
8040₀  000021
8041₀  000022
8042₀  000023
8043₀  000024
8044₀  000024
8045₀  000024
8046₀  000024
8047₀  000024
8048₀  000025
8049₀  000025
8050₀  000025
8051₀  000025
8052₀  000025
8053₀  000026
8054₀  000027
8055₀  000030
8056₀  000031
8057₀  000032
8058₀  000033
8059₀  000034
8060₀  000035
8061₀  000036
8062₀  000037
8063₀  000040
8064₀  000041
8065₀  000042
8066₀  000043
8067₀  000044
8068₀  000045
```

```
00203    102*          NZN = C                                  8069₀    000046
00204    103*          NZD = D                                  8070₀    000047
00205    104*          NV = 0                                   8071₀    000050
00206    105*          NEV = 0                                  8072₀    000051
00207    106*          NCOFV = 0                                8073₀    000052
00210    107*          NEIGZT = 0                               8074₀    000053
00211    108*          PNOM = HBLK                              8075₀    000054
00212    109*          PCPL = HBLK                              8076₀    000055
00213    110*          PFAC = HBLK                              8077₀    000056
00214    111*          PSLOSH = HBLK                            8078₀    000057
00215    112*          PVAR = HBLK                              8079₀    000060
00216    113*          STAGE = HBLK                             8080₀    000061
00217    114*          YESMTX = .FALSE.                         8081₀    000062
00220    115*          YESRAN = .FALSE.                         8082₀    000063
00221    116*          LFLT = .FALSE.                           8083₀    000064
00222    117*          YESSRL = .FALSE.                         8084₀    000065
00223    118*          YESSRP = .FALSE.                         8085₀    000066
00224    119*          DEBUG = .FALSE.                          8086₀    000067
00225    120*          GPRINT = .FALSE.                         8087₀    000070
00226    121*          LSORL = .FALSE.                                   000071
00227    122*          LGAIN = .FALSE.                                   000072
00230    123*          LPHASE = .FALSE.                                  000073
00231    124*          LPOLES = .FALSE.                                  000074
00232    125*          NOMNAL = .FALSE.                         8089₀    000075
00233    126*          LZEROS = .FALSE.                                  000076
00234    127*          NOTYET = .TRUE.                          8090₀    000077
00235    128*          RETURN                                   8091₀    000101
00236    129*          END                                      8092₀    000111

          END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,US F.RLOCUS,F.RLOCUS
FOR SE1X-05/23/74-08:46:31 (2.3)

      SUBROUTINE RLOCUS    ENTRY POINT 000132

      STORAGE USED: CODE(1) 0n0145  DATA(D) 0n0102  BLANK COMMON(2) 000000

      COMMON BLOCKS:

      C003  KEEP10 021620
      C004  KEEP16 000031

      EXTERNAL REFERENCES (BLOCK, NAME)

      C005  ROOTER
      C006  PUTOUT
      C007  NJOU$
      C010  NI02$
      C011  NI01$
      C012  NERR4$
      C013  NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      C001  000071 10L      0001  000014 114G     0001  000023 117G     0001  000065 134G     0001  000076 20L
      C001  000105 2000L    0000  000006 600F     0000  000052 601F     0004 L 000022 DEBUG    0003 R 016664 EV
      C000 I 000000 I       0000  000062 INJP$    0000  000010 IOPEN    0003 I 000004 IR       0000 I 000001 J
      C003 I 001754 JC      0004  000011 JOPEN    0000  000005 K        0004 I 000001 KODE     0000 I 000003 LREG
      C000 I 000004 LEND    0004 L 000023 LFLT    0003 I 005674 LL       0003 I 007644 LOCPOL   0003   000003 NCOF
      C003 I 003724 ND      0003  000000 NDEG     0003  000002 NE       0003 I 000001 NEQ      0004 L 000027 NOMNAL
      C004 L 000030 NOTYET  0000 I 000002 NPT     0000  000002 NRCLPL   0004  000003 NRPOLE   0004   000004 NRZERO
      C004  000006 NXB      0004  000006 NXN      0004  000007 NXR      0004  000014 PCPL     0004   000015 PFAC
      C004  000013 PNOM     0004  000016 PSLOSH   0000  000012 PVAR     0004  000000 STAGE    0004 L 000017 YESNTX
      C004 L 000024 YESPCH  0004 L 000020 YESRAW  0004 L 000025 YESRLP  0004 L 000026 YESSRL  0004 L 000021 YESSRP

00101     1*          SUBROUTINE RLOCUS(*)                                          80930  000000
00102     2*          COMMON/KEEP10/NDEG,NEQ,NE,NCOF,IR(1000),JC(1000),ND(1000),    80940  000000
00103     3*         1          LL(1000),LOCPOL(60,60),EV(150)                      80950  000000
00104     4*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,    80960  000000
00104     5*         1          IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,             80970  000000
00104     6*         2          YESNTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      80980  000000
00104     7*         3          YESSRL,NOMNAL,NOTYET                                80990  000000
00105     8*          LOGICAL    YESNTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      81000  000000
00105     9*         1          YESSRL,NOMNAL,NOTYET                                81010  000000
00105    10*     C                                                                   81020  000000
00105    11*     C                                                                   81030  000000
00106    12*     C    DEBUGGING OUTPUT                                               81040  000000
00107    13*     C                                                                   81050  000000
00107    14*          IF (.NOT.DEBUG) GO TO 20                                       81060  000000
00110    15*          WRITE (6,600)                                                  81070  000001
```

```
00112      16*      600 FORMAT (1H1,3(/),20X,'B E G I N N I N G     O F     D     81080   000006
00112      17*     1E B U G G I N G     O U T P U T :4(/),20X,'M A T R I X     81090   000006
00112      18*     2    P O L Y N O M I A L'3(/),3X,'ROW'1X'COLUMN'7X,'POLYNOMIAL COEF  81100   000006
00112      19*     3FICIENTS'/1H )                                                     81110   000006
00113      20*           DO 10 I=1,NEQ                                                 81120   000006
00116      21*           DO 10 J=1,NEQ                                                 81130   000023
00121      22*           NPT = LOCPOL(I,J)                                             81140   000023
00122      23*           IF (NPT.EQ.0) GO TO 10                                        81150   000025
00124      24*           IF (IR(NPT).NE.I .OR. JC(NPT).NE.J) GO TO 2000                81160   000027
00126      25*           LBEG = LL(NPT)                                                81170   000044
00127      26*           LEND = LBEG + ND(NPT) - 1                                     81180   000046
00130      27*           WRITE(6,601) I,J,(EV(K),K=LBEG,LEND)                          81190   000051
00140      28*      601  FORMAT(1X,2I5,1PE20.5,6E14.5/(17X,7E14.5))                    81200   000076
00141      29*       10  CONTINUE                                                      81210   000076
00148      30*       20  CONTINUE                                                      81220   000076
00145      31*     C                                                                   81230   000076
00145      32*     C     CALCULATE EIGENVALUES                                         81240   000076
00145      33*     C                                                                   81250   000076
00145      34*           CALL ROOTER                                                           000076
00145      35*     C                                                                   81270   000076
00145      36*     C     PRINT CHARACTERISTIC ROOTS (EIGENVALUES)                      81280   000076
00145      37*     C                                                                   81290   000076
00147      38*           CALL PUTOUT                                                   81300   000077
00147      39*           RETURN                                                        81310   000101
00147      40*     C                                                                   81380   000101
00147      41*     C                                                                   81390   000101
00147      42*     C  LOGIC ERROR STORING MATRIX                                       81400   000101
00147      43*     C                                                                   81410   000101
00150      44*      2000 CONTINUE                                                      81420   000105
00151      45*           KODE = 54                                                     81430   000105
00152      46*           RETURN 1                                                      81440   000106
00153      47*           END                                                           81450   000144

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
?FOR,US F,RLPLOT,F,RLPLOT
FOR (EIX-05/23/74-08:46:47-(2,3)

    SUBROUTINE RLPLOT    ENTRY POINT 000011

  STORAGE USED: CODE(1) 000013; DATA(0) 000004; BLANK COMMON(2) 000000

  COMMON BLOCKS:

  (003 - KEEP16 000034

  EXTERNAL REFERENCES (BLOCK, NAME)

    0004   NERR3S

  STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

  0003 L 000022 DEBUG     0003  000000 INJPS    0003  000010 IOPEN    0003  000011 JOPEN    0003  000001 KODE
  0003 L 000023 LFLT      0003 L 000027 NOMNAL  0003 L 000030 NOTYET  0003  000002 NRCLPL   0003  000003 NRPOLE
  0003   000004 NRZERO    0003  000005 NXB     0003  000006 NXN     0003  000007 NXR      0003  000014 PCPL
  0003   000015 PFAC      0003  000013 PNOM    0003  000016 PSLOSH   0003  000012 PVAR     0003  000000 STAGE
  0003 L 000017 YESMTX    0003 L 000024 YESPCH  0003 L 000020 YESRAW  0003 L 000025 YESRLP  0003 L 000026 YESSRL
  0003 L 000021 YESSRP

00101     1*          SUBROUTINE RLPLOT                                                    000000
00102     2*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,          240  000000
00103     3*         1                 IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,          250  000000
00103     4*         2                 YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,   260  000000
00103     5*         3                 YESSRL,NOMNAL,NOTYET                             270  000000
00104     6*          LOGICAL          YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,   280  000000
00105     7*         1                 YESSRL,NOMNAL,NOTYET                             290  000000
00106     8*     C                                                                              000000
00107     9*     C                                                                              000000
00108     10*    C   DUMMY SUBROUTINE FOR ROOT LOCUS PLOT                                       000000
00108     11*    C                                                                              000000
00105     12*         NXR = NXR + 1                                                          000002
00106     13*         RETURN                                                                 000012
00107     14*         END

      END OF COMPILATION:      NO  DIAGNOSTICS.
```

```
@FOR,USN F,RLPRNT,F-RLPRNT
FOR NEIX-05/23/74-08:46:53 (6.7)

    SUBROUTINE RLPRNT    ENTRY POINT 000473

    STORAGE USED: CODE(1) 000502; DATA(0) 000322; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003  KEEP5  000074
    0004  KEEP6  000134
    0005  KEEP9  000705
    0006  KEEP14 000031
    0007  KEEP16 000031
    0010  KEEP19 000005
    0011  CRUD1  000241
    0012  CRUD2  003737

    EXTERNAL REFERENCES (BLOCK, NAME)

    0013  XCNG
    0014  DET
    0015  PEVAL
    0016  N.OUS
    0017  NIO3S
    0020  NIO2S
    0021  CABS
    0022  ATAN2
    0023  ALOG
    0024  SQRT
    0025  CDVS
    0026  NERR35

    STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000  000023 1F      0001  000205 11L     0001  000211 12L     0001  000101 167G    0001  000306 20L
    0001  000322 21L     0001  000151 210G    0001  000160 214G    0000  000212 22F     0001  000220 227G
    0001  000343 23L     0000  000041 3F      0001  000357 30L     0000  000221 38F     0001  000450 39L
    0000  000055 4F      0000  000251 40F     0001  000457 41L     0000  000107 5F      0000  000124 6F
    0000  000072 7F      0011 R 000000 AMP     0003 L 000071 BOTH    0004  000064 CARD    0000 C 000002 CN
    0000  000000 CO      0000 C 000006 DCO     0007 L 000022 DEBUG   0006  000025 DEG     0011 R 000236 DEL
    0012  003734 DELTA   0006  000012 D0      0006  000013 D1      0006  000014 D2      0006  000015 D3
    0006  000016 D4      0005 C 000226 EA      0011 R 000233 EI      0012  001754 EIP     0000 R 000015 EI1
    0011 R 000232 ER      0012  000004 ERP     0000 R 000014 ERI     0011 R 000235 FDAMP   0006 C 000010 FIFTEN
    0006  000017 FIFTY    0011 R 000234 FN      0006 C 000006 FOUR    0000 C 000010 FSF     0000 C 000012 FSP
    0012 R 000002 GAINV   0003 L 000073 GPRINT  0006 C 000000 HALF    0000  000306 INJP$    0007  000010 IOPEN
    0003  000067 ITHZT    0007  000011 JOPEN    0005 I 000572 KD      0007  000001 KODE    0011 I 000230 L
    0004  000110 LABEL    0007 L 000023 LFLT    0010 L 000000 LGAIN   0011 I 000113 LOC     0010 L 000001 LPHASE
    0010 L 000002 LPOLES   0010 L 000003 LSORL   0010 L 000004 LZEROS  0000 I 000016 M     0003 L 000072 MODIFY
    0011 I 000227 M1      0011 I 000231 N       0005  000455 NA      0004 R 000062 NAME    0000 I 000022 NCN
    0000 I 000017 NCO     0005 I 000454 NEIG    0011 I 000226 NEXO    0012  003735 NGSYM    0005 I 000457 NI
    0007 L 000027 NORMAL  0007 L 000030 NOTYET  0012  003724 NPG     0012  003725 NPP     0012  003736 NPSYM
    0005  000456 NP      0007  000002 NRCLPL    0007  000003 NRPCLE   0007  000004 NRZERO   0000 I 000021 NSF
```

```
 0012  000726 NSHIFT     0000 I 000020 NSP         0007   000005 NXB         0007   000006 NXN         0007   000007 NXR
 0003  000004 NZT       0006   000026 NI          0006   000027 N2          0006   000030 N3          0006 C 000002 ONE
 0007  000014 PCPL      0007   000015 PFAC        0012 C 000000 PHAVAR      0006   000020 PI          0006 R 000021 PI2
 0007  000013 PNOM      0007   000016 PSLOSH      0007   000012 PVAR        0006   000024 RADDEG      0005 C 000000 ROOT
 0006  000023 RPI       0005 R 000000 RPIP        0012 R 000003 SHIFT       0006   000022 SMALL       0007   000000 STAGE
 0003  000000 SUPERK    0003   000003 TD          0011 R 000240 TEMP        0011 R 000237 THETA       0004 R 000000 TITLE
 0004  000029 TITLE1    0004 R 000036 TITLE2      0004   000050 TITLE3      0006 C 000004 TWO         0012   003727 XR
 0012  003730 YB        0007 L 000017 YESMTX      0007 L 000024 YESPCH      0007 L 000020 YESRAW      0007 L 000025 YESRLP
 0007 L 000026 YESSRL    0007 L 000021 YESSRP      0003 L 000070 YESZOH      0000 C 000004 Z           0003   000002 ZM
 0003 R 000001 ZT        0003   000005 ZTVAL
```

```
00101     1*          SUBROUTINE RLPRNT                                                    8174₀  000000
00102     2*          COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,       8175₀  000000
00103     3*         1          MODIFY,GPRINT                                             8176₀  000000
00104     4*          LOGICAL         YESZOH,BOTH,MODIFY,GPRINT                          8178₀  000000
00105     5*          COMMON/KEEP6/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),  8179₀  000000
00106     6*         1          CARD(20),LABEL(20)                                        8180₀  000000
00106     7*          REAL            NAME                                               8181₀  000000
00107     8*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)             8182₀  000000
00110     9*          COMPLEX         ROOT,EA                                                   000000
00111    10*          COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,    8185₀  000000
00112    11*         1          PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                         8186₀  000000
00112    12*          COMPLEX         HALF,ONE,TWO,FOUR,FIFTEN                                  000000
00113    13*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXD,NXN,NXR,         8189₀  000000
00113    14*         1          IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                   8190₀  000000
00113    15*         2          YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,            8191₀  000000
00113    16*         3          YESSRL,NOMNAL,NOTYET                                      8192₀  000000
00114    17*          LOGICAL         YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,     8193₀  000000
00114    18*         1          YESSRL,NOMNAL,NOTYET                                      8194₀  000000
00115    19*          COMMON/KEEP19/LGAIN,LPHASE,LPOLES,LSDRL,LZEROS                            000000
00115    20*          LOGICAL         LGAIN,LPHASE,LPOLES,LSDRL,LZEROS                          000000
00117    21*          COMMON/CRUD1/ AMP(75),LOC(75),NEND,M1,L,N,ER,EI,FN,FDAMP,DEL,      8195₀  000000
00117    22*         1          THETA,TEMP                                                8196₀  000000
00121    23*          COMMON/CRUD2/ PHAVAR,GAINV,SHIFT,ERP(1000),EIP(1000),NPG,NPP,      8197₀  000000
00121    24*         1          NSHIFT,XR,YB(4),DELTA,NGSYM,NPSYM                         8198₀  000000
00121    25*          COMPLEX         PHAVAR                                             8199₀  000000
00122    26*          DIMENSION       RPIP(150)                                                 000000
00123    27*          EQUIVALENCE     (RPIP(1),ROOT(1))                                  8201₀  000000
00124    28*          COMPLEX         CO,CN                                              8202₀  000000
00125    29*          COMPLEX         Z,DCD,FSF,FSP                                             000000
00125    30*      C                                                                      8204₀  000000
00125    31*      C   PRINT SAMPLED DATA ROOT LOCUS RESULTS                               8205₀  000000
00125    32*      C                                                                      8206₀  000000
00125    33*      C                                                                      8207₀  000000
00125    34*      C   PRINT TITLES                                                       8208₀  000000
00125    35*      C                                                                      8209₀  000000
00126    36*              WRITE(6,1) TITLE,TITLE1,TITLE2                                 8210₀  000000
00133    37*         1  FORMAT('1',19X,'S A M P L E D  D A T A  R O O T  L O C U S'//     8211₀  000015
00133    38*         1        20X,20A4//20X,20A4)                                       8212₀  000015
00133    39*      C                                                                      8213₀  000015
00133    40*      C                                                                      8214₀  000015
00133    41*      C   PRINT SAMPLING PERIOD                                              8215₀  000015
00133    42*      C                                                                      8216₀  000015
00134    43*              IF (NOMNAL) WRITE(6,3) ZT                                      8217₀  000015
```

```
                3 FORMAT(//20X,'NOMINAL CASE FOR SAMPLING PERIOD OF ',F5.2,2X,      8218₀  000025
     *             'SECONDS')                                                       8219₀  000025
C       PRINT GAIN VALUE AND SAMPLING PERIOD                                       8220₀  000025
C                                                                                  8221₀  000025
            IF (LGAIN .AND. .NOT.NOMNAL .AND. .NOT.LZEROS)                                000025
     *       WRITE(6,4) GAINV,ZT                                                          000025
          4 FORMAT(//20X,'GAIN = ',1PE12.5,10X,'SAMPLING PERIOD OF ',0PF5.2,       8223₀  000044
     1       2X,'SECONDS')                                                        8224₀  000044
            IF (LZEROS) WRITE(6,7) ZT                                                     000044
          7 FORMAT(//20X,'GAIN = INFINITY',14X,'SAMPLING PERIOD OF ',0PF5.2,               000054
     1       2X,'SECONDS')                                                               000054
C                                                                                  8225₀  000054
C       PRINT PHASE SHIFT AND SAMPLING PERIOD                                      8226₀  000054
C                                                                                  8227₀  000054
            IF (LPHASE .AND. .NOT.NOMNAL) WRITE(6,5) SHIFT,ZT                             000054
          5 FORMAT(//20X,'PHASE = ',1PE12.5,10X,'SAMPLING PERIOD OF ',0PF5.2,      8229₀  000066
     *       2X,'SECONDS')                                                        8230₀  000066
C                                                                                  8231₀  000066
C       PRINT COLUMN HEADINGS                                                      8232₀  000066
C                                                                                  8233₀  000066
            WRITE(6,6)                                                                    000066
          6 FORMAT(3(/),41X,'E I G E N V A L U E S'//107X,'NATURAL',6X,            8235₀  000073
     1       'DAMPING'/2X,'ROOT',18X,'ITERATION',12X,'REAL',12X,                  8236₀  000073
     2       'IMAGINARY',29X,'DAMPING',5X,'FREQUENCY',4X,'FREQUENCY'/                     000073
     3       1X,'NUMBER',1X,'SEQUENCE',2X,'CODE',4X,'COUNT',14X,'PART',           8238₀  000073
     4       14X,'PART',13X,'AMPLITUDE',5X,'RATIO',8X,'(CPS)',8X,                 8239₀  000073
     5       '(CPS)'/)                                                            8240₀  000073
C                                                                                  8241₀  000073
C       ARRANGE EIGENVALUES IN ORDER OF DECREASING ABSOLUTE VALUE                  8242₀  000073
C                                                                                  8243₀  000073
            DEL = .0001                                                                   000073
            DO 10 N=1,NEIG                                                                000101
            ER1 = RPIP(2*N-1)                                                             000101
            EI1 = RPIP(2*N)                                                               000103
            IF (ABS(ER1) .LT. DEL) ER1=0.0                                                000105
            IF (ABS(EI1) .LT. DEL) EI1=0.0                                                000113
            ROOT(N) = CMPLX(ER1,EI1)                                                      000121
            LOC(N) = N                                                             8245₀  000124
            AMP(N) = CABS(ROOT(N))                                                        000126
         10 CONTINUE                                                                      000136
            IF (NEIG.EQ.1) GO TO 12                                                8251₀  000136
            NEND = NEIG - 1                                                        8252₀  000142
            DO 11 N=1,NEND                                                         8253₀  000145
            M1 = N + 1                                                             8254₀  000151
            DO 11 M=M1,NEIG                                                        8255₀  000154
            IF (AMP(M).LE.AMP(N)) GO TO 11                                         8256₀  000160
            CALL XCHG (LOC(M),LOC(N))                                              8257₀  000164
            CALL XCHG (AMP(M),AMP(N))                                              8258₀  000174
         11 CONTINUE                                                               8259₀  000211
         12 CONTINUE                                                               8260₀  000211
C                                                                                  8261₀  000211
C       PRINT OUTPUT DATA                                                          8262₀  000211
C                                                                                  8263₀  000211
            DO 30 N=1,NEIG                                                         8264₀  000211
            L = LOC(N)                                                             8265₀  000220
            ER = RPIP(2*L-1)                                                       8266₀  000223
            EI = RPIP(2*L)                                                         8267₀  000225
```

```
00234     101              IF (LPHASE) GO TO 23                                                  000227
00235     102*             FN = 0.0                                                    82690     000231
00236     103*             FDAMP = 0.0                                                 82700     000232
00237     104*             DEL = 1.0                                                   82710     000233
00238   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00238     105*             IF (ER.EQ.0.0 .AND. EI.EQ.0.0) GO TO 21                     82720     000235
00239     106*             THETA = ATAN2(EI,ER)                                        82730     000245
00240     107*             TEMP=ALOG(AMP(N))                                           82740     000252
00241     108*             DEL = 0.0                                                   82750     000260
00242   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00242     109*             IF (THETA.EQ.0.0 .AND. TEMP.EQ.0.0) GO TO 20                82760     000261
00243     110*             DEL=-TEMP/SQRT(THETA**2 + TEMP**2)                          82770     000271
00244     111*         20  CONTINUE                                                     82780     000306
00245     112*             FN=ABS(THETA)/(ZPI2)                                        82790     000306
00246     113*             FDAMP=FN*SQRT(1.0-DEL**2)                                   82800     000310
00247     114*         21  CONTINUE                                                     82810     000322
00248     115*             WRITE(6,22)  N,L,KD(L),NI(L),ER,EI,AMP(N),DEL,FN,FDAMP      82820     000322
00249     116*         22  FORMAT(1X,I4,3I4,1PE25.7,E18.7,0PF16.5,3F13.5)             82830     000341
00250     117*             GO TO 30                                                    82840     000341
00251     118*         23  CONTINUE                                                    82850     000343
00252     119*             WRITE(6,22) N,L,KD(L),NI(L),ER,EI                           82860     000343
00253     120*         30  CONTINUE                                                    82870     000360
00254     121*     C                                                                   82880     000360
00255     122*     C   CALCULATE AND PRINT LEADING AND LAST COEFFICIENTS                82890     000360
00256     123*             IF (NONNAL .AND. .NOT. LGAIN .AND. .NOT. LPHASE) RETURN                000360
00257     124*     C                                                                   82900     000360
00258     125*             Z = (0.,0.)                                                            000370
00259     126*             CALL DET(Z,DCO,NCO)                                                    000372
00260     127*             CO = DCO                                                    82920     000377
00261     128*             Z = (1.0,0.0)                                                          000401
00262     129*             CALL PEVAL(Z,FSP,NSP)                                       82940     000403
00263     130*             CALL DET(Z,FSF,NSF)                                         82950     000410
00264     131*             NC=NSF-NSP                                                  82960     000415
00265     132*             CN=FSF/FSP                                                  82970     000420
00266     133*             IF((NCP.EQ.0) .AND. (NCN.EQ.0)) GO TO 39                    82980     000425
00267     134*             WRITE(6,38) CN,NCN,CO,NCO                                   82990     000435
00268     135*        38   FORMAT(2(/),20X,'CHARACTERISTIC'/20X,'POLYNOMIAL'/20X,      83000     000446
00269     136*            1'LEADING COEFFICIENT',1PE15.7,E18.7,5X,I5//20X,             83010     000446
00270     137*            2'LAST COEFFICIENT',1PE15.7,E18.7,5X,I5)                     83020     000446
00271     138*             GO TO 41                                                    83030     000446
00272     139*        39   CONTINUE                                                    83040     000450
00273     140*             WRITE (6,40) CN,CO                                          83050     000450
00274     141*        40   FORMAT (2(/),20X,'CHARACTERISTIC'/20X,'POLYNOMIAL'/20X,'LEADING CO  83060     000457
00275     142*            1EFFICIENT'1PE15.7,E18.7//20X,'LAST COEFFICIENT'2E18.7)       83070     000457
00276     143*        41   CONTINUE                                                    83080     000457
00277     144*             RETURN                                                      83090     000457
00278     145*             END                                                         83100     000501

      END OF COMPILATION:  2 DIAGNOSTICS.
```

```
*FOR,US* F.ROUTER,F.ROOTER
FOR UE1x-05/23/74-08:47:11 (2,3)

      SUBROUTINE ROOTER     ENTRY POINT 000322

      STORAGE USED: CODE(1) 000326; DATA(0) 000062; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP1  000026
      0004   KEEP9  000705
      0005   KEEP16 000031
      0006   CRUD3  016115

      EXTERNAL REFERENCES (BLOCK, NAME)

      0007   ESTMAT
      0010   START
      0011   DECIDE
      0012   AGAIN
      0013   COABV
      0014   SOLVE
      0015   SAVE
      0016   NADUS
      0017   NIOZS
      0020   NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001  000006 10L     0001  000016 20L     0001  000152 222G    0001  000046 25L     0001  000106 30L
      0001  000122 40L     0001  000134 50L     0001  000150 60L     0000  000004 600F    0000  000022 601F
      0000  000027 602F    0000  000041 603F    0001  000166 61L     0001  000176 62L     0001  000203 70L
      0001  000253 75L     0001  000211 80L     0001  000304 90L     0006 L 016113 AUTO    0006 C 016072 B
      0013 R 000000 COABV   0006 L 016112 CONJ    0006 C 000000 CU      0005 L 000022 DEBUG   0006 L 016110 DONE
      0006 R 016074 E       0004 C 000226 EA      0000 R 000001 EI      0006 C 016040 FPR0    0006 C 016042 FPR1
      0006 C 016044 FPR2    0006 C 016046 FPR     0006 C 016050 FR1     0006 C 016052 FR2     0000 I 000003 I
      0000  000054 INJP$    0005  000010 IOPEN    0005  000011 JOPEN    0004  000572 KD      0005  000001 KODE
      0000 L 000000 LETGO   0005 L 000023 LFLT    0003 I 000000 MAXIT   0003  000001 MXEIG   0003  000024 MXEIGT
      0003  000023 MXEST    0003  000002 MXFRM    0003  000003 MXNBM   0003  000004 MXNCOF   0003  000005 MXNCT
      0003  000025 MXNCV    0003  000006 MXNE     0003  000007 MXNEQ   0003  000010 MXNFI    0003  000011 MXNG
      0003  000012 MXNPH    0003  000013 MXNPP    0003  000014 MXNQPT  0003  000015 MXNSM    0003  000016 MXNSP
      0003  000017 MXNTM    0003  000020 MXNV     0003  000021 MXNZT   0003  000022 MXPOLY   0004  000455 NA
      0006 I 016102 NCT     0004 I 000454 NEIG    0006 I 016107 NEST    0006  016076 NFP0    0006  016077 NFP1
      0006  016100 NFP2    0004  000457 NI      0006 I 016101 NITER   0006 I 016104 NKODE   0005 L 000027 NOMINAL
      0005 L 000030 NOTYET  0004 I 000456 NR      0005  000002 NRCLPL  0006 I 016103 NREG    0005  000003 NRPOLE
      0005  000004 NRZERO   0006 I 016105 NSTART  0004 I 016106 NTIME   0000 I 000002 NTR     0005  000005 NX8
      0005  000006 NXN      0005  000007 NXR     0005  000014 PCPL    0005  000015 PFAC    0005  000013 PNOM
      0006 C 016054 PR0     0006 C 016056 PR1     0006 C 016060 PR2     0005  000016 PSLOSH  0005  000012 PVAR
      0006 L 016114 PFGSEL  0006 L 016111 RESTRT  0000 C 000000 ROOT    0006 C 016062 R0      0006 C 016064 R1
      0006 C 016066 R2      0006 C 016066 R2R1    0006 C 016070 R3      0006 R 016070 R3R1    0005  000000 STAGE
      0006 C 016074 U       0005 L 000017 YESNTX  0005 L 000024 YESPCH   0005 L 000020 YESRAW   0005 L 000025 YESRLP
      0005 L 000026 YESSRL  0005 L 000021 YESSRP

801-9
```

```
00101     1          SUBROUTINE ROOTER                                          8495₀  000000
00102     2*         COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,  8496₀  000000
00103     3*             1         MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,  8497₀  000000
00104     4*             2         MXNV,MXNZT,MXPOLY,MXREST,MXEIGT,MXNCV          8498₀  000000
00105     5*         COMMON/KEEP9/ ROOT(75),EA(75),NEIGHA,NR,NI(75),KD(75)        8499₀  000000
00106     6*         COMPLEX       ROOT,EA                                                 000000
00106     7*         COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,     8502₀  000000
00106     8*             1         IOPEN,JOPEN,PVAR,PNON,PCPL,PFAC,PSLOSH,        8503₀  000000
00106     9*             2         YESMTX,YESRAW,YESRP,DEBUG,LFLT,YESPCH,YESRLP,  8504₀  000000
00107    10*             3         YESSRL,NOMNAL,NOTYET                           8505₀  000000
00107    11*         LOGICAL       YESMTX,YESRAW,YESRP,DEBUG,LFLT,YESPCH,YESRLP,  8506₀  000000
00107    12*             1         YESSRL,NOMNAL,NOTYET                           8507₀  000000
00110    13*         COMMON/CRUD3/ CU(60,60),FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,  8508₀  000000
00111    14*             1         R0,R1,R2,R3,U,NFP0,NFP1,NFP2,NITER,NCT,NREG,NKODE,  8509₀  000000
00111    15*             2         NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL  8510₀  000000
00111    16*         COMPLEX       CU,FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,             000000
00111    17*             1         R0,R1,R2,R3,U,U                                8512₀  000000
00112    18*         LOGICAL       DONE,RESTRT,CONJ,AUTO,REGSEL                   8513₀  000000
00113    19*         LOGICAL       LETGO                                          8514₀  000000
00114    20*         DIMENSION     F(2)                                                   000000
00115    21*         DIMENSION     R2R1(2),R3R1(2)                                        000000
00116    22*         EQUIVALENCE   (R2,R2R1(1)),(R3,R3R1(1))                              000000
00117    23*         EQUIVALENCE   (F(1),U)                                       8516₀  000000
00117    24*     C                                                                8518₀  000000
00117    25*     C                                                                8519₀  000000
00117    26*     C*****       INITIALIZE FOR ROOTING ROUTINE                      8520₀  000000
00117    27*     C                                                                8521₀  000000
00120    28*           E1 = 1.E-4                                                         000000
00121    29*           NREG=0                                                     8523₀  000001
00122    30*           NEST=0                                                     8524₀  000002
00123    31*           NEIG = 0                                                   8525₀  000003
00124    32*           CONJ=.FALSE.                                               8526₀  000004
00124    33*     C                                                                8527₀  000004
00124    34*     C                                                                8528₀  000004
00124    35*     C*****       INITIALIZE FOR NEXT ROOT                            8529₀  000004
00124    36*     C                                                                8530₀  000004
00125    37*    10 CONTINUE                                                       8531₀  000006
00126    38*           NCT=0                                                      8532₀  000006
00127    39*           NTR=0                                                      8533₀  000006
00130    40*           NITER=0                                                    8534₀  000007
00131    41*           NKODE=0                                                    8535₀  000010
00132    42*           NSTART=0                                                   8536₀  000011
00133    43*           NTIME=0                                                    8537₀  000012
00134    44*           REGSEL=.FALSE.                                             8538₀  000013
00135    45*           RESTRT=.FALSE.                                             8539₀  000014
00135    46*     C                                                                8540₀  000014
00135    47*     C                                                                8541₀  000014
00135    48*     C*****       ENTRY POINT FOR A NEW START                         8542₀  000014
00135    49*     C                                                                8543₀  000014
00136    50*        20 CONTINUE                                                   8544₀  000016
00137    51*           LETGO=.FALSE.                                              8545₀  000016
00140    52*           IF(RESTRT .AND. NCT.EQ.0) LETGO=.TRUE.                     8546₀  000016
00142    53*           NCT=0                                                      8547₀  000026
00143    54*           NTR=NTR+NITER                                              8548₀  000027
00144    55*           NITER=1                                                    8549₀  000032
00145    56*           NSTART=NSTART+1                                            8550₀  000034
```

```
00184*      IF(NSTART.GT.50) GO TO 90                                  85510  000087
00184*    C                                                            85520  000087
00184*    C                                                            85530  000087
00184*    C*****     MAKE AN ESTIMATE OF A ROOT                        85540  000087
00184*    C                                                            85550  000087
00185*        CALL ESTMAT                                              85560  000043
00185*     25 CONTINUE                                                 85570  000046
00185*        IF (ABS(E(2)) .LT. 1.E-9) E(2)=0.0                                000046
00185*        IF(DEBUG) WRITE(6,600) B,U,NSTART,DONE                   85590  000053
00186*    600 FORMAT(' ESTMAT',5X,'B ',2E14.5,5X,'U ',2E14.5,5X,'NSTART',I3,   000066
00186*       15X,'DONE',L3/)                                                 000066
00186*        IF(DONE) RETURN                                          85620  000066
00186*    C                                                            85630  000066
00186*    C                                                            85640  000066
00186*    C*****     SELECT AND EVALUATE FIRST THREE ITERANTS           85650  000066
00186*    C                                                            85660  000066
00186*        CALL START                                               85670  000073
00186*    C                                                            85680  000073
00186*    C                                                            85690  000073
00186*    C*****     DETERMINE WHETHER TO FIND ANOTHER ROOT            85700  000073
00186*    C                                                            85710  000073
00186*        CALL DECIDE                                              85720  000075
00187*        IF(DONE) RETURN                                          85730  000077
00187*        GO TO 40                                                 85740  000104
00187*    C                                                            85750  000104
00187*    C                                                            85760  000104
00187*    C*****     SOLVE DETERMINATE AND POLYNOMIAL FOR NEW ITERANT  85770  000104
00187*    C                                                            85780  000104
00187*     30 CONTINUE                                                 85790  000106
00187*        CALL AGAIN                                               85800  000106
00187*        IF(DEBUG) WRITE(6,601) RESTRT                            85810  000107
00188*    601 FORMAT(' ',12X,'RESTRT',L4/)                             85820  000117
00188*        IF(RESTRT) GO TO 20                                      85830  000117
00188*    C                                                            85840  000117
00188*    C                                                            85850  000117
00188*    C*****     TEST FUNCTIONAL VALUE OF ITERANT FOR ZERO         85860  000117
00188*    C                                                            85870  000117
00188*     40 CONTINUE                                                 85880  000122
*DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00189*        IF (CDABS(EPR2).NE.0.0) GO TO 50                                  000122
00189*    C                                                            85900  000122
00189*    C                                                            85910  000122
00189*    C*****     ROOT FOUND                                        85920  000122
00189*    C                                                            85930  000122
00190*        NROOT = 1                                                85940  000126
00191*        R3=R2                                                    85950  000130
00191*        GO TO 80                                                 85960  000132
00191*    C                                                            85970  000132
00191*    C                                                            85980  000132
00191*    C*****     CALCULATE NEW ITERANT                             85990  000132
00191*    C                                                            86000  000132
00192*     50 CONTINUE                                                 86010  000134
00192*        CALL SOLVE                                               86020  000134
00193*    C                                                            86030  000134
00193*    C                                                            86040  000134
00193*    C*****     TEST NEW ITERANT FOR CONVERGENCE TO ZERO           86050  000134
00193*    C                                                            86060  000134
```

```
00214   113*           IF (CDABV(R3) .GT. E1) GO TO 60                          000135
00214   114*     C                                                     8608₀     000135
00214   115*     C                                                     8609₀     000135
00214   116*     C*****     ROOT FOUND                                 8610₀     000135
00214   117*     C                                                     8611₀     000135
00215   118*           NKODE = 2                                       8612₀     000144
00217   119*           GO TO 80                                        8613₀     000146
00217   120*     C                                                     8614₀     000146
00217   121*     C                                                     8615₀     000146
00217   122*     C*****     TEST FOR CONVERGENCE OF LAST TWO ITERANTS   8616₀     000146
00217   123*     C                                                     8617₀     000146
00221   124*      60 CONTINUE                                          8618₀     000152
00221   125*           DO 62 I=1,2                                               000152
00224   126*           IF (ABS(R3RI(I)) .GT. E1) GO TO 61                        000152
00226   127*           IF (ABS(R2RI(I)) .GT. E1) GO TO 70                        000157
00231   128*           GO TO 62                                                  000164
00231   129*        61 IF (ABS((R3RI(I) - R2RI(I))/R3RI(I)) .GT. E1) GO TO 70    000166
00233   130*        62 CONTINUE                                                  000177
00233   131*     C                                                     8620₀     000177
00233   132*     C                                                     8621₀     000177
00233   133*     C*****     ROOT FOUND                                 8622₀     000177
00233   134*     C                                                     8623₀     000177
00235   135*           NKODE = 3                                       8624₀     000177
00235   136*           GO TO 80                                        8625₀     000201
00235   137*     C                                                     8626₀     000201
00235   138*     C                                                     8627₀     000201
00235   139*     C*****     TEST ITERATIONS                            8628₀     000201
00235   140*     C                                                     8629₀     000201
00237   141*        70 CONTINUE                                        8630₀     000203
00243   142*           IF(NITER.LT.MAXIT) GO TO 30                     8631₀     000203
00242   143*           GO TO 20                                        8632₀     000207
00242   144*     C                                                     8633₀     000207
00242   145*     C                                                     8634₀     000207
00242   146*     C*****     SAVE ROOT                                  8635₀     000207
00242   147*     C                                                     8636₀     000207
00243   149*        80 CONTINUE                                        8637₀     000211
00244   149*           IF(NITER.NE.1 .OR. LETGO) GO TO 75              8638₀     000211
00245   150*           LETGO=.TRUE.                                    8639₀     000220
00247   151*           B = 0.01*B                                                000222
00250   152*           U=R3                                            8641₀     000230
00251   153*           NCT=0                                           8642₀     000232
00252   154*           NTR=NTR+NITER                                   8643₀     000233
00253   155*           NITER=1                                         8644₀     000236
00254   156*           NKODE = 0                                       8645₀     000240
00255   157*           DONE=.FALSE.                                    8646₀     000241
00255   158*           IF(DEBUG) WRITE(6,602)                          8647₀     000242
00261   159*        602 FORMAT('0 ROOTER',5X,'ACCURATE ESTIMATE CAUSE OF RESTART')  8648₀  000251
00262   160*           GO TO 25                                        8649₀     000251
00263   161*        75 CONTINUE                                        8650₀     000253
00264   162*           IF((REGSFL).AND.(NITER.GT.251)) NREG=NREG+1     8651₀     000253
00265   163*           NITER=NITER+NTR                                 8652₀     000266
00267   164*           CALL SAVE                                       8653₀     000271
00267   165*     C                                                     8654₀     000271
00267   166*     C                                                     8655₀     000271
00267   167*     C*****     TEST IF ALL ROOTS FOUND                    8656₀     000271
00267   168*     C                                                     8657₀     000271
00271   169*           IF (NEIG.LT.NR) GO TO 10                        8658₀     000273
```

```
00272   170*          RETURN                                                  8659D    000300
00272   171*      C                                                           8660D    000300
00272   172*      C                                                           8661D    000300
00272   173*      C*****      CAN NOT CONVERGE AFTER 10 RESTARTS              8662D    000300
00272   174*      C                                                           8663D    000300
00273   175*           90 CONTINUE                                            8664D    000304
00274   176*              WRITE(6,603)                                        8665D    000304
00275   177*          603 FORMAT(' I CAN NOT CONVERGE AFTER 10 RESTARTS')     8666D    000310
00277   178*              RETURN                                              8667D    000310
00300   179*              END                                                 8668D    000325

END OF COMPILATION:      1 DIAGNOSTICS.
```

```
@FOR,USW F.RUTER,F.RUTER
FOR SEIX-05/23/74-08:47:27 (3,4)

    SUBROUTINE RUTER      ENTRY POINT 000152

    STORAGE USED: CODE(1) 000162; DATA(0) 000027; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003  KEEP1   000026
    0004  KEEP2   000047
    0005  KEEP4   000263
    0006  KEEP9   000705
    0007  KEEP10  021620
    0010  KEEP16  000031
    0011  KEEP19  000005
    0012  CRUD2   003737

    EXTERNAL REFERENCES (BLOCK, NAME)

    0013  ROOTER
    0014  RLPRNT
    0015  NWDU$
    0016  NIO2$
    0017  NERR4$
    0020  NERR3$

    STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000044 L36G      0001 000074 L52G      0001 000130 2000L     0001 000067 40L       0001 000117 60L
0000 000001 602F      0005 000212 BY        0005 000166 DD        0010 L 000022 DEBUG   0012 003734 DELTA
0004 D 000002 DHFILT  0006 C 000226 EA      0012 R 001754 EIP     0012 R 000004 ERP     0007 016664 EV
0005 000004 GAIN      0012 000002 GAINV     0005 000151 GSYM      0004 000037 HACC      0004 000025 HAST
0004 000035 HATT      0004 D 000000 HBLANK  0004 R 000004 HBLK    0004 000034 HCCW      0004 000033 HCW
0004 000032 HDEC      0004 000027 HDOT      0004 000006 HESTI     0004 000043 HFGD      0004 000044 HFGN
0004 000040 HFPD      0004 000045 HFPDD     0004 000042 HFPDN     0004 000041 HFPN      0004 000010 HGENE
0004 000031 HINC      0004 000005 HKEY      0004 000007 HMATR     0004 000014 HNEW      0004 000046 HNOM1
0004 000015 HNYQU     0004 000023 HO        0004 000026 HPLUS     0004 000036 HRATE     0004 000011 HRAN
0004 000012 HRETA     0004 000017 HROLL     0004 000016 HROOT     0004 000013 HSTAN     0004 000030 HSTAR
0004 000020 HSIC      0004 000021 HS2       0004 000022 HS4B      0004 000024 HX        0000 I 000000 I
0000 000016 INJP$     0010 000010 IOPEN     0007 000004 IR        0007 001754 JC        0010 000011 JOPEN
0006 000572 KD        0010 I 000001 KODE    0010 L 000023 LELT    0011 L 000000 LGAIN   0007 005674 LL
0007 007644 LOCPOL    0011 L 000001 LPHASE  0011 L 000002 LPOLES  0011 L 000003 LSDRL   0011 L 000004 LZEROS
0003 000000 MAXIT     0003 I 000001 MXEIG   0003 000024 MXEIGT    0003 000023 MXEST     0003 000002 MXFRM
0003 000003 MXGAM     0003 000004 MXNCOF    0003 000005 MXNCT     0003 000025 MXNCV     0003 000006 MXNE
0003 000007 MXNEQ     0003 000010 MXNFI     0003 000011 MXNG      0003 000012 MXNPH     0003 I 000013 MXNPP
0003 000014 MXNOPT    0003 000015 MXNSM     0003 000016 MXNSP     0003 000017 MXNTM     0003 000020 MXNV
0003 000021 MXNZI     0003 000022 MXPOLY    0006 000456 NA        0007 000003 NCOF      0007 003724 ND
0007 000000 NDEG      0007 000002 NE        0006 I 000454 NEIG    0007 000001 NEQ       0005 000003 NGAIN
0005 000154 NGR       0012 003735 NGSYM     0004 000457 NI        0010 L 000027 NOMNAL  0010 L 000030 NOTYET
0012 I 003724 NPG     0005 000066 NPHASE    0012 I 003725 NPP     0012 003736 NPSYM     0006 I 000456 NR
0010 000002 NRCLPL    0005 000153 NRLFR     0010 000003 NRPOLE    0010 000004 NRZERO    0012 003726 NSHIFT
0010 000005 NXB       0010 000006 NXN       0010 000007 NXR       0010 000014 PCPL      0010 000015 PFAC
```

```
0005  000000 PG      0005  000067 PHASE     0012 C 000000 PHAVAR    0010  000013 PNOM      0005  000001 PP
0005 R 000002 PPLT    0010  000016 PSLOSH    0005  000152 PSYM      0010  000012 PVAR      0006 C 000000 ROOT
0006 R 000000 RPIP    0005  000200 RX        0012  000003 SHIFT     0010  000000 STAGE     0012  003727 XR
0012  003730 YB      0010 L 000017 YESMTX    0010 L 000024 YESPCH    0010 L 000020 YESRAW    0005 L 000262 YESRL
0010 L 000025 YESRLP  0010 L 000026 YESSRL    0010 L 000021 YESSRP
```

```
00101     1*          SUBROUTINE RUTER(*)                                          8669  000000
00102     2*          COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,  8670  000000
00103     3*         1            MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,    8671  000000
00104     4*         2            MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV                8672  000000
00105     5*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,    8673  000000
00106     6*         1            HRETA,HSTAN,HNEW,HNYOU,HROOT,HROLL,HS1C,HS2,HS4B,    8674  000000
00107     7*         2            HQ,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,      8675  000000
00108     8*         3            HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,     8676  000000
00109     9*         4            HNOHI                                                8677  000000
00110    10*          DOUBLE PRECISION HBLANK,DHFILT                                          000000
00111    11*          COMMON/KEEP4/ PG,PP,PPLT,NGAIN,GAIN(50),NPHASE,PHASE(50),GSYM,  8679  000000
00112    12*         1            PSYM,NRLFR,NGR(10),DD(10),RX(10),BY(4,10),YESRL      8680  000000
00113    13*          LOGICAL      YESRL                                                8681  000000
00114    14*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)            8682  000000
00115    15*          COMPLEX      ROOT,EA                                                 000000
00116    16*          COMMON/KEEP10/NDEG,NEQ,NE,NCOF,IR(1000),JC(1000),ND(1000),      8685  000000
00117    17*         1            LL(1000),LOCPOL(60,60),EV(1500)                      8686  000000
00118    18*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,      8687  000000
00119    19*         1            IOPEN,JOPEN,PVAR,PNOM,PCEL,PEAC,PSLOSH,              8688  000000
00120    20*         2            YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,       8689  000000
00121    21*         3            YESSRL,NOMNAL,NOTYET                                 8690  000000
00122    22*          LOGICAL      YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,       8691  000000
00123    23*         1            YESSRL,NOMNAL,NOTYET                                 8692  000000
00124    24*          COMMON/KEEP19/LGAIN,LPHASE,LPOLES,LSDRL,LZEROS                          000000
00125    25*          LOGICAL      LGAIN,LPHASE,LPOLES,LSDRL,LZEROS                        000000
00126    26*          COMMON/CRUD2/ PHAVAR,GAINV,SHIFT,ERP(1000),EIP(1000),NPG,NPP,   8693  000000
00127    27*         1            NSHIFT,XR,YB(4),DELTA,NGSYM,NPSYM                    8694  000000
00128    28*          COMPLEX      PHAVAR                                               8695  000000
00129    29*          DIMENSION    RPIP(150)                                               000000
00130    30*          EQUIVALENCE  (RPIP(1),ROOT(1))                                    8697  000000
00131    31*     C                                                                      8718  000000
00132    32*     C                                                                      8719  000000
00133    33*     C  CALCULATE THE EIGENVALUES                                           8720  000000
00134    34*     C                                                                      8721  000000
00135    35*          IF (NR .GT. MXEIG) NR = MXEIG                                           000000
00136    36*          CALL ROOTER                                                             000006
00137    37*          IF (NEIG.EQ.0) GO TO 2000                                         8723  000010
00138    38*     C                                                                      8724  000010
00139    39*     C                                                                      8725  000010
00140    40*     C  PRINT THE ROOT LOCUS RESULTS                                        8726  000010
00141    41*     C                                                                      8727  000010
00142    42*          CALL RLPRNT                                                       8728  000012
00143    43*     C                                                                      8729  000012
00144    44*     C                                                                      8730  000012
00145    45*     C                                                                      8731  000012
00146    46*     C  SAVE ROOTS FOR PLOTTING                                             8732  000012
00147    47*     C                                                                      8733  000012
00148   **,AGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
```

```
00131     48*           IF (PPLT.EQ.PBLK .OR. NPP=NPG.EQ.1) RETURN                    87340   000014
00132     49*           IF (LPHASE) GO TO 40                                                  000034
00133     50*     C                                                                   87360   000034
00134     51*     C  GAIN ROOT LOCUS                                                  87370   000034
00135     52*     C                                                                   87360   000034
00136     53*           DO 30 I=1,NEIG                                                87390   000044
00140     54*             IF (NPP=NPG .EQ. 1) GO TO 60                                87400   000044
00142     55*             NPG = NPG + 1                                               87410   000051
00143     56*           ERP(NPG) = RPIP(2*I-1)                                        87420   000055
00144     57*           EIP(NPG) = RPIP(2*I)                                          87430   000057
00145     58*        30 CONTINUE                                                      87440   000063
00147     59*           RETURN                                                        87450   000063
00147     60*     C                                                                   87460   000063
00147     61*     C  PHASE ROOT LOCUS                                                 87470   000063
00147     62*     C                                                                   87480   000063
00150     63*        40 CONTINUE                                                      87490   000067
00151     64*           DO 50 I=1,NEIG                                                87500   000067
00154     65*           IF (NPP=NPG .EQ. 1) GO TO 60                                  87510   000074
00156     66*           NPP = NPP + 1                                                 87520   000101
00157     67*           ERP(NPP) = RPIP(2*I-1)                                        87530   000105
00160     68*           EIP(NPP) = RPIP(2*I)                                          87540   000107
00161     69*        50 CONTINUE                                                      87550   000113
00163     70*           RETURN                                                        87560   000113
00163     71*     C                                                                   87570   000113
00163     72*     C                                                                   87580   000113
00163     73*     C  TOO MANY PLOT-POINTS                                             87590   000113
00163     74*     C                                                                   87600   000113
00164     75*        60 CONTINUE                                                      87610   000117
00165     76*           WRITE(6,602) NXNPP                                            87620   000117
00170     77*       602 FORMAT(////' ONLY FIRST ',I4,' ROOTS SAVED FOR ROOT LOCUS PLOTS')  87630   000124
00171     78*           RETURN                                                        87640   000124
00171     79*     C                                                                   87710   000124
00171     80*     C                                                                   87720   000124
00171     81*     C  NO ROOT COMPUTED                                                 87730   000124
00171     82*     C                                                                   87740   000124
00172     83*     2000 CONTINUE                                                       87750   000130
00173     84*           KODE = 56                                                     87760   000130
00174     85*           RETURN 1                                                      87770   000131
00175     86*           END                                                           87850   000161

END OF COMPILATION:  0 DIAGNOSTICS.
```

```
?FOR,USR F,SAVE,F,SAVE
FOR SEIX-05/23/74-08:47:36 (1,2)

    SUBROUTINE SAVE       ENTRY POINT 000134

    STORAGE USED: CODE(1) 000137; DATA(0) 000047; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003  KEEP1  000026
    0004  KEEP2  000705
    0005  KEEP16 000031
    0006  CRUD3  016115

    EXTERNAL REFERENCES (BLOCK, NAME)

    0007  NRDU$
    0010  NI02$
    0011  NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001  000062 IOL      0001  000117 ZOL      0000  000002 600F     0006 R 016070 AA       0006 L 016113 AUTO
    0006 C 016072 B       0006 L 016112 CONJ     0006 C 000000 CU     0005 L 000022 DEBUG    0006 L 016110 DONE
    0004 C 000226 EA      0006 C 016040 FPR0     0006 C 016042 FPR1   0006 C 016044 FPR2     0006 C 016046 FR0
    0006 C 016050 FR1     0006 C 016052 FR2      0000  000092 INJPS    0005  000010 IOPEN     0005  000011 JOPEN
    0004 I 000572 KD      0005  000001 KODE     0005 L 000023 LFLT     0003 I 000000 MAXIT    0003 I 000001 MXEIG
    0003  000024 MXEIGT   0003  000023 MXEST    0003  000002 MXFRM     0003  000003 MXNBM    0003  000004 MXNCOF
    0003  000005 MXNCT    0003  000025 MXNCV    0003  000006 MXNE      0003  000007 MXNEQ    0003  000010 MXNFI
    0003  000011 MXNG     0003  000012 MXNPH    0003  000013 MXNPP     0003  000014 MXNQPT   0003  000015 MXNSM
    0003  000016 MXNSP    0003  000017 MXNTM    0003  000020 MXNV      0003  000021 MXNZT    0003  000022 MXPOLY
    0004  000455 NA       0006  016102 NCT      0004 I 000454 NEIG     0006 I 016107 NEST     0006  016076 NFPO
    0006  016077 NFPI     0006  016100 NFP2     0004 I 000457 NI       0006 I 016101 NITER    0006 I 016104 NKODE
    0005  000027 NORMAL   0005 L 000030 NOTYET   0004 I 000456 NR       0005  000002 NRCLPL   0006  016103 NREG
    0005  000003 NRPOLE   0005  000004 NRZERO    0006 I 016105 NSTART   0006 I 016106 NTIME    0005  000005 NXB
    0005  000006 NXN      0005  000007 NXR      0005  000014 PCPL      0005  000015 PEAC     0005  000013 PNOM
    0006 C 016054 PR0     0006 C 016056 PR1      0006 C 016060 PR2    0005  000016 PSLOSH   0005  000012 PVAR
    0006 L 016114 REGSEL  0006 L 016111 RESTRT   0000 R 000001 RIHAG   0004 C 000000 ROOT     0000 R 000000 RREAL
    0006 C 016062 R0      0006 C 016064 R1       0006 C 016066 R2     0006 C 016070 R3       0005  000000 STAGE
    0006 C 016074 U       0005 L 000017 YESMTX   0005 L 000024 YESPCH  0005 L 000020 YESRAW   0005 L 000025 YESRLP
    0005 L 000026 YESSRL  0005 L 000021 YESSRP

    00101   1*          SUBROUTINE SAVE                                                    92580  000000
    00103   2*          COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,     92590  000000
    00103   3*      1                 MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSH,MXNSP,MXNTM,   92600  000000
    00103   4*      2                 MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV               92610  000000
    00104   5*          COMMON/KEEP2/ NGOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)           92620  000000
    00105   6*          COMPLEX       ROOT,EA                                                  000000
    00106   7*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,       92650  000000
    00106   8*      1                 IOPEN,JOPEN,PVAR,PNOM,PCPL,PEAC,PSLOSH,          92660  000000
```

```
00106     9*          2               YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      92670  000000
00106    10*          3               YESSRL,NOMNAL,NOTYET                                92680  000000
00107    11*          LOGICAL         YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      92690  000000
00107    12*          1               YESSRL,NOMNAL,NOTYET                                92700  000000
00111    13*          COMMON/CRUD3/   CU(60,60),FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,   92710  000000
00111    14*          1               R0,R1,R2,R3,B,U,NFP0,NFP1,NFP2,NITER,NCT,NREG,NKODE,92720  000000
00113    15*          2               NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL      92730  000000
00115    16*          COMPLEX         CU,FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,                000000
00117    17*          1               R0,R1,R2,R3,B,U                                     92790  000000
00117    18*          LOGICAL         DONE,RESTRT,CONJ,AUTO,REGSEL                        92760  000000
00113    19*          DIMENSION       AA(2)                                                      000000
00119    20*          EQUIVALENCE     (R3,AA(1))                                                 000000
00119    21*     C                                                                        92790  000000
00119    22*     C                                                                        92800  000000
00119    23*     C  PROGRAM CODING                                                        92810  000000
00119    24*     C                                                                        92820  000000
00119    25*          RREAL=AA(1)                                                         92830  000000
00119    26*          RIMAG=AA(2)                                                         92840  000001
00117    27*          NEIG = NEIG + 1                                                     92850  000003
00120    28*          KD(NEIG) = NKODE                                                    92860  000006
00121    29*          NI(NEIG) = NITER                                                    92870  000011
00122    30*          IF (DEBUG) WRITE(6,600) NEIG,R3,NKODE,NITER,NSTART,NTIME,NEST,      92880  000013
00122    31*         *                MAXIT                                               92890  000013
00134    32*          600 FORMAT('0 SAVE',7X,'R-O-O-T     S-A-V E-D',13X,'NEIG',I4,5X,'R3 ',      000032
00135    33*         *12E14.5,5X,'NKODE',I3,5X,'NITER',I4,5X,'NSTART',I4/13X,'NTIME',I4,      000032
00135    34*         *25X,'NEST',I4,5X,'MAXIT',I4/)                                           000032
00136    35*          IF (ABS(RIMAG) .LT. 0.1) GO TO 20                                       000032
00141    36*          IF (ABS(RIMAG) .LE. 0.001*ABS(RREAL)) GO TO 20                          000041
00141    37*     C                                                                        92950  000041
00141    38*     C                                                                        92960  000041
00141    39*     C*****       SAVE COMPLEX ROOT                                           92970  000041
00141    40*     C                                                                        92980  000041
00142    41*          ROOT(NEIG) = R3                                                     92990  000050
00143    42*          IF(AUTO) GO TO 10                                                   93000  000052
00143    43*     C                                                                        93010  000052
00143    44*     C                                                                        93020  000052
00143    45*     C*****       TRY CONJUGATE AS NEXT ESTIMATION IF NOT ALREADY SAVED       93030  000052
00143    46*     C                                                                        93040  000052
00145    47*          CONJ=.NOT.CONJ                                                      93050  000054
00145    48*          RETURN                                                              93060  000056
00145    49*     C                                                                        93070  000056
00145    50*     C                                                                        93080  000056
00145    51*     C*****       SAVE COMPLEX CONJUGATE OF ROOT                              93090  000056
00145    52*     C                                                                        93100  000056
00147    53*          10 CONTINUE                                                         93110  000062
00151    54*          CONJ=.FALSE.                                                        93120  000062
00151    55*          IF (NEIG.EQ.NR) RETURN                                                     000062
00151    56*          IF (NEIG.EQ.MXEIG) RETURN                                           93130  000071
00153    57*          NEIG = NEIG + 1                                                     93140  000100
00155    58*          ROOT(NEIG)=CONJG(R3)                                                       000103
00157    59*          KD(NEIG) = 4                                                        93160  000110
00161    60*          NI(NEIG) = 0                                                        93170  000112
00161    61*          RETURN                                                              93180  000113
00161    62*     C                                                                        93190  000113
00161    63*     C                                                                        93200  000113
00161    64*     C*****       SAVE REAL ROOT                                              93210  000113
00161    65*     C                                                                        93220  000113
```

```
00162      66*          20 CONTINUE
00163      67*             CONJ=.FALSE.
00164      68*             ROOT(NEIG) = R3
00165      69*             RETURN
00166      70*             END
```

```
93230    000117
93240    000117
93250    000117
93260    000123
93270    000136
```

END OF COMPILATION: NO DIAGNOSTICS.

```
@FOR,USM F,SAVEST,F,SAVEST
FOR (EIX-05/23/74-08:47:46 (D,I)

      SUBROUTINE SAVEST     ENTRY POINT 000054

      STORAGE USED: CODE(I) 000060; DATA(D) 000237; BLANK COMMON(2) 000000

      COMMON BLOCKS:

        0003   KEEP9   000705

      EXTERNAL REFERENCES (BLOCK, NAME)

        0004   NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001  000325 1001      0001  000017 I14G      0001  000040 I27G      0003 C 000226 EA      0000 I 000227 I
      0000  000231 INJP$     0003  000572 KD        0000 I 000226 N       0003 I 000455 NA      0003  000454 NEIG
      0003  000457 NI        0003  000456 NR        0003 C 000000 ROOT    0000 C 000000 TMPEST

00101     1*          SUBROUTINE SAVEST(M)                                          000000
00103     2*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)        000000
00104     3*          COMPLEX       ROOT,EA                                         000000
00105     4*          COMPLEX       TMPEST(75)                                      000000
00105     5*     C                                                                  000000
00105     6*     C                                                                  000000
00105     7*     C  SAVE THE CONTINUOUS SYSTEM ROOT ESTIMATES                       000000
00105     8*     C                                                                  000000
00105     9*          IF (M.EQ.2) GO TO 100                                         000003
00110    10*          N = NA                                                        000005
00111    11*          IF (N.EQ.0) RETURN                                            000012
00113    12*          DO 10 I=1,N                                                   000017
00114    13*          TMPEST(I) = EA(I)                                             000021
00117    14*       10 CONTINUE                                                      000021
00121    15*          RETURN                                                        000021
00121    16*     C                                                                  000021
00121    17*     C                                                                  000021
00121    18*     C  RESTORE THE CONTINUOUS SYSTEM ROOT ESTIMATES                    000021
00121    19*     C                                                                  000025
00122    20*      100 CONTINUE                                                      000025
00123    21*          NA = N                                                        000026
00124    22*          IF (NA.EQ.0) RETURN                                           000033
00126    23*          DO 110 I=1,NA                                                 000040
00131    24*          EA(I) = TMPEST(I)                                             000042
00132    25*      110 CONTINUE                                                      000042
00134    26*          RETURN                                                        000057
00135    27*          END
```

END OF COMPILATION: NO DIAGNOSTICS.

```
SFOR.USR [illegible]AVRUT,F.SAVRUT
FOR SEIX-05/23/74-08:47:57 (1,2)

      SUBROUTINE SAVRUT     ENTRY POINT 000070

      STORAGE USED: CODE(1) 000100; DATA(0) 000015; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP1   000026
      0004   KEEP5   000074
      0005   KEEP9   000705
      0006   KEEP11  007723
      0007   KEEP16  000031

      EXTERNAL REFERENCES (BLOCK, NAME)

      0010   NERR35

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      1001   000053 .1000L    0001   000024 122G     0004 L 000071 BOTH     0007 L 000022 DEBUG    0005 C 000226 EA
      0004 L 000073 GPRINT    0000 I 000000 I        0000   000001 INJPS    0007   000010 IOPEN    0004 I 000067 ITHZT
      0007   000011 JOPEN     0005 I 000572 KD       0004 I 005670 KDS      0007   000001 KODE     0007 L 000023 LFLT
      0003   000000 MAXIT     0004 L 000072 MODIFY   0003   000001 MXEIG    0003 I 000024 MXEIGT   0003   000023 MXEST
      0003   000002 MXFRM     0003   000003 MXNBM    0003   000004 MXNCOF   0003   000005 MXNCT    0003   000025 MXNCV
      0003   000006 MXNE      0003   000007 MXNEQ    0003   000010 MXNFI    0003   000011 MXNG     0003   000012 MXNPH
      0003   000013 MXNPP     0003   000014 MXNQPT   0003   000015 MXNSM    0003   000016 MXNSP    0003   000017 MXNTM
      0003   000020 MXNV      0003   000021 MXNZT    0003   000022 MXPOLY   0005   000455 NA       0006 I 000454 NEIG
      0006   007722 NEIGZT    0005 I 000457 NI       000A I 003720 NIS      0007 L 000027 NOMNAL   0007 L 000030 NOTYET
      0005   000456 NR        0007   000002 NRCLPL   0007   000003 NRPOLE   0007   000004 NRZERO   0006 I 007640 NSEIG
      0007   000005 NXB       0007   000006 NXN      0007   000007 NXR      0004   000004 NZT      0007   000014 PCPL
      0007   000015 PFAC      0007   000013 PVAR     0007   000016 PSLOSH   0007   000012 PVAR     0005 C 000000 ROOT
      0006 C 000000 SEIG      0007   000000 STAGE    0007   000000 SUPERK   0004   000003 TD       0007 L 000017 YESMTX
      0007 L 000024 YESPCH    0007 L 000020 YESRAN   0007 L 000025 YESRLP   0007 L 000026 YESSRL   0007 L 000021 YESSRP
      0004 L 000070 YESZOH    0004   000002 ZM       0004   000001 ZT       0004   000005 ZTVAL

      00101    1*          SUBROUTINE SAVRUT                                                          93280   000003
      00103    2*          COMMON/KEEP1/ MAXIT,MXEIG,MFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,              93290   000003
      00103    3*        1             MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,            93300   000003
      00103    4*        2             MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV                        93310   000003
      00104    5*          COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,             93320   000003
      00104    6*        1             MODIFY,GPRINT                                                93330   000003
      00105    7*          LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                                    93350   000003
      00105    8*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)                     93360   000003
      00107    9*          COMPLEX       ROOT,EA                                                      000003
      00111   10*          COMMON/KEEP11/SEIG(1000),NIS(10-0),KDS(1000),NSEIG(50),NEIGZT              93390   000003
      00111   11*          COMPLEX       SEIG                                                         93400   000003
      00113   12*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,                 93410   000003
      00112   13*        1             IOPEN,JOPEN,PVAR,PNOV,PCPL,PFAC,PSLOSH,                      93420   000003
```

```
00112    14*          2              YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,     93430   000003
00112    15*          3              YESSRL,NOMNAL,NOTYET                               93440   000003
00113    16*        LOGICAL          YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,     93450   000003
00113    17*          1              YESSRL,NOMNAL,NOTYET                               93460   000003
00113    18*      C                                                                     93470   000003
00113    19*      C                                                                     93480   000003
00113    20*      C  SAVE ROOTS FOR SAMPLE RATE ROOT LOCUS                              93490   000003
00113    21*      C                                                                     93500   000003
00114    22*            NOTYET = .FALSE.                                                93510   000003
00115    23*            NSEIG(ITHZT) = 0                                                93520   000004
00116    24*            IF (NEIGZT.GE.MXEIGT) RETURN                                    93530   000005
00120    25*            NSEIG(ITHZT) = NEIGZT + 1                                       93540   000015
00121    26*            DO 100 I=1,NEIG                                                 93550   000024
00124    27*            NEIGZT = NEIGZT + 1                                             93560   000024
00125    28*            SEIG(NEIGZT) = ROOT(I)                                          93570   000031
00126    29*            WIS(NEIGZT) = W1(I)                                             93580   000033
00127    30*            KDS(NEIGZT) = KD(I)                                             93590   000035
00130    31*            IF (NEIGZT.GE.MXEIGT) GO TO 1000                                93600   000037
00132    32*        100 CONTINUE                                                        93610   000047
00134    33*            RETURN                                                          93620   000047
00134    34*      C                                                                     93630   000047
00134    35*      C                                                                     93640   000047
00134    36*      C                                                                     93650   000047
00135    37*       1000 CONTINUE                                                        93660   000053
00136    38*            RETURN                                                          93670   000053
00137    39*            END                                                             93680   000077

      END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
FOR,USW F*SCALE,F*SCALE
FOR SEIX=05/23/74-08:48:10 (2.3)

      SUBROUTINE SCALE      ENTRY POINT 000164

      STORAGE USED: CODE(I) 000216; DATA(D) 000021; BLANK COMMON(2) 000000

      EXTERNAL REFERENCES (BLOCK, NAME)

          0003    FIXIT
          0004    NERR35

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

          0001    000102 10L        0001    000123 20L        0001    000146 50L        0000 R 000003 A        0000 R 000005 B
          0000 R  000007 C          0000 R  000002 D          0000    000013 INJPS      0000 I 000001 M        0000 I 000000 N
          0000 C  000003 X          0000 C  000005 Y          0000 C  000007 Z

00001      1*          SUBROUTINE SCALE(FX,NX,FY,NY,FZ,NZ)                                    000000
00002      2*          COMPLEX       FX,FY,FZ                                                 000000
00003      3*          COMPLEX       X,Y,Z                                                    000000
00004      4*          DIMENSION     A(2),B(2),C(2)                                           000000
00005      5*          EQUIVALENCE   (X,A(1)),(Y,B(1)),(Z,C(1))                               000000
00006      6*    C                                                          93710             000000
00007      7*    C                                                          93720             000000
00008      8*    C  SCALE THREE PARAMETER VALUES                            93730             000000
00009      9*    C                                                          93740             000000
00010     10*          X = FX                                                                  000000
00011     11*          Y = FY                                                                  000001
00012     12*          Z = FZ                                                                  000003
00013     13*          N=NX                                                       93750        000005
00014     14*          IF(NY.GT.N) N=NY                                           93760        000007
00015     15*          IF(NZ.GT.N) N=NZ                                           93770        000016
00016     16*          CALL FIXIT(X,NX,N)                                         93780        000025
00017     17*          CALL FIXIT(Y,NY,N)                                         93790        000032
00018     18*          CALL FIXIT(Z,NZ,N)                                         93800        000037
00019     19*          M = N                                                                   000044
00020     20*          D = AMAX1(ABS(A(1)),ABS(A(2)),ABS(B(1)),ABS(B(2)),                      000046
00021     21*         *          ABS(C(1)),ABS(C(2)))                                        000046
00022     22*       10 CONTINUE                                                               000102
00023     23*          IF (D.GT.1.0) GO TO 20                                                  000102
00024     24*          D = D*1.E10                                                             000106
00025     25*          N = N - 10                                                              000111
00026     26*          IF (N .EQ. M-20) GO TO 20                                               000114
00027     27*          GO TO 10                                                                000121
00028     28*       20 CONTINUE                                                               000123
00029     29*          IF (D.EQ.0) GO TO 50                                                    000123
00030     30*          CALL FIXIT(X,NX,N)                                                      000126
00031     31*          CALL FIXIT(Y,NY,N)                                                      000133
00032     32*          CALL FIXIT(Z,NZ,N)                                                      000140
```

```
00142     33*        50 CONTINUE                                                  000146
00143     34*           FX = X                                                    000146
00144     35*           FY = Y                                                    000147
00145     36*           FZ = Z                                                    000151
00146     37*           RETURN                                          93810     000153
00147     39*           END                                             93820     000215

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
>FOR,US# F,SEPDPC,F,SEPDPC
FOR SEIX-05/23/74-08:48:21 (1,2)

      SUBROUTINE SEPDPC    ENTRY POINT 000014

   STORAGE USED: CODE(1) 000024; DATA(0) 000010; BLANK COMMON(2) 000000

      EXTERNAL REFERENCES (BLOCK, NAME)

         0003    NERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

         0000 R 000000 D          0000 C 000000 E          0000  000002 INJP$

00101     1*          SUBROUTINE SEPDPC (A,B,C)                    9383  000000
00103     2*          COMPLEX      A,E                                     000000
00105     3*          DIMENSION    D(2)                                    000000
00105     4*          EQUIVALENCE  (D(1),E)                        9386  000000
00105     5*    C                                                  9387  000000
00105     6*    C                                                  9388  000000
00105     7*    C     PURPOSE                                      9389  000000
00105     8*    C           SEPARATE COMPLEX                               000000
00105     9*    C                                                  9391  000000
00105    10*    C     PARAMETERS                                   9392  000000
00105    11*    C                                                  9393  000000
00105    12*    C       A    =  DOUBLE PRECISION COMPLEX NUMBER    9394  000000
00105    13*    C       B    =  DOUBLE PRECISION REAL PART OF A    9395  000000
00105    14*    C       C    =  DOUBLE PRECISION IMAG PART OF A    9396  000000
00105    15*    C                                                  9397  000000
00106    16*          E = A                                        9398  000000
00107    17*          B = D(1)                                     9399  000001
00110    18*          C = D(2)                                     9400  000003
00111    19*          RETURN                                       9401  000005
00112    20*          END                                          9402  000023

      END OF COMPILATION:    NO DIAGNOSTICS.
```

```
FOR:USR F:SOLVE,F:SOLVE
FOR SEIX-05/23/74-08:48:35 (1.2)

      SUBROUTINE SOLVE      ENTRY POINT 000552

      STORAGE USED: CODE(1) 000555; DATA(0) 000052; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP14 000031
      0004   KEEP16 000031
      0005   CRUD1  000016
      0006   CRUD3  016115

      EXTERNAL REFERENCES (BLOCK, NAME)

      0007   CDABV
      0010   CDV$
      0011   CSQRT
      0012   U%DU$
      0013   NIO2$
      0014   UERR3$

      STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001  000460 1L      0000  000000 600F    0006 L 016113 AUTO    0006 C 016072 B       0005 C 000000 B2
0006 R 016070 C      0007 R 000000 CDABV   0006 L 016112 CONJ    0006 C 000000 CU      0004 L 000022 DEBUG
0003   000025 DEG    0005 C 000012 DENOM   0005 C 000006 DEN1    0005 C 000010 DEN2    0006 L 016110 DONE
0005 C 000002 DV2    0005 C 000004 DV3     0003   000012 DD      0003   000013 D1      0003   000014 D2
0003   000015 D3     0003   000016 D4      0003 C 000010 FIFTEN  0003   000017 FIFTY   0003 C 000006 FOUR
0006 C 016040 FPR0   0006 C 016042 FPR1    0006 C 016044 FPR2    0006 C 016046 FR0     0006 C 016050 FR1
0006 C 016052 FR2    0003 C 000000 HALF    0004   000040 INUPS   0004   000010 IOPEN   0004   000011 JOPEN
0004   000001 KODE   0004 L 000023 LFLT    0006   016102 NCT     0006   016107 NEST    0006   016076 NFPO
0006   016077 NFP1   0006   016100 NFP2    0006   016101 NITER   0006   016104 NKODE   0004 L 000027 NOMNAL
0004 L 000030 NOTYET 0004   000002 NRCLPL  0006   016103 NREG    0004   000003 NRPOLE  0004   000004 NRZERO
0006   016105 NSTART 0006   016106 NTIME   0004   000005 NXB     0004   000006 NXN     0004   000007 NXR
0003   000026 N1     0003   000027 N2      0003   000030 N3      0003 C 000002 ONE     0004   000014 PCPL
0004   000015 PFAC   0003   000020 PI      0003   000021 PI2     0004   000013 PNOM    0006 C 016054 PRO
0006 C 016056 PRI    0006 C 016060 PR2     0004   000016 PSLOSH  0004   000012 PVAR    0003   000024 RADDEG
0006 L 016114 REGSEL 0006 L 016111 RESTRT  0003   000023 RPI     0006 C 016062 RQ      0006 C 016064 R1
0006 C 016066 R2     0006 C 016070 R3      0003   000022 SMALL   0004   000000 STAGE   0005 C 000014 TEMP
0003 C 000004 TWO    0006 C 016074 UL      0004 L 000017 YESHIX  0004 L 000024 YESPCH  0004 L 000020 YESRAN
0004 L 000025 YESSLP 0004 L 000026 YESSRL  0004 L 000021 YESSRP

00101          1*          SUBROUTINE SOLVE                                          94030  000000
00103          2*          COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIETEN,DD,D1,D2,D3,D4,FIFTY,PI,   94040  000000
00103          3*         1      PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                          94050  000000
00103          4*          COMPLEX      HALF,ONE,TWO,FOUR,FIETEN                          000000
00105          5*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,    94080  000000
00105          6*         1      IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                   94090  000000
```

```
00105      7*          2           YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      94100  000000
00105      8*          3           YESSRL,NOMNAL,NOTYET                                94110  000000
00106      9*        LOGICAL       YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,      94120  000000
00106     10*          1           YESSRL,NOMNAL,NOTYET                                94130  000000
00107     11*        COMMON/CRUD1/ B2,DV2,DV3,DEN1,DEN2,DENOM,TEMP                     94140  000000
00110     12*        COMPLEX       B2,DV2,DV3,DEN1,DEN2,DENOM,TEMP                     94150  000000
00111     13*        COMMON/CRUD3/ CU(60,60),FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,   94160  000000
00111     14*          1           R0,R1,R2,R3,B,U,NFP0,NFP1,NFP2,NITER,NCT,NREG,NKODE, 94170  000000
00111     15*          2           NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL      94180  000000
00112     16*        COMPLEX       CU,FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,          94190  000000
00112     17*          1           R0,R1,R2,R3,B,U                                     94200  000000
00113     18*        LOGICAL       DONE,RESTRT,CONJ,AUTO,REGSEL                        94210  000000
00114     19*        DIMENSION     C(2)                                                94220  000000
00115     20*        EQUIVALENCE   (C(1),R3)                                           94230  000000
00115     21*    C                                                                     94240  000000
00115     22*    C                                                                     94250  000000
00115     23*    C*****       FORM QUADRATIC FIT OF LAST THREE ITERANTS AND FUNCTIONAL  94260  000000
00115     24*    C*****       VALUES                                                   94270  000000
00115     25*    C*****       SELECT ROOT OF QUADRATIC CLOSEST TO THE LAST ITERANT AS THE 94280  000000
00115     26*    C*****       NEW ITERANT                                              94290  000000
00115     27*    C                                                                     94300  000000
00115     28*          DV2=(R2-R1)/(R1-R0)                                             94310  000000
00117     29*          B2=(DV2**2)*FPR0-((ONE+DV2)**2)*FPR1+(ONE+TWO*DV2)*FPR2         94320  000020
00120     30*          TEMP=B2*CSQRT(ONE-FOUR*DV2*(ONE+DV2)*(FPR2/B2)*((DV2*FPR0       000154
00120     31*         *        -(ONE+DV2)*FPR1+FPR2)/B2))                              000154
00121     32*          DEN1=B2+TEMP                                                    94350  000347
00122     33*          DEN2=B2-TEMP                                                    94360  000354
00123     34*          DENOM=DEN1                                                      94370  000363
00124     35*          IF(CDABV(DEN2).GT.CDABV(DEN1)) DENOM=DEN2                       94380  000365
00126     36*          DV3=ONE                                                         94390  000402
00127    *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00127     37*          IF (CDABV(DENOM).EQ.0.0) GO TO 1                                000404
00131     38*          DV3=-TWO*(ONE+DV2)*FPR2/DENOM                                   94410  000411
00132     39*        1 CONTINUE                                                        94420  000460
00133     40*          R3=R2+DV3*(R2-R1)                                               94430  000460
00134     41*          IF (ABS(C(1)).LT.1.E-4) C(1)=0.0                                000506
00135     42*          IF (ABS(C(2)).LT.1.E-4) C(2)=0.0                                000514
00140     43*          IF (DEBUG) WRITE(6,600) DV2,B2,TEMP,DEN1,DEN2,DENOM,DV3,R3      94450  000522
00153     44*      600 FORMAT('0 SOLVE',6X,'DV2  ',2E14.5,5X,'B2  ',2E14.5,5X,'TEMP  ',  000541
00153     45*         12E14.5/13X,'DEN1 ',2E13.5,9X,'DEN2 ',2E14.5,5X,'DENOM ',        000541
00153     46*         22E14.5/13X,'DV3  ',2E14.5,5X,'R3  ',2E14.5/)                    000541
00154     47*          RETURN                                                          94490  000541
00155     48*          END                                                             94500  000554

      END OF COMPILATION:       1 DIAGNOSTICS.
```

```
FOR:USW F:SRRL,F:SRRL
FOR SEIX-05/23/74-08:48:54 (4,5)

SUBROUTINE SRRL      ENTRY POINT 000206

STORAGE USED: CODE(1) 000222; DATA(0) 000054; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003  KEEP1  000026
0004  KEEP4  000263
0005  KEEP5  000074
0006  KEEP9  000705
0007  KEEP11 007723
0010  KEEP15 000260
0011  KEEP16 000031
0012  KEEP19 000005
0013  KEEP21 001133
0014  CRU02  003736
0015  CRU03  016115

EXTERNAL REFERENCES (BLOCK, NAME)

0016  SAVEST
0017  GETEST
0020  ROOTER
0021  WRDU$
0022  NIO2$
0023  NERR4$
0024  NERR3$

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000141 1000L   0001 000112 172G    0001 000147 2000L   0001 000160 3000L   0000 000001 600F
0015 L 016113 AUTO  0015 C 016072 B     0005 L 000071 BOTH  0004 000212 BY      0015 L 016112 CONJ
0010 000116 CPSLBL  0015 C 000000 CU    0010 000024 DA      0010 000130 DCBLBL  0004 000166 DD
0011 L 000022 DEBUG 0010 000131 DEGLBL  0014 003733 DELTA   0015 L 016110 DONE  0006 C 000226 EA
0014 001753 EIP     0014 000003 ERP     0010 000146 FMAT    0010 000132 FMT     0015 C 016040 FPRO
0015 C 016042 FPR1  0015 C 016044 FPR2  0015 C 016046 FR0   0015 C 016050 FR1   0015 C 016052 FR2
0010 000162 F6      0010 000163 F7      0004 000004 GAIN    0014 R 000001 GAINV 0005 L 000073 GPRINT
0004 000151 GSYM    0000 I 000000 I     0010 000100 ID3     0010 000107 IDG     0000 000034 INJP$
0011 000010 IOPEN   0005 I 000067 ITHZT 0011 000011 JobEN   0006 I 000572 KD    0007 I 005670 KDS
0011 I 000001 KDOF  0010 000164 LABEL1  0010 000210 LABEL2  0010 000234 LABEL3  0011 L 000023 LFLT
0012 L 000001 LGAIN 0013 I 000704 LOCDZ 0013 000703 LOCNZ   0012 L 000001 LPHASE 0012 L 000002 LPOLES
0012 L 000003 LSOPL 0012 L 000004 LZEROS 0003 000000 MAXIT  0005 L 000072 MODIFY 0003 I 000001 MXEIG
0003 I 000024 MXEIGT 0003 000023 MXEST  0003 000002 MAXRM   0003 000003 MXNBM   0003 000004 MXNCOF
0003 000005 MXNCT   0003 000025 MXNCV   0003 000006 MXNE    0003 000007 MXNEQ   0003 000010 MXNFI
0003 000011 MXNG    0003 000012 MXNRH   0003 000013 MXNRP   0003 000014 MXNRT   0003 000015 MXNSM
0003 000016 MXNSP   0003 000017 MXNTM   0003 000020 MXNV    0003 000021 MXNZT   0003 000022 MXPOLY
0006 000456 NA      0015 016102 NCT     0013 001020 NDCZ    0006 I 000454 NEIG  0007 I 007722 NEIGZT
0015 016107 NEST    0015 016076 NFP0    0015 016077 NFP1    0015 016100 NFP2    0004 000003 NGAIN
0004 000154 NGR     0014 003734 NGSYM   0004 I 000457 NI    0007 I 003720 NIS   0015 016101 NITER
0015 016104 NKODE   0013 000705 NNCZ    0011 L 000027 NOMNAL 0011 L 000030 NOTYET 0014 003723 NPG
```

```
0004    016066 NPHASE   0014   003724 NPP      0014   003735 NPSYM    0006 I 000456 NR       0011   000002 NRCLPL
0015    016103 NREG     0004   000153 NRLFR    0011   000003 NRPOLE   0011   000004 NRZERO   0007 I 007640 NSEIG
0014    003725 NSHIFT   0015   016105 NSTART   0015   016106 NTIME    0013 I 000702 NUMZ     0011   000005 NXB
0011    000006 NXN      0011   000007 NXR      0005   000004 NZT      0011   000014 PCPL     0013   000341 PDZ
0011    000015 PFAC     0004   000000 PG       0004   000067 PHASE    0014   000000 PHAVAR   0011   000013 PNOM
0013    000000 PNZ      0004   000001 PP       0004   000002 PPLT     0015 C 016054 PRO      0015 C 016056 PR1
0015 C  016060 PR2      0011   000016 PSLOSH   0004   000152 PSYM     0011   000012 PVAR     0010   000034 P36
0010 C  000000 REGION   0015 L 016114 REGSEL   0015 L 016111 RESTRT   0010   000030 RID      0006 C 000000 ROOT
0004    000200 RX       0015 C 016062 R0       0015 C 016064 R1       0015 C 016066 R2       0015 C 016070 R3
0007 C  000000 SEIG     0014   000002 SHIFT    0014   000014 SPACE    0011   000000 STAGE    0005   000000 SUPERK
0005    000003 TD       0015 C 016074 U        0010   000010 WIDTH    0014   003726 XR       0014   003727 YB
0011 L  000017 YESMTX   0011 L 000024 YESPCH   0011 L 000020 YESRAW   0004 L 000262 YESRL    0011 L 000025 YESRLP
0011 L  000026 YESSRL   0011 L 000021 YESSRP   0005 L 000070 YESZOH   0010   000020 YINC    0005   000002 ZM
0005    000001 ZT       0005   000005 ZTVAL
```

```
00101     1*          SUBROUTINE SRRL(*)                                                    9513₀  000000
00103     2*          COMMON/KEEP1/ MAXIT,MXEIG,MYFRM,MXNBM,MXNCOF,MXNGT,MXNE,MXNEQ,       9514₀  000000
00103     3*                1       MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,      9515₀  000000
00103     4*                2       MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV                  9516₀  000000
00104     5*          COMMON/KEEP4/ PG,PP,PPLT,NGAIN,GAIN(50),NPHASE,PHASE(50),GSYM,       9517₀  000000
00104     6*                1       PSYM,NRLFR,NGR(10),DD(10),RX(10),BY(4,10),YESRL        9518₀  000000
00105     7*          LOGICAL       YESRL                                                 9519₀  000000
00106     8*          COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,      9520₀  000000
00106     9*                1       MODIFY,GPRINT                                         9521₀  000000
00107    10*          LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                             9523₀  000000
00113    11*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)              9524₀  000000
00111    12*          COMPLEX       ROOT,EA                                                      000000
00112    13*          COMMON/KEEP11/SEIG(1000),NIS(1000),KDS(1000),NSEIG(50),NEIGZT       9527₀  000000
00113    14*          COMPLEX       SEIG                                                  9528₀  000000
00114    15*          COMMON/KEEP15/REGION(4),WIDTH(4),SPACE(4),YINC(4),                         000000
00114    16*                1       DA(4),RID(4),A36(36),IDB(7),IDG(7),GPSLBL(40),             000000
00114    17*                2       DCOLBL,DEGLBL,FMT(12),FMAT(12),F6,F7,LABEL1(20),           000000
00114    18*                3       LABEL2(20),LABEL3(20)                                      000000
00115    19*          COMPLEX       REGION                                                     000000
00115    20*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,         9529₀  000000
00115    21*                1       IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,            9530₀  000000
00115    22*                2       YESMTX,YESRAW,YESRP,DEBUG,LFLT,YESPCH,YESRLP,      9531₀  000000
00115    23*                3       YESSRL,NOMNAL,NOTYET                               9532₀  000000
00117    24*          LOGICAL       YESMTX,YESRAW,YESRP,DEBUG,LFLT,YESPCH,YESRLP,      9533₀  000000
00117    25*                1       YESSRL,NOMNAL,NOTYET                               9534₀  000000
00120    26*          COMMON/KEEP19/LGAIN,LPHASE,LPOLES,LSORL,LZEROS                             000000
00121    27*          LOGICAL       LGAIN,LPHASE,LPOLES,LSORL,LZEROS                             000000
00122    28*          COMMON/KEEP21/PNZ(225),PDZ(225),NUMZ,LOCNZ,LOCDZ,NNCZ(75),NDCZ(75)         000000
00123    29*          COMMON/CRUD2/ PHAVAR,GAINV,SHIFT,ERP(1000),EIP(1000),NPG,NPP       9618   000000
00123    30*                1       NSHIFT,XR,YB(4),DELTA,NGSYM,NPSYM                  9619   000000
00124    31*          COMMON/CRUD3/ CU(60,60),FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,    9535₀  000000
00124    32*                1       R0,R1,R2,R3,B,U,NFP0,NFP1,NFP2,NITER,NCT,NREG,NKODE,9536₀  000000
00124    33*                2       NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL     9537₀  000000
00125    34*          COMPLEX       CU,FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,                000000
00125    35*                1       R0,R1,R2,R3,B,U                                    9539₀  000000
00125    36*          LOGICAL       DONE,RESTRT,CONJ,AUTO,REGSEL                       9540₀  000000
00125    37*     C                                                                     9541₀  000000
00125    38*     C                                                                     9542₀  000000
00125    39*     C  SAVE NOMINAL ROOTS FOR SAMPLE RATE                                 9543₀  000000
```

```
00125    40*     C
00127    41*           IF (NEIGZT.GE.MXEIGT) RETURN
00131    42*           AUTO = .TRUE.
00131    43*     C
00131    44*     C
00131    45*     C  CHECK FOR GAIN OF 1.0
00131    46*     C
00131    47*     C
00131    48*     C
00131    49*     C  EVALUATE THE NOMINAL ROOTS FOR THE SAMPLING RATE
00131    50*     C
00132    51*           IF (.NOT.NOTYET) RETURN
00134    52*           REGION(1) = (0.,1.)
00135    53*           REGION(2) = (-1.,0.)
00136    54*           REGION(3) = (1.,0.)
00137    55*           REGION(4) = (0.,0.)
00141    56*           LSORL = .TRUE.
00141    57*           IF (NUMZ.EQ.0) GO TO 3000
00143    58*           LGAIN = .TRUE.
00144    59*           LPHASE = .FALSE.
00145    60*           LPOLES = .FALSE.
00145    61*           NOMNAL = .TRUE.
00147    62*           LZEROS = .FALSE.
00151    63*           GAINV = 1.0
00151    64*           NR = LOCDZ - NUMZ
00152    65*           IF (MODIFY) NR=NR+1
00154    66*           IF (NR .GT. MXEIG) NR = MXEIG
00156    67*           CALL SAVEST(1)
00157    69*           CALL GETEST
00161    69*           CALL ROOTER
00161    70*           CALL SAVEST(2)
00162    71*           LSORL = .FALSE.
00163    72*           LGAIN = .FALSE.
00164    73*           NOMNAL = .FALSE.
00165    74*           NSEIG(ITHZT) = 0
00165    75*           IF (NEIG.EQ.0) GO TO 1000
00171    76*           NSEIG(ITHZT) = NEIGZT + 1
00171    77*           DO 100 I=1,NEIG
00174    78*           NEIGZT = NEIGZT + 1
00175    79*           SEIG(NEIGZT) = ROOT(I)
00176    80*           KDS(NEIGZT) = KD(I)
00177    81*           NIS(NEIGZT) = NI(I)
00201    82*           IF (NEIGZT.GE.MXEIGT) GO TO 2000
00202    83*       100 CONTINUE
00204    84*           RETURN
00204    85*     C
00204    86*     C
00204    87*     C  NO NOMINAL ROOTS
00204    88*     C
00205    89*      1000 CONTINUE
00205    90*           KODE = 59
00207    91*           RETURN 1
00207    92*     C
00207    93*     C
00207    94*     C  CAPACITY FOR STORING ROOTS FOR SAMPLE RATE ROOT LOCUS IS REACHED
00207    95*     C
00211    96*      2000 CONTINUE
```

```
95440  000000
95450  000000
95460  000007
95470  000007
95480  000007
95490  000007
95500  000007
95510  000007
95520  000007
95530  000007
95540  000007
95550  000011
       000016
       000020
       000022
       000024
       000026
       000030
       000032
       000033
       000034
       000035
       000036
       000037
       000041
       000044
       000051
       000060
95570  000063
95580  000065
       000067
       000072
       000073
       000074
95590  000075
95600  000077
95610  000101
95620  000104
95630  000112
95640  000117
95650  000121
95660  000123
95670  000125
95680  000135
95690  000135
95700  000135
95710  000135
95720  000135
95730  000135
95740  000141
95750  000141
95760  000142
95770  000142
95780  000142
95790  000142
95800  000142
95810  000147
```

```
00211                 WRITE(6,600) NEIGZT                                          95820   000147
00214    92*      600 FORMAT('1','ONLY A MAXIMUM OF ',I4,' ROOTS HAVE BEEN SAVED FOR SAM  95830   000154
00214    99*         *PLE RATE ROOT LOCUS')                                       95840   000154
00215   100*          RETURN                                                      95850   000154
00215   101*      C                                                                       000154
00215   102*      C                                                                       000154
00215   103*      C  NO Z DOMAIN TRANSFER FUNCTION HAS BEEN STORED                        000154
00215   104*      C                                                                       000154
00216   105*       3000 CONTINUE                                                          000160
00217   106*          KODE = 109                                                          000160
00221   107*          RETURN 1                                                            000161
00221   108*          END                                                          95920   000221

      END OF COMPILATION:       NO DIAGNOSTICS.
```

?FOR,USW e[redacted]LPP,F,SRRLPP
FOR SEIX-05/23/74-08:49:07 (3,9)

SUBROUTINE SRRLPP ENTRY POINT 000255

STORAGE USED: CODE(1) 002722; DATA(0) 000045; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003	KEEP2	000047
0004	KEEP4	000263
0005	KEEP5	000074
0006	KEEP9	000705
0007	KEEP11	007723
0010	KEEP16	000031
0011	KEEP19	000005
0012	CRUD2	003737
0013	CRUD3	000112

EXTERNAL REFERENCES (BLOCK, NAME)

0014	RLPRNT
0015	RLPLOT
0016	NERR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001		000103	100L	0001		000015	135G	0001		000061	153G	0001		000117	175G	0001		000140	211G
0001		000147	217G	0001		000232	225L	0001		000216	247G	0001		000225	255G	0005	L	000071	BOTH
0004	R	000212	BY	0013	R	000042	BYTEMP	0004	R	000166	DD	0013	R	000016	DDTEMP	0010	L	000022	DEBUG
0012		003734	DELTA	0003	D	000002	DHFILT	0006	C	000226	EA	0012	R	001754	EIP	0012	R	000004	ERP
0004		000004	GAIN	0012	R	000002	GAINV	0005	L	000073	GPRINT	0004	R	000151	GSYM	0013	R	000002	GSYMTP
0003		000037	HACC	0003	R	000025	HAST	0003		000035	HATT	0003	D	000000	HBLANK	0003	R	000004	HBLK
0003		000034	HCC#	0003		000033	HCW	0003		000032	HDEC	0003		000027	HDOT	0003		000006	HESTI
0003		000043	HFGO	0003		000044	HFGN	0003		000040	HFPD	0003		000045	HFPOD	0003		000042	HFPDN
0003		000041	HFPU	0003		000010	HGENE	0003		000031	HHC	0003		000005	HKEY	0003		000007	HNATR
0003		000014	HNE#	0003		000046	HNOMI	0003		000015	HNYQU	0003		000023	HO	0003		000026	HPLUS
0003		000036	HRATE	0003		000011	HRAW	0003		000012	HRETA	0003		000017	HROLL	0003		000016	HROOT
0003		000013	HSTAN	0003		000030	HSTAR	0003		000020	HS1C	0003		000021	HS2	0003		000022	HS4B
0003	B	000024	HX	0000	I	000002	I	0000		000020	INJPS	0010		000010	IOPEN	0005		000067	ITHZT
0000	I	000005	J	0010		000011	JOPEN	0006	I	000572	KD	0007	I	005670	KDS	0010		000001	KODE
0000	I	000004	L	0000	L	000003	LBEG	0000	I	000004	LEND	0010	L	000023	LFLT	0011	L	000000	LGAIN
0011	L	000031	LPHASE	0011	L	000002	LPOLES	0011	L	000003	LSORL	0011	L	000034	LZEROS	0005	L	000072	MODIFY
0004		000455	MA	0004	I	000454	MEIG	0007	I	007722	MEIGZT	0004		000003	NGAIN	0004	I	000154	NGR
0013	I	000034	NGRTP	0012		003735	NGSYM	0005	I	000457	NI	0007	I	003720	NIS	0010	L	000027	NOMNAL
0010	L	000033	NOTYET	0012	I	003724	NPG	0004		000046	NPHASE	0012		003725	NPP	0012		003736	NPSYM
0006		000456	NR	0010		000002	NRCLPL	0004	I	000153	NRLFR	0010		000003	NRPOLE	0010		000004	NRZERO
0007	I	007640	NSEIG	0012		003726	NSHIFT	0013	I	000003	NTEMP	0010		000005	NX8	0010		000006	NXN
0010		000007	NXR	0005	I	000004	NZT	0010		000014	PCPL	0010		000015	PFAC	0004	R	000000	PG
0013	R	000000	PGTEMP	0004		000067	PHASE	0012	C	000000	PHAVAR	0010		000013	PNOM	0004	R	000001	PP
0004		000002	PPLT	0013	R	000001	PPTEMP	0010		000016	PSLOSH	0004		000152	PSYM	0010		000012	PVAR
0004	C	000000	RDOT	0004	R	000200	RX	0013	R	000030	RXTEMP	0007	C	000000	SEIG	0012		000003	SHIFT
0010		000000	STAGE	0005		000000	SUPERK	0005		000003	TD	0000	C	000000	VAL	0012		003727	XR

```
0012   00373D YB          0010 L 000017 YESMTX     0010 L 000024 YESPCH     0010 L 00002D YESRAW     0004 L 000262 YESRL
0010 L 000025 YESRLP      0010 L 000026 YESSRL     0010 L 000021 YESSRP     0005 L 00007D YESZOH     0005   000002 ZM
0005 R 000001 ZT          0005 R 000005 ZTVAL
```

```
00101     1*        SUBROUTINE SFRLPP                                                    959300  000000
00102     2*        COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,        959400  000000
00103     3*     1                HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS4B,    959500  000000
00104     4*     2                HD,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,      959600  000000
00105     5*     3                HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,     959700  000000
00106     6*     4                HNOMI                                                959800  000000
00107     7*        DOUBLE PRECISION HBLANK,DHFILT                                       000000
00108     8*        COMMON/KEEP4/ PG,PP,PPLT,NGAIN,GAIN(50),NPHASE,PHASE(50),GSYM,       960000  000000
00109     9*     1                PSYM,NRLFR,NGR(10),DD(10),RX(10),BY(4,10),YESRL      960100  000000
00110    10*        LOGICAL       YESRL                                                960200  000000
00111    11*        COMMON/KEEP5/ SUPERK,ZT,ZMATD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,     960300  000000
00112    12*     1                MODIFY,GPRINT                                        960400  000000
00113    13*        LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                            960600  000000
00114    14*        COMMON/KEEP7/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)             960700  000000
00115    15*        COMPLEX       ROOT,EA                                                      000000
00116    16*        COMMON/KEEP11/SEIG(1000),NIS(1000),KDS(1000),NSEIG(50),NEIGZT      961000  000000
00117    17*        COMPLEX       SEIG                                                 961100  000000
00118    18*        COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,           961200  000000
00119    19*     1                IOPEN,JOPEN,PVAR,NOM,PCPL,PFAC,PSLOSH,               961300  000000
00120    20*     2                YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,       961400  000000
00121    21*     3                YESSRL,NOMNAL,NOTYET                                 961500  000000
00122    22*        LOGICAL       YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,       961600  000000
00123    23*     1                YESSRL,NOMNAL,NOTYET                                 961700  000000
00124    24*        COMMON/KEEP19/LGAIN,LPHASE,LPOLES,LSORL,LZEROS                             000000
00125    25*        LOGICAL       LGAIN,LPHASE,LPOLES,LSORL,LZEROS                             000000
00126    26*        COMMON/CRUD2/ PHAVAR,GAINV,SHIFT,ERP(1000),EIP(1000),NPG,NPP,        961800  000000
00127    27*     1                NSHIFT,XR,YB(4),DELTA,NGSYM,NPSYM                    961900  000000
00128    28*        COMPLEX       PHAVAR                                               962000  000000
00129    29*        COMMON/CRUD3/ PGTEMP,PPTEMP,GSYTP,NTEMP,NGRTP(10),DDTEMP(10),        962100  000000
00130    30*     1                RXTEMP(10),BYTEMP(4,10)                              962200  000000
00131    31*        COMPLEX       VAL                                                  962300  000000
00132    32*     C                                                                      962400  000000
00133    33*     C                                                                      962500  000000
00134    34*     C  LOOP ON THE SAMPLING RATES                                          962600  000000
00135    35*     C                                                                      962700  000000
00136    36*           LSORL = .TRUE.                                                           000000
00137    37*           LGAIN = .FALSE.                                                          000001
00138    38*           LPHASE = .FALSE.                                                         000002
00139    39*           LPOLES = .FALSE.                                                         000003
00140    40*           NOMNAL = .TRUE.                                                          000004
00141    41*           LZEROS = .FALSE.                                                         000005
00142    42*           GAINV = 1.0                                                              000006
00143    43*           DO 100 I=1,NZT                                                   962900  000015
00144    44*           ZT = ZTVAL(I)                                                    963000  000015
00145    45*           LBEG = NSEIG(I)                                                  963100  000017
00146    46*           IF (LBEG.EQ.0) GO TO 100                                         963200  000021
00147    47*           IF (I.EQ.NZT) LEND=NEIGZT                                        963300  000023
00148    48*           IF (I.NE.NZT) LEND=NSEIG(I+1)-1                                  963400  000033
00149    49*           IF (LEND.LE.0) LEND=NEIGZT                                       963500  000042
00150    50*     C                                                                      963600  000042
```

```
00147    51*     C
00147    52*     C  PRINT RESULTS FOR A SAMPLING RATE
00147    53*     C
00151    54*           J = 0
00152    55*           DO 50 L=LBEG,LEND
00155    56*           J = J + 1
00156    57*           ROOT(J) = SEIG(L)
00157    58*           NI(J) = NIS(L)
00160    59*           KD(J) = KDS(L)
00161    60*        50 CONTINUE
00163    61*           NEIG = LEND - LBEG + 1
00164    62*           CALL RLPRNT
00165    63*       100 CONTINUE
00167    64*           LSORL = .FALSE.
00170    65*           NOMNAL = .FALSE.
00170    66*     C
00170    67*     C
00170    68*     C  GENERATE SAMPLING RATE ROOT LOCUS PLOT
00170    69*     C
00171    70*           IF (.NOT.YESSRP) GO TO 225
00173    71*           NPG = NEIGZT
00174    72*           DO 150 I=1,NEIGZT
00177    73*           VAL = SEIG(I)
00200    74*           ERP(I) = REAL(VAL)
00201    75*           EIP(I) = AIMAG(VAL)
00202    76*       150 CONTINUE
00204    77*           PPTEMP = PP
00205    78*           PGTEMP = PG
00206    79*           GSYMTP = GSYM
00207    80*           NTEMP = NRLFR
00210    81*           DO 175 J=1,10
00211    82*           NGRTP(J) = NGR(J)
00214    83*           DDTEMP(J) = DD(J)
00215    84*           RXTEMP(J) = RX(J)
00216    85*           DO 175 I=1,4
00221    86*           BYTEMP(I,J) = BY(I,J)
00222    87*       175 CONTINUE
00225    88*           PP = HBLK
00226    89*           PG = HAST
00227    90*           GSYM = HX
00230    91*           NRLFR = 2
00231    92*           NGR(1) = 1
00232    93*           NGR(2) = 1
00233    94*           DD(1) = .1
00234    95*           DD(2) = .2
00235    96*           RX(1) = 1.0
00236    97*           RX(2) = 2.0
00237    98*           BY(1,1) = -1.0
00240    99*           BY(1,2) = -2.0
00241   100*           CALL RLPLOT
00241   101*     C
00241   102*     C
00241   103*     C  END OF PLOTTING *** RESTORE PARAMETERS
00241   104*     C
00242   105*           PP = PPTEMP
00243   106*           PG = PGTEMP
00244   107*           GSYM = GSYMTP
```

```
9637    000042
9638    000042
9639    000042
9640    000050
9641    000051
9642    000061
9643    000063
9644    000066
9645    000071
9646    000074
9647    000074
9648    000100
9649    000104
        000104
        000105
9650    000105
9651    000105
9652    000105
9653    000105
9654    000106
9655    000110
9656    000117
9657    000117
9658    000120
9659    000121
9660    000124
9661    000124
9662    000126
9663    000130
9664    000132
9665    000140
9666    000140
9667    000141
9668    000143
9669    000147
9670    000147
9671    000153
9672    000153
9673    000155
9674    000157
9675    000161
9676    000163
9677    000165
9678    000166
9679    000170
9680    000172
9681    000174
9682    000176
9683    000177
9684    000200
9685    000200
9686    000200
9687    000200
9686    000200
9689    000202
9690    000204
9691    000206
```

```
00245     108*            NRLFR = NTEMP                                   96920     000210
00246     109*            DO 200 J=1,10                                   96930     000216
00251     110*            NGR(J) = NGRTP(J)                               96940     000216
00252     111*            DD(J) = DDTEMP(J)                               96950     000217
00253     112*            RX(J) = RXTEMP(J)                               96960     000221
00254     113*            DO 200 I=1,4                                    96970     000225
00257     114*            BY(I,J) = RYTEMP(I,J)                           96980     000225
00260     115*        200 CONTINUE                                        96990     000232
00263     116*        225 CONTINUE                                        97000     000232
00265     117*            NORMAL = .FALSE.                                97010     000232
00266     118*            RETURN                                          97020     000232
00267     119*            END                                             97030     000271

      END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
0FOR,USN F,START,F,START
FOR SE1X-05/23/74-08:49:25 (1,2)

    SUBROUTINE START     ENTRY POINT 000311

    STORAGE USED; CODE(1) 000313; DATA(0) 000123; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   KEEP14 000031
    0004   KEEP16 000031
    0005   CRUD3  016115

    EXTERNAL REFERENCES (BLOCK, NAME)

    0006   CDABV
    0007   PEVAL
    0010   DET
    0011   SCALE
    0012   COV$
    0013   NRDU$
    0014   N102$
    0015   NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001    000076 IL       0001    000126 2L       0000    000006 600F     0005 L 016113 AUTO     0005 C 016072 B
0004 R  000000 CDABV    0005 L 016112 CONJ      0005 C 000000 CU        0004 L 000022 DEBUG    0003   000025 DEG
0005 L  016110 DONE     0003   000012 D0        0003   000013 D1        0003   000014 D2       0003   000015 D3
0003    000016 D4       0003 C 000010 FIETEN    0003   000017 FIFTY     0003 C 000006 FOUR     0005 C 016040 FPRO
0005 C  016042 FPR1     0005 C 016044 FPR2      0005 C 016046 FRO       0005 C 016050 FRI      0005 C 016052 FRZ
0003 C  000000 HALF     0003   000017 INJP5     0004   000010 IOPEN     0004   000011 JOPEN    0004   000001 KODE
0004 L  000023 LFLT     0005   016102 NCT       0005   016107 NEST      0005 I 016076 NFPO     0005 I 016077 NFP1
0005 I  016100 NFP2     0000 I 000003 NFO       0000 I 000004 NFI       0000 I 000005 NF2      0005   016101 NITER
0005    016104 NKONE    0004 L 000027 NOMNAL    0004 L 000030 NOTYET    0000 I 000000 NPO      0000 I 000001 NP1
0000 I  000002 NR2      0004   000002 NRCLPL    0005   016103 NREG      0004   000003 NRPOLE   0004   000004 NRZERO
0005    016105 NSTART   0005   016106 NTIME     0004   000005 NXB       0004   000006 NXN      0004   000007 NXR
0003    000026 N1       0003   000027 N2        0003   000030 N3        0003 C 000002 ONE      0004   000014 PCPL
0004    000015 PFAC     0003   000020 PI        0003   000021 PI2       0004   000013 PNOM     0005 C 016054 PRO
0005 C  016056 PRI      0005 C 016060 PR2       0004   000016 PSLOSH    0004   000012 PVAR     0003   000024 RADDEG
0005 L  016114 REGSEL   0005 L 016111 RESTRT    0003   000023 RPI       0005 C 016062 RO       0005 C 016064 R1
0005 C  016066 R2       0005 C 016070 R3        0003   000022 SMALL     0004   000000 STAGE    0003 C 000004 TWO
0005 C  016074 U        0004 L 000017 YESHTX    0004 L 000024 YESPCH    0004 L 000020 YESRAW    0004 L 000025 YESRLP
0004 L  000024 YESSRL   0004 L 000021 YESSPP

00101     1*          SUBROUTINE START                                          94510   000000
00103     2*          COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,   94520   000000
00104     3*     1                PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                 94530   000000
00106     4*          COMPLEX        HALF,ONE,TWO,FOUR,FIETEN                               000000
00106     5*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,        94560   000000
```

```
00105                  1          IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH)                    94570   000000
00106          7*      2          YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,             94580   000000
00107          8*      3          YESSRL,NOMNAL,NOTYET                                       94590   000000
00108          9*     LOGICAL     YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP)             94600   000000
00109         10*      1          YESSRL,NOMNAL,NOTYET                                       94610   000000
00110         11*     COMMON/CPUD3/CU(60,60),FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,          94620   000000
00111         12*      1          R0,R1,R2,R3,B,U,NFP0,NFP1,NFP2,NITER,NCT,NREG,NKODE,       94630   000000
00112         13*      2          NSTART,NTIME,NEST,DONE,RESTRT,CONJ,AUTO,REGSEL             94640   000000
00113         14*     COMPLEX     CU,FPR0,FPR1,FPR2,FR0,FR1,FR2,PR0,PR1,PR2,                         000000
00114         15*      1          R0,R1,R2,R3,B,U                                            94660   000000
00115         16*     LOGICAL     DONE,RESTRT,CONJ,AUTO,REGSEL                               94670   000000
00116         17*     C                                                                      94680   000000
00117         18*     C                                                                      94690   000000
00118         19*     C*****      GIVEN-ESTIMATE-FORM-FIRST THREE ESTIMATES                   94700   000000
00119         20*     C                                                                      94710   000000
00120         21*           IF (CDABV(U) .LT. 1.E-4) GO TO 1                                         000000
00121         22*           R0 = U*(ONE + T*0*B)                                                     000006
00122         23*           R1 = U*(ONE - B)                                                         000046
00123         24*           R2 = U                                                                   000072
00124         25*           GO TO 2                                                    94760         000074
00125         26*         1 CONTINUE                                                   94770         000076
00126         27*           R0 = (-1.,0.)                                                            000076
00127         28*           R1 = B*(ONE - B)                                                         000077
00128         29*           R2 = (0.,0.)                                                             000123
00129         30*     C                                                                      94810   000123
00130         31*     C                                                                      94820   000123
00131         32*     C*****      DETERMINE FUNCTIONAL EVALUATIONS OF EACH ITERANT           94830   000123
00132         33*     C                                                                      94840   000123
00133         34*         2 CONTINUE                                                         94850   000126
00134         35*           CALL PEVAL(R0,PR0,NP0)                                           94860   000126
00135         36*           CALL PEVAL(R1,PR1,NP1)                                           94870   000132
00136         37*           CALL PEVAL(R2,PR2,NP2)                                           94880   000137
00137         38*           CALL DET(R0,FR0,NF0)                                             94890   000144
00138         39*           CALL DET(R1,FR1,NF1)                                             94900   000151
00139         40*           CALL DET(R2,FR2,NF2)                                             94910   000156
00140         41*           NFP0=NF0-NP0                                                     94920   000163
00141         42*           NFP1=NF1-NP1                                                     94930   000166
00142         43*           NFP2=NF2-NP2                                                     94940   000171
00143         44*           FPR0=FR0/PR0                                                     94950   000174
00144         45*           FPR1=FR1/PR1                                                     94960   000201
00145         46*           FPR2=FR2/PR2                                                     94970   000206
00146 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*                    000213
00147         47*           IF (CDABV(FPR0) .EQ. 0.0) NFP0=0
00148 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*                    000221
00149         48*           IF (CDABV(FPR1) .EQ. 0.0) NFP1=0
00150 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*                    000227
00151         49*           IF (CDABV(FPR2) .EQ. 0.0) NFP2=0                                         000235
00152         50*           CALL SCALE(FPR0,NFP0,FPR1,NFP1,FPR2,NFP2)                  95010   000245
00153         51*           IF(DEBUG) WRITE(6,ADD) R0,R1,R2,FPR0,FPR1,FPR2,NFP0,NFP1,NFP2,  95020   000245
00154         52*          1          FR0,FR1,FR2,NF0,NF1,NF2,PR0,PR1,PR2,NP0,NP1,NP2    95030   000301
00155         53*     ADD FORMAT('0 START',6X,'R0  ',2E14.5,5X,'R1  ',2E14.5,5X,'R2  ',           000301
00156         54*          12E14.5/13X,'FPR0',2E14.5,5X,'FPR1',2E14.5,5X,'FPR2',                    000301
00157         55*          22E14.5/13X,'NFP0',I5,10X,'NFP1',I5,10X,'NFP2',I5/13X,                   000301
00158         56*          3'FR0 ',2E14.5,5X,'FR1 ',2E14.5,5X,'FR2 ',2E14.5/13X,                    000301
00159         57*          4'NF0 ',I5,10X,'NF1 ',I5,10X,'NF2 ',I5/13X,                              000301
00160         58*          5'PR0 ',2E14.5,5X,'PR1 ',2E14.5,5X,'PR2 ',2E14.5/13X,                    000301
00161         59*          6'NP0 ',I5,10X,'NP1 ',I5,10X,'NP2 ',I5/)                                 000301
```

```
00201    60*          RETURN                                  95110    000301
00202    61*          END                                     95120    000312

END OF COMPILATION:        3 DIAGNOSTICS*
```

SUBROUTINE STNNYQ ENTRY POINT 000033

STORAGE USED: CODE(1) 000041; DATA(0) 000021; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003	KEEP2	000047
0004	KEEP3	000102
0005	KEEP5	000074

EXTERNAL REFERENCES (BLOCK, NAME)

0006	NERR3$

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0005	L	000071	BOTH	0003	D	000002	DHFILT	0004	R	000063	DP	0005	L	000073	GPRINT	0003		000037	HACC
0003		000025	HAST	0003		000035	HATT	0003	D	000000	HBLANK	0003	R	000004	HBLK	0003		000034	HCCW
0003		000033	HCC&	0003		000032	HDEC	0003		000027	HDOT	0003		000006	HESTI	0003		000043	HFGD
0003		000044	HFGN	0003		000040	HFPD	0003		000045	HFPDD	0003		000042	HFPDN	0003		000041	HFPN
0003		000010	HGENE	0003		000031	HINC	0003		000005	HKEY	0003		000007	HMATR	0003		000014	HNEW
0003		000046	HNOMI	0003		000015	HNYQU	0003		000023	HQ	0003		000026	HPLUS	0003		000036	HRATE
0003		000011	HRAW	0003		000012	HRETA	0003		000017	HROLL	0003		000016	HROOT	0003		000013	HSTAN
0003		000030	HSTAR	0003		000020	HSIC	0003		000021	HS2	0003		000022	HS4B	0003		000024	HX
0000		000005	INJP$	0005		000067	ITHZT	0004	R	000051	MAX	0004	R	000037	MIN	0005	L	000072	MODIFY
0004		000000	NFI	0005		000004	NZT	0004		000076	PB	0004	R	000025	PCT	0004		000075	PN
0004		000077	P180	0004		000101	STNDRD	0004	R	000013	STP	0004	R	000001	STR	0005		000000	SUPERK
0005		000003	TD	0004	L	000100	YESNYQ	0005	L	000070	YESZOH	0005		000002	ZN	0005	R	000001	ZT
0005		000005	ZTVAL																

```
00101     1*          SUBROUTINE STNNYQ                                          97040    000000
00102     2*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,  97050    000000
00103     3*         1            HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS4B,  97060    000000
00104     4*          2           HQ,HX,HAST,HPLUS,HDOT,HSTAR,HIOC,HDEC,HCW,HCC&,    97070    000000
00105     5*          3           HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,   97080    000000
00106     6*          4           HNOMI                                              97090    000000
00107     7*          DOUBLE PRECISION HBLANK,DHFILT                                        000000
00108     8*          COMMON/KEEP3/ NFI,STR(10),STP(10),PCT(10),MIN(10),MAX(10),DP(10),  97110    000000
00109     9*         1            FN,PB,P180,YESNYQ,STNDRD                           97120    000000
00110    10*          REAL        MIN,MAX                                            97130    000000
00111    11*          LOGICAL     YESNYQ,STNDRD                                      97140    000000
00112    12*          COMMON/KEEP5/ SUPERK,ZT,ZN,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,     97150    000000
00113    13*         1            MODIFY,GPRINT                                      97160    000000
00114    14*          LOGICAL     YESZOH,BOTH,MODIFY,GPRINT                          97180    000000
00115    15*     C                                                                   97190    000000
00116    16*     C                                                                   97200    000000
00117    17*     C INCORPORATE STANDARD NYQUIST FREQUENCY INTERVAL                   97210    000000
```

```
00111     18*  C
00112     19*       NF1 = 1
00113     20*       STR(1) = .001
00114     21*       STP(1) = 1.0/(2.0*PI)
00115     22*       PCT(1) = 25.0
00116     23*       MIN(1) = 1.0
00117     24*       MAX(1) = 3.0
00120     25*       DP(1) = MILK
00121     26*       RETURN
00122     27*       END
```

```
9722 000000
9723 000000
9724 000001
9725 000003
9726 000010
9727 000012
9728 000014
9729 000016
9730 000020
9731 000040
```

END OF COMPILATION: NO DIAGNOSTICS.

```
FOR.USW F.STNRL,F.STNRL
FOR SE1X-05/23/74-08:49:45-(1,2)

      SUBROUTINE STNRL     ENTRY POINT 000122

   STORAGE USED: CODE(1) 000131; DATA(0) 000034; BLANK COMMON(2) 000000

   COMMON BLOCKS:

      0003   KEEP2  000047
      0004   KEEP4  000263

   EXTERNAL REFERENCES (BLOCK, NAME)

      0005   NERR3S

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000032 IOL       0001   000056 ZOL       0001   000106 3OL       0004 R 000212 BY        0004 R 000166 DD
      0003 D 000002 DHFILT     0004 R 000004 GAIN      0004 R 000151 GSYM      0003   000037 HACC      0003   000025 HAST
      0003   000035 HATT      0003 D 000000 HBLANK    0003 R 000004 HBLK      0003   000034 HCCW      0003   000033 HCW
      0003   000032 HDEC      0003   000027 HDOT      0003   000006 HEST1     0003   000043 HFGD      0003   000044 HFGN
      0003   000040 HFPD      0003   000045 HFPDD     0003   000042 HFPDN     0003   000041 HFPN      0003   000010 HGENE
      0003   000031 HINC      0003   000005 HKEY      0003   000007 HMATR     0003   000014 HNEW      0003   000046 HNOM1
      0003   000015 HNYQU     0003   000023 HO        0003 R 000026 HPLUS     0003   000036 HRATE     0003   000011 HRAN
      0003   000012 HRETA     0003   000017 HROLL     0003   000016 HROOT     0003   000013 HSTAN     0003   000030 HSTAR
      0003   000020 HSIC      0003   000021 HS2       0003   000022 HS4B      0003 R 000024 HX        0000   000016 INJPS
      0004 I 000003 NGAIN     0004 I 000154 NGR       0004 I 000066 NPHASE    0004 I 000153 NRLFR     0004 R 000000 PG
      0004 R 000067 PHASE     0004 R 000001 PP        0004 R 000002 PPLT      0004 R 000152 PSYM      0004 R 000200 RX
      0004 L 000262 YESRL

00101     1*          SUBROUTINE STNRL                                                          97320   000000
00102     2*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HEST1,HMATR,HGENE,HRAN,             97330   000000
00103     3*         1              HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS4B,          97340   000000
00104     4*         2              HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,            97350   000000
00105     5*         3              HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,           97360   000000
00106     6*         4              HNOM1                                                      97370   000000
                                                                                                      000000
00107     7*          DOUBLE PRECISION HBLANK,DHFILT                                            97390   000000
00108     8*          COMMON/KEEP4/ PG,PP,PPLT,NGAIN,GAIN(50),NPHASE,PHASE(50),GSYM,            97400   000000
00109     9*         1              PSYM,NRLFR,NGR(10),DD(10),RX(10),BY(4,10),YESRL            97410   000000
00110    10*          LOGICAL YESRL                                                             97420   000000
00111    11*      C                                                                             97430   000000
00112    12*      C                                                                             97440   000000
00113    13*      C  INCORPORATE STANDARD ROOT LOCUS                                            97450   000000
00114    14*      C                                                                             97460   000000
00115    15*          NGAIN = 0
*DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00116    16*          IF (PG.EQ.HBLK) GO TO 10                                                  97470   000000
00117    17*          NGAIN = 10                                                                97480   000004
00118    18*          GAIN(1) = 0.0                                                             97490   000006
```

```
00114     19*           GAIN(2) = 0.2                                    97500    000007
00115     20*           GAIN(3) = 0.4                                    97510    000011
00116     21*           GAIN(4) = 0.6                                    97520    000013
00117     22*           GAIN(5) = 0.8                                    97530    000015
00120     23*           GAIN(6) = 1.0                                    97540    000017
00121     24*           GAIN(7) = 1.25                                   97550    000021
00122     25*           GAIN(8) = 1.5                                    97560    000023
00123     26*           GAIN(9) = 2.0                                    97570    000025
00124     27*           GAIN(10) = 12345.                                         000027
00125     28*        10 CONTINUE                                         97590    000032
00126     29*           NPHASE = 0                                       97600    000032
00127  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00127     30*           IF (PP.EQ.HBLK) GO TO 20                         97610    000032
00131     31*           NPHASE = 9                                       97620    000036
00132     32*           PHASE(1) = -60.                                  97630    000040
00133     33*           PHASE(2) = -45.                                  97640    000042
00134     34*           PHASE(3) = -30.                                  97650    000044
00135     35*           PHASE(4) = -15.                                  97660    000046
00136     36*           PHASE(5) = 0.0                                   97670    000050
00137     37*           PHASE(6) = 15.                                   97680    000051
00140     38*           PHASE(7) = 30.                                   97690    000052
00141     39*           PHASE(8) = 45.                                   97700    000053
00142     40*           PHASE(9) = 60.                                   97710    000054
00143     41*        20 CONTINUE                                         97720    000056
00144     42*           NRLFR = 0                                        97730    000056
00145  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00145     43*           IF (PPLT.EQ.HBLK) GO TO 30                       97740    000056
00147     44*           GSYM = HX                                        97750    000062
00150     45*           PSYM = H+LUS                                     97760    000064
00151     46*           NRLFR = 2                                        97770    000066
00152     47*           NGR(1) = 1                                       97780    000070
00153     48*           NGR(2) = 1                                       97790    000072
00154     49*           DD(1) = .1                                       97800    000073
00155     50*           DD(2) = .2                                       97810    000075
00156     51*           RX(1) = 1.0                                      97820    000077
00157     52*           RX(2) = 2.0                                      97830    000101
00160     53*           BY(1,1) = -1.0                                   97840    000103
00161     54*           BY(1,2) = -2.0                                   97850    000104
00162     55*        30 CONTINUE                                         97860    000106
00163     56*           RETURN                                           97870    000106
00164     57*           END                                              97880    000130

      END OF COMPILATION:  3 DIAGNOSTICS.
```

SUBROUTINE SUMMRY ENTRY POINT 001063

STORAGE USED: CODE(1) 001074; DATA(0) 000900; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003	KEEP1	000026
0004	KEEP2	000047
0005	KEEP6	000134
0006	CRU01	000015
0007	CRU03	011650

EXTERNAL REFERENCES (BLOCK, NAME)

0010	DB
0011	SYSERR
0012	NWDUS
0013	NIO3$
0014	NIO2$
0015	NIO1$
0016	NERR3$

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001	000072	10L	0001	000731	100L	0001	000734	101L	0001	000751	102L	0001	000756	103L	
0001	000767	104_L	0001	000142	11L	0000	000066	110F	0000	000067	111F	0000	000160	1111F	
0000	000076	112_F	0000	000164	121_F	0000	000105	113_F	0000	000170	1131_F	0000	000115	114_F	
0000	000175	1141_F	0000	000122	115_F	0000	000201	1151_F	0000	000134	116_F	0000	000210	1161_F	
0000	000146	117_F	0000	000217	1171_F	0000	000232	120_F	0001	000015	2L	0001	000150	20L	
0001	000774	200L	0001	000777	210L	0000	000233	211F	0000	000236	222F	0000	000257	223F	
0000	000306	224_F	0000	000334	225_F	0000	000000	3F	0001	000221	30L	0001	000325	40_L	
0000	000006	5F	0001	000373	50L	0001	000474	60_L	0001	000575	70L	0007 R	000070	ADIR	
0007	000006	AERR	0007	011616	AML	0007	011617	AMP	0006 R	000005	AMPONE	0007 R	000152	APHA	
0006 R	000000	A1	0007	011620	BIG	0005	000064	CARD	0010 R	000000	DB	0006 R	000002	DBI	
0006 R	000003	DBZ	0007	011642	DECR	0007	011621	DF	0004 D	000002	DHFILT	0006 R	000004	DPHA	
0007 L	011643	DPI	0007 L	011645	ERAM	0007 L	011647	ERGP	0007 L	011644	ERP	0007 L	011646	ERPH	
0007	011622	FL	0007	011623	FR	0007 C	000000	GAINS	0004	000037	HACC	0004	000025	HAST	
0004	000035	HATT	0004 D	000000	HBLANK	0004 R	000004	HBLK	0004	000034	HCCW	0004	000033	HCW	
0004	000032	HDEC	0004	000027	HDOT	0004	000006	HEST	0004	000043	HFGD	0004	000044	HFGN	
0004	000040	HFPD	0004	000045	HFPDD	0004	000042	HFPDN	0004	000041	HFPN	0004	000010	HGENE	
0004	000031	HINC	0004	000005	HKEY	0004	000007	HMATR	0004	000014	HNEN	0004	000046	HNONI	
0004	000015	HNYQU	0004	000023	HO	0004	000026	HPLUS	0004	000036	HRATE	0004	000011	HRAW	
0004	000012	HRETA	0004	000017	HROLL	0004	000016	HROOT	0004	000013	HSTAN	0004	000030	HSTAR	
0004	000020	HSIC	0004	000021	HS^2	0004	000022	HS^{48}	0004	000024	HX	0007 R	011624	IM	
0000	000361	INJP$	0007	011634	INT	0005	000110	LABEL	0007	011641	LMX	0007	011635	LRPR	
0003	000000	MAXIT	0006 I	000011	MM	0007	011636	MPPP	0006 I	000007	MR	0006 I	000010	MT	
0003	000001	MXEIG	0003	000024	MXEIGT	0003	000023	MXEST	0003	000002	MXFRM	0003	000003	MXNBM	
0003	000004	MXNCOF	0003	000005	MXNCT	0003	000025	MXNCV	0003	000006	MXNE	0003	000007	MXNEQ	
0003	000010	MXNFI	0003	000011	MXNG	0003	000012	MXNPH	0003	000013	MXNPP	0003 I	000014	MXNQPT	
0003	000015	MXNSM	0003 I	000016	MXNSP	0003	000017	MXNTM	0003	000020	MXNV	0003	000021	MXNZT	

```
0003  000022 MXPOLY     0005 R 000062 NAME      0007   011637 NEXT      0007 I 000003 NGNPK     0006 I 000014 NM
0007  011640 NPPP      0007 I 000004 NP180     0006 I 000012 NR        0006 I 000013 NT        0007 I 000005 NYQPTS
0007  000002 NIAMP     0007 R 000316 PAMP      0007 R 000900 PDIR      0007   011625 PER       0007 R 000234 PFRQ
0007  011626 PHA       0007 R 000626 PHAMP     0006 R 000006 PHA180    0007 R 000710 PHDIR     0007 R 000544 PHFRQ
0007  011627 PHL       0007 R 000462 PPHA      0007   011630 RE        0007   003726 SAVAMP    0007   000772 SAVFRQ
0007  006662 SAVPHA    0007   011631 SMA       0007   011632 STA       0007   011633 STO       0005 R 000000 TITLE
0005 R 000024 TITLE1    0005 R 000036 TITLE2    0005 R 000050 TITLE3
```

```
00101     1*        SUBROUTINE SUMRY                                                    9839D  000000
00102     2*          COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBN,MXNCOF,MXNCT,MXNE,MXNEQ,   9840D  000000
00103     3*        1               MXNEI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,   9841D  000000
00104     4*        2               MXNV,MXNZT,MAPOLY,MXEST,MXEIGT,MXNCV                9842D  000000
00105     5*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,     9843D  000000
00106     6*        1               HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HS1C,HS2,HS4B,   9844D  000000
00107     7*        2               HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,     9845D  000000
00108     8*        3               HATT,HRATE,HACC,HFPD,HFPN,HFPDM,HFGD,HFGN,HFPDD,    9846D  000000
00109     9*        4               HNON1                                               9847D  000000
00110    10*          DOUBLE PRECISION HBLANK,DHFILT                                             000000
00111    11*          COMMON/KEEP6/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),  9849D  000000
00112    12*        1               CARD(20),LABEL(20)                                  9850D  000000
00113    13*          REAL          NAME                                                9851D  000000
00114    14*          COMMON/CRUD1/ A1(2),DB1,DR2,DPHA,AMPONE,PHA180,                   9852D  000000
00115    15*        1               HR,HT,HM,NR,NT,NM                                   9853D  000000
00116    16*          COMMON/CRUD3/ GAINS,NIAMP,NGNPK,NP180,NYQPTS,AFRQ(50),ADIR(50),   9854D  000000
00117    17*        1               APHA(50),PFRQ(50),PAMP(50),PDIR(50),PPHA(50),       9855D  000000
00118    18*        2               PHFRQ(50),PHAMP(50),PHDIR(50),SAVFRQ(500),          9856D  000000
00119    19*        3               SAVAMP(1500),SAVPHA(1500),AML,AMP,BIG,DF,FL,FR,IM,  9857D  000000
00120    20*        4               PER,PHA,PHL,RE,SMA,STA,STO,                         9858D  000000
00121    21*        5               INI,LRPR,MPPP,NEXT,NPPP,LMX,                        9859D  000000
00122    22*        6               DECR,DPI,ERP,ERAM,ERPH,ERGP                         9860D  000000
00123    23*          COMPLEX       GAINS                                               9861D  000000
00124    24*          REAL          IM                                                  9862D  000000
00125    25*          LOGICAL       DECR,DPI,ERP,ERAM,ERPH,ERGP                         9863D  000000
00126    26*    C                                                                       9864D  000000
00127    27*    C                                                                       9865D  000000
00128    28*    C     WRITE OUT ERRORS, IF ANY                                          9866D  000000
00129    29*    C                                                                       9867D  000000
00130    30*          IF (ERAM .OR. ERPH .OR. ERGP) GO TO 210                           9868D  000000
00131    31*          IF (ERP .AND. NYQPTS.GT.0) GO TO 210                              9869D  000004
00132    32*    C                                                                       9870D  000004
00133    33*    C     WRITE SUMMARY HEADING                                             9871D  000004
00134    34*    C                                                                       9872D  000004
00135    35*        2 WRITE(6,3) TITLE,TITLE1,TITLE2,TITLE3                             9873D  000015
00136    36*        3 FORMAT('1',8X,20A4//9X,20A4//9X,10A4/)                            9874D  000035
00137    37*          WRITE(6,5)                                                        9875D  000035
00138    38*        5 FORMAT('0',13X,29HU N I T   A M P L I T U D E S,8X,'180 DEG      9876D  000042
00139    39*       1P E A K S',15X,'GAIN                                              9877D  000042
00140    40*       2P E A K S'//39X,5HPHASE/12X,3HCPS,3X,10HAMPLITUDES,2X,6HPHASES,2X,  9878D  000042
00141    41*       36HMARGIN,6X,3HCPS,3X,10HAMPLITUDES,1X,8HDECIBELS,1X,6HPHASES,10X,3  9879D  000042
00142    42*       4HCPS,3X,10HAMPLITUDES,1X,8HDECIBELS,1X,6HPHASES/1H )               9880D  000042
00143    43*    C                                                                       9881D  000042
00144    44*    C     P R I N T   O U T   S U M M A R Y   I N F O R M A T I O N         9882D  000042
00145    45*    C                                                                       9883D  000042
00146    46*    C                                                                       9884D  000042
```

```
00137     47      C                                                                  9885.0    000042
00132     48.     C                                                                  9886.0    000042
00132     49.     C  ON COMPUTER PRINTOUT, THE FIRST FREQUENCY TO APPEAR IS THE REAL  9887.0    000042
00132     50.     C (OR SYSTEM) FREQUENCY.  UNDERNEATH IT, ENCLOSED IN PARENTHESES, IS  9888.0  000042
00132     51.     C  THE OMEGA SUB-R DOMAIN FREQUENCY.                               9889.0    000042
00132     52.     C                                                                  9890.0    000042
00132     53.     C                                                                  9891.0    000042
00132     54.     C                                                                  9892.0    000042
00133     55.           MR = 0                                                       9893.0    000042
00134     56.           MT = 0                                                       9894.0    000043
00135     57.           MM = 0                                                       9895.0    000044
00136     58.           A1(1) = NAME(1)                                              9896.0    000045
00137     59.           A1(2) = NAME(2)                                              9897.0    000047
00140     60.           NR = N1AMP                                                   9898.0    000051
00141     61.           NT = NP180                                                   9899.0    000053
00142     62.           NM = NGNPK                                                   9900.0    000055
00143     63.           AMPONE = 1.0                                                 9901.0    000057
00144     64.           PHA180 = 180.                                                9902.0    000061
00145     65.           WRITE(6,120)                                                 9903.0    000063
00147     66.           GO TO 101                                                    9904.0    000070
00150     67.        10 MM = MM + 1                                                  9905.0    000072
00151     68.           DB1 = DB(PAMP(MM))                                           9906.0    000074
00152     69.           WRITE(6,111) A1,PFRQ(MM),PAMP(MM),DB1,PPHA(MM),PDIR(MM)      9907.0    000103
00162     70.           CALL SYSFRQ(PFRQ(MM),PFRQ(MM))                               9908.0    000122
00163     71.           WRITE(6,1111) PFRQ(MM)                                       9909.0    000132
00166     72.        11 WRITE(6,110)                                                 9910.0    000142
00170     73.           GO TO 100                                                    9911.0    000146
00171     74.        20 MT = MT + 1                                                  9912.0    000150
00172     75.           DB1 = DB(PHAMP(MT))                                          9913.0    000152
00173     76.           WRITE(6,112) A1,PHFRQ(MT),PHAMP(MT),DB1,PHA180,PHDIR(MT)     9914.0    000161
00203     77.           CALL SYSFRQ(PHFRQ(MT),PHFRQ(MT))                             9915.0    000200
00204     78.           WRITE(6,1121) PHFRQ(MT)                                      9916.0    000210
00207     79.           GO TO 11                                                     9917.0    000217
00210     80.        30 MT = MT + 1                                                  9918.0    000221
00211     81.           MM = MM + 1                                                  9919.0    000223
00212     82.           DB1 = DB(PHAMP(MT))                                          9920.0    000227
00213     83.           DB2 = DB(PAMP(MT))                                           9921.0    000235
00214     84.           WRITE(6,113) A1,PHFRQ(MT),PHAMP(MT),DB1,PHA180,PHDIR(MT),    9922.0    000244
00216     85.          1             PFRQ(MM),PAMP(MM),DB2,PPHA(MM),PDIR(MM)       9923.0    000244
00231     86.           CALL SYSFRQ(PHFRQ(MM),PHFRQ(MM))                             9924.0    000271
00232     87.           CALL SYSFRQ(PFRQ(MT),PFRQ(MT))                               9925.0    000301
00233     88.           WRITE(6,1131) PHFRQ(MM),PFRQ(MT)                             9926.0    000312
00237     89.           GO TO 11                                                     9927.0    000323
00240     90.        40 MR = MR + 1                                                  9928.0    000325
00241     91.           DPHA = ABS(APHA(MR) - 180.)                                  9929.0    000327
00242     92.           WRITE(6,114) A1,AFRQ(MR),AMPONE,ADIR(MR),APHA(MR),DPHA       9930.0    000334
00252     93.           CALL SYSFRQ(AFRQ(MR),AFRQ(MR))                               9931.0    000352
00253     94.           WRITE(6,1141) AFRQ(MR)                                       9932.0    000362
00256     95.           GO TO 11                                                     9933.0    000371
00257     96.        50 MR = MR + 1                                                  9934.0    000373
00260     97.           MM = MM + 1                                                  9935.0    000375
00261     98.           DB1 = DB(PAMP(MM))                                           9936.0    000400
00262     99.           DPHA = ABS(APHA(MR) - 180.)                                  9937.0    000407
00263    100.           WRITE(6,115) A1,AFRQ(MR),AMPONE,ADIR(MR),APHA(MR),DPHA,      9938.0    000415
00263    101.          1             PFRQ(MM),PAMP(MM),DB1,PPHA(MM),PDIR(MM)       9939.0    000415
00300    102.           CALL SYSFRQ(AFRQ(MR),AFRQ(MR))                               9940.0    000440
00301    103.           CALL SYSFRQ(PFRQ(MM),PFRQ(MM))                               9941.0    000450
```

```
00302     1          WRITE(6,1151) AFRQ(MR),PFRQ(MM)                                  99420   000461
00306   105*         GO TO 11                                                         99430   000472
00307   106*     60  MR = MR + 1                                                      99440   000474
00310   107*         MT = MT + 1                                                      99450   000476
00311   108*         DB1 = DB(PHAMP(MT))                                              99460   000501
00312   109*         DPHA = ABS(APHA(MR) - 180.)                                      99470   000510
00313   110*         WRITE(6,116) A1,AFRQ(MR),AMPONE,ADIR(MR),APHA(MR),DPHA,          99480   000516
00313   111*        1                PHFRQ(MT),PHAMP(MT),DB1,PHA180,PHDIR(MT)       99490   000516
00331   112*         CALL SYSFRQ(AFRQ(MR),AFRQ(MR))                                   99500   000541
00332   113*         CALL SYSFRQ(PHFRQ(MT),PHFRQ(MT))                                 99510   000561
00333   114*         WRITE(6,1161) AFRQ(MR),PHFRQ(MT)                                 99520   000562
00336   115*         GO TO 11                                                         99530   000573
00337   116*     70  MR = MR + 1                                                      99540   000575
00340   117*         MT = MT + 1                                                      99550   000577
00341   118*         MM = MM + 1                                                      99560   000602
00342   119*         DB1 = DB(PHAMP(MT))                                              99570   000606
00343   120*         DB2 = DB(PAMP(MM))                                               99580   000614
00344   121*         DPHA = ABS(APHA(MR) - 180)                                       99590   000623
00345   122*         WRITE(6,117) A1,AFRQ(MR),AMPONE,ADIR(MR),APHA(MR),DPHA,          99600   000632
00345   123*        1                PHFRQ(MT),PHAMP(MT),DB1,PHA180,PHDIR(MT),      99610   000632
00346   124*        2                PFRQ(MM),PAMP(MM),DB2,PPHA(MM),PDIR(MM)        99620   000632
00347   125*         CALL SYSFRQ(AFRQ(MR),AFRQ(MR))                                   99630   000662
00370   126*         CALL SYSFRQ(PHFRQ(MT),PHFRQ(MT))                                 99640   000672
00371   127*         CALL SYSFRQ(PFRQ(MM),PFRQ(MM))                                   99650   000703
00372   128*         WRITE(6,1171) AFRQ(MR),PHFRQ(MT),PFRQ(MM)                        99660   000714
00377   129*         GO TO 11                                                         99670   000727
00377   130*     C                                                                    99680   000727
00377   131*     C   DETERMINE FORMAT OF NEXT LINE                                    99690   000727
00377   132*     C                                                                    99700   000727
00400   133*   100 CONTINUE                                                         99710   000731
00400   134*         A1(1) = HOLK                                                     99720   000731
00402   135*         A1(2) = HOLK                                                     99730   000732
00403   136*   101 IF (MR.NE.NR) GO TO 103                                          99740   000734
00407   137*         IF (MT.NE.NT) GO TO 102                                          99750   000737
00407   138*         IF (MM.NE.NM) GO TO 10                                           99760   000743
00411   139*         GO TO 200                                                        99770   000747
00412   140*   102 IF (MM.EQ.NM) GO TO 20                                           99780   000751
00414   141*         GO TO 30                                                         99790   000754
00415   142*   103 IF (MT.NE.NT) GO TO 104                                          99800   000756
00417   143*         IF (MM.EQ.NM) GO TO 40                                           99810   000761
00421   144*         GO TO 50                                                         99820   000765
00422   145*   104 IF (MM.EQ.NM) GO TO 60                                           99830   000767
00424   146*         GO TO 70                                                         99840   000772
00425   147*   110 FORMAT (1H )                                                     99850   000774
00425   148*   111 FORMAT(1X,2A4,79X,          F10.4,1PE12.4,2(0PF8.2),1XA3)        99860   000774
00427   149*   112 FORMAT(1X,2A4,37X,          F10.4,1PE12.4,2(0PF8.2),1XA3)        99870   000774
00431   150*   113 FORMAT(1X,2A4,37X,  2(F10.4,1PE12.4,2(0PF8.2),1XA3))             99880   000774
00431   151*   114 FORMAT(1X,2A4,1X2F8.4,1XA3,2F8.2)                                99890   000774
00432   152*   115 FORMAT(1X,2A4,1X2F8.4,1XA3,2F8.2,42X,F10.4,1PE12.4,2(0PF8.2),1XA3) 99900   000774
00433   153*   116 FORMAT(1X,2A4,1X2F8.4,1XA3,2F8.2,  F10.4,1PE12.4,2(0PF8.2),1XA3) 99910   000774
00434   154*   117 FORMAT(1X,2A4,1X2F8.4,1XA3,2F8.2, 2(F10.4,1PE12.4,2(0PF8.2),1XA3)) 99920   000774
00435   155*  1111 FORMAT(89X,'(',F8.4,')')                                         99930   000774
00436   156*  1121 FORMAT(47X,'(',F8.4,')')                                         99940   000774
00437   157*  1131 FORMAT(47X,2('(',F8.4,')',3X))                                   99950   000774
00440   158*  1141 FORMAT(9X,'(',F8.4,')')                                          99960   000774
00441   159*  1151 FORMAT(9X,'(',F8.4,')',79X,'(',F8.4,')')                         99970   000774
00442   160*  1161 FORMAT(9X,'(',F8.4,')',29X,'(',F8.4,')')                         99980   000774
```

```
00443   161*      1171 FORMAT(9X,'(',F8.4,')',28X,'(',F8.4,')',32X,'(',F8.4,')')                    9999₀   000774
00444   162*       120 FORMAT (/1H )                                                                 10000₀   000774
00445   163*       200 CONTINUE                                                                      10001₀   000774
00446   164*           RETURN                                                                        10002₀   000774
00445   165*     C                                                                                   10003₀   000774
00445   166*     C     ERROR MESSAGES FROM FREQUENCY RESPONSE COMPUTATIONS                           10004₀   000774
00445   167*     C                                                                                   10005₀   000774
00447   168*           210 WRITE(6,211)                                                              10006₀   000777
00451   169*           211 FORMAT (1H1,4(/),1H )                                                     10007₀   001003
00452   170*               IF (ERA) WRITE(6,222) MXNSP                                               10008₀   001003
00455   171*               IF (ERPH) WRITE(6,223) MXNSP                                              10009₀   001013
00462   172*               IF (ERGP) WRITE(6,224) MXNSP                                              10010₀   001023
00465   173*               IF (ERP) WRITE(6,225) MXNQPT                                              10011₀   001033
00472   174*               GO TO 2                                                                   10012₀   001043
00473   175*           222 FORMAT (2(/),20X,64HCAPACITY FOR STORAGE OF UNIT AMPLITUDES EXCEED        10013₀   001073
00473   176*          1ED. ONLY THE FIRST,I3,17H WILL BE PRINTED.)                                   10014₀   001073
00474   177*           223 FORMAT (2(/),20X,66HCAPACITY FOR STORAGE OF 180 DEGREE PHASES EXCE        10015₀   001073
00474   178*          1EDED. ONLY THE FIRST,I3,' WITH AMPLITUDE LESS THAN 5.0E-4 WILL BE            10016₀   001073
00474   179*          2PRINTED.')                                                                    10017₀   001073
00475   180*           224 FORMAT (2(/),20X,59HCAPACITY FOR STORAGE OF GAIN PEAKS EXCEEDED. O       10018₀   001073
00475   181*          1NLY THE FIRST,I3,' WITH AMPLITUDE LESS THAN 5.0E-4 WILL BE PRINTED           10019₀   001073
00475   182*          2.')                                                                           10020₀   001073
00475   183*           225 FORMAT(2(/),20X,'CAPACITY FOR STORAGE OF NYQUIST POINTS EXCEEDED.         10021₀   001073
00475   184*          1  ONLY THE FIRST ',I4,' WILL BE PLOTTED.')                                    10022₀   001073
00477   185*               END                                                                       10023₀   001073

      END OF COMPILATION:      NO DIAGNOSTICS.
```

```
@FOR,USY F.SYSFRQ,F.SYSFRQ
FOR 3E1X-05/23/74-08:50:39 (1,2)

      SUBROUTINE SYSFRQ     ENTRY POINT 00002₀

      STORAGE USED: CODE(1) 0₀0024; DATA(0) 000007; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003    KEEPS  000074

      EXTERNAL REFERENCES (BLOCK, NAME)

      0004    COTAN
      0005    NERR35

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0003 L 000071 BOTH      0003 L 000073 GPRINT    0000   000003 INJPS     0003   000067 ITHZT     0003 L 000072 MODIFY
      0003   000004 NZT       0000 R 000000 RADIAN    0003   000000 SUPERK    0003   000003 TD        0003 L 000070 YESZOH
      0003   000002 ZM        0003 R 000001 ZT        0003   000005 ZTVAL
00101     1*          SUBROUTINE SYSFRQ(FRIN,FROUT)                                          10024₀  000000
00103     2*          COMMON/KEEPS/ SUPERK,ZJ,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,         10025₀  000000
00103     3*         1            MODIFY,GPRINT                                            10026₀  000000
00104     4*          LOGICAL      YESZOH,BOTH,MODIFY,GPRINT                                10028₀  000000
00104     5*      C                                                                         10029₀  000000
00104     6*      C                                                                         10030₀  000000
00104     7*      C  PROGRAM CODING                                                         10031₀  000000
00104     8*      C                                                                         10032₀  000000
00105     9*          RADIAN = 3.1415927 * ZT * FRIN                                        10033₀  000000
00106    10*          FROUT  = COTAN(RADIAN)/ 6.28318531                                    10034₀  000003
00107    11*          RETURN                                                                10035₀  000010
00110    12*          END                                                                   10036₀  000023

      END OF COMPILATION:      NO  DIAGNOSTICS.
```

```
@FOR,USW F.TEST,F.TEST
FOR SEIX-05/23/74-08:50:48 (1.2)

      SUBROUTINE TEST       ENTRY POINT 000223

   STORAGE USED: CODE(1) 000226; DATA(0) 000020; BLANK COMMON(2) 000000

   COMMON BLOCKS:

      0003   KEEP14 000031
      0004   CRUD3  011650

   EXTERNAL REFERENCES (BLOCK, NAME)

      0005   SQRT
      0006   ATAN2
      0007   NERR35

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001    000024 1L          0001    000130 11L         0001    000041 2L          0001    000202 21L         0001    000210 22L
      0004    000070 ADIR        0004    000006 AFRQ        0004    011616 AML         0004 R  011617 AMP         0004    000152 APHA
      0004 R  011620 BIG         0004 L  011642 DECR        0003 R  000025 DEG         0004 R  011621 DF          0000 R  000001 DIFF
      0004 L  011643 DPI         0003    000012 D0          0003    000013 D1          0003    000014 D2          0003    000015 D3
      0003    000016 D4          0004 L  011645 ERAM        0004 L  011647 ERGP        0004 L  011644 ERP         0004 L  011646 ERPH
      0003 C  000010 FIFTEN      0003    000017 FIFTY       0004 R  011622 FL          0003 C  000006 FOUR        0004 R  011623 FR
      0004 C  000000 GAINS       0003 C  000000 HALF        0004 R  011624 IM          0000    000011 INJPS       0004    011634 INT
      0004    011641 LMX         0004    011635 LRPR        0004    011636 MPPP        0004 I  011637 NEXT        0004    000003 NGNPK
      0004    011640 NPPP        0004    000004 NPI80       0004    000005 NYQPTS      0003    000026 N1          0004    000002 NIAMP
      0003    000027 N2          0003    000030 N3          0003 C  000002 ONE         0004    000316 PAMP        0004    000400 PDIR
      0004 R  011625 PER         0004    000234 PFRQ        0004 R  011626 PHA         0004    000626 PHAMP       0004    000710 PHDIR
      0004    000544 PHFRQ       0004 R  011627 PHL         0003    000020 PI          0003    000021 PI2         0004    000462 PPHA
      0003    000024 RADDEG      0004 R  011630 RE          0003    000023 RPI         0004    003726 SAVAMP      0004    000772 SAVFRQ
      0004    006662 SAVPHA      0004 R  011631 SMA         0003    000022 SMALL       0004 R  011632 STA         0004    011633 STO
      0000 R  000002 TEST        0000 L  000000 TOOFIN      0003 C  000004 TWO

6-209

00101    1*          SUBROUTINE TEST                                                          100370   000000
00103    2*          COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,          100380   000000
00103    3*         1  PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                                     100390   000000
00104    4*          COMPLEX        HALF,ONE,TWO,FOUR,FIFTEN                                          000000
00105    5*          COMMON/CRUD3/ GAINS,NIAMP,NGNPK,NPI80,NYQPTS,AFRQ(50),ADIR(50),          100420   000000
00105    6*         1  APHA(50),PFRQ(50),PAMP(50),PDIR(50),PPHA(50),                         100430   000000
00105    7*         2  PHFRQ(50),PHAMP(50),PHDIR(50),SAVFRQ(1500),                           100440   000000
00105    8*         3  SAVAMP(1500),SAVPHA(1500),AML,AMP,BIG,DF,FL,FR,IM,                    100450   000000
00105    9*         4  PER,PHA,PHL,RE,SMA,STA,STO,                                           100460   000000
00105   10*         5  INT,LRPR,MPPP,NEXT,NPPP,LMX,                                          100470   000000
00105   11*         6  DECR,DPI,ERP,ERAM,ERPH,ERGP                                           100480   000000
00105   12*          COMPLEX        GAINS                                                    100490   000000
00107   13*          REAL           IM                                                       100500   000000
00111   14*          LOGICAL        DECR,DPI,ERP,ERAM,ERPH,ERGP                              100510   000000
```

```
00111    15*          LOGICAL       TOOFIN                                  10052₀  000000
00111    16*     C                                                          10053₀  000000
00111    17*     C  PROGRAM CODING                                          10054₀  000000
00111    18*     C                                                          10055₀  000000
00111    19*     C                                                          10056₀  000000
00111    20*     C                                                          10057₀  000000
00111    21*     C     SET STANDARD RETURN, COMPUTE AMPLITUDES AND PHASES   10058₀  000000
00111    22*     C                                                          10059₀  000000
00112    23*           NEXT = 2                                             10060₀  000000
00113    24*           RE = REAL(GAINS)                                     10061₀  000001
00114    25*           IM = AIMAG(GAINS)                                    10062₀  000003
00115    26*           AMP=SQRT(RE**2 + IM**2)                              10063₀  000005
00115  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00115    27*           IF(AMP.NE.0.) GO TO 1                                10064₀  000017
00120    28*           PHA = 0.                                             10065₀  000021
00121    29*           GO TO 2                                              10066₀  000022
00122    30*         1 PHA = DEG * ATAN2(IM,RE)                             10067₀  000024
00123    31*           IF(PHA .LT. 0.) PHA = PHA + 360.0                    10068₀  000031
00126    32*         2 CONTINUE                                             10069₀  000041
00126    33*     C                                                          10070₀  000041
00126    34*     C     CHECK PHASE SHIFTS                                   10071₀  000041
00126    35*     C                                                          10072₀  000041
00126  *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00126    36*           IF (FR.EQ.STA)  RETURN                               10073₀  000041
00131    37*           TOOFIN = .TRUE.                                      10074₀  000047
00131    38*           DIFF = ABS(PHA - PHL)                                10075₀  000051
00132    39*           DIFF = AMIN1(DIFF,ABS(360.-DIFF))                    10076₀  000055
00133    40*           IF (DIFF.GT.SMA)  TOOFIN = .FALSE.                   10077₀  000064
00135    41*         3 IF (DIFF.GT.BIG)  GO TO 11                           10078₀  000072
00137    42*           IF (.NOT.TOOFIN .OR. DECR)  RETURN                   10079₀  000077
00137    43*     C                                                          10080₀  000077
00137    44*     C     INCREMENT TOO SMALL                                  10081₀  000077
00137    45*     C                                                          10082₀  000077
00140    46*           DF = 2.*DF                                           10083₀  000105
00143    47*           IF(ABS(DF).GT.(PER*FR)) GO TO 21                     10084₀  000110
00144    48*           FR = FL + DF                                         10085₀  000117
00145    49*           NEXT = 1                                             10086₀  000122
00146    50*           RETURN                                               10087₀  000124
00146    51*     C                                                          10088₀  000124
00146    52*     C     INCREMENT TOO LARGE                                  10089₀  000124
00146    53*     C                                                          10090₀  000124
00147    54*        11 DF = DF/2.                                           10091₀  000130
00150    55*           DECR = .TRUE.                                        10092₀  000132
00151    56*           TEST = .0002                                         10093₀  000134
00152    57*           DIFF = ABS(PHA - PHL)                                10094₀  000136
00153    58*           IF(DIFF.GT.BIG .AND. AMP .GT. .1)  TEST = .00002     10095₀  000142
00156    59*           IF(ABS(DF).LT.(TEST*FR)) GO TO 22                    10096₀  000162
00157    60*           FR = FL + DF                                         10097₀  000171
00160    61*           NEXT = 1                                             10098₀  000174
00161    62*           RETURN                                               10099₀  000176
00161    63*     C                                                          10100₀  000176
00161    64*     C     INCREMENT MUST REMAIN WITHIN LIMITS                  10101₀  000176
00161    65*     C                                                          10102₀  000176
00162    66*        21 DF = DF/2.                                           10103₀  000202
00163    67*           RETURN                                               10104₀  000204
00164    68*        22 DF = 2.*DF                                           10105₀  000210
00165    69*           RETURN                                               10106₀  000212
```

00164 END 10107g 000225

END OF COMPILATION: 2 DIAGNOSTICS.

```
*FOR,US# [redacted]NSFR,F*TRNSFR
FOR SEIX-05/23/74-08:51:05 (1,2)

      SUBROUTINE TRNSFR     ENTRY POINT 00007₀

      STORAGE USED: CODE(1) 000074; DATA(0) 000020; BLANK COMMON(2) 000000

      COMMON BLOCKS:

 0003   KEEP2   000047
 0004   KEEP3   000102
 0005   CRUD2   001215
 0006   CRUD3   011650

      EXTERNAL REFERENCES (BLOCK, NAME)

 0007   POLVAL
 0010   COV$
 0011   NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

 0001   000011 122₈      0006   00007₀ ADIR      0006   000006 AFOO      0006   011616 AML       0006   011617 AMP
 0006   000152 APHA      0006   011620 BIG       0004 L 011642 DECR      0006   011621 DF        0003 D 000002 DHFILT
 0004   000063 DP        0006 L 011643 DPI       0006 L 011645 ERAM      0006 L 011647 ERGP      0006 L 011644 ERP
 0006 L 011646 ERPH      0006   011622 FL        0006   011623 FR        0006 C 000000 GAINS     0003   000037 HACC
 0003   000025 HAST      0003   00003₅ HATT      0003 D 000000 HBLANK    0003 R 000004 HBLK      0003   000034 HCCW
 0003   000033 HCW       0003   00003₂ HDEC      0003   000027 HDOT      0003   000006 HESTI     0003   000043 HFGD
 0003   000044 HFGN      0003   000040 HFPD      0003   000045 HFPDD     0003   000042 HFPDN     0003   000041 HFPN
 0003   000010 HGENE     0003   00003₁ HINC      0003   000005 HKEY      0003   000007 HMATR     0003   000014 HNEW
 0003   000046 HNOMI     0003   000015 HNYQU     0003   000023 HO        0003   000026 HPLUS     0003   000036 HRATE
 0003   000011 HRAW      0003   000012 HRETA     0003   000017 HROLL     0003   000016 HROOT     0003   000013 HSTAN
 0003   00003₀ HSTAR     0003   000020 HS1C      0003   000021 HS2       0003   000022 HS4B      0003   000024 HX
 0000 I 000006 I         0006 R 011624 IM        0000   000012 INJPS     0006   011634 INT       0006   011641 LMX
 0005   001214 LOCD      0000 I 000005 LOCDEN     0005   001213 LOCN      0000 I 000004 LOCNUM    0006   011635 LRPR
 0004 R 000051 MAX       0004 R 00003₇ MIN       0006   011636 MPPP      0005 I 001077 NDCPER    0006   011637 NEXT
 0004   000000 NFI       0006   000003 NGNPK     0005 * 000764 NNCPER    0006   011640 NPPP      0006   000004 NP180
 0005 I 001212 NUMPOL    0006   000005 NYQPTS    0004   000002 NIAMP     0006   000316 PAMP      0004   000076 PB
 0004   000025 PCT       0005 R 00031₀ PD        0004   000400 PDIR      0006   011625 PER       0006   000234 PFRQ
 0006   011626 PHA       0006   000526 PHAMP     0006   000710 PHDIR     0006   000544 PHFRQ     0006   011627 PHL
 0004   000075 PN        0005 R 000000 PNN       0004   000462 PPHA      0004 R 000077 P180      0006   011630 RE
 0006   003726 SAVAMP    0006   000772 SAVERQ    0004   006662 SAVPHA    0006   011631 SMA       0006   011632 STA
 0004 L 000101 STNDRD    0006   011633 STO       0004   000013 STP       0004   000001 STR       0000 C 000000 VALD
 0000 C 000002 VALN      0004 L 000100 YESNYQ

 00101      1*        SUBROUTINE TRNSFR
 00102      2*        COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,    101008  000000
 00102      3*    1             HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HS1C,HS2,HS4B,    101009  000000
 00103      4*    2             HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,      101010  000000
 00103      5*    3             HATT,HRATE,HACC,HFPD,HFPN,HFPDD,HFGD,HFGN,HFPDD,     101011  000000
```

```
00103                 *
00104                 *
00105           8*    
00106           9*    
00107           10*   
00110           11*   
00111           12*   
00112           13*   
00113           14*   
00114           15*   
00115           16*   
00116           17*   
00117           18*   
00120           19*   
00121           20*   
00122           21*   
00123           22*   
00124           23*   
00125           24*   
00126           25*   
00127           26*   
00130           27*   
00131           28*   
00132           29*   
00133           30*   
00134           31*   
00135           32*   
00136           33*   
00137           34*   
00140           35*   
00141           36*   
00142           37*   
00143           38*   
00144           39*   
```

```
      4                 HNOMI
      DOUBLE PRECISION HBLANK,DHFILT                                    101130    000000
      COMMON/KEEP3/ NFI,STR(10),STP(10),PCT(10),MIN(10),MAX(10),DP(10),  101150    000000
     1              PN,PD,P180,YESNYQ,STNDRD                            101160    000000
      REAL          MIN,MAX                                             101170    000000
      LOGICAL       YESNYQ,STNDRD                                       101180    000000
      COMMON/CPUD2/ PNN(200),PD(300),NNCPER(75),NDCPER(75),                       000000
     1              NUMPOL,LOCN,LOCD                                    101200    000000
      COMMON/CPUD3/ GAINS,NIAMP,NGNPK,NP180,NYQPTS,AFRQ(50),ADIR(50),   101220    000000
     1              APHA(50),PFRQ(50),PAMP(50),PDIR(50),PPHA(50),       101230    000000
     2              PHFRQ(50),PHAMP(50),PHDIR(50),SAVFRQ(1500),         101240    000000
     3              SAVAMP(1500),SAVPHA(1500),AML,AMP,BIG,DF,FL,FR,IM,  101250    000000
     4              PER,PHA,PHL,RF,SMA,STA,STO,                         101260    000000
     5              INT,LPPP,MPPP,NEXT,NPPP,LMX,                        101270    000000
     6              DECR,DPI,ERP,ERAM,ERPH,ERGP                         101280    000000
      COMPLEX       GAINS                                               101290    000000
      REAL          IM                                                  101300    000000
      LOGICAL       DECR,DPI,ERP,ERAM,ERPH,ERGP                         101310    000000
      COMPLEX       VALD,VALN                                           101320    000000
C                                                                       101330    000000
C  PROGRAM CODING                                                       101340    000000
C                                                                       101350    000000
      GAINS = (0.0,0.0)                                                 101360    000000
      LOCNUM = 1                                                        101370    000001
      LOCDEN = 1                                                        101380    000003
      DO 1 I = 1, NUMPOL                                                101390    000011
      CALL POLVAL (LOCNUM,NNCPER(I),PNN,VALN)                           101400    000011
      CALL POLVAL (LOCDEN,NDCPER(I),PD,VALD)                            101410    000020
      GAINS = GAINS * VALN/VALD                                         101420    000030
      LOCNUM = LOCNUM + NNCPER(I)                                       101430    000041
    1 LOCDEN = LOCDEN + NDCPER(I)                                       101440    000044
*DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
      IF (P180.NE.HBLK) GAINS=-GAINS                                    101450    000050
      RETURN                                                            101460    000056
      END                                                               101470    000073
```

END OF COMPILATION; 1 DIAGNOSTICS.

```
>FOR,USK F   YGEN,F*VRYGEN
FOR $E1X=05/23/74-08:51:37 (1,2)

      SUBROUTINE VRYGEN     ENTRY POINT 000226

   STORAGE USED: CODE(1) 000240; DATA(0) 000031; BLANK COMMON(2) 000000

   (COMMON BLOCKS:

   (003   KEEP1  000026
   (004   KEEP10 021620
   (005   KEEP13 000313
   (006   KEEP16 000031
   (007   CRUD2  000115

   EXTERNAL REFERENCES (BLOCK, NAME)

   (010   NRDUS
   (011   NIO2S
   (012   NIO1S
   (013   NERR4S
   (014   NEPR3S

   STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

(001  000032 1OL      0001  000160 1000L    0001  000040 125G     0001  000051 136G     0001  000166 2000L
(001  000174 3000L    0001  000202 4000L    0000  000007 500F     0000  000010 501F     0007  000024 ANORM
(007  000000 BCD      0006 L 000022 DEBUG    0004 R 016664 EV      0000 I 000002 I       0007 I 000075 II
(000  000015 INJP$    0006  000010 IOPEN     0004  000004 IR      0000 I 000000 ITHNV     0000 I 000003 J
(004  001754 JC       0007 I 000101 JJ       0006  000011 JOPEN    0000 I 000004 K       0007 I 000105 KK
(006 I 000001 KODE     0000 I 000005 L        0004 L 000023 LFLT    0004 I 005674 LL      0004 I 007644 LOCPOL
(005 I 000144 LOCV     0000 I 000001 M        0003  000000 MAXIT    0003  000001 MXEIG    0003  000024 MXEIGT
(003  000023 MXEST    0003  000002 MXFRM     0003  000003 MXNBM    0003  000004 MXNCOF   0003  000005 MXNCT
(003  000025 MXNCV    0003  000006 MXNE      0003  000007 MXNEQ    0003  000010 MXNFI    0003  000011 MXNG
(003  000012 MXNPH    0003  000013 MXNPP     0003  000014 MXNQPT   0003  000015 MXNSM    0003  000016 MXNSP
(003  000017 MXNTM    0003 I 000020 MXNV     0003  000021 MXNZT    0003  000022 MXPOLY   0000 I 000006 N
(004  000003 NCOF     0005  000312 NCOFV    0004 I 003724 ND      0004  000000 NDEG     0004  000002 NE
(004 I 000001 NEQ      0005  000311 NEV      0007  000074 NFILT    0006 L 000027 NOMNAL   0006 L 000030 NOTYET
(006  000002 NPCLPL   0006  000003 NRPOLE   0006  000004 NRZERO   0007  000071 NTMPO    0007  000072 NTMPOC
(007  000073 NTMPIC   0005 I 000310 NV       0006  000005 NXB      0006  000006 NXN      0006  000007 NXR
(007  000020 OPTINP   0007  000003 OPTTYP   0007  000021 OPT1     0007  000022 OPT2     0007  000023 OPT3
(007  000111 PC       0006  000014 PCPL     0006  000015 PEAC     0006  000013 PNOM     0006  000016 PSLOSH
(006  000012 PVAR     0007  000017 REQEST   0004  000000 STAGE    0005 R 000000 SV      0007  000006 TEMP
(007  000025 TEMP0    0007  000045 TEMP1    0007 R 000065 VAL      0007  000011 VFILT    0006 L 000017 YESMTX
(006  000024 YESPCH   0006 L 000020 YESRAN   0006 L 000025 YESRLP   0006 L 000026 YESSRL   0006 L 000021 YESSRP

00101      1*          SUBROUTINE VRYGEN(*,*)                                          101480   000000
00102      2*          COMMON/KEEP1/ MAXIT,MXEIG,MXFRM,MXNBM,MXNCOF,MXNCT,MXNE,MXNEQ,  101490   000000
00103      3*        1             MXNFI,MXNG,MXNPH,MXNPP,MXNQPT,MXNSM,MXNSP,MXNTM,  101500   000000
00104      4*        2             MXNV,MXNZT,MXPOLY,MXEST,MXEIGT,MXNCV              101510   000000
```

```
00109          COMMON/KEEP10/NDEG,NEQ,NE,NCOF,IR(1000),JC(1000),ND(1000),
     6*       1          LL(1000),LOCPOL(60,20),EV(1500)
00105  7*      COMMON/KEEP13/SV(100),LOCV(100),NV,NEV,NCOFV
00106  8*      COMMON/KEEP16/STAGE,KODE,NRGLPL,NRPOLE,NRZERO,NXD,NXN,NXR,
00107  9*      1          IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,
00108 10*      2          YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,
00109 11*      3          YESSRL,NOMNAL,NOTYET
00110 12*      LOGICAL    YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,
00111 13*      1          YESSRL,NOMNAL,NOTYET
00112 14*      COMMON/CROD2/ NCD(3),OPTYP(3),TEMP(3),VFILT(6),REQEST,OPTINP,
00113 15*      1          OPT1,OPT2,OPT3,ANORM,TEMP0(16),TEMP1(16),VAL(4),
00114 16*      2          NTMP0,NTMP0C,NTMP1C,NFILT,II(4),JJ(4),KK(4),PC(4)
00115 17*      C
00116 18*      C
00117 19*      C  PARAMETER VARIATIONS IN GENERAL FORM
00118 20*      C
00119 21*      C
00120 22*      C
00121 23*      C  READ VARIATION IN GENERAL FORMAT
00122 24*      C
00123 25*            ITHNV = 0
00124 26*            READ(5,500) NV
00125 27*        500 FORMAT(I5)
00126 28*            IF (NV.EQ.0) RETURN 2
00127 29*            IF (NV.LT.0 .OR. NV.GT.MXNV) GO TO 1000
00128 30*         10 CONTINUE
00129 31*            READ(5,501) (II(M),JJ(M),KK(M),VAL(M),M=1,4)
00130 32*        501 FORMAT(4(3I2,E14.6))
00131 33*            DO 20 M=1,4
00132 34*            I = II(M)
00133 35*            J = JJ(M)
00134 36*            K = KK(M) + 1
00135 37*            IF (I.LT.1 .OR. I.GT.NEQ) GO TO 2000
00136 38*            IF (J.LT.1 .OR. J.GT.NEQ) GO TO 2000
00137 39*            L = LOCPOL(I,J)
00138 40*            IF (L.EQ.0) GO TO 3000
00139 41*            IF (K.GT.ND(L)) GO TO 4000
00140 42*            N = LL(L) + K - 1
00141 43*            ITHNV = ITHNV + 1
00142 44*            LOCV(ITHNV) = N
00143 45*            SV(ITHNV) = EV(N)
00144 46*            EV(N) = VAL(M)
00145 47*            IF (ITHNV.GE.NV) RETURN
00146 48*         20 CONTINUE
00147 49*            GO TO 10
00148 50*      C
00149 51*      C
00150 52*      C  NUMBER OF VARIATIONS OUT OF RANGE
00151 53*      C
00152 54*      1000 CONTINUE
00153 55*            KODE = 61
00154 56*            RETURN 1
00155 57*      C
00156 58*      C
00157 59*      C  VARIED NON-EXISTENT ELEMENT
00158 60*      C
00159 61*        2000 CONTINUE
```

```
10152₀  000000
10153₀  000000
10154₀  000000
10155₀  000000
10156₀  000000
10157₀  000000
10158₀  000000
10159₀  000000
10160₀  000000
10161₀  000000
10162₀  000000
10163₀  000000
10165₀  000000
10166₀  000000
10167₀  000000
10168₀  000000
10169₀  000000
10170₀  000000
10171₀  000000
10172₀  000000
10173₀  000000
10174₀  000000
10175₀  000006
10176₀  000006
10177₀  000014
10178₀  000032
10179₀  000032
10180₀  000051
10181₀  000051
10182₀  000051
10183₀  000053
10184₀  000055
10185₀  000060
10186₀  000075
10187₀  000112
10188₀  000117
10189₀  000121
10190₀  000126
10191₀  000133
10192₀  000137
10193₀  000140
10194₀  000142
10195₀  000144
10196₀  000156
10197₀  000156
10198₀  000156
10199₀  000156
10200₀  000156
10201₀  000156
10202₀  000160
10203₀  000160
10204₀  000161
10205₀  000161
10206₀  000161
10207₀  000161
10208₀  000161
10209₀  000166
```

```
00172      62            KODE = 62
00173      63*           RETURN 1
00173      64*     C
00173      65*     C
00173      66*     C  VARIED ELEMENT OUTSIDE MATRIX DIMENSION
00173      67*     C
00174      69*      3000 CONTINUE
00175      69*           KODE = 63
00175      70*           RETURN 1
00175      71*     C
00175      72*     C
00175      73*     C  VARIED NON-EXISTENT COEFFICIENT
00175      74*     C
00177      75*      4000 CONTINUE
00200      76*           KODE = 64
00201      77*           RETURN 1
00202      79*           END
```

```
10210₀  000166
10211₀  000167
10212₀  000167
10213₀  000167
10214₀  000167
10215₀  000167
10216₀  000174
10217₀  000174
10218₀  000175
10219₀  000175
10220₀  000175
10221₀  000175
10222₀  000175
10223₀  000202
10224₀  000202
10225₀  000203
10226₀  000237
```

END OF COMPILATION: NO DIAGNOSTICS.

```
$FOR,USR F,RAW,F,VRYRAW
FOR SEIX-05/23/74-08:51:47 (1,2)

      SUBROUTINE VRYRAW     ENTRY POINT 000015

      STORAGE USED: CODE(1) 000021; DATA(0) 000004; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP16 000031

      EXTERNAL REFERENCES (BLOCK, NAME)

      0004   NERR4$
      0005   NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0003 L 000022 DEBUG     0000   000000 INJPS     0003   000010 IOPEN     0003   000011 JOPEN     0003 I 000001 KODE
      0003 L 000023 LFLT      0003 L 000027 NOMNAL    0003 L 000030 NOTYET    0003   000002 NRCLPL    0003   000003 NRPOLE
      0003   000004 NRZERO    0003   000005 NXB       0003   000006 NXN       0003   000007 NXR       0003   000014 PCPL
      0003   000015 PFAC      0003   000013 PNOM      0003   000016 PSLOSH    0003   000012 PVAR      0003   000000 STAGE
      0003 L 000017 YESMTX    0003 L 000024 YESPCH    0003 L 000020 YESRAW    0003 L 000025 YESRLP    0003 L 000026 YESSRL

6-217

00101     1*          SUBROUTINE VRYRAW(I)                                                    000000
00102     2*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,              240   000000
00103     3*         1 IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                                250   000000
00104     4*         2 YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,                         260   000000
00105     5*         3 YESSRL,NOMNAL,NOTYET                                                   270   000000
00106     6*          LOGICAL YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,                  280   000000
00107     7*         1 YESSRL,NOMNAL,NOTYET                                                   290   000000
00108     8*     C                                                                                 000000
00109     9*     C                                                                                 000000
00110    10*     C                                                                                 000000
00111    11*     C DUMMY SUBROUTINE FOR RAW PARAMETER VARIATIONS                                   000000
00105    12*          KODE = 107                                                                   000000
00106    13*          RETURN 1                                                                     000001
00107    14*          END                                                                          000020

      END OF COMPILATION:  NO DIAGNOSTICS.
```

```
?FOR.USR F.  TE,F.WRITE
FOR.SE1x-05/23/74-08:51:55 (1,2)

SUBROUTINE WRITE      ENTRY POINT 000136

STORAGE USED: CODE(1) 000142; DATA(0) 000122; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003   KEEP2   000047
0004   KEEP3   000102
0005   KEEP6   000134
0006   KEEP1A  000031
0007   CRUD1   000004
0010   CRUD3   011650

EXTERNAL REFERENCES (BLOCK, NAME)

0011   NWDUS
0012   NIO3S
0013   NIO2S
0014   NIO1S
0015   NERR3S

STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0000  000002 1F      0001  000033 134G    0001  000066 157G    0000  000010 21F     0000  000051 23F
0000  000056 30F     0000  000105 34F     0010  000070 ADIR    0010  000006 AFRQ    0010  011616 AML
0010  011617 AMP     0010  000152 APHA    0010  011620 BIG     0005  000064 CARD    0006 L 000022 DEBUG
0010  011642 DEGR    0010  011621 DF      0003 D 000002 DHFILT   0004 R 000063 DP      0010 L 011643 DPI
0010 L 011645 ERAM    0010 L 011647 ERGP    0010 L 011644 ERP     0010 L 011646 ERPH    0010  011622 FL
0010  011623 FR      0010 C 000000 GAINS   0003  000037 HACC    0003 R 000025 HAST    0003  000035 HATT
0003 D 000000 HBLANK  0003 R 000004 HBLK    0003  000034 HCCW    0003  000033 HCW     0003  000032 HDEC
0003  000027 HDOT    0003  000006 HEST1   0003  000043 HFGD    0003  000044 HFGN    0003  000040 HFPD
0003  000045 HFPDD   0003  000042 HFPDN   0003  000041 HFPN    0003  000010 HGENE   0003  000031 HINC
0003  000005 HKEY    0003  000007 HMATR   0003  000014 HNEW    0003  000046 HNOMI   0003  000015 HNYQU
0003  000023 HO      0003  000026 HPLUS   0003  000036 HRATE   0003  000011 HRAW    0003  000012 HRETA
0003  000017 HROLL   0003  000016 HROOT   0003  000013 HSTAN   0003  000030 HSTAR   0003  000020 HSTC
0003  000021 HS2     0003  000022 HS4B    0003  000024 HX      0010 R 011624 IM      0000  000113 INJPS
0010  011634 INT     0006 I 000010 IOPEN   0006 I 000011 IOPEN   0006  000001 KODE    0005  000110 LABEL
0006 I 000023 LFLT    0010  011641 LMX     0010  011635 LRPR    0000 I 000001 M       0004 R 000051 MAX
0004 R 000037 MIN     0010  011636 MPPP    0000 I 000000 N       0005 R 000062 NAME    0010  011637 NEXT
0004 I 000000 NFI     0010  000003 NGNPK   0004 L 000027 NOMNAL  0006 L 000030 NOTYET  0010  011640 NPPP
0010  000004 NP180   0006  000002 NRCLPL  0006  000003 NRPOLE  0006  000004 NRZERO  0006  000005 NXR
0006  000006 NXN     0006  000007 NXR     0010  000005 NYQPTS  0010  000002 N1AMP   0010  000316 PAMP
0004 R 000076 PB      0006  000014 PCPL    0004 R 000025 PCT     0010  000400 PDIR    0010  011625 PER
0006  000015 PFAC    0010  000234 PFRQ    0010  011626 PHA     0010  000626 PHAMP   0010  000710 PHDIR
0010  000544 PHFRQ   0010  011627 PHL     0004 R 000075 PN      0006  000013 PNOM    0010  000462 PPHA
0006  000016 PSLOSH  0006  000012 PVAR    0004  000077 P1R0    0010  011630 RE      0010  003726 SAVAMP
0010  000772 SAVFRQ  0010  006662 SAVPHA  0010  011631 SHA     0010  011632 STA     0006  000000 STAGE
0004 I 000101 STARRD  0010  011633 STO     0004 R 000013 STP     0004 R 000001 STR     0005 R 000000 TITLE
0005 R 000029 TITLE1  0005 R 000036 TITLE2  0005 R 000050 TITLE3  0007 R 000000 XXX     0007 R 000001 X3
0006 L 000017 YESMTX  0004 L 000100 YESNYQ  0006 L 000024 YESPCH  0006 L 000020 YESRAW  0006 L 000025 YESRLP
```

```
00101     1*            SUBROUTINE WRITE                                                    10336₀  000002
00102     2*            COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,       10337₀  000002
00103     3*           1              HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS4B,    10338₀  000002
00104     4*           2              H0,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,      10339₀  000002
00105     5*           3              HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPOD,     10340₀  000002
00106     6*           4              HNOMI                                                10341₀  000002
00107     7*            DOUBLE PRECISION HBLANK,DHFILT                                              000002
00108     8*            COMMON/KEEP3/ NFI,STR(10),STP(10),PCT(10),MIN(10),MAX(10),DP(10),    10343₀  000002
00109     9*           1              PN,P8,P180,YESNYQ,STNORD.                            10344₀  000002
00110    10*            REAL          MIN,MAX                                              10345₀  000002
00111    11*            LOGICAL       YESNYQ,STNORD                                        10346₀  000002
00112    12*            COMMON/KEEP4/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),  10347₀  000002
00113    13*           1              CARD(20),LABEL(20)                                   10348₀  000002
00114    14*            REAL          NAME                                                 10349₀  000002
00115    15*            COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,         10350₀  000002
00116    16*           1              IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,              10351₀  000002
00117    17*           2              YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,       10352₀  000002
00118    18*           3              YESSRL,NOMNAL,NOTYET                                 10353₀  000002
00119    19*            LOGICAL       YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,       10354₀  000002
00120    20*           1              YESSRL,NOMNAL,NOTYET                                 10355₀  000002
00121    21*            COMMON/CRUDI/ XXX,X3(3)                                            10356₀  000002
00122    22*            COMMON/CRUD3/ GAINS,NIAMP,NGNPK,NP180,NYQPTS,AFRQ(50),ADIR(50),    10357₀  000002
00123    23*           1              APHA(50),PFRQ(50),PAMP(50),PDIR(50),PPHA(50),        10358₀  000002
00124    24*           2              PHFRQ(50),PHAMP(50),PHDIR(50),SAVFRQ(1500),          10359₀  000002
00125    25*           3              SAVAMP(1500),SAVPHA(1500),AML,AMP,BIG,DF,FL,FR,IM,   10360₀  000002
00126    26*           4              PER,PHA,PHL,RE,SMA,STA,STO,                          10361₀  000002
00127    27*           5              INT,LRPR,MPPP,NEXT,NPPP,LMX,                         10362₀  000002
00128    28*           6              DECR,DPI,ERP,ERAM,ERPH,ERGP                          10363₀  000002
00129    29*            COMPLEX       GAINS                                                10364₀  000002
00130    30*            REAL          IM                                                   10365₀  000002
00131    31*            LOGICAL       DECR,DPI,ERP,ERAM,ERPH,ERGP                          10366₀  000002
00132    32*      C                                                                         10367₀  000002
00133    33*      C     PROGRAM CODING                                                      10368₀  000002
00134    34*      C                                                                         10369₀  000002
00135    35*      C                                                                         10370₀  000002
00136    36*            WRITE(6,1) TITLE,TITLE1,TITLE2,TITLE3                               10371₀  000002
00137    37*           1 FORMAT('1',8X,20A4//9X,20A4//9X,20A4/)                             10372₀  000023
00138    38*      C                                                                         10373₀  000023
00139    39*      C                                                                         10374₀  000023
00140    40*      C     PRINT FREQUENCY INTERVALS                                           10375₀  000023
00141    41*      C                                                                         10376₀  000023
00142    42*            20 WRITE(6,21)                                                      10377₀  000023
00143    43*           21 FORMAT (3(/),8X,25HFREQUENCY INTERVALS . . ./44X,8HPER CENT,11X,  10378₀  000033
00144    44*           1 7HMINIMUM,10X,7HMAXIMUM,12X,6HDETAIL/8X,11HSTART (CPS),7X,        10379₀  000033
00145    45*           2 10HSTOP (CPS),8X,10HMAX. INCR,8X,11HPHASE SHIFT,6X,11HPHASE SHIFT 10380₀  000033
00146    46*           3 9X,5HPRINT/1H )                                                   10381₀  000033
00147    47*            DO 22 N=1,NFI                                                      10382₀  000033
00150    48*            XXX = HBLK                                                         10383₀  000033
00151  *D,AGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00152    49*            IF (DP(N).NE.HBLK) XXX=HASL                                        10384₀  000034
00153    50*           22 WRITE(6,23) STR(N),STP(N),PCT(N),MIN(N),MAX(N),XXX               10385₀  000042
00154    51*           23 FORMAT (1PE17.5,3E18.5,E17.5,11X,A1)                             10386₀  000056
```

```
00152      52*      C                                                    10387₀    000056
00152      53*      C     PRINT EXECUTION OPTIONS                        10388₀    000056
00152      54*      C                                                    10389₀    000056
00153      55*            WRITE(6,30)                                    10390₀    000056
00155      56*       30 FORMAT(3(/),8X,9HOPEN,25X,5HPHASE/8X,4HLOOP,25X,5HSHIFT,4X,  10391₀    000066
00159      57*      1 7HNYQUIST,3X,4HBODE/8X,9HGAIN,12X,3HROW,2X,3HCOL,5X,4HTEST,5X,  10392₀    000066
00159      58*      2 4HPLOT,4X,4HPLOT//)                                10393₀    000066
00159      59*            DO 31 M=1,3                                    10394₀    000066
00161      60*       31 X3(M) = HBLK                                     10395₀    000066
00162      61*            X3(1) = HAST                                   10396₀    000070
00164   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00164      62*            IF (PA .NE. HBLK) X3(2) = HAST                 10397₀    000072
00166   *DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL*
00166      63*            IF (PB .NE. HBLK) X3(3) = HAST                 10398₀    000100
00170      64*            WRITE(6,34) NAME,IOPEN,JOPEN,X3                10399₀    000106
00176      65*       34 FORMAT(8X,2A4,7X,2I5,3(8X,A1))                   10400₀    000124
00177      66*            RETURN                                         10401₀    000124
00200      67*            END                                            10402₀    000141

      END OF COMPILATION:        3 DIAGNOSTICS*
```

```
@FOR,USR F*XCNG,F*XCNG
FOR 5E(X=05/23/74-08:52:05 (),2)

      SUBROUTINE XCNG       ENTRY POINT 000013

      STORAGE USED: CODE(1) 0n0023; DATA(D) 0n00005; BLANK COMMON(2) 000000

      EXTERNAL REFERENCES (BLOCK, NAME)

       0003    NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0000   000001-INJP$    0000 R 000000 Z


00101     1*          SUBROUTINE XCNG (X,Y)                    10403₀  000000
00101     2*     C                                             10404₀  000000
00101     3*     C                                             10405₀  000000
00101     4*     C  PROGRAM CODING                             10406₀  000000
00101     5*     C                                             10407₀  000000
00103     6*           Z = X                                   10408₀  000000
00104     7*           X = Y                                   10409₀  000001
00105     8*           Y = Z                                   10410₀  000003
00106     9*           RETURN                                  10411₀  000004
00107    10*           END                                     10412₀  000022

      END OF COMPILATION:  NO DIAGNOSTICS.
```

```
FOR,USN F[illegible]PLS,F,ZEEPLS
FOR SEIX-05/23/74-08:52:12 (2,3)

      SUBROUTINE ZEEPLS     ENTRY POINT 000063

      STORAGE USED: CODE(1) 000070; DATA(0) 000114; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   KEEP2   000047
      0004   KEEP8   000342
      0005   CRUD1   000113

      EXTERNAL REFERENCES (BLOCK, NAME)

      0006   NWDU$
      0007   NIO1$
      0010   NIO2$
      0011   NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000012 113G      0001   000043 126G      0000   000001 600F      0003 D 000002 DHFILT    0003   000037 HACC
      0003   000025 HAST      0003   000035 HATT      0003 D 000000 HBLANK    0003 R 000004 HBLK      0003   000034 HCCW
      0003   000033 HCW       0003   000032 HDEC      0003   000027 HDOT      0003   000006 HESTI     0003   000043 HFGD
      0003   000044 HFGN      0003   000040 HFPD      0003   000045 HFPDD     0003   000042 HFPDN     0003   000041 HFPN
      0003   000010 HGENE     0003   000031 HINC      0003   000005 HKEY      0003   000007 HMATR     0003   000014 HNEW
      0003   000046 HNOMI     0003   000015 HNYQU     0003   000023 HO        0003 R 000026 HPLUS     0003   000036 HRATE
      0003   000011 HRAW      0003   000012 HRETA     0003   000017 HROLL     0003   000016 HROOT     0003   000013 HSTAN
      0003   000030 HSTAR     0003   000020 HSIC      0003   000021 HS2       0003   000022 HS48      0003   000024 HX
      0000 I 000000 I         0000   000104 INJP$     0004 I 000341 NZPOLE    0005 R 000000 ZID       0004 R 000226 ZMAG
      0004 C 000000 ZPOLE

00101     1*          SUBROUTINE ZEEPLS                                          104130   000000
00103     2*          COMMON/KEEP2/ HBLANK,DHFILT,HBLK,HKEY,HESTI,HMATR,HGENE,HRAW,  104140   000000
00103     3*         1            HRETA,HSTAN,HNEW,HNYQU,HROOT,HROLL,HSIC,HS2,HS48,  104150   000000
00103     4*         2            HO,HX,HAST,HPLUS,HDOT,HSTAR,HINC,HDEC,HCW,HCCW,    104160   000000
00103     5*         3            HATT,HRATE,HACC,HFPD,HFPN,HFPDN,HFGD,HFGN,HFPDD,   104170   000000
00103     6*         4            HNOMI                                              104180   000000
00104     7*          DOUBLE PRECISION HBLANK,DHFILT                                000000
00105     8*          COMMON/KEEP8/ ZPOLE(75),ZMAG(75),NZPOLE                    104200   000000
00106     9*          COMPLEX       ZPOLE                                        000000
00107    10*          COMMON/CRUD1/ ZID(75)                                      104230   000000
00107    11*     C                                                               104240   000000
00107    12*     C                                                               104250   000000
00107    13*     C  PROGRAM CODING                                               104260   000000
00107    14*     C                                                               104270   000000
00110    15*          IF (NZPOLE.EQ.0) RETURN                                    104280   000000
00112    16*          DO 10 I=1,NZPOLE                                           104290   000004
00113    17*          ZID(I) = HBLK                                              104300   000012
```

```
00116     18          IF (ZMAG(I) .LT. 0.0001) ZID(I)=HPLUS                                  000013
00121     19*         IF (ZMAG(I) .GT. 1.0) ZID(I)=HSTAR                                     000022
00122     20*       10 CONTINUE                                                  10432D      000033
00123     21*         WRITE(6,600) (ZPOLE(I),ZMAG(I),ZID(I),I=1,NZPOLE)          10433D      000033
00131     22*     600 FORMAT('1',29X,'O P E N   L O O P   P O L E S   I N   T H E   Z        000050
00131     23*        1D O M A I N'//21X,'( ZERO ORDER HOLD NEGLECTED ••• STAR INDICATES     000050
00131     24*        2MAGNITUDE OUTSIDE UNITY CIRCLE )'//15X'( PLUS INDICATES MAGNITUDE    000050
00131     25*        3 WITHIN 0.0001 AND TERM DELETED FROM Z DOMAIN TRANSFER FUNCTION )'    000050
00131     26*        4///43X,'POLE',34X,'MAGNITUDE'//                                       000050
00131     27*        534X,'REAL',11X,'IMAGINARY'//(30X,1PE12.5,6X,E12.5,19X,E12.5,A4))      000050
00135     28*         RETURN                                                     10438D      000050
00135     29*         END                                                        10439D      000067

          END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,US F.ZE   ,F.ZEROS
FOR SEIx-05/23/74-08:52:24 (1,2)

    SUBROUTINE ZEROS      ENTRY POINT 000033

    STORAGE USED: CODE(1) 000041; DATA(0) 000010; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   KEEP6   000134
    0004   KEEP9   000705
    0005   KEEP15  000260
    0006   KEEP16  000031

    EXTERNAL REFERENCES (BLOCK, NAME)

    0007   COMPUT
    0010   NERR4S
    0011   NERR3S

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000015 1000L     0001   000004 115G      0003   000064 CARD      0005   000116 CPSLBL    0005   000024 DA
    0005   000130 DCBLBL    0006 L 000022 DEBUG     0005   000131 DEGLBL    0004 C 000226 EA        0005   000146 FMAT
    0005   000132 FMT       0005   000162 F6        0005   000163 F7        0000 I 000000 I         0005   000100 IDB
    0005   000107 IDG       0000   000002 INJP$     0006   000010 IOPEN     0006   000011 JOPEN     0004   000572 KD
    0006   000001 KODE      0003 I 000110 LABEL     0005 I 000164 LABEL1    0005   000210 LABEL2    0005   000234 LABEL3
    0005   000023 LFLT      0004   000455 NA        0003 R 000062 NAME      0004   000454 NEIG      0004   000457 NI
    0006 L 000027 NOMNAL     0006 L 000030 NOTYET    0005 I 000456 NR        0006   000002 NRCLPL    0006   000003 NRPOLE
    0006 I 000004 NRZERO     0006   000005 NXB       0006   000006 NXN       0006   000007 NXR       0006   000014 PCPL
    0006   000015 PFAC      0006   000013 PNOM      0006   000016 PSLOSH    0006   000012 PVAR      0005   000034 P36
    0005 C 000000 REGION     0005   000030 RID       0004 C 000000 ROOT      0005   000014 SPACE     0006   000000 STAGE
    0003   000000 TITLE     0003   000024 TITLE1    0003   000036 TITLE2    0003   000050 TITLE3    0005   000010 WIDTH
    0006 L 000017 YESMTX     0006 L 000024 YESPCH    0006 L 000020 YESRAW    0006 L 000025 YESRLP    0006 L 000026 YESSRL
    0006 L 000021 YESSRP     0005   000020 YINC

00101    1*          SUBROUTINE ZEROS(*)                                                    104400   000000
00103    2*          COMMON/KEEP6/ TITLE(20),TITLE1(10),TITLE2(10),TITLE3(10),NAME(2),      104410   000000
00103    3*        1             CARD(20),LABEL(20)                                       104420   000000
00104    4*          REAL          NAME                                                     104430   000000
00105    5*          COMMON/KEEP9/ ROOT(75),EA(75),NEIG,NA,NR,NI(75),KD(75)                 104440   000000
00106    6*          COMPLEX       ROOT,EA                                                           000000
00107    7*          COMMON/KEEP15/REGION(4),WIDTH(4),SPACE(4),YINC(4),                      104460   000000
00107    8*        1             DA(4),RID(4),P36(36),IDB(7),IDG(7),CPSLBL(10),           104480   000000
00107    9*        2             DCBLBL,DEGLBL,FMT(12),FMAT(12),F6,F7,LABEL1(20),         104490   000000
00107   10*        3             LABEL2(20),LABEL3(20)                                    104500   000000
00110   11*          COMPLEX       REGION                                                   104510   000000
00111   12*          COMMON/KEEP16/STAGE,KODE,NRCLPL,NRPOLE,NRZERO,NXB,NXN,NXR,             104530   000000
00111   13*        1             IOPEN,JOPEN,PVAR,PNOM,PCPL,PFAC,PSLOSH,                  104540   000000
00111   14*        2             YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,           104550   000000
```

```
00111     15*          3              YESSRL,NOMNAL,NOTYET                                  10456₀    000000
00112     16*         LOGICAL         YESMTX,YESRAW,YESSRP,DEBUG,LFLT,YESPCH,YESRLP,        10457₀    000000
00112     17*          1              YESSRL,NOMNAL,NOTYET                                  10458₀    000000
00112     18*     C                                                                         10465₀    000000
00112     19*     C                                                                         10466₀    000000
00112     20*     C   CALCULATE THE OPEN LOOP ZEROS                                        10467₀    000000
00112     21*     C                                                                         10468₀    000000
00113     22*           NR=NRZERO                                                           10472₀    000000
00114     23*           DO 10 I=1,20                                                        10473₀    000004
00117     24*        10 LABEL(I) = LABEL1(I)                                                10474₀    000004
00121     25*           CALL COMPUT(*1000)                                                           000006
00122     26*           RETURN                                                              10477₀    000011
00122     27*     C                                                                         10478₀    000011
00122     28*     C                                                                         10479₀    000011
00122     29*     C     ERROR IN COMPUTING THE OPEN LOOP ZEROS                              10480₀    000011
00122     30*     C                                                                         10481₀    000011
00123     31*      1000 CONTINUE                                                            10482₀    000015
00124     32*           RETURN 1                                                            10483₀    000015
00125     33*           END                                                                 10484₀    000040

     END OF COMPILATION:      NO DIAGNOSTICS.
```

SUBROUTINE ZTRAN ENTRY POINT 001120

STORAGE USED: CODE(1) 001144; DATA(0) 000160; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003	KEEP5	000074
0004	KEEP8	000342
0005	KEEP14	000031
0006	KEEP20	000227
0007	KEEP21	001133
0010	CRUD1	000043
0011	CRUD2	001215
0012	CRUD4	000002

EXTERNAL REFERENCES (BLOCK, NAME)

0013	NCDU$
0014	NIO2$
0015	EXP
0016	COS
0017	SIN
0020	CSQRT
0021	CABS
0022	NERR3$

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001		000053	10L	0001		000333	100L	0001		000513	110L	0001		000644	150L
0001		000150	50L	0000		000000	$60^{0}F$	0000		000011	$60^{1}F$	0000		000022	$60^{2}F$
0000		000033	$60^{3}F$	0000		000045	$60^{4}F$	0000		000056	$60^{5}F$	0000		000067	$60^{6}F$
0000		000100	$60^{7}F$	0000		000112	$60^{8}F$	0003	L	000071	BOTH	0005		000025	DEG
0005	R	000012	D^{0}	0005	R	000013	D^{1}	0005	R	000014	D^{2}	0005		000015	D^{3}
0005	R	000016	D^{4}	0006	C	000000	ESIZ	0005	C	000010	FIFTEN	0005		000017	FIFTY
0005	C	000006	FOUR	0003	L	000073	GPRINT	0005	C	000000	HALF	0000		000130	INJP$
0003		000067	ITHZT	0011	I	001214	LOCD	0007	I	000704	LOCDZ	0011	I	001213	LOCN
0007	I	000703	LOCNZ	0003	L	000072	MODIFY	0011	I	001077	NDCPER	0007	I	001024	NDCZ
0012		000000	NERZ	0004	I	000226	NESTZ	0011	I	000764	NNCPER	0007	I	000705	NNCZ
0011	I	001212	NUMPOL	0007	I	000702	NUMZ	0004	I	000341	NZPOLE	0003		000004	NZT
0005		000026	N1	0005		000027	N2	0005		000030	N3	0005	C	000002	ONE
0011	R	000310	PD	0007	R	000341	PDZ	0005	C	000020	PI	0005		000021	PI^{2}
0011	R	000000	PNI	0007	R	000000	PNZ	0005		000024	RADDEG	0010	R	000035	ROH^{1}
0010	R	000036	ROH^{2}	0010	R	000037	ROH^{3}	0010	R	000040	ROH^{4}	0010	R	000041	ROH^{5}
0005		000023	RPI	0010	R	000021	R1	0010	R	000022	R2	0010	R	000023	R^{3}
0010	R	000024	R^{4}	0010	R	000025	R^{5}	0010	R	000026	R^{6}	0005		000022	SMALL
0003	R	000000	SUPERK	0003		000003	TD	0010	C	000002	TERM	0005	C	000004	TWO
0010	R	000006	T1	0010	R	000007	T2	0010	R	000010	T3	0010	R	000011	T4
0010	R	000012	T^{5}	0010	C	000000	VAL	0012	L	000001	YESZM	0003	L	000070	YESZOH
0003		000002	ZM	0004	R	000226	ZMAG	0010	R	000027	ZOH^{1}	0010	R	000030	ZOH^{2}
0010	R	000031	ZOH^{3}	0010	R	000032	ZOH^{4}	0010	R	000033	ZOH^{5}	0010	R	000034	ZOH^{6}
0004	C	000000	ZPOLE	0003	R	000001	ZT	0003		000005	ZTVAL	0010	R	000013	Z1
0010	R	000014	Z2	0010	R	000015	Z^{3}	0010	R	000016	Z4	0010	R	000042	Z4R
0010	R	000017	Z^{5}	0010	C	000004	Z5TWO	0010	R	000020	Z^{6}				

```
00101     1       SUBROUTINE ZTRANS(NC,RTR,RTI,RSR,RSI)                          106360  000000
00102     2*      COMMON/KEEP5/ SUPERK,ZT,ZM,TD,NZT,ZTVAL(50),ITHZT,YESZOH,BOTH,
00103     3*     1            MODIFY,GPRINT                                      106390  000000
00104     4*      LOGICAL       YESZOH,BOTH,MODIFY,GPRINT                       106410  000000
00105     5*      COMMON/KEEP8/ ZPOLE(75),ZMAG(75),NZPOLE                       106420  000000
00106     6*      COMPLEX       ZPOLE                                                   000000
00107     7*      COMMON/KEEP14/HALF,ONE,TWO,FOUR,FIFTEN,D0,D1,D2,D3,D4,FIFTY,PI,  106450  000000
00107     8*     1            PI2,SMALL,RPI,RADDEG,DEG,N1,N2,N3                  106460  000000
00110     9*      COMPLEX       HALF,ONE,TWO,FOUR,FIFTEN                                000000
00111    10*      COMMON/KEEP20/ESTZ(75),NESTZ                                          000000
00112    11*      COMPLEX       ESTZ                                                    000000
00113    12*      COMMON/KEEP21/PNZ(225),PDZ(225),NUMZ,LOCNZ,LOCDZ,NNCZ(75),NDCZ(75)   000000
00114    13*      COMMON/CRUD1/ VAL,TERM,ZSTWO,                                 106490  000000
00114    14*     1            T1,T2,T3,T4,T5,Z1,Z2,Z3,Z4,Z5,Z6,R1,R2,R3,R4,R5,R6,  106500  000000
00114    15*     2            ZOH1,ZOH2,ZOH3,ZOH4,ZOH5,ZOH6,                      106510  000000
00114    16*     3            ROH1,ROH2,ROH3,ROH4,ROH5,Z4N                        106520  000000
00115    17*      COMPLEX       VAL,TERM,ZSTWO                                          000000
00116    18*      COMMON/CRUD2/ PNN(200),PD(300),NNCPER(75),NDCPER(75),               000000
00116    19*     1            NUMPOL,LOCN,LOCD                                    106580  000000
00117    20*      COMMON/CRUD4/ NEQZ,YESZM                                          106600  000000
00121    21*      LOGICAL       YESZM                                               106610  000000
00121    22*      C                                                                 106620  000000
00121    23*      C                                                                 106630  000000
00121    24*      C                                                                 106640  000000
00121    25*      C                                                                 106650  000000
00121    26*            IF (GPRINT) WRITE(6,600) RTR,RTI,RSR,RSI                    106660  000000
00131    27*      600 FORMAT(///10X,1PE12.5,5X,E12.5,10X,'RESIDUE',2(5X,E12.5))     106670  000013
00131    28*            IF (NINC.EQ.2) GO TO 100                                    106680  000013
00131    29*      C                                                                 106690  000013
00131    30*      C                                                                 106700  000013
00131    31*      C  REAL NON-ZERO ROOT                                             106710  000013
00131    32*      C                                                                 106720  000013
00133    33*            Z1=SUPERK*RSR                                               106730  000017
00134    34*            Z2 = D0                                                     106740  000022
00135    35*            Z3 = D1                                                     106750  000024
00136    36*            Z4 = -EXP(RTR*ZT)                                                   000026
00137    37*            IF (BOTH) GO TO 10                                          106770  000035
00141    38*            NZPOLE=NZPOLE+1                                             106780  000037
00142    39*            Z4N=-Z4                                                     106790  000042
00143    40*            ZPOLE(NZPOLE)=CMPLX(Z4N,0.0)                                106800  000043
00144    41*            ZMAG(NZPOLE) = ABS(Z4)                                              000050
00145    42*         10 CONTINUE                                                    106820  000053
00146    43*            IF (YESZOH) GO TO 50                                        106830  000053
00146    44*      C                                                                 106840  000053
00146    45*      C                                                                 106850  000053
00146    46*      C  PRINT Z COEFFICIENTS                                           106860  000053
00146    47*      C                                                                 106870  000053
00150    48*            IF (GPRINT) WRITE(6,601) Z1,Z2,Z3,Z4                        106880  000054
00157    49*      601 FORMAT(1/51X,'Z NUM',2(5X,1PE12.5)/53X,'DEN',2(5X,E12.5))     106890  000067
00160    50*            R1=Z1                                                       106900  000067
00161    51*            R2=Z1                                                       106910  000071
00162    52*            R3=Z3+Z4                                                    106920  000072
00163    53*            R4=Z3-Z4                                                    106930  000075
00163    54*      C                                                                 106940  000075
00163    55*      C                                                                 106950  000075
00163    56*      C  PRINT R COEFFICIENTS                                           106960  000075
```

```
00163          C
00164     58*      IF (GPRINT) WRITE(6,602) R1,R2,R3,R4
00173     59*    602 FORMAT(/51X,'R NUM',2(5X,1PE12.5)/53X,'DEN',2(5X,E12.5))
00173     60*    C
00173     61*    C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS
00173     62*    C
00173     63*    C  NUMERATOR COEFFICIENTS
00174     64*        LOCN = LOCN + 1
00175     65*        PNN (LOCN) = R1
00176     66*        LOCN = LOCN + 1
00177     67*        PNN (LOCN) = R2
00177     68*    C  DENOMINATOR COEFFICIENTS
00200     69*        LOCD = LOCD + 1
00201     70*        PD (LOCD) = R3
00202     71*        LOCD = LOCD + 1
00203     72*        PD (LOCD) = R4
00204     73*        NNCPER(NUMPOL) = 2
00205     74*        NDCPER(NUMPOL) = 2
00206     75*        RETURN
00206     76*    C
00206     77*    C  ZERO ORDER HOLD SELECTED
00206     78*    C
00207     79*       50 CONTINUE
00210     80*        ZOH1=Z1
00211     81*        ZOH2=-Z1
00212     82*        ZOH3=Z3
00213     83*        ZOH4=Z4
00213     84*    C
00213     85*    C
00213     86*    C  PRINT ZOH COEFFICIENTS
00213     87*    C
00214     88*        IF (GPRINT) WRITE(6,603) ZOH1,ZOH2,ZOH3,ZOH4
00223     89*    603 FORMAT(/49X,'ZOH NUM',2(5X,1PE12.5)/53X,'DEN',2(5X,E12.5))
00224     90*        ROH1=ZOH1+ZOH2
00225     91*        ROH2=ZOH3+ZOH4
00226     92*        ROH3=ZOH3-ZOH4
00226     93*    C
00226     94*    C
00226     95*    C  PRINT ROH COEFFICIENTS
00226     96*    C
00227     97*        IF (GPRINT) WRITE(6,604) ROH1,ROH2,ROH3
00235     98*    604 FORMAT(/49X,'ROH NUM',5X,1PE12.5/53X,'DEN',2(5X,E12.5))
00235     99*    C
00235    100*    C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS
00235    101*    C
00235    102*    C  NUMERATOR COEFFICIENTS
00236    103*        LOCN = LOCN + 1
00237    104*        PNN (LOCN) = ROH1
00237    105*    C  DENOMINATOR COEFFICIENTS
00240    106*        LOCD = LOCD + 1
00241    107*        PD (LOCD) = ROH2
00242    108*        LOCD = LOCD + 1
00243    109*        PD (LOCD) = ROH3
00244    110*        NNCPER(NUMPOL) = 1
00245    111*        NDCPER(NUMPOL) = 2
00246    112*        IF (ZMAG(NZPOLE).LT.1.0D-4 .OR. .NOT.YESZH) RETURN
00251    113*        NUMZ = NUMZ + 1
```

```
10697₀  000075
10698₀  000100
10699₀  000113
10700₀  000113
10701₀  000113
10702₀  000113
10703₀  000113
10704₀  000113
10705₀  000116
10706₀  000121
10707₀  000123
10708₀  000123
10709₀  000126
10710₀  000131
10711₀  000134
10712₀  000137
10713₀  000141
10714₀  000143
10715₀  000144
10716₀  000144
10717₀  000144
10718₀  000144
10719₀  000150
10720₀  000150
10721₀  000151
10722₀  000152
10723₀  000154
10724₀  000154
10725₀  000154
10726₀  000154
10727₀  000154
10728₀  000156
10729₀  000171
10730₀  000171
10731₀  000174
10732₀  000177
10733₀  000177
10734₀  000177
10735₀  000177
10736₀  000177
10737₀  000202
10738₀  000221
10739₀  000221
10740₀  000221
10741₀  000221
10742₀  000221
10743₀  000221
10744₀  000224
10745₀  000224
10746₀  000227
10747₀  000232
10748₀  000235
10749₀  000237
10750₀  000242
10751₀  000244
10752₀  000246
        000263
```

```
00251    114*          NNCZ(NUMZ) = 2                                                    000266
00252    115*          LOCNZ = LOCNZ + 1                                                 000271
00253    116*          PNZ(LOCNZ) = ZOH1                                                 000274
00254    117*          LOCNZ = LOCNZ + 1                                                 000277
00255    118*          PNZ(LOCNZ) = ZOH2                                                 000301
00256    119*          NDCZ(NUMZ) = 2                                                    000304
00257    120*          LOCDZ = LOCDZ + 1                                                 000305
00260    121*          PDZ(LOCDZ) = ZOH3                                                 000310
00261    122*          LOCDZ = LOCDZ + 1                                                 000313
00262    123*          PDZ(LOCDZ) = ZOH4                                                 000315
00263    124*          NESTZ = NESTZ + 1                                                 000320
00264    125*          ESTZ(NESTZ) = ZPOLE(NZPOLE)                                       000323
00265    126*          RETURN                                              10755₀        000327
00265    127*    C                                                         10756₀        000327
00265    128*    C                                                         10757₀        000327
00265    129*    C  COMPLEX NO = ZERO ROOT                                 10758₀        000327
00265    130*    C                                                         10759₀        000327
00266    131*      100 CONTINUE                                            10760₀        000333
00267    132*          T1 = EXP(RTR*ZT)                                                  000333
00270    133*          T2 = D1                                           10762₀        000341
00271    134*          T3 = ABS(RTI)*ZT                                                  000343
00272    135*          T4 = D2*RSD*T2                                    10764₀        000346
00273    136*          T5 = -D2*RSI*T2                                   10765₀        000352
00274    137*          IF (RTI.LT.0.0) T5=-T5                                          000356
00276    138*          Z1=SUPERK*T4                                      10767₀        000364
00277    139*          Z2 = SUPERK*(T5*T1*SIN(T3) - T4*T1*COS(T3))                     000367
00300    140*          Z3 = D0                                           10769₀        000410
00301    141*          Z4 = D1                                           10770₀        000412
00302    142*          Z5 = -D2*T1*COS(T3)                                             000414
00303    143*          Z6 = EXP(D2*RTR*ZT)                                             000420
00304    144*          IF (90TH) GO TO 110                               10773₀        000430
00306    145*          VAL = Z5**2/D4 - Z6                               10774₀        000432
00307    146*          TERM = CSQRT(VAL)                                               000440
00310    147*          NZPOLE=NZPOLE+1                                   10776₀        000444
00311    148*          Z5TWO = Z5/D2                                     10777₀        000447
00312    149*          ZPOLE(NZPOLE)=-Z5TWO+TERM                         10778₀        000453
00313    150*          ZMAG(NZPOLE) = CABS(ZPOLE(NZPOLE))                              000461
00314    151*          NZPOLE=NZPOLE+1                                   10780₀        000470
00315    152*          ZPOLE(NZPOLE)=-Z5TWO-TERM                         10781₀        000475
00316    153*          ZMAG(NZPOLE) = CABS(ZPOLE(NZPOLE))                              000504
00317    154*      110 CONTINUE                                          10783₀        000513
00320    155*          IF (YESZOH) GO TO 150                             10784₀        000513
00322    156*          IF (GPRINT) WRITE(6,605) Z1,Z2,Z3,Z4,Z5,Z6        10785₀        000514
00331    157*    605 FORMAT(/51X,'Z NUM',3(5X,1PE12.5)/53X,'DEN',3(5X,E12.5))  10786₀  000531
00331    158*          R1=Z1+Z2                                          10787₀        000531
00332    159*          R2=-D2*Z1                                         10788₀        000534
00335    160*          R3=Z1-Z2                                          10789₀        000537
00337    161*          R4=Z4+Z5+Z6                                       10790₀        000542
00341    162*          R5 = D2*Z4 - D2*Z6                                10791₀        000546
00341    163*          R6=Z4-Z5+Z6                                       10792₀        000554
00341    164*    C                                                       10793₀        000554
00341    165*    C                                                       10794₀        000554
00341    166*    C  PRINT R COEFFICIENTS                                 10795₀        000554
00341    167*    C                                                       10796₀        000554
00342    168*          IF (GPRINT) WRITE(6,606) R1,R2,R3,R4,R5,R6        10797₀        000560
00351    169*    606 FORMAT(/51X,'R NUM',3(5X,1PE12.5)/53X,'DEN',3(5X,E12.5))  10798₀  000575
00351    170*    C                                                       10799₀        000575
```

```
00353   171*      C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS
00353   172*      C
00353   173*      C  NUMERATOR COEFFICIENTS
00354   174*            LOCN = LOCN + 1
00355   175*            PNN (LOCN) = R1
00356   176*            LOCN = LOCN + 1
00357   177*            PNN (LOCN) = R2
00360   178*            LOCN = LOCN + 1
00361   179*            PNN (LOCN) = R3
00361   180*      C  DENOMINATOR COEFFICIENTS
00362   181*            LOCD = LOCD + 1
00363   182*            PD (LOCD) = R4
00364   183*            LOCD = LOCD + 1
00365   184*            PD (LOCD) = R5
00366   185*            LOCD = LOCD + 1
00367   186*            PD (LOCD) = R6
00370   187*            NNCPER(NUMPOL) = 3
00371   188*            NDCPER(NUMPOL) = 3
00372   189*            RETURN
00372   190*      C
00372   191*      C
00372   192*      C  ZERO ORDER HOLD SELECTED
00372   193*      C
00373   194*      160 CONTINUE
00374   195*            ZOH1=Z1
00375   196*            ZOH2=Z2-Z1
00376   197*            ZOH3=-Z2
00377   198*            ZOH4=Z4
00400   199*            ZOH5=Z5
00401   200*            ZOH6=Z6
00401   201*      C
00401   202*      C
00401   203*      C  PRINT ZOH COEFFICIENTS
00401   204*      C
00402   205*            IF (GPRINT) WRITE(6,607) ZOH1,ZOH2,ZOH3,ZOH4,ZOH5,ZOH6
00413   206*        607 FORMAT(/49X,'ZOH NUM',3(5X,1PE12.5)/53X,'DEN',3(5X,E12.5))
00414   207*            ROH1 = D2*ZOH1 - D2*ZOH3
00415   208*            ROH2=ZOH1-ZOH2+ZOH3
00415   209*            ROH3=ZOH4+ZOH5+ZOH6
00417   210*            ROH4 = D2*ZOH4 - D2*ZOH6
00420   211*            ROH5=ZOH4-ZOH5+ZOH6
00420   212*      C
00421   213*      C
00421   214*      C  PRINT ROH COEFFICIENTS
00421   215*      C
00421   216*            IF (GPRINT) WRITE(6,608) ROH1,ROH2,ROH3,ROH4,ROH5
00431   217*        608 FORMAT(/49X,'ROH NUM',2(5X,1PE12.5)/53X,'DEN',3(5X,E12.5))
00431   218*      C
00431   219*      C  THE FOLLOWING IS USED FOR NYQUIST ANALYSIS
00431   220*      C
00431   221*      C  NUMERATOR COEFFICIENTS
00432   222*            LOCN = LOCN + 1
00433   223*            PNN (LOCN) = ROH1
00434   224*            LOCN = LOCN + 1
00435   225*            PNN (LOCN) = ROH2
00436   226*      C  DENOMINATOR COEFFICIENTS
00436   227*            LOCD = LOCD + 1
```

```
108000   000575
108010   000575
108020   000575
108030   000575
108040   000600
108050   000603
108060   000605
108070   000610
108080   000612
108090   000612
108100   000615
108110   000620
108120   000623
108130   000625
108140   000630
108150   000633
108160   000635
108170   000637
108180   000640
108190   000640
108200   000640
108210   000640
108220   000640
108230   000644
108240   000644
108250   000645
108260   000647
108270   000651
108280   000653
108290   000655
108300   000655
108310   000655
108320   000655
108330   000655
108340   000657
108350   000674
108360   000674
108370   000702
108380   000706
108390   000712
108400   000720
108410   000720
108420   000720
108430   000720
108440   000720
108450   000724
108460   000745
108470   000745
108480   000745
108490   000745
108500   000745
108510   000745
108520   000750
108530   000753
108540   000755
108550   000755
108560   000760
```

```
00437     228*          PD (LOCD) = ROH3                                    10857₀    000763
00440     229*          LOCD = LOCD + 1                                     10858₀    000766
00441     230*          PD (LOCD) = ROH4                                    10859₀    000770
00442     231*          LOCD = LOCD + 1                                     10860₀    000773
00443     232*          PD (LOCD) = ROH5                                    10861₀    000775
00444     233*          NNCPER(NUMPOL) = 2                                  10862₀    001000
00445     234*          NDCPER(NUMPOL) = 3                                  10863₀    001002
00446     235*          IF (ZMAG(NZPOLE).LT.1.OD-4 .OR. .NOT.YESZM) RETURN  10864₀    001004
00451     236*          NUMZ = NUMZ + 1                                                001021
00451     237*          NNCZ(NUMZ) = 3                                                 001024
00452     238*          LOCNZ = LOCNZ + 1                                              001027
00453     239*          PNZ(LOCNZ) = ZOH1                                              001032
00454     240*          LOCNZ = LOCNZ + 1                                              001035
00455     241*          PNZ(LOCNZ) = ZOH2                                              001037
00456     242*          LOCNZ = LOCNZ + 1                                              001042
00457     243*          PNZ(LOCNZ) = ZOH3                                              001044
00460     244*          NDCZ(NUMZ) = 3                                                 001047
00461     245*          LOCDZ = LOCDZ + 1                                              001050
00462     246*          PDZ(LOCDZ) = ZOH4                                              001053
00463     247*          LOCDZ = LOCDZ + 1                                              001056
00464     248*          PDZ(LOCDZ) = ZOH5                                              001060
00465     249*          LOCDZ = LOCDZ + 1                                              001063
00466     250*          PDZ(LOCDZ) = ZOH6                                              001065
00467     251*          NESTZ = NESTZ + 1                                              001070
00470     252*          ESTZ(NESTZ) = ZPOLE(NZPOLE)                                    001073
00471     253*          RETURN                                              10867₀    001100
00472     254*          END                                                 10868₀    001143

     END OF COMPILATION:  NO DIAGNOSTICS.
```

```
PMAP,L F,*E   F,AERO
MAP 0026-05/23-09:53  (3,)

      1*      TYPE SETAFCM
      2*      LIB F,LEC*UR
      3*        LIB MSC*PLTLIB
      4*      SEG A
      5*      IN MAIN,ADDZOH,BLOCK,CSNTRX,DEZOH,ERROR,LIMIT,NOMMTX,RESET,ZEEPLS
      6*      IN KEEP1,KEEP2,KEEP3,KEEP4,KEEP5,KEEP6,KEEP7,KEEP8,KEEP9,KEEP10,KEEP11
      7*      IN KEEP13,KEEP14,KEEP15,KEEP16,KEEP18,KEEP19,KEEP20,KEEP21
      8*      IN CRUD1,CRUD2,CRUD3,CRUD4,CRUD5,PLT,PLTARY,NICHOL
      9*      SEG B1*,(A)
     10*      IN AFTVAR,DATA,GENMTX,INPEST,INPMTX,INPNYQ,INPRL,INPUT,PREVAR,RAWMTX
     11*      IN STNRL,VRYGEN,VRYRAN
     12*      SEG B2*,(A)
     13*      IN AGAIN,CDARY,DECIDE,DET,ESTMAT,FIXIT,KCALC,PEVAL,ROOTER,SAVE
     14*      IN SCALE,SOLVE,START,XCNG
     15*      SEG C1*,(B2)
     16*      IN CLPOLE,COMPUT,CSOLTF,DETCS,FORM,FRMTX,POLES,PRINTT,PUTOUT,RLOCUS
     17*      IN ZEROS
     18*      SEG C2*,(B2)
     19*      IN BHA272,DETSD,GETEST,RLPLOT,RLPRNT,RUTER,SAVOUT,SRRL,SRRLPP,SAVEST
     20*      SEG B3*,(A)
     21*      IN BHA341,MLIZRO,MODZRO,NZTRAN,PFE,PFEZRO,SFPOPC,ZTRAN
     22*      SEG B4*,(A)
     23*      IN BEGIN,BHAD91,BODE,DB,GRAPHS,INITAL,INT1,INT2,NYQIST,OUTPUT,POINT
     24*          IN POLVAL,STNNYQ,SUMMRY,SYSERR,TEST,TRNSFR,WRITE

FILE  F              NEEDS A PREP IN ORDER TO BE SEARCHED
AFCM STATUS OF OUTPUT ELEMENT=SETAFCM

ADDRESS LIMITS    001000 033506     13639 IBANK WORDS DECIMAL
                  040000 142522     34131 DBANK WORDS DECIMAL
SEGMENT LOAD TABLE      040000 040033
INDIRECT LOAD TABLE.    040034 040377
STARTING ADDRESS  023417

          SEGMENT A             001000 024573    040400 140306

SYS$*RLIB$.NSNTCS/FOR69
                    S(1)  001000 001024
     EXTERNAL REFERENCES:  NTAB$, FNCTB$, IOCOD$, WRBLK$

SYS$*RLIB$.NRBLK$/FOR68
                    S(1)  001025 001047
     EXTERNAL REFERENCES:  NTAB$, UNIT$, WAIT$, NIOER$, RS, UPDOA$,
     IO$

SYS$*RLIB$.NRAND$/FOR68
                    S(1)  001050 001131     S(2)  040400 040411
     EXTERNAL REFERENCES:  NTAB$, NSII$, NHPFA$, IOCOD$, NFCHK$, WAIT$,
```

NIOER$, MB$, DRAIN$, NRBF$, REW$, IO$, STREG$, PRINT$, NWALK$

SYS$*RLIB$.NKEF$/FOR69

$(1) 001432 001336 $(2) 040412 040431

EXTERNAL REFERENCES: NTAB$, NSLI$, NHPFA$, IOCOD$, NFCHK$,
NBFMG$, PACKT$, RDBLK$, UNIT$, UPDDA$, WAIT$, BSIBL$, DRAIN$,
NRFGT$, NIOER$, NBFRL$, NSWTC$, NRDFA$, PUNCH$, PUCHA$, STREG$,
PRINT$, NWALK$, CLOSE$, NEF$, IO$

SYS$*RLIB$.NFICHS/FOR69

$(1) 001336 001620 $(2) 040432 040445

EXTERNAL REFERENCES: NTAB$, RDBLK$, WAIT$, NIOER$, IOCOD$,
NBFRL$, NBFGT$, NBFMG$, R$, NFBY1$, NIOERSA, NBFRS$, NFRONF$, MB$,
UNIT$, MF$, IOW$, FNCTB$, UPDDA$, STREG$, NSTAT$, NERCT$

SYS$*RLIB$.NBDCV$/FOR64

$(1) 001621 001746 $(2) 040446 040510

EXTERNAL REFERENCES: NC1ULO, NFOP$, NC1UL1

SYS$*RLIB$.NFTV$/FOR

$(1) 001747 001771

SYS$*RLIB$.NCNVT$/FOR68

$(1) 001772 002213 $(2) 040511 040605

EXTERNAL REFERENCES: STREG$, NSTSV$, NSTAT$, NCOM3$, NERCR$,
NFTGL$, NCOOF$, NERCT$

SYS$*RLIB$.NCLOS$/FOR68

$(1) 002214 002404 $(2) 040606 040636

EXTERNAL REFERENCES: NTAB$, NSLI$, UNIT$, CSF$, IOW$, MB$, NWEF$,
WAIT$, NREW$, NRBF$, STREG$, NCEF$, PRINT$, NWALK$, NTBSZ$,
NIOER$, W$, IO$

SYS$*RLIB$.NABLK$/FOR68

$(1) 002405 002516

EXTERNAL REFERENCES: NTAB$, UNIT$, WAIT$, NIOER$, W$, UPDDA$,
IO$

SYS$*RLIB$.NBSBL$/FOR68

$(1) 002517 002557

EXTERNAL REFERENCES: NTAB$, MB$, WAIT$, NIOER$, IOW$, UPDDA$

SYS$*RLIB$.NUPDA$/FOR68

$(1) 002560 002613

EXTERNAL REFERENCES: NTAB$, WAIT$, MB$

SYS$*RLIB$.NBFDC$/FOR

$(2) 040637 043040

SYS$*RLIB$.NININ$/FOR68

$(1) 002614 003004 $(2) 043041 043044

EXTERNAL REFERENCES: NTAB$, PACKT$, NERR$, NREC$, NERU$, NRD$,
NKLN$, NKL2$, NFRA$, NLLM$, NRTR$, NFTCB$, TEMP$, UNIT$, NFTCH$,
NRCA$, NIIC$, NCSP$, NBIPA$, NEFCL$, READA$

SYS$*RLIB$.NINPT$/FOR69

$(1) 003005 004013 $(2) 043045 043075

EXTERNAL REFERENCES: NNG90$, NFGTS, IOCOD$, NR92$, NR93$, NLLC$,
NFM96$, NFAR$, NFRZ$, NP92$, NP91$, STREG$, NSTSV$, STAT$,
NCOM3$, NFTGL$, NERCR$, NFC1$, NCNV9$, NSF$, NFSG$, NFDB$, NDBF1$,
NDBCV$, NFRC$, NFRH$, NEFCL$, NFCM$, NDBIN$, NGC9$, NPCT$, NT10$,
NFGC$, NRTR$, NFRG$, NDBLT$, READ$, NCSP$, NVEC$

SYS$*RLIB$.FSR1/FOR69

$(1) 004015 004040 $(2) 043076 043103

SYS$*RLIB$.NOTIN$/FOR68

$(1) 004041 004335 $(2) 043104 043107

EXTERNAL REFERENCES: NTAB$, NFRJ$, NREC$, NFPC$, NSTSV$, PACKT$,
NERU$, NPU$, NPP$, NKLN$, NKL2$, NFRA$, NOLM$, NTEND$, NBFMG$,
NCIUL0, NBFGT$, NBFR$$, WAIT$, NIOER$, UPDDA$, BSIBL$, FNCTB$,
PUNCHA$, NEXIT$, NCCC$, PPP$, PRNTA$, NCSP$, TEMP$, DRAIN$, UNIT$,
NBFRL$, NCJNIO2$, CFE, NIO2$

SYS$*RLIB$.NOUT$/FOR69

$(1) 004336 005512 $(2) 043110 043146

EXTERNAL REFERENCES: NCSP$, NFRJ$, NFPC$, IOCOD$, NPCT$, NR92$,
NR93$, NRM92$, NFM96$, NFAR$, NFRZ$, NP91$, N0I$, NFNS1$, FMTOP,
NFNS2$, NFNS3$, NDIG$, NSL$, NDOUT$, NIND$, NFGC$, NGC9$, NT10$,
NFPA$, XFOR$, NR91$, NFMT$, PRNTA$, PRINT$, PUNCH$, NVEC$

SYS$*RLIB$.NFMT$/FOR69

$(1) 005513 006367 $(2) 043147 043223

EXTERNAL REFERENCES: NTAB$, NFRZ$, NFRZS$, NFMTR$, NFTGL$,
NIO1V$, NFNIO1$, NIO3V$, NFNIO1D$, NIO3VA$, NDBI$, NAB7$, NAB0$,
NAB4$, NAB2$, NAB5$, NAB3$, NAB1$, NAB6$, STREG$, NSTAT$, NERCR$,
NFCS$, NDBCV$, NHVC$, NDBIN$, NXVC$, NAVC$, NFRG$, NRTR$, PRINT$,
NFCA$, NVEC$, NIO2$, IOCOD$, NCA$, NCHAR$, NSTSV$

SYS$*RLIB$.NIOER$/FOR69

$(1) 006370 006557 $(2) 043220 043362

EXTERNAL REFERENCES: NTAB$, STREG$, UNIT$, NLRT$, NLTB$, NSTAT$,
NCJNIO2$, NTEND$, NSI1$, NRSF$, NSA0$, PRINT$, PACKT$, NWALK$

SYS$*RLIB$.NFCHK$/FOR69

$(1) 006560 007545 $(2) 043363 043536
$(4) 043537 043610

EXTERNAL REFERENCES: NTAB$, NERU$, NTBSZ$, UNIT$, NBTOD$, FITEN$,
PL$, BL$, PACKT$, IOCOD$, STREG$, NSTAT$, PRINT$, NWALK$, NSI1$,
CSF$, WAIT$, NIOER$, WS, IO$, UPDDA$, BSIBL$, MB$, TEMP$, DRAIN$,
NPBLK$, NCIUL0, NCIUL1, B2L$, B20$, B10$, BIL$, CLOSE$, EXIT

SYS$*RLIB$.NTAB$/JSC

$(2) 043611 043650

SYS$*RLIB$.NEXP6$/FOR68

$(1) 007546 007742 $(2) 043651 043722

EXTERNAL REFERENCES: NERRA$, NERRB$, NERRC$

SYS$*RLIB$.NEXP1$/FOR68

$(1) 007743 010000 $(0) 043723 043723

EXTERNAL REFERENCES: NERRB$, NERRC$

SYS$*RLIB$.FRU$/SYSA9

```
SYS$*RLIB$*[illegible]FS/FOR68
                    $(1)  000001 010040      $(2)  043724 043724
        EXTERNAL REFERENCES: NTAB$, NHPFA$, NRSX$, LOCOD$, NFCHK$, NIO2$,
        NIO2V$, NR91$, FHS1$, FHS2$, NINI1$, NFMT$, NKLN$, NFRA$, NRTR$,
        NF3H$, NSTSV$, NNG90$

SYS$*RLIB$*EXP$/FOR59
                    $(1)  010041 01013n      $(2)  043725 043745
        EXTERNAL REFERENCES: NERRA$, NERRB$

SYS$*RLIB$*NEXP55/FORA8
                    $(1)  010131 010216      $(2)  043746 043755
        EXTERNAL REFERENCES: NERRA$, NERRB$, NERRC$

SYS$*RLIB$*CABS$/FOR59
                    $(1)  010217 010253      $(2)  043756 043760
        EXTERNAL REFERENCES: NERRA$, FSR$1, NERRC$

SYS$*RLIB$*CSQRT$/FOR69
                    $(1)  010254 01033n      $(2)  043761 043764
        EXTERNAL REFERENCES: CABS, FSR$1

SYS$*RLIB$*ALOG$/FOR59
                    $(1)  010331 010450      $(2)  043765 044025
        EXTERNAL REFERENCES: NERRA$

SYS$*RLIB$*SINCOS$/FOR59
                    $(1)  010451 0106n3      $(2)  044026 044047
        EXTERNAL REFERENCES: NERRA$, NERRB$

SYS$*RLIB$*ATAN$/FOR59
                    $(1)  010604 011007      $(2)  044050 044101
        EXTERNAL REFERENCES: NERRB$, NERRA$

SYS$*RLIB$*SQRT$/FOR59
                    $(1)  011010 011050      $(2)  044102 044113
        EXTERNAL REFERENCES: NERRA$

SYS$*RLIB$*CDV$/FOR68
                    $(1)  011051 011146      $(2)  044114 044131
        EXTERNAL REFERENCES: NERRB$, NERRA$, NERRC$

SYS$*RLIB$*NERR$/FOR69
                    $(1)  011147 011547      $(2)  044132 044322
        EXTERNAL REFERENCES: PRINT$, NEE$, EABT$, NS11$

SYS$*RLIB$*NLER$/FOR69
                    $(1)  011550 011731      $(2)  044323 044443
        EXTERNAL REFERENCES: NR93$, NS11$

SYS$*RLIB$*NOBUF$/FOR68
                    $(1)  011732 011772
        EXTERNAL REFERENCES: NTAB$, NHPFA$, NRSX$, LOCOD$, NFCHK$, NERU$,
        PACKT$, NIOER$A, NTSTO$, NIO2V$, NR91$, NBLNK$, FHS1O$, FHS2O$,
        NOTI1$, MR$, NFMT$, WAIT$
```

SYS*RLIB$*[illegible]S/64
$(1) 011773 012041
EXTERNAL REFERENCES: SLTS, LOADS

MSC*PLTLIB*PRINTV/SMICH1
$(1) 012042 012154 $(0) 044444 044452
EXTERNAL REFERENCES: SCCTAB, PLOTS

MSC*PLTLIB*BXLTR
$(0) 044453 045102
EXTERNAL REFERENCES: PRINTV

MSC*PLTLIB*PAC
$(1) 012155 012177

MSC*PLTLIB*UNPAC
$(1) 012200 012214

MSC*PLTLIB*PUT
$(1) 012215 012232

MSC*PLTLIB*SCTZ
$(0) 045103 045202

MSC*PLTLIB*GET
$(1) 012233 012254 $(0) 045203 045204

MSC*PLTLIB*IDF
$(1) 012255 012332 $(2) IDINFO
EXTERNAL REFERENCES: BXLTRS, OUTQZZ

MSC*PLTLIB*PACKZZ
$(1) 012333 012475 $(0) 045205 045243
$(3) IDINFO $(2) BLANK$COMMON
EXTERNAL REFERENCES: OUTQZZ, GET, PUT, OUT6ZZ, NERR6$, NERR2$,
NERR3$
IDINFO(COMMONBLOCK) 045244 045251

MSC*PLTLIB*IDENT
$(1) 012476 013400 $(0) 045252 046370
$(2) IDINFO
EXTERNAL REFERENCES: NENTR$, PCT$, DATE$, OPT$, CSE$, IOW$, COM$,
TIME$, ERR$, TWAIT$, OUTQZZ, ENDJB

MSC*PLTLIB*VECTR/MSC8
$(1) 013402 014001 $(0) 046371 046431
$(3) DEPTH $(2) BLANK$COMMON
EXTERNAL REFERENCES: SCALEX, SCALEY, FIXELT, STOREX, NERR3$

MSC*PLTLIB*CONVR/MSC8
$(1) 014002 014170 $(0) 046432 046467
$(2) BLANK$COMMON
EXTERNAL REFERENCES: BINDEC, NERR3$

MSC*PLTLIB*BINDEC/MSC8
$(1) 014171 014560 $(0) 046470 046551
$(2) BLANK$COMMON

EXTERNAL REFERENCES: NERR3$

MSC*>LTLIB*CNVRT

$(1) 014541 01463n3 $(0) 046552 047105
$(2) BLANK$COMMON

EXTERNAL REFERENCES: OUTOZZ, UNPAC, PAC, GET, PUT, NERR2$, NERR3$

MSC*>LTLIB*BUFRZZ

$(1) 016304 01654n $(0) 047106 047177
$(2) 047200 047447

EXTERNAL REFERENCES: NOP, PEXIT, GATE, PCT$, COM$, IDENT, PACKZZ, IOFRMZ, GET, SCCTZZ, PUT, SWRITE, ERR$

MSC*>LTLIB*PLOTR/MSC8

$(1) 016541 016745 $(2) 047450 047502
$(13) DEPTH $(14) SC4020

EXTERNAL REFERENCES: SCALEX, SCALEY, STOREX, VECTR

MSC*>LTLIB*SCCTAB/SMICH

$(0) 047503 047606

MSC*>LTLIB*RVSX/MSC8

$(1) 016746 017044 $(0) 047607 047622
$(3) DEPTH $(2) BLANK$COMMON
$(4) SC4020

EXTERNAL REFERENCES: FIXFLT, NEXP6$, NERR3$

MSC*>LTLIB*LABELY/MSC8

$(1) 017045 017255 $(0) 047623 047652
$(3) DEPTH $(2) BLANK$COMMON
$(4) SC4020

EXTERNAL REFERENCES: SCALEY, BINDEC, FIXFLT, CONVR, STOREX, NERR3$

MSC*>LTLIB*VLAG/MSC8

$(1) 017256 017410 $(0) 047653 047702
$(3) DEPTH $(2) BLANK$COMMON

EXTERNAL REFERENCES: STOREX, NERR3$

MSC*>LTLIB*LABELX/MSC8

$(1) 017411 017636 $(0) 047703 047733
$(3) DEPTH $(2) BLANK$COMMON
$(4) SC4020

EXTERNAL REFERENCES: SCALEX, BINDEC, FIXFLT, CONVR, STOREX, NERR3$

MSC*>LTLIB*SCALEX/MSC8

$(1) 017636 017735 $(0) 047734 047746
$(3) DEPTH $(2) BLANK$COMMON
$(4) SC4020

EXTERNAL REFERENCES: FIXFLT, ALOG1n, NERR3$

MSC*>LTLIB*COMSCL/MSC8

$(1) 017736 020074 $(0) 047747 047762
$(3) DEPTH $(2) BLANK$COMMON
$(4) SC4020

EXTERNAL REFERENCES: FIXFLT, ALOG10, NERR3$

MSC*?LTLIB*FINDV/MSC8
$(1) 020075 020460 $(0) 047763 050017
$(2) BLANK$COMMON
EXTERNAL REFERENCES: NEXP$$, NERR3$

MSC*?LTLIB*LGRD/MSC8
$(1) 020461 021025 $(0) 050020 050065
$(3) DEPTH $(2) BLANK$COMMON
$(5) LABL $(4) SC4020
EXTERNAL REFERENCES: COMSCL, VLAG, SCALEX, SCALEY, NEXP1$, NERR3$

MSC*?LTLIB*FIXFLT/MSC8
$(1) 021026 021067 $(0) 050066 050076
$(2) BLANK$COMMON
EXTERNAL REFERENCES: NERR3$

MSC*?LTLIB*PLOT$$
$(1) 021070 021077 $(0) 050077 050102
EXTERNAL REFERENCES: FIRST$, CONVT

MSC*?LTLIB*FPACK$/MSC8
$(013)DEPTH $(0) 050103 050231
EXTERNAL REFERENCES: ERR$, PLOT$, PLS#$

MSC*?LTLIB*FMODE$/LLIB11
$(1) 021100 021146 $(4) 050232 050232
$(013)DEPTH
EXTERNAL REFERENCES: FINST$

MSC*?LTLIB*PLOT1/MSC8
$(1) 021147 021272 $(2) 050233 050241
$(013)DEPTH $(014)SC4020
EXTERNAL REFERENCES: COMSCL, ADRSES, SCCTAB, PLOTR

MSC*?LTLIB*PRINT/MSC8
$(1) 021273 021441 $(0) 050242 050276
$(3) DEPTH $(2) BLANK$COMMON
EXTERNAL REFERENCES: HOLTRL, STOREX, NEXP1$, NERR3$
LABL(COMMONBLOCK) 050277 050323
SC4020(COMMONBLOCK) 050324 050546

MSC*?LTLIB*GRID/MSC8
$(1) 021442 022102 $(0) 050547 050605
$(3) DEPTH $(2) BLANK$COMMON
$(5) LABL $(4) SC4020
EXTERNAL REFERENCES: FIXFLT, LGRD, FINDV, COMSCL, SCALEX, LABCLX, VLAG, SCALEY, LABELY, RVSX, RVSY, NERR3$

MSC*?LTLIB*GRAC$/SMICH1
$(1) 022103 022210 $(0) 050606 050610
EXTERNAL REFERENCES: ALOG, PLOT$

MSC*?LTLIB*MRGSET/MSC8
$(1) 022211 022344 $(0) 050611 050662

```
                                        $(2)  BLANK$COMMON

    EXTERNAL REFERENCES: SETHIV, NERR3$

MSC.PLTLIB.NBLANK/MSCR
                        $(1)  022347 022420      $(0)  050663 050702
                                                 $(2)  BLANK$COMMON

        EXTERNAL REFERENCES:  NERR3$

MSC.PLTLIB.NFLNKS/LLIB11
                        $(1)  022421 022562      $(0)  050703 050723
                        $(013)DEPTH
        EXTERNAL REFERENCES:  FDARK$, FLIE$, F#ID$, FNARO$, FCAM1$,
        FCAM2$, FCAM3$, FNCU$, FCUT$, DATE$, FINST$, FRAME$
DEPT1(COMMONBLOCK)                                   050724 050731

MSC.PLTLIB.QUIKHLAN5CR
                        $(1)  022567 023014      $(0)  050732 050762
                        $(3)  DEPTH              $(2)  BLANK$COMMON

        EXTERNAL REFERENCES:  FILMAV, N3LANK, MRGSET, SET.OV, GRID, PRINT,
        PLOTIV, NERR3$

LEC.JR.NSTOPS/JSC
                        $(1)  023017 023044      $(2)  050763 051022

        EXTERNAL REFERENCES:  CON$, EXIT$, NRSF$, REST$, COND$, EABT$,
        IALL$, ERR$, PRINT$

LEC.JR.NINTRS/JSC
                        $(1)  023065 023413      $(2)  051023 051073

        EXTERNAL REFERENCES:  CEND$, FIEL$, PRINT$, NEE$, IALL$
BLANK$COMMON(COMMONBLOCK)

FD36-001121*F*MAIN
                        $(1)  023414 023744      $(0)  051074 051220
                        $(3)  KEEP2              $(2)  BLANK$COMMON
                        $(5)  KEEP4              $(4)  KEEP3
                        $(7)  KEEP6              $(6)  KEEP5
                        $(011)KEEP16             $(010)KEEP11
                                                 $(012)PLT

        EXTERNAL REFERENCES:  LIMIT, RESET, INPUT, CSOLTF, CSMTRX, BHA341,
        BHAD91, ADDZOH, DELZOH, ZEEPL$, BHA272, SRBL, SRBLPP, NONMTX,
        ERROR, NINTR$, NWDU$, NIO2$, NRDU$, NIO3$, NSTOP$

FD36-001121*F*ADDZOH
                        $(1)  023750 023777      $(0)  051221 051232
                        $(3)  KEEP1              $(2)  BLANK$COMMON
                        $(6)  KEEP16             $(4)  KEEP7

        EXTERNAL REFERENCES:  DERR4$, NERR3$

FD36-001121*F*BLOCK
                        $(3)  KEEP2              $(2)  BLANK$COMMON
                        $(5)  KEEP14             $(4)  KEEP6
                        $(7)  KEEP16             $(6)  KEEP15

FD36-001121*F*CSMTRX
                        $(1)  024000 024077      $(0)  051233 051373
                        $(3)  KEEP5              $(2)  BLANK$COMMON
                        $(5)  KEEP10             $(4)  KEEP7
```

EXTERNAL REFERENCES: NWDUS, NIO2$, NIO1$, NERR3$

FD36-001121•F•DELZOH

$(1) 024100 024137 $(0) 051374 051404
$(3) KEEP7 $(2) BLANK$COMMON

EXTERNAL REFERENCES: NERR3$

FD36-001121•F•ERROR

$(1) 024140 024160 $(0) 051405 051415
$(3) KEEP16 $(2) BLANK$COMMON

EXTERNAL REFERENCES: RESET, NWDUS, NIO2$, NERR3$

FD36-001121•F•LIMIT

$(1) 024161 024230 $(0) 051416 051431
$(3) KEEP1 $(2) BLANK$COMMON

EXTERNAL REFERENCES: NERR3$

FD36-001121•F•NOMMIX

$(1) 024231 024267 $(0) 051432 051443
$(3) KEEP1 $(2) BLANK$COMMON
$(5) KEEP10 $(4) KEEP9
$(7) KEEP16 $(6) KEEP13
$(10)CRUD4

EXTERNAL REFERENCES: NERR3$

FD36-001121•F•RESET

$(1) 024270 024401 $(0) 051444 051447
$(3) KEEP2 $(2) BLANK$COMMON
$(5) KEEP4 $(4) KEEP3
$(7) KEEP7 $(6) KEEP5
$(11)KEEP9 $(10)KEEP8
$(13)KEEP11 $(12)KEEP10
$(15)KEEP16 $(14)KEEP13
$(17)KEEP20 $(16)KEEP19
$(20)PLT

EXTERNAL REFERENCES: NERR3$

FD36-001121•F•ZEEPLS

$(1) 024402 024471 $(0) 051450 051563
$(3) KEEP2 $(2) BLANK$COMMON
$(5) CRUD1 $(4) KEEP8

EXTERNAL REFERENCES: NWDUS, NIO1$, NIO2$, NERR3$

KEEP1(COMMONBLOCK) 051564 051611
KEEP2(COMMONBLOCK) 051612 051660
KEEP3(COMMONBLOCK) 051661 051762
KEEP4(COMMONBLOCK) 051763 052245
KEEP5(COMMONBLOCK) 052246 052341
KEEP6(COMMONBLOCK) 052342 052475
KEEP7(COMMONBLOCK) 052476 053407
KEEP8(COMMONBLOCK) 053410 053751
KEEP9(COMMONBLOCK) 053752 054656
KEEP10(COMMONBLOCK) 054657 076476
KEEP11(COMMONBLOCK) 076477 106421
KEEP13(COMMONBLOCK) 106422 106734
KEEP14(COMMONBLOCK) 106735 106765
KEEP15(COMMONBLOCK) 106766 107245
KEEP16(COMMONBLOCK) 107246 107276

KEEP18(COMMONBLOCK) 107277 107300
KEEP19(COMMONBLOCK) 107301 107305
KEEP20(COMMONBLOCK) 107306 107534
KEEP21(COMMONBLOCK) 107535 110667
CRUD1(COMMONBLOCK) 110670 111352
CRUD2(COMMONBLOCK) 111353 115311
CRUD3(COMMONBLOCK) 115312 133426
CRUD4(COMMONBLOCK) 133427 133430
CRUD5(COMMONBLOCK) 133431 133440
PLT(COMMONBLOCK) 133441 133452
PLTARY(COMMONBLOCK) 133453 140212

FD36-001121*F*NICHOL

$(1) 024472 024573 $(0) 140213 140306
$(3) PLT $(2) BLANK$COMMON
$(4) PLTARY

EXTERNAL REFERENCES: QUIKML, NRDU$, NIO2$, NERR3$

SEGMENT B1* 024574 030034 140307 140623
FOLLOWS SEGMENT A

FD36-001121*F*AFTVAR

$(1) 024574 024621 $(0) 140307 140312
$(3) KEEP1A $(2) BLANK$COMMON

EXTERNAL REFERENCES: NERR4$, NERR3$

FD34-001121*F*DATA

$(1) 024622 025400 $(0) 140313 140353
$(3) KEEP1 $(2) BLANK$COMMON
$(5) KEEP3 $(4) KEEP2
$(7) KEEP5 $(6) KEEP4
$(11) KEEP9 $(10) KEEP6
$(13) CRUD2 $(12) KEEP16
$(14) PLT

EXTERNAL REFERENCES: RESET, INPNYQ, INPRL, INPEST, INPMTX, NRDU$, NIO3$, NIO2$, NIO1$, NERR4$, NERR3$

FD36-001121*F*GENMTX

$(1) 025406 026165 $(0) 140354 140425
$(3) KEEP1 $(2) BLANK$COMMON
$(5) KEEP16 $(4) KEEP10
$(6) CRUD2

EXTERNAL REFERENCES: NRDU$, NIO1$, NIO2$, NERR4$, NERR3$

FD3A-001121*F*INPEST

$(1) 026163 026306 $(0) 140426 140443
$(3) KEEP1 $(2) BLANK$COMMON
$(5) KEEP9 $(4) KEEP2
$(7) CRUD2 $(6) KEEP16

EXTERNAL REFERENCES: NRDU$, NIO2$, NIO1$, NERR4$, NERR3$

FD36-001121*F*INPMTX

$(1) 026307 026411 $(0) 140444 140447
$(3) KEEP2 $(2) BLANK$COMMON
$(5) CRUD2 $(4) KEEP16

EXTERNAL REFERENCES: GENMTX, RAWMTX, NERR4$, NERR3$

FD36-001121*F*INPNYQ

$(1) 024412 026564 $(0) 140450 140466
$(3) KEEP1 $(2) BLANK$COMMON
$(5) KEEP3 $(4) KEEP2
$(7) CRUD2 $(6) KEEP16

EXTERNAL REFERENCES: NRDU$, NI02$, NI01$, NERR4$, NERR3$

FD36-001121*F*INPRL

$(1) 026566 027262 $(0) 140467 140516
$(3) KEEP1 $(2) BLANK$COMMON
$(5) KEEP4 $(4) KEEP2
$(7) CRUD2 $(6) KEEP16

EXTERNAL REFERENCES: STNRL, NRDU$, NI02$, NI01$, NERR4$, NERR3$

FD36-001121*F*INPUT

$(1) 027263 027334 $(0) 140517 140522
$(3) KEEP2 $(2) BLANK$COMMON
$(4) KEEP16

EXTERNAL REFERENCES: AFTVAR, PREVAR, DATA, NERR4$, NERR3$

FD36-001121*F*PREVAR

$(1) 027335 027401 $(0) 140523 140526
$(3) KEEP2 $(2) BLANK$COMMON
$(4) KEEP16

EXTERNAL REFERENCES: VRYGEN, VRYRAW, NERR4$, NERR3$

FD36-001121*F*RAWMTX

$(1) 027402 027422 $(0) 140527 140532
$(3) KEEP16 $(2) BLANK$COMMON

EXTERNAL REFERENCES: NERR4$, NERR3$

FD36-001121*F*STNRL

$(1) 027423 027553 $(0) 140533 140566
$(3) KEEP2 $(2) BLANK$COMMON
$(4) KEEP4

EXTERNAL REFERENCES: NERR3$

FD36-001121*F*VRYGEN

$(1) 027554 030013 $(0) 140567 140617
$(3) KEEP1 $(2) BLANK$COMMON
$(5) KEEP13 $(4) KEEP10
$(7) CRUD2 $(6) KEEP16

EXTERNAL REFERENCES: NRDU$, NI02$, NI01$, NERR4$, NERR3$

FD36-001121*F*VRYRAW

$(1) 030014 030034 $(0) 140620 140623
$(3) KEEP16 $(2) BLANK$COMMON

EXTERNAL REFERENCES: NERR4$, NERR3$

SEGMENT 82* 024574 030365 140307 141164
FOLLOWS SEGMENT A

FD36-001121*F*AGAIN

```
                    $(1)  024574 025044      $(0)  140307 140371
                    $(3)  KEEP1              $(2)  BLANK$COMMON
                    $(5)  KEEP14             $(4)  KEEP14
                                             $(6)  CRUD3

        EXTERNAL REFERENCES:  PEVAL, DET, CDABV, SCALE, CDVS, NWDUS,
    NI02$, NERR3$

FD36-001121-F-CDABV
                    $(1)  025047 025134      $(0)  140372 140407
                                             $(2)  BLANK$COMMON

        EXTERNAL REFERENCES:  SQRT, NERR3$

FD36-001121-F-DECIDE
                    $(1)  025135 025270      $(0)  140410 140416
                    $(3)  KEEP9              $(2)  BLANK$COMMON
                                             $(4)  CRUD3

        EXTERNAL REFERENCES:  CDABV, CDVS, NERR3$

FD36-001121-F-DET
                    $(1)  025271 025335      $(0)  140417 140424
                    $(3)  KEEP19             $(2)  BLANK$COMMON

        EXTERNAL REFERENCES:  DET5D, DETC5, NERR3$

FD36-001121-F-ESTMAT
                    $(1)  025336 025623      $(0)  140425 140464
                    $(3)  KEEP1              $(2)  BLANK$COMMON
                    $(5)  KEEP15             $(4)  KEEP9
                                             $(6)  CRUD3

        EXTERNAL REFERENCES:  CDABV, NWDU$, NI02$, NERR3$

FD36-001121-F-FIXIT
                    $(1)  025624 025672      $(0)  140465 140476
                                             $(2)  BLANK$COMMON

        EXTERNAL REFERENCES:  XPR1, CDVS, NERR3$

FD36-001121-F-KCALC
                    $(1)  025673 025770      $(0)  140477 140556
                    $(3)  KEEP5              $(2)  BLANK$COMMON
                                             $(4)  KEEP19

        EXTERNAL REFERENCES:  PEVAL, DET, CDVS, NWDUS, NI02$, NERR3$

FD36-001121-F-PEVAL
                    $(1)  025771 026347      $(0)  140557 140610
                    $(3)  KEEP9              $(2)  BLANK$COMMON
                    $(5)  CRUD3              $(4)  KEEP14

        EXTERNAL REFERENCES:  CDABV, CDVS, NERR3$

FD36-001121-F-ROOTER
                    $(1)  026350 026675      $(0)  140611 140672
                    $(3)  KEEP1              $(2)  BLANK$COMMON
                    $(5)  KEEP16             $(4)  KEEP9
                                             $(6)  CRUD3

        EXTERNAL REFERENCES:  ESTMAT, START, DECIDE, AGAIN, CDABV, SOLVE,
    SAVE, NWDUS, NI02$, NERR3$

FD36-001121-F-SAVE
                    $(1)  026674 027034      $(0)  140673 140741
```

```
                    $(3)  KEEP1                $(2)  BLANK$COMMON
                    $(5)  KEEP16               $(4)  KEEP9
                                               $(6)  CRUD3
      EXTERNAL REFERENCES:  NWDUS, NIO2$, NERR3$

FD36-001121*F*SCALE
                    $(1)  027035 027252        $(0)  140742 140762
                                               $(2)  BLANK$COMMON
      EXTERNAL REFERENCES:  FIXIT, NERR3$

FD36-001121*F*SOLVE
                    $(1)  027253 030027        $(0)  140763 141034
                    $(3)  KEEP14               $(2)  BLANK$COMMON
                    $(5)  CRUD1                $(4)  KEEP16
                                               $(6)  CRUD3
      EXTERNAL REFERENCES:  CDABV, CDVS, CSQRT, NWDU$, NIO2$, NERR3$

FD36-001121*F*START
                    $(1)  030030 030342        $(0)  141035 141157
                    $(3)  KEEP14               $(2)  BLANK$COMMON
                    $(5)  CRUD3                $(4)  KEEP16
      EXTERNAL REFERENCES:  CDABV, PEVAL, DET, SCALE, CDVS, NWDUS,
      NIO2$, NERR3$

FD36-001121*F*XCNG
                    $(1)  030343 030365        $(0)  141160 141164
                                               $(2)  BLANK$COMMON
      EXTERNAL REFERENCES:  NERR3$

      SEGMENT C1*           030366 033343      141165 142056
      FOLLOWS SEGMENT B2

FD36-001121*F*CLPOLE
                    $(1)  030366 030426        $(0)  141165 141174
                    $(3)  KEEP6                $(2)  BLANK$COMMON
                    $(5)  KEEP15               $(4)  KEEP9
                                               $(6)  KEEP16
      EXTERNAL REFERENCES:  COMPUT, NERR4$, NERR3$

FD36-001121*F*COMPUT
                    $(1)  030427 030611        $(0)  141175 141210
                    $(3)  KEEP1                $(2)  BLANK$COMMON
                    $(5)  KEEP9                $(4)  KEEP5
                    $(7)  KEEP16               $(6)  KEEP10
                    $(011)CRUD3                $(010)KEEP19
                                               $(012)CRUD5
      EXTERNAL REFERENCES:  RLOCUS, KCALC, NWDUS, NIO2$, NERR4$, NERR3$

FD36-001121*F*CSOLTF
                    $(1)  030612 031107        $(0)  141211 141240
                    $(3)  KEEP5                $(2)  BLANK$COMMON
                    $(5)  KEEP10               $(4)  KEEP7
                    $(7)  KEEP16               $(6)  KEEP15
                    $(011)KEEP19               $(010)KEEP18
                                               $(012)CRUD5
```

EXTERNAL REFERENCES: PRINTT, POLES, FORM, ZEROS, CLPOLE, XPRI, CDVS, NERR4$, NERR3$

FD36-001121*F*DETCS

$(1)	031102 031503	$(0)	141241 141321
$(3)	KEEP10	$(2)	BLANK$COMMON
$(5)	CRUD3	$(4)	KEEP14

EXTERNAL REFERENCES: FRMTX, CDVS, NERR3$

FD36-001121*F*FORM

$(1)	031504 032060	$(0)	141322 141365
$(3)	KEEP9	$(2)	BLANK$COMMON
		$(4)	CRUD1

EXTERNAL REFERENCES: NERR3$

FD36-001121*F*FRMTX

$(1)	032061 032270	$(0)	141366 141430
$(3)	KEEP10	$(2)	BLANK$COMMON
$(5)	KEEP16	$(4)	KEEP19
$(7)	CRUD3	$(6)	CRUD2

EXTERNAL REFERENCES: NERR3$

FD36-001121*F*POLES

$(1)	032271 032331	$(0)	141431 141440
$(3)	KEEP6	$(2)	BLANK$COMMON
$(5)	KEEP15	$(4)	KEEP9
		$(6)	KEEP16

EXTERNAL REFERENCES: COMPUT, NERR4$, NERR3$

FD36-001121*F*PRINTT

$(1)	032332 032571	$(0)	141441 141543
$(3)	KEEP5	$(2)	BLANK$COMMON
$(5)	KEEP10	$(4)	KEEP6
$(7)	CRUD2	$(6)	KEEP16

EXTERNAL REFERENCES: NWDUS, NIO3$, NIO2$, NIO1S, NERR4S, NERR3$

FD36-001121*F*PUTOUT

$(1)	032572 033135	$(0)	141544 141744
$(3)	KEEP5	$(2)	BLANK$COMMON
$(5)	KEEP9	$(4)	KEEP6
$(7)	KEEP16	$(6)	KEEP14
$(10)	CRUD1	$(8)	KEEP19
		$(12)	CRUD5

EXTERNAL REFERENCES: XCNG, NWDUS, NIO3S, NIO2$, NIO1S, CABS, NERR3$

FD36-001121*F*RLOCUS

$(1)	033136 033302	$(0)	141745 142046
$(3)	KEEP10	$(2)	BLANK$COMMON
		$(4)	KEEP16

EXTERNAL REFERENCES: ROOTER, PUTOUT, NWDUS, NIO2$, NIO1S, NERR4$, NERR3$

FD36-001121*F*ZEROS

$(1)	033303 033347	$(0)	142047 142056
$(3)	KEEP6	$(2)	BLANK$COMMON
$(5)	KEEP15	$(4)	KEEP9

```
                                        $(6)  KEEP16
    EXTERNAL REFERENCES:  COMPUT, NERR4$, NERR3$

            SEGMENT C2*          030366 033506      141165 142522
            FOLLOWS SEGMENT BZ

SYS*RLIB$,CEXP$/FORS9
                $(1)  030366 030446      $(2)  141165 141173
    EXTERNAL REFERENCES:  EXP, SIN, COS

FD36-001121*F*DHA272
                $(1)  030446 030772      $(0)  141174 141232
                $(3)  KEEP1              $(2)  BLANK$COMMON
                $(5)  KEEP4              $(4)  KEEP2
                $(7)  KEEP9              $(6)  KEEP5
                $(011)KEEP15             $(010)KEEP10
                $(013)KEEP19             $(012)KEEP16
                $(015)CRUD2              $(014)KEEP21
                                         $(016)CRUD3
    EXTERNAL REFERENCES:  SAVEGT, GETEST, RUTER, SAVRGT, RLPLOT,
    NERR4$, NERR3$

FD36-001121*F*DETSD
                $(1)  030773 031656      $(0)  141233 141540
                $(3)  KEEP5              $(2)  BLANK$COMMON
                $(5)  KEEP19             $(4)  KEEP14
                $(7)  CRUD2              $(6)  KEEP21
    EXTERNAL REFERENCES:  CDABV, XPRI, CDV$, CEXP, NERR3$

FD36-001121*F*GETEST
                $(1)  031657 031713      $(0)  141541 141550
                $(3)  KEEP9              $(2)  BLANK$COMMON
                                         $(4)  KEEP20
    EXTERNAL REFERENCES:  NERR3$

FD36-001121*F*RLPLOT
                $(1)  031714 031726      $(0)  141551 141554
                $(3)  KEEP16             $(2)  BLANK$COMMON
    EXTERNAL REFERENCES:  NERR3$

FD36-001121*F*RLPRNT
                $(1)  031727 032430      $(0)  141555 142076
                $(3)  KEEP5              $(2)  BLANK$COMMON
                $(5)  KEEP9              $(4)  KEEP6
                $(7)  KEEP16             $(6)  KEEP14
                $(011)CRUD1              $(010)KEEP19
                                         $(012)CRUD2
    EXTERNAL REFERENCES:  XCNG, DET, PEVAL, NWDU$, NIO3$, NIO2$, CABS,
    ATAN2, ALOG, SQRT, CDV$, NERR3$

FD36-001121*F*RUTER
                $(1)  032431 032612      $(0)  142077 142125
                $(3)  KEEP1              $(2)  BLANK$COMMON
                $(5)  KEEP4              $(4)  KEEP2
                $(7)  KEEP10             $(6)  KEEP9
```

```
                $(011)KEEP19            $(010)KEEP16
                                        $(012)CRUD2
    EXTERNAL REFERENCES:  ROOTER, RLPRNT, NWDU$, NIO2$, NERR4$,
    NERR3$

FD36-001121*F*SAVRUT
                $(1)  032413 032712     $(0)  142126 142142
                $(3)  KEEP1             $(2)  BLANK$COMMON
                $(5)  KEEP9             $(4)  KEEP5
                $(7)  KEEP16            $(6)  KEEP11
    EXTERNAL REFERENCES:  NERR3$

FD36-001121*F*SRRL
                $(1)  032713 033134     $(0)  142143 142216
                $(3)  KEEP1             $(2)  BLANK$COMMON
                $(5)  KEEP5             $(4)  KEEP4
                $(7)  KEEP11            $(6)  KEEP9
                $(011)KEEP16            $(010)KEEP15
                $(013)KEEP21            $(012)KEEP19
                $(015)CRUD3             $(014)CRUD2
    EXTERNAL REFERENCES:  SAVEST, GETEST, ROOTER, NWDU$, NIO2$,
    NERR4$, NERR3$

FD36-001121*F*SRRLPP
                $(1)  033135 033426     $(0)  142217 142263
                $(3)  KEEP2             $(2)  BLANK$COMMON
                $(5)  KEEP5             $(4)  KEEP4
                $(7)  KEEP11            $(6)  KEEP9
                $(011)KEEP19            $(010)KEEP16
                $(013)CRUD3             $(012)CRUD2
    EXTERNAL REFERENCES:  RLPRNT, RLPLOT, NERR3$

FD36-001121*F*SAVEST
                $(1)  033427 033506     $(0)  142264 142522
                $(3)  KEEP9             $(2)  BLANK$COMMON
    EXTERNAL REFERENCES:  NERR3$

        SEGMENT 83*             024574 032442   140307 141516
        FOLLOWS SEGMENT A

FD36-001121*F*BHA341
                $(1)  024574 025063     $(0)  140307 140411
                $(3)  KEEP4             $(2)  BLANK$COMMON
                $(5)  KEEP6             $(4)  KEEP5
                $(7)  KEEP3             $(6)  KEEP7
                $(011)KEEP20            $(010)KEEP16
                $(013)CRUD2             $(012)KEEP21
                                        $(014)CRUD4
    EXTERNAL REFERENCES:  PFE, PFEZRO, NWDU$, NIO2$, NIO3$, NIO1$,
    NERR4$, NERR3$

FD36-001121*F*MLTZRO
                $(1)  025064 026016     $(0)  140412 140630
                $(3)  KEEP5             $(2)  BLANK$COMMON
                $(5)  KEEP16            $(4)  KEEP14
```

```
                $(7)  CRUD1              $(6)  KEEP21
                $(11) CRUD4              $(10) CRUD2
    EXTERNAL REFERENCES:  N$DU$, NIO2$, NERR4$, NERR3$

FD36-001121*F*MODZRO
                $(1)  026017 027113      $(0)  140631 141055
                $(3)  KEEP5              $(2)  BLANK$COMMON
                $(5)  KEEP16             $(4)  KEEP14
                $(7)  CRUD1              $(6)  KEEP21
                $(11) CRUD4              $(10) CRUD2
    EXTERNAL REFERENCES:  N$DU$, NIO2$, NERR4$, NERR3$

FD36-001121*F*MZTRAN
                $(1)  027113 030369      $(0)  141056 141235
                $(3)  KEEP5              $(2)  BLANK$COMMON
                $(5)  KEEP14             $(4)  KEEP8
                $(7)  KEEP21             $(6)  KEEP20
                $(11) CRUD2              $(10) CRUD1
                                         $(12) CRUD4
    EXTERNAL REFERENCES:  N$DU$, NIO2$, EXP, SIN, COS, CSQRT, CABS,
    NERR3$

FD36-001121*F*PFE
                $(1)  030363 030736      $(0)  141236 141301
                $(3)  KEEP5              $(2)  BLANK$COMMON
                $(5)  CRUD2              $(4)  KEEP7
    EXTERNAL REFERENCES:  SEPDPC, MZTRAN, ZTRAN, CDV$, NERR3$

FD36-001121*F*PFEZRO
                $(1)  030737 031252      $(0)  141302 141326
                $(3)  KEEP5              $(2)  BLANK$COMMON
                $(5)  KEEP16             $(4)  KEEP7
                $(7)  CRUD2              $(6)  KEEP18
    EXTERNAL REFERENCES:  MODZRO, MLTZRO, XPRI, NERR4$, NERR3$

FD36-001121*F*SEPDPC
                $(1)  031253 031276      $(0)  141327 141336
                                         $(2)  BLANK$COMMON
    EXTERNAL REFERENCES:  NERR3$

FD36-001121*F*ZTRAN
                $(1)  031277 032442      $(0)  141337 141516
                $(3)  KEEP5              $(2)  BLANK$COMMON
                $(5)  KEEP14             $(4)  KEEP8
                $(7)  KEEP21             $(6)  KEEP20
                $(11) CRUD2              $(10) CRUD1
                                         $(12) CRUD4
    EXTERNAL REFERENCES:  N$DU$, NIO2$, EXP, COS, SIN, CSQRT, CABS,
    NERR3$

        SEGMENT 84*          024574 031204    140307 141404
        FOLLOWS SEGMENT A

SYS$*RL19$*TANCOTANS/FOR59
                $(1)  024574 024771      $(2)  140307 140327
```

EXTERNAL REFERENCES: NERR8$, NERR4$, NERRC$

FD36-001121*F*BEGIN

$(1) 024772 025065 $(0) 140330 140342
$(3) KEEP2 $(2) BLANK$COMMON
$(5) KEEP5 $(4) KEEP3
$(6) CRUD3

EXTERNAL REFERENCES: NERR3$

FD36-001121*F*BHAD91

$(1) 025064 025164 $(0) 140343 140351
$(3) KEEP3 $(2) BLANK$COMMON
$(4) CRUD3

EXTERNAL REFERENCES: INITAL, WRITE, BEGIN, TRNSFR, TEST, OUTPUT, POINT, SUMMRY, GRAPHS, NERR2$, NERR4$, NERR3$

FD36-001121*F*BODE

$(1) 025165 025174 $(0) 140352 140355
$(2) BLANK$COMMON

EXTERNAL REFERENCES: NERR3$

FD36-001121*F*DB

$(1) 025175 025225 $(0) 140356 140364
$(2) BLANK$COMMON

EXTERNAL REFERENCES: ALOG10, NERR3$

FD36-001121*F*GRAPHS

$(1) 025224 025264 $(0) 140365 140379
$(3) KEEP2 $(2) BLANK$COMMON
$(5) KEEP16 $(4) KEEP3
$(6) PLT

EXTERNAL REFERENCES: NYQIST, BODE, NICHOL, NERR3$

FD36-001121*F*INITAL

$(1) 025265 025343 $(0) 140371 140400
$(3) KEEP2 $(2) BLANK$COMMON
$(5) KEEP5 $(4) KEEP3
$(7) CRUD3 $(6) KEEP16

EXTERNAL REFERENCES: STNNYQ, NERR4$, NERR3$

FD36-001121*F*INT1

$(1) 025364 025421 $(0) 140401 140410
$(2) BLANK$COMMON

EXTERNAL REFERENCES: NERR3$

FD36-001121*F*INT2

$(1) 025422 025437 $(0) 140411 140415
$(2) BLANK$COMMON

EXTERNAL REFERENCES: NERR3$

FD36-001121*F*NYQIST

$(1) 025440 025447 $(0) 140416 140421
$(2) BLANK$COMMON

EXTERNAL REFERENCES: NERR3$

FD36-001121*F*OUTPUT

$(1) 025450 026166 $(0) 140422 140523

```
$(3)  KEEP3                $(2)  BLANK$COMMON
$(5)  KEEP14               $(4)  KEEP6
$(7)  PLT                  $(6)  CRUD3
                           $(10) PI_TABY
    EXTERNAL REFERENCES:  SYSFRQ, NMDU$, NIO3$, NIO2$, SQRT, ATAN2,
    NERR2$, ALOG10, NIO1$, NERR3$

FD36-001121*F*POINT
$(1)  026157 027207        $(0)  140524 140545
$(3)  KEEP1                $(2)  BLANK$COMMON
$(5)  KEEP3                $(4)  KEEP2
$(7)  KEEP14               $(6)  KEEP5
$(11) CRUD3                $(10) CRUD1
    EXTERNAL REFERENCES:  INT1, INT2, COS, NERR3$

FD36-001121*F*POLVAL
$(1)  027210 027337        $(0)  140546 140572
$(3)  KEEP14               $(2)  BLANK$COMMON
$(5)  CRUD3                $(4)  KEEP15
    EXTERNAL REFERENCES:  SYSFRQ, NERR3$

FD36-001121*F*SINNYQ
$(1)  027340 027400        $(0)  140573 140613
$(3)  KEEP2                $(2)  BLANK$COMMON
$(5)  KEEP5                $(4)  KEEP3
    EXTERNAL REFERENCES:  NERR3$

FD36-001121*F*SUMMRY
$(1)  027401 030474        $(0)  140614 141213
$(3)  KEEP1                $(2)  BLANK$COMMON
$(5)  KEEP6                $(4)  KEEP2
$(7)  CRUD3                $(6)  CRUD1
    EXTERNAL REFERENCES:  DB, SYSFRQ, NMDU$, NIO3$, NIO2$, NIO1$,
    NERR3$

FD36-001121*F*SYSFRQ
$(1)  030475 030520        $(0)  141214 141222
$(3)  KEEP5                $(2)  BLANK$COMMON
    EXTERNAL REFERENCES:  COTAN, NERR3$

FD36-001121*F*TEST
$(1)  030521 030746        $(0)  141223 141242
$(3)  KEEP14               $(2)  BLANK$COMMON
                           $(4)  CRUD3
    EXTERNAL REFERENCES:  SQRT, ATAN2, NERR3$

FD36-001121*F*TRNSFR
$(1)  030747 031042        $(0)  141243 141262
$(3)  KEEP2                $(2)  BLANK$COMMON
$(5)  CRUD2                $(4)  KEEP3
                           $(6)  CRUD3
    EXTERNAL REFERENCES:  POLVAL, COV$, NERR3$

FD36-001121*F*WRITE
$(1)  031043 031204        $(0)  141263 141404
$(3)  KEEP2                $(2)  BLANK$COMMON
$(5)  KEEP6                $(4)  KEEP3
```

$(7) CRUD1 $(6) KEEP16
$(10)CRUD3

EXTERNAL REFERENCES: N@DUS, NIO3$, NIO2$, NIO1$, NERR3$

IBANK DRAWN TO SCALE: 200 WORDS DECIMAL PER DASH

A (13108)

B4* (2313)

B3* (2983)

B2* (1914)

C2* (1617)

C1* (1518)

B1* (1697)

DBANK DRAWN TO SCALE: 500 WORDS DECIMAL PER DASH

A (32711)

B4* (574)

B3* (648)

B2* (430)

C2* (734)

C1* (442)

B1* (205)

INDIRECT LOAD TABLE

CALLS ON THE FOLLOWING IBANK ENTRY POINTS IN INDIRECT LOAD SEGMENTS ARE ROUTED VIA THESE INDIRECT LOAD ADDRESSES, TO INSURE SEGMENTS ARE LOADED

AFTVAR	040034	AGAIN	040037	BEGIN	040042
BHA@?1	040045	BHA2?2	040050	BHA3$1	040053
BODE	040056	CDABV	040061	CEXP	040064
CLPOLE	040067	COMPUT	040072	COTAN	040075
CSOLTF	040100	DATA	040103	DB	040106
DECIDE	040111	DET	040114	DETCS	040117
DETSO	040122	ESTIAT	040125	FIXIT	040130
FORM	040133	FRQTA	040136	GENMTX	040141
GETEST	040144	GRAPHS	040147	INITAL	040152

INPEST	040155	INPMTX	040160	INPNTQ	040163
INPPL	040166	INPUT	040171	INT1	040174
INT2	040177	KCALC	040202	MLTZRO	040205
MODZRO	040210	MZTRAN	040213	NYQIST	040216
OUTPUT	040221	PEVAL	040224	PFE	040227
PFEZRO	040232	POINT	040235	POLES	040240
POLVAL	040243	PREVAR	040246	PRINTT	040251
PUTOUT	040254	RANMTX	040257	RLOCUS	040262
RLPLOT	040265	RLPRNT	040270	ROOTER	040273
RUTER	040276	SAVE	040301	SAVEST	040304
SAVRUT	040307	SCALE	040312	SEPDPC	040315
SOLVE	040320	SRRL	040323	SRRLPP	040326
START	040331	STNRYQ	040336	STNRL	040337
SUMRTY	040342	SYSERQ	040345	TEST	040350
TRNSFR	040353	VRYGEN	040356	VRYRAN	040361
WRITE	040365	XCNG	040367	ZEROS	040372
ZTRAN	040375				

EXTERNAL DEFINITIONS (REFERENCED ENTRY POINTS MARKED *)(UNDEFINED SYMBOLS MARKED ?)(LOCAL SYMBOLS MARKED L)

ABORT$(ERU$)	000000000012	ABR$(ERU$)	000000000027	ABSAD$(ERU$)	0000000003o
ABS$(ERU$)	000000000014	ACLIST$(ERU$)	000000000141	ACOS$(ERU$)	000000001020
ACQ$(ERU$)	000000000016	ACSF$(ERU$)	000000000140	ACT$(ERU$)	000000000147
ADACT$(ERU$)	000000000154	*ADDZOH(ADDZOH)	023771	ADED$(ERU$)	000000000161
*ADPSE$(PLOTR)	047450	*AFTVAR(AFTVAR)	024616	*AGAIN(AGAIN)	025040
ALGER$(ERU$)	000000001007	ALG10$(ERU$)	000000001001	*ALOG(ALOG$)	010340
ALOGC$(ALOG$)	010335	ALOGN$(ERU$)	000000001002	ALOG$(ERU$)	000000001003
*ALOG10(ALOG$)	010332	ALOG10$(ERU$)	000000001001	ALTPK2(NOUT$)	043111
APCHCA$(ERU$)	000000000077	APCHCN$(ERU$)	000000000075	APNCHA$(ERU$)	000000000073
APRINT$(ERU$)	000000000070	APRNTA$(ERU$)	000000000071	APRTCA$(ERU$)	000000000076
APRTCN$(ERU$)	000000000074	APUNCH$(ERU$)	000000000072	ARDA$(ERU$)	000000000167
AREADA$(ERU$)	000000000167	AREAD$(ERU$)	000000000164	ARRE(NOUT$)	043126
ARRN(NOUT$)	043127	ASIN$(ERU$)	000000001021	ATAN(ATAN$)	010646
ATAN$(ERU$)	000000001017	*ATAN2(ATAN$)	010605	ATAN2$(ERU$)	000000001016
ATPD$(ERU$)	000000000170	ATREAD$(ERU$)	000000000170	AWAIT$(ERU$)	000000000134
AICA$(NFCHK$)	043537	BANK$(ERU$)	000000000160	BBEDF$(ERU$)	000000000036
*BEGIN(BEGIN)	025061	BFCIND(NFCHK$)	043530	BFCTL$(NFCHK$)	000000000050
*BHA09(BHA091)	025157	*BHA272(BHA272)	030757	*BHA341(BHA341)	025055
BIGV(NFLNK$)	022425	*BINDEC(BINDEC)	014543	*BL$(NOFOD$)	000000000400
*BODE(BODE)	025173	BRDS(ERU$)	000000000024	BRITEV(GRACS)	022111
BSD$(ERU$)	000000000035	BSR$(ERU$)	000000000037	*BSIBL$(NBSBL$)	002517
*BXLTR$(BXLTR)	044673	*BIL$(NBFOO$)	000000002202	*BIO$(NBFOO$)	040637
B2L$$UNDEFINED!	000000000000 ?	B2O$$UNDEFINED!	000000000000 ?	*CABS(CABS$)	010220
CA?S$(ERU$)	000000001013	CAUD$(ERU$)	000000000057	CAMRAV(NFLNK$)	022566
CAMRA1(NFLNK$)	022431	CAMRA2(NFLNK$)	*022433	CAMRA3(NFLNK$)	022435
CBRT$(ERU$)	000000001007	CCBRT$(ERU$)	000000001004	CCOSH$(ERU$)	000000001005
CCOS$(ERU$)	000000001007	*CDARV(CDARV)	025131	CDIV(CDV$)	011052
CDIV$(ERU$)	000000001014	*CDV$(CDV$)	011052	*CEND$(ERU$)	000000000100
*CEXP(CEXP$)	030366	CEXP$(ERU$)	000000001002	*CFE(NFCHK$)	007131
CGET$(ERU$)	000000000054	CJOIN$(ERU$)	000000000151	CLEANV(BUFRZZ)	016444
CLEAN$(BUFRZZ)	016445	CLIST$(ERU$)	000000000153	CLOG$(ERU$)	000000001001
CLOSE(NCLOS$)	002214	*CLOSES(NCLOSS)	002347	*CLPOLE(CLPOLE)	030421
CMD$(ERU$)	000000000051	CMH$(ERU$)	000000000052	CMI$(ERU$)	000000000047
CMO$(ERU$)	000000000050	CMSA$(ERU$)	000000000053	CMS$(ERU$)	000000000045
CMT$(ERU$)	000000000046	CNTNG$(ERU$)	000000001004	*COMPUT(COMPUT)	030606
*CONSCL(CONSCL)	020062	*CONS(ERU$)	000000000010	*COND$(ERU$)	000000000066
*CONVR(CONVR)	014145	*CONVT(CONVRT)	016243	*COS(SINCOS$)	010455

```
COSH$(ERU$)                    0000000001023
*COTAN(TANCOTANS)                   024575
CREL$(ERU$)                    0000000DC152
CSINH$(ERU$)                   0000000001006
*CSQLTF(CSQLTF)                     031072
CTANH$(ERU$)                   0000000001011
CTSQ$(ERU$)                    0000000000122
DACOS$(ERU$)                   0000000001015
DARK(NFLNK$)                        022421
DATAN$(ERU$)                   0000000001014
DATN2$(ERU$)                   0000000001013
DB$(ERU$)                      0000000001036
DCOSH$(ERU$)                   0000000001020
DCOTN$(ERU$)                   0000000001011
*DELZOH(DELZOH)                     024133
*DETSD(DETSD)                      031633
DGETP$(ERU$)                   0000000000016
DLAP$(ERU$)                    0000000000032
DLOG$(ERU$)                    0000000001002
*DRAIN$(NWBLK$)                     002405
DSINH$(ERU$)                   0000000001017
DTANH$(ERU$)                   0000000001021
DS(ERU$)                       0000000000002
*ENDJB(BUFRZZ)                      016444
ENDTV(BUFRZZ)                       016540
ERR.DS(NINTR$)                      023065
*EXIT(NSTOP$)                       023025
*EXP(EXP$)                          010044
EXP$(ERU$)                     0000000001004
FACIT$(ERU$)                   0000000000143
*FCAM2$(FMODE$)                     021116
*FDARK$(FMODE$)                     021106
*FNS10$(NOUT$)                      043117
*FIELD$(NERR$)                      011517
*FINST$(FPACK$)                     050103
*FIXFLT(FIXFLT)                     021066
FLINE$(FPACK$)                      050144
*FNARO$(FMODE$)                     021102
FORK$(ERU$)                    0000000000013
FORMV(NFLNK$)                       022564
FRAMEV(NFLNK$)                      022443
*FSRS1(FSRS1$)                      004014
*GENMTX(GENMTX)                     026146
*GRAPHS(GRAPHS)                     025263
G..$(ERU$)                     0000000000015
*HOLTBL(SCOTAB)                     047503
*IDENT(IDENT)                       012476
*IDJS(IDLS)                         011773
ILGOP$(ERU$)                   0000000001010
*INPEST(INPEST)                     026300
*INPRL(INPRL)                       027252
INTMSG(NINTR$)                      023374
*INT2(INT2)                         026436
*IOCOD$(NIER$)                      044343
IOI$(ERU$)                     0000000000062
IOXI$(ERU$)                    0000000000025
*LABELX(LABELX)                     017622

COSN$(ERU$)                    0000000001013
COTAN$(ERU$)                   0000000001014
CRTN$(ERU$)                         000000035
CSIN$(ERU$)                    0000000001010
*CSQRT(CSQRT$)                      010255
CTAN$(ERU$)                    0000000001012
CTS$(ERU$)                          000123
DACT$(ERU$)                         000150
DASIN$(ERU$)                   0000000001016
DATAN2$(ERU$)                  0000000001013
*DB(DB)                             025222
DBLT$$(ERU$)                   0000000000030
DCOS$(ERU$)                    0000000001007
DCYC$(ERU$)                    0000000000034
*DET(DET)                           025321
DEXPN$(ERU$)                   0000000001004
DGET$(ERU$)                    0000000000015
DLGIO$(ERU$)                   0000000001001
DLOGIO$(ERU$)                  0000000001001
DREAD$(ERU$)                   0000000000020
DSIN$(ERU$)                    0000000001010
DTAN$(ERU$)                    0000000001012
*EABT$(ERU$)                        000000026
END.OB(IDENT)                       013210
ERCLO$(ERU$)                   0000000001005
*ERR$(ERU$)                    0000000000040
*EXIT$(ERU$)                   0000000000011
EXPAND(NFLNK$)                      022425
EXSNP$(ERU$)                   0000000000132
FAINTV(GRAC$)                       022113
*FCAM3$(FMODE$)                     021120
FFORK$(ERU$)                   0000000000131
*FHS2$(NINPT$)                      043050
*FILMAV(NFLNK$)                     022443
*FIRST$(BUFRZZ)                     016311
*FIXIT(FIXIT)                       025661
*FLITE$(FMODE$)                     021104
*FNCTB$(NFCHK$)                     043470
*FORM(FORM)                         032022
FRCDO(NOUT$)                        004405
*FRAME$(FPACK$)                     050135
*FWIDE$(FMODE$)                     021100
*GET(GET)                           012233
GRDGET(FMODE$)                      021146
HOLDIV(GRAC$)                       022140
HPFLAG(NOUT$)                       043110
IDENT$(ERU$)                   0000000000034
IDSLJ$(IDLS)                        012011
INCNTD(NINPT$)                      043054
*INPMTX(INPMTX)                     026406
*INPUT(INPUT)                       027331
INI$(ERU$)                     0000000000033
IOARR$(ERU$)                   0000000000021
IODYA$(ERU$)                   0000000000005
IOI$(ERU$)                     0000000000024
*IO$(ERU$)                     0000000000001
*LABELY(LABELY)                     017242

COS$(ERU$)                     0000000000000
CPOOL$(ERU$)                   0000000000035
*CSF$(ERU$)                    0000000000017
*CSMTRX(CSMTRX)                     024074
CSQRT$(ERU$)                   0000000001003
CTSA$(ERU$)                    0000000000124
CW$(ERU$)                      0000000000012
DADD$(ERU$)                    0000000000041
*DATA(DATA)                         025376
*DATE$(ERU$)                   0000000000022
DBACK$(ERU$)                   0000000000031
DCURT$(ERU$)                   0000000001006
DCOTAN$(ERU$)                  0000000001011
*DECIDE(DECIDE)                     025267
*DETCS(DETCS)                       031444
DEXP$(ERU$)                    0000000001003
DKEY$(ERU$)                    0000000000035
DLINK$(ERU$)                   0000000000040
DMPBUF(NFLNK$)                      022563
DREG$(ERU$)                    0000000000037
DSQRT$(ERU$)                   0000000001005
DUNLD$(ERU$)                   0000000000033
ENDEC$(NIER$)                       044327
ENDPLT(BUFRZZ)                      016444
*ERROR(ERROR)                       024157
*ESTMAT(ESTMAT)                     025620
EXLNK$(ERU$)                   0000000000173
EXPN$(ERU$)                    0000000001005
FACIL$(ERU$)                   0000000000114
*FCAN1$(FMODE$)                     021114
*FCUT$(FMODE$)                      021112
*FHS1$(NINPT$)                      043047
*FHS20$(NOUT$)                      043120
*FINDV(FINDV)                       020403
*FITEM$(ERU$)                  0000000000032
FLAG$(NFLNK$)                       022563
*FMTOP(NFMTS)                       043204
*FNCUT$(FMODE$)                     021110
FORMAIN$(MAIN)                      023414
FPRNT$(FPACK$)                      050162
*FRMTX(FRMTX)                       032257
*GATE(PLOT$$)                       050102
*GETEST(GETEST)                     031710
*GRID(GRID)                         022054
HOLDOV(GRAC$)                       022143
*IALL$(ERU$)                   0000000000101
*IDFRMZ(IDF)                        012255
II$(ERU$)                      0000000000027
*INITAL(INITAL)                     025356
*INPNYQ(INPNYQ)                     026557
INSTAT(NIER$)                       011550
*INTI(INTI)                         025412
IOAXI$(ERU$)                   0000000000020
IODYR$(ERU$)                   0000000000004
*IOW$(FRU$)                    0000000000003
*KCALC(KCALC)                       025757
LABEL$(ERU$)                   0000000000031
```

LCORES(ERUS)	000000000044	*LGRD(LGRD)	021012	LIGHT(NFLNKS)	000000000023
*LIMIT(LIMIT)	024223	LINKS(ERUS)	000000000171	*LOADS(ERUS)	000000000111
LWIDTH(FMODE$)	021145	*MB$(ERUS)	000000000051	MCORES(ERUS)	000000000043
MCTS(ERUS)	000000000041	*MF$(ERUS)	000000000050	*NLTZRO(MLTZRO)	025771
*MODZRO(MODZRO)	027065	*MRGSET(MRGSET)	022333	MSALLS(ERUS)	000000000060
MSCONS(ERUS)	000000000125	*MZTRAN(MZTRAN)	030337	*NAB0S(NFTVS)	001762
*NAB1S(NFTVS)	001763	*NAB2S(NFTVS)	001764	*NAB3S(NFTVS)	001765
*NAB4S(NFTVS)	001766	*NAB5S(NFTVS)	001767	*NAB6S(NFTVS)	001770
*NAB7S(NFTVS)	001771	NAMES(ERUS)	000000000146	*NAVCS(NFTVS)	001761
*NBCAS(NIOERS)	006455	*NBEGTS(NFCHKS)	007157	*NBFMGS(NFCHKS)	007162
*NBFRLS(NFCHKS)	007524	*NBFRSS(NFCHKS)	007510	*NBIPAS(NINPTS)	004001
*NBIS(NBDCVS)	001621	*NBLANK(NBLANK)	022412	*NBLNKS(NOUTS)	004336
NBMSGS(NFMTS)	043220	*NBTODS(NERRS)	011527	*NCAS(NIERS)	044427
*NCCCS(NOUTS)	004344	*NCDOFS(NINTRS)	051023	*NCEFS(NSTOPS)	050765
*NCHARS(NIERS)	044426	*NCONIDZS(NIERS)	044437	*NCNVTS(NCNVTS)	040605
*NCOM3S(NFMTS)	043151	*NCSPS(NIERS)	044440	NCSS(SINCOSS)	010452
*NCIULO(NIERS)	044434	*NCIULI(NIERS)	044435	NDANWS(NIERS)	011647
*NDBCVS(NCNVTS)	001776	*NDBFIS(NCNVTS)	002057	*NDBINS(NCNVTS)	040603
*NDDIS(NCNVTS)	001772	*NDBLTS(NFMTS)	043207	NDBSFS(NCNVTS)	040600
*NDIGS(NBDCVS)	040457	NDONES(NOTINS)	004315	*NDOUTS(NBDCVS)	001672
NDPFDLS(NIERS)	044341	NDTS(NIERS)	011635	*NEES(NSTOPS)	023046
*NEFCLS(NIOERS)	006473	*NENTRS(NSTOPS)	023050	*NERCRS(NIOERS)	006505
*NERCTS(NIOERS)	006503	*NERRAS(NERRS)	011174	*NERRBS(NERRS)	011177
*NERRCS(NERRS)	011202	NERRPS(NERRS)	011206	*NERR2S(NERRS)	011321
*NERR3S(NERRS)	011350	*NERR4S(NERRS)	011357	NERR5S(NERRS)	011364
*NERR6S(NERRS)	011430	*NERUS(NIOERS)	006432	NETDDS(NINPTS)	043053
NETOS(NINPTS)	043052	NETFS(NOUTS)	043124	NEWFRM(NFLNKS)	022534
NEWPAG(NFLNKS)	022540	*NEXITS(NOUTS)	005502	*NEXP1S(NEXP1S)	007744
*NEXP5S(NEXP5S)	010132	*NEXP6S(NEXP6S)	007547	NEXS(EXPS)	010042
NFAFS(NFCHKS)	007124	*NFARS(NFMTS)	006302	*NFBY1S(NIOERS)	006467
*NFCAS(NCNVTS)	002035	*NFCHKS(NFCHKS)	006560	*NFCIS(NCNVTS)	002031
*NFCMS(NCNVTS)	002064	*NFCSS(NCNVTS)	002023	*NFDBS(NCNVTS)	002100
*NFDPS(NCNVTS)	002073	*NFGCS(NFMTS)	006242	*NFGTS(NFMTS)	006253
*NFMIRS(NCNVTS)	002046	*NFINTS(NFMTS)	005513	*NFN96S(NFMTS)	043217
NFNIO1AS(NIERS)	011610	*NFNIO1DS(NIERS)	011617	*NFNI01S(NIERS)	011572
*NFNS1S(NBDCVS)	040503	*NFNS2S(NBDCVS)	040504	*NFNS3S(NBDCVS)	040505
*NFPCS(NIERS)	044373	NFPKTS(NFCHKS)	043500	*NFRAS(NFMTS)	043221
*NFRCS(NFMTS)	005550	*NFRGS(NIERS)	011657	*NFRHS(NIERS)	044345
*NFRJS(NIERS)	044374	*NFRONFS(NIOERS)	006463	*NFRZS(NIERS)	044324
*NFRZSS(NIERS)	044325	*NFSGS(NCNVTS)	002103	*NFTCBS(NFTCHS)	001417
*NFTCHS(NFTCHS)	001336	*NFTGLS(NIERS)	044330	*NGC9S(NFMTS)	043210
*NHPFAS(NIERS)	011715	NHPFBS(NIERS)	011711	*NHVCS(NFTVS)	001757
*NICHOL(NICHOL)	024567	*NIICS(NINPTS)	004007	*NINDS(NBDCVS)	040454
*NINIIS(NININS)	002614	*NINTRS(NINTRS)	023406	NINTS(NCNVTS)	040601
*NIOERS(NIOERS)	006370	*NIOERSA(NIOERS)	006407	NIO18S(NIERS)	011625
*NIO1VS(NIERS)	011564	*NIO1S(NIERS)	011562	*NIO2VS(NIERS)	011566
*NIO2S(NIERS)	011565	*NIO3VAS(NIERS)	011612	*NIO3VS(NIERS)	011571
*NIO3S(NIERS)	011567	NIO1S(NIERS)	011562	*NIO2S(NIERS)	011565
*NKLNS(NIERS)	011554	*NKL2S(NIERS)	011560	NLI0S(NIERS)	011557
*NLLCS(NIERS)	044333	*NLLMS(NIERS)	044336	*NLRTS(NIERS)	044335
*NLTBS(NIERS)	044334	NNBKSS(ERUS)	000000001012	NNCLOS(ERUS)	000000001027
NNDECS(ERUS)	000000001015	NNDEFS(ERUS)	000000001022	NNENCS(ERUS)	000000001016
NNENDS(ERUS)	000000001021	*NNG90S(NIERS)	044344	NNPCHS(ERUS)	000000001005
NNPRTS(ERUS)	000000001004	NNRCDS(ERUS)	000000001003	NNRDAS(ERUS)	000000001017
NNREWS(ERUS)	000000001013	NNRFTS(ERUS)	000000001023	NNRFUS(ERUS)	000000001001
NNRNLS(ERUS)	000000001010	NNRSXS(NIERS)	011704	NNRTS(ERUS)	000000001025
NNRUS(ERUS)	000000001006	NNWDAS(ERUS)	000000001020	NNWDLS(NIERS)	044433

```
NNWEF$(ERU$)      000000001014    NNWFT$(ERU$)      000000001024    NNWFU$(ERU$)      000000001002
NNWNL$(ERU$)      000000001011    NNWT$(ERU$)       000000001026    NNWU$(ERU$)       000000001007
NOFRV(NFLNK$)        022565       NOLC$(NIERS)         044340    *NOLM$(NIERS)         044337
*NOMNTX(NOMNTX)      024263       *NOP(PLOT$$)         021072    *NOTIS(NOTIN$)        004041
*NPCT$(NFMT$)        006261       *NPR$(NOUT$)         005465    *NPUS(NOUT$)          005506
*NPW2$(NFMT$)        043205       *NP91$(NFMT$)        043211    NRABC$(ERU$)      000000001001
*NRBFA$(NFCHK$)      007540       *NRBF$(NFCHK$)       007534    NRDTS(NIBUF$)         010001
*NRDU$(NIBUF$)       010001       *NRD$(NIOPT$)        003773    *NREC$(NIOER$)        043226
*NREZ$(NRWND$)       001050       *NRM92$(NIERS)       044422    *NRSF$(NERR$)         044140
*NRSX$(NIERS)        011667       *NRTR$(NTERS)        011553    NRT$(ERU$)        000000000062
NR-B$(ERU$)       000000001002    *NRY1$(NFMT$)        043206    *NR92$(NFMT$)         043212
*NR93$(NFMT$)        043222       *NSA0$(NERRS)        044143    *NSF$(NCNVT$)         040577
*NSL$(NBOCV$)        001631       NSNS(SINCOS$)        010453    *NSTAT$(NIERS)        044331
*NSTOP$(NSTOP$)      023017       *NSTSV$(NIERS)       044332    *NSWTC$(NSWTC$)       001000
*NS11$(NSTOP$)       050763       *NTAB$(NTAB$)        043611    *NTBSZ$(NTAB$)    000000000040
*NTEND$(NFMT$)       006301       NTPER$(NSTOP$)       023046    *NTST0$(NFMT$)        006300
*NTI0$(NFMT$)        006541       *NVEC$(NFTV$)        001747    *NWALK$(NERR$)        011420
NWDTS(NOBUF$)        011732       *NWD0$(NOBUF$)       011732    *NWEF$(NWEF$)         001132
NYQB$(ERU$)       000000001003    *NXVC$(NFTV$)        001760    *NYQIST(NYQIST)       025446
NS(ERU$)          000000000001    *OPT$(ERU$)       000000000063    OUTCNT(NOUT$)         043125
*OUTPUT(OUTPUT)      026152       *OUT0ZZ(BUFRZZ)      016362    *OUT6ZZ(BUFRZZ)       016355
*PAC(PAC)            012155       *PACKTS(NIERS)       044323    *PACKZZ(PACKZZ)       012461
PCHCA$(ERU$)      000000000165    PCHCN$(ERU$)      000000000164    PCTBD$(ERU$)      000000000001
*PCT$(ERU$)       000000000064    *PEVAL(PEVAL)        026303    *PEXIT(PLOT$$)        021077
PFD$(ERU$)        000000000106    *PFE(PFE)            030717    *PFEZRO(PFEZRO)       031221
PFI$(ERU$)        000000000104    PFS$(ERU$)        000000000105    PFUNL$(ERU$)      000000000107
PFUL$(ERU$)       000000000110    *PLOTR(PLOTR)        016541    *PLOTS(PLOTS$)        021070
PLOT$$(PLOTS$$)      021070       PLOTI(PLOTI)         021147    *PLOTIV(PLOTI)        021260
*PLS$(FMODE$)        050232       *PL$(NBFOO$)      000000000015    *PNCHA$(ERU$)     000000000145
*POINT(POINT)        027202       *POLES(POLES)        032324    *POLVAL(POLVAL)       027314
*PPP$(NOUT$)         043114       *PREVAR(PREVAR)      027376    *PRINT(PRINT)         021424
*PRINTT(PRINTT)      032554       *PRINTV(PRINTV)      012042    *PRINT$(ERU$)     000000000016
*PRNTA$(ERU$)     000000000144    PRTCA$(ERU$)      000000000155    PRTCN$(ERU$)      000000000137
PSA$(ERU$)        000000000032    PSF$(ERU$)        000000000033    PSRSD$(ERU$)      000000000163
PSP$(ERU$)        000000000157    *PUNCH$(ERU$)     000000000130    *PUT(PUT)             012215
*PUTOUT(PUTOUT)      033127       PS1NT$(ERU$)      000000000016    *QUIKML(QUIKML)       022773
*RAMTX(RAMNTX)       027417       RB$(ERU$)         000000000021    *RDBLK$(NRBLK$)       001025
RDL$(ERU$)        000000000025    *READA$(ERU$)     000000000042    *READ$(ERU$)      000000000015
REL$(ERU$)        000000000023    *RESET(RESET)        024400    RESETV(NFLNK$)        022437
*REST$(NERR$)        011506       REWI$(ERU$)       000000000041    *REW$(ERU$)       000000000040
RLIB$(ERU$)    667105050505       RLINK$(ERU$)      000000000172    RLIST$(ERU$)      000000000175
*RLOCUS(RLOCUS)      033270       *RLPLOT(RLPLOT)      031725    *RLPRNT(RLPRNT)       032422
*ROOTER(ROOTER)      026672       ROUTE$(ERU$)      000000000133    RPCTA$(ERU$)      000000777000
RR$(ERU$)         000000000022    RSET(NFLNK$)         022437    RSWAP$(ERU$)      000000000135
RT$(ERU$)         000000000061    *RUTER(RUTER)        032603    *RVSX(RVSX)           017027
*RVSY(RVSX)          017036       *R$(ERU$)         000000000020    *SAVE(SAVE)           027032
*SAVEST(SAVEST)      033503       *SAVRUT(SAVRUT)      032703    *SCALE(SCALE)         027221
*SCALEX(SCALEX)      017720       *SCALEY(SCALEX)      017727    *SCCTAB(SCCTAB)       047507
*SCCTZZ(SCTZ)        045103       SCR(NIER$)           044326    SCRB$(ERU$)       000000000044
SCR$(ERU$)        000000000043    SD$(ERU$)         000000000034    *SEPDPC(SEPDPC)       031267
SETCIV(GRAC$)        022175       SETCOV(GRAC$)        022202    SETCS(ERU$)       000000000065
SETFRV(NFLNK$)       022565       *SETMIV(GRAC$)       022147    *SETMOV(GRAC$)        022160
SHRINK(NFLNK$)       022427       *SIN(SINCOS$)        010460    SINH$(ERU$)       000000001022
SINS(ERU$)        000000001012    SINS(ERU$)        000000001011    *SLT$             000000040000
SMALLV(NFLNK$)       022427       SMS(ERU$)         000000000042    SNAP$(ERU$)       000000000126
*SOLVE(SOLVE)        030025       *SQRT(SQRTS)         011011    SQRT$(ERU$)       000000001006
SRD$(ERU$)        000000000036    *SRRL(SRRL)          033121    *SRRLPP(SRRLPP)       033412
```

*START(START)	030341	*STNNYQ(STNNYQ)	027373	*STNRL(STNRL)	022543
STOPTV(GRAC$)	022104	STOREA(NFLNK$)	022546	STOREB(NFLNK$)	016304
*STOREX(NFLNK$)	022556	*STREG$(NERRS)	011470	STRTZZ(BUFRZZ)	012775
*SUMMRY(SUMMRY)	030464	SWAIT$(ERU$)	000000000103	*SWRITE(IDENT)	051061
SWTCH$(ERU$)	000000000127	SX$(ERU$)	000000000013	SXII(NINTR$)	024600
SYSBAL(ERU$)	000000000176	*SYSERQ(SYSERQ)	030515	TAN(TANCOTAN$)	000000000054
TAN$(ERU$)	000000001024	TANS(ERU$)	000000001015	TDATE$(ERU$)	000000000014
*TEMP$(NIER$)	044423	*TEST(TEST)	030744	TFORK$(ERU$)	022566
*TIME$(ERU$)	000000000023	TINTL$(ERU$)	000000000136	TPNUMV(NFLNK$)	000000000030
TREAD$(ERU$)	000000000102	*TRNSER(TRNSER)	031037	TSA$(ERU$)	000000000121
TSF$(ERU$)	000000000031	TSWCL$(ERU$)	000000000113	TSWRG$(ERU$)	044342
TSWAP$(ERU$)	000000000135	*TWAIT$(ERU$)	000000000060	*UNIT$(NIER$)	000000000026
UNLCK$(ERU$)	000000000067	UNLNK$(ERU$)	000000000174	UNL$(ERU$)	013753
*UNPAC(UNPAC)	012200	*UPDDAS(UPDDA$)	002560	*VECTR(VECTR)	030031
*VLAG(VLAG)	017376	*VRYGEN(VRYGEN)	030002	*VRYRAW(VRYRAW)	000000001006
*WAIT$(ERU$)	000000000006	WANY$(ERU$)	000000000007	WBKST$(ERU$)	031201
*WEF$(ERU$)	000000000011	*WRBLK$(WBLK$)	002477	*WRITE(WRITE)	000000000162
*WS(ERU$)	000000000010	*XCNG(XCNG)	030356	XCTS$(ERU$)	000000001032
*XFOR$(NFMT$)	006337	XMODV(GRAC$)	022103	XPCC$(ERU$)	000000001025
XPCI$(ERU$)	000000001015	XPCR$(ERU$)	000000001031	XPDO$(ERU$)	000000001030
XPDI$(ERU$)	000000001026	XPDR$(ERU$)	000000001024	XPIC$(ERU$)	000000001025
XPID$(ERU$)	000000001022	XPI$(NEXPI$)	007744	XPII$(ERU$)	000000001023
XPIR$(ERU$)	000000001027	XPRC$(ERU$)	000000001027	XPRO$(ERU$)	007547
*XPRI(NEXPS$)	010132	XPRI$(ERU$)	000000001026	XPRR(NEXP6$)	024465
XPRR$(ERU$)	000000001030	YMODV(GRAC$)	022103	*ZEEPLS(ZEEPLS)	000000000000
*ZEROS(ZEROS)	033336	*ZTRAN(ZTRAN)	032417	*A$SEG-NAME]	000000000005
*B1$SEG-NAME]	000000000001	*B2$SEG-NAME]	000000000002	*B3$SEG-NAME]	000000000004
*B4$SEG-NAME]	000000000006	*C1$SEG-NAME]	000000000003	*C2$SEG-NAME]	

SYS$*RLIB$, LEVEL 69
END OF COLLECTION - TIME 13.417 SECONDS

.FIN

APPENDIX A - GENERATION OF A SAMPLED DATA OPEN LOOP TRANSFER FUNCTION

NOMENCLATURE

$A(S)$	Characteristic matrix polynomial obtained from the Laplace transformation of the system dynamic and control equations
$GH(S)$	Open loop transfer function of the continuous system in the S domain.
$GH(Z)$	Open loop transfer function of the sampled data system in the Z domain.
$GH(Z=\frac{R+1}{R-1})$	Open loop transfer function of the sampled data system after the bi-linear transformation $Z=\frac{R+1}{R-1}$ from the Z domain to the R domain.
$G_{ho}GH(Z)$	Open loop transfer function of the sampled data system in the Z domain with zero order hold.
$G_{ho}GH(Z=\frac{R+1}{R-1})$	Open loop transfer function of the sampled data system with zero order hold after the bi-linear transformation $Z=\frac{R+1}{R-1}$ from the Z domain to the R domain.
$P(S)$	Numerator polynomial of $GH(S)$.
K_p	Leading coefficient of $P(S)$.
$Q(S)$	Denominator polynomial of $GH(S)$.
K_Q	Leading coefficient of $Q(S)$.
K	Leading coefficient of $GH(S)$ where $K=K_p/K_Q$
T	Sampling period.
T_D	Transport Lag.

NOMENCLATURE

M	$M = 1 - \frac{T_D}{T}$
RTR	Real part of a root.
RTI	Imaginary part of a root.
RSR	Real part of a residue.
RSI	Imaginary part of a residue.

PROBLEM INPUT

$$A(S) = \begin{vmatrix} S & 10 & 16 \\ 1 & S^2+2 & S \\ 0 & 1 & 1 \end{vmatrix}$$

Open the loop at matrix location row 3, column 2

$T = 0.5$
$T_D = 0.1$
$\Rightarrow M = 0.8$

PROBLEM OUTPUT

Characteristic roots of the matrix polynomial (poles of the closed loop transfer function)

Poles of the open loop transfer function (roots of Q(S))

The Q(S) polynomial and leading coefficient K_Q

Zeros of the open loop transfer function (roots of P(S))

The P(S) polynomial and leading coefficient K_P

Leading coefficient of GH(S) denoted by $K=\frac{K_P}{K_Q}$

Partial fraction expansion of GH(S) (residues and poles)

Transformation of GH(S) to GH(Z) in partial fraction expansion form

Transformation of GH(Z) to $GH(Z=\frac{R+1}{R-1})$ partial fraction expansion form (bi-linear transformation from the Z domain to the R domain)

Open loop poles of GH(Z) and magnitudes

$G_{ho}GH(Z)$ transfer function with zero order hold in partial fraction expansion form

Transformation of the $G_{ho}GH(Z)$ transfer function with zero order hold to the $G_{ho}GH(Z=\frac{R+1}{R-1})$ transfer function in partial fraction expansion form.

SOLUTION

Characteristic matrix

$$A(S) = \begin{bmatrix} S & 0 & 16 \\ 1 & S^2+2 & S \\ 0 & 1 & 1 \end{bmatrix}$$

Characteristic polynomial

$$C(S) = |A(S)|$$

$$= S(S^2+2-S)-1(-16)$$

$$= S^3+2S-S^2+16$$

$$= S^3-S^2+2S+16$$

$$= (S^2-3S+8)(S+2)$$

$$= (S-1.5-\frac{\sqrt{23}}{2}\,i)(S-1.5+\frac{\sqrt{23}}{2}\,i)(S+2)$$

$$= (S-1.5-2.397916i)(S-1.5+2.397916i)(S+2)$$

Characteristic roots

$$1.5\pm 2.397916i\ ,\ -2.0$$

The loop is opened at matrix location row 3, column 2.

$$\begin{vmatrix} S & 0 & 16 \\ 1 & 0 & S \\ 0 & 1 & 1 \end{vmatrix} = S(-S)-1(-16) = -S^2+16 = -1.0(S^2-16) = -1.0\ (S+4)(S-4)$$

Hence;

$$P(S) = S^2-16 = (S+4)(S-4)$$

$$K_p = -1.0$$

Open loop zeros ± 4.0

$$\begin{vmatrix} S & 0 & 16 \\ 1 & S^2+2 & S \\ 0 & 0 & 1 \end{vmatrix} = S(S^2+2) = S^3+2S = S\ (S+\sqrt{2}\ i)(S-\sqrt{2}\ i)$$

Also;

$$Q(S) = S^3+2S = S(S+\sqrt{2}\ i)(S-\sqrt{2}\ i)$$

$$K_Q = 1.0$$

open loop poles $0.0, \pm\sqrt{2}\ i$

where $\sqrt{2} = 1.41421$

$$K = \frac{K_P}{K_Q} = \frac{-1.0}{1.0} = -1.0$$

$$\frac{P(S)}{Q(S)} = \frac{S^2-16}{S^3+2S} = \frac{S^2-16}{S(S+\sqrt{2}\,i)(S-\sqrt{2}\,i)} = \frac{A}{S} + \frac{B}{S+\sqrt{2}\,i} + \frac{\bar{B}}{S-\sqrt{2}\,i}$$

$$\rightarrow \quad S^2-16 = (S+\sqrt{2}\,i)(S-\sqrt{2}\,i)A + S(S-\sqrt{2}\,i)B + S(S+\sqrt{2}\,i)\bar{B}$$

Hence

$S = 0 \rightarrow -16 = 2A \rightarrow A = -8$

$S = -\sqrt{2}\,i \rightarrow -2-16 = -\sqrt{2}\,i(-2\sqrt{2}\,i)B \rightarrow B = 4.5$

$\bar{B} = 4.5$

Therefore

$$\frac{P(S)}{Q(S)} = \frac{-8}{S} + \frac{4.5}{S+\sqrt{2}\,i} + \frac{4.5}{S-\sqrt{2}\,i}$$

$$= \frac{-8}{S} + \frac{9S}{S^2+2}$$

Hence the open loop transfer function GH(S) can be expressed in partial fraction expansion form

$$GH(S) = K\frac{P(S)}{Q(S)} = -1.0\left[\frac{-8.0}{S} + \frac{9.0S}{S^2+2.0}\right] = \frac{8.0}{S} - \frac{9.0S}{S^2+2.0}$$

where

Open Loop Pole	Residue
0.0	-8.0
$-\sqrt{2}\ i$	4.5
$+\sqrt{2}\ i$	4.5

GH(S) is transformed into GH(Z) by transforming the individual terms of the partial fraction expansion.

$$\frac{8.0}{S} \longrightarrow \frac{8.0}{Z-1.0}$$

$$\frac{-9.0S}{S^2+2.0} \longrightarrow \frac{aZ+b}{cZ^2+dZ+e}$$

where

$$T1 = e^{RTR \times T} = e^{0.0 \times 0.5} = e^{0.0} = 1.0$$

$$T2 = e^{RTR \times M \times T} = e^{0.0 \times 0.8 \times 0.5} = e^{0.0} = 1.0$$

$$T3 = |RTI| \times T = \sqrt{2} \times 0.5 = 1.41421 \times 0.5 = 0.707105$$

$$T4 = 2.0 \times RSR \times T2 = 2.0 \times 4.5 \times 1.0 = 9.0$$

$$T5 = -2.0 \times RSI \times T2 = -2.0 \times 0.0 \times 1.0 = 0.0$$

and

$$a = K \times [T4 \times \cos (M \times T3) + T5 \times \sin (M \times T3)]$$

$$= -1.0[9.0 \times \cos (0.8 \times 0.707105) + 0.0 \times \sin (0.8 \times 0.707105)]$$

$$= -9.0 \cos (0.565684)$$

$$= -9.0 \cos (32.425°)$$

$$= -9.0 \cos (32° 25')$$

$$= -9.0 \times 0.84417$$

$$= -7.59753$$

b = K x [T5 x T1 x sin [(1.0-M) x T3]-T4 x T1 x cos [(1.0-M) x T3]]

= -1.0[0.0 x 1.0 x sin[(1.0-0.8)x0.707105]-9.0 x 1.0 x cos[(1.0-0.8)x0.707105]]

= 9.0 cos (0.2x0.707105)

= 9.0 cos (0.141421)

= 9.0 cos (8.08°)

= 9.0 cos (8° 6')

= 9.0 x 0.99002

= 8.91018

c = 1.0

$$d = -2.0 \times T1 \times \cos(T3)$$

$$= -2.0 \times 1.0 \times \cos(0.707105)$$

$$= -2.0 \cos(40.508°)$$

$$= -2.0 \cos(40° 30.48')$$

$$= -2.0 \times 0.76025$$

$$= -1.52050$$

$$e = e^{2.0 \times RTR \times T}$$

$$= e^{2.0 \times 0.0 \times 0.5}$$

$$= e^{0.0}$$

$$= 1.0$$

hence $\dfrac{-9.0\ S}{S^2+2.0} \longrightarrow \boxed{\dfrac{-7.59753Z+8.910}{Z^2 - 1.52050Z+1.0}}$

The open loop transfer function in the Z domain GH(Z) can be expressed in partial fraction expansion form where

$$GH(Z) = \frac{8.0}{Z-1.0} + \frac{-7.59753Z+8.910}{Z^2-1.52050Z+1.0}$$

The open loop poles of GH(Z) are the roots of (Z-1.0) and (Z^2-1.52050Z+1.0).

Open Loop Poles of GH(Z)	Magnitude
1.0	1.0
.760245 + .649637 i	1.0
.760245 - .649637 i	1.0

GH(Z) is transformed into the R domain by the bi-linear transformation $Z=\frac{R+1}{R-1}$ which is applied to each term of the partial fraction expansion of GH(Z)

$$\frac{8.0}{Z-1.0} \longrightarrow \frac{8.0}{\frac{R+1}{R-1} - 1.0} = \frac{8.0(R-1.0)}{(R+1.0)-1.0(R-1.0)} = \boxed{\frac{8.0R-8.0}{2.0}}$$

$$\frac{-7.59753Z+8.910}{Z^2-1.52050Z+1.0} = \frac{aZ+b}{cZ^2+dZ+e} \longrightarrow \frac{a(\frac{R+1}{R-1})+b}{c(\frac{R+1}{R-1})^2+d(\frac{R+1}{R-1})+e}$$

where $$\frac{a\left(\frac{R+1}{R-1}\right) + b}{c\left(\frac{R+1}{R-1}\right)^2 + d\left(\frac{R+1}{R-1}\right) + e} = \frac{(a+b)R^2-2.0bR+(b-a)}{(c+d+e)R^2+(2.0c-2.0e)R+(c-d+e)}$$

$$= \frac{(-7.59753+8.910)R^2-2.0x8.910R+(8.910+7.59753)}{(1.0-1.5205+1.0)R^2+(2.0x1.0-2.0x1.0)R+(1.0+1.5205+1.0)}$$

$$= \frac{1.312R^2-17.820R+16.508}{0.4795R^2+0.0R+3.5205}$$

hence

$$\frac{-7.59753Z+8.910}{Z^2-1.5205Z+1.0} \longrightarrow \boxed{\frac{1.312R^2-17.820R+16.508}{0.4795R^2+0.0R+3.5205}}$$

The open loop transfer function in the R domain $GH(Z=\frac{R+1}{R-1})$ can be expressed in partial fraction expansion form where

$$GH(Z=\frac{R+1}{R-1}) = \frac{8.0R-8.0}{2.0} + \frac{1.312R^2-17.820R+16.508}{0.4795R^2+0.0R+3.5205}$$

ZERO ORDER HOLD

A zero order hold is represented by $\frac{1-e^{-ST}}{S}$ which is multiplied times the open loop transfer function.

$$G_{ho}GH(S) = \left(\frac{1-e^{-ST}}{S}\right)GH(S)$$

$$= \left(\frac{1-e^{-ST}}{S}\right)\left[-1.0\left(\frac{-8.0}{S} + \frac{9.0S}{S^2+2.0}\right)\right]$$

$$= (1-e^{-ST})\left[-1.0\left(\frac{-8.0}{S^2} + \frac{9.0}{S^2+2.0}\right)\right]$$

$$= (1-e^{-ST})\left[-1.0\left(\frac{S^2-16}{S^4+2S^2}\right)\right]$$

The open loop poles of $G_{Ho}GH(S)$ are 0.0, 0.0, $\sqrt{2}i$, and $-\sqrt{2}i$. The zero order hold introduced an additional open loop pole at the origin. Let $P(S)=S^2-16$

$$Q(S)=S^4+2S^2$$

$$K = -1.0$$

then

$$\frac{P(S)}{Q(S)} = \frac{S^2-16}{S^4+2S^2}$$

and

$$G_{ho}GH(S)=(1-e^{-ST})K\frac{P(S)}{Q(S)}$$

C-5

Expanding $\frac{P(S)}{Q(S)}$ into partial fraction form yields the following:

$$\frac{P(S)}{Q(S)} = \frac{A}{S^2} + \frac{B}{S} + \frac{C}{S+\sqrt{2}i} + \frac{\bar{C}}{S-\sqrt{2}i}$$

Let $R(S) = S^2 + 2$ then

$$A = \frac{P(0)}{R(0)} = \frac{-16}{2} = -8$$

$$B = \frac{P'(0)-AR'(0)}{R(0)} = \frac{0.0-8.0(0.0)}{2.0} = 0.0$$

$$C = \frac{-18}{4\sqrt{2}i} = \frac{-18i}{-4\sqrt{2}} = \frac{4.5i}{\sqrt{2}} = 3.18198i$$

$$\bar{C} = -3.18198i$$

Hence,

$$\frac{P(S)}{Q(S)} = \frac{-8}{S^2} + \frac{0}{S} + \frac{3.18198i}{S+\sqrt{2}i} - \frac{3.18198i}{S-\sqrt{2}i}$$

$$= \frac{-8}{S^2} + \frac{\frac{4.5}{\sqrt{2}i}}{S+\sqrt{2}i} - \frac{\frac{4.5}{\sqrt{2}i}}{S-\sqrt{2}i}$$

$$= \frac{-8}{S^2} + \frac{9.0}{S^2+2.0}$$

The open loop transfer function $G_{ho}GH(S)$ with zero order hold can be expressed in partial fraction expansion form

$$G_{ho}GH(S) = (1-e^{-ST})K\frac{P(S)}{Q(S)}$$

$$= (1-e^{-ST})[-1(-\frac{8.0}{S^2} + \frac{9.0}{S^2+2.0})]$$

$$= (1-e^{-ST})(\frac{8.0}{S^2} - \frac{9.0}{S^2+2.0})$$

where

	Open Loop Pole	Residue
$\frac{A}{S^2}$ ⟶	0.0	-8.0
$\frac{B}{S}$ ⟶	0.0	0.0
	$\sqrt{2}i$	3.18198i
	$-\sqrt{2}i$	-3.18198i

By definition $Z=e^{+ST}$, hence the Z transformation of $G_{ho}GH(S)$ becomes

$$\mathcal{Z}[G_{ho}GH(S)] = \mathcal{Z}\left[\left(\frac{1-e^{-ST}}{S}\right)GH(S)\right]$$

$$= (1-Z^{-1})\,\mathcal{Z}\left[\frac{GH(S)}{S}\right]$$

$$= \frac{Z-1}{Z}\,\mathcal{Z}\left[\frac{GH(S)}{S}\right]$$

$G_{ho}GH(S)$ is transformed into $G_{ho}GH(Z)$ by transforming the individual terms of the partial fraction expansion.

$$\frac{8.0}{S^2}(1-e^{-ST}) \longrightarrow \frac{aZ+b}{Z^2-2Z+1}\left(\frac{Z-1}{Z}\right)$$

where $a = RSRxMxTxK$ $\quad b = KxRSRxTx(1-M)$

$a = -8.0x0.8x0.5x-1.0$ $\quad b = -1.0x(-8.0)x0.5x(1.0-0.8)$

$a = 3.2$ $\quad b = 0.8$

hence $$\frac{8.0}{S^2}(1-e^{-ST}) \longrightarrow \frac{3.2Z+0.8}{Z^2-2.0Z+1.0}\left(\frac{Z-1}{Z}\right) = \frac{3.2Z+0.8}{(Z-1)^2}\left(\frac{Z-1}{Z}\right)$$

$$= \boxed{\frac{3.2Z+0.8}{Z^2-Z+0.0}}$$

$$\frac{0.0}{S}(1-e^{-ST}) \longrightarrow \boxed{0.0}$$

$$\frac{-9.0}{S^2+2}(1-e^{-ST}) \longrightarrow \frac{a\ Z\ +\ b}{cZ^2+dZ+e}\left(\frac{Z-1}{Z}\right)$$

where

$T1 = e^{0.0 \times 0.5} = e^{0.0} = 1.0$

$T2 = e^{0.0 \times 0.8 \times 0.5} = e^{0.0} = 1.0$

$T3 = |\sqrt{2}| \times 0.5 = \sqrt{2} \times 0.5 = 1.41421 \times 0.5 = 0.707105$

$T4 = 2.0 \times 0.0 \times 1.0 = 0.0$

$T5 = -2.0 \times 3.18198 \times 1.0 = -6.36396$

since $RTI = -\sqrt{2} < 0.0$ then $T5 = 6.36396$

and

$$a = -1.0[0.0\times\cos(0.8\times0.707105)+6.36396\times\sin(0.8\times0.707105)]$$
$$= -6.36396\times\sin(0.565684)$$
$$= -6.36396\times\sin(32.425°)$$
$$= -6.36396\times\sin(32°25')$$
$$= -6.36396\times0.53607$$
$$= -3.411528$$

$$b = -1.0\times[+6.36396\times1.0\times\sin[(1.0-0.8)\times.707105]-0.0\times1.0\times\cos[(1.0-.8)\times.707105]]$$
$$= -6.36396\times\sin(0.2\times.707105)$$
$$= -6.36396 \times \sin(.1414210)$$
$$= -6.36396 \times \sin(8.10°)$$
$$= -6.36396 \times \sin (8°6')$$
$$= -6.36396 \times .14090$$
$$= -8.96745$$

$$c = 1.0$$

$$b = -2.0\times1.0\times\cos(0.707105)$$

$$= -2.0\times\cos(40.508°)$$

$$= -2.0\times\cos(40°30.48')$$

$$= -2.0\times0.76025$$

$$= -1.52050$$

$$e = e^{2.0\times0.0\times0.5}$$

$$= e^{0.0}$$

$$= 1.0$$

hence

$$\frac{-9.0}{S^2+2}(1-e^{-ST}) \longrightarrow \frac{-3.411528Z-.896745}{Z^2-1.52050Z+1.0}\left(\frac{Z-1.0}{Z}\right)$$

$$= \boxed{\frac{-3.411528Z^2+2.51405Z+.896745}{Z^3-1.5205Z^2+1.0Z+0.0}}$$

The open loop transfer function $G_{ho}GH(Z)$ with zero order hold can be expressed in partial fraction expansion form where

$$G_{ho}GH(Z) = \frac{3.2Z+0.8}{Z^2-Z+0.0} + \frac{-3.411528Z^2+2.51405Z+.896745}{Z^3-1.5205Z^2+1.0Z+0.0}$$

and the open loop poles in the Z domain are

Open Loop Pole of $G_{ho}GH(Z)$	Magnitude	
1.0	1.0	
.760245+.649637i	1.0	
.760245-.649637i	1.0	
0.0	0.0	generated by the zero order hold option
0.0	0.0	

$G_{ho}GH(Z)$ with zero order hold is transformed into the R domain by the bi-linear transformation $Z = \frac{R+1}{R-1}$ which is applied to each term of the partial fraction expansion of $G_{ho}GH(Z)$

$$\frac{3.2Z+0.8}{Z^2-Z} \longrightarrow \frac{3.2\left(\frac{R+1}{R-1}\right)+0.8}{\left(\frac{R+1}{R-1}\right)^2-\left(\frac{R+1}{R-1}\right)} = \frac{3.2(R^2-1.0)+0.8(R^2-2.0R+1.0)}{R^2+2.0R+1.0-(R^2-1.0)}$$

$$= \boxed{\frac{4.0R^2-1.6R-2.4}{2.0R+2.0}}$$

$$\frac{-3.411528Z^2+2.51405Z+0.896745}{Z^3-1.5205Z^2+1.0Z+0.0} = \frac{aZ^2+bZ+c}{dZ^3+eZ^2+fZ+0.0}$$

where $\frac{aZ^2+bZ+c}{dZ^3+eZ^2+fZ+0.0} \longrightarrow \frac{a(\frac{R+1}{R-1})^2+b(\frac{R+1}{R-1})+c}{b(\frac{R+1}{R-1})^3+e(\frac{R+1}{R-1})^2+f(\frac{R+1}{R-1})+0.0} =$

$$\frac{a(\quad-3c)R+(3c-a-b)R+(b-a-c)}{(d\quad+f)R^3+(e-f+3)R^2+(3-e-f)R+(f-e+d)}$$

$$= \frac{(-3.411528-2.51405-3.0x.896745)R^2+(3.0x.8\,\,745+3.411528-2.51405)R+(2.51405+3.41152S-.89674S)}{(1.0-1.5205+1.0)R^3+(-1.5205-1.0+3.0)R^2+(3.\,\,)+1.5205-1.0)R+(1.0+1.5205+1.0)}$$

$$= \frac{(-5.925578-2.690235)R+(2.690235+0.897478)R+(5.925578-.896745)}{.4795R^3+.4795R^2+3.5205R+3.5205}$$

$$= \boxed{\frac{-8.615813R^2+3.587713R+5.028833}{.4795R^3+.4795R^2+3.5205R+3.5205}}$$

The open loop transfer function with zero order hold in the R domain $G_{ho}GH(Z=\frac{R+1}{R-1})$ can be expressed in partial fraction expansion form.

$$G_{ho}GH(Z=\frac{R+1}{R-1}) = \frac{4.0R^2-1.6R-2.4}{2.0R+2.0} + \frac{-8.615813R^2+3.587713R+5.028833}{.4795R^3+.4795R^2+3.5205R+3.5205}$$

APPENDIX B

ERROR PROCEDURE AND COMMENTS

Any input, execution, or logic error incurred during program processing is assigned a unique error code which is printed by the program.

ERROR CODE = XXX

Following the error message the program sloughs and prints data cards until a data card with 'KEY' punched in colums 1 4 is encountered.

SKIPPING DATA CARDS UNTIL KEY WORD IS FOUND

(1-st data card sloughed)
(2-nd data card sloughed)
.
.
.
(n-th data card sloughed)

Program processing resumes with the next data case. The error codes assigned by the program are described by the following chart.

ERROR CODE	SUBROUTINE	EXPLANATION
2	ADDZOH	Adding the zero order hold circuit caused the continuous system to exceed the maximum allowable open poles at the origin.
4	BHA341	The continuous system open loop transfer function had more than the maximum allowable poles at the origin.
5	COMPUT	The program is attempting to compute the continuous system open loop transfer function but the characteristic matrix has not been defined.
9	COMPUT	No continuous system open loop poles have been computed.
10	COMPUT	No continuous system nominal closed loop poles have been computed.
11	DATA	Number of sample periods exceeds the maximum.
12	DATA	Neither Z transformation with or without a zero order hold circuit has been requested.
13	DATA	Program could not interpret a data input or processing request.

ERROR CODE	SUBROUTINE	EXPLANATION
14	DATA	User requested the program not to execute the nominal matrix and at the same time did not input parameter variations.
20	GENMTX	The row or column designation of an element in the continuous system characteristic matrix is out of range.
21	GENMTX	The number of non-zero polynomial elements in the continuous system characteristic matrix exceeds the maximum.
22	GENMTX	The number of coefficients in the continuous system characteristic matrix exceeds the maximum.
23	GENMTX	The continuous system characteristic matrix has a zero row or column.
24	INITAL	A frequency interval exceeds the maximum allowable frequency of $\frac{1}{T}$ H_Z where T equals the sample period.
25	INPEST	The program could not interpret the option to retain or input new eigenvalue estimates to the continuous system.
26	INPEST	The user requested to retain the previous case continuous system eigenvalue estimates but none exist.
27	INPEST	The number of continuous system eigenvalue estimates is out of range.
32	INPMTX	Program could not interpret request to retain the previous case nominal matrix or input a new matrix in either general or raw data format.
33	INPMTX	User requested to use the previous case characteristic matrix, but none exists.
34	INPNYQ	Program could not interpret request to retain previous case, implement standard, or input new Nyquist data.

ERROR CODE	SUBROUTINE	EXPLANATION
35	INPNYQ	User requested to use previous case Nyquist data, but none exists.
36	INPNYQ	The number of Nyquist frequency intervals is out of range.
37	INPRL	Program could not interpret request to retain previous cases, implement standard, or input new root locus data.
38	INPRL	Sampled data root locus requested but neither gain nor phase root locus requested.
39	INPRL	User requested to use previous case root locus data, but none exists.
40	INPRL	Sampled data gain root locus requested as in the previous case but no gain variations exist in the previous case.
41	INPRL	Sampled data phase root locus requested as in the previous case but no phase variations exist in the previous case.
42	INPRL	Sampled data root locus plots requested as in the previous case but no plot specifications exist in the previous case.
43	INPRL	Number of sampled data root locus gain variations is out of range.
44	INPRL	Number of sampled data root locus phase variations is out of range.
45	INPRL	Number of sampled data root locus plot frames is out of range.
46	INPRL	Number of grids on a sampled root locus plot frame is out of range.
47	MLTZRO	Continuous system without a zero order hold circuit has three open loop poles at the origin. Such a system is restricted to a maximum of two.
48	MODZRO	Continuous system without a zero order hold circuit has three open loop poles at the origin. Such a system is restricted to a maximum of two.

ERROR CODE	SUBROUTINE	EXPLANATION
49	PFEZRO	Program logic error in working with the poles at the origin in the partial fraction expansion of the continuous system open loop transfer function.
52	PRINT	Program logic error in storing the continuous system characteristic matrix elements.
54	RLOCUS	Program logic error in storing the continuous system characteristic matrix elements.
56	RUTER	No eigenvalues computed in a sampled data gain or phase variation case.
59	SRRL	No eigenvalues computed in a sample period root locus case.
61	VRYGEN	Number of coefficients in the nominal characteristic matrix to vary in general format is out of range.
62	VRYGEN	User attempted to vary a zero element in the nominal characteristic matrix which did not exist in the storage arrays. Input the zero coefficients in the nominal matrix prior to the variation.
63	VRYGEN	A characteristic matrix element variation had a row or column designation which was less than 1 or exceeded the matrix order.
64	VRYGEN	User attempted to vary a coefficient in the nominal matrix that was not previously defined in the storage arrays. Define the coefficient as zero in the nominal matrix input prior to the variation.
101	AFTVAR	Cannot restore nominal raw data parameter values since only a general matrix definition is permitted.
102	BHA272	Requested sampled data root locus but no Z-domain transfer function exists.
103	CSOLTF	Sampling device location is outside the matrix dimension.

ERROR CODE	SUBROUTINE	EXPLANATION
104	CSOLTF	The matrix is ill-defined for an open loop zeros computation.
105	CSOLTF	The matrix is ill-defined for an open loop poles computation.
106	RAWMTX	User attempted to define a matrix by inputting raw data parameters. Only a general matrix definition is permitted.
107	VRYRAW	User attempted to vary raw parameter values off-nominal. Only variations of a general matrix definition are permitted.
108	DATA	The user input a root locus request for a system without a zero order hold device which is illegal.
109	SRRL	Requested nominal case sampled data root locus calculations but no Z-domain transfer function exists.