

UNCLASSIFIED

AD NUMBER

AD871086

LIMITATION CHANGES

TO:

Approved for public release; distribution is unlimited.

FROM:

Distribution: Further dissemination only as directed by Army Materiel Command, Washington, DC 20315, MAR 1970, or higher DoD authority.

AUTHORITY

AMC ltr 31 Aug 1971

THIS PAGE IS UNCLASSIFIED

AD

AMSAA.TM 69



AMSAA

TECHNICAL MEMORANDUM NO. 69

A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL:
LINE PRINTER PLOTS OF CHARACTERISTIC
EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF
SYSTEM TRANSFER FUNCTIONS

by

Harold H. Burke
Robert L. Payne, Jr.

March 1970

This document may be further distributed by the holder only with specific prior approval of the Commanding General, U.S. Army Materiel Command, ATTN: AMCPA-SA, Washington, D.C. 20315.

U.S. ARMY ABERDEEN RESEARCH AND DEVELOPMENT CENTER
ARMY MATERIEL SYSTEMS ANALYSIS AGENCY
ABERDEEN PROVING GROUND, MARYLAND

AD 871086 L

AD 871086



DISPOSITION

Destroy this report when no longer needed. Do not return it to the originator.

DISCLAIMER

The findings in this report are not to be construed as an official Department of the Army position.

WARNING

Information and data contained in this document are based on the input available at the time of preparation. The results may be subject to change and should not be construed as representing the AMC position unless so specified.

A R M Y M A T E R I E L S Y S T E M S A N A L Y S I S A G E N C Y

TECHNICAL MEMORANDUM NO. 69

MARCH 1970

A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL:
LINE PRINTER PLOTS OF CHARACTERISTIC
EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS

Harold H. Burke

Robert L. Payne, Jr.

Combat Support Division

This document may be further distributed by the holder only with specific prior approval of the Commanding General, U.S. Army Materiel Command, ATTN: AMCPA-SA, Washington, D.C. 20315.

Project No. RDT&E 1P765801M1102

A B E R D E E N P R O V I N G G R O U N D , M A R Y L A N D

TECHNICAL MEMORANDUM NO. 69

HHBurke/RLPayne, Jr./flz
Aberdeen Proving Ground, Md.
March 1970

A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL:
LINE PRINTER PLOTS OF CHARACTERISTIC
EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS

ABSTRACT

A computer program to perform dynamic systems analyses and plot the results is presented. Linear systems and a large classification of nonlinear systems representing engineering, scientific, and economic disciplines can be modeled to permit application of the computer program. Two examples are given to demonstrate the capabilities of the analysis tool. The mathematical model of a missile guidance and control system is analyzed and a ratio of polynomials representing the closed loop transfer function of a high performance model follower aircraft is evaluated. Linear differential equations to the 100th order having real or complex roots can be studied. System characteristic equation root loci and system transfer functions are plotted.

TABLE OF CONTENTS

ABSTRACT 3

LIST OF FIGURES 6

LIST OF TABLES 7

1. INTRODUCTION 9

2. THE PROBLEM 11

 2.1 Linear System 11

 2.2 Nonlinear System 15

3. GRAPHICAL METHODS 18

4. DATA FORMATS 19

 4.1 General Description 19

 4.2 Input Description 20

 4.3 Output Description

5. EXAMPLES 22

6. CONCLUSIONS 25

REFERENCES 55

APPENDIX A - LISTING OF SOURCE DECK 57

APPENDIX B - FLOW CHART OF SOURCE DECK 87

DISTRIBUTION LIST 229

LIST OF FIGURES

1. Linear Closed Loop System
2. Nonlinear Closed Loop System
3. Stability Boundary Popov Modified Phase Amplitude Characteristic
4. System Nonlinearity
5. Example 1, Linear System Bode Plot, Phase Angle
6. Example 1, Linear System Bode Plot, Magnitude
7. Example 1, Nonlinear System, Bode Plot, Phase Angle
8. Example 1, Nonlinear System, Bode Plot, Magnitude
9. Example 1, Linear System, Expanded Bode Plot (1 to 300 rad/s)
Phase Angle
10. Example 1, Linear System, Expanded Bode Plot (1 to 300 rad/s)
Magnitude
11. Example 1, Nonlinear System Expanded Bode Plot (1 to 300 rad/s)
Phase Angle
12. Example 1, Nonlinear System Expanded Bode Plot (1 to 300 rad/s)
Magnitude
13. Example 1, Linear System Expanded Bode Plot (120 to 200 rad/s)
Phase Angle
14. Example 1, Linear System Expanded Bode Plot (120 to 200 rad/s)
Magnitude
15. Example 1, Nonlinear System Expanded Bode Plot, (120 to 200 rad/s)
Phase Angle
16. Example 1, Nonlinear System Expanded Bode Plot, (120 to 200 rad/s)
Magnitude
17. Example 2, Linear System, Bode Plot, Phase Angle
18. Example 2, Linear System Bode Plot, Magnitude
19. Example 2, Nonlinear System, Bode Plot, Phase Angle
20. Example 2, Nonlinear System, Bode Plot, Magnitude
21. Example 2, Linear System, Expanded Bode Plot (1 to 200 rad/s),
Phase Angle
22. Example 2, Linear System, Expanded Bode Plot (1 to 200 rad/s),
Magnitude
23. Example 2, Nonlinear System, Expanded Bode Plot (1 to 200 rad/s),
Phase Angle
24. Example 2, Nonlinear System, Expanded Bode Plot (1 to 200 rad/s),
Magnitude

LIST OF TABLES

1. Input for Example 1.
2. Mirror of Input, Example 1.
3. Equivalent System Open Loop.
4. Standard Frequency Response and Popov Response, Example 1.
5. Mirror of Input, Example 2.
6. Equivalent System Open Loop Polynomials and Roots, Example 2.

1. INTRODUCTION

Systems analyses require that mathematical models be developed and exercised to determine cause-and-effect relationships. One powerful method is to generate linearized equations for each physical phenomenon occurring in the process and combine these into input-output relationships through the application of Laplace transformations.

Studying the characteristic equations of the input-output relations determines parametric stability trends.⁽¹⁾ Steady state frequency response of the system provides another useful characterization of system stability. Application of the inverse Laplace transformation to the process transfer functions provides transient histories of output for specific inputs.⁽²⁾

This report discusses a program that mechanizes another part of the analytical procedure, the determination of the frequency response of linear systems and the modified phase-amplitude characteristic for nonlinear systems. Reference 1, published in November 1968, discusses a program that mechanized the determination of the roots of the characteristic equation. Reference 2, discusses a program that determines the transient response of the process when it is subjected to input forcing functions.

Application of this computer program, which is an extension of the material contained in Reference 1, along with the computer program presented in Reference 2, permits one to determine quickly and accurately the stability and performance characteristics of high-order linear dynamic processes and to qualitatively predict the stability and performance of a large class of nonlinear systems.

The techniques developed in this report have been applied to specific studies.⁽³⁾⁽⁴⁾ A more extensive utilization of the methods will be contained in a forthcoming report which expands Chapter 4.2 of Reference 5. Preliminary work has been commenced to explore the utility of these methods in providing solutions to optimal inventory problems.⁽⁶⁾⁽⁷⁾

A useful linear system analysis method is to determine the steady state frequency response of the system. Evaluation of the polynomials representing the system's transfer function is a time-consuming task. Conversion of the tabulated results to a graphical display is then required before conclusions relevant to system stability can be appreciated.

The analysis of nonlinear systems is a difficult task, but for certain classes of systems it is possible to obtain qualitative indications of stability. Similarly, the task of obtaining the tabular data and converting it to a graphical form before conclusions can be drawn is a laborious process.

These two methods (one linear and one nonlinear) have been coded in a FORTRAN IV program and integrated into the root locus methods discussed in Reference 1. Main features of the program are:

1. FORTRAN IV program. No machine-oriented or object language.
2. No FORTRAN complex type statements necessary.
3. No special graphical plotting equipment.
4. Order of polynomial may be up to 100.
5. Number of variations of given parameter may be up to 100.

Main features of the graphical displays are:

ROOT LOCUS Method⁽¹⁾

1. Log plot of the third and fourth quadrants of the complex frequency plane from 0 to 10,000 radians/second.
2. Linear plots of selected regions of the third and fourth quadrants of the complex frequency plane with arbitrary scales.

LINEAR FREQUENCY RESPONSE Method

1. Plot of Magnitude (in decibels) and phase (in degrees) of transfer function vs 6 decades of frequency, beginning at a selectable minimum frequency.
2. Linear frequency plots of magnitude (in decibels) and phase (in degrees) of transfer function for selectable ranges of frequency.

NONLINEAR FREQUENCY RESPONSE Method

1. Plot of "Modified Magnitude Characteristic" (in decibels) and "Modified Phase Characteristic" (in degrees) of transfer function vs 6 decades of frequency, beginning at same selectable minimum frequency as linear frequency response method.

2. Linear frequency plots of "Modified Magnitude Characteristics" (in decibels) and "Modified Phase Characteristic" (in degrees) of transfer function for same selectable ranges of frequency as linear frequency response methods.

2. THE PROBLEM

2.1 Linear System.

Regardless of the complexity of a linear closed-loop system, its transfer function can be reduced to the equivalent form shown in Figure 1. For multiple loop systems, the G's and H's are readily expressed as sums of products of polynomials which are identified with individual elements making up the complete system.

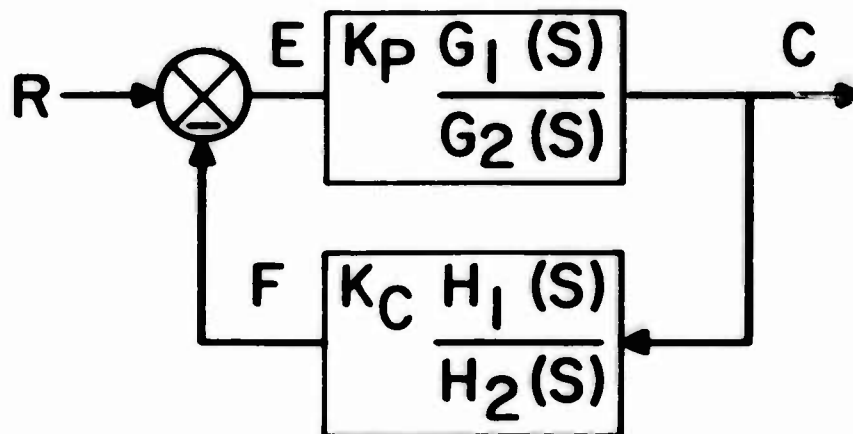


Figure 1. Linear Closed Loop System.

R = system input,
C = system output,
F = system feedback,
E = system error,
 K_p = process gain, and
 K_c = controller gain.

The fractions $G_1(s)/G_2(s)$ and $H_1(s)/H_2(s)$ are equivalent transfer fractions of the system and are represented by ratios of polynomials which upon expansion can be put into the form

$$\frac{\sum_{\mu=0}^{n-k} a_{\mu} s^{n-\mu}}{\sum_{i=0}^{m+n-l} b_i s^{m+n-i}}, \quad (1)$$

or in factored form

$$\frac{s^k \prod_{\mu=1}^v (s+z_{\mu}) \prod_{q=v+1}^y [s+(\sigma_q+j\omega_q)] [s+(\sigma_q-j\omega_q)]}{s^l \prod_{i=1}^g (s+p_i) \prod_{t=g+1}^r [s+(\sigma_t+j\omega_t)] [s+(\sigma_t-j\omega_t)]} \quad (2)$$

where $y = \frac{n-k-v}{2}$, $r = \frac{m+n-l-g}{2}$

where the a's and b's are real, the poles and zeros can be either real or complex in conjugate pairs, and $m \geq 1$.

There are two transfer functions of fundamental importance.

OPEN LOOP TRANSFER FUNCTION:

$$\frac{F}{E} = \left[K_p \frac{G_1(s)}{G_2(s)} \right] \left[K_c \frac{H_1(s)}{H_2(s)} \right]. \quad (3)$$

CLOSED LOOP TRANSFER FUNCTION:

$$\frac{C}{R} = \frac{K_p G_1(s) H_2(s)}{K_c K_p G_1(s) H_1(s) + G_2(s) H_2(s)} \quad (4)$$

The linear system's closed-loop characteristic equation, which determines stability is

$$K_c K_p G_1(s)H_1(s) + G_2(s)H_2(s) = 0. \quad (5)$$

Dividing both sides of Equation (5) by $G_2(s)H_2(s)$ gives

$$K_c K_p \frac{G_1(s)H_1(s)}{G_2(s)H_2(s)} + 1 = 0 \quad (6)$$

If we let

$$K_c K_p = K^*, \quad (7)$$

$$G_1(s)H_1(s) = A(s), \text{ and} \quad (8)$$

$$G_2(s)H_2(s) = B(s), \text{ then Equation (6) becomes} \quad (9)$$

$$\frac{K^* A(s)}{B(s)} + 1 = 0. \quad (10)$$

In terms of the system open-loop transfer function,

$$\frac{F}{E} + 1 = 0. \quad (11)$$

This expression relates the system's closed-loop characteristic equation to the system's open-loop transfer function.

The steady state frequency response of the open-loop transfer is obtained by evaluating Equation (3) along the imaginary axis for values of ω between zero and infinity rad/s.

The result is a vector quantity,

$$\frac{F}{E} \Big|_{0 \leq \omega \leq \infty} = K^* \frac{A(s)}{B(s)} \Big|_{0 \leq s \leq \infty} = M_{oL} \angle \phi_{oL}, \quad (12)$$

where

M_{oL} = scalar magnitude of open-loop transfer function between
 $0 \leq \omega \leq \infty$, and

ϕ_{oL} = direction of magnitude of open-loop transfer function
between $0 \leq \omega \leq \infty$.

Equation (10) can be rewritten as

$$M_{oL} \angle \phi_{oL} + 1 \angle 0^\circ = 0, \quad (13)$$

or

$$M_{oL} \angle \phi_{oL} = 1 \angle 180^\circ. \quad (14)$$

Equation 14 is the form required to provide quantitative stability information for a closed-loop system.

There are two figures of merit which provide a measure of system stability. The only restriction is that all roots of the system's open-loop transfer function denominator, $B(s)$, be ≤ 0 .

GAIN MARGIN is the amount that M_{oL} must be increased or decreased to make it equal to 1, only when the orientation of magnitude of the open-loop transfer function, $\angle \frac{A(s)}{B(s)}$, is 180° .

PHASE MARGIN is the amount that ϕ_{oL} must be increased or decreased to make it equal to 180° , only when the magnitude of the open-loop transfer function, $\left| \frac{A(s)}{B(s)} \right|$, is 1.

These concepts can be related directly to Equation 14. Incipient instability is identified with either a zero gain or zero phase margin.

The scalar magnitude of the open-loop transfer function is generally expressed in decibels. The decibel equivalent, $M_{oL} \text{ (db)}$, of a number, M_{oL} , is $M_{oL} \text{ (db)} = 20 \log_{10} M_{oL}$.

The gain margin and phase margin of a linear closed-loop system as shown in Figure 1 and Equation (4) can be readily determined by analyzing the open-loop transfer functions of Equation (3) in accordance with Equation (14).

2.2 Nonlinear System.

V.M. Popov's plots of system transfer functions give some insight into stability of nonlinear systems.⁽⁸⁾⁽⁹⁾ Figure 2 shows the system configurations considered in this analysis. It contains a linear part and one nonlinear element and is subject to the following restrictions.

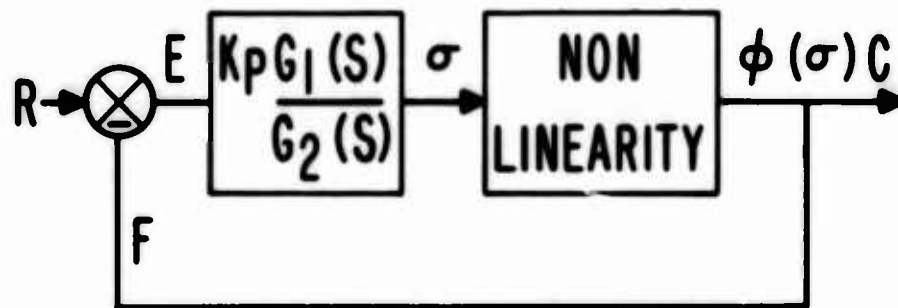


Figure 2. Nonlinear Closed Loop System

1. The roots of $G_2(s)$ have negative real parts and simple or multiple roots on the imaginary axis (one root permitted at origin).

2.
$$\epsilon \leq \frac{\phi(\sigma)}{\sigma} \leq K. \quad (15)$$

3. The linear system of Figure 2, obtained by substituting $\phi = \epsilon\sigma$, is stable.

If there exists a non-negative real number q such that

$$\operatorname{Re} \left[(1 + jq\omega) G(j\omega) \right] + \frac{1}{K} > 0 \quad (16)$$

for all $\omega > 0$, where G is defined by $\frac{G_1(s)}{G_2(s)}$, the nonlinear system is

asymptotically stable in the large.

Since G has been defined as the ratio of two polynomials, it may be expressed as

$$G(j\omega) = \frac{G_{1R}(\omega) + jG_{1I}(\omega)}{G_{2R}(\omega) + jG_{2I}(\omega)}, \quad (17)$$

where

G_{1R} , G_{2R} = real parts of G_1 , and G_2 , respectively, and

G_{1I} , G_{2I} = imaginary parts of G_1 and G_2 , respectively.

Redefining $G(j\omega)$, we have

$$G(j\omega) = G_R(\omega) + jG_I(\omega), \quad (18)$$

where

$$G_R = \frac{G_{1R}G_{2R} + G_{1I}G_{2I}}{(G_{2R})^2 + (G_{2I})^2}, \text{ and}$$

$$G_I = \frac{G_{1I}G_{2R} - G_{1R}G_{2I}}{(G_{2R})^2 + (G_{2I})^2}.$$

As a result, the inequality,

$$\operatorname{Re} \left[(1 + jq\omega) G(j\omega) \right] + \frac{1}{K} > 0, \quad (19)$$

becomes

$$G_R(\omega) - q\omega G_I(\omega) + \frac{1}{K} > 0. \quad (20)$$

To aid in a graphical interpretation of these results, we define

$$U = G_R(\omega), \text{ and} \tag{21}$$

$$W = \omega G_I(\omega).$$

The inequality of Equation (20) may be written as

$$U - q W + \frac{1}{K} > 0, \tag{22}$$

where

$$q \geq 0, K > 0.$$

If we replace the inequality of Equation (22) by an equal sign,

$$U - q W + \frac{1}{K} = 0. \tag{23}$$

Then it may be seen that inequality of Equation (22) represents those points in the U-W plane of the modified phase amplitude characteristic which are to the right of the line of Equation 23, as shown in Figure 3.

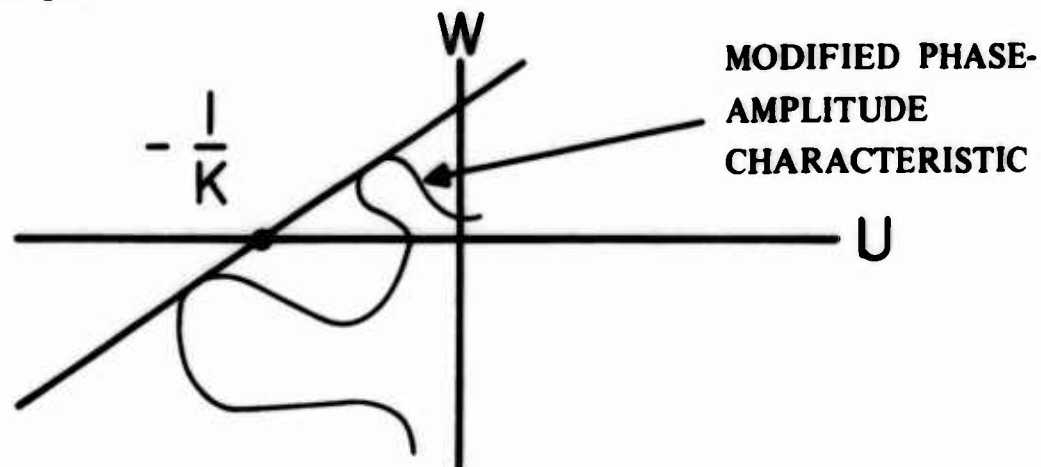


Figure 3. Stability Boundary-Popov Modified Phase Amplitude Characteristic.

The nonlinear system synthesis problem will be concerned with determining the variation of K for a given linear G(s). Figure 4 shows the boundary constraints of the nonlinearity $\phi(\sigma)/\sigma$ of the system shown in Figure 2.

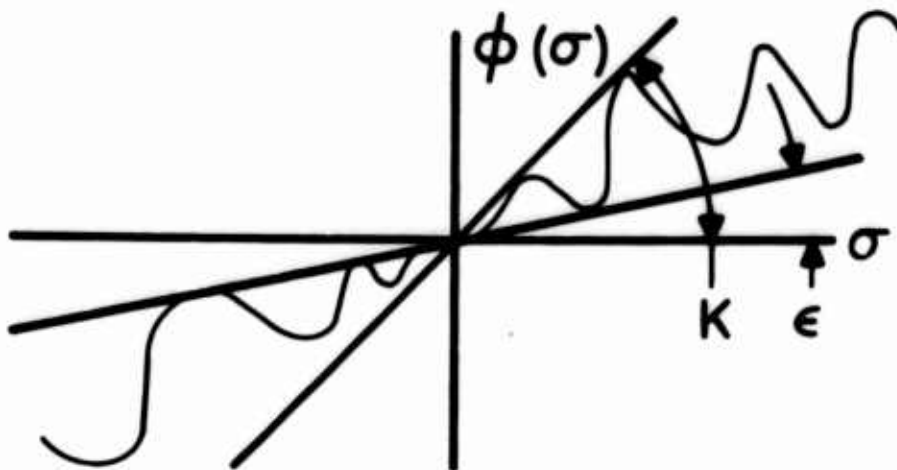


Figure 4. System Nonlinearity.

To maintain symmetry with the Bode Plots, the modified magnitude and phase characteristics are plotted on identical axes. Values of the modified amplitude associated with the modified phase characteristic for $90^\circ \leq \phi \leq 270^\circ$ (-90°) establish the contour described in Figure 3.

Once this contour is established in the U-W plane, the maximum K obtainable is determined by inspection from the intersection of the U axis and a straight line tangent to the modified phase amplitude characteristic.

3. GRAPHICAL METHODS

An on-line printout of the Bode diagram for an open-loop transfer function, F/E , or for a closed-loop transfer function, C/R , is generated for the linear system. Similarly, printouts of the Popov "modified amplitude-phase characteristics" are generated. The phase and amplitude axes are scaled in degrees and decibels with an arbitrarily selectable scale between $\pm \phi$ max degrees $\pm M$ max db, respectively. The frequency axes are logarithmically scaled in radians per second over a range of 6 decades, beginning with an arbitrarily selected lowest

decade. Values of amplitude and phase are calculated for 55 uniformly distributed frequencies over the 6-decade range, plus 17 additional points centered about each root of $\frac{A(s)}{K^*A(s) + B(s)}$. This additional distribution of printouts expressed as a multiple of each frequency is:

(0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8,
0.9, 1.0, 2.0, 4.0, 6.0, 8.0, 10).

An option of the program permits the density of the Bode plot to be either sparse or dense. The sparse plot requires $K^* = 0$ and only one set of frequencies are available, from which the amplitude and phase of $A(s)/B(s)$ are calculated. The dense plot permits K^* to have values consistent with the root locus subroutine.

For each value of K^* , a different set of frequencies will exist for $K^* A(s) + B(s)$.

A larger number of amplitude and phase values will be generated. Use of the dense plot capability may be necessary to obtain precise definition of the graphical plot in some cases.

Any number of simultaneous expands of different portions of the 6-decade frequency range may be plotted.

4. DATA FORMATS

4.1 General Description.

This is a subroutine designed to supplement the methods presented in AMSAA TM No. 21, A Linear Closed Loop System Analysis Procedure Using Line Printer Plots of Characteristic Equation Root Loci. This subroutine utilizes the polynomial multiplication and root locus methods discussed in AMSAA TM No. 21, and extends the analysis capabilities to include the following available options:

1. Root Locus Plots.
2. Frequency Response Plots.
3. Popov Modified Phase-Amplitude Characteristics Plots.

For the root locus option, the output is identical with the output obtained from AMSAA TM No. 21.

For the frequency-response option, the output is a log-log plot of steady state amplitude vs frequency and a semi-log plot of steady state phase angle vs frequency. Linear expands of amplitude and phase for any frequency range may be plotted.

For the Popov modified phase-amplitude characteristic, the output is a log-log plot of steady state amplitude vs frequency and semi-log plot of steady state phase angle vs frequency. Linear expands of amplitude and phase for any frequency range may be plotted.

4.2 Input Description.

a. Program option control card that precedes all data (see below).

b. Program option control card.

To run the program in its entirety with a dense plot in the frequency response, i.e.: Root Locus plus Expands and Frequency Response and Popov Plots plus expands place a one (1) in columns 10, 20, 30, and 40.

To run only the root locus log plot, place a one (1) in column 10 and zeros (0) in columns 20, 30, 40.

To run the root locus expand plot, place a one (1) in column 20 and zeros (0) in columns 10, 30, 40.

To run the sparse frequency response Popov plot only place a one (1) in column 30 and zeros in columns 10, 20, and 40.

To run the dense frequency response plot only, place a one (1) in columns 30 and 40 and zeros in columns 10 and 20.

To void all plot routines place a zero in columns 10, 20, 30, and 40.

c. Comment Card that follows the program option control card for identification purposes.

d. Program Data Cards.

Same format as described in Chapter IV (Data Formats), paragraph A2 of AMSAA TM No. 21. Control Card used to call All Plot Subroutines is same as control card used to call log plot subroutine of AMSAA TM No. 21 (Section IV, paragraph B2).

Control cards used to call root locus option are same as control cards used to call root locus expand option, paragraph B2, section IV of AMSAA TM No. 21. Table 1 demonstrates this input format for the dense frequency response option. Control cards used to call frequency response, Popov plots, and expands of both are additional features of this report.

<u>DESCRIPTION</u>	<u>COLUMNS</u>	<u>DATA</u>
Data appears only once	a. - 1-10	Smallest frequency, rad/s (real)
	b. - 11-20	Max phase angle, deg, (real)
	c. - 21-30	Max amplitude, DB (real)
Frequency response, Popov	a. - 1-10	<u>Zero</u> , if no expands desired (integer), One, if expands desired
Number of expands desired	a. - 1-10	No. of freq, response phase angle expands (integer)
	b. - 11-20	No. of freq response amplitude expands (integer)
	c. - 21-30	No. of Popov phase angle expands (integer)
	d. - 31-40	No. of Popov amplitude expands (integer)
Region of expand, one card for each expand in consecutive order with a,b,c,d, above.	1-10	Smallest freq, rad/s (real)
	11-20	Largest freq, rad/s (real)

4.3. Output Description.

Depending on the options specified, the output consists of:

Root Locus Plot Option

- a. A list of all of the roots plotted on the complex frequency plane log plot.
- b. Two log plots of the roots calculated by the polynomial multiplication and root locus program.
- c. The specified number of linear expand plots of selectable regions in the complex frequency plane.

FREQUENCY RESPONSE/POPOV PLOT OPTION

- a. A list of amplitude, amplitude in decibels, and phase angle in degrees for various frequencies specified in Section 3.
- b. A log-log plot of amplitude vs frequency and a semi-log plot of phase angle vs frequency.
- c. The specified number of linear expands of the frequency axis for the amplitude and phase angles of the specified transfer function.

5. EXAMPLES

The two examples discussed in AMSAA TM No. 21 will be analyzed, using the frequency response/Popov plot option. The root loci of $A(s)/B(s)$ using K^* as a parameter have previously been determined. The characteristic equation of the system shown in Figure 2 of Reference 1 is

$$K^*A(s) + B(s) = 0.$$

The open-loop transfer function for this expression, as given in Equation (12) is

$$\frac{F}{E} = \frac{K^*A_1(s) + A_2(s) + A_3(s)}{B(s)} \quad (24)$$

EXAMPLE 1.

where

$$K^* = -0.005$$

$$A_1(s) = 2092(s^2 + 1.2s - 1672)(s^2 + 3.2s + 25,600) + 491s^2(s^2 + 6s + 400)$$

$$A_2(s) = 0.081(464)(s + 2.7)(s^2 + 3.2s + 25,600)$$

$$A_3(s) = 0.95(464)(s + 2.7)(s^2 + 3.2s + 25,600)$$

$$B(s) = s(s^2 + 6s + 400)(s^2 + 3.2s + 25,600)$$

Table 1 shows the input card format for this open-loop transfer function.

Lines 2 through 24 apply to the polynomial multiplication and root locus methods and lines 26 through 36 apply to the frequency response and Popov condition plot subroutines.

The basic tabulated numerical output is shown in Tables 2 and 3 and is similar to Tables 2 and 3 of Reference 1. A tabulation of the open-loop frequency response gain and phase and Popov modified phase amplitude characteristics are shown in Table 4. Figures 5 and 6 are the Bode plots of F/E and Figures 7 and 8 are the Popov plots of F/E . Two regions are expanded to demonstrate capability of this program. First the regions of frequencies between 1.0 rad/s and 300 rad/s are shown in Figures 9 and 10 (frequency response), 11 and 12 (Popov response) and secondly the region of frequencies between 120 rad/s and 200 rad/s shown in Figures 13 and 14 (frequency response), 15 and 16 (Popov response) are plotted. Slope information consistent with Bode principles may be applied directly to these plots.

TABLE 1. INPUT FOR EXAMPLE 1

1-10,	11-20	COLUMNS		Remarks
		21-30	31-40	
0	0	1	1	Program option control card
	Frequency Response			
0	4	1	504	Comment Problem (Polynomial multiplication and Root Locus program) ↑ card data cards Same format as AMSAA TM No. 21, Chapter IV, Paragraph A2. ↓
0	1×10^{-9}	1×10^{-9}		
4	3	4	4	
3				
1	.005			
3	25600.	3.2	1.	
3	-1672.	1.2	1.	
1	-2092.			
1	.005			
3	0.	0.	-491.5	
3	400.	6.	1.	
1	.081			
2	0.	464.		
2	2.7	1.		
3	25600.	3.2	1.	
1	.95			
1	464.			
2	2.7	1.		
3	25600.	3.2	1.	
2	0.	1.		
3	400.	6.	1.	
3	25600.	3.2	1.	
0	0	0	10000	Same as paragraph B2, Chapter IV, TM No. 21. Control card used to call all plot sub-routines.
.01	180.	100.		↑ Frequency response and Popov response plot subroutine. ↓
1				
2	2	2	2	
1.	300.			
120.	200.			
1.	300.			
120.	200.			
1.	300.			
120.	200.			
1.	300.			
120.	200.			

EXAMPLE 2

Table 6, and Figures 12 through 14 of Reference 1 are the $A(s)/B(s)$ for a 24th order linear system. Table 5 gives the F/E data input for this system, and Table 6 lists open-loop transfer function. Figures 17 and 18 show the Bode plot and Figures 19 and 20 show the Popov plot. An expand plot of the region between $1 \text{ rad/s} \leq \omega \leq 200 \text{ rad/s}$ is shown in Figures 21 and 22 for the frequency response and in Figures 23 and 24 for the Popov response.

These two examples demonstrate the significant systems analysis capability that can be realized by using the tools developed in this document and the related programs contained in References 1 and 2. Detailed conclusions can be realized when high order feedback control systems are encountered.

6. CONCLUSIONS

The method described in this memorandum for computing linear system root loci, steady state amplitude and phase, and the nonlinear system steady state modified phase-amplitude characteristic has the following merits:

1. Accepts input in the form of sums of products of polynomials.
2. Produces a log plot of the entire complex plane in two graphical displays.
3. Produces selectable scaled linear plots of regions in complex frequency plane.
4. Relates the closed-loop gain to the graphical display.
5. Produces a magnitude (db) vs log frequency steady state response plot.
6. Produces a phase (deg) vs log frequency steady state response plot.
7. Produces selectable scaled linear plots of magnitude and phase of the steady state frequency response.

TABLE 2. MIRROR OF INPUT, EXAMPLE 1.

POLYNOMIAL MULTIPLICATION AND ROOT LOCUS		PROBLEM NO. 504	
DELTA=	0	4	1
K-INITIAL=	.0000000000	.0000000010	.0000000010
NUMBER OF POLY. IN GROUP 1 OF NUMERATOR=	25600.0000000000	4	4
NUMBER OF POLY. IN GROUP 2 OF NUMERATOR=	3.2000000000	3	3
NUMBER OF POLY. IN GROUP 3 OF NUMERATOR=	1.0000000000	4	4
NUMBER OF POLY. IN GROUP 4 OF NUMERATOR=	-1672.0000000000	4	4
NUMBER OF POLY. IN GROUP 1 OF DENOMINATOR=	.0050000000	3	3
C1 1)=	25600.0000000000		
C1 2)=	3.2000000000		
C1 3)=	1.0000000000		
C1 4)=	-1672.0000000000		
C1 5)=	1.2000000000		
C1 6)=	1.0000000000		
C1 7)=	-2092.0000000000		
C1 8)=	.0050000000		
C1 9)=	.0000000000		
C1 10)=	.0000000000		
C1 11)=	-491.5000000000		
C1 12)=	400.0000000000		
C1 13)=	6.0000000000		
C1 14)=	1.0000000000		
C1 15)=	.0810000000		
C1 16)=	.0000000000		
C1 17)=	464.0000000000		
C1 18)=	2.7000000000		
C1 19)=	1.0000000000		
C1 20)=	25600.0000000000		
C1 21)=	3.2000000000		
C1 22)=	1.0000000000		
C1 23)=	.9500000000		
C1 24)=	464.0000000000		
C1 25)=	2.7000000000		
C1 26)=	1.0000000000		
C1 27)=	25600.0000000000		
C1 28)=	3.2000000000		
C1 29)=	1.0000000000		
C1 30)=	.0000000000		
C1 31)=	1.0000000000		
C1 32)=	400.0000000000		
C1 33)=	6.0000000000		
C1 34)=	1.0000000000		
C1 35)=	25600.0000000000		
C1 36)=	3.2000000000		
C1 37)=	1.0000000000		

TABLE 3. EQUIVALENT SYSTEM OPEN LOOP POLYNOMIALS
AND ROOTS, EXAMPLE 1.

COEFFICIENTS ARE GIVEN IN ASCENDING ORDER

THE COEFFICIENTS OF POLYNOMIAL A (ORDER = 4)

4.7818957E 08	1.3620729E 07	7.1376580E 05	6.0177660E 02	2.4666500E 01
---------------	---------------	---------------	---------------	---------------

THE ROOTS OF A

-9.7472771E 00	+1	2.4364008E 01
-2.4509797E 00	+1	-1.6776908E 02
	+1	-2.4509797E 00
	+1	1.6776908E 02

THE COEFFICIENTS OF POLYNOMIAL B (ORDER = 5)

0.0000000E 00	1.0240000E 07	1.5488000E 05	2.6019200E 04	9.2000000E 00	1.0000000E 00
---------------	---------------	---------------	---------------	---------------	---------------

THE ROOTS OF B

0.0000000E 00	+1	0.0000000E 00
-1.6000000E 00	+1	-1.5999200E 02
	+1	-3.0000000E 00
	+1	-1.6000000E 00
	+1	1.9773720E 01
	+1	1.5999200E 02
	+1	-3.0000000E 00
	+1	-1.9773720E 01

TABLE 4. STANDARD FREQUENCY RESPONSE AND POPOV RESPONSE, EXAMPLE 1.

OMEGA	/M/	/M/-DB	PMI	/M/P	/M/P-DB	PMI-P
1.0000E-02	4.6692E 03	7.3386E 01	-8.99923E 01	4.67023E 01	3.33868E 01	-8.92385E 01
2.0000E-02	2.3349E 03	6.7365E 01	-8.99847E 01	4.67024E 01	3.33868E 01	-8.92366E 01
3.0000E-02	1.5566E 03	6.3843E 01	-8.99770E 01	4.67024E 01	3.33868E 01	-8.92359E 01
4.0000E-02	1.1674E 03	6.1344E 01	-8.99694E 01	4.67024E 01	3.33868E 01	-8.92356E 01
5.0000E-02	9.3396E 02	5.9406E 01	-8.99617E 01	4.67025E 01	3.33868E 01	-8.92354E 01
6.0000E-02	7.7830E 02	5.7823E 01	-8.99541E 01	4.67026E 01	3.33868E 01	-8.92353E 01
7.0000E-02	6.6712E 02	5.6484E 01	-8.99464E 01	4.67026E 01	3.33868E 01	-8.92352E 01
8.0000E-02	5.8373E 02	5.5324E 01	-8.99388E 01	4.67027E 01	3.33868E 01	-8.92351E 01
9.0000E-02	5.1887E 02	5.4301E 01	-8.99311E 01	4.67028E 01	3.33869E 01	-8.92351E 01
1.0000E-01	4.6698E 02	5.3386E 01	-8.99235E 01	4.67029E 01	3.33869E 01	-8.92350E 01
2.0000E-01	2.3350E 02	4.7365E 01	-8.98469E 01	4.67047E 01	3.33872E 01	-8.92348E 01
3.0000E-01	1.5567E 02	4.3844E 01	-8.97704E 01	4.67076E 01	3.33878E 01	-8.92348E 01
4.0000E-01	1.1677E 02	4.1346E 01	-8.96938E 01	4.67117E 01	3.33885E 01	-8.92347E 01
5.0000E-01	9.3427E 01	3.9405E 01	-8.96173E 01	4.67170E 01	3.33895E 01	-8.92347E 01
6.0000E-01	7.7867E 01	3.7827E 01	-8.95408E 01	4.67234E 01	3.33907E 01	-8.92347E 01
7.0000E-01	6.6755E 01	3.6489E 01	-8.94643E 01	4.67310E 01	3.33921E 01	-8.92347E 01
8.0000E-01	5.8422E 01	3.5317E 01	-8.93878E 01	4.67398E 01	3.33937E 01	-8.92348E 01
9.0000E-01	5.1943E 01	3.4310E 01	-8.93112E 01	4.67498E 01	3.33956E 01	-8.92348E 01
1.0000E 00	4.6760E 01	3.3397E 01	-8.92348E 01	4.67609E 01	3.33977E 01	-8.92348E 01
2.0000E 00	2.3475E 01	2.7412E 01	-8.84708E 01	4.69385E 01	3.34306E 01	-8.92352E 01
3.0000E 00	1.5758E 01	2.3950E 01	-8.77089E 01	4.72408E 01	3.34863E 01	-8.92360E 01
4.0000E 00	1.1935E 01	2.1536E 01	-8.69521E 01	4.76782E 01	3.35664E 01	-8.92374E 01
5.0000E 00	9.6735E 00	1.9711E 01	-8.62039E 01	4.82660E 01	3.36728E 01	-8.92397E 01
6.0000E 00	8.1958E 00	1.8271E 01	-8.54702E 01	4.90260E 01	3.38085E 01	-8.92435E 01
7.0000E 00	7.1705E 00	1.7111E 01	-8.47602E 01	4.99883E 01	3.39774E 01	-8.92494E 01
8.0000E 00	6.4329E 00	1.6168E 01	-8.40883E 01	5.11943E 01	3.41844E 01	-8.92585E 01
9.0000E 00	5.8933E 00	1.5407E 01	-8.34762E 01	5.27006E 01	3.44363E 01	-8.92720E 01
1.0000E 01	5.4962E 00	1.4806E 01	-8.29572E 01	5.45854E 01	3.47415E 01	-8.92922E 01
2.0000E 01	5.4906E 00	1.4792E 01	-1.26454E 02	8.83866E 01	3.89277E 01	-9.21153E 01
3.0000E 01	1.0616E 00	5.1926E-01	-1.19123E 02	2.03845E 01	2.61860E 01	-9.22945E 01
4.0000E 01	6.6870E-01	3.4953E 00	-1.10005E 02	2.33687E 01	2.73727E 01	-9.07980E 01
5.0000E 01	5.3123E-01	5.4949E 00	-1.05283E 02	2.49600E 01	2.79449E 01	-9.04172E 01
6.0000E 01	4.4704E-01	6.9929E 00	-1.02396E 02	2.58746E 01	2.82575E 01	-9.02609E 01
7.0000E 01	3.8773E-01	8.2292E 00	-1.00430E 02	2.65091E 01	2.84679E 01	-9.01799E 01
8.0000E 01	3.0990E-01	9.2784E 00	-9.89894E 01	2.70354E 01	2.86387E 01	-9.01318E 01
9.0000E 01	2.8321E-01	1.0175E 01	-9.87867E 01	2.75487E 01	2.88020E 01	-9.01007E 01
1.0000E 02	1.0142E-01	1.9877E 01	-9.61141E 01	2.81243E 01	2.89816E 01	-9.00793E 01
2.0000E 02	7.8874E-02	2.2061E 01	-9.31037E 01	2.81696E 01	2.60940E 01	-9.00307E 01
3.0000E 02	6.0455E-02	2.4371E 01	-9.22471E 01	2.36277E 01	2.74684E 01	-9.00104E 01
4.0000E 02	4.8750E-02	2.6240E 01	-9.17748E 01	2.41635E 01	2.76632E 01	-9.00056E 01
5.0000E 02	4.0784E-02	2.7790E 01	-9.14698E 01	2.44628E 01	2.77349E 01	-9.00036E 01
6.0000E 02	3.5036E-02	2.9109E 01	-9.12553E 01	2.45196E 01	2.77701E 01	-9.00025E 01
7.0000E 02	3.0698E-02	3.0257E 01	-9.10959E 01	2.45196E 01	2.77903E 01	-9.00018E 01
8.0000E 02	2.7314E-02	3.1272E 01	-9.09727E 01	2.45554E 01	2.78029E 01	-9.00014E 01
9.0000E 02	2.4599E-02	3.2181E 01	-9.08745E 01	2.45794E 01	2.78114E 01	-9.00011E 01
1.0000E 03	1.2325E-02	3.8184E 01	-9.04358E 01	2.45963E 01	2.78174E 01	-9.00009E 01
2.0000E 03	8.2197E-03	4.1702E 01	-9.02905E 01	2.46493E 01	2.78361E 01	-9.00002E 01
3.0000E 03	6.1656E-03	4.4200E 01	-9.02177E 01	2.46589E 01	2.78395E 01	-9.00001E 01
4.0000E 03	4.9327E-03	4.6138E 01	-9.01742E 01	2.46622E 01	2.78406E 01	-9.00001E 01
5.0000E 03	4.1107E-03	4.7721E 01	-9.01451E 01	2.46638E 01	2.78412E 01	-9.00000E 01
6.0000E 03	3.5235E-03	4.9060E 01	-9.01244E 01	2.46651E 01	2.78417E 01	-9.00000E 01
7.0000E 03	3.0831E-03	5.0220E 01	-9.01088E 01	2.46654E 01	2.78418E 01	-9.00000E 01
8.0000E 03	2.7406E-03	5.1243E 01	-9.00967E 01	2.46657E 01	2.78419E 01	-9.00000E 01
9.0000E 03	2.4665E-03	5.2158E 01	-9.00871E 01	2.46658E 01	2.78419E 01	-9.00000E 01
1.0000E 04	1.9168E 02	4.5651E 01	-8.98135E 01	4.67058E 01	3.33874E 01	-8.92348E 01

TABLE 4. Continued

OMEGA	/M/	/M/-DB	PHI	/M/P	/M/P-DB	PHI-P
1.21020E 00	3.04103E 01	3.16889E 01	-8.90679E 01	4.67894E 01	3.34029E 01	-8.92348E 01
2.43640E 00	1.93216E 01	2.57204E 01	-8.81378E 01	4.70546E 01	3.34520E 01	-8.92355E 01
4.07280E 00	9.90778E 00	1.99195E 01	-8.62984E 01	4.81822E 01	3.36577E 01	-8.92394E 01
7.30920E 00	6.91694E 00	1.67983E 01	-8.45476E 01	5.03329E 01	3.40370E 01	-8.92518E 01
9.74560E 00	5.58789E 00	1.49450E 01	-8.30780E 01	5.40646E 01	3.46583E 01	-8.92863E 01
1.21820E 01	5.01793E 00	1.40105E 01	-8.24340E 01	6.05999E 01	3.56494E 01	-8.93753E 01
1.46184E 01	4.99486E 00	1.39705E 01	-8.42032E 01	7.26453E 01	3.72241E 01	-8.94021E 01
1.70548E 01	5.51717E 00	1.48343E 01	-8.32164E 01	9.39466E 01	3.94576E 01	-8.94188E 01
1.94912E 01	5.74973E 00	1.51929E 01	-1.19288E 02	9.77842E 01	3.98054E 01	-8.94164E 01
2.19276E 01	3.81459E 00	1.6290E 01	-1.47424E 02	4.51501E 01	3.30932E 01	-8.94082E 01
2.43640E 01	2.22777E 00	6.95739E 00	-1.54187E 02	2.37191E 01	2.75020E 01	-8.94850E 01
4.87280E 01	5.44700E-01	-5.27885E 00	-1.10831E 02	2.78979E 01	2.78918E 01	-8.94474E 01
9.74560E 01	2.89893E-01	-1.07552E 01	-9.81357E 01	2.79675E 01	2.89331E 01	-8.94084E 01
1.46184E 02	2.68970E-01	-1.14059E 01	-9.57300E 01	3.91226E 01	3.18486E 01	-8.94039E 01
1.94912E 02	1.00522E-01	-1.99547E 01	-9.66996E 01	1.94593E 01	2.57825E 01	-8.94034E 01
2.43640E 02	9.34485E-02	-2.05886E 01	-9.40766E 01	2.27102E 01	2.71244E 01	-8.94016E 01
1.67769E 00	2.79435E 01	2.89247E 01	-8.87167E 01	4.68680E 01	3.34175E 01	-8.92350E 01
3.38845E 00	6.20329E 00	1.58524E 01	-8.38419E 01	5.17400E 01	3.42765E 01	-8.92631E 01
1.67769E 01	5.43532E 00	1.47045E 01	-9.15162E 01	9.11560E 01	3.91957E 01	-8.90904E 01
3.35338E 01	8.47953E-01	-1.43257E 00	-1.30640E 02	2.15969E 01	2.66878E 01	-8.91465E 01
5.03307E 01	5.27868E-01	-5.54949E 00	-1.09800E 02	2.49979E 01	2.79581E 01	-8.94098E 01
6.71076E 01	4.03019E-01	-7.89307E 00	-1.03111E 02	2.63421E 01	2.84130E 01	-8.94198E 01
8.38845E 01	3.29467E-01	-9.64377E 00	-9.98217E 01	2.72321E 01	2.87016E 01	-8.94182E 01
1.00661E 02	2.82415E-01	-1.09816E 01	-9.78117E 01	2.81665E 01	2.89947E 01	-8.94078E 01
1.17438E 02	2.53446E-01	-1.19223E 01	-9.64261E 01	2.89772E 01	2.94191E 01	-8.94054E 01
1.34215E 02	2.44201E-01	-1.22451E 01	-9.54978E 01	3.26247E 01	3.02709E 01	-8.94041E 01
1.50992E 02	3.09045E-01	-1.01996E 01	-9.71819E 01	4.62973E 01	3.33111E 01	-8.94047E 01
1.67769E 02	4.62744E-02	-2.66932E 01	-1.73228E 02	9.16547E-01	-7.56909E-01	-8.92873E 01
3.35539E 02	7.12907E-02	-2.29393E 01	-9.27262E 01	2.38937E 01	2.75657E 01	-8.94008E 01
6.71076E 02	3.65269E-02	-2.87477E 01	-9.13105E 01	2.45059E 01	2.77854E 01	-8.94002E 01
1.00661E 03	2.44385E-02	-3.22385E 01	-9.08687E 01	2.45973E 01	2.77817E 01	-8.94009E 01
1.34215E 03	1.83508E-02	-3.47269E 01	-9.06503E 01	2.46280E 01	2.78286E 01	-8.94000E 01
1.67769E 03	1.46886E-02	-3.66604E 01	-9.05198E 01	2.46420E 01	2.78335E 01	-8.94000E 01
1.97737E-01	2.36175E 02	4.74647E 01	-8.98487E 01	4.67046E 01	3.33872E 01	-8.92348E 01
9.88686E-01	4.72946E 01	3.4962E 01	-8.92434E 01	4.67596E 01	3.33974E 01	-8.92348E 01
1.97737E 00	2.37412E 01	2.75101E 01	-8.84878E 01	4.69331E 01	3.34296E 01	-8.92352E 01
3.95474E 00	1.20658E 01	2.16311E 01	-8.69862E 01	4.76553E 01	3.35622E 01	-8.92373E 01
5.93212E 00	8.27937E 00	1.83599E 01	-8.55194E 01	4.89684E 01	3.37983E 01	-8.92432E 01
7.90949E 00	6.49057E 00	1.62457E 01	-8.41471E 01	5.10737E 01	3.41640E 01	-8.92575E 01
9.88686E 00	5.53794E 00	1.48670E 01	-8.30098E 01	5.43500E 01	3.47040E 01	-8.92895E 01
1.18642E 01	5.06005E 00	1.40831E 01	-8.24330E 01	5.95146E 01	3.54925E 01	-8.93585E 01
1.38416E 01	4.94359E 00	1.38809E 01	-8.31900E 01	6.79470E 01	3.66434E 01	-8.95057E 01
1.58190E 01	5.18828E 00	1.43005E 01	-8.71953E 01	8.19754E 01	3.82737E 01	-8.98226E 01
1.77963E 01	5.73111E 00	1.51648E 01	-9.69839E 01	1.00746E 02	4.00645E 01	-8.95090E 01
1.97737E 01	5.62277E 00	1.49940E 01	-1.23276E 02	9.30043E 01	3.93701E 01	-8.91909E 01
3.95474E 01	6.77500E-01	-3.38181E 00	-1.19733E 02	2.32685E 01	2.73354E 01	-8.948274E 01
7.90949E 01	3.47132E-01	-9.19010E 00	-1.00582E 02	2.69895E 01	2.86239E 01	-8.941353E 01
1.18642E 02	2.52016E-01	-1.19714E 01	-9.63434E 01	2.97168E 01	2.94600E 01	-8.940537E 01
1.58190E 02	6.52422E-01	-3.70943E 00	-1.22404E 02	8.71371E 01	3.88041E 01	-8.940229E 01
1.97737E 02	1.01144E-01	-1.99012E 01	-9.63511E 01	1.98773E 01	2.59671E 01	-8.940323E 01
1.59992E 00	2.92886E 01	2.93340E 01	-8.87761E 01	4.68529E 01	3.34147E 01	-8.92350E 01
7.90949E 00	6.43321E 00	1.61686E 01	-8.40886E 01	5.11937E 01	3.41843E 01	-8.92584E 01
1.59992E 01	5.22917E 00	1.43687E 01	-8.78503E 01	8.36039E 01	3.84445E 01	-8.98656E 01
3.14984E 01	9.23181E-01	-6.94260E-01	-1.34545E 02	2.10632E 01	2.64705E 01	-8.917618E 01
4.79976E 01	5.52811E-01	-5.14847E 00	-1.11339E 02	2.47154E 01	2.78594E 01	-8.94663E 01
6.39964E 01	4.21072E-01	-7.51288E 00	-1.03974E 02	2.61494E 01	2.83492E 01	-8.940229E 01
7.99960E 01	3.43635E-01	-9.27806E 00	-1.00431E 02	2.70352E 01	2.86386E 01	-8.9401319E 01

TABLE 4. Continued

OMEGA	/M/	/M/-DB	PHI	/M/P	/M/P-DB	PHI-P
9.59957E 01	2.93512E-01	-1.06475E 01	-9.82913E 01	2.78812E 01	2.89062E 01	-9.00870E 01
1.11994E 02	2.61011E-01	-1.16662E 01	-9.68262E 01	2.90268E 01	2.92560E 01	-9.00612E 01
1.27994E 02	2.44497E-01	-1.22345E 01	-9.57759E 01	3.11352E 01	2.98650E 01	-9.00453E 01
1.43993E 02	2.59742E-01	-1.17092E 01	-9.55129E 01	3.72280E 01	3.14174E 01	-9.00384E 01
1.59992E 02	8.01243E-01	-1.42472E 00	-1.67605E 02	2.75277E 01	2.87954E 01	-9.16290E 01
3.19984E 02	7.44499E-02	-2.25627E 01	-9.28779E 01	2.37927E 01	2.75289E 01	-9.00090E 01
6.39468E 02	3.82770E-02	-2.83412E 01	-9.13757E 01	2.44890E 01	2.77794E 01	-9.00022E 01
9.59952E 02	2.56193E-02	-3.18287E 01	-9.09113E 01	2.45902E 01	2.78152E 01	-9.00009E 01
1.27994E 03	1.92399E-02	-3.43160E 01	-9.06821E 01	2.46241E 01	2.78272E 01	-9.00005E 01
1.59992E 03	1.54011E-02	-3.62489E 01	-9.05451E 01	2.46395E 01	2.78326E 01	-9.00003E 01

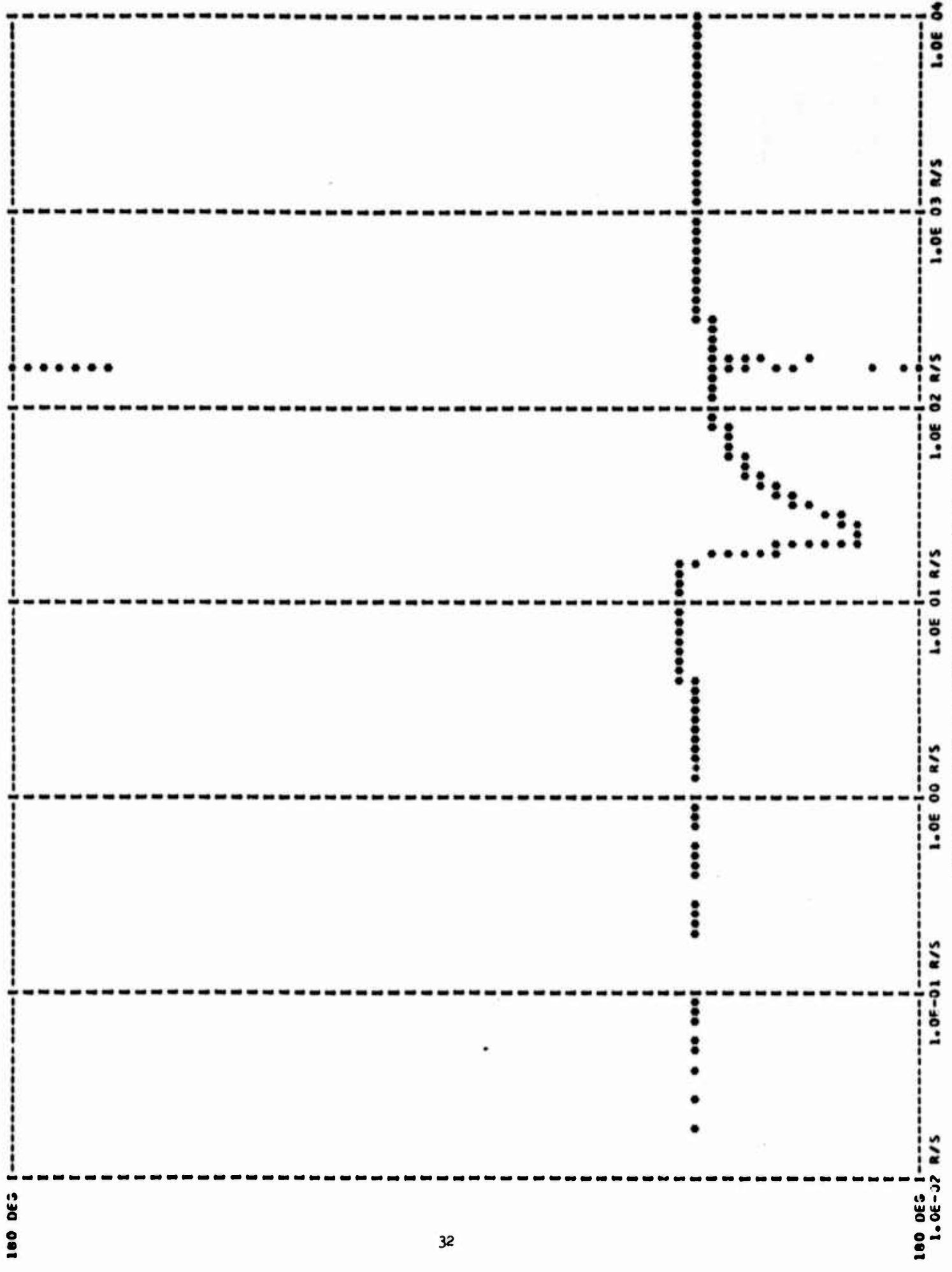


FIGURE 5. EXAMPLE 1, LINEAR SYSTEM BODE PLOT, PHASE ANGLE

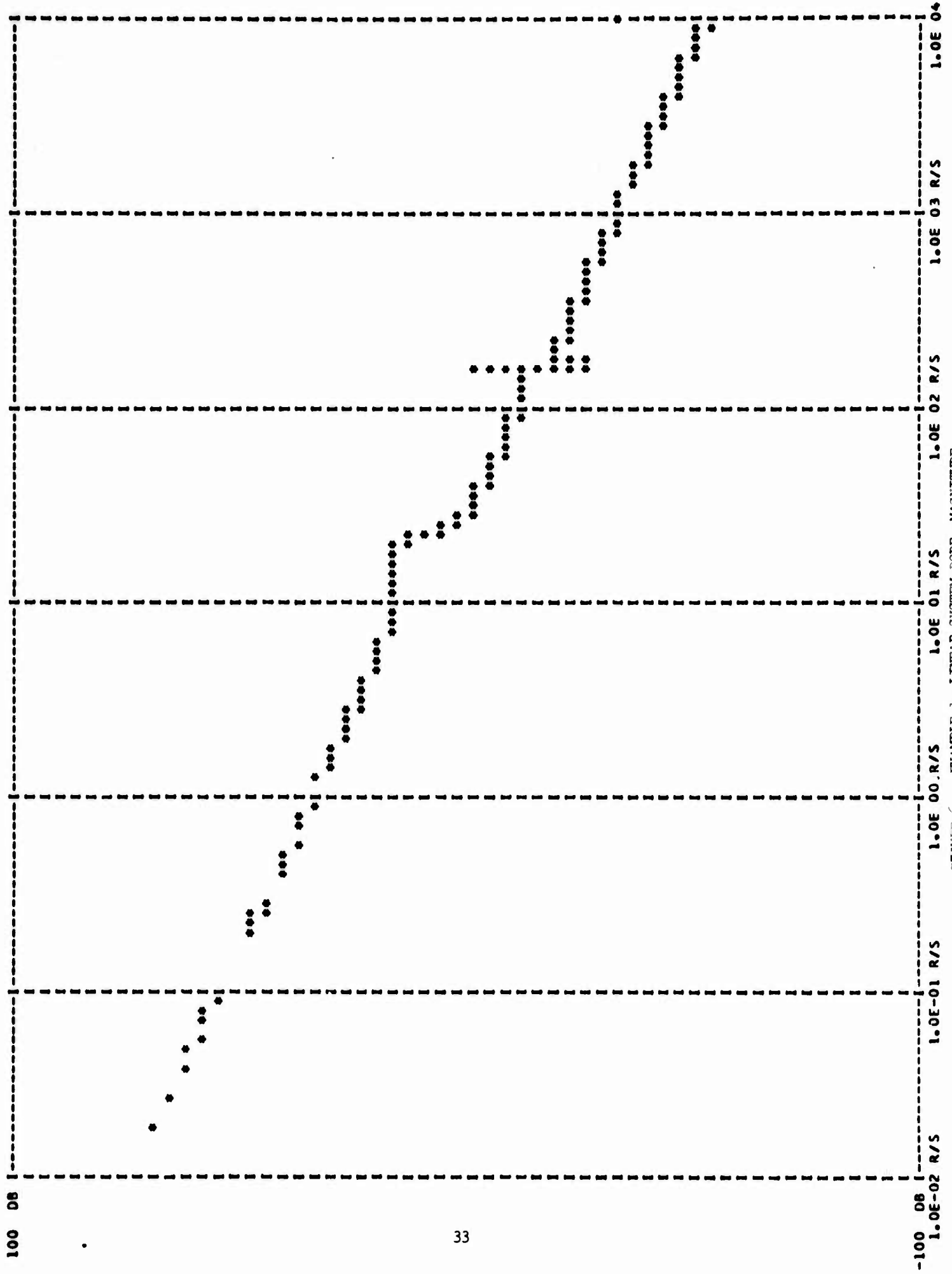


FIGURE 6. EXAMPLE 1, LINEAR SYSTEM BODE, MAGNITUDE

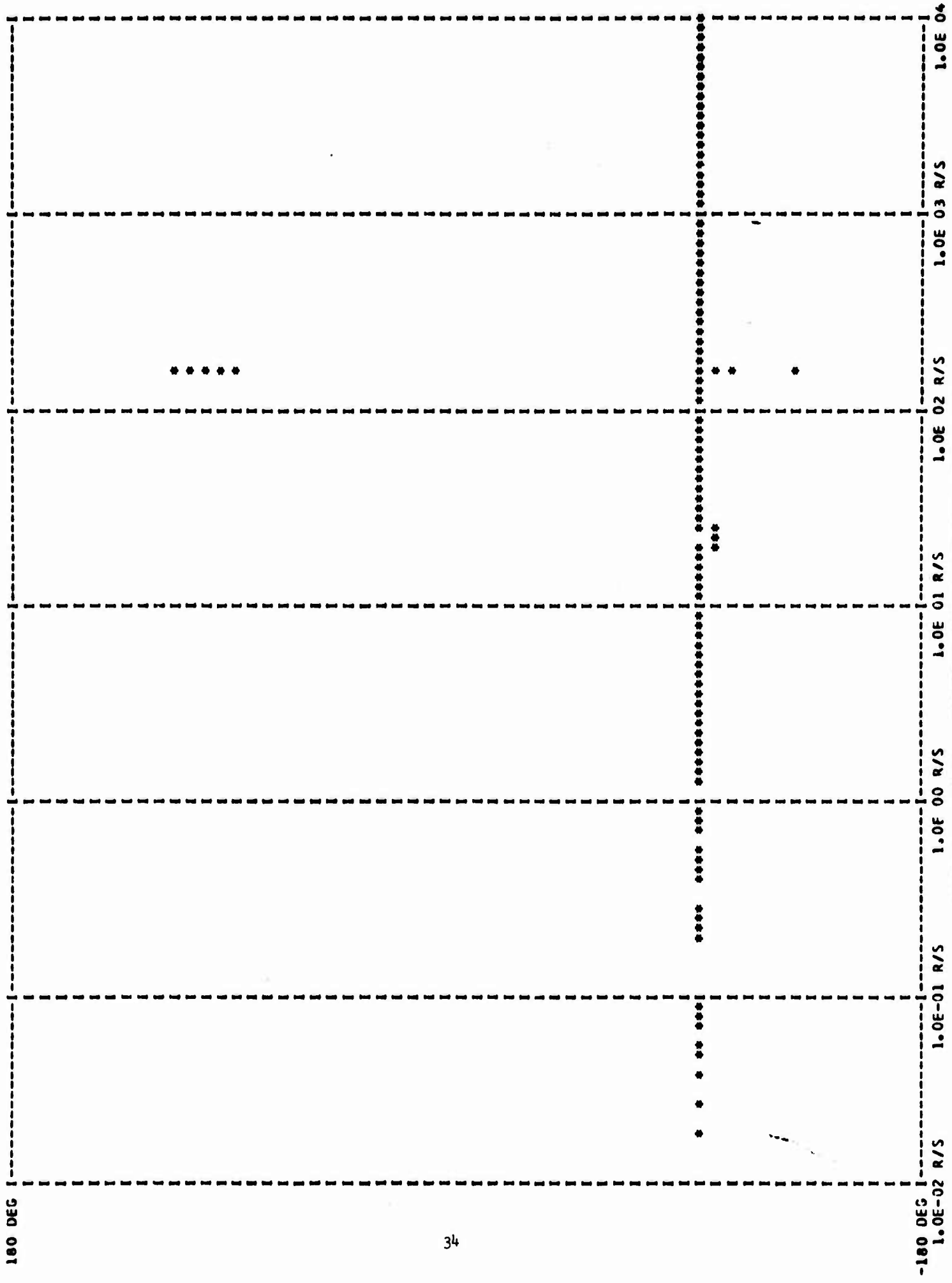


FIGURE 7. EXAMPLE 1. NONLINEAR SYSTEM, BODE PLOT, PHASE ANGLE

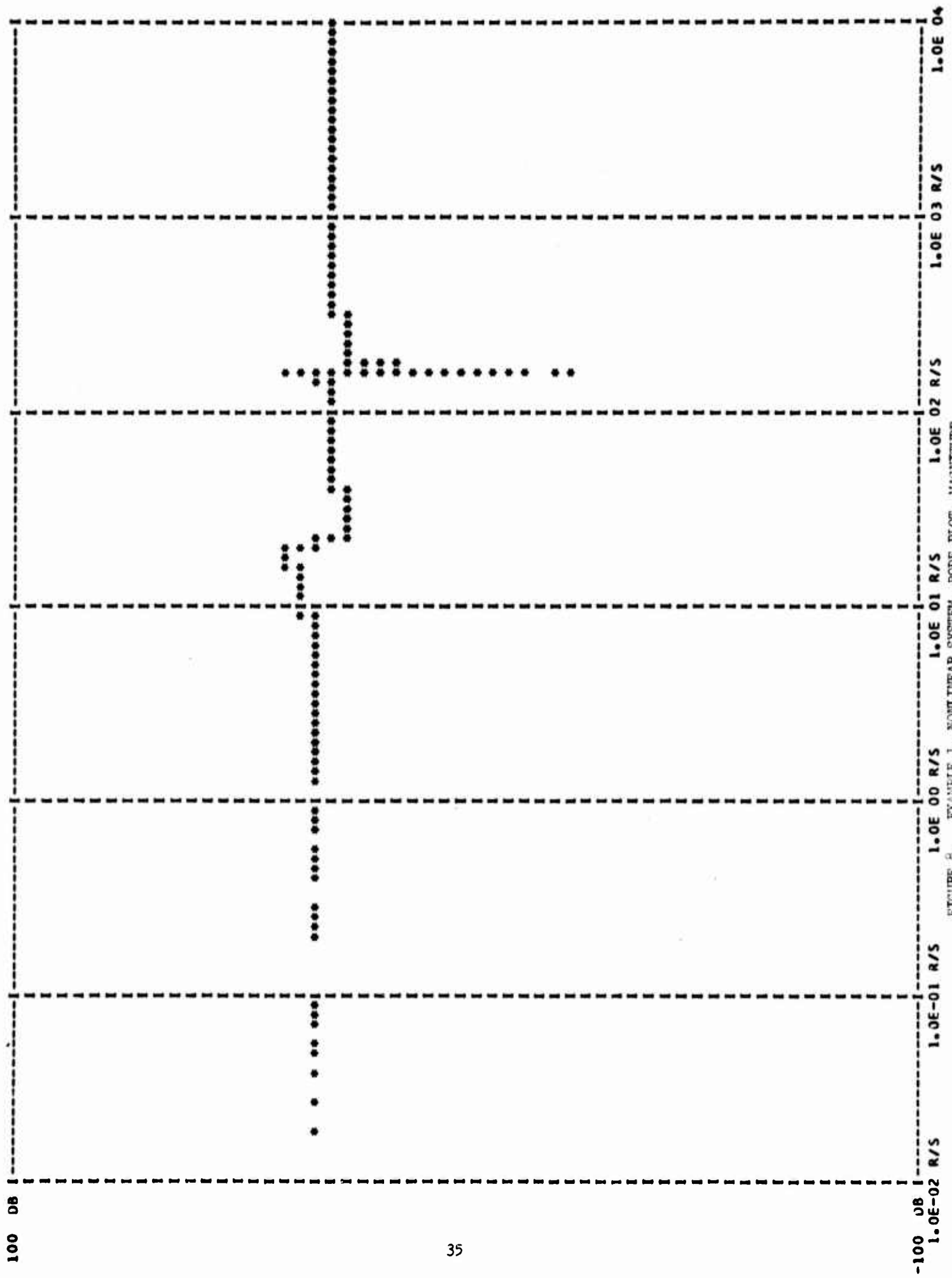


FIGURE 8. EXAMPLE 1, NONLINEAR SYSTEM, BODE PLOT, MAGNITUDE

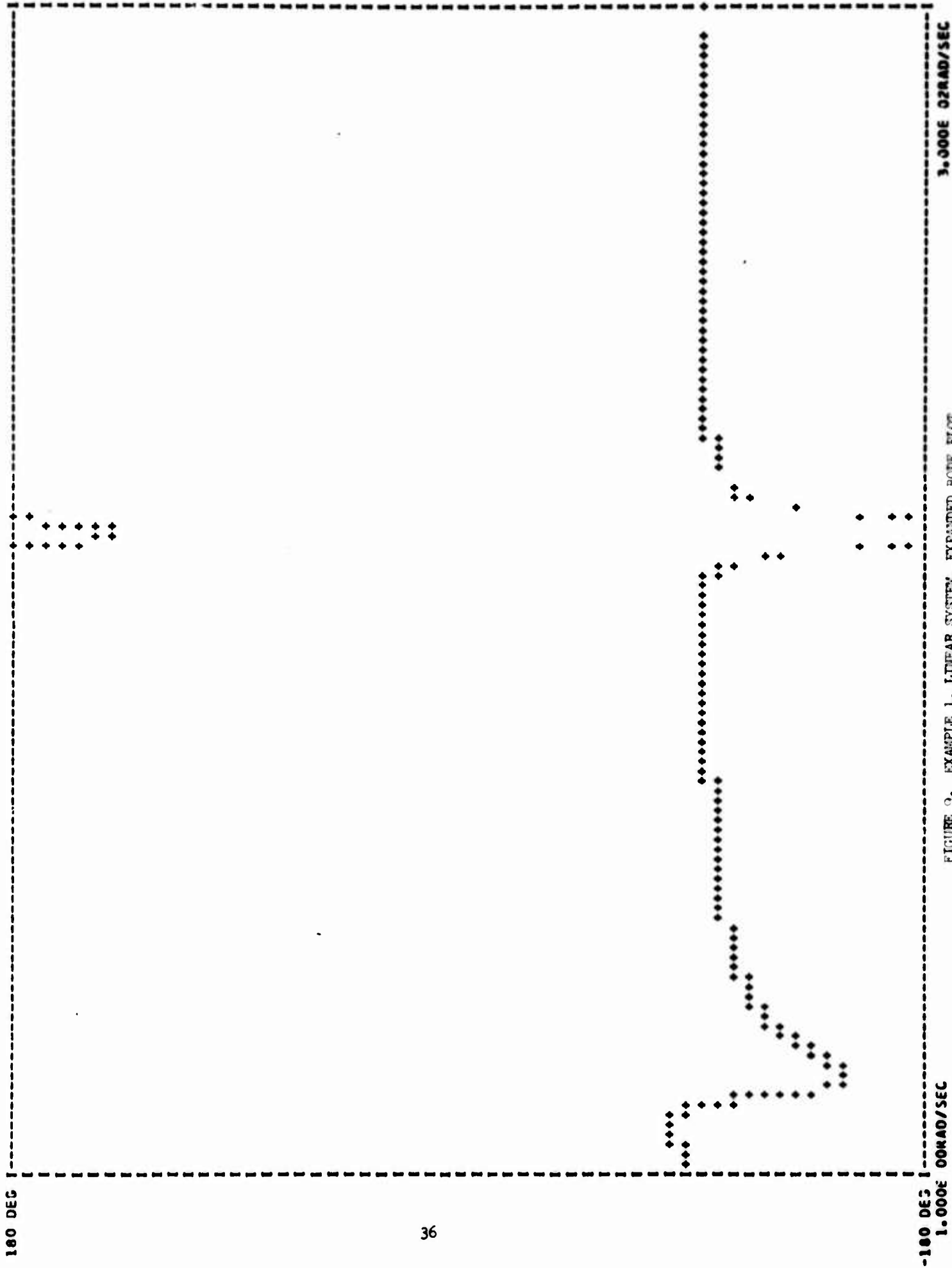


FIGURE 9. EXAMPLE 1, LINEAR SYSTEM, EXPANDED BODE PLOT
 (-180 to 180 DEG) PHASE ANGLE

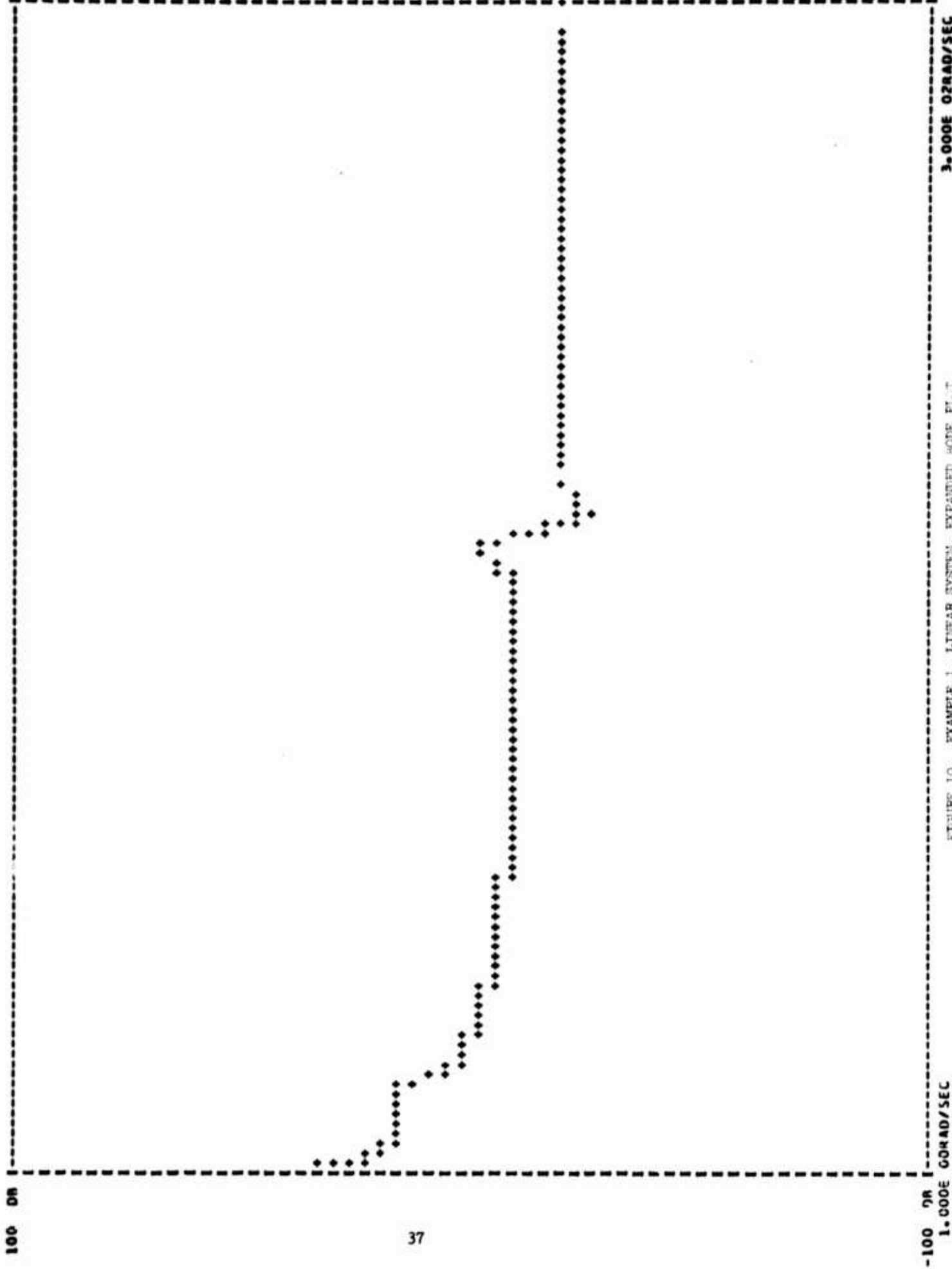


FIGURE 10. EXAMPLE 1, LINEAR SYSTEM, EXPANDED MODE PLOT
(1 TO 300 RAD/S) : GHIUUE

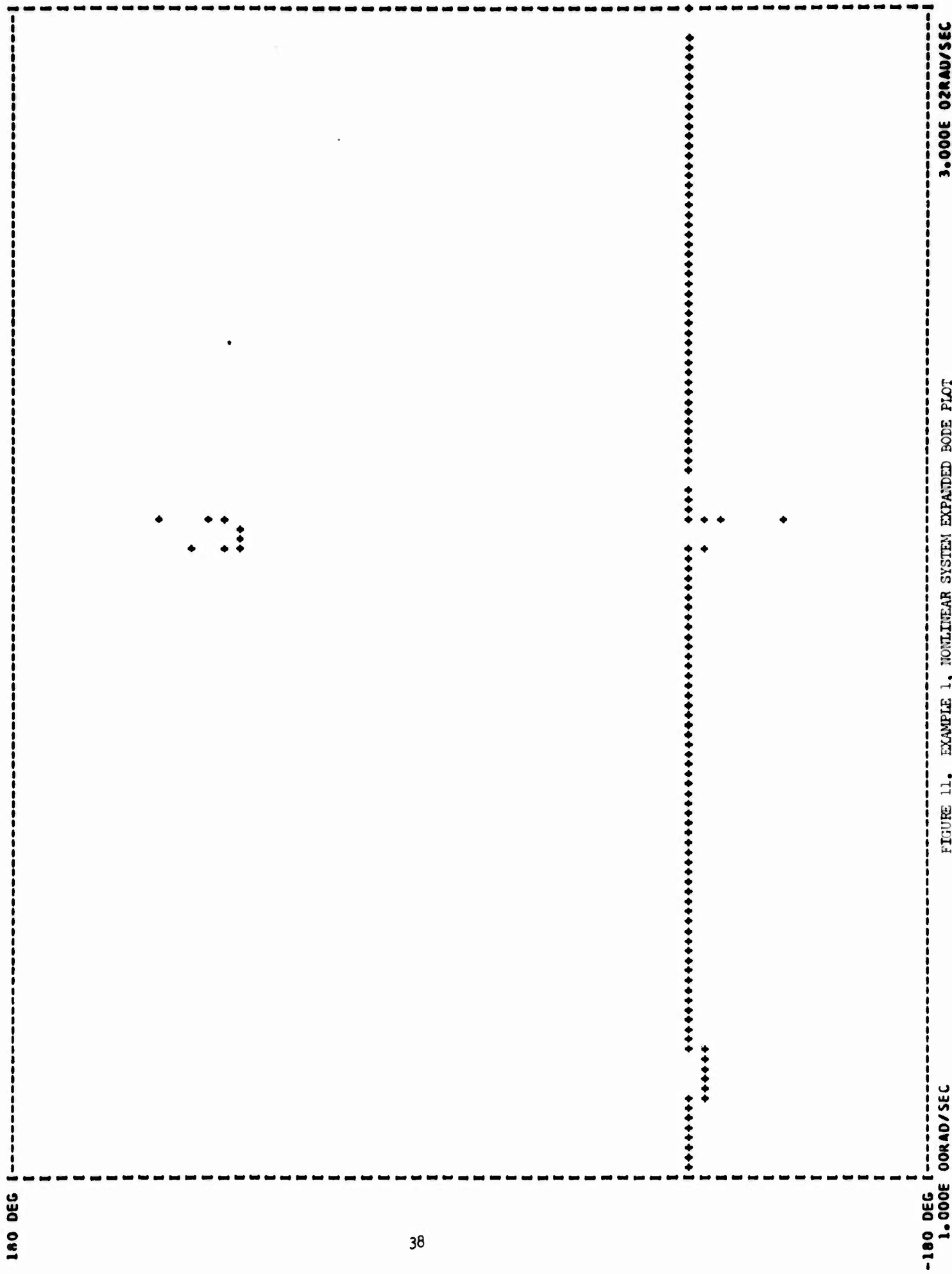


FIGURE 11. EXAMPLE 1, NONLINEAR SYSTEM EXPANDED BODE PLOT
(1 TO 300 RAD/S) PHASE ANGLE

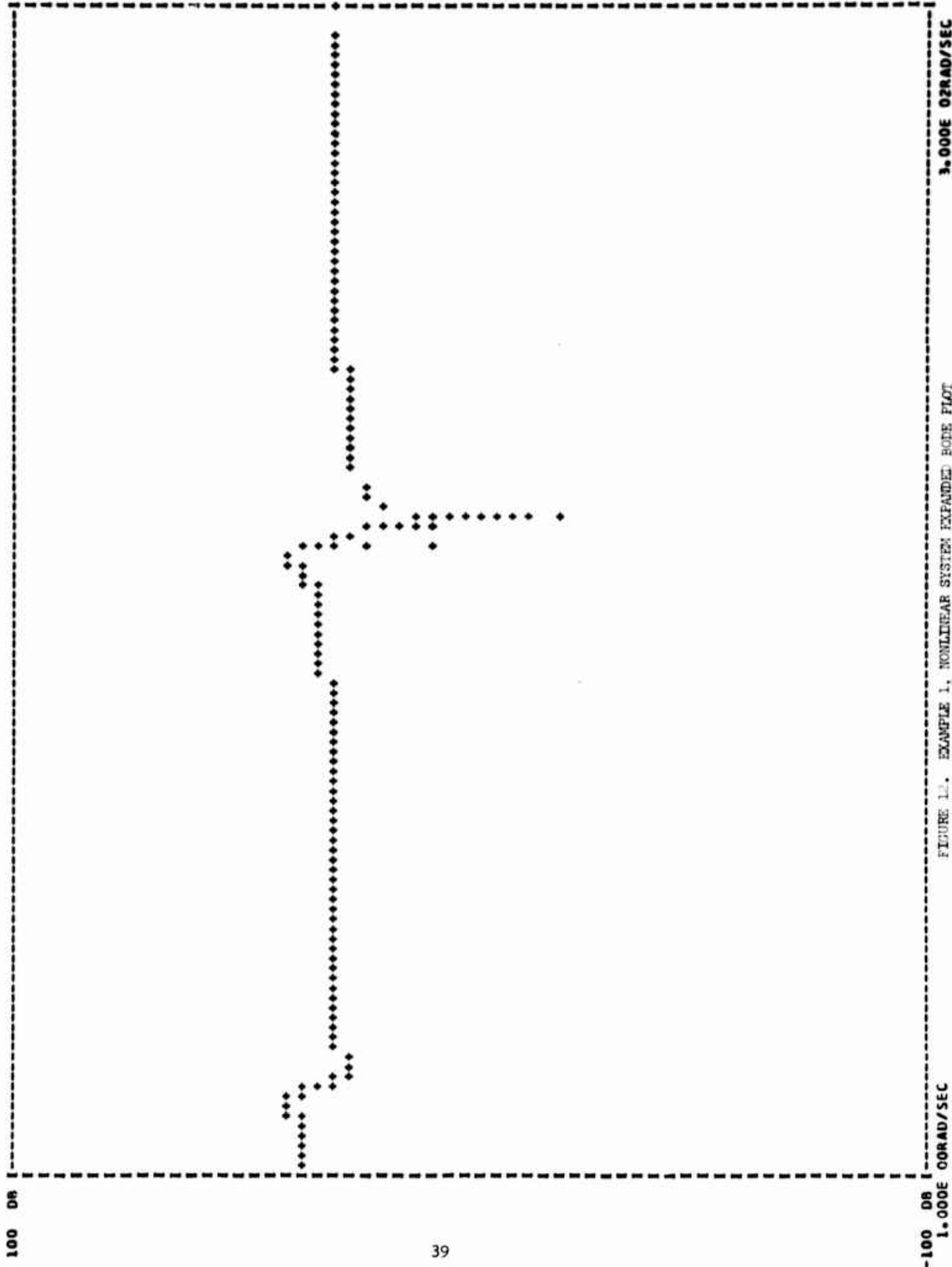


FIGURE 1.1. EXAMPLE 1.1. NONLINEAR SYSTEM EXPANDED BODE PLOT
(1 TO 300 RAD/S) MAGNITUDE

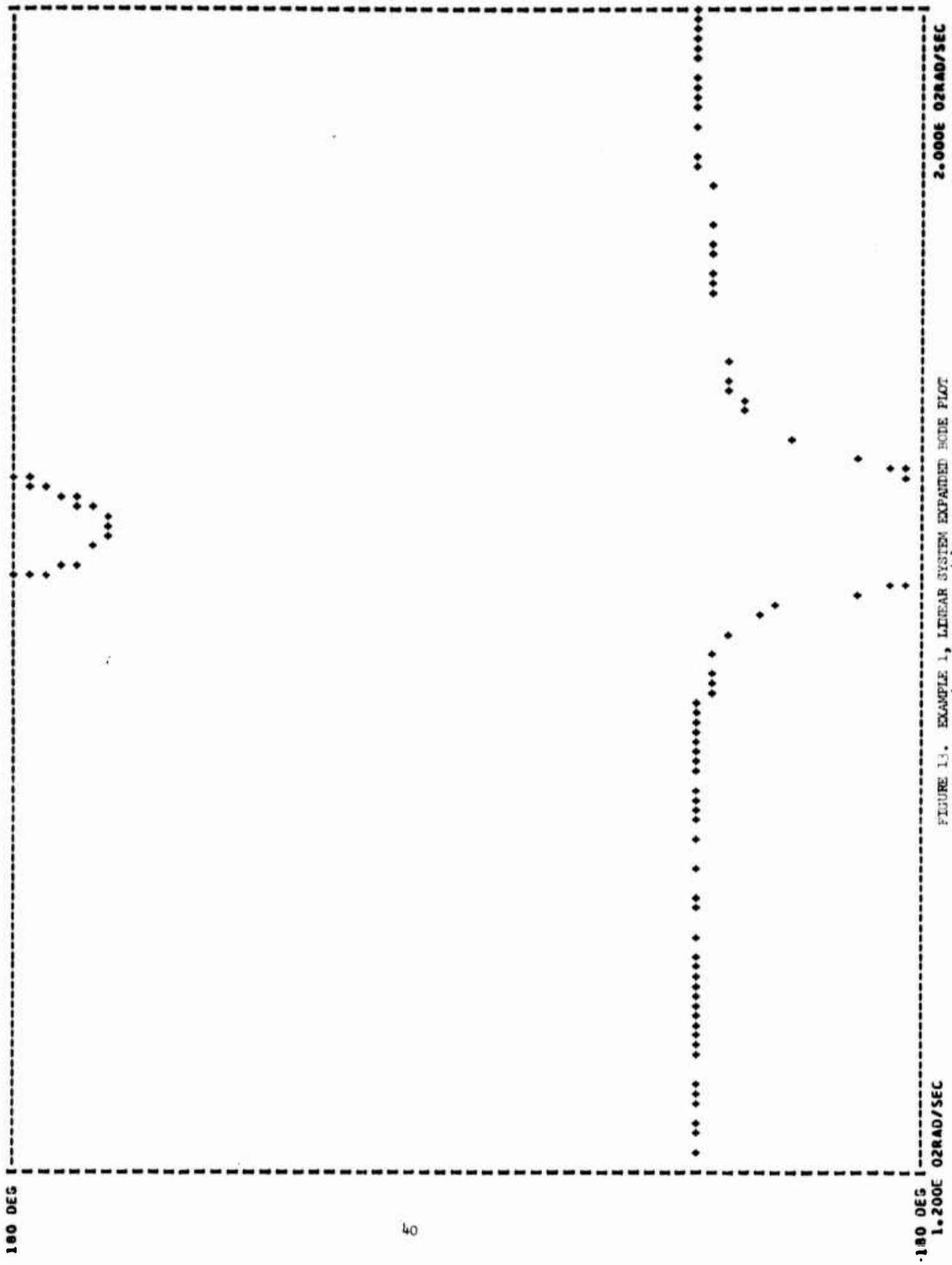


FIGURE 13. EXAMPLE 1, LINEAR SYSTEM EXPANDED BODE PLOT
(120 TO 200 RAD/S) PHASE ANGLE

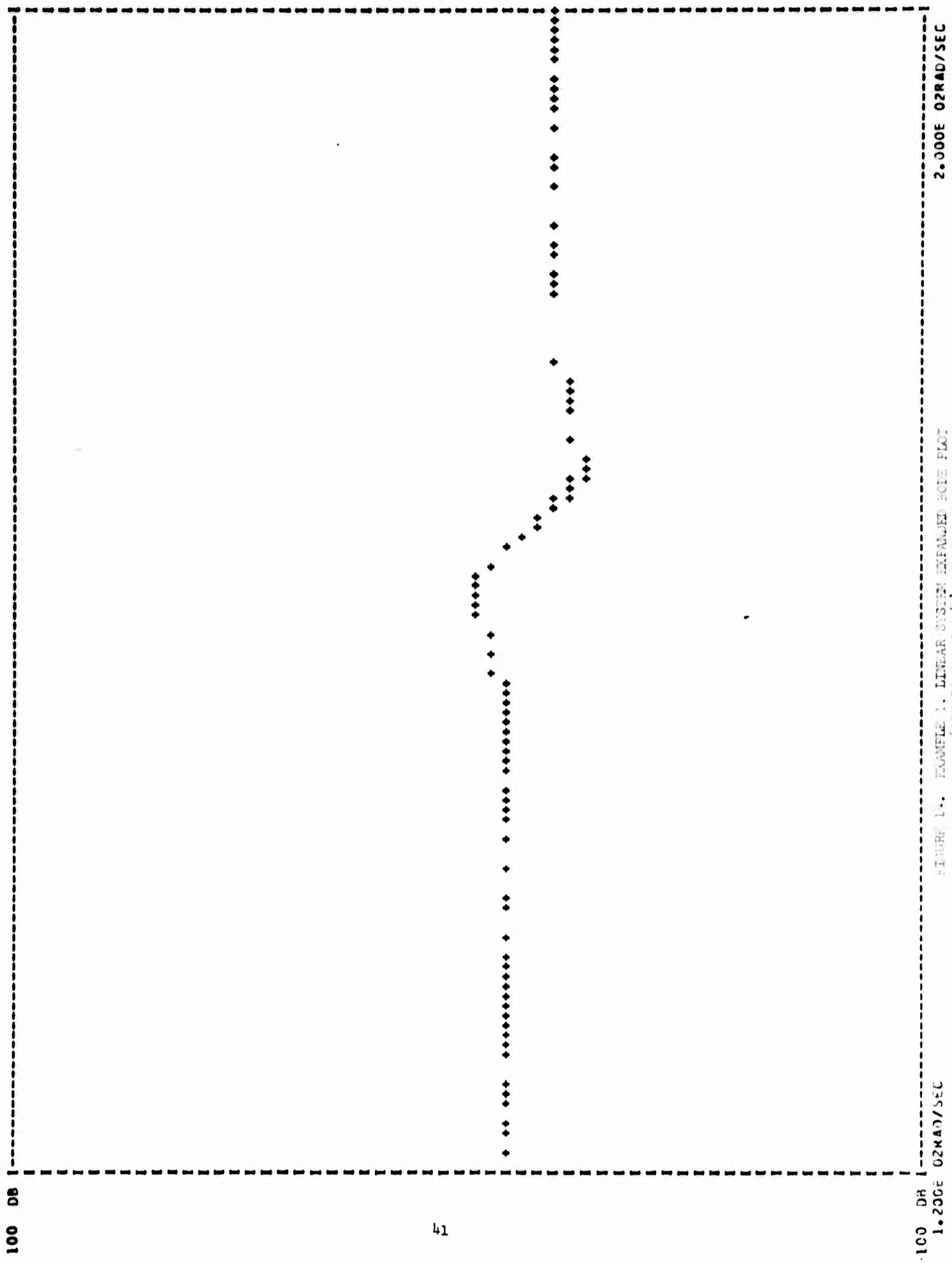
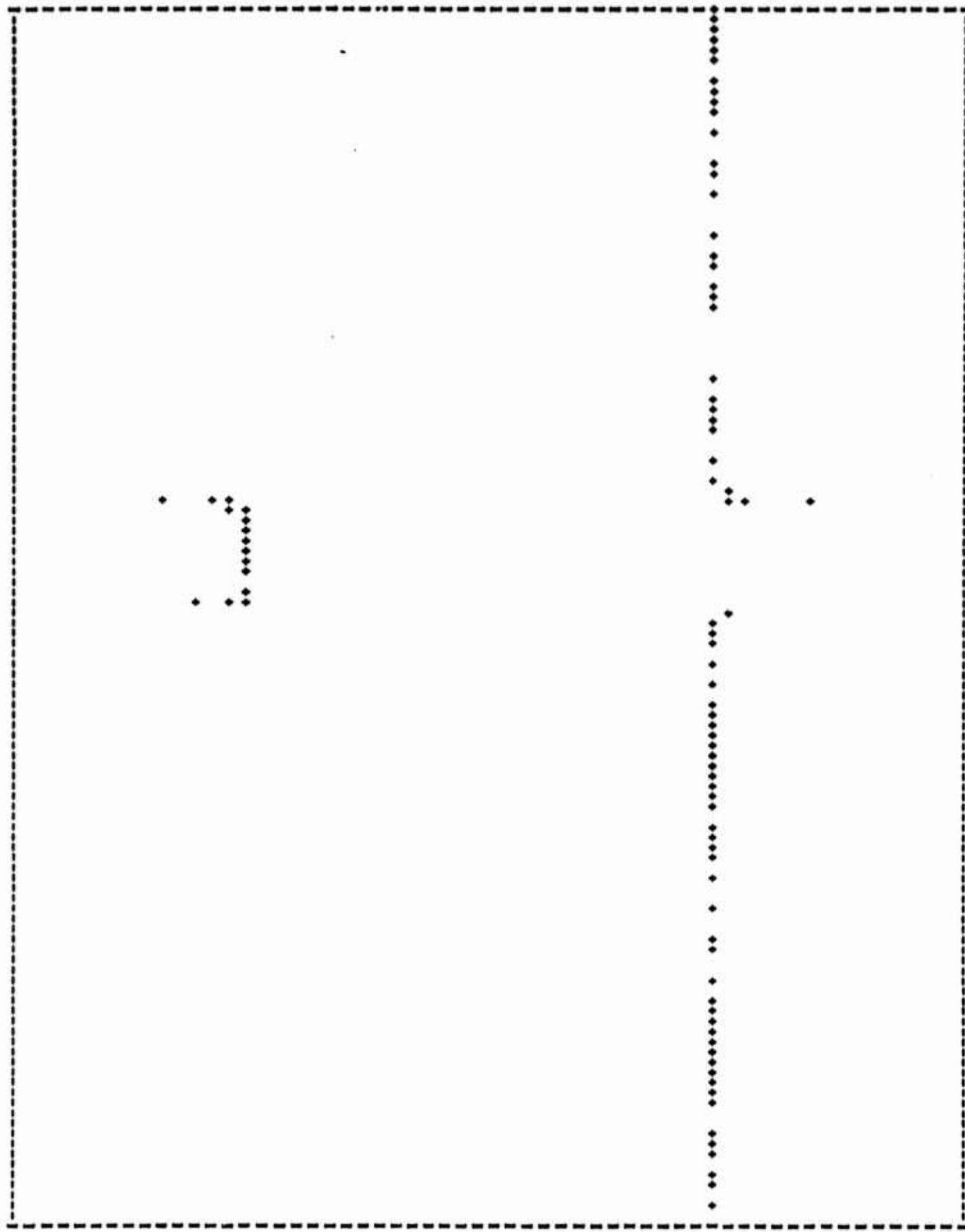


FIGURE 1. LINEAR SYSTEM IDENTIFIED BODE PLOT
(LEAD TO 100 RAD/S) MAGNITUDE

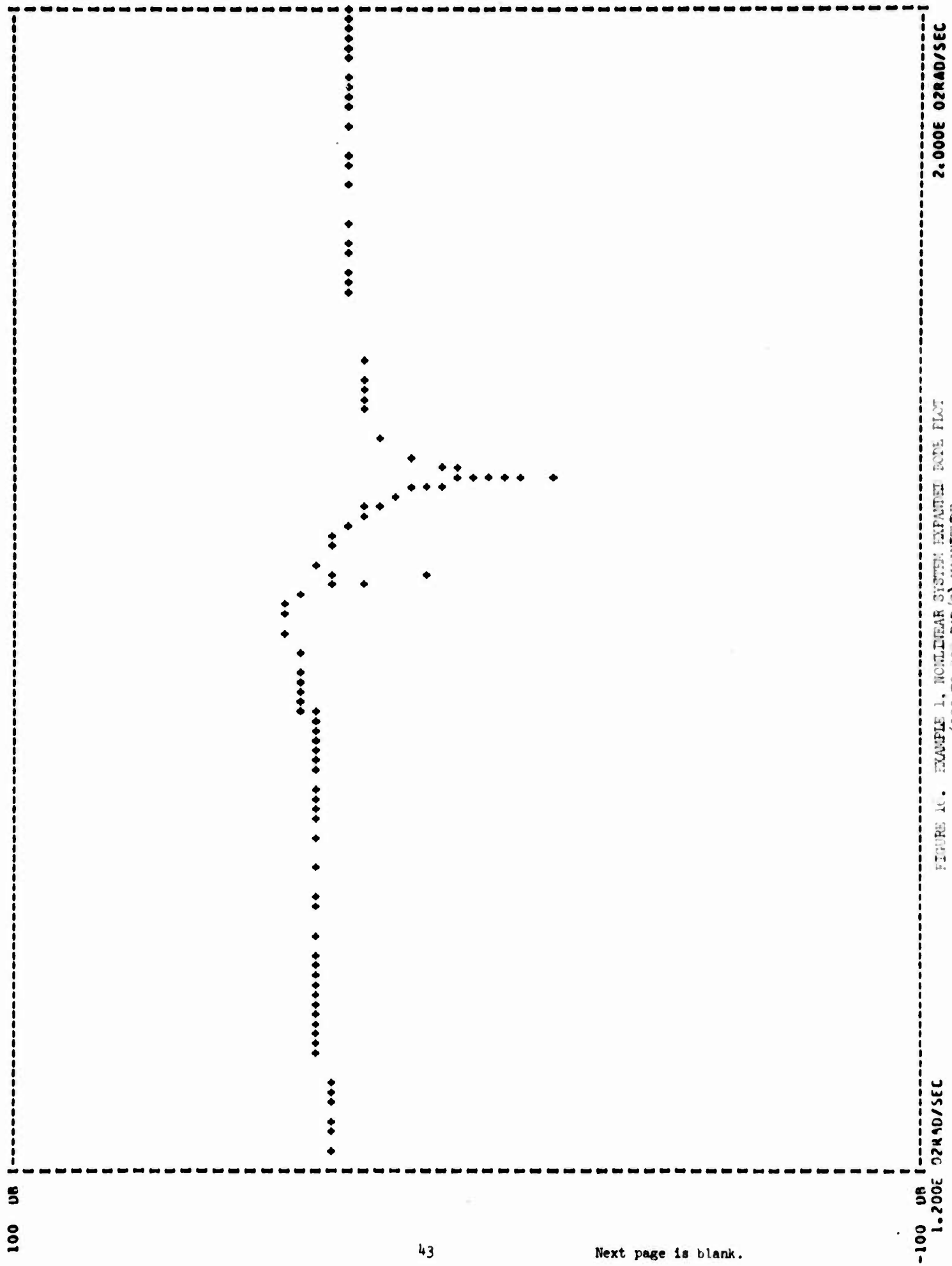
180 DEG



-180 DEG
1.200E 02RAD/SEC

2.000E 02RAD/SEC

FIGURE 15. EXAMPLE 1. NONLINEAR SYSTEM EXPANDED BODE PLOT
(120 TO 200 RAD/S) PHASE ANGLE



2.000E 02RAD/SEC

FIGURE 10. EXAMPLE 1. NONLINEAR SYSTEM EXPANDED BODE PLOT
(1.20 TO 200 RAD/S) MAGNITUDE

1.200E 02RAD/SEC

100 UR

TABLE 5. MIRROR OF INPUT, EXAMPLE 2.

POLYNOMIAL MULTIPLICATION AND ROOT LOCUS		PROBLEM NO. 37	
DELTA K =	0	2	1
K-INITIAL =	.00000000	.00000000	.00000000
NUMBER OF POLY. IN GROUP 1 OF NUMERATOR =	61	8	8
NUMBER OF POLY. IN GROUP 2 OF NUMERATOR =	9194	9	9
NUMBER OF POLY. IN GROUP 1 OF DENOMINATOR =	61	7	7
Ci 1) =	.0015000000	Ci 1) =	-1.0000000000
Ci 2) =	-.3981900000	Ci 2) =	.1500000000
Ci 3) =	61.1407000000	Ci 3) =	.0000000000
Ci 4) =	9194.1800000000	Ci 4) =	-.1754300000
Ci 5) =	45.7742000000	Ci 5) =	-12.0360000000
Ci 6) =	-2.6617000000	Ci 6) =	-13.2940000000
Ci 7) =	.0815440000	Ci 7) =	-.0247200000
Ci 8) =	.0285990000	Ci 8) =	-.0088740000
Ci 9) =	.0000673410	Ci 9) =	-.0000085040
Ci 10) =	.0000159750	Ci 10) =	-.0000009454
Ci 11) =	.0000000172	Ci 11) =	-.0000000001
Ci 12) =	.0000000026	Ci 12) =	.0000000002
Ci 13) =	.0000000000	Ci 13) =	.0000000000
Ci 14) =	.0000000000	Ci 14) =	.0000000000
Ci 15) =	.0000000000	Ci 15) =	.0000000000
Ci 1) =	119300.0000000000	Ci 1) =	.0000000000
Ci 2) =	37.5000000000	Ci 2) =	.0000000000
Ci 3) =	12.5000000000	Ci 3) =	.0000000000
Ci 4) =	35344.0000000000	Ci 4) =	1.0000000000
Ci 5) =	62.0000000000	Ci 5) =	207.3000000000
Ci 6) =	1.0000000000	Ci 6) =	1.0000000000
Ci 7) =	1.0000000000	Ci 7) =	37.5000000000
Ci 8) =	2.0000000000	Ci 8) =	1.0000000000
Ci 9) =	180000.0000000000	Ci 9) =	3.0000000000
Ci 10) =	4000.0000000000	Ci 10) =	180000.0000000000
Ci 11) =	3600.0000000000	Ci 11) =	4000.0000000000
Ci 12) =	30.0000000000	Ci 12) =	3600.0000000000
Ci 13) =		Ci 13) =	30.0000000000
Ci 14) =		Ci 14) =	
Ci 15) =		Ci 15) =	
Ci 1) =		Ci 1) =	119300.0000000000
Ci 2) =		Ci 2) =	207.3000000000
Ci 3) =		Ci 3) =	1.0000000000
Ci 4) =		Ci 4) =	37.5000000000
Ci 5) =		Ci 5) =	3.0000000000
Ci 6) =		Ci 6) =	62.0000000000
Ci 7) =		Ci 7) =	1.0000000000
Ci 8) =		Ci 8) =	1.0000000000
Ci 9) =		Ci 9) =	180000.0000000000
Ci 10) =		Ci 10) =	4000.0000000000
Ci 11) =		Ci 11) =	3600.0000000000
Ci 12) =		Ci 12) =	30.0000000000
Ci 13) =		Ci 13) =	
Ci 14) =		Ci 14) =	
Ci 15) =		Ci 15) =	
Ci 1) =		Ci 1) =	119300.0000000000
Ci 2) =		Ci 2) =	207.3000000000
Ci 3) =		Ci 3) =	1.0000000000
Ci 4) =		Ci 4) =	37.5000000000
Ci 5) =		Ci 5) =	3.0000000000
Ci 6) =		Ci 6) =	62.0000000000
Ci 7) =		Ci 7) =	1.0000000000
Ci 8) =		Ci 8) =	1.0000000000
Ci 9) =		Ci 9) =	180000.0000000000
Ci 10) =		Ci 10) =	4000.0000000000
Ci 11) =		Ci 11) =	3600.0000000000
Ci 12) =		Ci 12) =	30.0000000000
Ci 13) =		Ci 13) =	
Ci 14) =		Ci 14) =	
Ci 15) =		Ci 15) =	
Ci 1) =		Ci 1) =	119300.0000000000
Ci 2) =		Ci 2) =	207.3000000000
Ci 3) =		Ci 3) =	1.0000000000
Ci 4) =		Ci 4) =	37.5000000000
Ci 5) =		Ci 5) =	3.0000000000
Ci 6) =		Ci 6) =	62.0000000000
Ci 7) =		Ci 7) =	1.0000000000
Ci 8) =		Ci 8) =	1.0000000000
Ci 9) =		Ci 9) =	180000.0000000000
Ci 10) =		Ci 10) =	4000.0000000000
Ci 11) =		Ci 11) =	3600.0000000000
Ci 12) =		Ci 12) =	30.0000000000
Ci 13) =		Ci 13) =	
Ci 14) =		Ci 14) =	
Ci 15) =		Ci 15) =	
Ci 1) =		Ci 1) =	119300.0000000000
Ci 2) =		Ci 2) =	207.3000000000
Ci 3) =		Ci 3) =	1.0000000000
Ci 4) =		Ci 4) =	37.5000000000
Ci 5) =		Ci 5) =	3.0000000000
Ci 6) =		Ci 6) =	62.0000000000
Ci 7) =		Ci 7) =	1.0000000000
Ci 8) =		Ci 8) =	1.0000000000
Ci 9) =		Ci 9) =	180000.0000000000
Ci 10) =		Ci 10) =	4000.0000000000
Ci 11) =		Ci 11) =	3600.0000000000
Ci 12) =		Ci 12) =	30.0000000000
Ci 13) =		Ci 13) =	
Ci 14) =		Ci 14) =	
Ci 15) =		Ci 15) =	

TABLE 6. EQUIVALENT SYSTEM OPEN LOOP POLYNOMIALS AND ROOTS, EXAMPLE 2.

COEFFICIENTS ARE GIVEN IN ASCENDING ORDER

THE COEFFICIENTS OF POLYNOMIAL A (ORDER = 20)					
-6.1198957E 16	9.2521124E 18	1.4379605E 21	3.5421717E 21	1.2905790E 21	5.9459794E 19
9.0365506E 17	3.3735915E 16	3.9371677E 15	1.1777247E 14	2.9117079E 12	6.2795230E 10
8.1833541E 08	1.1560946E 07	1.0646439E 05	9.3728004E 02	6.6668516E 00	3.3289968E-02
1.8402498E-04	4.2351181E-07	1.7475004E-09			
THE ROOTS OF A					
-1.0694428E-02	+1 0.000000E 00	-4.8391511E-01	+1 1.0342048E-14	4.0455430E-03	+1 0.000000E 00
-2.6281354E 06	+1 0.000000E 00	-1.9524174E 01	+1 -1.2192916E 01	-1.9524174E 01	+1 1.2192916E 01
1.8019734E 01	+1 -1.8023020E 01	1.8019734E 01	+1 1.8023020E 01	-4.4083790E-01	+1 5.8588405E 01
-4.4083790E-01	+1 -5.8588405E 01	-1.0455319E 01	+1 1.0028177E 02	-1.0455319E 01	+1 -1.0028177E 02
-4.500000E 01	+1 0.000000E 00	9.5034570E 00	+1 9.8992955E 01	9.5034570E 00	+1 -9.8992955E 01
-3.2752621E 01	+1 1.9552017E 02	-3.2752621E 01	+1 -1.8552017E 02	-1.2000000E 02	+1 0.000000E 00
-1.4673431E 00	+1 1.6720316E 02	-1.4673431E 00	+1 -1.6720316E 02		
THE COEFFICIENTS OF POLYNOMIAL B (ORDER = 24)					
4.4350635E 18	2.7125958E 19	6.4489032E 20	1.5105939E 21	7.3257871E 20	3.1217410E 20
4.6274362E 19	3.1607955E 18	1.2877217E 17	3.99226549E 15	1.0008113E 14	1.9949714E 12
3.4274584E 10	4.7956796E 08	5.9506779E 06	5.8881896E 04	5.5416836E 02	3.8551691E 00
2.8002674E-02	1.3359390E-04	7.3606134E-07	2.2365276E-09	8.7580072E-12	1.3306326E-14
3.0588488E-17					
THE ROOTS OF B					
-5.000000E-01	+1 0.000000E 00	-1.3437380E-02	+1 8.4953454E-02	-1.3437380E-02	+1 -8.4953454E-02
-8.5697546E-01	+1 2.2620353E 00	-8.5697546E-01	+1 -2.2620353E 00	-2.000000E 01	+1 -8.0754456E-14
-1.250000E 01	+1 -8.7707630E-14	-4.6644787E-01	+1 5.1981022E 01	-4.6644787E-01	+1 -5.1981022E 01
-4.3478261E 01	+1 4.4974263E 01	-4.3478261E 01	+1 -4.4974263E 01	-3.000000E 01	+1 0.000000E 00
-6.0901678E-01	+1 6.0974886E 01	-6.0901678E-01	+1 -6.0974886E 01	-1.5634793E 00	+1 1.1108566E 02
-1.5634793E 00	+1 -1.1108566E 02	-2.6156110E 00	+1 1.3130735E 02	-2.6156110E 00	+1 -1.3130735E 02
-1.9225814E 00	+1 1.5928538E 02	-1.9225814E 00	+1 -1.5928538E 02	-3.140000E 01	+1 1.8535922E 02
-3.140000E 01	+1 -1.8535922E 02	-1.0365000E 02	+1 3.2947940E 02	-1.0365000E 02	+1 -3.2947940E 02

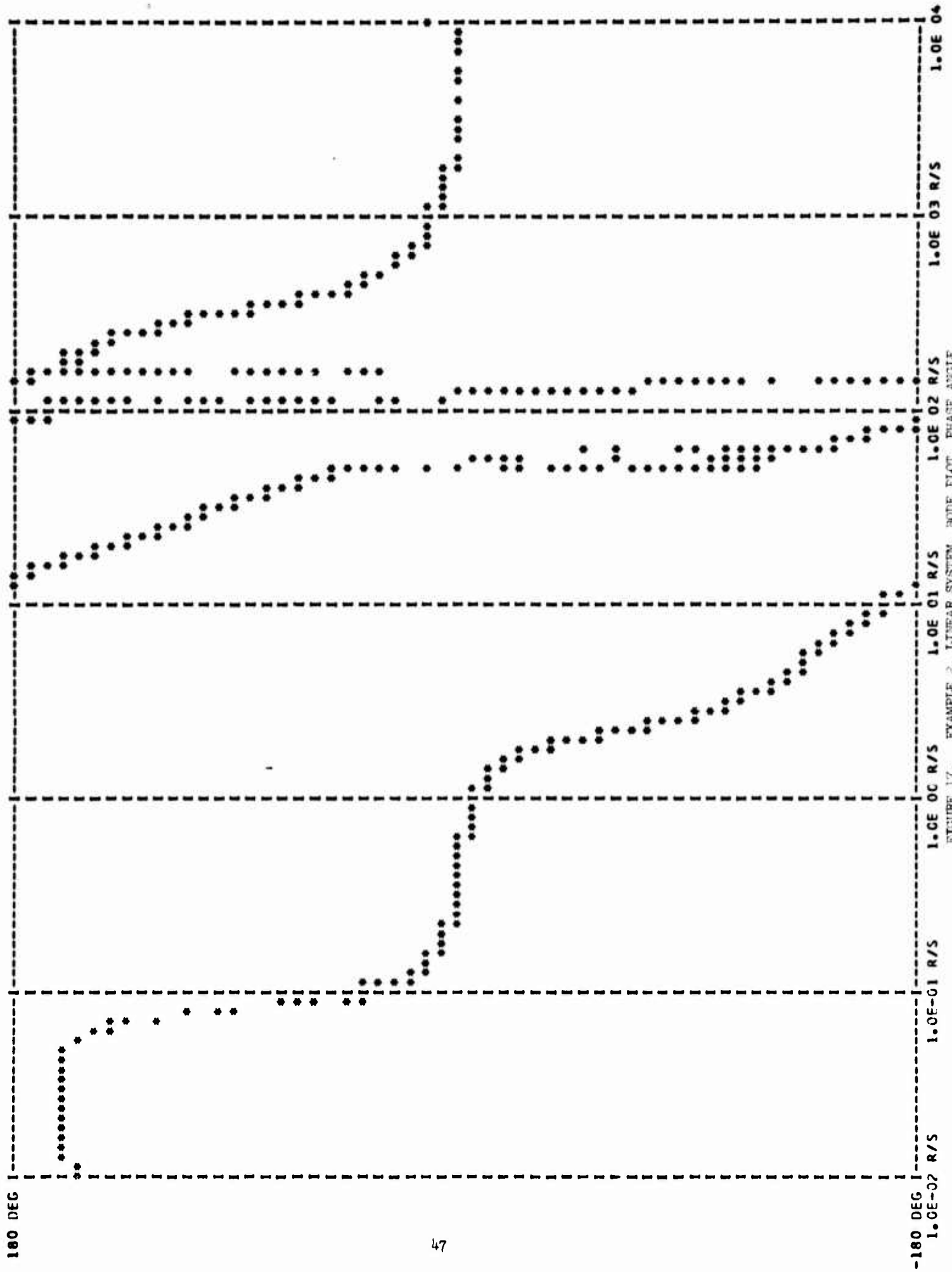


FIGURE 17. EXAMPLE 2. LINEAR SYSTEM, NODE PLOT, PHASE ANGLE

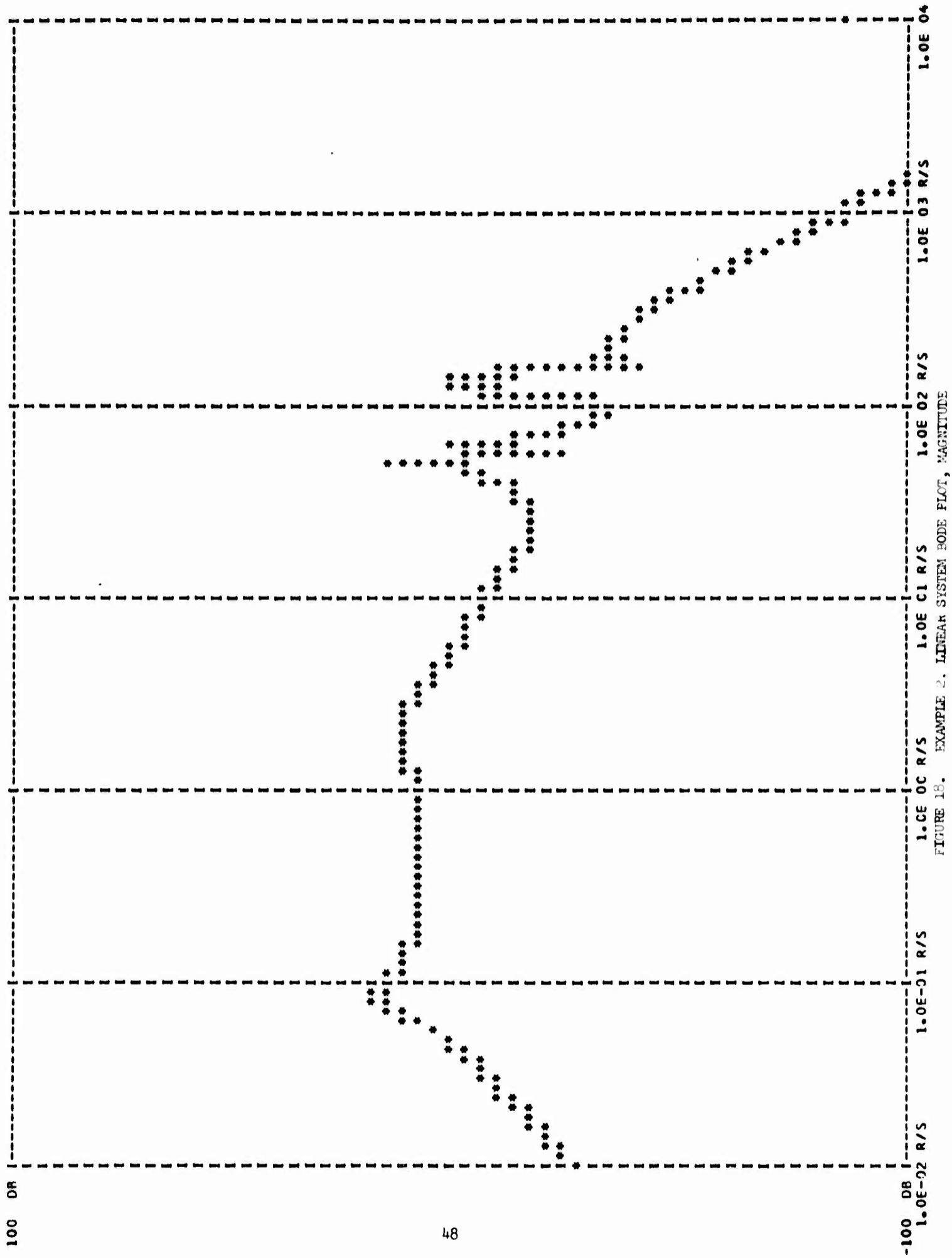


FIGURE 18. EXAMPLE 2. LINEAR SYSTEM BODE PLOT, MAGNITUDE

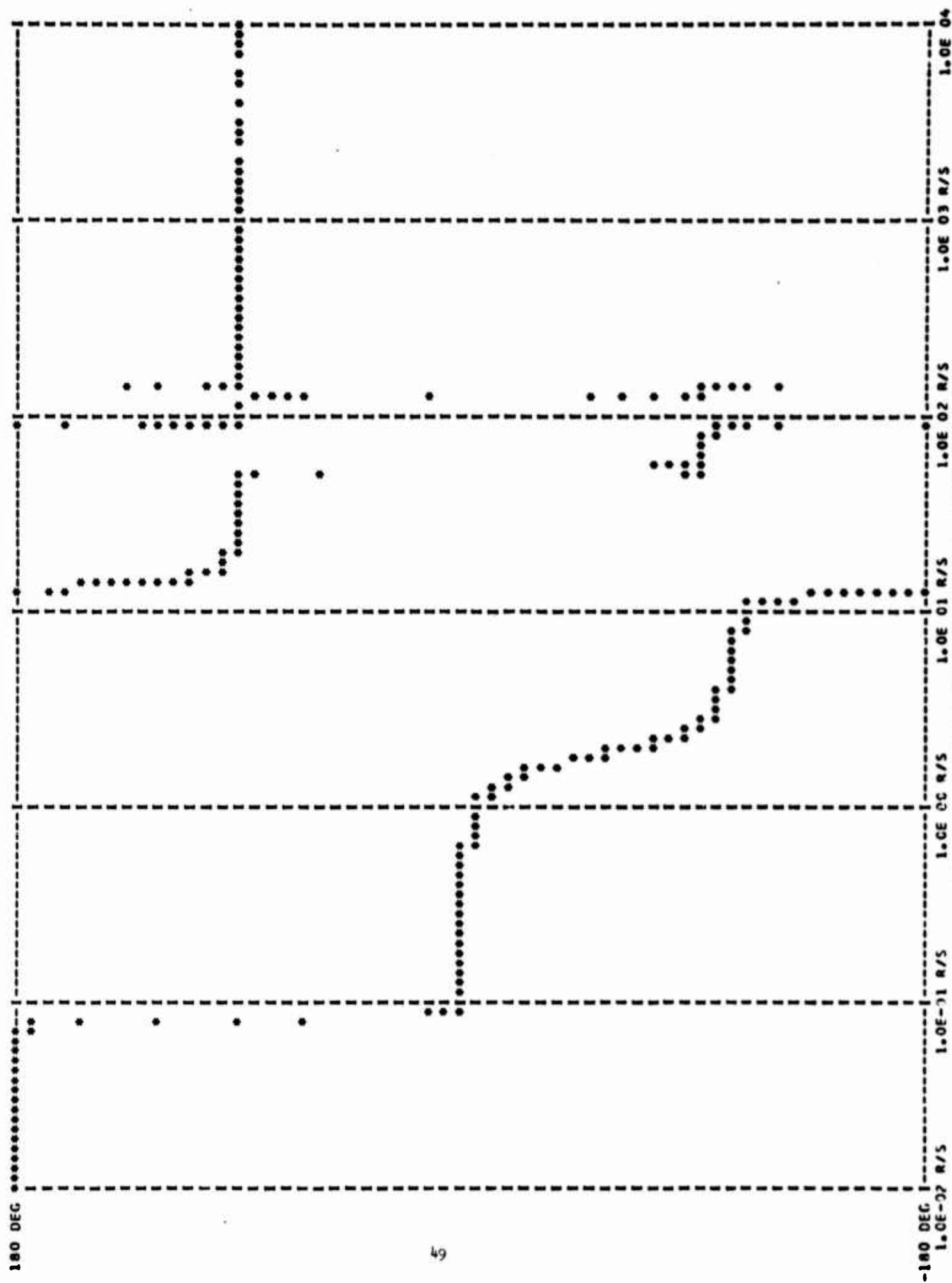


FIGURE 19. EXAMPLE 2. NONLINEAR SYSTEM, BODE PLOT, PHASE ANGLE

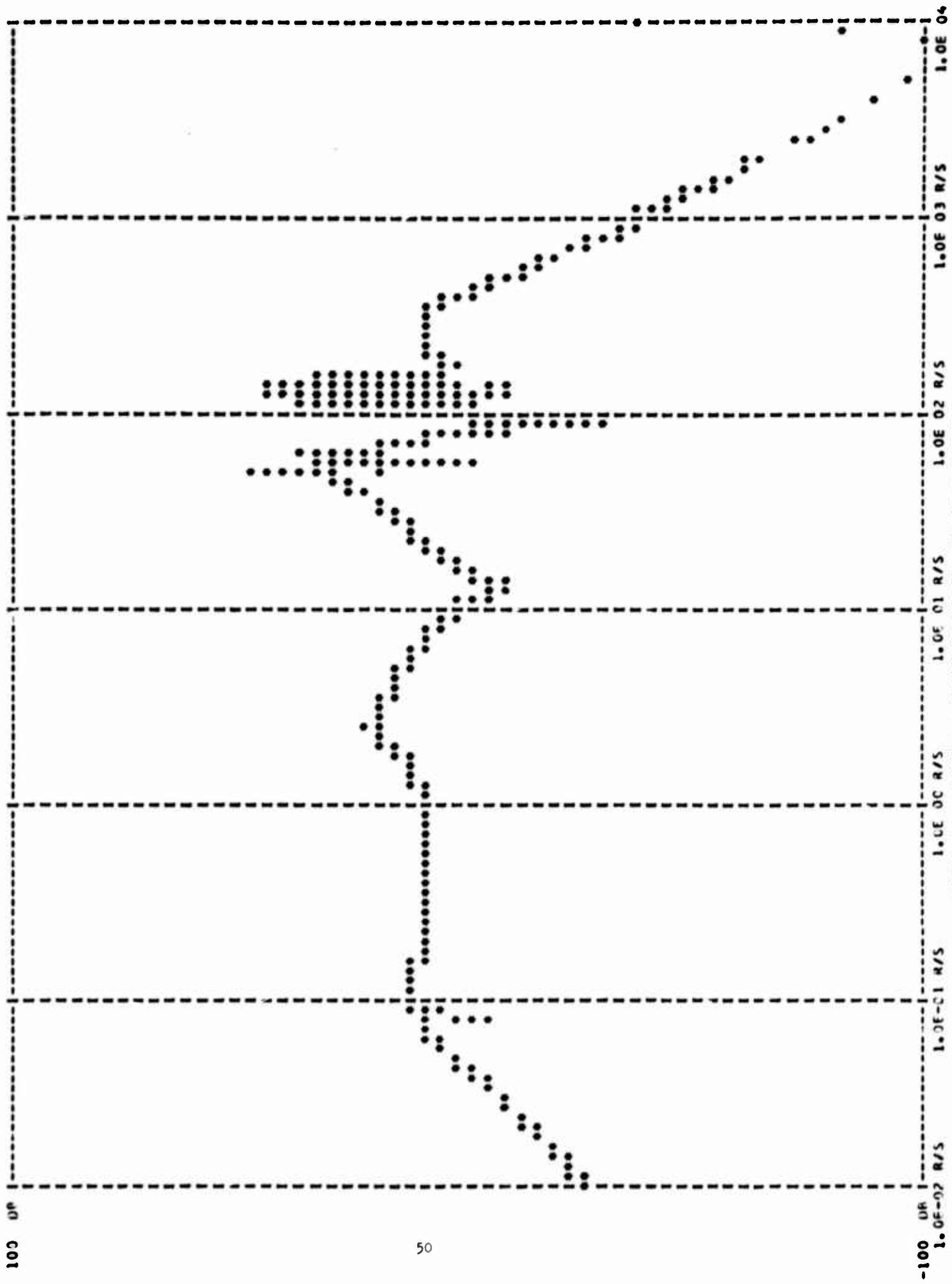


FIGURE NO. EXAMPLE 1. NONLINEAR SYSTEM, NOISE PLT, MAGNITUDE

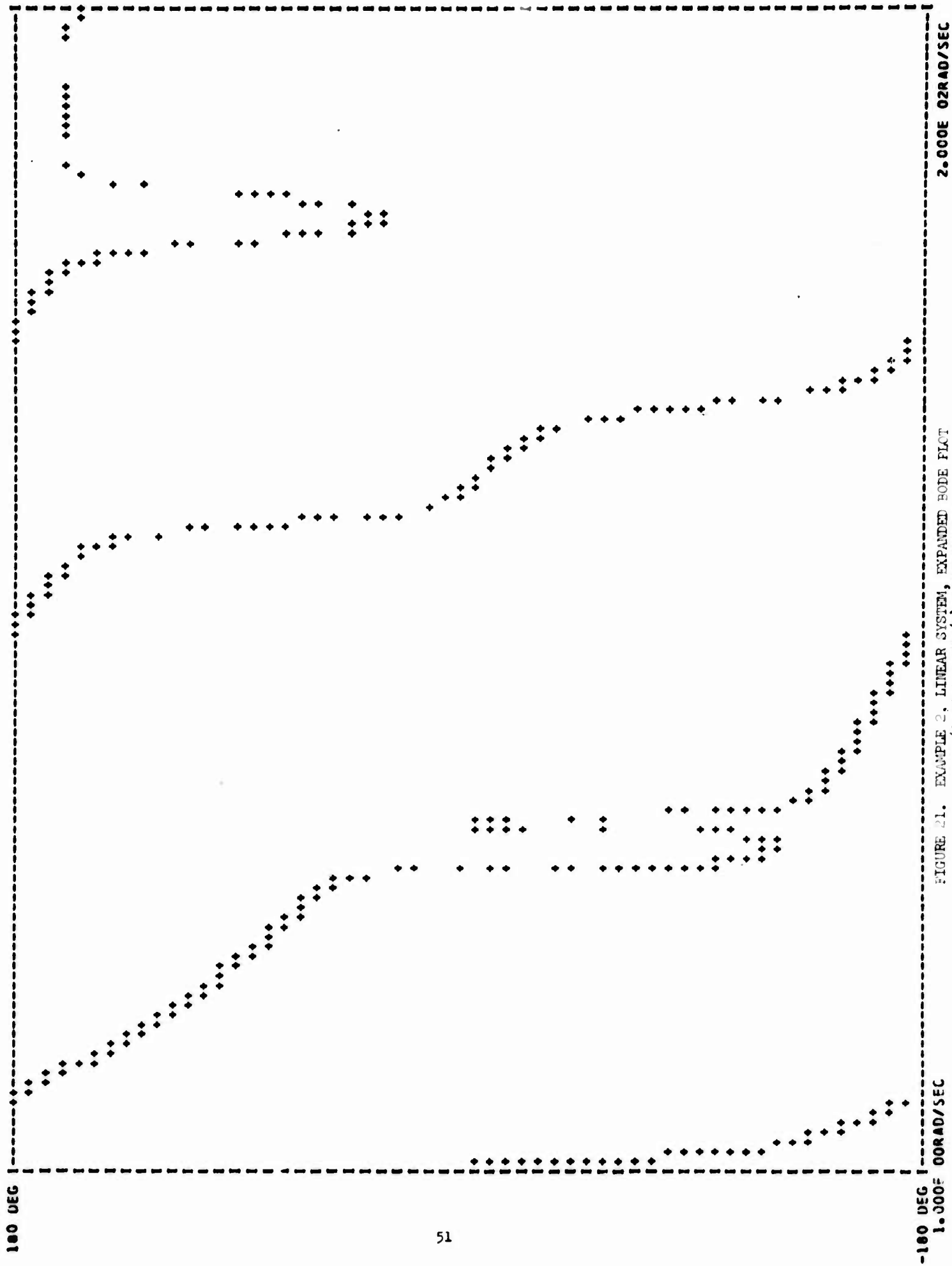
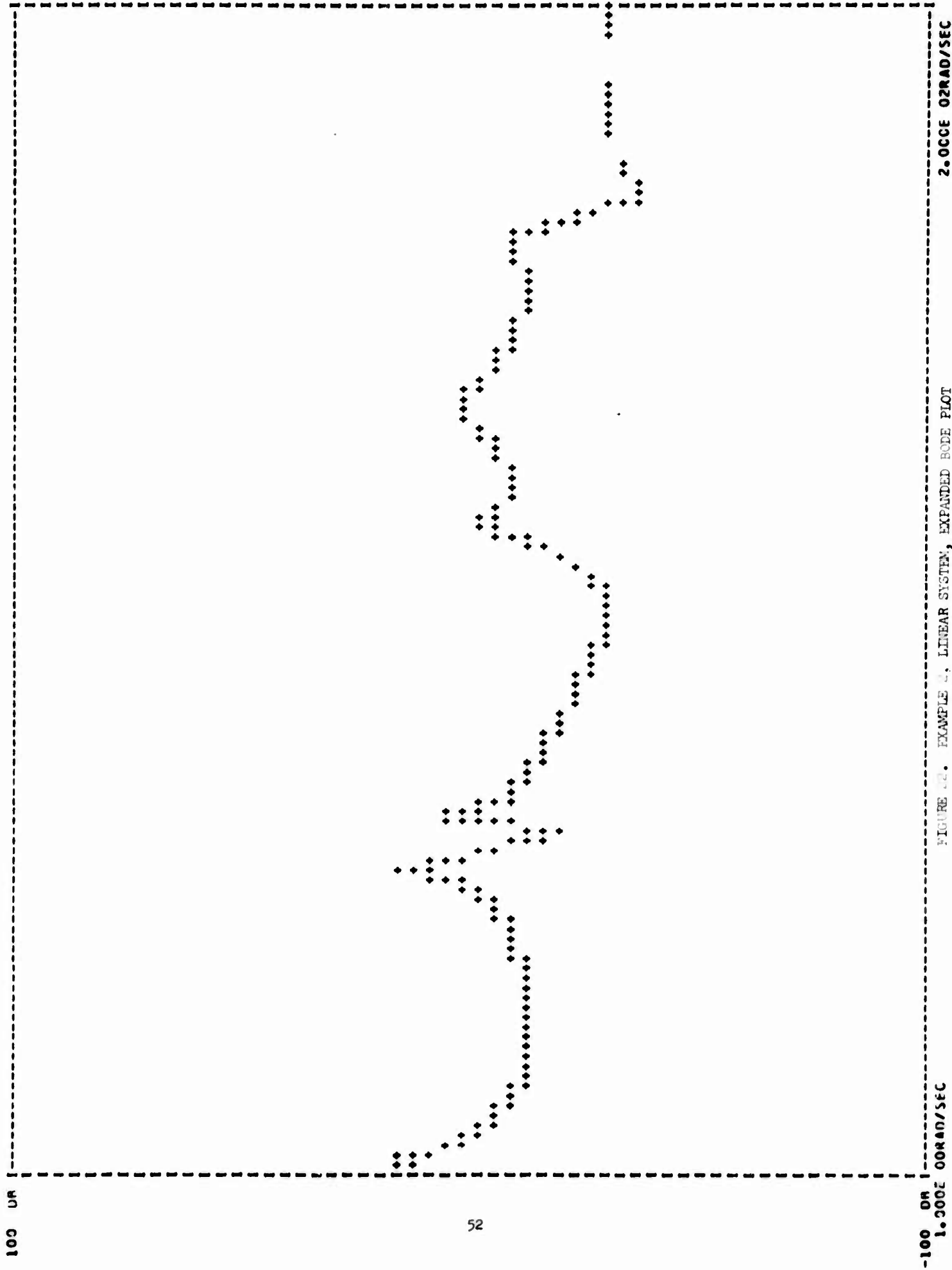


FIGURE 21. EXAMPLE 2, LINEAR SYSTEM, EXPANDED BODE PLOT
(1 TO 200 RAD/S) PHASE ANGLE



2.000E 02 RAD/SEC

FIGURE 2. EXAMPLE 2; LINEAR SYSTEM, EXPANDED BODE PLOT
(1 TO 200 RAD/S) MAGNITUDE

1.000E 00 RAD/SEC

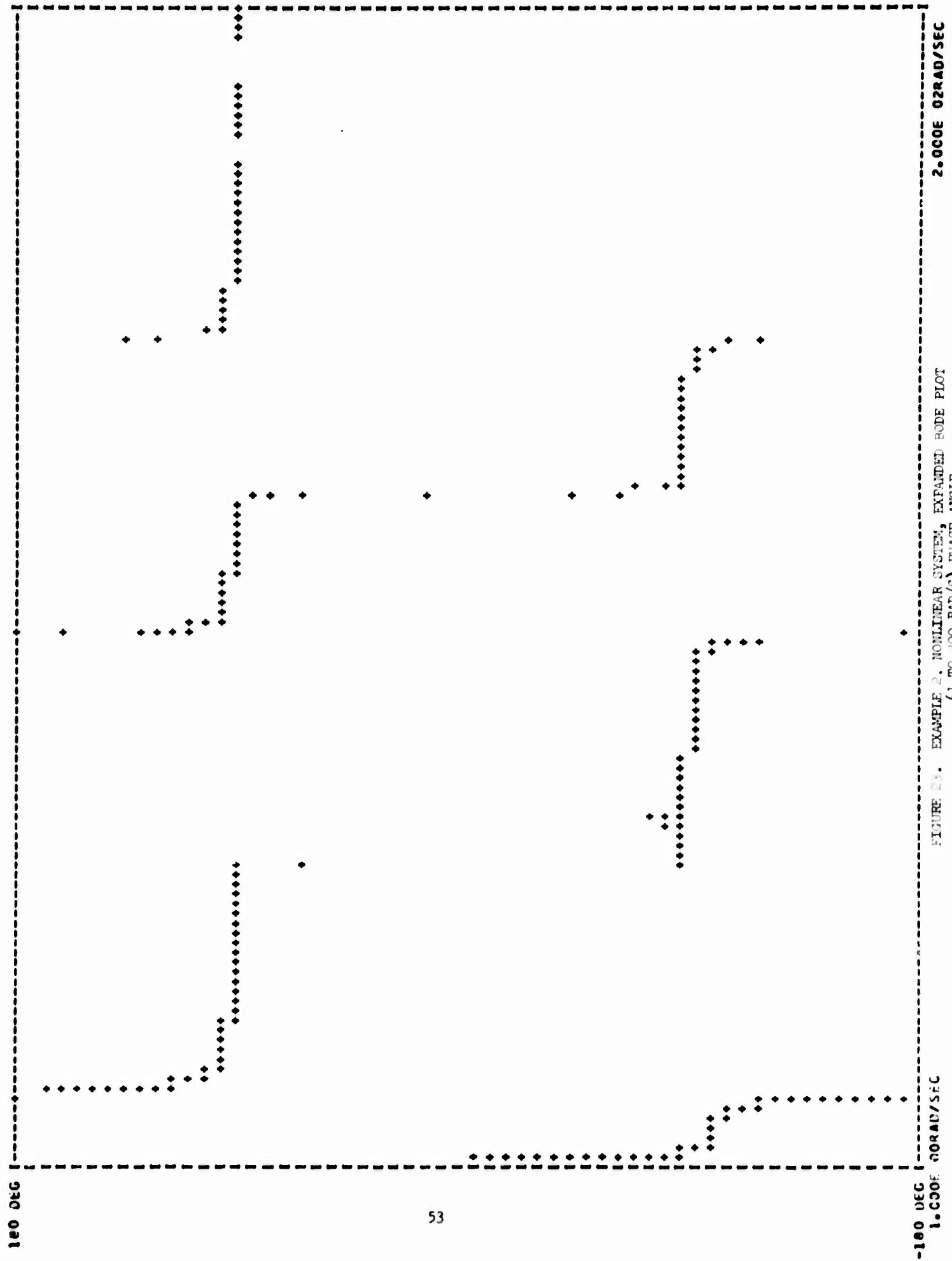


FIGURE 2. EXAMPLE 2. NONLINEAR SYSTEM, EXPANDED BODE PLOT
(1 TO 100 RAD/S) PHASE ANGLE

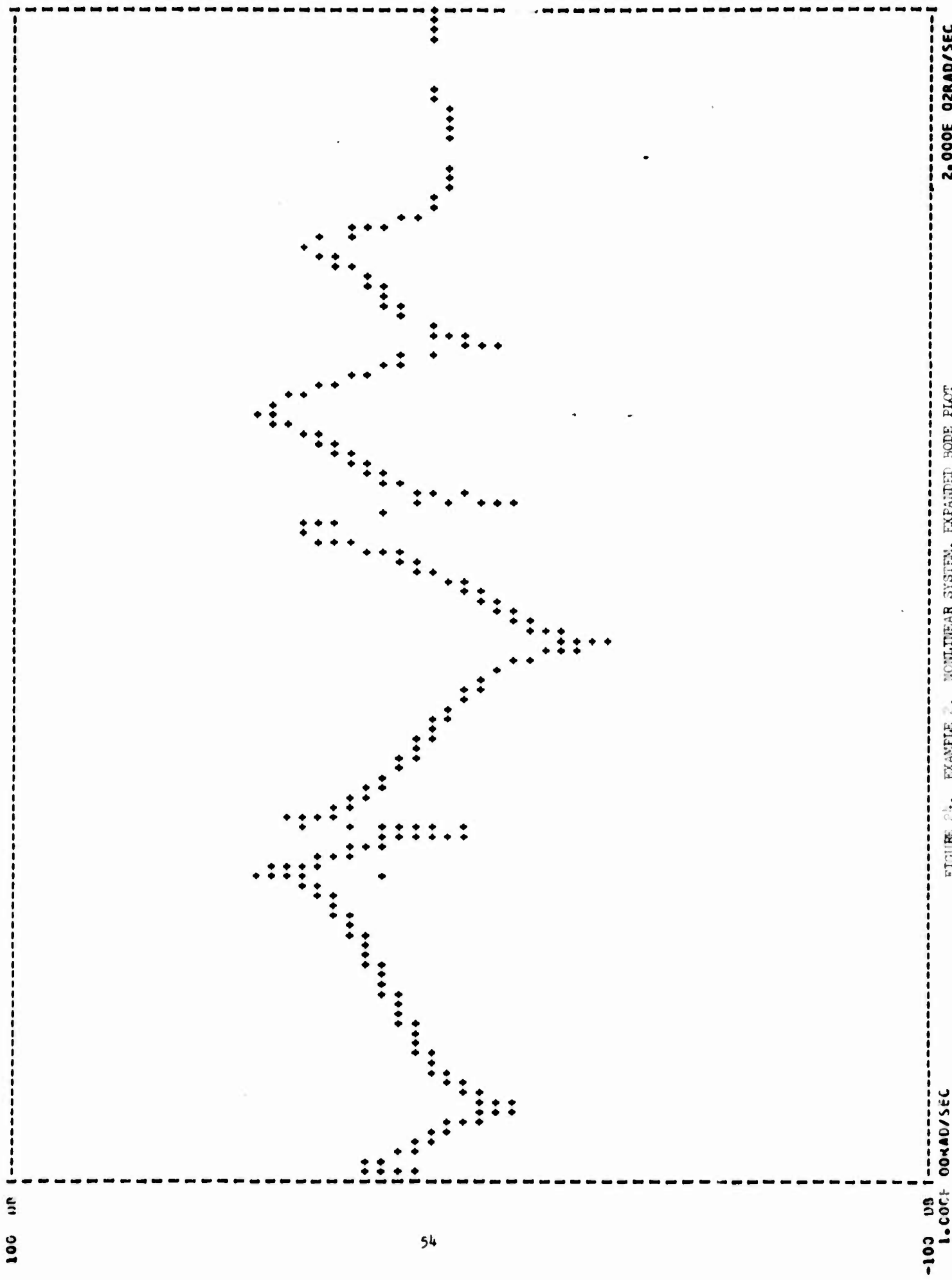


FIGURE 24. EXAMPLE 2. NONLINEAR SYSTEM, EXPANDED BODE PLOT
(1 TO 200 RAD/S) MAGNITUDE

REFERENCES

1. Burke, Harold H.; Payne, Robert L. Jr.; A Linear Closed Loop Systems Analysis Procedure Using Line Printer Plots of Characteristic Equation Root Loci, AMSAA Technical Memorandum No. 21, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md., November 1968, UNCLASSIFIED.
2. Burke, Harold H., Andrese, Joseph A.; A Fortran IV Program to Compute the Inverse Laplace Transform and Plot the Response of a Linear System. AMSAA Technical Memorandum No. 60, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md., (In publication), UNCLASSIFIED.
3. Burke, Harold H.; Mathematical Model and Simulation Studies of a Pneumatic Valve-Actuator-Load Impedance Configuration, AMSAA Technical Report No. 25, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md., April 1969, UNCLASSIFIED.
4. Burke, Harold H., Ogorzalek, M. J.; A Surface-To-Air Tactical Missile Guidance and Control Concept for the Advanced Forward Area Air Defense (AFAAD) Requirement, AMSAA Technical Memorandum No. 31, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md., September 1969 (CONFIDENTIAL).
5. Brown, James W.; A Preliminary Study of Shillelagh in a Combat, Optical Countermeasure Environment, AMSAA Technical Memorandum No. 62, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md., (In publication) (SECRET).
6. Sivaglian, B. D.; An Asymptotic Study of a Stationary (σ -s) Inventory Problem, Technical Report No. 8, June 1968, Themis Project, Systems Engineering Department, University of Florida, Gainesville, Fla.
7. Leon, Jesus, Herrero, Emilio E.; Determination of the Time-Domain Equivalents of Higher Order Laplace Transfer Functions By Analog Computation, Themis Project, Systems Research Center Industrial and Systems Engineering Department, University of Florida, Gainesville, Fla., November 1968.
8. Lefschetz, Solomon, Stability of Nonlinear Control Systems, Academic Press, New York, 1965.
9. Greensite, Arthur L.; Analysis and Design of Space Vehicle Flight Control Systems, Volume IV, Nonlinear Systems, NASA CR 823, July 1967.

APPENDIX A

LISTING OF SOURCE DECK

```
8      MAXT(10)MINS
      DO 10 J=1,100
      CALL RTLOCS
10     CONTINUE
      END
C
C
C
      SUBROUTINE RTLOCS
      DIMENSION SAVE1(100,100),SAVE2(100,100),XA(2000),XB(2000)
      DIMENSION X(100),IGA(100),IGR(100),C(100),D(100),ANS(100),
1SAVE(100),ERA(100),AS(100),BS(100),A(100),R(100),ROGTR(100),
2ROOTI(100),ATK(100),CK(100),CKS(100)
      INTEGER COMENT(20),DENSE,ROTLCS,EXPND,FREQSR,STAR,STAR1,STAR2,
*      STAR3,ONE,ZERO,DOT,VEE,AYE,CASH,BLANK,Q,
*      POINT(130,100),SPOT(130,50)
      DATA ONE/1H1/,ZERO/1H0/,DOT/1H./,AYE/1HA/,VEE/1HV/,BLANK/1H /,
*      DASH/1H-/,Q/1HQ/,STAR1/1H1/,STAR2/1H2/,STAR3/1H3/
      LOGICAL PRNT
      EQUIVALENCE (SPOT(1,1),POINT(1,1)),(STAR1,ZERO),(STAR3,DOT),
*      (JACKIE,KJ,1),(JOANN,KK,1)
      COMMON/FREEK/A,B
      COMMON /INFO4/ Q,BLANK,DASH
      COMMON /INFO7/ ICOUNT
      COMMON SAVE1,SAVE2,POINT
      COMMON /INFO8/ROTLCS,EXPND,FREQSR
      COMMON /INFO9/ DENSE
      READ (5,105) ROTLCS,EXPND,FREQSR,DENSE
      READ(5,11) COMENT
      WRITE(6,13) COMENT
      IF(ROTLCS.NE.0.AND.EXPND.NE.0.AND.FREQSR.NE.0) GO TO 37
      IF(ROTLCS.EQ.0.AND.EXPND.EQ.0) WRITE(6,30)
      IF(FREQSR.EQ.0.AND.EXPND.EQ.0) WRITE(6,31)
      IF(FREQSR.EQ.0.AND.ROTLCS.EQ.0) WRITE(6,32)
      IF(ROTLCS.NE.0.AND.EXPND.NE.0) WRITE(6,33)
      IF(FREQSR.NE.0.AND.EXPND.NE.0) WRITE(6,34)
      IF(FREQSR.NE.0.AND.ROTLCS.NE.0) WRITE(6,35)
38     DO277 IOU=1,2000
      XA(IOU)=0.0
277    XB(IOU)=0.0
      DO 401 KK=1,100
      DO 401 KJ=1,130
401    POINT(KJ,KK)=BLANK
      DO 402 KK=1,130
      POINT(KK,51)=DASH
402    POINT(KK,50)=DASH
      DO 403 KK=1,100
      DO 403 KJ=34,125,13
403    POINT(KJ,KK)=Q
      DO 404 KJ=34,124
      DO 404 KK=7, 93,7
      IF(KK.EQ.49) KK=58
404    POINT(KJ,KK)=DASH
      DO 405 KK=7, 93,7
      IF(KK.EQ.49) KK=58
405    POINT(25,KK)=ONE
      DO 406 KK=26,28
406    POINT(KK,7)=ZERO
```

	DO 407 KK=26,27	59
407	POINT(KK,14)=ZERO	60
	POINT(26,21)=ZERO	61
	DO 408 KK=35,65,7	62
	IF(KK.EQ.49) KK=58	63
408	POINT(24, KK)=DOT	64
	POINT(25,42)=ZERO	65
	POINT(25,58)=ZERO	66
	POINT(26,42)=ONE	67
	POINT(26,58)=ONE	68
	POINT(26,79)=ZERO	69
	POINT(26,86)=ZERO	70
	POINT(27,86)=ZERO	71
	DO 409 KK=26,28	72
409	POINT(KK,43)=ZERO	73
	DO 420 KK=1,100	74
	IF(KK.EQ.4) KK=98	75
420	POINT(26, KK)=G	76
	POINT(26,4)=VEE	77
	POINT(26,97)=AYE	78
	DO 7681 JACKIE=1,100	79
	DO 7681 JOANN=1,100	80
	SAVE1(JACKIE, JOANN)=0.0	81
7681	SAVE2(JACKIE, JOANN)=0.0	82
	NO=1	83
	ICOUNT=0	84
	MI=0	85
	JZO=1	86
	K1=1	87
	III=1	88
140	FORMAT(10X,7HDELTA=,110,5X,19HPOLY.ADDED IN A(S)=,110,5X,19HPOLY.	89
	*ADDED IN B(S)=,110,5X,9HPROR.NC.=,110)	90
12	READ(5,101) N,IA,IB,IPROB	91
	IF(IP.NE.0) GO TO 10	92
	IF(IA.NE.0) GO TO 10	93
	IF(N.NE.0) GO TO 10	94
	IF(IPROB.EQ.10000) GO TO 1016	95
101	FORMAT(7I10)	96
10	WRITE(6,1)	97
	DO 5 I = 1,100	98
	A(I) = 0.0	99
5	B(I) = 0.0	100
	FORK1=0.0	101
	FORK2 = 0.0	102
	FORK3 = 0.0	103
	WRITE(6,102)IPROB	104
102	FORMAT(1H1,9X,40HPOLYNOMIAL MULTIPLICATION AND ROOT LOCUS,44X,11	105
	1HPROBLEM NO.,15//)	106
	IF(N) 21, 20, 21	107
20	READ(5,103)Y,DY,YT	108
	WRITE(6,140)N,IA,IB,IPROB	109
	WRITE(6,141)Y,DY,YT	110
141	FORMAT(1H0,5X,10HK-INITIAL=F20.10,5X,12HINCREMENT K=,F20.10,5X,12H	111
	*K-TERMINATE=,F20.10)	112
103	FORMAT(4F10.0)	113
	GO TO 25	114
21	READ(5,104)(X(I),I=1,N)	115
	WRITE(6,142)(I,X(I),I=1,N)	116
104	FORMAT(7E10.0)	117
25	READ(5,105)(IGA(I),I=1,IA)	118

WRITE(6,143) (I,IGA(I),I=1,IA)	119
143 FORMAT(10X,24HNUMBER OF POLY. IN GROUP, I F NUMERATOR=,I10)	120
142 FORMAT(10X,2HX(,I3,2H)=,F20.10)	121
READ (5 ,105)(IGB(J), J= 1, IB)	122
WRITE(6,144) (J,IGB(J),J=1,IB)	123
144 FORMAT(10X,24HNUMBER CF POLY. IN GROUP, I3,16H OF DENOMINATOR=,I10	124
*)	125
105 FORMAT(7I10)	126
300 DO 15 I = 1,100	127
15 SAVF(I) = 0.0	128
ISDEG = 0	129
JJ = 1	130
200 READ (5 ,106)MDEG,(C(I), I=1, 6)	131
IF(MDEG.GT.6) GO TO 48	132
WRITE(6,145) (I,C(I),I=1,MDEG)	133
IF (MDEG - 6) 49, 49, 48	134
48 READ (5 ,104)(C(I), I = 7, MDEG)	135
WRITE(6,145) (I,C(I),I=1,MDEG)	136
106 FORMAT(I10,6E10.0)	137
49 IF (IGA(JJ)-1) 50, 51, 50	138
51 DO 56 I= 1, MDEG	139
56 ANS(I) = C(I)	140
IADEG = MDEG	141
IF (ISDEG - MDEG) 52, 52, 53	142
53 INDEG = ISDEG	143
GO TO 68	144
52 INDEG = MDEG	145
GO TO 68	146
50 READ (5 ,106)NDEG, (D(I), I =1, 6)	147
IF(NDEG.GT.6) GO TO 54	148
WRITE(6,145) (I,D(I),I=1,NDEG)	149
IF (NDEG - 6) 55, 55, 54	150
54 READ (5 ,104)(D(I), I=7, NDEG)	151
WRITE(6,145) (I,D(I),I=1,NDEG)	152
55 IADEG = NDEG + MDEG -1	153
CALL POLMPY (C,MDEG,D,NDEG,ANS)	154
IGA (JJ) = IGA(JJ) -1	155
IF (IGA(JJ) -1) 65, 65, 64	156
64 DO 60 I = 1, IADEG	157
60 C(I) = ANS(I)	158
MDEG =IADEG	159
GO TO 50	160
65 IF (ISDEG - IADEG) 66,66,67	161
66 INDEG = IADEG	162
GO TO 68	163
67 INDEG = ISDEG	164
68 CALL POLADD (SAVE,ISDEG,ANS,IADEG,ERA)	165
145 FORMAT(10X,2HC(,I3,2H)=,F20.10)	166
WRITE(6,1)	167
1 FORMAT(IH1//)	168
6800 IF (FRA(INDEG))6803, 6802, 6803	169
6802 INDEG = INDEG - 1	170
IF (INDEG) 6801, 6801, 6800	171
6801 INDEG = 1	172
6803 JJ = JJ + 1	173
DO 70 I = 1, INDEG	174
70 SAVE(I) = ERA(I)	175
ISDEG = INDEG	176
IA = IA -1	177
IF (IA) 201, 201, 200	178

201	IF (FORK1) 202, 202, 203	179
C	SAVE NUMERATOR.	180
202	DO 220 I = 1, ISDEG	181
220	A(I) = SAVE(I)	182
	IDA = ISDEG	183
	IA = IB	184
	FORK1 = 1.0	185
	DO 230 I=1, IA	186
230	IGA(I) = IGB(I)	187
C	START DENOMINATOR	188
	GO TO 300	189
C	SAVE DENOM.	190
203	DO 240 I = 1, ISDEG	191
240	B(I) = SAVE(I)	192
	IDB = ISDEG	193
	WRITE (6,109)	194
109	FORMAT (10X,41HCOEFFICIENTS ARE GIVEN IN ASCENDING ORDER/////)	195
339	IF (A(IDA)) 340, 341, 340	196
341	IDA = IDA - 1	197
	IF (IDA) 345, 345, 339	198
345	WRITE (6,120)	199
120	FORMAT (1H0,10X,20HPOLYNOMIAL A IS ZERO//)	200
	FORK2 = 1.0	201
	GO TO 410	202
340	IF (IDA - 2) 346, 347, 335	203
346	WRITE(6,121)A(I)	204
	STAR=STAR1	205
	PRNT=.TRUE.	206
121	FORMAT (1H0,10X,20HPOLYNOMIAL A IS A CONSTANT =,1P1E16.7//)	207
	GO TO 410	208
347	ROOT = - A(1) / A(2)	209
	WRITE (6,133)A(1), A(2)	210
133	FORMAT (10X,21HTHE COEFFICIENTS OF A/1P2E20.7)	211
	WRITE (6,122)ROOT	212
	STAR=STAR1	213
	PRNT=.TRUE.	214
	ANUMB1=ROOT	215
	ANUMB2=0.0	216
	CALL PLOTTER(SAVE1,SAVE2,ANUMB1,ANUMB2,POINT,XA,XB,MI,PRNT,STAR,III	217
	*,NO)	218
122	FORMAT (1H0,10X,23HROOT OF POLYNOMIAL A IS,1P1E16.7//)	219
	GO TO 410	220
C	WRITE POLYS	221
335	IDIA = IDA - 1	222
	WRITE (6,107)IDIA,(A(I),I=1,IDA)	223
	K = IDA	224
	DO 800 I = 1, IDA	225
	AS(I) = A(K)	226
800	K = K-1	227
	IDP2A=IDA *2	228
	ID2A= 2 *IDIA	229
	CALL MULLER (AS, IDIA,ROOTR,ROOTI)	230
	DO 805 I = 1, IDIA	231
	SAM = 100. * AMAX1(ABS(ROOTR(I)),ABS(ROOTI(I)))	232
	IF (SAM + ABS(ROOTR(I)).EQ. SAM) RCCTR(I) = 0.0	233
	IF (SAM + ABS(ROOTI(I)).EQ. SAM) RCOTI(I) = 0.0	234
805	CONTINUE	235
400	WRITE (6,111) (ROOTR(I),ROOTI(I),I=1, IDIA)	236
	CALL ERCHEK(ROOTI, IDIA)	237
	PRNT=.TRUE.	238

STAR=STAR1	239
DO 2 III=1, ID1A	240
ANUMB1=ROOTR(III)	241
ANUMB2=ABS(ROOTI(III))	242
CALL PLOTER(SAVE1,SAVE2,ANUMB1,ANUMB2,POINT,XA,XB,MI,PRNT,STAR,III	243
*,NO)	244
2 CONTINUE	245
410 IF(B(IDB)) 411, 412, 411	246
412 IDB = IDB - 1	247
IF (IDB) 445, 445, 410	248
445 WRITE (6 ,123)	249
123 FORMAT (1H0,10X,20HPOLYNOMIAL B IS ZERO//)	250
IF (FORK2)12,450,12	251
450 FORK3 = 1.0	252
GO TO 698	253
411 IF (IDB - 2) 451, 452, 499	254
451 WRITE (6 ,124)B(IDB)	255
STAR=STAR2	256
PRNT=.FALSE.	257
STAR=STAR3	258
124 FORMAT (1H0,10X,28HPOLYNOMIAL B IS A CONSTANT =,1P1E16.7//)	259
GO TO 698	260
452 ROOT = -B(1) / B(2)	261
WRITE (6 ,134)B(1), B(2)	262
134 FORMAT (10X,21HTHE COEFFICIENTS OF B/1P2E20.7)	263
WRITE (6 ,125)ROOT	264
STAR=STAR2	265
PRNT=.TRUE.	266
ANUMB1=ROOT	267
ANUMB2=0.0	268
CALL PLOTER(SAVE1,SAVE2,ANUMB1,ANUMB2,POINT,XA,XB,MI,PRNT,STAR,III	269
*,NO)	270
PRNT=.FALSE.	271
STAR=STAR3	272
125 FORMAT (1H0,10X,23HROOT OF POLYNOMIAL B IS,1P1E16.7//)	273
GO TO 698	274
107 FORMAT (10X,42HTHE COEFFICIENTS OF POLYNOMIAL A (ORDER = I3,1H)/ (275
11P6E20.7))	276
499 ID1B = IDB -1	277
WRITE (6,108)ID1B,(R(I),I=1, IDB)	278
108 FORMAT (////10X,42HTHE COEFFICIENTS OF POLYNOMIAL B (ORDER = I3,1H	279
1)/ (1P6E20.7))	280
K = IDB	281
DO 801 I = 1, ICB	282
BS(I) = B(K)	283
801 K = K-1	284
IDP2B= IDB * 2	285
ID2B = 2 * ID1B	286
CALL MULLER (BS, ID1B,ROOTR,ROOTI)	287
DO 806 I = 1, ID1B	288
SAM = 100. * AMAX1(ABS(ROOTR(I)),ABS(ROOTI(I)))	289
IF (SAM + ABS(ROOTR(I)).EQ. SAM) ROOTR(I)= 0.0	290
IF (SAM + ABS(ROOTI(I)).EQ. SAM) RCOTI(I)= 0.0	291
806 CONTINUE	292
500 WRITE (6,112)(ROOTR(I),ROOTI(I),I= 1, ID1B)	293
CALL ERCHEK(ROOTI, ID1B)	294
STAR=STAR2	295
PRNT=.TRUE.	296
DO 3 III=1, ID1B	297
ANUMB1=ROOTR(III)	298

ANUMB2=ABS(ROOTI(III))	299
CALL PLOTER(SAVE1,SAVE2,ANUMB1,ANUMB2,POINT,XA,XB,MI,PRNT,STAR,III	300
*,NO)	301
3 CONTINUE	302
PRNT=.FALSE.	303
STAR=STAR3	304
111 FORMAT (1H0,11X,14HTHE ROOTS OF A/ (1P1E20.7,6H +I ,1P1E14.7,1P1	305
1E20.7,6H +I ,1P1E14.7,1P1E20.7,6H +I ,1P1E14.7))	306
112 FORMAT (1H0,11X,14HTHE ROOTS OF B/ (1P1E20.7,6H +I ,1P1E14.7,1P1	307
1E20.7,6H +I ,1P1E14.7,1P1E20.7,6H +I ,1P1E14.7))	308
698 IF (FORK2)12,699,12	309
699 IF (FORK3)12,699,12	310
6991 WRITE (6 ,102)IPRJB	311
MSHEET = 5	312
C START K CALCULATIONS	313
IF (N) 702,702,533	314
533 DO 550 I= 1, N	315
DO 541 J= 1, IDA	316
541 ATK(J) = X(I) * A(J)	317
C COMPUTE ROOTS OF K * A + B	318
IDC= MAX0(IDA, IDB)	319
CALL POLADD (ATK,IDA,R,IDB,CK)	320
IDS = IDC	321
554 IF (CK(IDS))555, 557, 555	322
557 IDS = IDS - 1	323
IF (IDS) 558,558, 554	324
558 WRITE (6 ,129)X(I)	325
129 FORMAT (1H0,10X,35HPOLYNOMIAL K*A + B IS ZERO FOR K =,1P1E16.7//)	326
GO TO 550	327
555 IF (IDS - 2) 559, 560, 561	328
559 WRITE (6 ,130)CK(IDS), X(I)	329
130 FORMAT (1H0,10X,35HPOLYNOMIAL K*A + B IS A CONSTANT = ,1P1E15.7,10	330
1H FOR K = ,1P1E14.7//)	331
GO TO 550	332
560 ROOT = -CK(1) / CK(2)	333
WRITE (6 ,131)ROOT, X(I)	334
131 FORMAT (1H0,10X,18HROOT OF K*A + B = ,1P1E15.7,10H FOR K = ,1P1E1	335
14.7//)	336
GO TO 550	337
561 K = IDS	338
DO 803 J = 1,IDS	339
CKS(J) = CK(K)	340
803 K = K - 1	341
ID1C = IDS - 1	342
IDP2C = IDS * 2	343
ID2C = 2 * ID1C	344
CALL MULLER (CKS,ID1C,ROOTR,ROOTI)	345
DO 807 J = 1,ID1C	346
SAM = 100. * AMAX1(ABS(ROOTR(J)),ABS(ROOTI(J)))	347
IF (SAM + ABS(ROOTR(J)).EQ.SAM) RCOTR(J) = 0.0	348
IF (SAM + ABS(ROOTI(J)).EQ.SAM) RCCTI(J) = 0.0	349
807 CONTINUE	350
WRITE (6,808)ID1C,X(I),(CK(J),J=1,IDS)	351
808 FORMAT (///10X,48HTHE COEFFICIENTS OF POLYNOMIAL K*A + B (ORDER =	352
113,7H) K = 1P1E16.7/(1P6E20.7))	353
545 WRITE (6,115)(ROOTR(J),ROOTI(J),J=1,ID1C)	354
CALL ERCHEK(ROOTI,ID1C)	355
CALL SAVER(ROOTR,ROOTI,ID1C,SAVE1,SAVE2,JZ0,K1)	356
115 FORMAT (1H0,9X,16HROOTS OF K*A + B/(1P1E20.7,6H + I ,1P1E14.7,1P1	357
1E20.7,6H + I ,1P1E14.7,1P1E20.7,6H + I ,1P1E14.7))	358

5452	MSHEET = MSHEET - 1	359
	IF (MSHEET) 546, 546, 550	360
546	WRITE (6,102)IPROB	361
	MSHEET = 5	362
550	CONTINUE	363
	GO TO 12	364
702	DO 705 J = 1, IDA	365
705	ATK(J) = Y * A(J)	366
C	COMPUTE ROOTS OF $K * A + B$	367
	IDC = MAX0(IDA, IDR)	368
	CALL POLADD (ATK, IDA, R, IDB, CK)	369
	IDS = IDC	370
754	IF (CK(IDS)) 755, 757, 755	371
757	IDS = IDS - 1	372
	IF (IDS) 758, 758, 754	373
758	WRITE (6,129)Y	374
	GO TO 711	375
755	IF (IDS - 2) 759, 760, 761	376
759	WRITE (6,130)CK(IDS), Y	377
	GO TO 711	378
760	ROOT = -CK(1) / CK(2)	379
	WRITE (6,131)ROOT, Y	380
	GO TO 711	381
761	K = IDS	382
	DO 804 I = 1, IDS	383
	CKS(I) = CK(K)	384
804	K = K - 1	385
	IDIC = IDS - 1	386
	IDP2C = IDS * 2	387
	ID2C = 2 * IDIC	388
	CALL MULLER (CKS, IDIC, ROOTR, ROOTI)	389
	DO 809 I = 1, IDIC	390
	SAM = 100. * AMAX1(ABS(ROOTR(I)), ABS(ROOTI(I)))	391
	IF (SAM + ABS(ROOTR(I)).EQ. SAM) ROOTR(I) = 0.0	392
	IF (SAM + ABS(ROOTI(I)).EQ. SAM) ROOTI(I) = 0.0	393
809	CONTINUE	394
	WRITE (6,808) IDIC, Y, (CK(I), I=1, IDS)	395
	WRITE (6,115) (ROOTR(J), ROOTI(J), J=1, IDIC)	396
	CALL ERCHEK(ROOTI, IDIC)	397
	CALL SAVER(ROOTR, ROOTI, IDIC, SAVE1, SAVE2, JZ0, K1)	398
711	Y = Y + DY	399
	IF (Y - YT) 712, 712, 12	400
712	MSHEET = MSHEET - 1	401
	IF (MSHEET) 713, 713, 702	402
713	WRITE (6,102)IPROB	403
	MSHEET = 5	404
	GO TO 702	405
37	WRITE(6,36)	406
	GO TO 38	407
1016	CALL PLOTTER(SAVE1,SAVE2,ANUMB1,ANUMB2,POINT,XA,XB,MI,PRNT,STAR,III	408
	*,NO)	409
1018	IF(EXPND.EQ.0) GO TO 1017	410
	READ(5,22)L	411
22	FORMAT(110)	412
	IF(L.EQ.0) GO TO 1017	413
	CALL EXPAND(L,XA,XB)	414
1017	IF(FREQSK.EQ.0) RETURN	415
	CALL FREQRS(XA,XB)	416
11	FORMAT(20A4)	417
13	FORMAT(//////////,20X,20A4)	418

```

30 FORMAT(////20X,44HONLY THE FREQUENCY RESPONSE HAS BEEN PLOTTED ) 419
31 FORMAT(////20X,45HONLY THE ROOT LOCUS LOG PLOT HAS BEEN PLOTTED ) 420
32 FORMAT(////20X,39HONLY THE LINEAR EXPAND HAS BEEN PLOTTED ) 421
33 FORMAT(////20X,59HTHE ROOT LOCUS LOG PLOT AND LINEAR EXPAND HAVE B 422
  *EEN PLOTTED ) 423
34 FORMAT(////20X,62HTHE LINEAR EXPAND AND THE FREQUENCY RESPONSE HAV 424
  *E BEEN PLOTTED ) 425
35 FORMAT(////20X,68HTHE ROOT LOCUS LOG PLOT AND THE FREQUENCY RESPON 426
  *SE HAVE BEEN PLOTTED ) 427
36 FORMAT(////20X,28HEVERYTHING HAS BEEN PLOTTED ) 428
  RETURN 429
  END 430

```

```

C 431
C 432
C 433
C 434

```

```

SUBROUTINE POLMPY (A,N,B,M,C) 435
DIMENSION A(1),B(1),C(1) 436
K = M+N 437
DO 5 I=1,K 438
5 C(I) = 0.0 439
DO 10 I=1,N 440
  L = I-1 441
DO 10 J=1,M 442
  L = L+1 443
10 C(L) = C(L)+A(I)*B(J) 444
  RETURN 445
  END 446

```

```

C 447
C 448
C 449
C 450

```

```

SUBROUTINE POLADD (A,N,B,M,C) 451
DIMENSION A(1),B(1),C(1) 452
IF (N-M) 1,1,2 453
1 NK = N 454
  GO TO 5 455
2 NK = M 456
5 DO 10 I=1,NK 457
10 C(I) = A(I)+B(I) 458
  NK = NK+1 459
  IF (N-M) 11,25,15 460
11 DO 20 I=NK,M 461
20 C(I) = B(I) 462
25 RETURN 463
15 DO 30 I=NK,N 464
30 C(I) = A(I) 465
  RETURN 466
  END 467

```

```

C 468
C 469
C 470
C 471

```

```

SUBROUTINE MULLER(COE,N1,ROOTR,ROOTI) 472
DIMENSION COE(1),ROOTR(1),ROOTI(1) 473
N2=N1+1 474
N4=0 475
I=N1+1 476
19 IF(COE(I))9,7,9 477
7 N4=N4+1 478

```

ROOTR(N4)=0.	479
ROOTI(N4)=0.	480
I=-1	481
IF(N4-N1)19,37,19	482
9 CONTINUE	483
10 AXR=0.8	484
AXI=0.	485
L=1	486
N3=1	487
ALP1R=AXR	488
ALP1I=AXI	489
M=1	490
GOTO99	491
11 BET1R=TEMR	492
BET1I=TEMI	493
AXR=0.85	494
ALP2R=AXR	495
ALP2I=AXI	496
M=2	497
GOTO99	498
12 RET2R=TEMR	499
RET2I=TEMI	500
AXR=C.9	501
ALP3R=AXR	502
ALP3I=AXI	503
M=3	504
GOTO99	505
13 RET3R=TEMR	506
BET3I=TEMI	507
14 TE1=ALP1R-ALP3R	508
TE2=ALP1I-ALP3I	509
TE5=ALP3R-ALP2R	510
TE6=ALP3I-ALP2I	511
TEM=TE5*TE5+TE6*TE6	512
TE3=(TE1*TE5+TE2*TE6)/TEM	513
TE4=(TE2*TE5-TE1*TE6)/TEM	514
TE7=TE3+1.	515
TE9=TE3*TE3-TE4*TE4	516
TE10=2.*TE3*TE4	517
DE15=TE7*BET3R-TE4*BET3I	518
DE16=TE7*BET3I+TE4*BET3R	519
TE11=TE3*BET2R-TE4*BET2I+BET1R-DE15	520
TE12=TE3*BET2I+TE4*BET2R+BET1I-DE16	521
TE7=TE9-1.	522
TE1=TE9*BET2R-TE10*BET2I	523
TE2=TE9*BET2I+TE10*BET2R	524
TE13=TE1-BET1R-TE7*BET3R+TE10*BET3I	525
TE14=TE2-BET1I-TE7*BET3I-TE10*BET3R	526
TE15=DE15*TE3-DE16*TE4	527
TE16=DE15*TE4+DE16*TE3	528
TE1=TE13*TE13-TE14*TE14-4.*(TE11*TE15-TE12*TE16)	529
TE2=2.*TE13*TE14-4.*(TE12*TE15+TE11*TE16)	530
TEM = SQRT (TE1*TE1+TE2*TE2)	531
IF(TE1)113,113,112	532
113 TE4 = SQRT (.5 * (TEM - TE1))	533
TE3=.5*TE2/TE4	534
GO TO 111	535
112 TE3 = SQRT (.5 * (TEM + TE1))	536
IF(TE2)110,200,200	537
110 TE3=-TE3	538

200	TE4=.5*TE2/TE3	539
111	TE7=TE13+TE3	540
	TE8=TE14+TE4	541
	TE9=TE13-TE3	542
	TE10=TE14-TE4	543
	TE1=2.*TE15	544
	TE2=2.*TE16	545
	IF(TE7*TE7+TE8*TE8-TE9*TE9-TE10*TE10)204,204,205	546
204	TE7=TE9	547
	TE8=TE10	548
205	TEM=TE7*TE7+TE8*TE8	549
	TE3=(TE1*TE7+TE2*TE8)/TEM	550
	TE4=(TE2*TE7-TE1*TE8)/TEM	551
	AXR=ALP3R+TE3*TE5-TE4*TE6	552
	AXI=ALP3I+TE3*TE6+TE4*TE5	553
	ALP4R=AXR	554
	ALP4I=AXI	555
	M=4	556
	GO TO 99	557
15	N6=1	558
38	IF (ABS (HELL) + ABS (BELL) - 1.E-20) 18,18,16	559
16	TE7 = ABS (ALP3R - AXR) + ABS (ALP3I - AXI)	560
	IF (TE7 / (ABS (AXR) + ABS (AXI)) - 1.E-7)18,18,17	561
17	N3=N3+1	562
	ALP1R=ALP2R	563
	ALP1I=ALP2I	564
	ALP2R=ALP3R	565
	ALP2I=ALP3I	566
	ALP3R=ALP4R	567
	ALP3I=ALP4I	568
	BET1R=BET2R	569
	BET1I=BET2I	570
	BET2R=BET3R	571
	BET2I=BET3I	572
	BET3R=TEMR	573
	BET3I=TEMI	574
	IF(N3-100)14,18,18	575
18	N4=N4+1	576
	ROOTR(N4)=ALP4R	577
	ROOTI(N4)=ALP4I	578
	N3=0	579
41	IF(N4-N1)30,37,37	580
37	RETURN	581
30	IF (ABS (ROOTI(N4)) - 1.E-5)10,10,31	582
31	GO TO(32,10),L	583
32	AXR=ALP1R	584
	AXI=-ALP1I	585
	ALP1I=-ALP1I	586
	M=5	587
	GO TO 99	588
33	BET1R=TEMR	589
	BET1I=TEMI	590
	AXR=ALP2R	591
	AXI=-ALP2I	592
	ALP2I=-ALP2I	593
	M=6	594
	GO TO 99	595
34	BET2R=TEMR	596
	BET2I=TEMI	597
	AXR=ALP3R	598

AXI=-ALP3I	599
ALP3I=-ALP3I	600
L=2	601
M=3	602
99 TEMR=COE(1)	603
TEMI=0.0	604
DO100I=1,N1	605
TE1=TEMR*AXR-TEMI*AXI	606
TEMI=TEMI*AXR+TEMR*AXI	607
100 TEMR= TE1+COE(I+1)	608
HELL=TEMR	609
HELL=TEMI	610
42 IF(N4)102,103,102	611
102 DO101I=1,N4	612
TEMI=AXR-ROOTK(I)	613
TEM2=AXI-ROOTI(I)	614
TE1=TEMI*TEMI+TEM2*TEM2	615
TE2=(TEMR*TEMI+TEMI*TEM2)/TE1	616
TEMI=(TEMI*TEMI-TEMR*TEM2)/TE1	617
101 TEMR=TE2	618
103 GO TO(11,12,13,15,33,34),M	619
END	620
C	621
C	622
C	623
C	624
SUBROUTINE EXPLOT(XARRAY, IDIMEN, YARRAY, KDIMEN, BLIMIT, LET)	625
INTEGER POINT(130,100), GRAPHL(120,58), STAR	626
DIMENSION XARRAY(IDIMEN), YARRAY(KDIMEN)	627
DIMENSION ENCRM1(120), ENCRM2(58)	628
DIMENSION SAVE1(100,100), SAVE2(100,100)	629
COMMON SAVE1, SAVE2, POINT	630
EQUIVALENCE(POINT(1,1), GRAPHL(1,1))	631
DATA STAR/1H+ /	632
WRITE(6,1)	633
READ(5,10)D1,D2	634
D3=BLIMIT	635
D4=-BLIMIT	636
IF(D2.GT.D1) CALL SWITCH(C1,D2)	637
IF(D4.GT.D3) CALL SWITCH(C3,D4)	638
IC1=D3	639
JDIMEN=KDIMEN	640
CALL BLANKR(GRAPHL,120,58,120)	641
DO 22 I=1,JDIMEN	642
X=XARRAY(I)	643
Y=YARRAY(I)	644
IF(X.EQ.0.0.AND.Y.EQ.0.0) GO TO 22	645
CALL LINAR (X,LIMITX,120,C1,D2,ENCRM1)	646
CALL LINAR (Y,LIMITY,58,D3,D4,ENCRM2)	647
IF(LIMITX.EQ.0.OR.LIMITY.EQ.0) GO TO 22	648
LIMITX=121-LIMITX	649
GRAPHL(LIMITX,LIMITY)=STAR	650
22 CONTINUE	651
WRITE(6,30) IC1,LET,(GRAPHL(I,1),I=1,120)	652
IC1=D4	653
WRITE(6,34)((GRAPHL(I,J),I=1,120),J=2,57)	654
WRITE(6,30) IC1,LET,(GRAPHL(I,58),I=1,120)	655
WRITE(6,35) D2,D1	656
1 FORMAT(1H1)	657
10 FORMAT(6F10,0,20X)	658

30	FORMAT(1X, I4, A4, 1X, 120A1)	659
34	FORMAT(10X, 120A1)	660
35	FORMAT(3X, 1P1E10.3, 7HRAD/SEC, 91X, 1P1E10.3, 7HRAD/SEC)	661
61	RETURN	662
	END	663
C		664
C		665
C		666
C		667
	SUBROUTINE LINAR (Y, LIMITY, IDIMEN, DIMNS1, DIMNS2, ENCRMT)	668
	DIMENSION ENCRMT(IDIMEN)	669
	DIMENS=DIMNS1-DIMNS2	670
	LIMITY=0	671
	J=IDIMEN-1	672
	A=J	673
	DELTA=DIMENS/A	674
	ENCRMT(1)=DIMNS1	675
	ENCRMT(IDIMEN)=DIMNS2	676
	DO 10 I=2, J	677
10	ENCRMT(I)=ENCRMT(I-1)-DELTA	678
	DO 11 I=1, J	679
11	IF(Y.LE.ENCRMT(I).AND.Y.GE.ENCRMT(I+1))GO TO 20	680
	GO TO 40	681
20	LIMITY=I	682
40	RETURN	683
	END	684
C		685
C		686
	SUBROUTINE SWITCH(X, Y)	687
	TEST=Y	688
	Y=X	689
	X=TEST	690
	RETURN	691
	END	692
C		693
C		694
	SUBROUTINE ERCHEK(X, I)	695
	DIMENSION X(100)	696
	DATA ERR LIM/0.1E-8/	697
	DO 10 J=1, I	698
10	IF(ABS(X(J)).LT.ERR LIM) X(J)=0.0	699
	RETURN	700
	END	701
C		702
C		703
C		704
C		705
	SUBROUTINE EXPAND(I LOVE, X, Y)	706
	DIMENSION SAVE1(100, 100), SAVE2(100, 100), X2(2000), Y2(2000), X(2000),	707
	Y(2000), X1(2000), Y1(2000), Z(125), Z1(87)	708
	INTEGER SPOT(130, 50), DASH, C, BLANK, PCINT(130, 100)	709
	DATA C/1H1/, DASH/1H-/, BLANK/1H /	710
	DATA APLUS/0.0/, BPLUS/0.0/, AMINUS/0.0/, BMINUS/0.0/	711
	COMMON /INFU4/ 0, BLANK, DASH	712
	COMMON /PAYNE/AM, AP, BM, BP	713
	COMMON /PJ/X1, Y1	714
	COMMON SAVE1, SAVE2, POINT	715
	EQUIVALENCE(AP, APLUS), (AM, AMINUS), (BM, BMINUS), (BP, BPLUS)	716
	EQUIVALENCE(X1(1), Z(1)), (Y1(1), Z1(1))	717
	EQUIVALENCE (SPOT(1, 1), POINT(1, 1)), (I1, I2)	718

DO 40 J=1,ILOVE	719
DO 50 M=1,2000	720
X1(M)=0.0	721
50 Y1(M)=0.0	722
CALL BLANKR(SPOT,130,50,126)	723
DO 54 M=1,2000	724
X2(M)=0.0	725
54 Y2(M)=0.0	726
READ(5,10) OMEGA, ENCRMT, SIGMA, DELTA	727
OMEGA=ABS(OMEGA)	728
10 FORMAT(4F10.0)	729
A=ENCRMT	730
PERCNT=A*0.01	731
B=OMEGA	732
B=B*PERCNT	733
APLUS=B+OMEGA	734
D=AMINUS	737
I1=1	738
AMINUS=OMEGA-B	735
C=APLUS	736
DO 41 L=1,2000	739
IF(Y(L).LT.D)GO TO 41	740
IF(Y(L).GT.C)GO TO 41	741
23 X1(I1)=X(L)	742
Y1(I1)=Y(L)	743
I1=I1+1	744
41 CONTINUE	745
A=DELTA	746
901 PERCNT=A*0.01	747
B=SIGMA	748
903 B=B*PERCNT	749
BPLUS=SIGMA+B	750
BMINUS=SIGMA-B	751
C=BPLUS	752
D=BMINUS	753
I2=1	754
910 DO 42 L=1,2000	755
IF(C.GT.0.0) GO TO 60	756
IF(X1(L).GT.D) GO TO 42	757
IF(X1(L).LT.C) GO TO 42	758
GO TO 25	759
60 IF(X1(L).GT.C) GO TO 42	760
IF(X1(L).LT.D) GO TO 42	761
25 X2(I2)=X1(L)	762
Y2(I2)=Y1(L)	763
I2=I2+1	764
42 CONTINUE	765
CALL SPLIT(X2,Y2,SPOT,APLUS,AMINUS,BPLUS,BMINUS)	766
40 CONTINUE	767
RETURN	768
END	769
	770
	771
	772
	773
	774
	775
	776
	777
	778

C
C
C
C

```

SUBROUTINE SPLIT(X,Y,SPOT,APLUS,AMINUS,BPLUS,BMINUS)
DIMENSION X1(2000),Y1(2000)
DIMENSION X(2000),Y(2000),Z(125),Z1(87)
INTEGER SPOT(130,50)
COMMON /PJ/X1,Y1

```

	EQUIVALENCE(X1(1),Z(1)),(Y1(1),Z1(1))	779
	A=APLUS	780
	B=AMINUS	781
	C=A-B	782
	D=BPLUS	783
	E=BMINUS	784
	G=ABS(D)	785
	H=ABS(E)	786
	F=G-H	787
	DELTA=C/124.0	788
	DIFF=F/50.0	789
	DO 11 J=1,124	790
	Z(J)=A	791
11	A=A-DELTA	792
	Z(125)=B	793
	IF(D.LT.0.0)DIFF=-DIFF	794
	DO 12 J=1,49	795
	Z1(J)=D	796
12	D=D-DIFF	797
	Z1(50)=E	798
	CALL BRAKUP(Z,X,Y,SPOT,Z1)	799
	RETURN	800
	END	801
C		802
C		803
C		804
C		805
	SUBROUTINE BRAKUP(YY,X,Y,SPCT,XX)	806
	LOGICAL SKIP,SKIPI	807
	INTEGER SPOT(130,50),	808
	XPT,YPT,STAR	808
	DATA STAR/1HX/	809
	DIMENSION X(2000),Y(2000),YY(125),XX(87)	810
	COMMON /PAYNE/AM,AP,BM,BP	811
	EQUIVALENCE(AP,APLUS),(AM,AMINUS),(BM,BMINUS),(BP,BPLUS)	812
	SKIP=.FALSE.	813
	SKIPI=.FALSE.	814
	DO 43 J=1,2000	815
	DO 43 I=1,125	816
	IF(Y(IJ).NE.0.0) GO TO 30	817
	IF(X(IJ).NE.0.0) GO TO 30	818
	L=J+1	819
	M=J+6	820
	DO 60 N=L,M	821
	IF(N.GT.2000) GO TO 60	822
	IF(Y(N).NE.0.0) GO TO 30	823
	IF(X(N).NE.0.0) GO TO 30	824
60	CONTINUE	825
	GO TO 40	826
30	IF(SKIP) GO TO 20	827
	IF(Y(J).LT.YY(I)) GO TO 20	828
	YPT=I	829
	SKIP=.TRUE.	830
20	IF(SKIPI)GO TO 48	831
	IF(I.GT.50) GO TO 43	832
	IF(BPLUS.GT.0.0) GO TO 10	833
	IF(X(J).GT.XX(I)) GO TO 43	834
	GO TO 11	835
10	IF(X(J).LT.XX(I)) GO TO 43	836
11	IF(BPLUS.LT.0.0) XPT=I	837
	IF(BPLUS.GE.0.) XPT=51-I	838

	SKIPI=.TRUE.	839
	IF(.NOT.SKIPI)GO TO 43	840
50	I=125	841
	SPOT(YPT,XPT)=STAR	842
	SKIP=.FALSE.	843
	SKIPI=.FALSE.	844
48	IF(SKIPI)GO TO 50	845
43	CONTINUE	846
40	CALL RITEIT(SPOT)	847
	RETURN	848
	END	849
C		850
C		851
C		852
C		853
	SUBROUTINE RITEIT(SPOT)	854
	COMMON /PAYNE/AM,AP,BM,BP	855
	EQUIVALENCE(AP,APLUS),(AM,AMINUS),(BM,BMINUS),(BP,BPLUS)	856
	INTEGER SPOT(130,50)	857
	IF(BP.LT.BM) GO TO 40	858
	A=BP	859
	B=BM	860
	WRITE(6,1)	861
1	FORMAT(1H1,40X,43HCOMPLEX FREQUENCY PLANE,RIGHT HAND QUADRANT)	862
	GO TO 30	863
40	A=BM	864
	B=BP	865
	WRITE(6,2)	866
2	FORMAT(1H1,40X,43HCOMPLEX FREQUENCY PLANE, LEFT HAND QUADRANT)	867
30	WRITE(6,12)AP,AM	868
12	FORMAT(114X,15HC-----J-OMEGA,/1X,F8.2,112X,F8.2)	869
	WRITE(6,14)B	870
14	FORMAT(124X,5HSIGMA,/126X,1HI,/126X,1HV,/121X,F8.2)	871
	WRITE(6,11) SPOT	872
11	FORMAT(1X,130A1)	873
	WRITE(6,15)A	874
15	FORMAT(60X,27HLINEAR EXPANC PLOT(RAD/SEC) ,34X,F8.2)	875
	RETURN	876
	END	877
C		878
C		879
	SUBROUTINE SAVER(ROOTR,ROOTI,IDIC,SAVE1,SAVE2,JZO,K1)	880
	DIMENSION SAVE1(100,100),SAVE2(100,100),ROOTR(100),ROOTI(100)	881
	IF(K1)30,9,10	882
9	K1=1	883
10	ID1=IDIC	884
	IDIC=IDIC+(K1-1)	885
	IF(IDIC.GE.100)GO TO 30	886
50	DO 40 IZAP=K1, IDIC	887
	IZA=IZAP-(K1-1)	888
	SAVE1(JZO,IZAP)=ROOTR(IZA)	889
	SAVE2(JZO,IZAP)=ROOTI(IZA)	890
40	CONTINUE	891
	K1=IDIC+1	892
	GO TO 20	893
30	JZO=JZO+1	894
	IDIC=ID1	895
	GO TO 9	896
20	RETURN	897
	END	898

C
C
C
C

```
      SUBROUTINE PLOTER(SAVE1,SAVE2,ANUMB1,ANUMB2,POINT,XA,XB,PI,PRNT,  
      *STAR,NBB,NO) 899  
      DIMENSION SAVE1(100,100),SAVE2(100,100),XA(2000),XB(2000) 900  
      INTEGER POINT(130,100),DASH,BLANK,STAR,ROTLCS,EXPND,FREQSR,DENSE 901  
      COMMON /INFO7/ ICOUNT 902  
      COMMON /INFO8/ROTLCS,EXPND,FREQSR 903  
      COMMON /INFO9/ DENSE 904  
      LOGICAL PRNT 905  
      IF(PRNT) GO TO 43 906  
      DO 40 NAR=1,100 907  
      DO 40 NBR=1,100 908  
      IF(SAVE1(NAB,NBR))41,42,41 909  
42 IF(SAVE2(NAB,NBR))41,40,41 910  
41 ANUMR1=SAVE1(NAB,NBR) 911  
      ANUMR2=SAVE2(NAB,NBR) 912  
      ANUMB2=ABS(ANUMR2) 913  
43 AZZ=ANUMB2 914  
      AZZ=AZZ*100000.0 915  
      NZZ=AZZ 916  
      IF(NZZ.EQ.0) GO TO 45 917  
50 NBB=NBB+1 918  
45 CALL EXCUTE(ANUMR1,ANUMB2,POINT,XA,XB,PI,NO,STAR) 919  
      IF (DENSE.NE.1) GO TO 10 920  
      ICOUNT=ICOUNT+1 921  
      GO TO 11 922  
10 IF(PRNT) ICOUNT=ICOUNT+1 923  
11 IF(PRNT) RETURN 924  
40 CONTINUE 925  
      IF(ROTLCS.EQ.0) RETURN 926  
70 CALL WRITIT(XA,XB) 927  
      CALL PREPAR(POINT) 928  
      RETURN 929  
      END 930  
931  
932  
933  
934  
935
```

C
C
C
C

```
      SUBROUTINE EXCUTE(ANUMB1,ANUMB2,POINT,XA,XB,PI,NO,STAR) 936  
      LOGICAL SKIP1,SKIP2,LESS 937  
      INTEGER ROTLCS,EXPND,FREQSR 938  
      DIMENSION XA(2000),XB(2000) 939  
      INTEGER POINT(130,100),DASH,BLANK,STAR 940  
      COMMON /INFO8/ROTLCS,EXPND,FREQSR 941  
      DATA K1/10/,K2/100/,K3/1000/,K4/10000/,NEGONE/-1/ 942  
      LESS=.FALSE. 943  
      I=0 944  
      J=0 945  
      XA(NO)=ANUMB1 946  
      XB(NO)=ABS(ANUMB2) 947  
      NO=NO+1 948  
      IF(ROTLCS.EQ.0) RETURN 949  
      IF(ABS(ANUMB1).GT.10000.0) GO TO 50 950  
      IF(ABS(ANUMB2).GT.10000.0) GO TO 50 951  
      IF(ABS(ANUMR1).EQ.0.0) GO TO 500 952  
      IF(ABS(ANUMB2).EQ.0.0) GO TO 500 953  
      IF(ABS(ANUMB2).LT.0.001) GO TO 50 954  
955  
956  
957  
958
```

	IF(ABS(ANUMB1).LT.0.001) GO TO 50	959
500	CALL SCALE1(K1,K2,K3,K4,NEGONE,I,ANUMB1,ICONS,LESS,SKIP1)	960
	CALL SCALE2(K1,K2,K3,K4,J,JCONS,ANUMB2,SKIP2)	961
	CALL WPOINT(J,JCONS,L,ANUMB2,SKIP2)	962
	CALL SPOINT(I,ANUMB1,LESS,ICONS,L,POINT,SKIP1,NO,XA,XB,STAR)	963
50	RETURN	964
	END	965
C		966
C		967
C		968
C		969
	SUBROUTINE SCALE1(K1,K2,K3,K4,NEGONE,I,ANUMB1,ICONS,LESS,SKIP1)	970
	LOGICAL LESS	971
	LOGICAL SKIP1	972
	SKIP1=.FALSE.	973
	ICONS=1	974
	I=1	975
	AKEEP=ANUMB1	976
21	NUMB1=ANUMB1	977
	NUMB=IABS(NUMB1)	978
	IF(NUMB.EQ.0)GOTO 12	979
	IF(NUMB1)31,40,40	980
40	IF(NUMB1.GE.10)GOTO 11	981
	GO TO 50	982
31	LESS=.TRUE.	983
	XXX=-NUMB1	984
	NUMB1=XXX	985
	GOTO 40	986
12	GOTO(1,2,3,4),I	987
1	ANUMB1=AKEEP	988
	ICONS=K1	989
	RK1=K1	990
	ANUMB1=ANUMB1*RK1	991
	I=2	992
	GOTO 21	993
2	ANUMB1=AKEEP	994
	ICONS=K2	995
	RK2=K2	996
	ANUMB1=ANUMB1*RK2	997
	I=3	998
	GOTO 21	999
3	ANUMB1=AKEEP	1000
	ICONS=K3	1001
	RK3=K3	1002
	ANUMB1=ANUMB1*RK3	1003
	I=4	1004
	GOTO 21	1005
4	ANUMB1=AKEEP	1006
	ICONS=K4	1007
	RK4=K4	1008
	ANUMB1=ANUMB1*RK4	1009
	NUMB1=ANUMB1	1010
	IF(NUMB1.EQ.0) GO TO 51	1011
	GO TO 50	1012
11	SKIP1=.TRUE.	1013
	GO TO (6,7,8,9),I	1014
6	ANUMB1=AKEEP	1015
	ICONS=K1	1016
	RK1=K1	1017
	ANUMB1=ANUMB1/RK1	1018

I=2	1019
GOTO 21	1020
7 ANUMB1=AKEEP	1021
ICONS=K2	1022
RK2=K2	1023
ANUMB1=ANUMB1/RK2	1024
I=3	1025
GOTO 21	1026
8 ANUMB1=AKEEP	1027
ICONS=K3	1028
RK3=K3	1029
ANUMB1=ANUMB1/RK3	1030
I=4	1031
GOTO 21	1032
9 ANUMB1=AKEEP	1033
ICONS=K4	1034
RK4=K4	1035
ANUMB1=ANUMB1/RK4	1036
GO TO 50	1037
51 I=5	1038
50 RETURN	1039
END	1040
	1041
	1042
	1043
	1044
	1045
	1046
	1047
	1048
	1049
	1050
	1051
	1052
	1053
	1054
	1055
	1056
	1057
	1058
	1059
	1060
	1061
	1062
	1063
	1064
	1065
	1066
	1067
	1068
	1069
	1070
	1071
	1072
	1073
	1074
	1075
	1076
	1077
	1078

C
C
C
C

SUBROUTINE SCALE2(K1,K2,K3,K4,J,JCONS,ANUMB2,SKIP2)

LOGICAL SKIP2

SKIP2=.FALSE.

JCONS=1

J=1

ANUMB2=ABS(ANUMB2)

BKEEP=ANUMB2

20 NUMB2=ANUMB2

IF(NUMB2.EQ.0)GO TO 10

IF(NUMB2.GE.10)GO TO 11

GO TO 50

10 GOTO(1,2,3,4),J

1 ANUMB2=BKEEP

JCONS=K1

RK1=K1

ANUMB2=ANUMB2*RK1

J=2

GO TO 20

2 ANUMB2=BKEEP

JCONS=K2

RK2=K2

ANUMB2=ANUMB2*RK2

J=3

GO TO 20

3 ANUMB2=BKEEP

JCONS=K3

RK3=K3

ANUMB2=ANUMB2*RK3

J=4

GO TO 20

4 ANUMB2=BKEEP

JCONS=K4

RK4=K4

ANUMB2=ANUMB2*RK4

NUMB2=ANUMB2	1079
IF(NUMB2.EQ.0) GO TO 51	1080
GO TO 50	1081
11 SKIP2=.TRUE.	1082
GO TO (6,7,8,9),J	1083
6 JCONS=K1	1084
ANUMB2=BKEEP	1085
RK1=K1	1086
ANUMB2=ANUMB2/RK1	1087
J=2	1088
GO TO 20	1089
7 JCONS=K2	1090
ANUMB2=BKEEP	1091
RK2=K2	1092
ANUMB2=ANUMB2/RK2	1093
J=3	1094
GO TO 20	1095
8 JCONS=K3	1096
ANUMB2=BKEEP	1097
RK3=K3	1098
ANUMB2=ANUMB2/RK3	1099
J=4	1100
GO TO 20	1101
9 JCONS=K4	1102
ANUMB2=BKEEP	1103
RK4=K4	1104
ANUMB2=ANUMB2/RK4	1105
GO TO 50	1106
51 J=5	1107
50 RETURN	1108
END	1109
	1110
	1111
	1112
	1113
	1114
	1115
	1116
	1117
	1118
	1119
	1120
	1121
	1122
	1123
	1124
	1125
	1126
	1127
	1128
	1129
	1130
	1131
	1132
	1133
	1134
	1135
	1136
	1137
	1138

C
C
C
C

SUBROUTINE WPOINT(J, JCONS, L, ANUMB2, SKIP2)
LOGICAL SKIP2

L=0
IDELTA=0
IF(ANUMB2.GE.9.2) IDELTA=12
IF(ANUMB2.GE.8.0) GO TO 200
IF(ANUMB2.GE.6.9) GO TO 201
IF(ANUMB2.GE.5.9) GO TO 202
IF(ANUMB2.GE.5.0) GO TO 203
IF(ANUMB2.GE.4.2) GO TO 204
IF(ANUMB2.GE.3.5) GO TO 205
IF(ANUMB2.GE.2.9) GO TO 206
IF(ANUMB2.GE.2.4) GO TO 207
IF(ANUMB2.GE.1.9) GO TO 208
IF(ANUMB2.GE.1.5) GO TO 209
IF(ANUMB2.GE.1.2) GO TO 210
GO TO 211

200 IF(ANUMB2.LT.9.2) IDELTA=11
201 IF(ANUMB2.LT.8.0) IDELTA=10
202 IF(ANUMB2.LT.6.9) IDELTA=9
203 IF(ANUMB2.LT.5.9) IDELTA=8
204 IF(ANUMB2.LT.5.0) IDELTA=7
205 IF(ANUMB2.LT.4.2) IDELTA=6
206 IF(ANUMB2.LT.3.5) IDELTA=5
207 IF(ANUMB2.LT.2.9) IDELTA=4

208	IF(ANUMB2.LT.2.4) IDELTA=3	1139
209	IF(ANUMB2.LT.1.9) IDELTA=2	1140
210	IF(ANUMB2.LT.1.5) IDELTA=1	1141
211	IF(ANUMB2.LT.1.2) IDELTA=0	1142
	IDELTA=13-IDELTA	1143
	IF(J.EQ.5) GO TO 50	1144
	IF(SKIP2) GO TO 41	1145
	IF(JCGNS.EQ.1)GOTO 1	1146
	IF(JCONS.EQ.10)GOTO 10	1147
	IF(JCONS.EQ.100)GOTO 100	1148
	IF(JCONS.EQ.1000)GOTO 1000	1149
	IF(JCONS.EQ.10000)GOTO 10000	1150
1	L=73+IDELTA	1151
	GO TO 40	1152
10	L=86+IDELTA	1153
	GO TO 40	1154
100	L=99+IDELTA	1155
	GO TO 40	1156
1000	L=112+IDELTA	1157
	GO TO 40	1158
10000	L=125	1159
	GO TO 40	1160
41	KCONS=JCONS+1	1161
	SKIP2=.FALSE.	1162
	IF(KCONS.EQ.11)GOTO 11	1163
	IF(KCONS.EQ.101)GOTO 101	1164
	IF(KCONS.EQ.1001)GOTO 1001	1165
	IF(KCONS.EQ.10001)GOTO 10001	1166
11	L=60+IDELTA	1167
	GO TO 40	1168
101	L=47+IDELTA	1169
	GO TO 40	1170
1001	L=34+IDELTA	1171
	GO TO 40	1172
50	L=125	1173
	GO TO 40	1174
10001	L=34	1175
40	RETURN	1176
	END	1177
C		1178
C		1179
C		1180
C		1181
	SUBROUTINE SPOINT(I,ANUMB1,LESS,ICONS,L,PCINT,SKIP1,NO,XA,XB,STAR)	1182
	DIMENSION XA(2000),XB(2000)	1183
	INTEGER POINT(130,100),DASH,BLANK,STAR,Q	1184
	INTEGER STAR1,STAR2	1185
	DATA STAR1/1H0/,STAR2/1H+/	1186
	DATA PLANK/1H /,DASH/1H-/, Q/1H1/	1187
	LOGICAL SKIP1 ,LESS	1188
	AKEEP=ANUMB1	1189
	ANUMB1=ARS(AKEEP)	1190
	KOOL=0	1191
	IF(I.EQ.5)GO TO 1000	1192
	IF(ANUMB1.GE.7.1)KOOL=7	1193
	IF(ANUMB1.GE.5.0)GO TO 1011	1194
	IF(ANUMB1.GE.4.0)GO TO 1111	1195
	IF(ANUMB1.GE.3.1) GO TO 1211	1196
	IF(ANUMB1.GE.2.3)GOTO1311	1197
	IF(ANUMB1.GE.1.6) GO TO 1411	1198

IF(ANUMB1.GE.1.0)GO TO 1511	1199
GO TO 4011	1200
1011 IF(ANUMB1.LT.7.1)KOOL=6	1201
1111 IF(ANUMB1.LT.5.0)KOOL=5	1202
1211 IF(ANUMB1.LT.4.0)KOOL=4	1203
1311 IF(ANUMB1.LT.3.1)KOOL=3	1204
1411 IF(ANUMB1.LT.2.3)KOOL=2	1205
1511 IF(ANUMB1.LT.1.6)KOOL=1	1206
4011 NCRMNT=8-KOGL	1207
IF(LESS)GO TO 40	1208
IF(SKIP1)GO TO 41	1209
IF(ICON5.EQ.1)GOTO 1	1210
IF(ICON5.EQ.10)GOTO 10	1211
IF(ICON5.EQ.100)GOTO 100	1212
IF(ICON5.EQ.1000)GOTO 1000	1213
1 KOOL=71+KOOL	1214
GO TO 50	1215
10 KOOL=64+KOOL	1216
GO TO 50	1217
100 KOOL=57+KOOL	1218
GO TO 50	1219
1000 KOOL=50+KOOL	1220
GO TO 50	1221
41 LCONS=ICON5+1	1222
SKIP1=.FALSE.	1223
IF(LCONS.EQ.11)GOTO 11	1224
IF(LCONS.EQ.101)GOTO 101	1225
IF(LCONS.EQ.1001)GOTO 1001	1226
11 KOOL=78+KOOL	1227
GO TO 50	1228
101 KOOL=85+KOOL	1229
GO TO 50	1230
1001 KOOL=92+KOOL	1231
50 LESS=.FALSE.	1232
IF(POINT(L,KOOL).EQ.DASH) GO TO 51	1233
IF(POINT(L,KOOL).EQ.C) GO TO 51	1234
14 IF(POINT(L,KOOL).EQ.STAR1) GO TO 15	1235
IF(POINT(L,KOOL).EQ.STAR2) GO TO 15	1236
IF(STAR.EQ.STAR1) GO TO 51	1237
IF(STAR.EQ.STAR2) GO TO 51	1238
IF(POINT(L,KOOL).NE.BLANK) GO TO 70	1239
51 POINT(L,KOOL)=STAR	1240
ANUMB1=AKEEP	1241
GO TO 70	1242
15 L=L+1	1243
IF(L.EQ.130) GO TO 51	1244
GO TO 14	1245
40 IF(SKIP1) GO TO 42	1246
IZ=ICON5+2	1247
IF(IZ.EQ.3)GOTO 3	1248
IF(IZ.EQ.12)GOTO 12	1249
IF(IZ.EQ.102)GOTO 102	1250
IF(IZ.EQ.1002)GOTO 1002	1251
3 KOOL=21+NCRMNT	1252
GO TO 50	1253
12 KOOL=28+NCRMNT	1254
GO TO 50	1255
102 KOOL=35+NCRMNT	1256
GO TO 50	1257
1002 KOOL=42+NCRMNT	1258

GO TO 50	1259
42 IY=ICONS+3	1260
SKIPI=.FALSE.	1261
IF(IY.EQ.13)GOTO 13	1262
IF(IY.EQ.103)GOTO 103	1263
IF(IY.EQ.1003)GOTO 1003	1264
IF(IY.EQ.10003)GOTO 10003	1265
13 KOOL=14+NCRMNT	1266
GO TO 50	1267
103 KOOL=7+NCRMNT	1268
GO TO 50	1269
1003 KOOL=NCRMNT	1270
GO TO 50	1271
10003 POINT(L,1)=STAR	1272
GO TO 70	1273
70 RETURN	1274
END	1275
C	1276
C	1277
C	1278
C	1279
SUBROUTINE PREPAR(POINT)	1280
INTEGER POINT(130,100),PRT	1281
DATA PRT /6/	1282
WRITE(6,1)	1283
1 FORMAT(1H1)	1284
WRITE(6,11)	1285
11 FORMAT(///)	1286
WRITE(PRT,14)	1287
14 FORMAT(75X,8HLOG PLOT,/58X,42HCOMPLEX FREQUENCY PLANE,LEFT HAND CU	1288
*ADRANT,/75X,9H(RAD/SEC))	1289
WRITE(6,12)	1290
12 FORMAT(32X,5H10000,9X,4H1000,9X,3H100,10X,2H10,12X,1H1,11X,2H.1,10	1291
*X,14H.01<---J-OMEGA,/ 18X,11HMINUS SIGMA)	1292
13 FORMAT(32X,5H10000,9X,4H1000,9X,3H100,10X,2H10,12X,1H1,11X,2H.1,10	1293
*X,14H.01<---J-OMEGA,/ 18X,11H PLUS SIGMA)	1294
DO 50 I=1,130	1295
50 POINT(I,50)=POINT(I,51)	1296
WRITE(PRT,10)POINT	1297
10 FORMAT(1X,130A1)	1298
WRITE(6,13)	1299
WRITE(PRT,15)	1300
15 FORMAT(75X,8HLOG PLOT,/58X,43HCOMPLEX FREQUENCY PLANE,RIGHT HAND Q	1301
*UADRANT,/75X,9H(RAD/SEC))	1302
RETURN	1303
END	1304
C	1305
C	1306
C	1306
C	1307
C	1308
SUBROUTINE WRITIT(XA,XB)	1309
DIMENSION XA(2000),XB(2000)	1310
WRITE(6,1)	1311
1 FORMAT(1H1)	1312
WRITE(6,11)	1313
11 FORMAT(5X,48HTHE FOLLOWING ROOTS ARE PLOTTED ON THE LOG PLOT,/	1314
15X,99HROOTS AT THE ORIGIN ARE NOT PRINTED OR PLOTTED, ROOTS ON THE	1315
* J-OMEGA AXIS ARE NOT PLOTTED.	1316
2,//16X,5HSIGMA,25X,7HJ-OMEGA,//)	1317

DO 77 IZO=1,2000	1318
XC=XB(IZO)	1319
IF(XC.GT.0.0) XC=-XC	1320
IF(XC.NE.0.0) GO TO 22	1321
20 IF(XA(IZO))22,40,22	1322
40 K1=IZO	1323
K2=IZO+12	1324
DO 50 K=K1,K2	1325
IF(XA(K).NE.0.0) GO TO 77	1326
50 IF(XB(K).NE.0.0) GO TO 77	1327
GO TO 60	1328
22 WRITE(6,10)XA(IZO),XC	1329
10 FORMAT(5X,F20.9,10X,F20.9)	1330
77 CONTINUE	1331
60 RETURN	1332
END	1333
C	1334
C	1335
SUBROUTINE FREQRS(SIGMAX,OMEGAY)	1336
DIMENSION SAVE1(100,100),SAVE2(100,100),ACOEFF(100),BCOEFF(100),	1337
* AAAAAA(99),BBBBBB(99),SIGMAX(2000),OMEGAY(2000),DEC(9),	1338
* OMEGAS(55),OMEGA(17),WLOG(3000),PHILIN(3000),MDBLIN(3000)	1339
* ,PHIPLN(3000),MPDBLN(3000),ALIM(7),BIGSEM(3000),	1340
* EMSUBS(3000)	1341
REAL MDBLIN,MPDBLN	1342
INTEGER FRQEXP,IEXP(4)	1343
INTEGER POINT(130,100)	1344
COMMON/FRFEK/ACOEFF,BCOEFF	1345
COMMON/INFG2/ALIM	1346
COMMON/INFO3/OMEGA	1347
COMMON/INFO6/EMSUBS	1348
COMMON/INFO7/ICOUNT	1349
COMMON SAVE1,SAVE2,POINT	1350
EQUIVALENCE (WLOG(1),SAVE1(1)),(PHILIN(1),SAVE2(1)),	1351
1 (SAVE1(3001),MDBLIN(1)),(SAVE2(3001),PHIPLN(1)),	1352
2 (SAVE1(6001),MPDBLN(1)),(SAVE2(6001),BIGSEM(1)),	1353
3 (ALIM(1),XSMALL),(AAAAAA(1),ACOEFF(2)),(BBBBBB(1),	1354
4 BCOEFF(2))	1355
TENS(X)=X*10.0	1356
DATA(DEC(I),I=1,9)/1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0/	1357
DATA LET1/4H CB/,LET2/4H CEG/	1358
K1=0	1359
KOUNT=1	1360
DO 61 I=1,4	1361
61 IEXP(I)=0	1362
DO 60 I=1,3000	1363
62 WLOG(I)=0.0	1364
PHILIN(I)=0.0	1365
BIGSEM(I)=0.0	1366
EMSUBS(I)=0.0	1367
MDBLIN(I)=0.0	1368
PHIPLN(I)=0.0	1369
60 MPDBLN(I)=0.0	1370
WRITE(6,51)	1371
READ(5,19)XSMALL,DEGREE,DBLIM	1372
READ(5,12)FRQEXP	1373
IF(FRQEXP.NE.0)READ(5,12)(IEXP(K),K=1,4)	1374
12 FORMAT(4I10)	1375
19 FORMAT(3E10.0,50X)	1376
DO 90 I=2,7	1377

90	ALIM(I)=TENS(ALIM(I-1))	1378
	DO 70 J=1,6	1379
	DO 71 I=1,9	1380
	K=K1+I	1381
71	OMEGAS(K)=DEC(I)*ALIM(J)	1382
70	K1=K1+9	1383
	OMEGAS(55)=ALIM(7)	1384
	K=1	1385
	CALL FIGURR(55,OMEGAS,KOUNT)	1386
41	Y=ABS(OMEGAY(K))	1387
	X=ABS(SIGMAX(K))	1388
	IF(Y.EQ.0.0) GO TO 13	1389
	YOMEGA=Y	1390
	GO TO 20	1391
13	IF(X.EQ.0.0) GO TO 10	1392
	YOMEGA=X	1393
20	CALL SWEEP(YOMEGA)	1394
	CALL FIGURR(17,OMEGA,KOUNT)	1395
	IF(KCUNT.GE.2984)GO TO 40	1396
	GO TO 80	1397
10	KK=K+1	1398
	KI=K+11	1399
	IF(KI.GT.2000) GO TO 40	1400
	DO 11 L=KK,KI	1401
	Y=ABS(OMEGAY(L))	1402
	X=ABS(SIGMAX(L))	1403
	IF(X.NE.0.0.OR.Y.NE.0.0) GO TO 80	1404
11	CONTINUE	1405
	GO TO 40	1406
80	IF(K.GE.ICOUNT) GO TO 40	1407
	K=K+1	1408
	GO TO 41	1409
40	DO 68 I=1,3000	1410
	IF(WLOG(I).EQ.0.0.AND.PHILIN(I).EQ.0.0.AND.PHIPLN(I).EQ.C.0.AND.	1411
	1MDBLIN(I).EQ.0.0.AND.MPDBLN(I).EQ.C.0.AND.RIGSEM(I).EQ.C.0.AND.	1412
	2EMSUBS(I).EQ.0.0) GO TO 69	1413
	JJ=MCD(I,57)	1414
	IF(JJ.EQ.0) WRITE(6,51)	1415
68	WRITE(6,50) WLOG(I),RIGSEM(I),MDBLIN(I),PHILIN(I),EMSUBS(I),	1415
	1MPDBLN(I),PHIPLN(I)	1417
69	WRITE(6,14)	1418
	CALL NEWPLT(DEGREE,WLOG,PHILIN,LET2)	1419
	IF(IEXP(1).EQ.0) GO TO 30	1420
	JJ=IEXP(1)	1421
	DO 34 I=1,JJ	1422
34	CALL EXPLOT(WLOG,3000,PHILIN,3000,DEGREE,LET2)	1423
30	CALL NEWPLT(DBLIM,WLOG,MDBLIN,LET1)	1424
	IF(IEXP(2).EQ.0) GO TO 31	1425
	JJ=IEXP(2)	1426
	DO 35 I=1,JJ	1427
35	CALL EXPLOT(WLOG,3000,MDBLIN,3000,DBLIM,LET1)	1428
31	WRITE(6,15)	1429
	CALL NEWPLT(DEGREE,WLOG,PHIPLN,LET2)	1430
	IF(IEXP(3).EQ.0) GO TO 32	1431
	JJ=IEXP(3)	1432
	DO 36 I=1,JJ	1433
36	CALL EXPLOT(WLOG,3000,PHIPLN,3000,DEGREE,LET2)	1434
32	CALL NEWPLT(DBLIM,WLOG,MPDBLN,LET1)	1435
	IF(IEXP(4).EQ.0) GO TO 33	1436
	JJ=IEXP(4)	1437

DO 37 I=1,JJ	1438
37 CALL EXPLOT(WLOG,3000,MPDBLN,3000,DBLIN,LET1)	1439
51 FORMAT(1H1,5X,5HOMEGA,13X,3H/M/,16X,6H/H/-DB,10X,3HPHI,20X,4H/M/P,	1440
*10X,7H/M/P-DB,14X,5HPHI-P,)	1441
50 FORMAT(2X,1P1E12.5,5X,1P1E12.5,6X,1P1E12.5,5X,1P1E12.5,11X,	1442
*1P1E12.5,3X,1P1E12.5,9X,1P1E12.5)	1443
14 FORMAT(1H1,////////////////////,25X,60HTHE FOLLOWING PLOTS ARE NOR	1444
*MAL FREQUENCY RESPONSE DATA)	1445
15 FORMAT(1H1,////////////////////,25X,36HTHE FOLLOWING PLOTS ARE PCP	1446
*OV DATA)	1447
33 RETURN	1448
END	1449
	1450
	1451
SUBROUTINE FIGURR(KONT,OMEGA,KOUNT)	1452
DIMENSION SAVE1(100,100),SAVE2(100,100)	1453
DIMENSION ACOEFF(100),BCOEFF(100),AAAAAA(99),BBBBBB(99)	1454
DIMENSION WLOG(3000),PHILIN(3000),MDBLIN(3000),PHIPLN(3000),	1455
*MPDBLN(3000),BIGSEM(3000),EMSUBS(3000),OMEGA(KONT)	1456
REAL MDBLIN,MPDBLN	1457
INTEGER POINT(130,100)	1458
COMMON/FREK/ACOEFF,BCOEFF	1459
EQUIVALENCE(AAAAAA(1),ACOEFF(2)),(BBBBBB(1),BCOEFF(2))	1460
COMMON/INFO6/EMSUBS	1461
COMMON SAVE1,SAVE2,POINT	1462
EQUIVALENCE (WLOG(1),SAVE1(1)),(PHILIN(1),SAVE2(1)),	1463
1 (SAVE1(3001),MDBLIN(1)),(SAVE2(3001),PHIPLN(1)),	1464
2 (SAVE1(6001),MPDBLN(1)),(SAVE2(6001),BIGSEM(1))	1465
DO 21 I=1,KONT	1466
CALL FIGURE(OMEGA(I),ACOEFF(1),AAAAAA,EMOFAS,PHIAS)	1467
CALL FIGURE(OMEGA(I),PCOEFF(1),BBBBBB,EMOFBS,PHIBS)	1468
BIGEMM=ABS(EMOFAS/EMOFBS)	1469
BIGSEM(KOUNT)=BIGEMM	1470
WLOG(KOUNT)=OMEGA(I)	1471
PHIEND=PHIAS-PHIBS	1472
IF(PHIEND.LT.-180.) PHIEND=PHIEND+360.	1473
IF(PHIEND.GT. 180.) PHIEND=PHIEND-360.	1474
PHILIN(KOUNT)=PHIEND	1475
PHIEND = PHIEND * 0.0174533	1476
REEL = BIGEMM * COS (PHIEND)	1477
EIMAG= BIGEMM * SIN (PHIEND)	1478
EMSUBP = SQRT ((REEL** 2) + ((OMEGA (I) * EIMAG) ** 2))	1479
EMSUBS(KOUNT)=EMSUBP	1480
QUANZ=OMEGA(I) * EIMAG	1481
PHIPEE=ATAN2(QUANZ,REEL)	1482
PHIPEE = PHIPEE * 57.2957795131	1483
IF(PHIPEE . LT . 0.) PHIPEE = PHIPEE + 360.	1484
IF(PHIPEE.LT.-180.) PHIPEE=PHIPEE+360.	1485
IF(PHIPEE.GT. 180.) PHIPEE=PHIPEE-360.	1486
PHIPLN(KOUNT)=PHIPEE	1487
DEEBEE=20.0*(ALOG10(BIGEMM))	1488
MDBLIN(KOUNT)=DEEBEE	1489
DEEBEP=20.0*(ALOG10(EMSUBP))	1490
MPDBLN(KOUNT)=DEEBEP	1491
KOUNT=KOUNT+1	1492
21 CONTINUE	1493
RETURN	1494
END	1495
	1496
	1497

C
C

C
C

	SUBROUTINE FIGURE(X,Y,Z,EMOFAS,PHIAS)	1498
	DIMENSION Z(99)	1499
	REELAS=Y	1500
	I=2	1501
	A=-1.0	1502
21	REELAS=((Z(I)*(X**I))*A)+REELAS	1503
25	I=I+2	1504
	IF(I.GT.99) GO TO 22	1505
	A=-A	1506
	IF(Z(I).EQ.0.)GO TO 25	1507
	GO TO 21	1508
22	I=1	1509
	A=1.0	1510
	AIMAGA=0.0	1511
24	AIMAGA=(Z(I)*(X**I)*A)+AIMAGA	1512
26	I=I+2	1513
	IF(I.GT.99) GO TO 23	1514
	A=-A	1515
	IF(Z(I).EQ.0.)GO TO 26	1516
	GO TO 24	1517
23	EMOFAS=SQRT((REELAS**2)+(AIMAGA**2))	1518
	PHIAS=ATAN2(AIMAGA,REELAS)	1519
	PHIAS = PHIAS*57.2957795131	1520
	IF(PHIAS.LT.0.) PHIAS=PHIAS+360.	1521
	RETURN	1522
	END	1523
C		1524
C		1525
	SUBROUTINE LOGGER(XX,LIMITW)	1526
	DIMENSION ALIM(7),DIMS(20)	1527
	EQUIVALENCE(WASTE,AXX)	1528
	COMMON/INFO2/ALIM	1529
	TENS(X)=X*10.0	1530
	DATA (DIMS(I),I=1,20) /1.0,1.15,1.3,1.5,1.7,1.95,2.25,2.5,2.85,	1531
	*3.15,3.5,3.95,4.5,4.85,5.4,6.05,6.8,7.7,8.8,9.999999/	1532
	LIMITW=0	1533
	LIMITX=0	1534
	LIMIT2=0	1535
	DO 20 I=1,6	1536
	AA=2*(I-1)	1537
20	IF(XX.LT.ALIM(I+1).AND.XX.GE.ALIM(I)) GO TO 21	1538
	RETURN	1539
21	LIMITX=TFNS(AA)	1540
	WASTE=ABS(XX)	1541
	IF(WASTE.EQ.0.0) RETURN	1542
30	IF(AXX.GE.1.0.AND.AXX.LT.10.0) GO TO 22	1543
	IF(AXX.LT.10.0)GO TO 23	1544
	AXX=AXX/10.0	1545
	GO TO 30	1546
23	AXX=TENS(AXX)	1547
	GO TO 30	1548
22	DO 24 I=1,19	1549
24	IF(AXX.GE.DIMS(I).AND.AXX.LT.DIMS(I+1)) GO TO 25	1550
25	LIMIT2=I	1551
	LIMITW=LIMITX+LIMIT2	1552
	RETURN	1553
	END	1554
C		1555
C		1556
	SUBROUTINE BLANKR (POINT,N1,N2,LIM1)	1557


```

INTEGER POINT(N1,N2),BLANK,DASH,Q
COMMON /INFO4/ Q,BLANK,DASH
DATA BLANK/1H /,DASH/1H-/,Q/1H/
DO 51 M=1,N1
DO 51 N=1,N2
51 POINT(M,N)=BLANK
DO 52 M=1,N1
POINT(M,1)=DASH
52 POINT(M,N2)=DASH
DO 53 M=1,N2
POINT(1,M)=Q
53 POINT(LIM1,M)=Q
RETURN
END

```

C
C

```

SUBROUTINE NEWPLT(X,Y,Z,LET)
INTEGER POINT(130,100),POINTF(120,58),STAR,Q
DIMENSION ALIM(7)
DIMENSION Y(3000),Z(3000)
DIMENSION SAVE1(100,100),SAVE2(100,100)
COMMON /INFO4/ Q,BLANK,DASH
COMMON/INFO2/ALIM
COMMON SAVE1,SAVE2,POINT
EQUIVALENCE(POINT(1,1),POINTF(1,1))
DATA STAR/1H*/,Q/1H/
1 FORMAT(1H1)
CALL BLANKR(POINTF,120,58,120)
DO 10 I=1,3000
IF(Y(I).EQ.0.0) GO TO 11
GO TO 12
11 IF(Z(I).EQ.0.0) GO TO 10
12 CALL LOGGER(Y(I),LIMITW)
IF(LIMITW.EQ.0) GO TO 10
CALL LINEAR(X,LIMITY,Z(I))
IF(LIMITY.EQ.0) GO TO 10
POINTF(LIMITW,LIMITY)=STAR
10 CONTINUE
DO 40 I=20,100,20
DO 40 J=1,58
40 POINTF(I,J)=Q
WRITE(6,1)
30 FORMAT(1X,6(1P1E10.1,1X,3HR/S,6X),1P1E10.1)
J=X
WRITE(6,51)J,LET,(POINTF(I,1),I=1,120)
51 FORMAT(1X,I4,A4,1X,120A1)
WRITE(6,20)((POINTF(I,J),I=1,120),J=2,57)
J=-X
WRITE(6,51)J,LET,(POINTF(I,58),I=1,120)
20 FORMAT(10X,120A1)
WRITE(6,30)ALIM
RETURN
END

```

C
C

```

SUBROUTINE LINEAR(X,LIMITY,Y)
DIMENSION ENCRMT(59)
LIMITY=0
FLIMIT=X*2.
DELTA=FLIMIT/58.

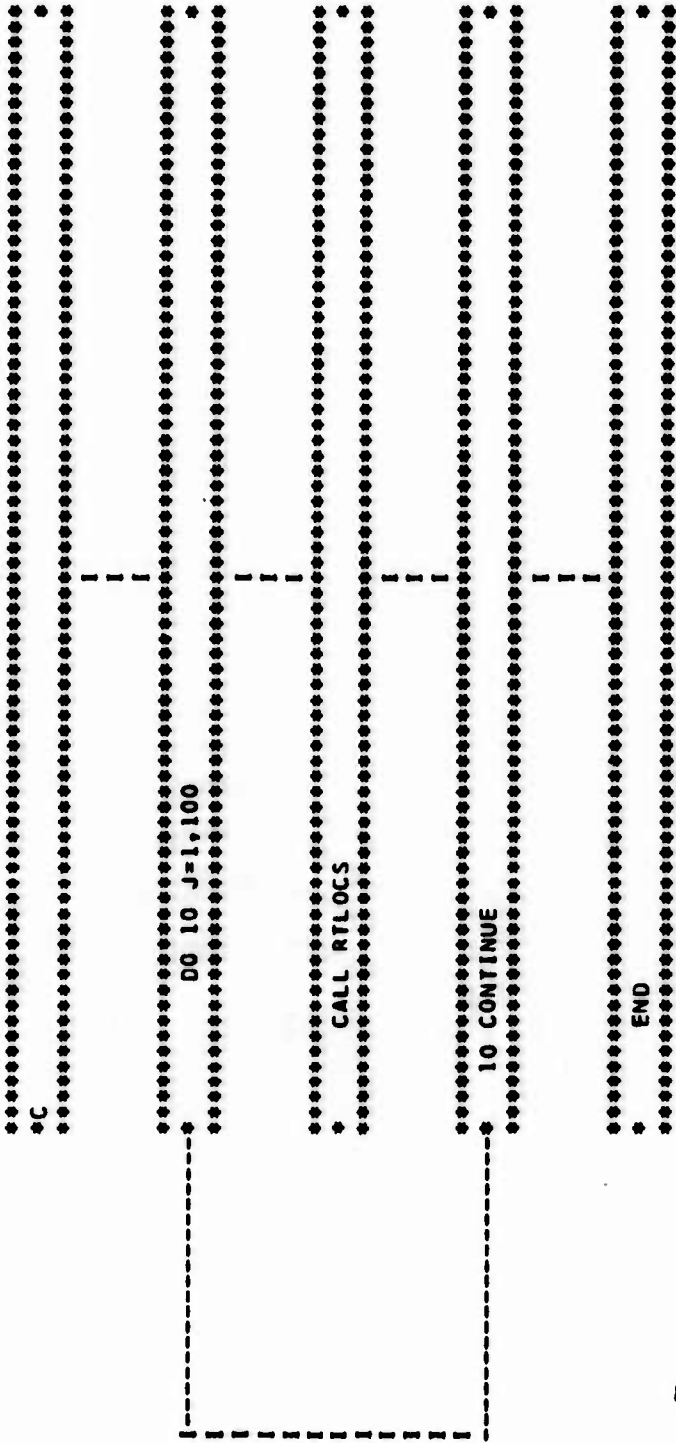
```

ENCRMT(1)=X	1618
DO 10 I=2,58	1619
10 ENCRMT(I)=ENCRMT(I-1)-DELTA	1620
ENCRMT(59)=-X	1621
DO 11 I=1,58	1622
11 IF(Y.LE.ENCRMT(I).AND.Y.GE.ENCRMT(I+1))GO TO 20	1623
GO TO 40	1624
20 LIMITY=I	1625
40 RETURN	1626
END	1627
C	1628
C	1629
SUBROUTINE SWEEPR(XOMEGA)	1630
DIMENSION OMEGA(17),SWEPER(17)	1631
COMMON/INFO3/OMEGA	1632
DATA (SWEPER(I),I=1,17) /C.C1,0.05,0.1,0.2,0.3,0.4,0.5,0.6,0.7,	1633
*C.8,0.9,1.0,2.0,4.0,6.0,8.0,10.0/	1634
DO 70 K=1,17	1635
70 OMEGA(K)= SWEPER(K)*XOMEGA	1636
RETURN	1637
END	1638
C	1639
C	1640

APPENDIX B

FLOW CHART OF SOURCE DECK

(ENTRANCE)



(ENTRANCE)

```
.....
* C
* C
* C
* C
* SUBROUTINE RTLOCS
* DIMENSION X(100), IGA(100), IGB(100), C(100), D(100), AMS(100),
* ISAVE(100), ERA(100), AS(100), RS(100), A(100), B(100), ROOTR(100),
* ZROOT(100), ATK(100), CK(100), CKS(100)
* INTEGER CMENT(20), DENSE, ROTLCS, EXPND, FREQSR, STAR, STAR1, STAR2,
* STAR3, ONE, ZERO, DOT, VEE, AVE, DASH, BLANK, Q
* POINT(130, 100), SPOT(130, 50)
* DATA ONE/IH1/, ZERO/IH0/, DOT/IH./, AVE/IHA/, VEE/IHV/, BLANK/IH /,
* CASH/IH-/, Q/IH/, STAR1/IH0/, STAR2/IH0/, STAR3/IH./
* LOGICAL PRNT
* EQUIVALENCE (SPOT(1,1), POINT(1,1)), (STAR1, ZERO), (STAR3, DOT),
* COMMON/FREK/A,B
* COMMON /INFO4/ Q, BLANK, DASH
* COMMON /INFO7/ ICOUNT
* COMMON SAVE1, SAVE2, POINT
* COMMON /INFO8/ ROTLCS, EXPND, FREQSR
* COMMON /INFO9/ DENSE
* READ(5,105) ROTLCS, EXPND, FREQSR, DENSE
* READ(5,11) CMENT
* WRITE(6,13) CMENT
.....
```

```
.....
* IF(ROTLCS.NE.0.AND.EXPND.NE.0.AND.FREQSR.NE.0) GO TO 37
.....
```

```
.....
* IF(ROTLCS.EQ.0.AND.EXPND.EQ.0) WRITE(6,30)
* IF(FREQSR.EQ.0.AND.EXPND.EQ.0) WRITE(6,31)
* IF(FREQSR.EQ.0.AND.ROTLCS.EQ.0) WRITE(6,32)
* IF(ROTLCS.NE.0.AND.EXPND.NE.0) WRITE(6,33)
* IF(FREQSR.NE.0.AND.EXPND.NE.0) WRITE(6,34)
* IF(FREQSR.NE.0.AND.ROTLCS.NE.0) WRITE(6,35)
.....
```

```
.....
* 18 00277 10U-1,2C00
.....
```

.....
* XA(10U)=0.0
.....

.....
* 277 XB(10U)=0.0
.....

.....
* DO 401 KK=1,100
.....

.....
* DO 401 KJ=1,130
.....

.....
* 401 POINT(KJ, KK)=BLANK
.....

.....
* DO 402 KK=1,130
.....

.....
* POINT(KK, 51)=DASH
.....

.....
* 402 POINT(KK, 50)=DASH
.....

.....
* DO 403 KK=1,100
.....

```

.....
DO 403 KJ=34,125,13
.....
.....
403 POINT(KJ, KK)=Q
.....

.....
DO 404 KJ=34,124
.....
.....
DO 404 KK=7, 93,7
.....
.....
IF(KK.EQ.49) KK=58
.....
.....
404 POINT(KJ, KK)=DASH
.....

.....
DO 405 KK=7, 93,7
.....
.....
IF(KK.EQ.49) KK=58
.....
.....
405 POINT(25, KK)=ONE
.....

```

```

.....
DO 406 KK=26,28
.....

.....
406 POINT(KK,7)=ZERO
.....

.....
DO 407 KK=26,27
.....

.....
407 POINT(KK,14)=ZERO
.....

.....
POINT(26,21)=ZERO
.....

.....
DO 408 KK=35,65,7
.....

.....
IF (KK.EQ.49) KK=58
.....

.....
408 POINT(24, KK)=DOT
.....

.....
POINT(25,42)=ZERO
POINT(25,58)=ZERO
POINT(26,42)=ONE
POINT(26,58)=ONE
POINT(26,79)=ZERO
POINT(26,86)=ZERO
.....

.....
POINT(27,86)=ZERO
.....

```


.....
DO 409 KK=26,28
.....

.....
409 POINT(KK,93)=ZERO
.....

.....
DO 420 KK=1,100
.....

.....
IF(KK.EQ.4) KK=98
.....

.....
420 POINT(26, KK)=0
.....

♀

.....
POINT(26,4)=VEE
POINT(26,97)=AVE
.....

.....
DO 7601 JACKIE=1,100
.....

.....
DO 7601 JCANN=1,100
.....

.....
SAVE1(JACKIE,JOANN)=0.0
.....

1 1
1 1
1 1
1 1

.....
* 7681 SAVE2(JACKIE,JOANN)=C.3
.....

.....
* NO=1
* ICOUNT=0
* MI=C
* JZ0=1
* K1=1
* III=1
* 140 FORMAT(10X,7HDELTA=,110,5X,19HPOLY,ADDED IN A(S)=,110,5X,19HPOLY,
* *ADDEC IN B(S)=,110,5X,9HPR0B.NO.=,110)
.....

1 0<-----0

.....
* 12 READ (5,101) N,IA,IB,IPROB
.....

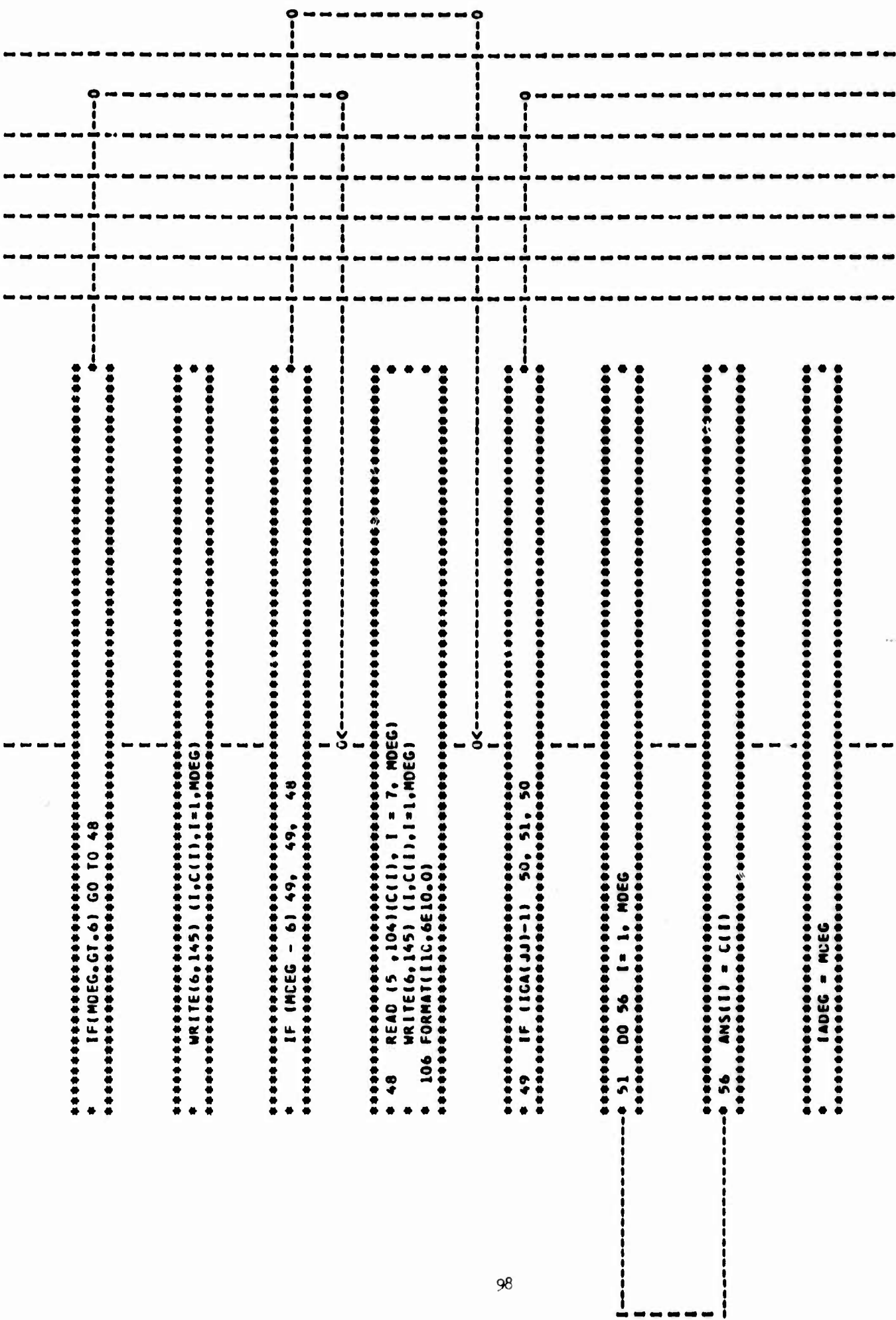
.....
* IF(IB.NE.0) GO TO 10
.....

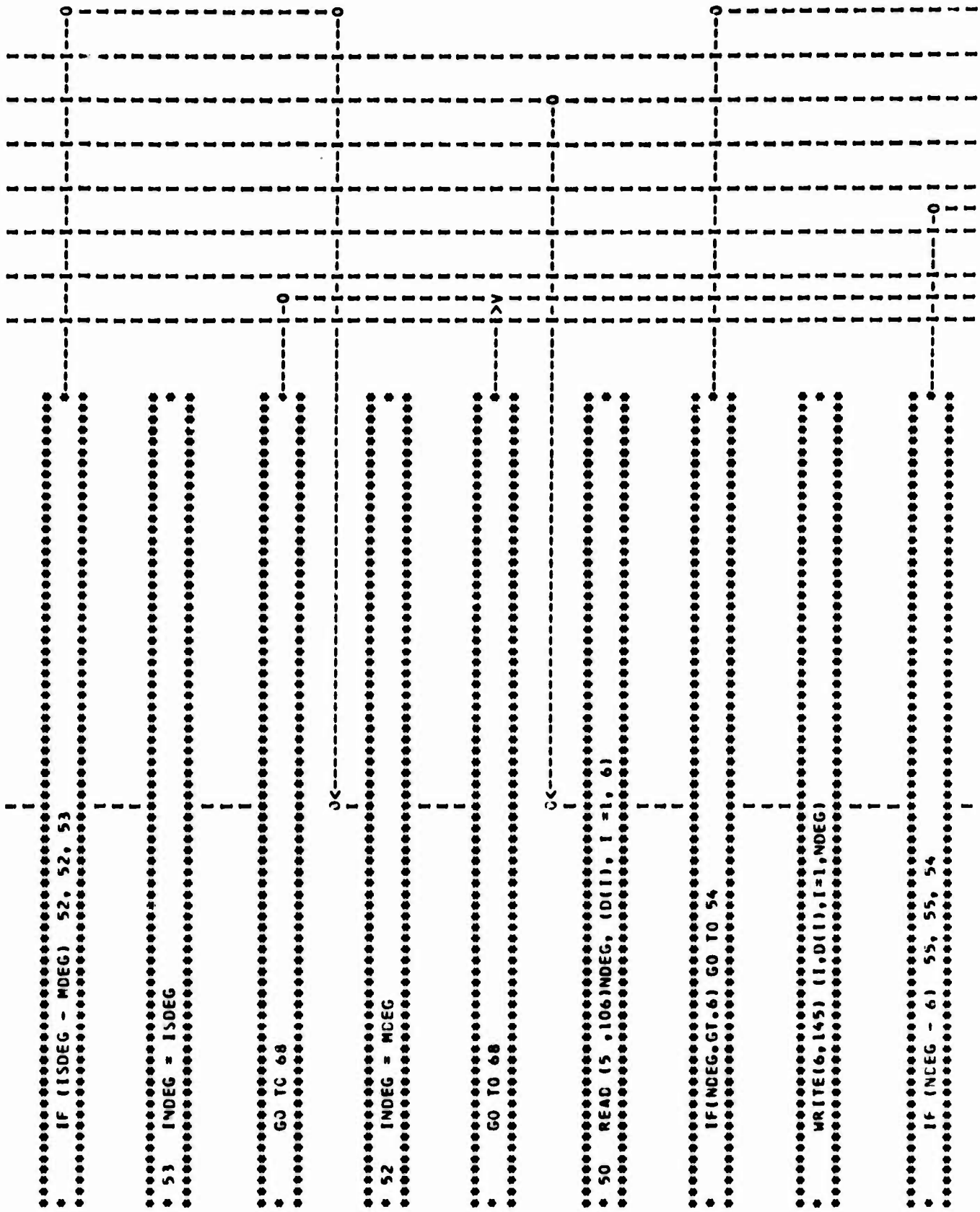
.....
* IF(IA.NE.0) GO TO 10
.....

.....
* IF(N.NE.0) GO TO 10
.....

.....
* IF(IPROB.NE.0) GO TO 1016
.....

.....
* 101 FORMAT (7I10)
.....





```

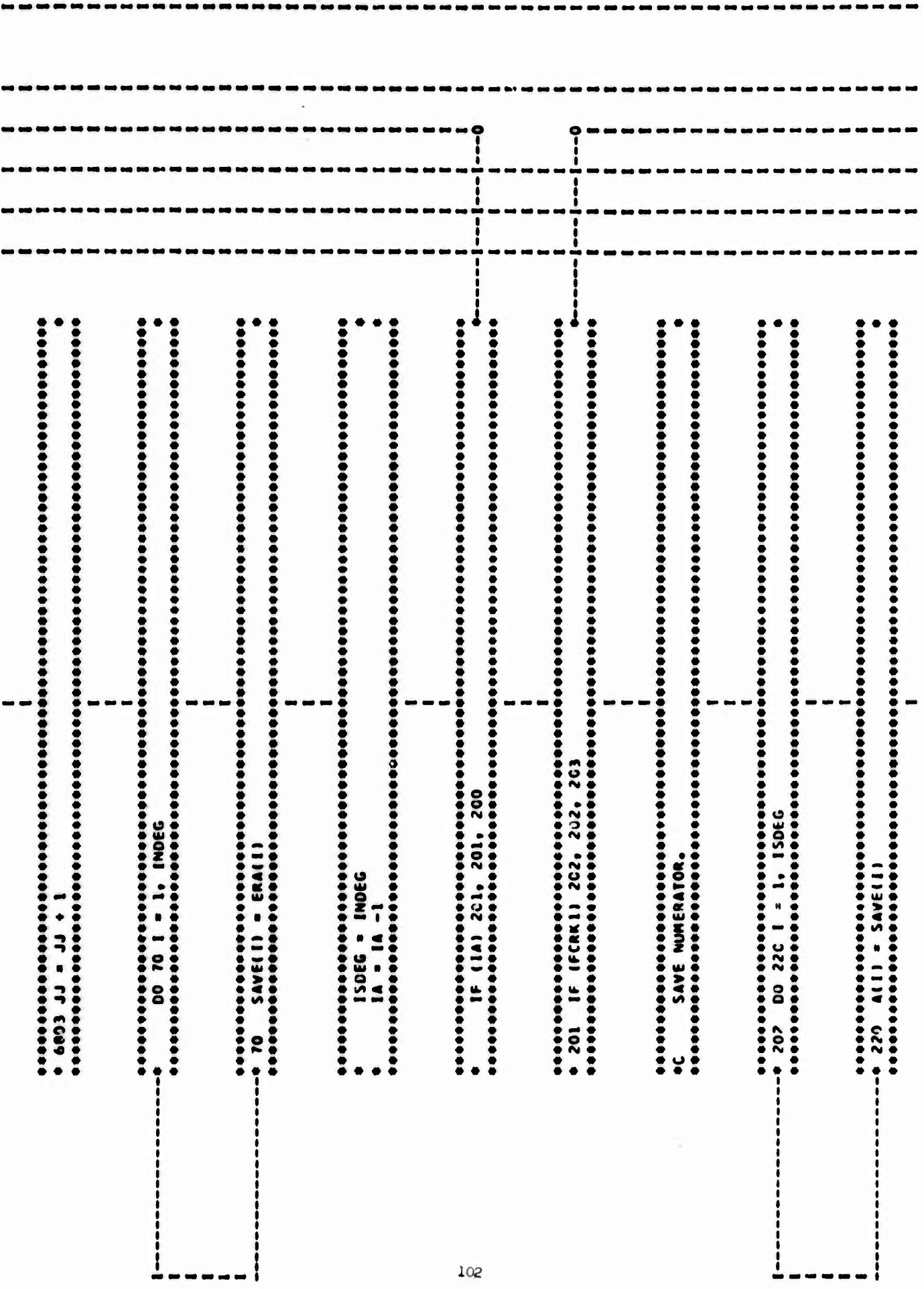
.....
* 54 REAC (5,104)(D(I), I=7, NDEG)
* WRITE(6,145) (I,D(I),I=1,NDEG)
.....
* 55 IADEG = NDEG + MDEG - 1
* CALL POLMRY (C,MDEG,D,NDEG,ANS)
* IGA (JJ) = IGA(JJ) - 1
.....
* IF (IGA(JJ) - 1) 65, 65, 64
.....
* 64 DO 60 I = 1, IADEG
.....
* 60 C(I) = ANS(I)
.....
* MDEG = IADEG
.....
* GO TO 50
.....
* 65 IF (ISDEG - IADEG) 66,66,67
.....

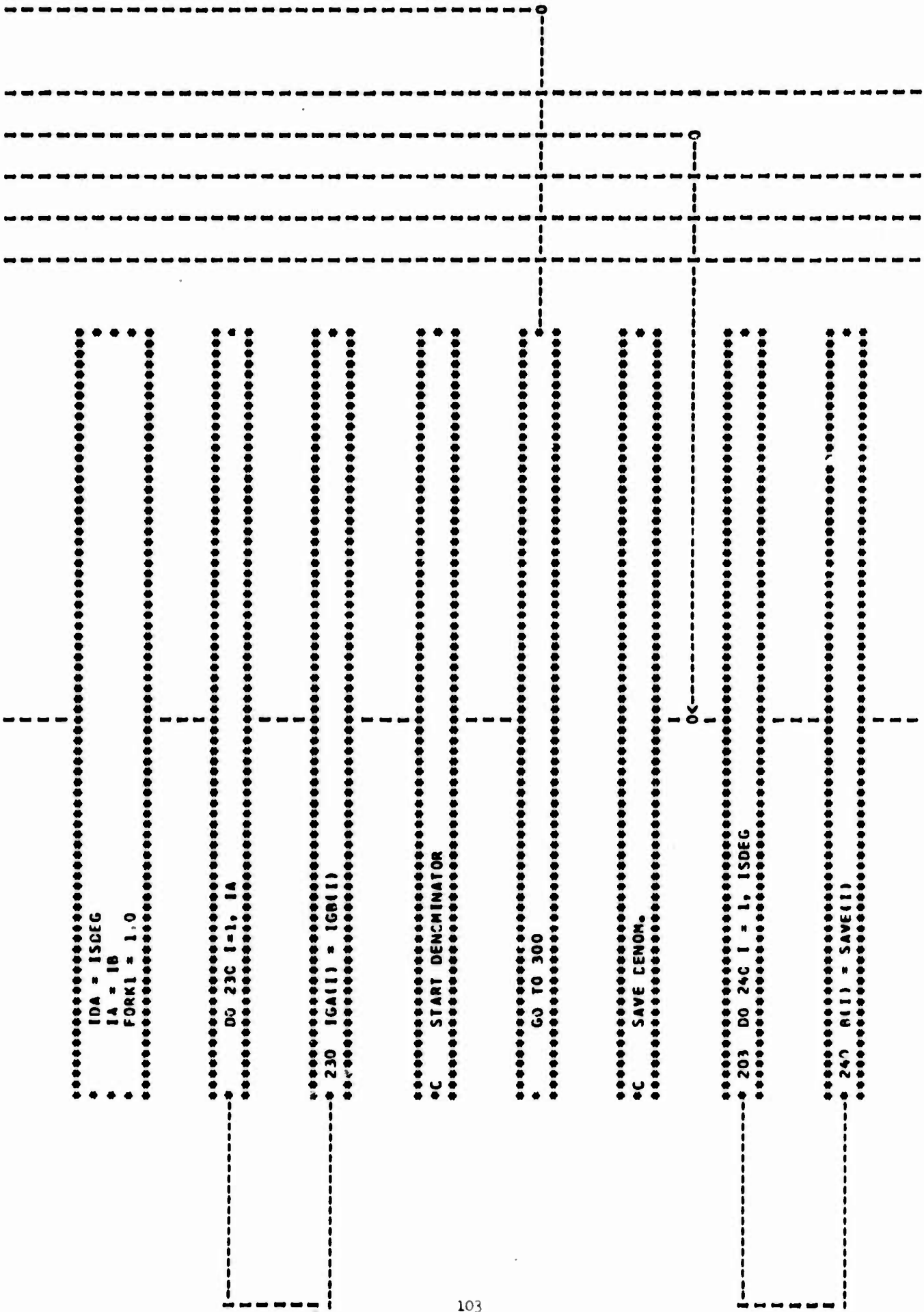
```

```

I
I
.....
* 66 INDEG = IADEG
.....
I
I
.....
* GO TO 68
.....
I
I
.....
OK-----I
I
.....
* 67 INDEG = ISDEG
.....
I
I
.....
OK-----I
I
.....
* 68 CALL POLACD (SAVE, ISDEG, ANS, IADEG, ERA)
* 145 FORMAT(10X, 2MC(1,3,2H)-, F20.10)
* WRITE(6,1)
* 1 FORMAT(1H1//)
.....
I
I
.....
OK-----I
I
.....
* 6800 IF (ERA(INDEG))6801, 6802, 6803
.....
I
I
.....
OK-----I
I
.....
* 6802 INDEG = INDEG - 1
.....
I
I
.....
* 6801 IF (INDEG) 6801, 6801, 6800
.....
I
I
.....
* 6801 INDEG = 1
.....
OK-----I
I
.....

```



```

.....
*   IDB = ISDEG
*   WRITE (6,109)
*   FORMAT (1X,41MCOEFFICIENTS ARE GIVEN IN ASCENDING ORDER////)
.....
OK
.....
*   339 IF (A(IDA)) 340, 341, 340
.....
*   341 IDA = IDA - 1
.....
*   IF (IDA) 345, 345, 339
.....
*   345 WRITE (6,120)
*   120 FORMAT (1H0,10X,20HPOLYNOMIAL A IS ZERO//)
*   FORK2 = 1.0
.....
*   GO TO 410
.....
OK
.....
*   340 IF (ICA - 2) 346, 347, 335
.....
*   346 WRITE(6,121)A(1)
*   STAR=STAR1
*   PRNT=.TRUE.
*   121 FORMAT (1H0,10X,20HPOLYNOMIAL A IS A CONSTANT =,1PIE16.7//)
.....

```

```

.....
* GJ TO 41C
.....
OK-----0
.....
* 347 ROOT = - A(1) / A(2)
* WRITE (6,133)A(1), A(2)
* 133 FORMAT (10X,21THE COEFFICIENTS OF A/1P2E2C.7)
* WRITE (6,122)ROOT
* STAR=STAR1
* PRNT=.TRUE.
* ANUMB1=ROOT
* ANUMB2=0.0
* CALL PLOT1(SAVE1,SAVE2,ANUMB1,ANUMB2,PCINT,XA,XB,MI,PRNT,STAR,III
*,NO)
* 122 FORMAT (10X,10X,23HROOT OF POLYNOMIAL A IS,1P1E16.7//)
.....
* GO TO 41C
.....
.....
*C WRITE POLYS
.....
OK-----0
.....
* 335 IJ1A =IDA -1
* WRITE (6,107)IDIA,(A(1),I=1,IDA)
* K = ICA
.....
.....
* DO 80C I = 1,IDA
.....
* AS(I) = A(K)
.....

```

```

.....
* 800 K = K-1
.....

.....
* IDP2A=IDA *2
* ID2A= 2 *IDIA
* CALL MULLER (AS, IDIA, ROOTR, ROOTI)
.....

.....
DO 805 I = 1, IDIA
.....

.....
* SAM = 100. * AMAXI(ABS(ROOTR(I)), ABS(ROOTI(I)))
* IF (SAM * ABS(ROOTR(I)).EQ. SAM) ROOTR(I)= 0.0
* IF (SAM * ABS(ROOTI(I)).EQ. SAM) ROOTI(I)= 0.0
.....

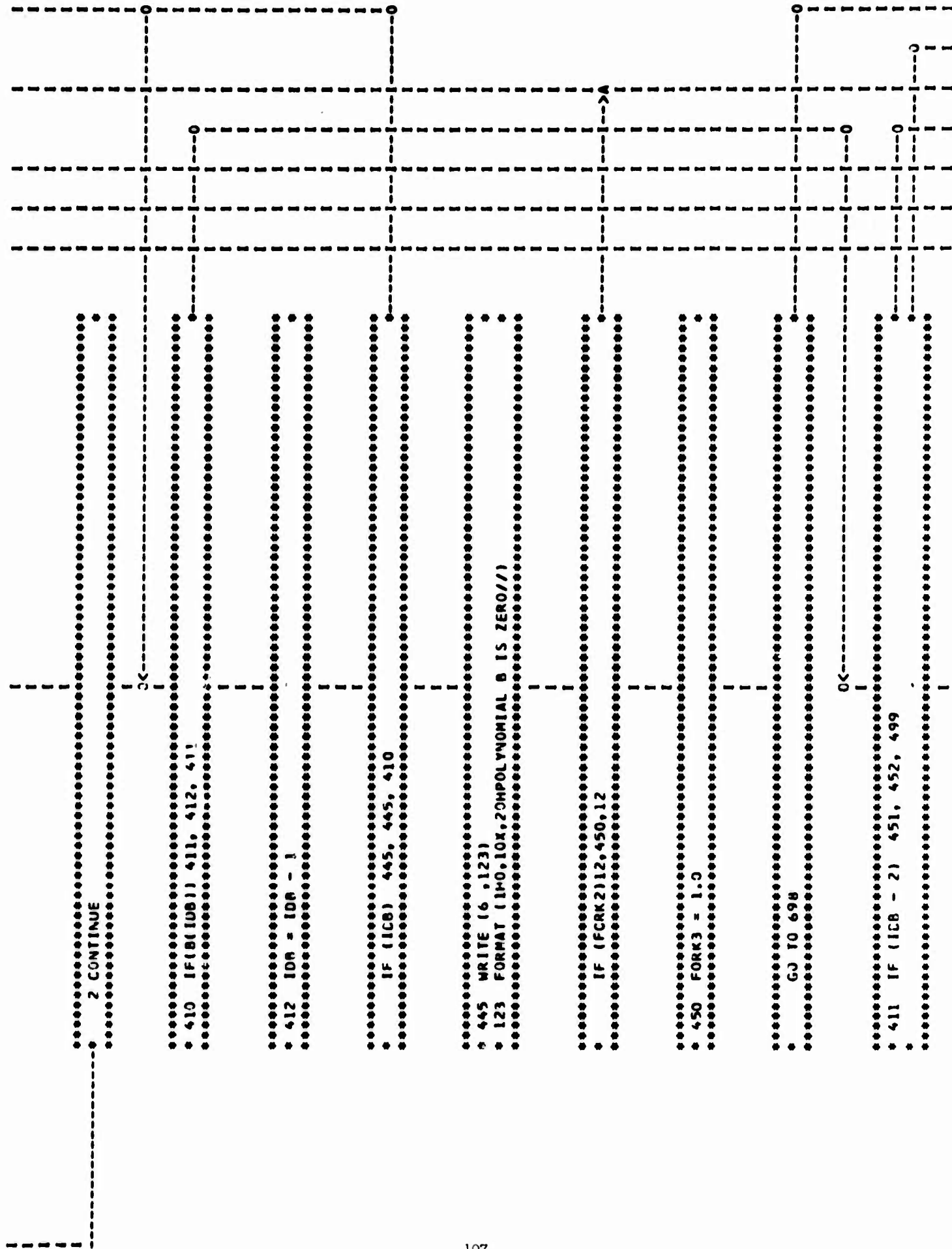
.....
* 805 CONTINUE
.....

.....
* 400 WRITE (6,111) (ROOTR(I), ROOTI(I), I=1, IDIA)
* CALL ERCHEK(RCOTI, IDIA)
* PRINT=TRUE.
* STAR=STAR1
.....

.....
DO 2 III=1, IDIA
.....

.....
* ANUMB1=ROOTR(III)
* ANUMB2=ABS(ROOTI(III))
* CALL PLOTER(SAVE1, SAVE2, ANUMB1, ANUMB2, PCINT, XA, XB, MI, PRNT, STAR, III,
* , NO)
.....

```



```

*****
* 451 WRITE (6,124)B(108)
* STAR=STAR2
* PRNT=.FALSE.
* STAR=STAR3
* 124 FORMAT (1H0,10X,28HPOLYNOMIAL B IS A CONSTANT =,1P1E16.7//)
*****
*****
* GO TO 698
*****
*****
* 452 ROOT = -B(1) / B(2)
* WRITE (6,134)B(1), B(2)
* 134 FORMAT (1CX,21HTHE COEFFICIENTS OF B/1P2E2C.7)
* WRITE (6,125)ROOT
* STAR=STAR2
* PRNT=.TRUE.
* ANUMB1=ROOT
* ANUMB2=0.C
* CALL PLOTTER(SAVE1,SAVE2,ANUMB1,ANUMB2,PCINT,XA,XB,MI,PRNT,STAR,III*
* ,NO)
* PRNT=.FALSE.
* STAR=STAR3
* 125 FORMAT (1H0,10X,23HROOT OF POLYNOMIAL B IS,1P1E16.7//)
*****
*****
* GO TO 698
*****
*****
* 107 FORMAT (1CX,42HTHE COEFFICIENTS OF POLYNOMIAL A (ORDER = 13,1M)/ (
* 11P6E2C.7))
*****
*****
* 499 ID18 = 10R -1
* WRITE (6,109)ID18,(B(I),I=1,108)
* 108 FORMAT (///10X,42HTHE COEFFICIENTS OF POLYNOMIAL B (ORDER = 13,1M*
* 1)/ (1P6E2C.7))
* K = ICB
*****

```

```

*****
* DO 801 I = 1, IDB
*****
      I
      I
      I
*****
*
* RS(I) = R(K)
*****
      I
      I
      I
*****
* 801 K = K-1
*****
      I
      I
      I
*****
* IDP28 = IDB * 2
* ID28 = 2 * ID18
* CALL MULLER (BS, ID18, ROOTR, ROOTI)
*****
      I
      I
      I
*****
* DO 806 I = 1, ID18
*****
      I
      I
      I
*****
* SAM = 100. * AMAX1(ABS(ROOTR(I)), ABS(ROOTI(I)))
* IF (SAM + ABS(ROOTR(I)).EQ. SAM) RCCTR(I) = 0.0
* IF (SAM + ABS(ROOTI(I)).EQ. SAM) RCOTI(I) = 0.0
*****
      I
      I
      I
*****
* 806 CONTINUE
*****
      I
      I
      I
*****
* 500 WRITE (6,112)(ROOTR(I), RCOTI(I), I = 1, ID18)
* CALL ERCHK(RCOTI, ID18)
* STAR = STAR2
* PRNT = .TRUE.
*****
      I
      I
      I
*****

```



```

.....
* MSHEET = 5
*C START K CALCULATIONS
.....

.....
* IF (N) 702,702,533
.....

.....
* 533 DO 55C I= 1, N
.....

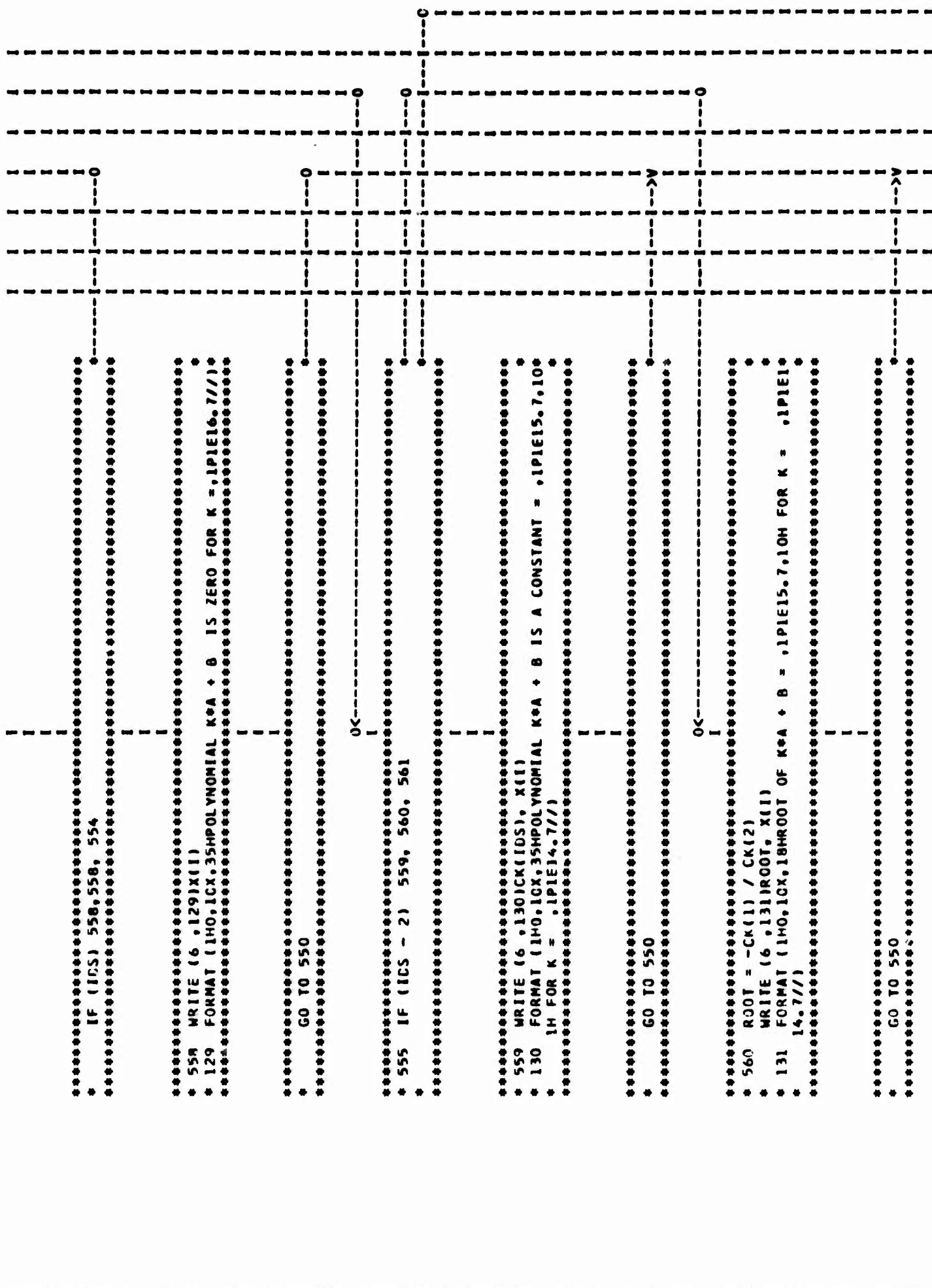
.....
* DO 541 J= 1, IDA
.....

.....
* 541 ATK(J) = X(I) * A(J)
.....

.....
*C COMPUTE ROOTS OF K * A * B
* IDC= MAX0( IDA, IDB)
* CALL POLACD (ATK,IDA,B,IDB,CK)
* IDS = IDC
.....
*C
.....
* 554 IF (CK(IDS))555, 557, 555
.....

.....
* 557 IDS = IDS - 1
.....

```



```

*****
* IF (IDS) 558,558, 554
*****

```

```

*****
* 558 WRITE (6 ,129)X(1)
* 129 FORMAT (1H0,1CX,35HPOLYNOMIAL K*A + B IS ZERO FOR K = ,1P1E16.7//)
*****

```

```

*****
* GO TO 550
*****

```

```

*****
* 555 IF (ICS - 2) 559, 560, 561
*****

```

```

*****
* 559 WRITE (6 ,130)CK(1DS), X(1)
* 130 FORMAT (1H0,1CX,35HPOLYNOMIAL K*A + B IS A CONSTANT = ,1P1E15.7,10
* 1H FOR K = ,1P1E14.7//)
*****

```

```

*****
* GO TO 550
*****

```

```

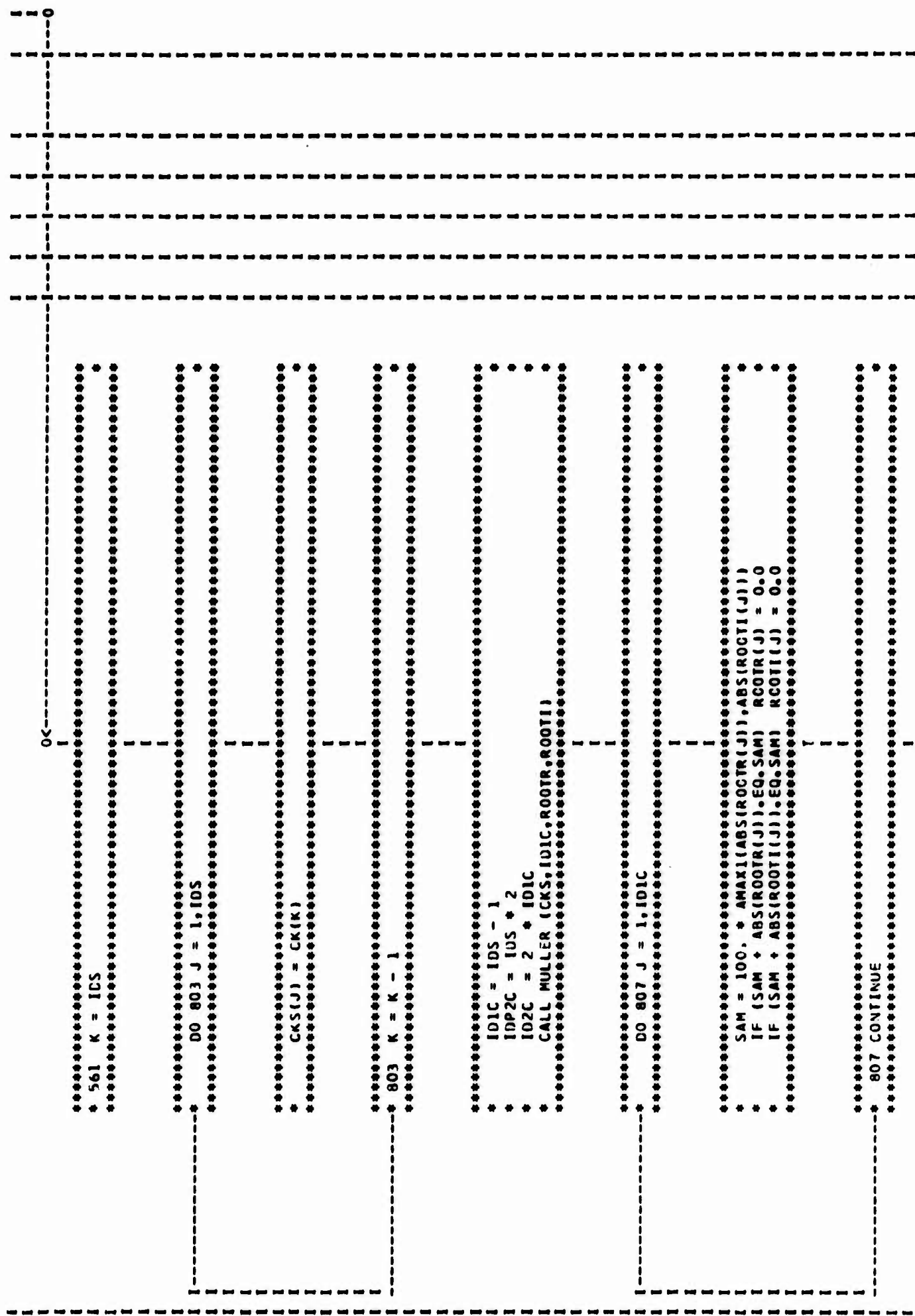
*****
* 560 ROOT = -CK(1) / CK(2)
* WRITE (6 ,131)ROOT, X(1)
* 131 FORMAT (1H0,1CX,18HROOT OF K*A + B = ,1P1E15.7,10H FOR K = ,1P1E1
* 14.7//)
*****

```

```

*****
* GO TO 550
*****

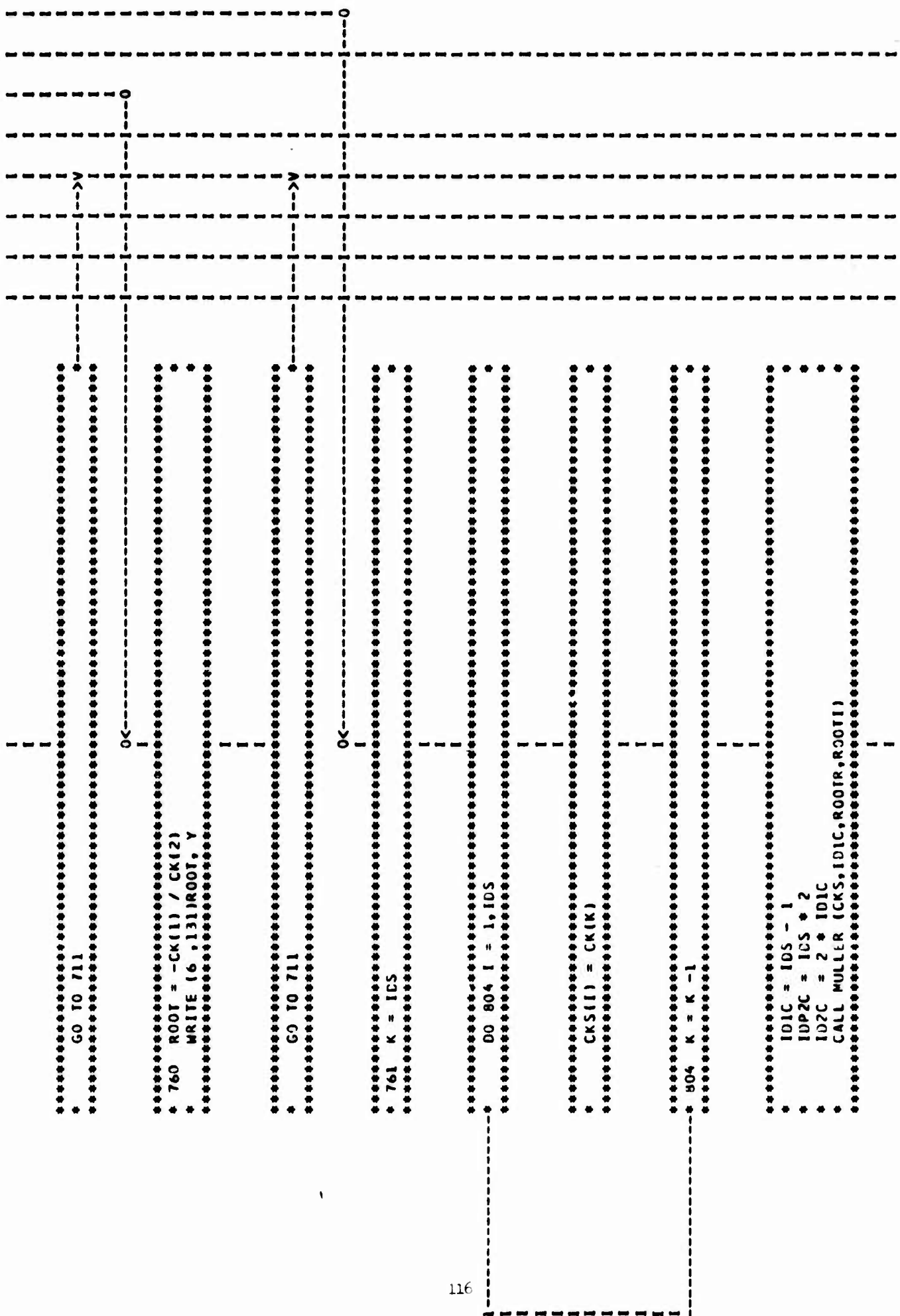
```

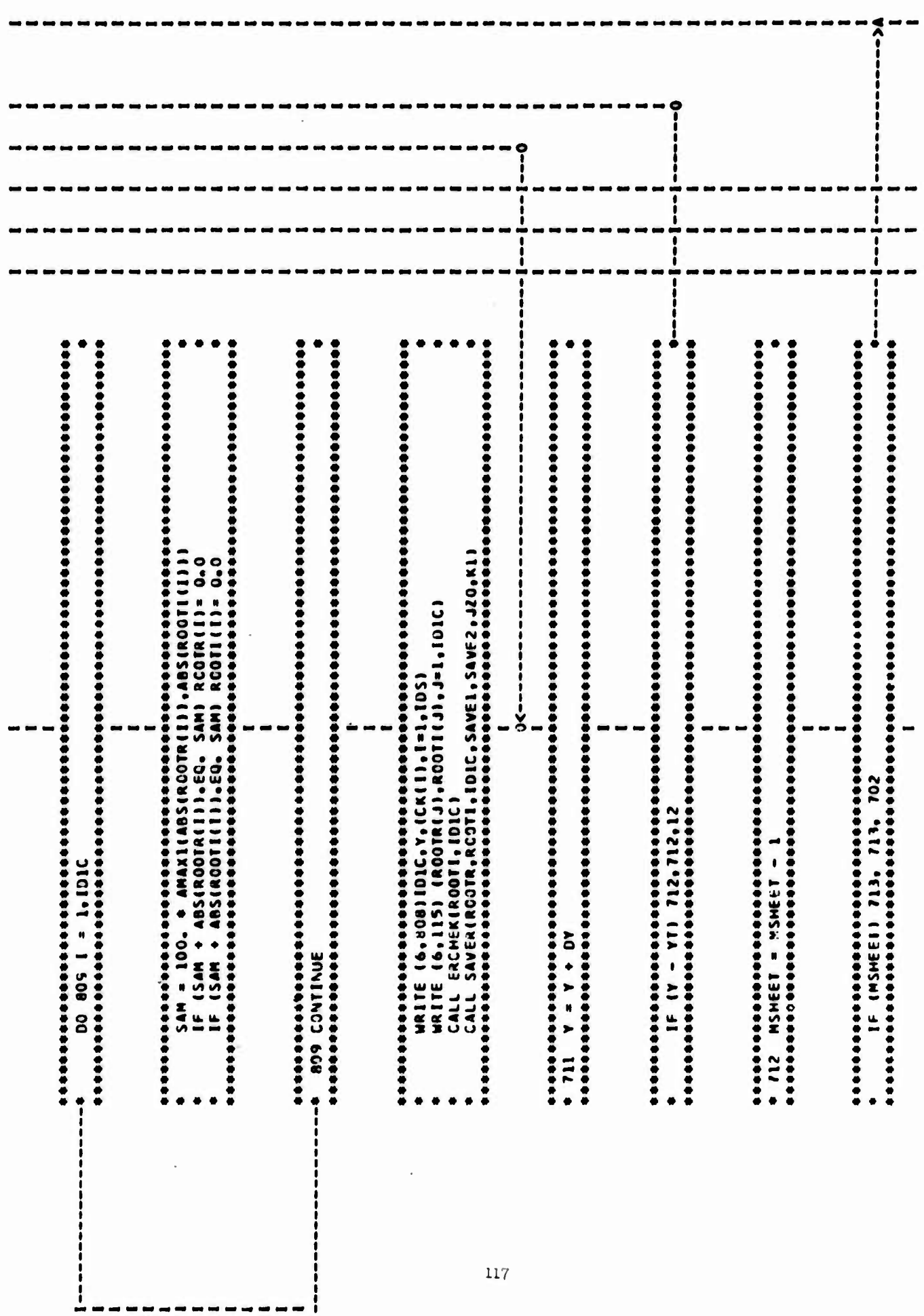


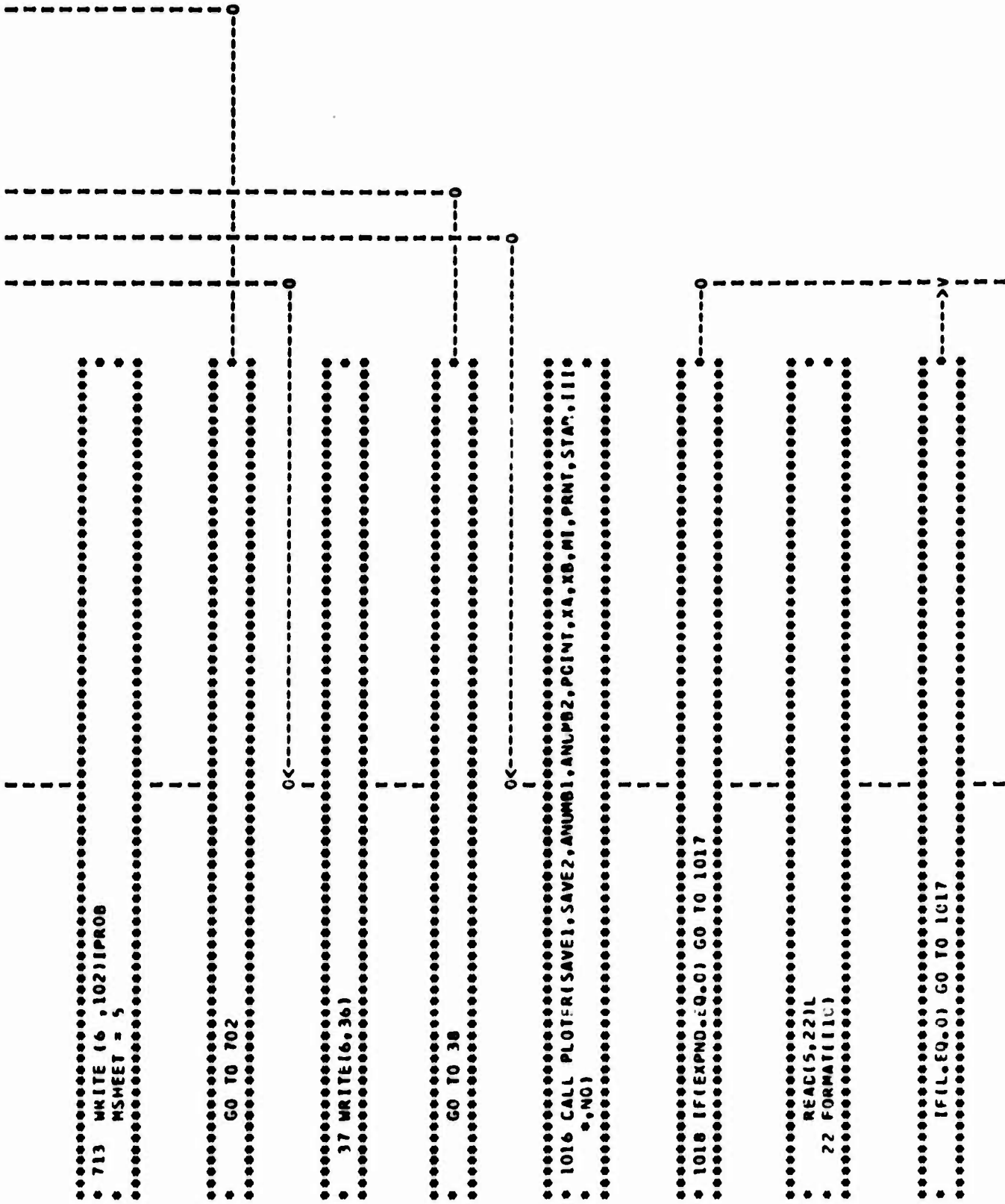

```

.....
* C COMPUTE ROOTS OF K * A * R
* IDC= MAXO(IDA, IDB)
* CALL POLACD (ATK, IDA, B, IDB, CK)
* IDS = IDC
.....
OK-----
* 754 IF (CK(IDS)) 755, 757, 755
.....
* 757 IDS = IDS - 1
.....
* IF (IDS) 758, 758, 754
.....
* 758 WRITE (6, 129)
.....
GO TO 711
.....
OK-----
* 755 IF (IDS - 2) 759, 760, 761
.....
* 759 WRITE (6, 130) CK(IDS), Y
.....

```







(ENTRANCE)

```
.....  
*C  
*C  
*C  
*C  
* SUBROUTINE POLMPY (A,N,B,M,C)  
* DIMENSION A(1),B(1),C(1)  
* K = M*N  
.....
```

```
.....  
DO 5 I=1,K  
.....
```

```
.....  
5 C(I) = 0.0  
.....
```

```
.....  
DO 10 I=1,N  
.....
```

```
.....  
L = I-1  
.....
```

```
.....  
DO 10 J=1,M  
.....
```

```
.....  
L = L+1  
.....
```

```
.....  
10 C(L) = C(L)+A(I)*B(J)  
.....
```

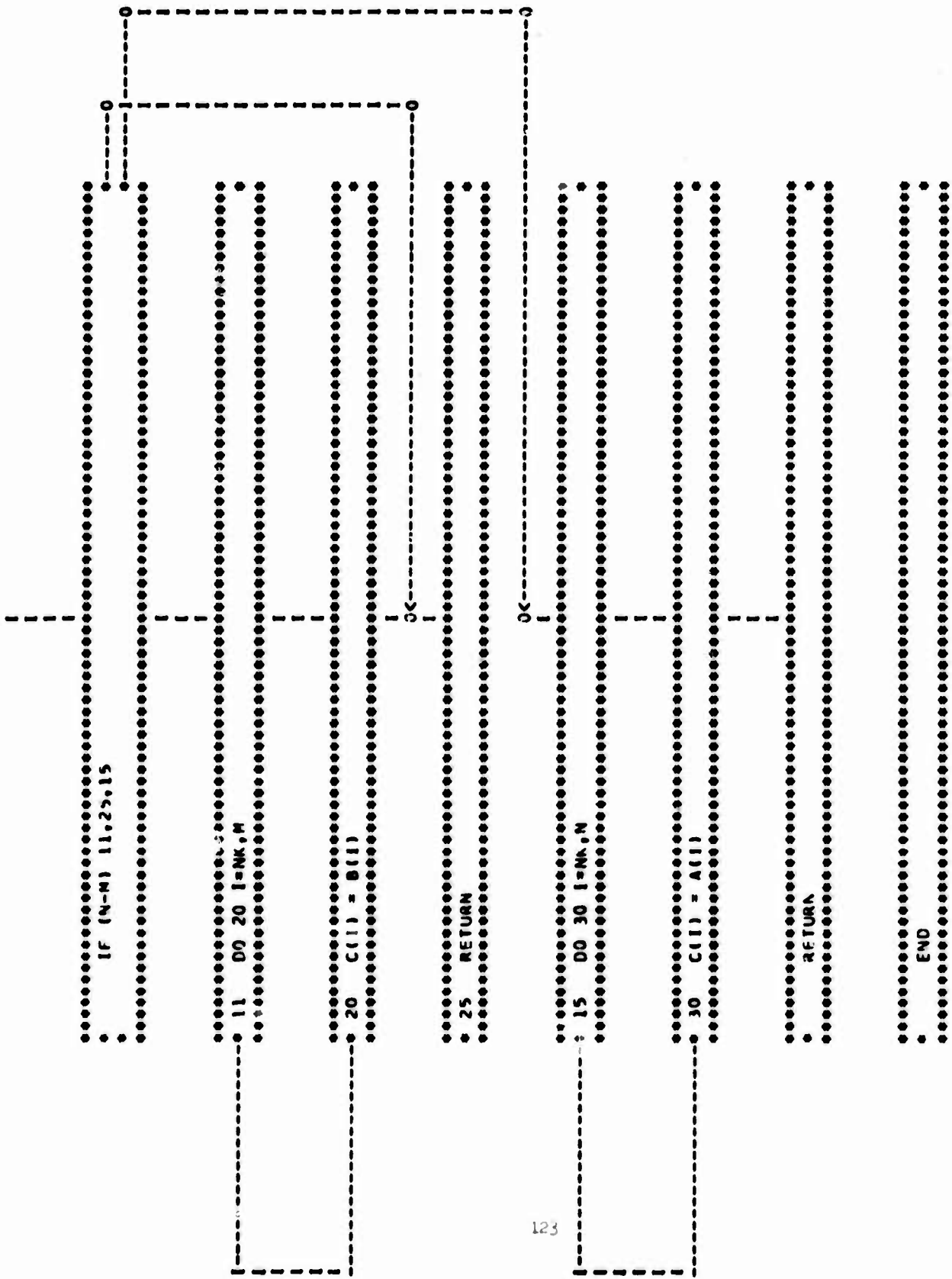
|||

.....
* RETURN *
.....

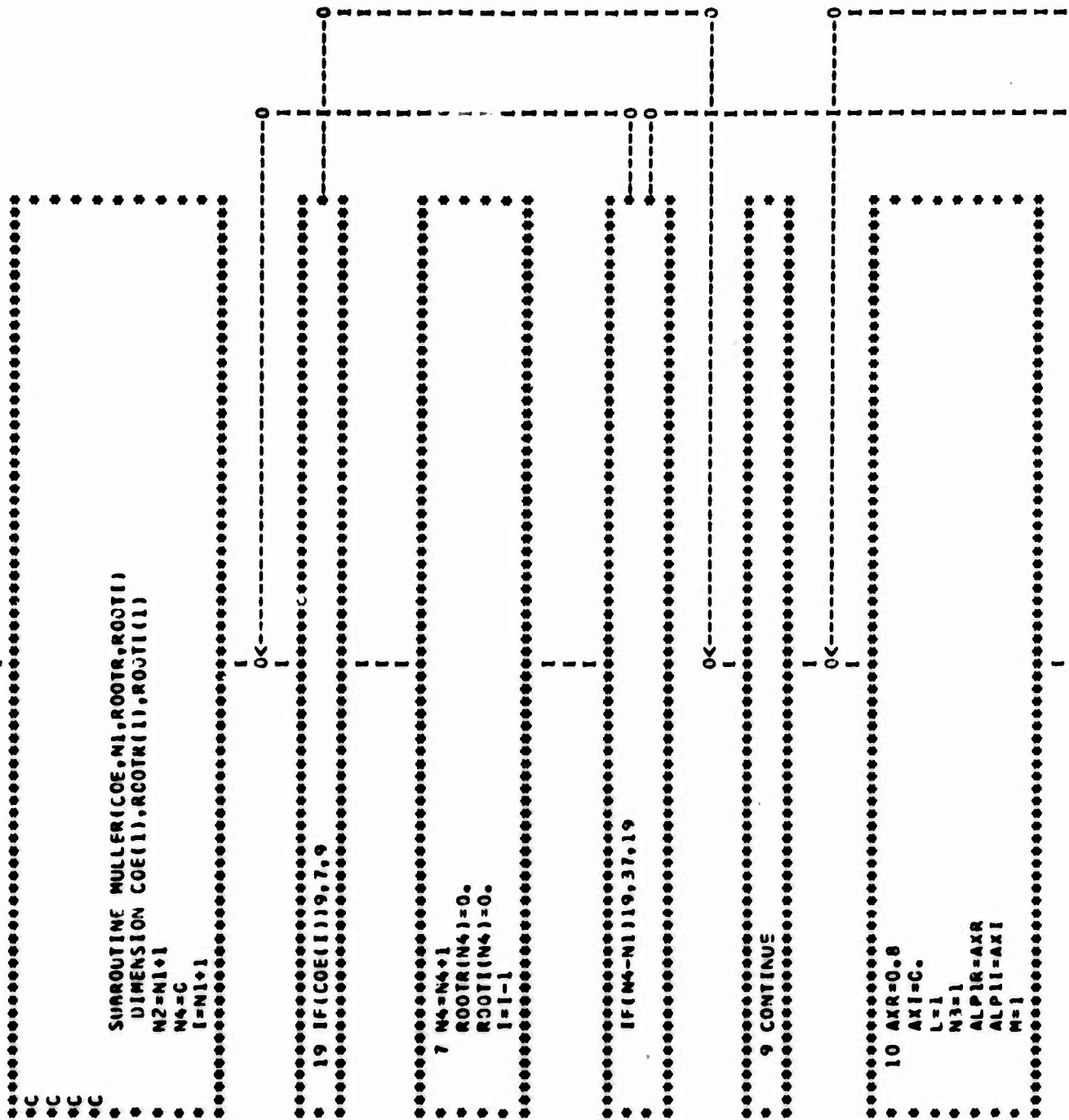
.....
* END *
.....

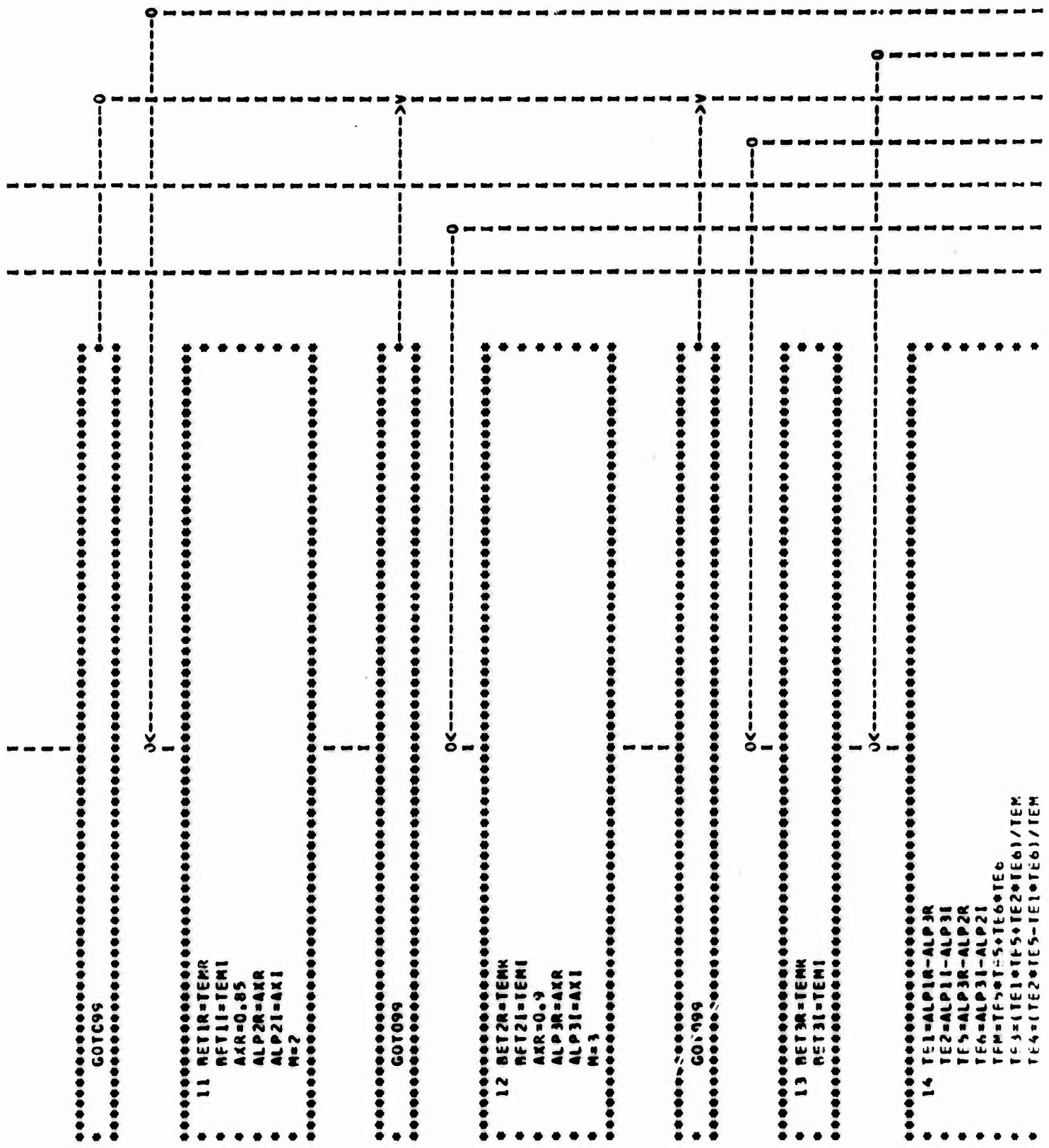
(ENTRANCE)

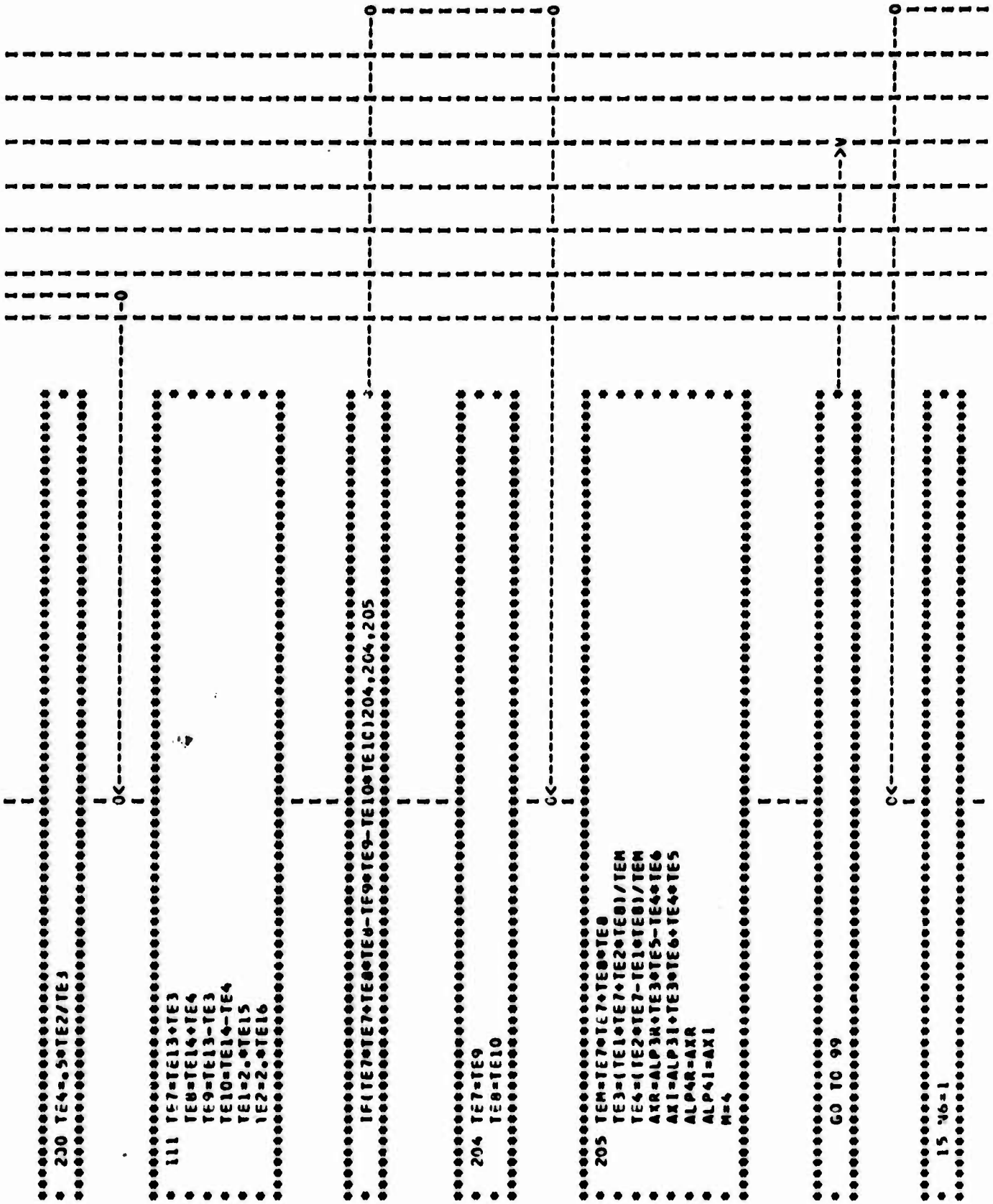
```
.....  
*C  
*C  
*C  
* SUBROUTINE POLADD (A,N,B,M,C)  
* DIMENSION A(1),B(1),C(1)  
*C  
.....  
.....  
.....  
* IF (N-M) 1,1,2  
.....  
.....  
* 1 NK = N  
.....  
.....  
* GO TO 5  
.....  
.....  
* 2 NK = P  
.....  
.....  
.....  
* 5 DO 10 I=1,NK  
.....  
.....  
* 10 C(I) = A(I)+B(I)  
.....  
.....  
* NK = NK+1  
.....  
.....
```



(ENTRANCE)



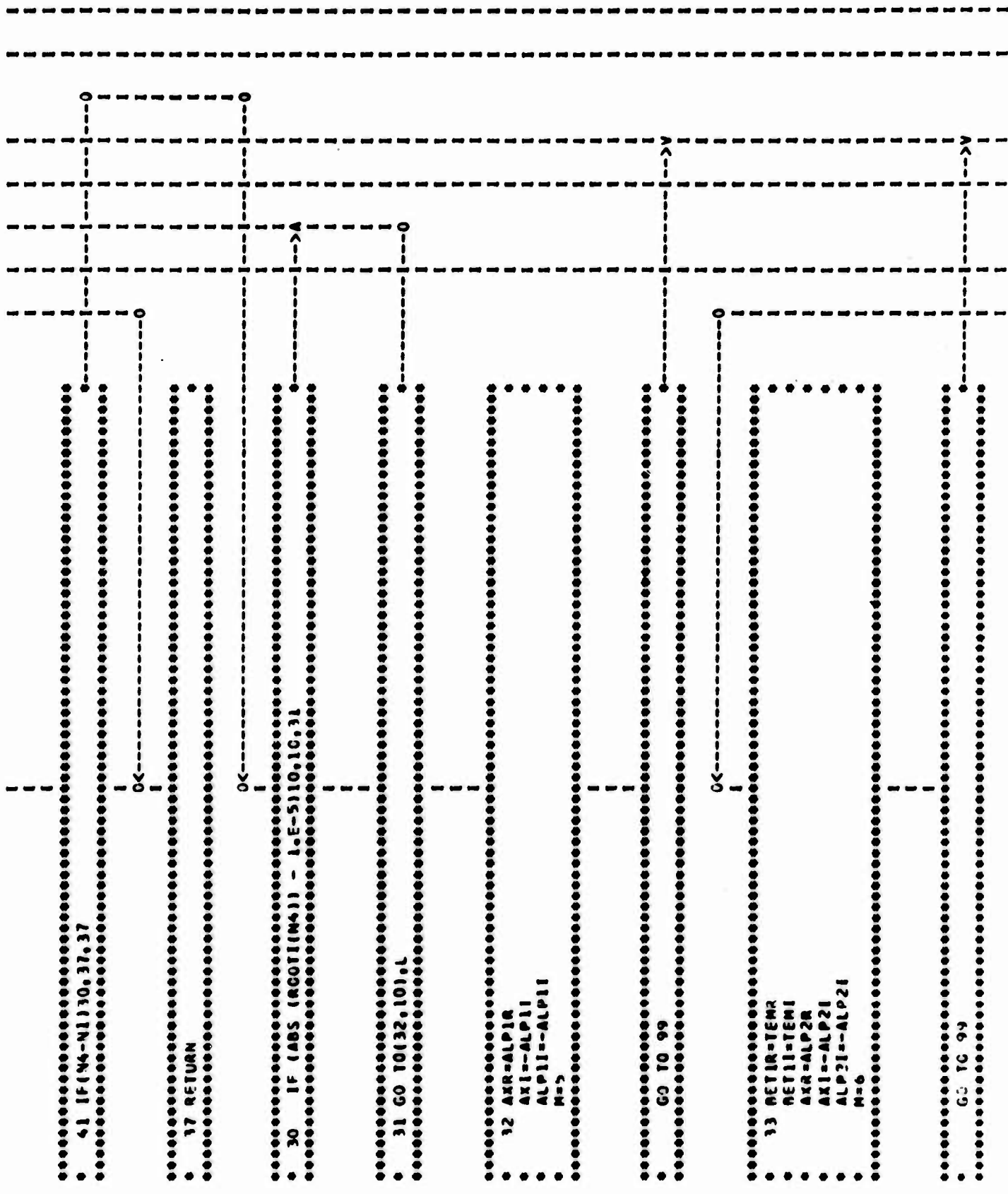




```

.....
* 38 IF (ABS (MELL) + ABS (BELL) - 1.E-20) 14,14,16
.....
.....
* 16 TE7 = ABS (ALP3R - AXR) + ABS (ALP3I - AXI)
.....
.....
* IF (TE7 / (ABS (AMR) + ABS (AXI)) - 1.E-7) 16,18,17
.....
.....
* 17 N3=N3+1
* ALP1R=ALP2R
* ALP1I=ALP2I
* ALP2R=ALP4R
* ALP2I=ALP3I
* ALP3R=ALP4R
* ALP3I=ALP4I
* BET1R=BET2R
* BET1I=BET2I
* BET2R=BET3R
* BET2I=BET3I
* BET3R=TEM3
* BET3I=TEMI
.....
.....
* IF(N3-100)14,18,18
.....
.....
* 18 N4=N4+1
* ROOTR(N4)=ALP4R
* ROOTI(N4)=ALP4I
* N3=0
.....
.....

```



0<

```

.....
* 14 SET2R=TEMR
*   GET2I=TEMI
*   AXQ=ALP3R
*   AXI=-ALP3I
*   ALP3I=-ALP3I
*   L=2
*   M=3
.....

```

0<

```

.....
* 99 TEMR=COE(1)
*   TEMI=C*O
.....

```

130

```

.....
* DO1001=1,N1
.....

```

```

.....
* TEL=TEMR*AXR-TEMI*AXI
*   TEMI=TEMI*AXR+TEMR*AXI
.....

```

```

.....
* 100 TEMR= TEL*COE(1+1)
.....

```

```

.....
* HELL=TEMR
*   BELL=TEMI
.....

```

```

.....
* 42 IF(N4)102,103,102
.....

```

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....



(ENTRANCE)

```
.....  
*C  
*C  
*C  
*C  
* SUBROUTINE EXPLOT(XARRAY, IDIMEN, YARRAY, KOIMEN, BLIMIT, LET)  
* INTEGER POINT(130, 100), GRAPHL(120, 50), STAR  
* DIMENSION XARRAY(ICIMEN), YARRAY(KDIMEN)  
* DIMENSION ENCRM1(120), ENCRM2(50)  
* DIMENSION SAVE1(100, 100), SAVE2(100, 100)  
* COMMON SAVE1, SAVE2, POINT  
* EQUIVALENCE(POINT(1, 1), GRAPHL(1, 1))  
* DATA STAR/347  
* WRITE(6, 1)  
* READ(5, 10) D1, D2  
* D3=BLIMIT  
* D4=-BLIMIT  
* IF(D2.GT.D1) CALL SWITCH(D1, D2)  
* IF(D4.GT.D3) CALL SWITCH(D3, D4)  
* IC1=D3  
* JDIMEN=KDIMEN  
* CALL PLANAR(GRAPHL, 120, 50, 120)  
.....
```

```
.....  
DO 22 I=1, JOIMEN  
.....  
* X=XARRAY(I)  
* Y=YARRAY(I)  
.....  
* IF(X.EQ.0.0.AND.Y.EQ.0.0) GO TO 22  
.....  
* CALL LINAR (X, LIMITX, 120, D1, D2, ENCRM1)  
* CALL LINAR (Y, LIMITY, 50, D3, D4, ENCRM2)  
.....  
.....
```

```

.....
* IF(LIMITX.EQ.C.OK.LIMITY.FQ.0) GO TO 22
.....
.....
* LIMITX=121-LIMITX
* GRAPHL(LIMITX,LIMITY)=STAR
.....
* OK
.....

```

```

.....
* 22 CONTINUE
.....

```

```

.....
* WRITE(6,30) IC1,LET,(GRAPHL(I,1),I=1,120)
* IC1=D4
* WRITE(6,34)((GRAPHL(I,J),I=1,120),J=2,57)
* WRITE(6,35) IC1,LET,(GRAPHL(I,58),I=1,120)
* WRITE(6,35) D2,D1
* 1 FORMAT(1H1)
* 10 FORMAT(6F10,0,20X)
* 30 FORMAT(1X,14,A4,1X,120A1)
* 34 FORMAT(10X,120A1)
* 35 FORMAT(3X,1P1E10.3,7HRAD/SEC,91X,1P1E10.3,7HRAD/SEC)
.....

```

```

.....
* 61 RETURN
.....

```

```

.....
* END
.....

```


(ENTRANCE)

```
.....  
*C  
*C  
*C  
*C  
SURROUTIN= LINAR (Y,LIMITY,LDIMEN,DIMNS1,DIMNS2,ENCRT)  
DIMENSION ENCRT(LDIMEN)  
DIMENS=DIMNS1-DIMNS2  
LIMITY=0  
J=LDIPEN-1  
A=J  
DELTA=DIMENS/A  
ENCRT(1)=DIMNS1  
ENCRT(LDIMEN)=DIMNS2  
.....
```

```
.....  
DO 10 I=2,J  
.....
```

```
.....  
10 ENCRT(I)=ENCRT(I-1)-DELTA  
.....
```

```
.....  
DO 11 I=1,J  
.....
```

```
.....  
11 IF(Y.LE.ENCRT(I).AND.Y.GE.ENCRT(I+1))GO TO 20  
.....
```

```
.....  
GO TO 40  
.....
```

```
.....  
20 LIMITY=I  
.....
```

```
.....  
40 RETURN  
.....
```

```
.....  
END  
.....
```

```

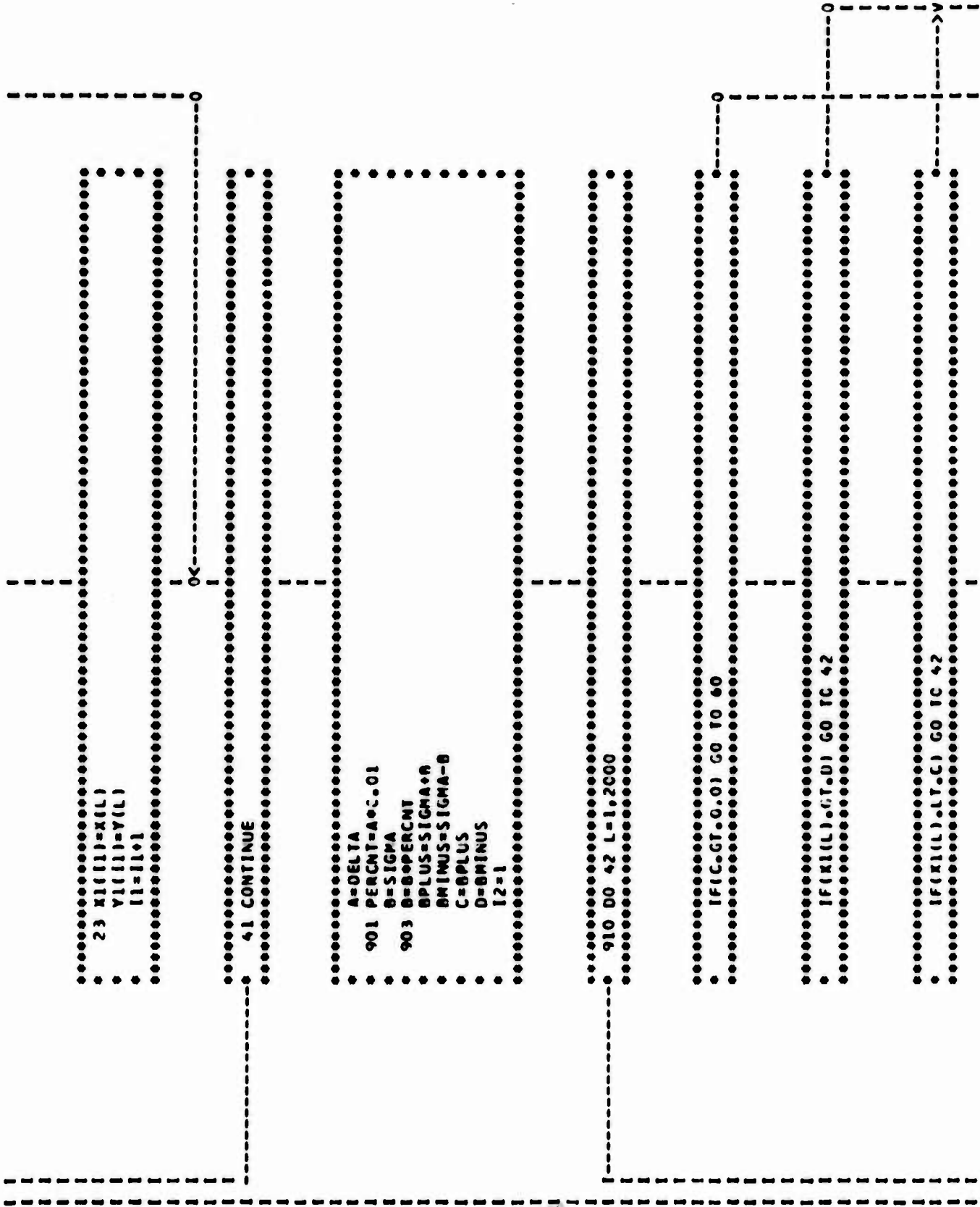
      (ENTRANCE)
      I
      I
      .....
    *C
    *C
    *C  SUBROUTINE SWITCH(X,Y)
    *C  TEST=1
    *C  V=X
    *C  X=TEST
    .....
      I
      I
      I
      .....
    *C
    *C  RETURN
    *C
    .....
  
```

```

    .....
    *C  END
    *C
    .....
  
```

```

      (ENTRANCE)
      I
      I
      .....
    *C
    *C  SUBROUTINE ERCHEK(X,I)
    *C  DIMENSION X(100)
    *C  DATA ERR/0.1E-8/
    .....
      I
      I
      I
      .....
      .....
    *C  DO 10 J=1,I
    *C
    *C
    *C
    *C
    *C
    *C
    *C  10 IF(ABS(X(J)).LT.ERR/1M) X(J)=0.0
    *C
    *C
    *C  RETURN
    *C
    .....
    *C
    *C  END
    *C
    .....
  
```

(ENTRANCE)

```

.....
OC
OC
OC
OC
SURMOUTINE SPLIT(X,Y,SPOT,APLUS,AMINUS,BPLUS,BMINUS)
DIMENSION X(200),Y(200)
INTEGER SPOT(130,50)
COMMON /PJ/XI,YI
EQUIVALENCE(X(1),Z(1)),(Y(1),Z(1))
A=APLUS
B=AMINUS
C=A-B
D=BPLUS
E=BMINUS
G=ABS(D)
H=ABS(E)
F=G-H
DELTA=C/124.0
DIFF=F/50.0
.....

```

```

.....
DO 11 J=1,124
.....

```

```

.....
Z(J)=A
.....

```

```

.....
11 A=A-DELTA
.....

```

```

.....
Z(125)=B
IF(C.LT.0.0)DIFF=-DIFF
.....

```

```

.....
DO 12 J=1,49
.....

```


(ENTRANCE)

```
.....  
*C .....  
*C .....  
*C .....  
*C .....  
.....  
SUBROUTINE MRKUP(Y,X,Y,SPOT,XX)  
LOGICAL SKIP,SKIPI  
INTEGER SPOT(130,50),  
DATA STAR/IM/ XPT,YPT,STAR  
DIMENSION X(200),Y(200),YY(125),X(87)  
COMMON /PAYNE/AM,AP,PM,PP  
EQUIVALENCE(AP,APLUS),(AM,AMINUS),(PM,PMPLUS)  
SKIP=.FALSE.  
SKIPI=.FALSE.  
.....
```

```
.....  
DC 43 J=1,200  
.....
```

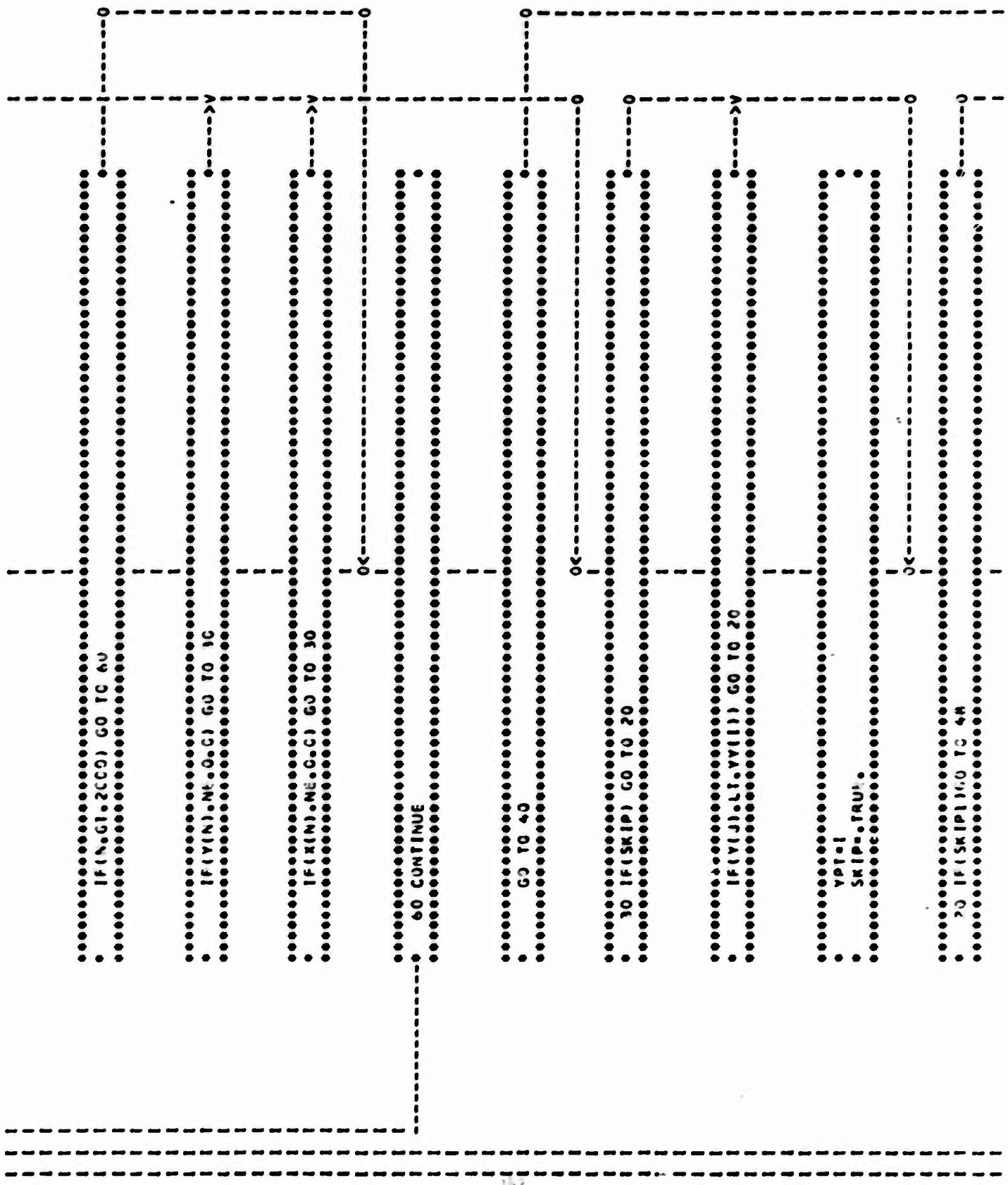
```
.....  
DO 43 I=1,125  
.....
```

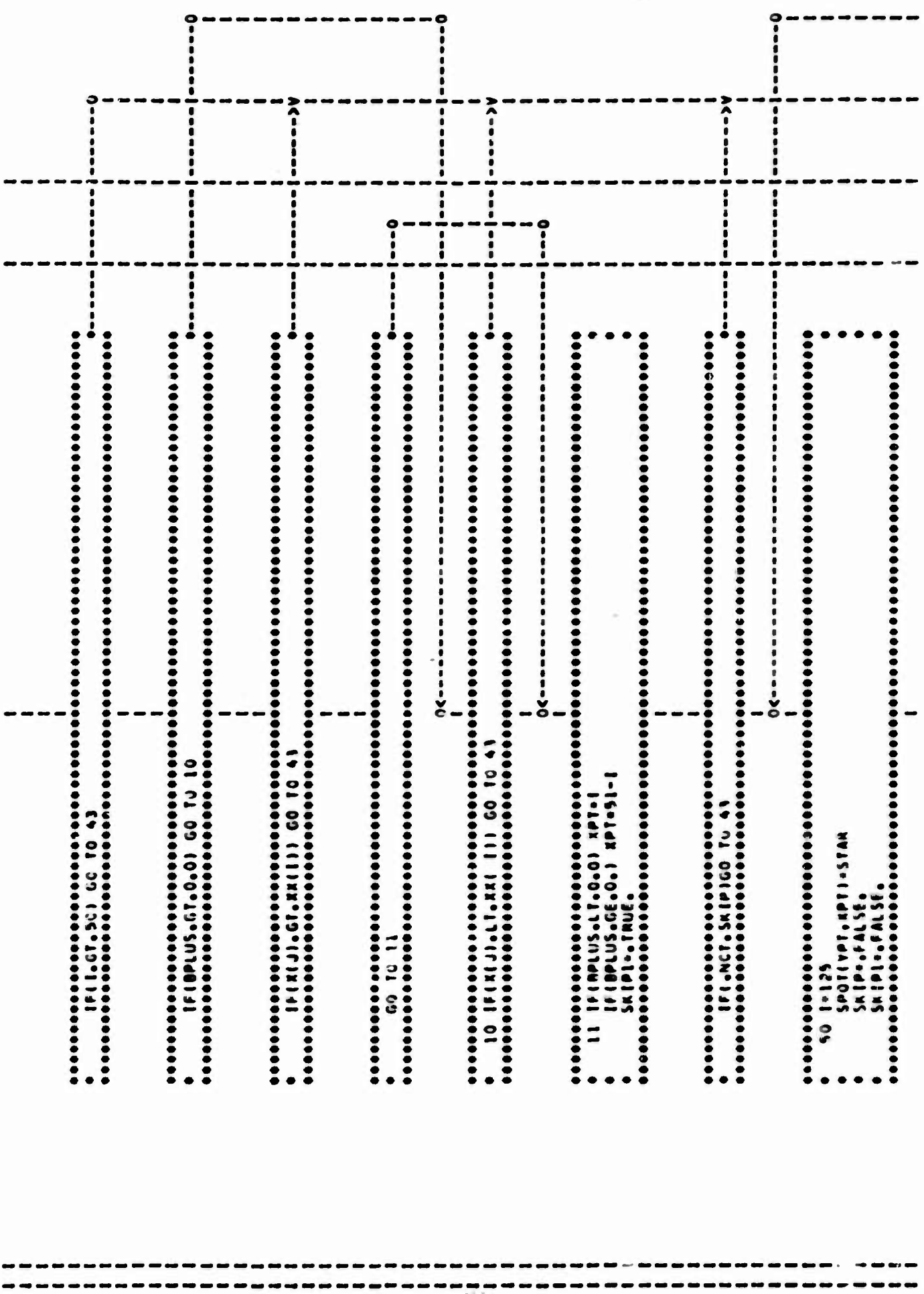
```
.....  
IF(Y(J),NT,0,C) GO TO 30  
.....
```

```
.....  
IF(X(J),NT,0,C) GO TO 30  
.....
```

```
.....  
L=J+1  
M=J+6  
.....
```

```
.....  
DC 40 N=L,M  
.....
```





```

.....
* IF(I.GT.50) GO TO 43
.....

```

```

.....
* IF(PLUS.GE.0.0) GO TO 10
.....

```

```

.....
* IF(X(J).GT.XX(1)) GO TO 41
.....

```

```

.....
* GO TO 11
.....

```

```

.....
* 10 IF(X(J).LT.XX(1)) GO TO 41
.....

```

```

.....
* 11 IF(PLUS.LT.0.0) XPT=1
* IF(PLUS.GE.0.0) XPT=51-1
* SKIP=.TRUE.
.....

```

```

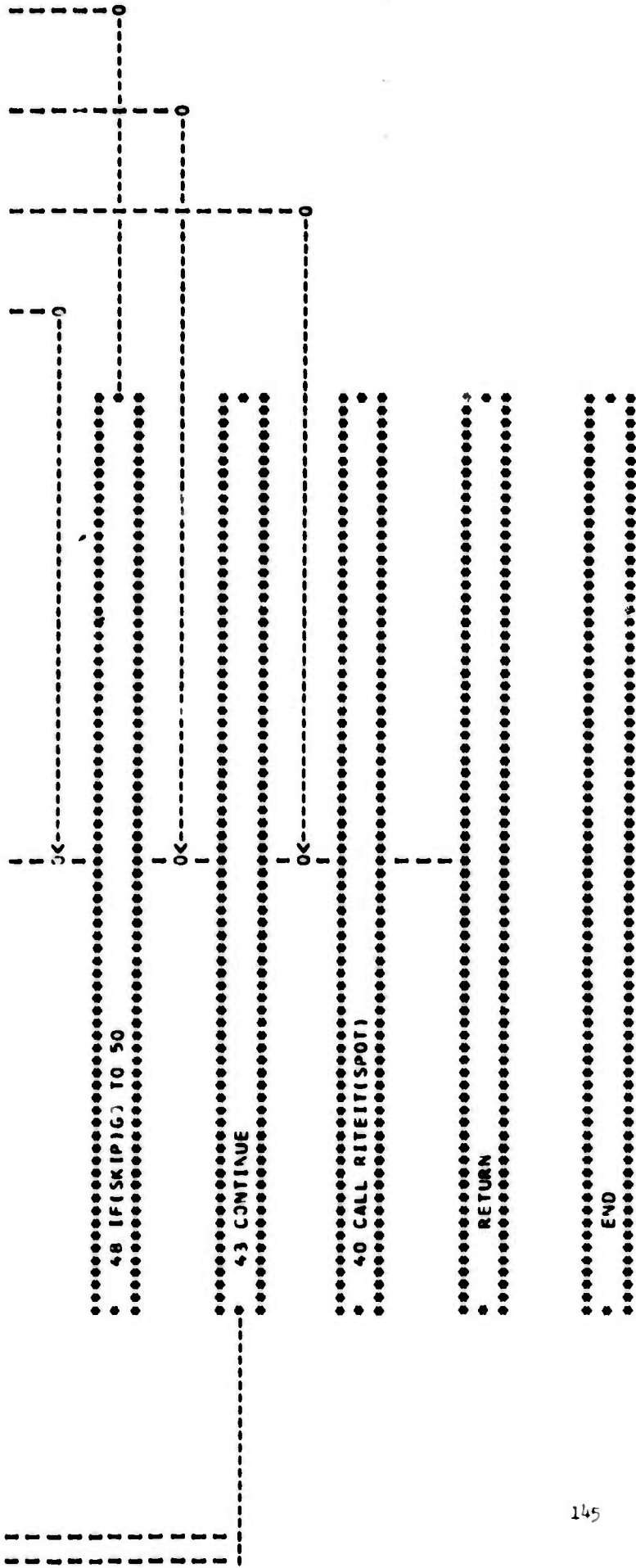
.....
* IF(.NOT.SKIP) GO TO 41
.....

```

```

.....
* 50 I=125
* SPOT(VPT,XPT)=STAN
* SKIP=.FALSE.
* SKIP=.FALSE.
.....

```



(ENTRANCE)

```
.....  
*C .....  
*C .....  
*C .....  
*C .....  
* SUBROUTINE RITEIT (SPOT)  
* COMMON /PAYNE/AM,AP,PM,BP  
* EQUIVALENCE (AP,APLUS), (AM,AMINUS), (PM,BMINUS), (BP,BPLUS)  
* INTEGER SPOT (130,5C)  
.....  
.....  
* IF (BP.LT.PM) GO TO 40  
.....  
.....  
* A=BP  
* B=BM  
* WRITE (6,1)  
* 1 FORMAT (1H1,40X,43HCOMPLEX FREQUENCY PLANE,RIGHT HAND QUADRANT )  
.....  
.....  
* GO TO 30  
.....  
* 0K-----0  
.....  
* 40 A=BM  
* B=BP  
* WRITE (6,2)  
* 2 FORMAT (1H1,40X,43HCOMPLEX FREQUENCY PLANE, LEFT HAND QUADRANT )  
.....  
.....  
* 0K-----0  
.....  
* 30 WRITE (6,12) AP,AM  
* 12 FORMAT (114X,15HC-----J-Omega,/1X,F8.2,112X,F8.2)  
* WRITE (6,14) B  
* 14 FORMAT (124X,5MSIGMA,/126X,1M1,/126X,1M1V,/121X,F8.2)  
* 11 FORMAT (1X,130A1)  
* WRITE (6,15) A  
* 15 FORMAT (60X,27HLINEAR EXPAND PLOT(RAD/SEC) ,34X,F8.2)  
.....  
* I
```

.....
RETURN
.....

.....
END
.....

(ENTRANCE)

```
.....  
*C  
*C  
* SUBROUTINE SAVER(RCOTR,RCOTI, IDIC,SAVE1,SAVE2,JZO,K1)  
* DIMENSION SAVE1(100,100),SAVE2(100,100),ROOTR(100),ROOTI(100)  
*.....
```

```
.....  
* IF(K1)30,9,10  
*.....
```

```
.....  
* 9 K1=1  
*.....
```

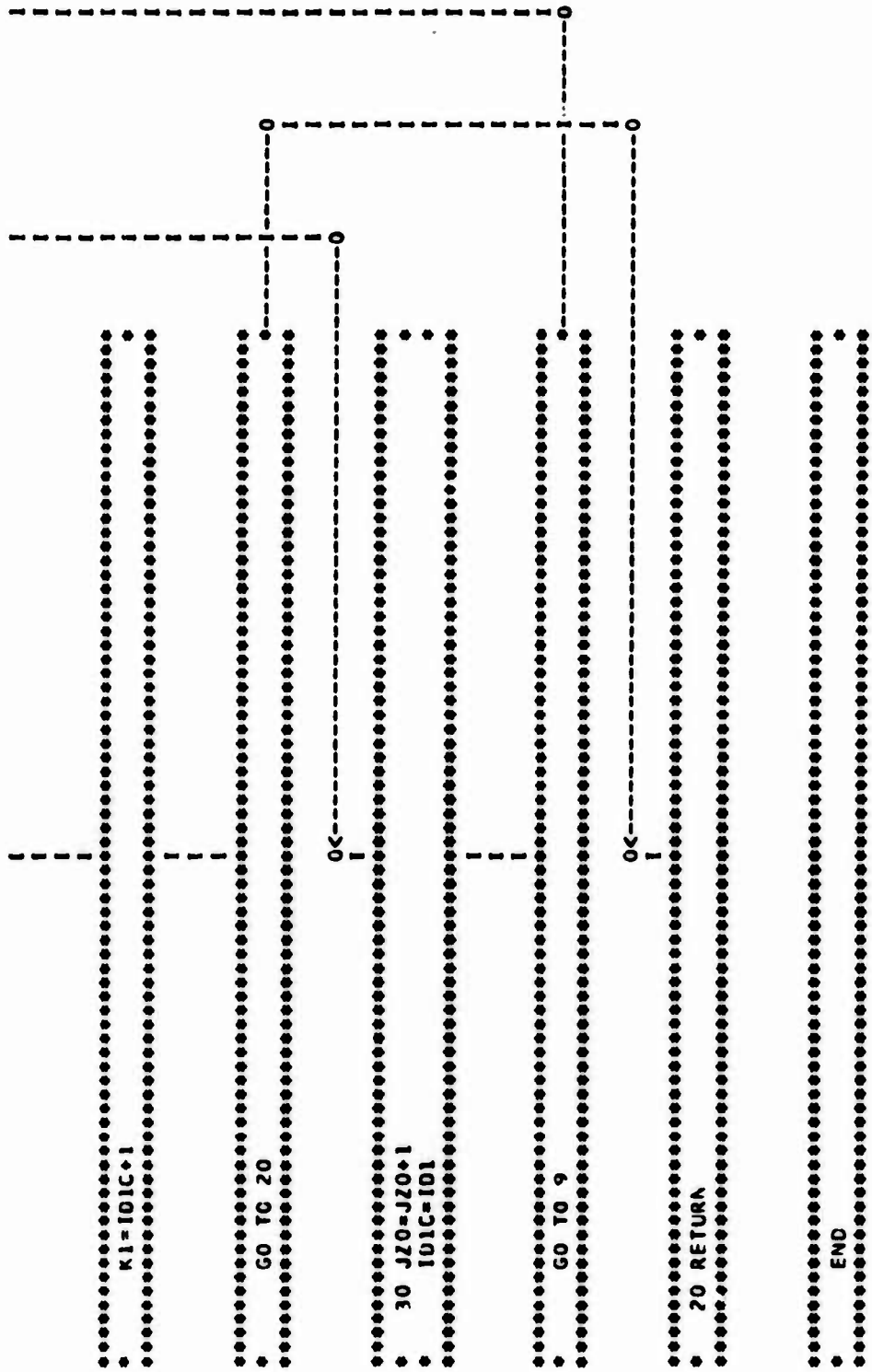
```
.....  
* 10 ID1=ICIC  
* IDIC=IDIC*(K1-1)  
*.....
```

```
.....  
* IF(IDIC.GE.100)GO TO 30  
*.....
```

```
.....  
* 50 DO 40 IZAP=K1, IDIC  
*.....
```

```
.....  
* IZA=IZAP-(K1-1)  
* SAVE1(JZO, IZAP)=ROOTR( IZA)  
* SAVE2(JZO, IZAP)=ROOTI( IZA)  
*.....
```

```
.....  
* 40 CONTINUE  
*.....
```



(ENTRANCE)

```
.....  
*C .....  
*C .....  
*C .....  
*C .....  
* SURROUTINE PLOTTER(SAVE1,SAVE2,ANUMB1,ANUMB2,POINT,XA,XB,PI,PRNT,  
* STAR,NBB,NO)  
* DIMENSION SAVE1(100,100),SAVE2(100,100),XA(2000),XB(2000)  
* INTEGER PLINT(130,100),DASH,BLANK,STAR,ROTCLS,EXPND,FREQSR,DENSE  
* COMMON /INFO7/ ICOUNT  
* COMMON /INFO8/ROTCLS,EXPND,FREQSR  
* COMMON /INFO9/ DENSE  
* LOGICAL PRNT  
.....
```

```
.....  
* IF(PRNT) GO TO 43  
.....
```

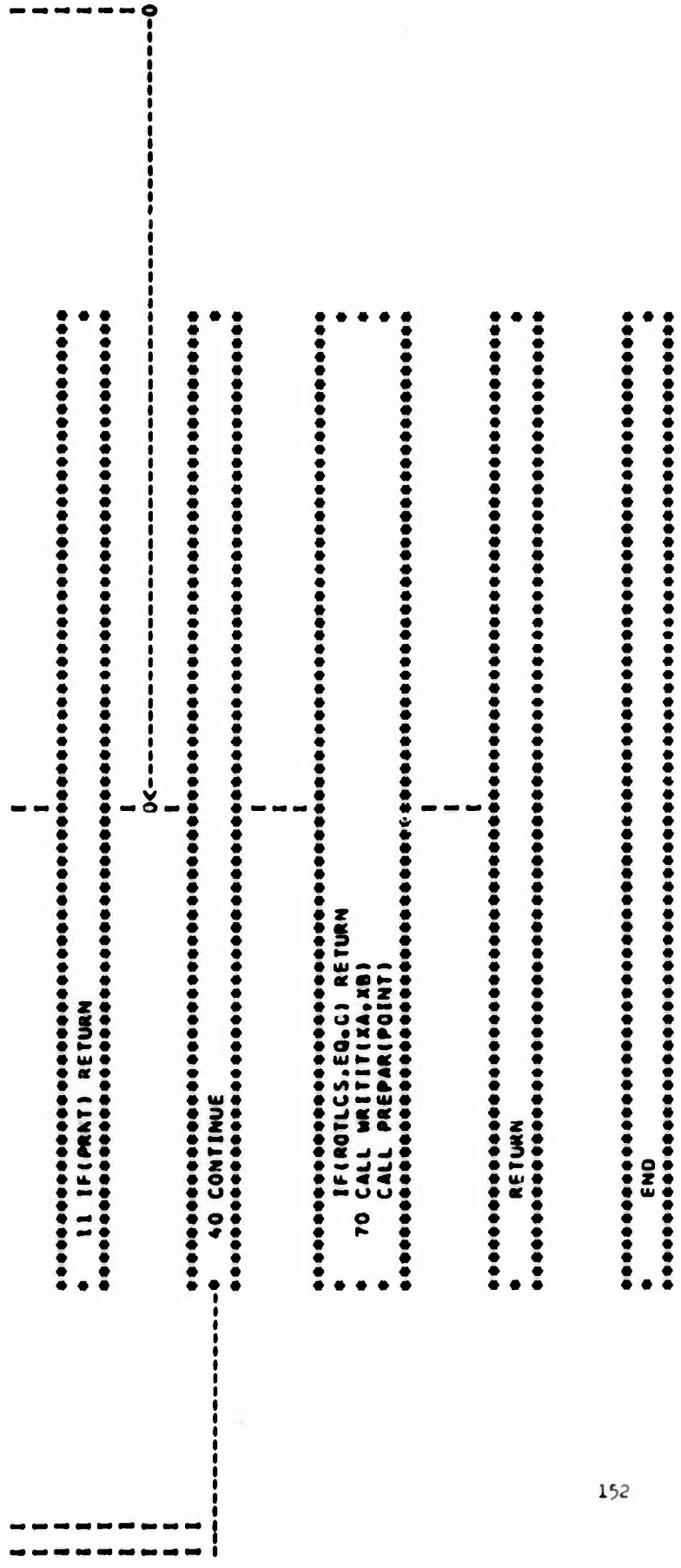
```
.....  
* DO 40 NAB=1,100  
.....
```

```
.....  
* DO 40 NBB=1,100  
.....
```

```
.....  
* IF(SAVE1(NAB,NBB)41,42,41  
.....
```

```
.....  
* 42 IF(SAVE2(NAB,NBB)41,40,41  
.....
```

```
.....  
* 41 ANUMB1=SAVE1(NAB,NBB)  
* ANUMB2=SAVE2(NAB,NBB)  
* ANUMB2=ABS(ANUMB2)  
.....
```

(ENTRANCE)

```
.....
*C
*C
*C
*C
* SUBROUTINE EXCUTE(ANUMR1,ANUMR2,POINT,XA,XH,MI,NG,STAR)
* LOGICAL SKIP1,SKIP2,LFSS
* INTEGER RGTLC5,EXPND,FREQSR
* DIMENSION XA(2000),XR(2000)
* INTEGER POINT(130,100),DASH,BLANK,STAR
* COMMON /INFO8/RGTLC5,EXPND,FREQSR
* DATA KI/10/,K2/100/,K3/1000/,K4/10000/,K5/100000/,-1/
* LESS=.FALSE.
* I=0
* J=U
* XA(NG)=ANUMR1
* XR(NG)=ARS(ANUMR2)
* NG=NG+1
* IF(RGTLC5 EQ.C) RETURN
*.....
*.....
* IF(ABS(ANUMR1).GT.10000.0) GO TO 5C
*.....
*.....
* IF(ABS(ANUMR2).GT.10000.0) GO TO 5C
*.....
*.....
* IF(ABS(ANUMR1).EQ.C.0) GO TO 5CC
*.....
*.....
* IF(ABS(ANUMR2).EQ.C.0) GO TO 500
*.....
*.....
* IF(ABS(ANUMR2).LT.0.001) GO TO 5C
*.....
*.....
*.....
```

```

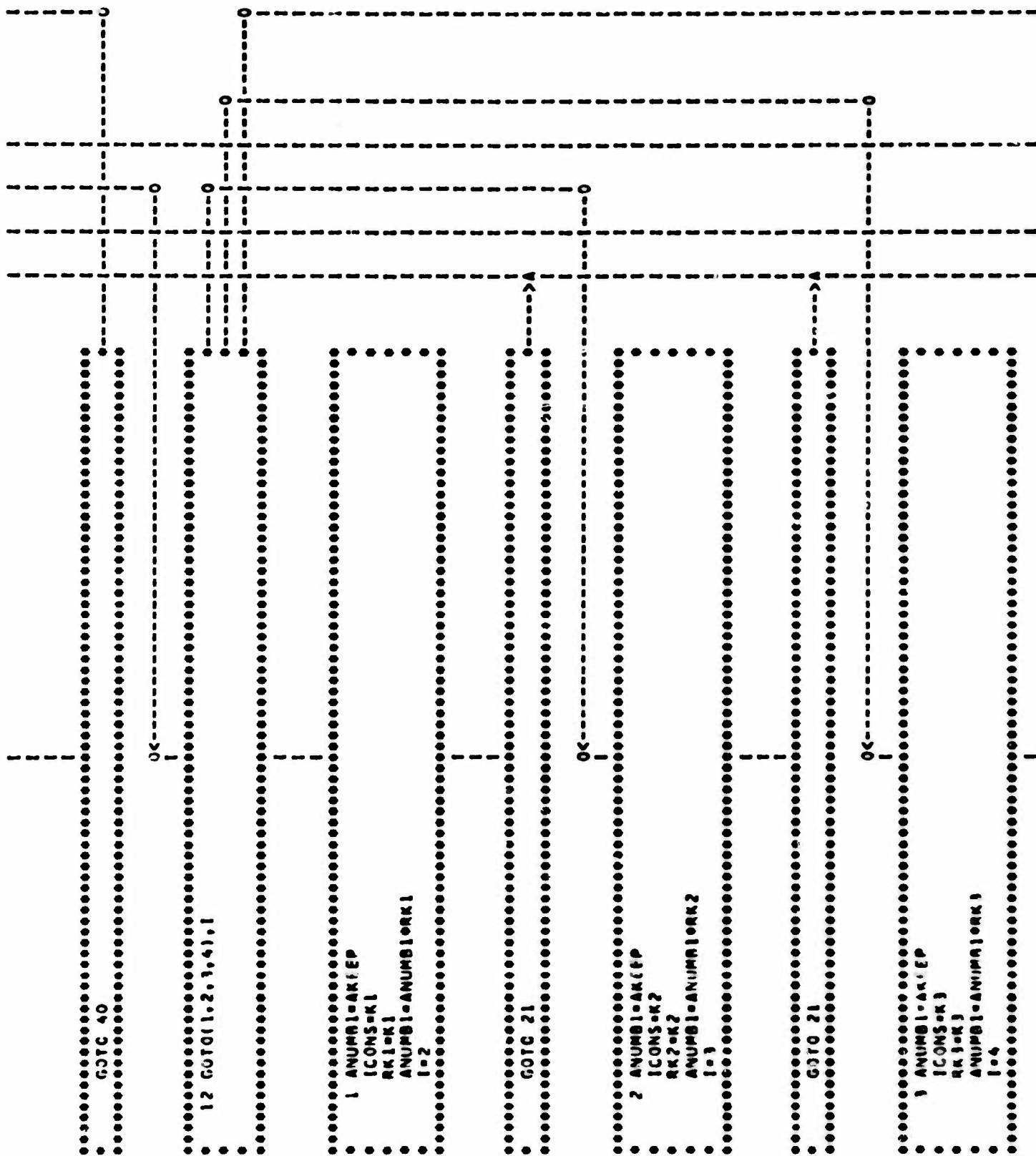
.....
*   IF (ABS (ANUMB1) .LT. 0.001) GO TO 50
.....
*  

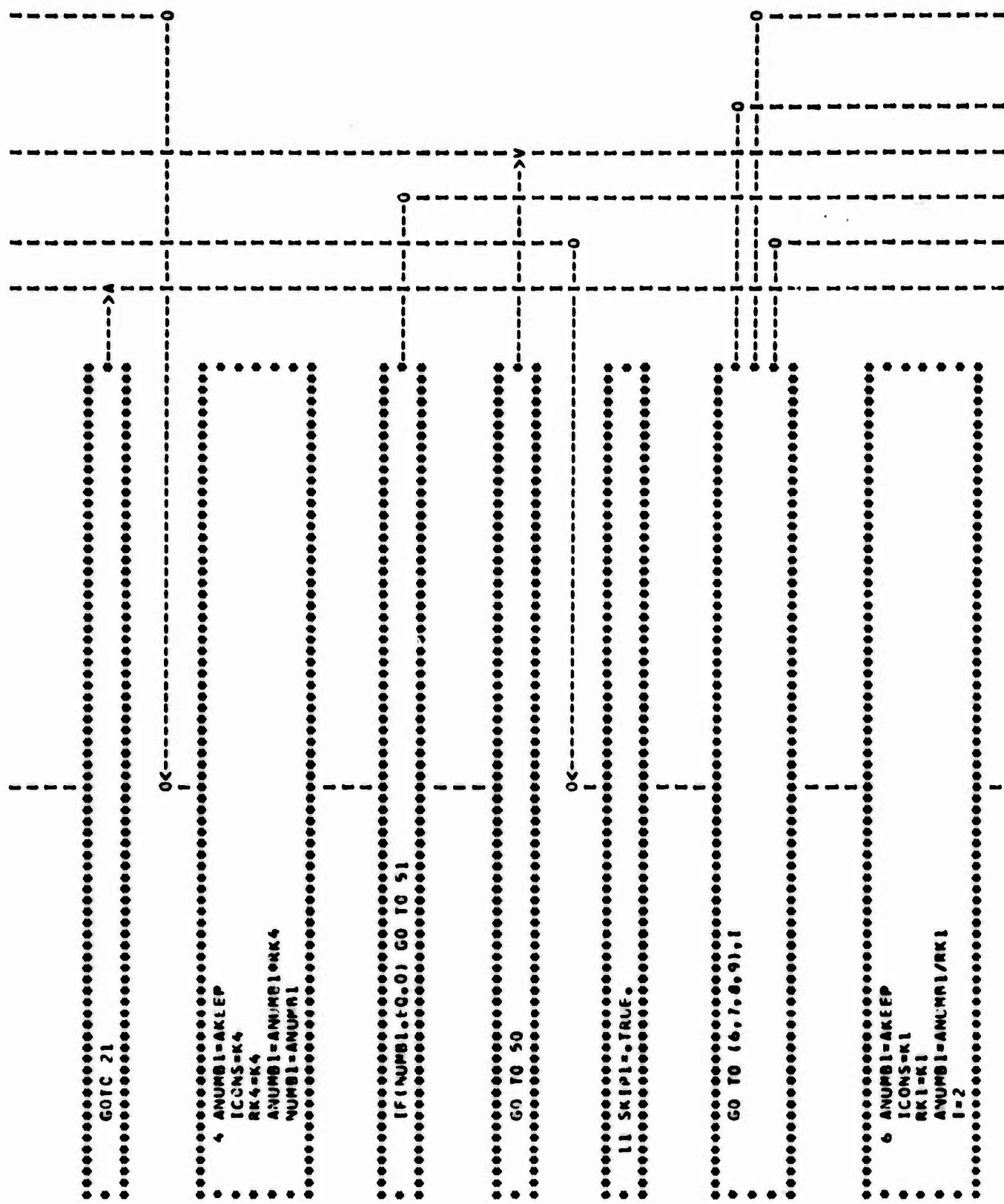
.....
*   CALL SCALE1 (K1, K2, K3, K4, NEGONE, I, ANUMB1, ICONS, LESS, SKIP1)
*   CALL SCALE2 (K1, K2, K3, K4, J, JCONS, ANLPB2, SKIP2)
*   CALL NPOINT (J, JCONS, L, ANUMB2, SKIP2)
*   CALL SPOINT (I, ANUMB1, LESS, ICONS, L, POINT, SKIP1, NO, XA, XB, STAR)
.....
*  

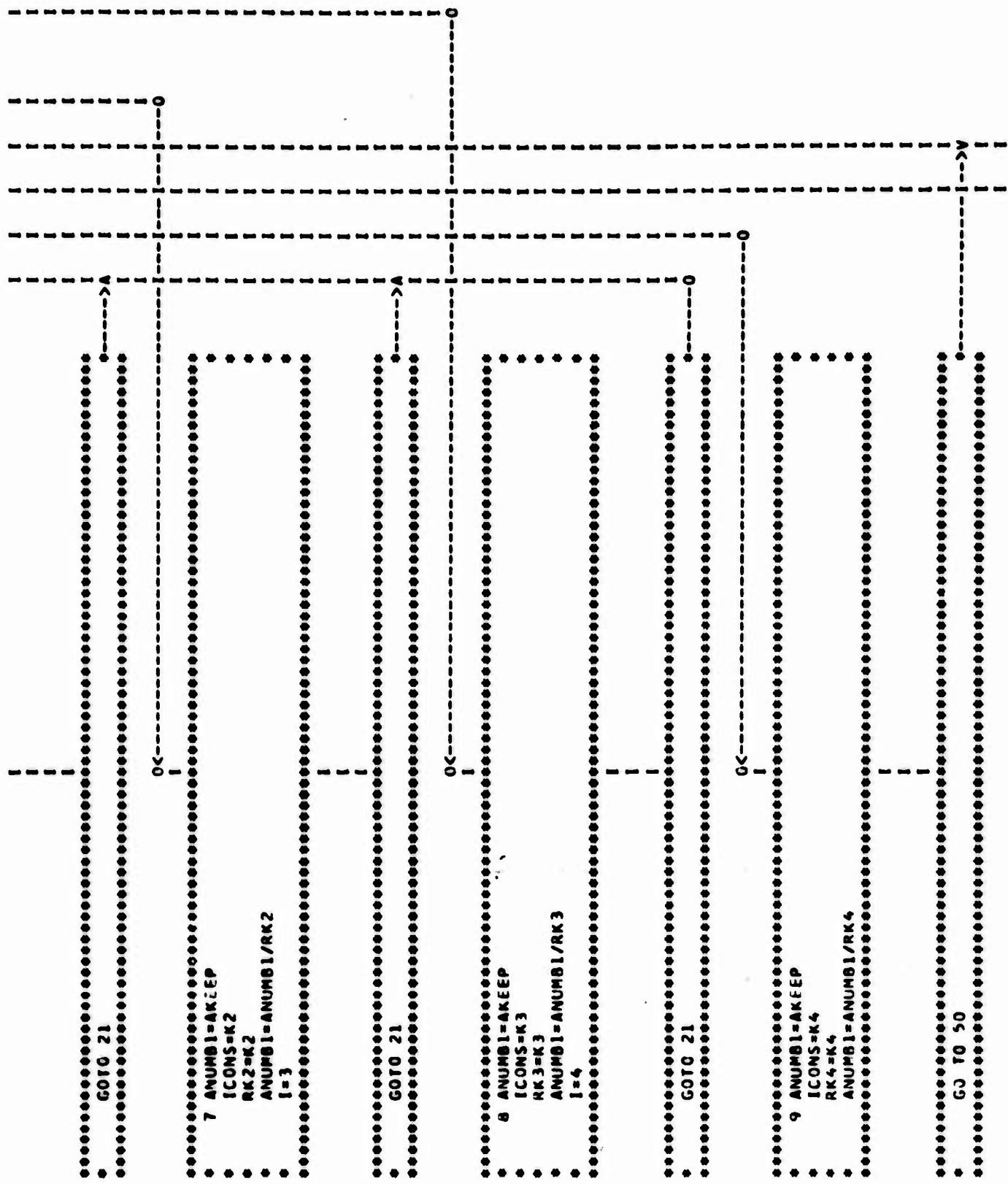
.....
*   50 RETURN
.....
*  

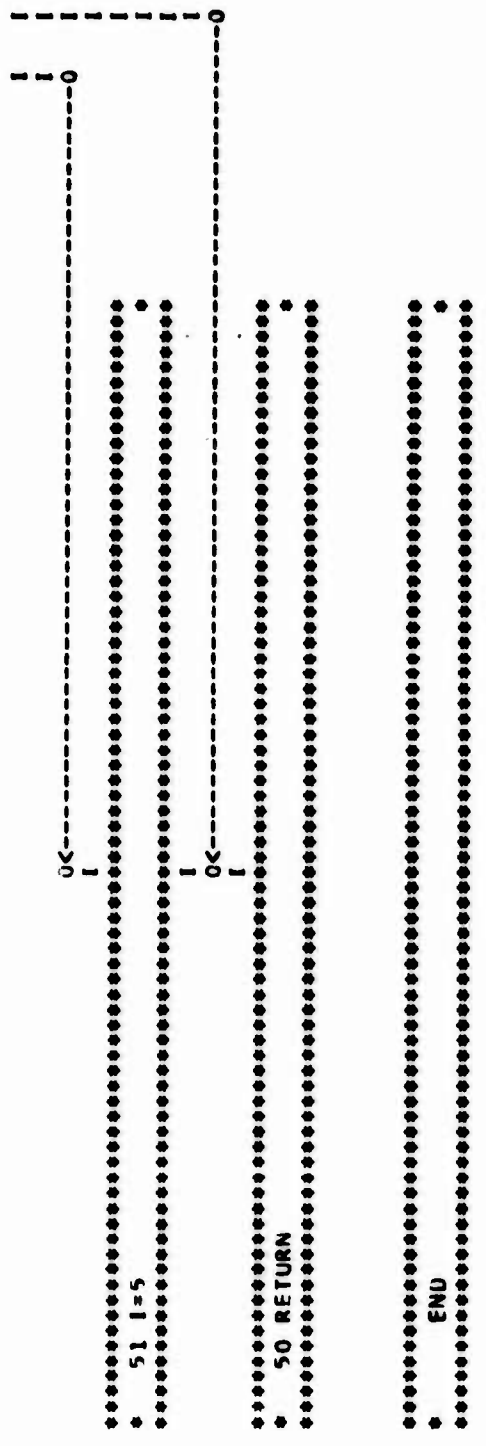
.....
*   END
.....

```

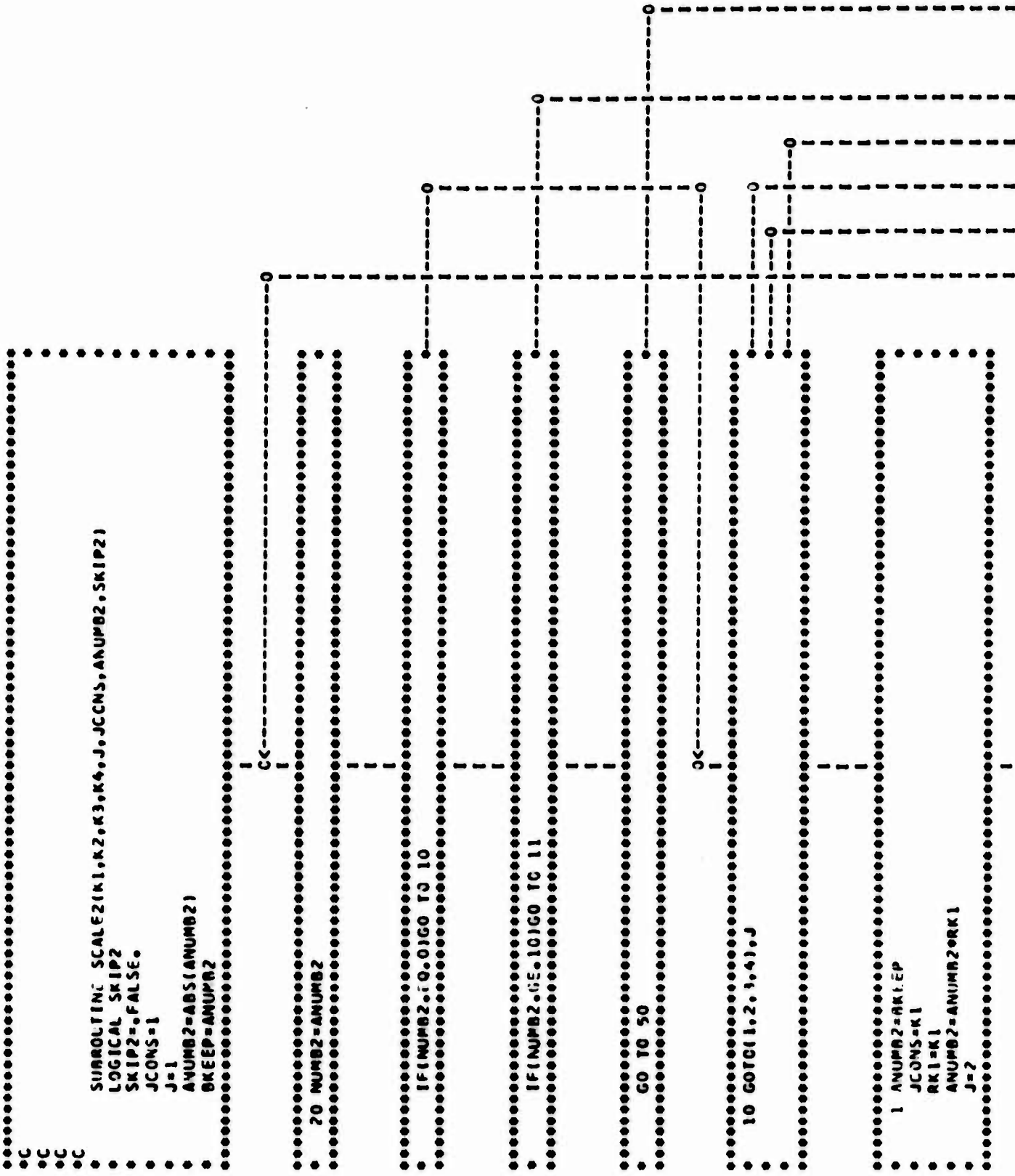



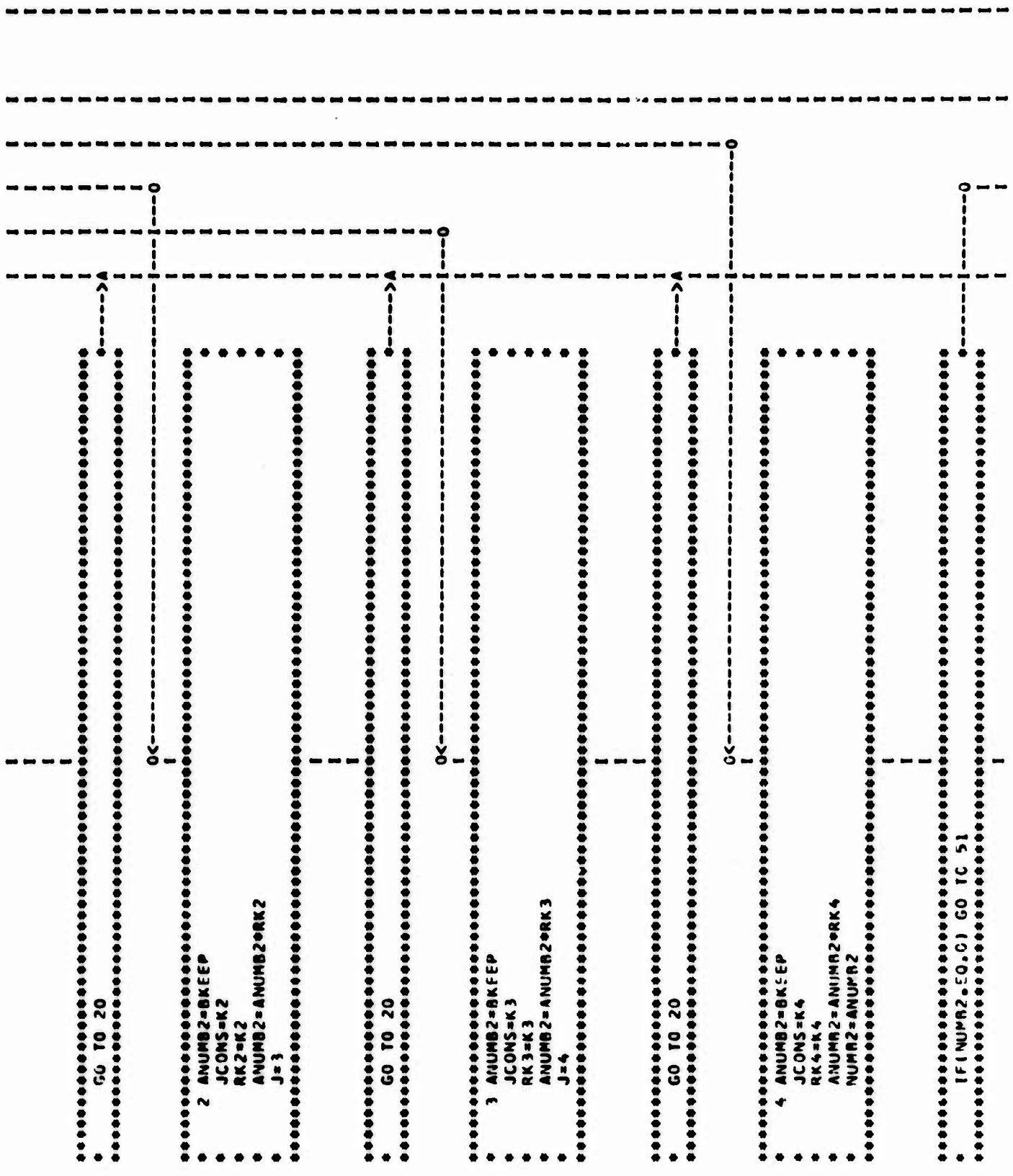


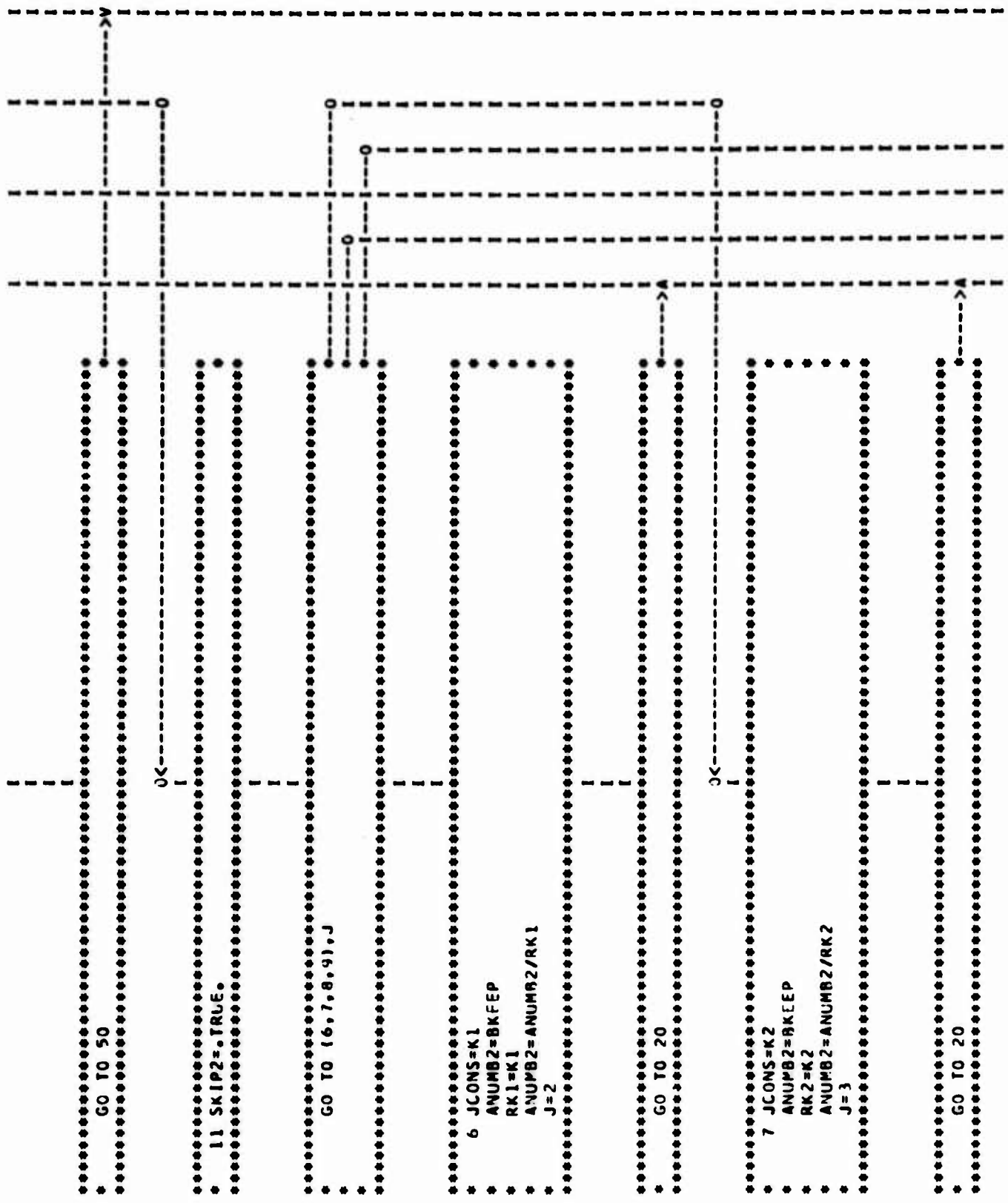


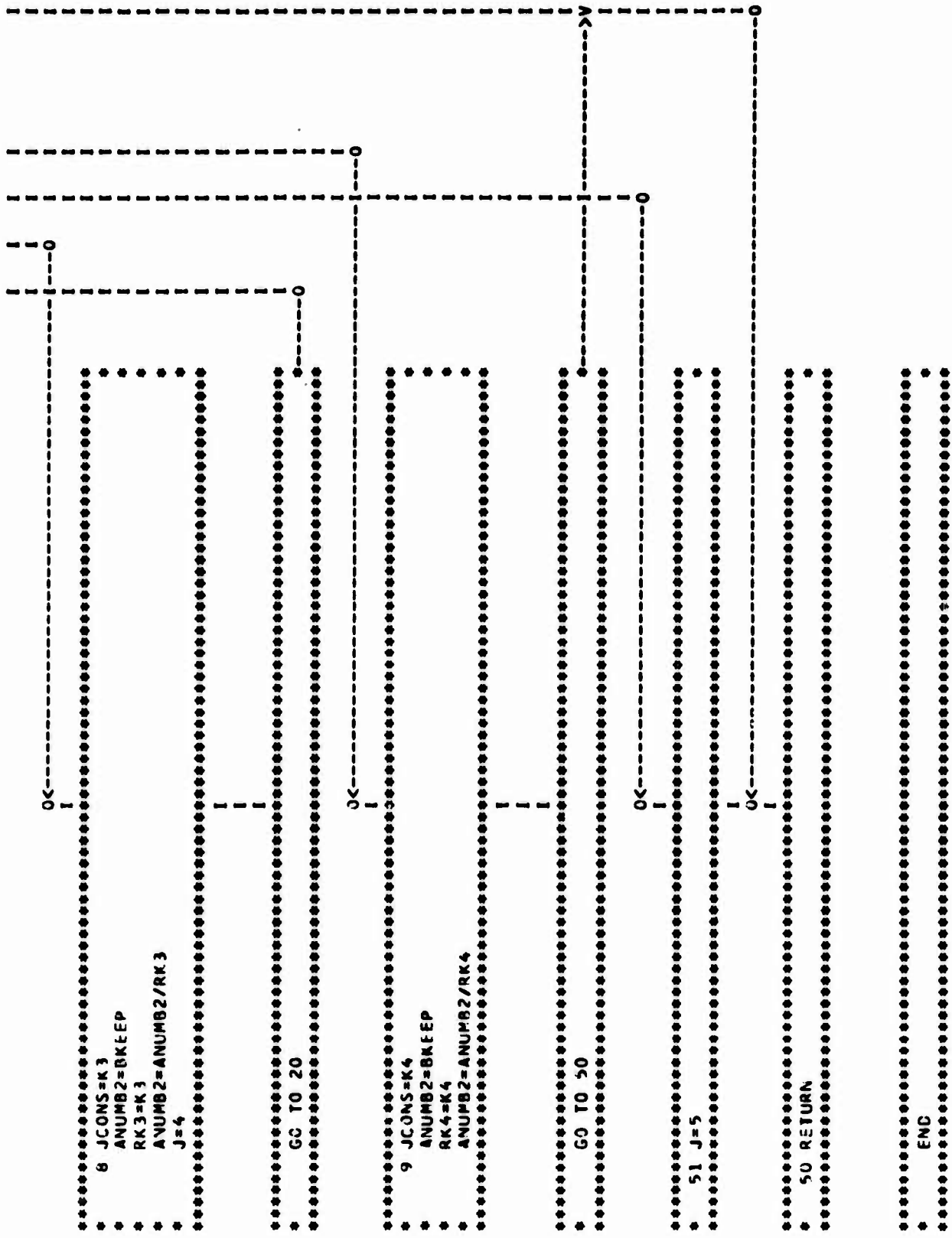


(ENTRANCE)









(ENTRANCE)

```
.....  
*C  
*C  
*C  
*C  
* SUBROUTINE #POINT(J,JCONS,L,ANUMB2,SKIP2)  
* LOGICAL SKIP2  
* L=0  
* IDELTA=0  
* IF(ANUMB2.GE.4.2) IDELTA=12  
.....
```

```
.....  
* IF(ANUMB2.GE.8.0) GO TO 200  
.....
```

```
.....  
* IF(ANUMB2.GE.6.9) GC TC 201  
.....
```

```
.....  
* IF(ANUMB2.GE.5.9) GC TC 202  
.....
```

```
.....  
* IF(ANUMB2.GE.5.0) GC TC 203  
.....
```

```
.....  
* IF(ANUMB2.GE.4.2) GC TC 204  
.....
```

```
.....  
* IF(ANUMB2.GE.3.5) GC TC 205  
.....
```

```
.....  
* IF(ANUMB2.GE.2.9) GC TC 206  
.....
```

```

.....
* IF(ANUMB2,GE,2.4) GO TO 207
.....

.....
* IF(ANUMB2,GE,1.9) GO TO 208
.....

.....
* IF(ANUMB2,GE,1.5) GC TO 209
.....

.....
* IF(ANUMB2,GE,1.2) GC TO 210
.....

.....
* GO TO 211
.....

0K-----0
.....
* 200 IF(ANUMB2,LT,9.2) IDELTA=11
.....

0K-----0
.....
* 201 IF(ANUMB2,LT,8.0) IDELTA=10
.....

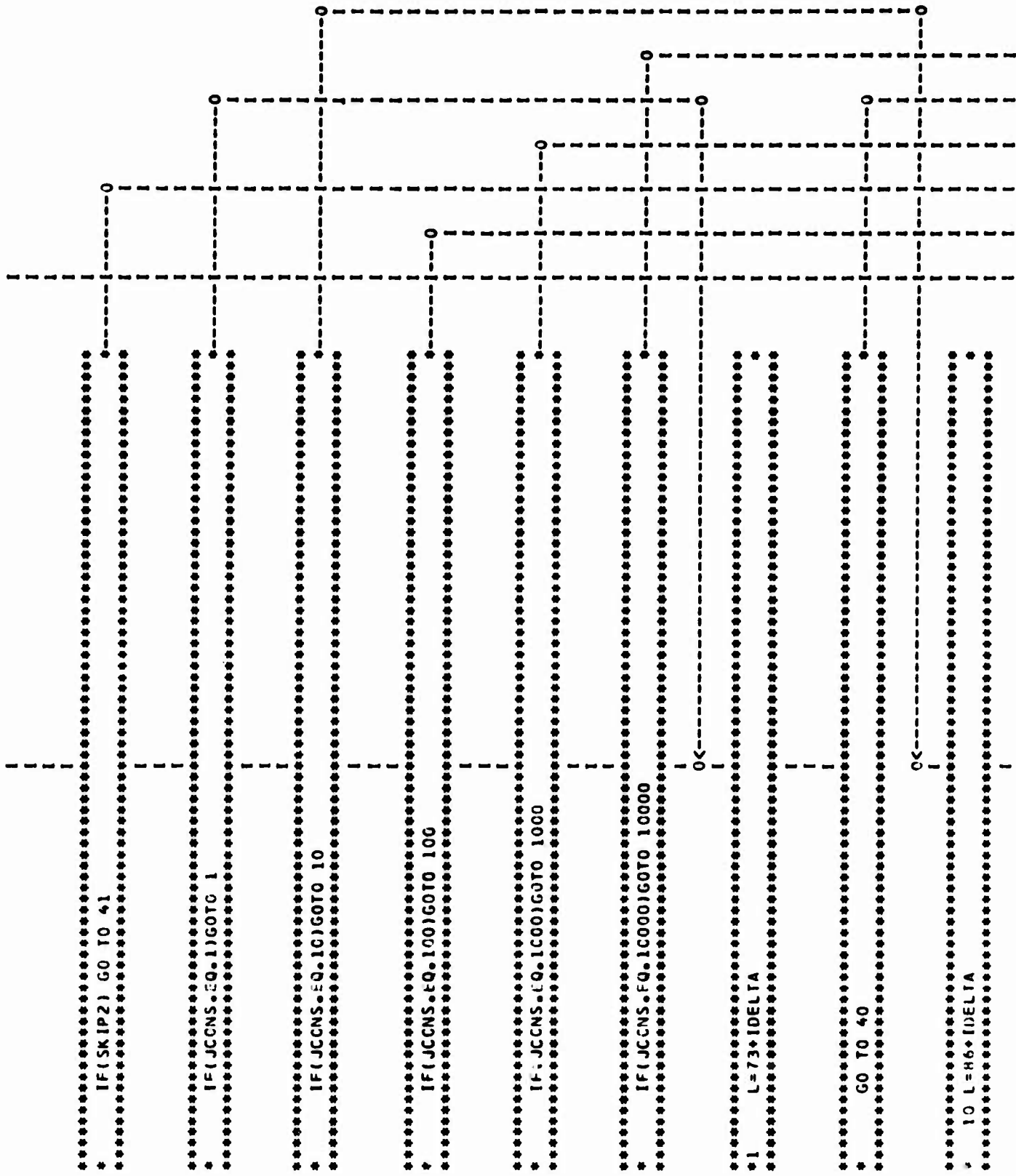
0K-----0
.....
* 202 IF(ANUMB2,LT,6.9) IDELTA=9
.....

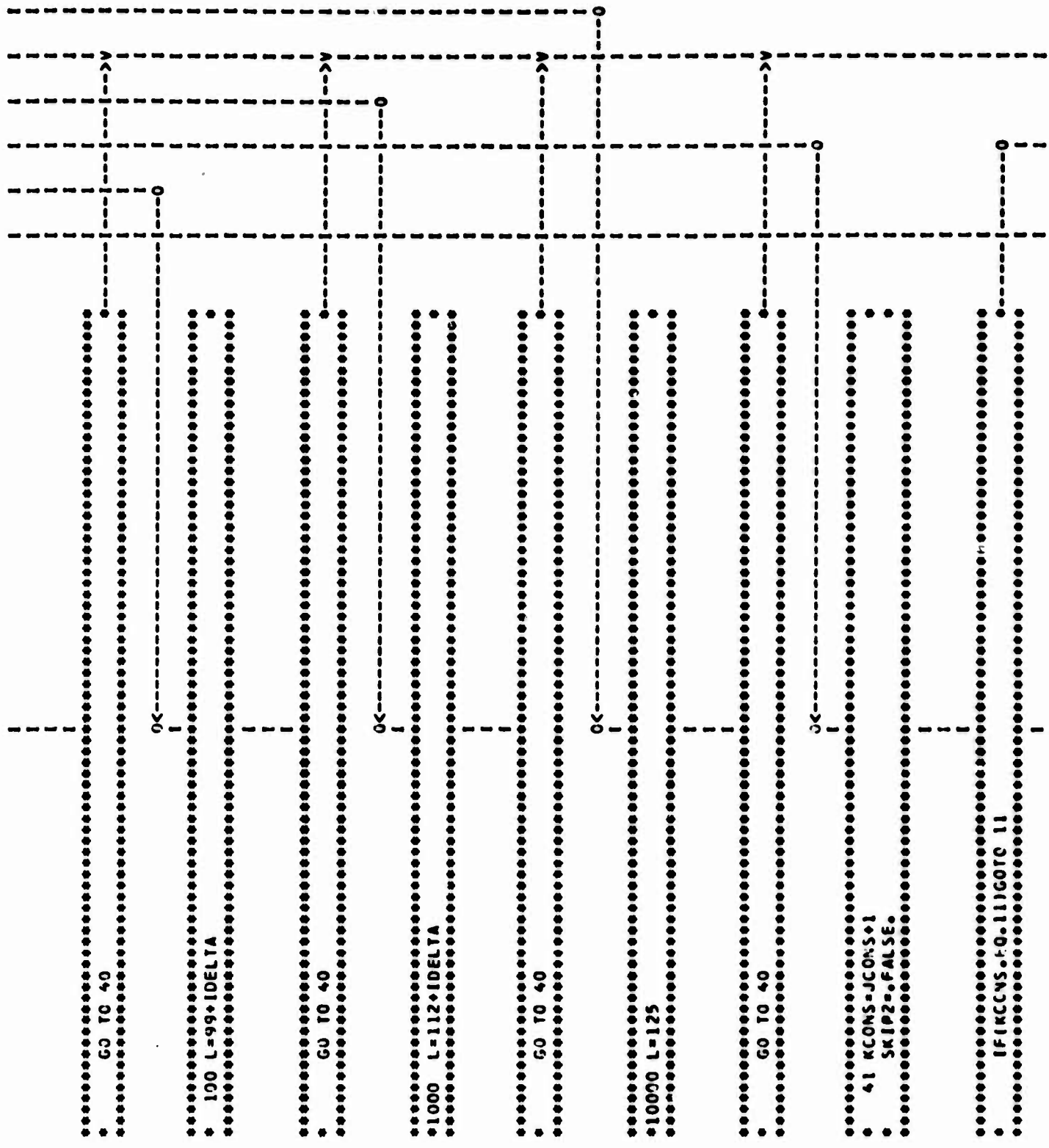
0K-----0
.....
* 203 IF(ANUMB2,LT,5.9) IDELTA=8
.....

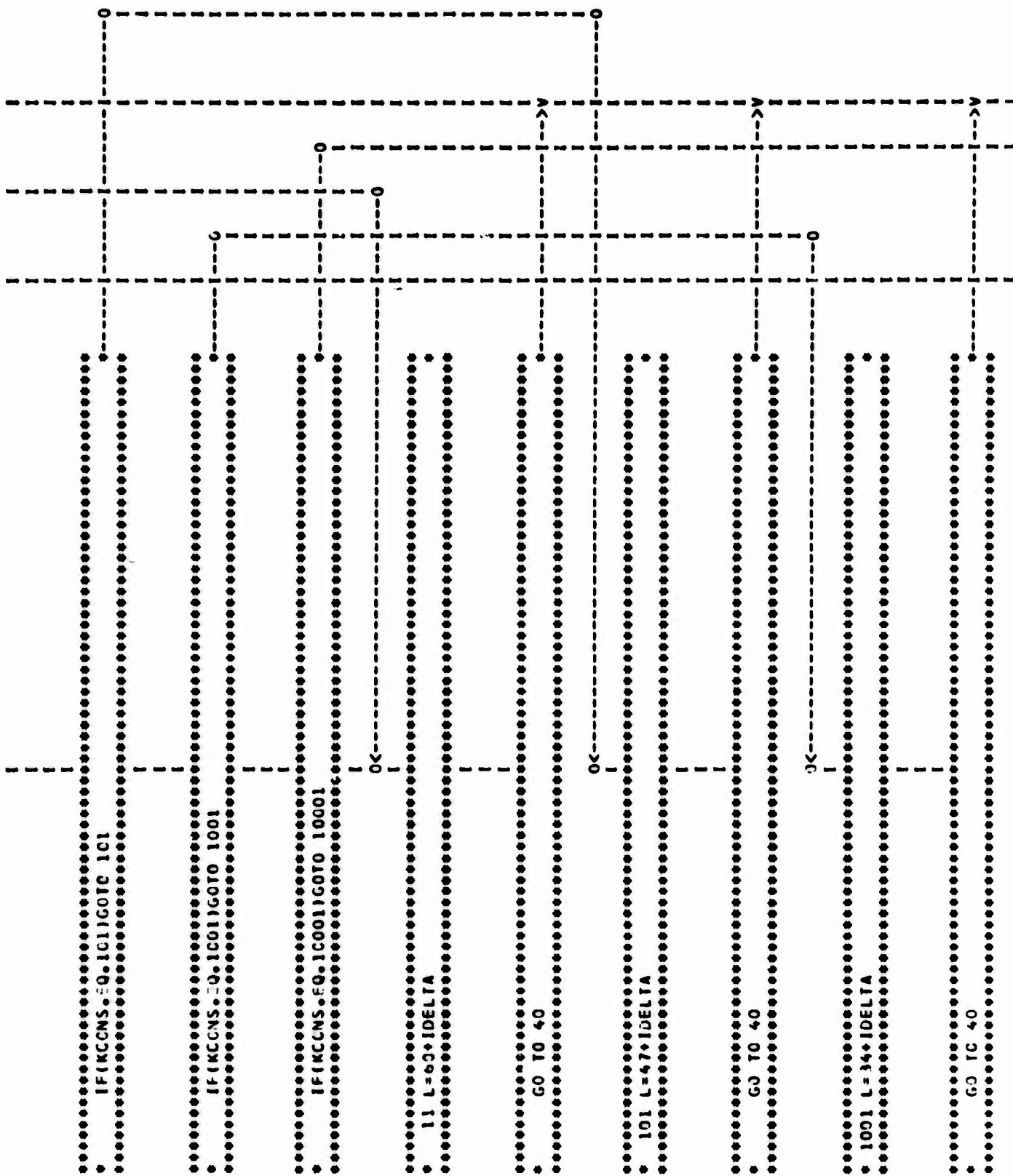
```

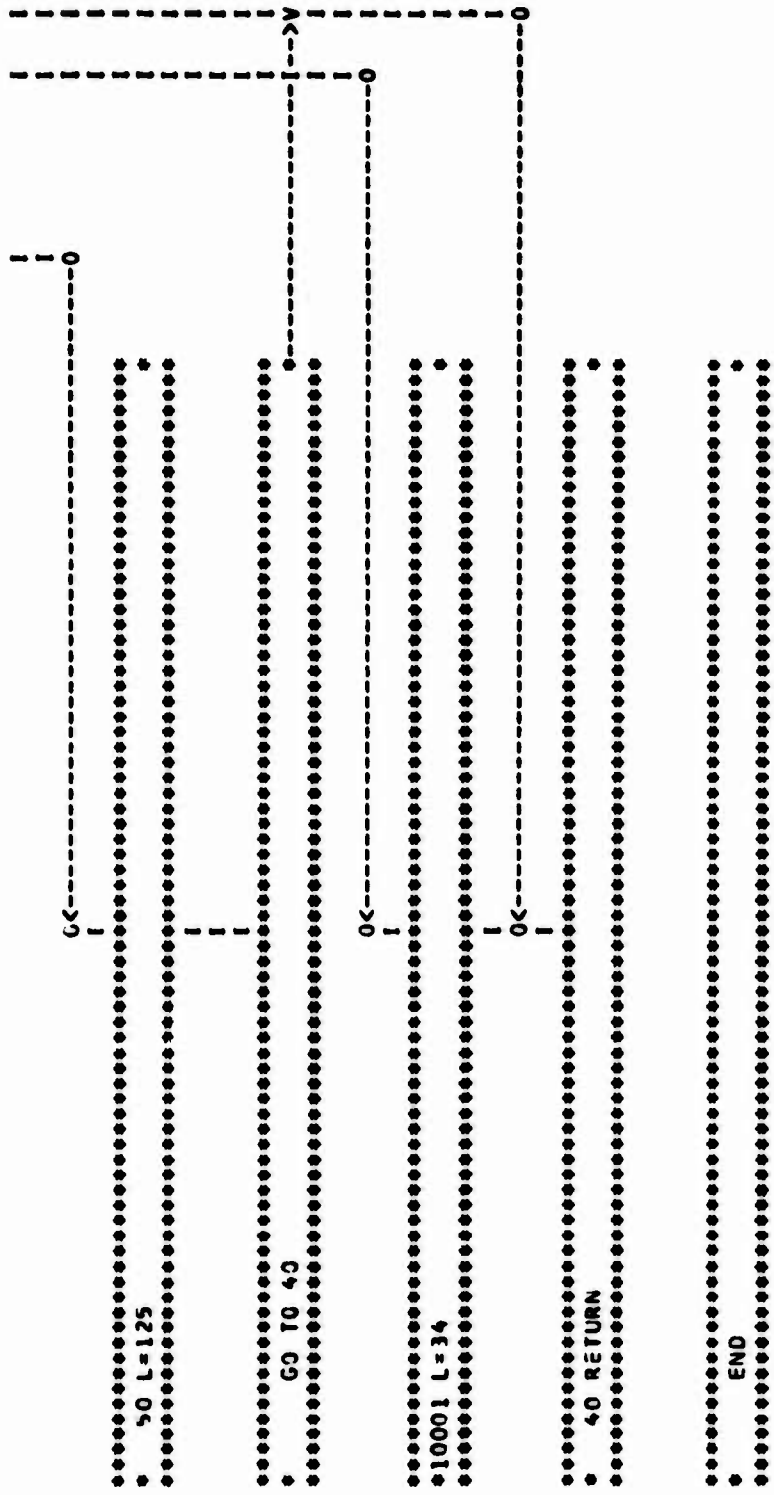


```
.....
* 204 IF(ANUMB2.LT.5.0) IDELTA=7
.....
.....
* 205 IF(ANUMB2.LT.4.2) IDELTA=6
.....
.....
* 206 IF(ANUMB2.LT.3.5) IDELTA=5
.....
.....
* 207 IF(ANUMB2.LT.2.9) IDELTA=4
.....
.....
* 208 IF(ANUMB2.LT.2.4) IDELTA=3
.....
.....
* 209 IF(ANUMB2.LT.1.9) IDELTA=2
.....
.....
* 210 IF(ANUMB2.LT.1.5) IDELTA=1
.....
.....
* 211 IF(ANUMB2.LT.1.2) IDELTA=0
* IDELTA=13-IDELTA
.....
.....
* IF(J.EQ.5) GO TO 5C
.....
```









(ENTRANCE)

```
.....  
*C .....  
*C .....  
*C .....  
*C .....  
* SUROUTINE SPCINT(I,ANUMB1,LESS,ICONS,L,POINT,SKIPI,NO,XA,XB,STAR)  
* DIMENSION XA(200),XR(200)  
* INTEGER PCINT(130,100),DASH,BLANK,STAR,C  
* INTEGER STAR1,STAR2  
* DATA STAR1/1HC/,STAR2/1H*/  
* DATA BLANK/1H/,DASH/1H-/,  
* LOGICAL SKIPI  
* AKEEP=ANUMB1  
* ANUMB1=ABS(AKEEP)  
* KOOL=C  
.....
```

```
.....  
* .....  
* IF(I.EQ.5)GO TO 100C  
* .....  
.....
```

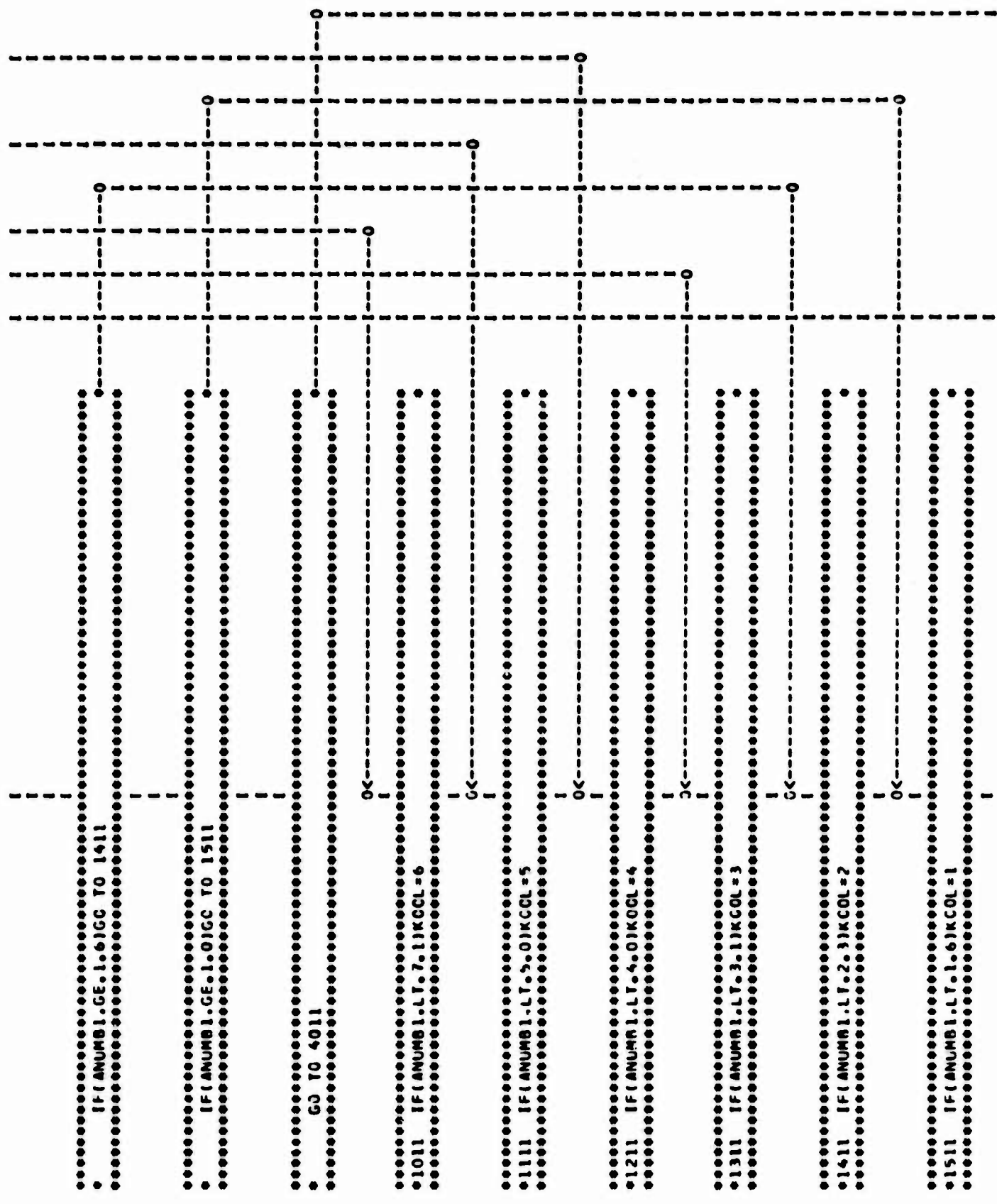
```
.....  
* IF(ANUMB1.GE.7.1)KCOL=7  
* .....  
.....
```

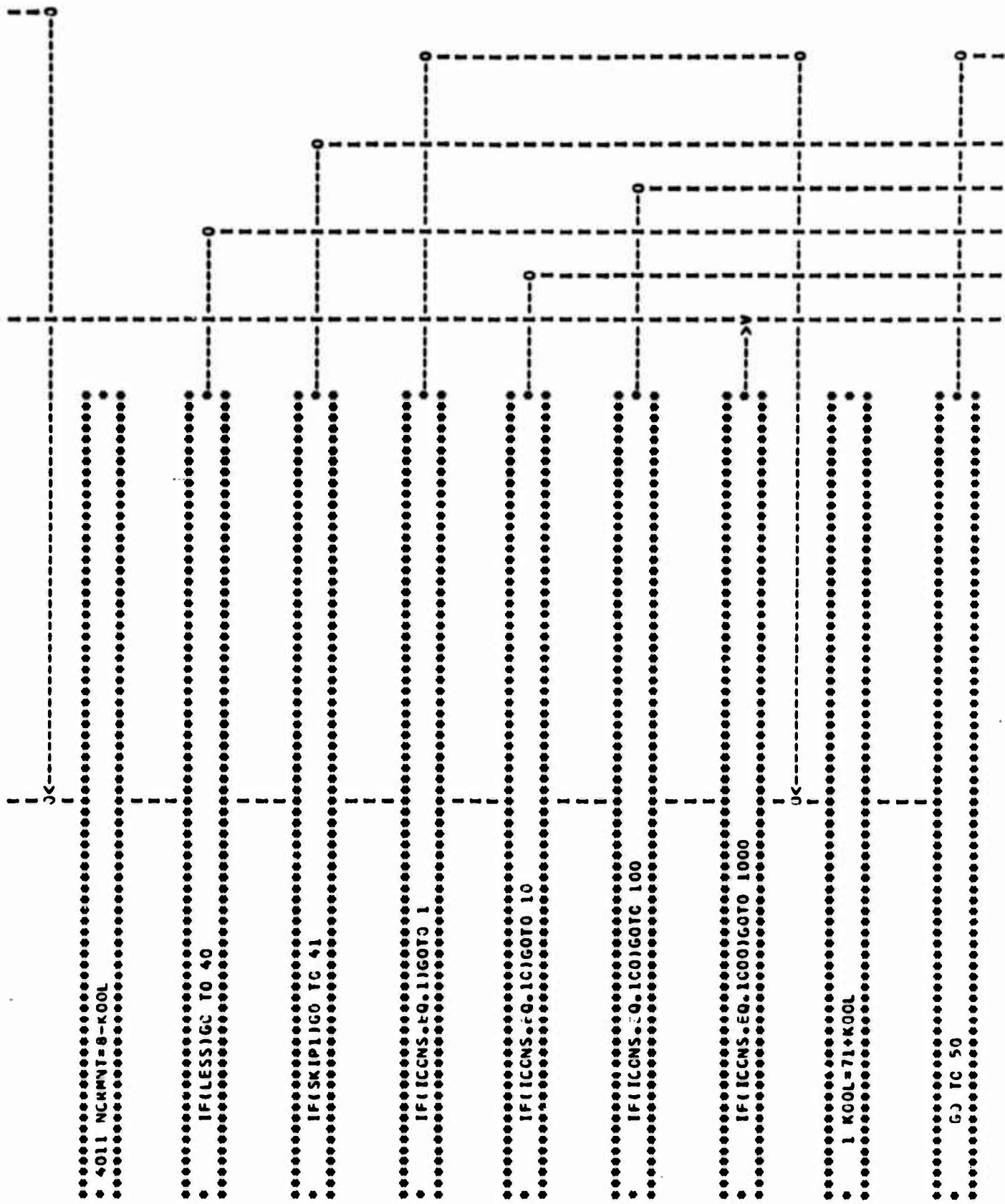
```
.....  
* IF(ANUMB1.GE.5.0)GO TO 1011  
* .....  
.....
```

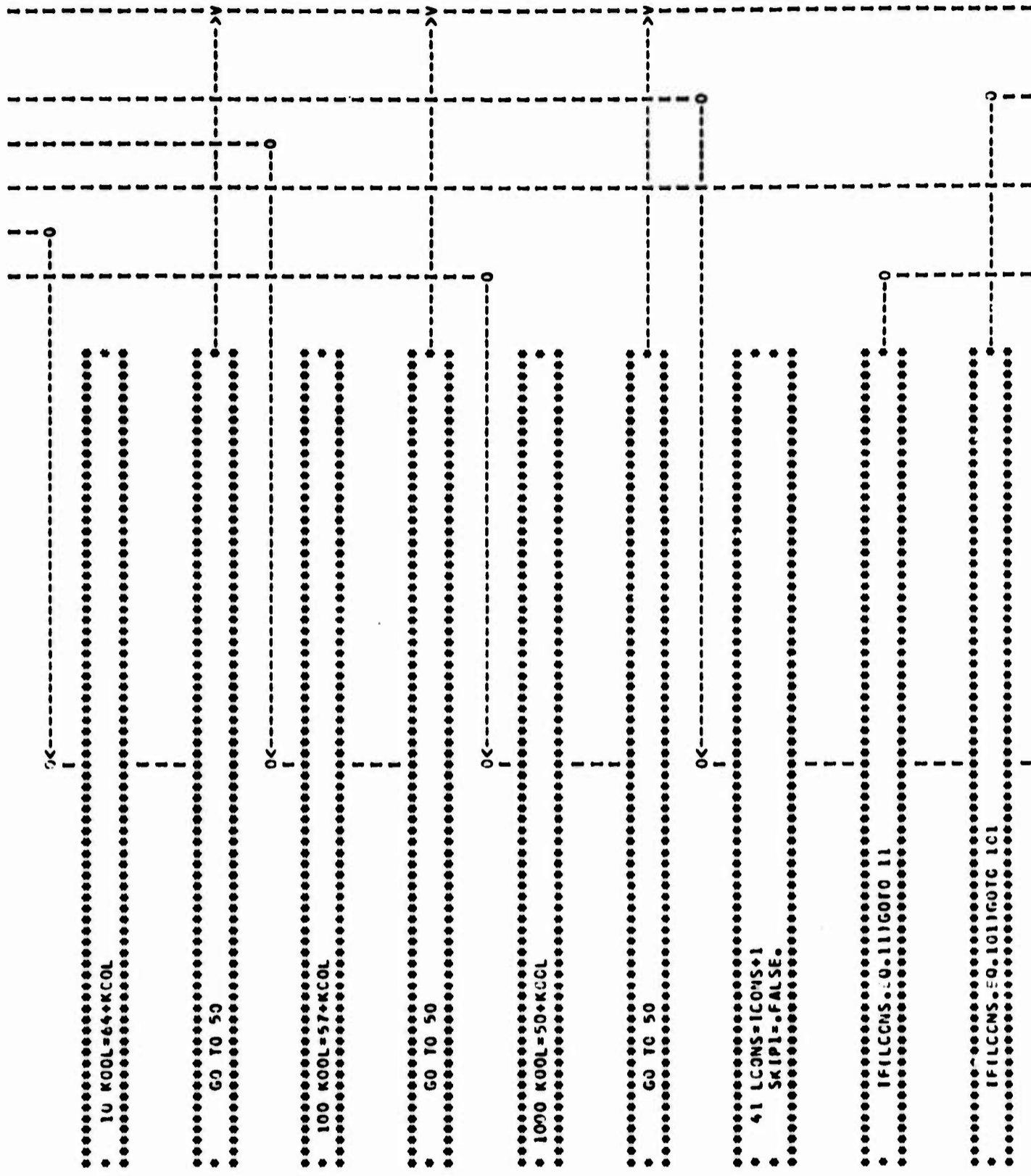
```
.....  
* IF(ANUMB1.GE.4.0)GO TO 1111  
* .....  
.....
```

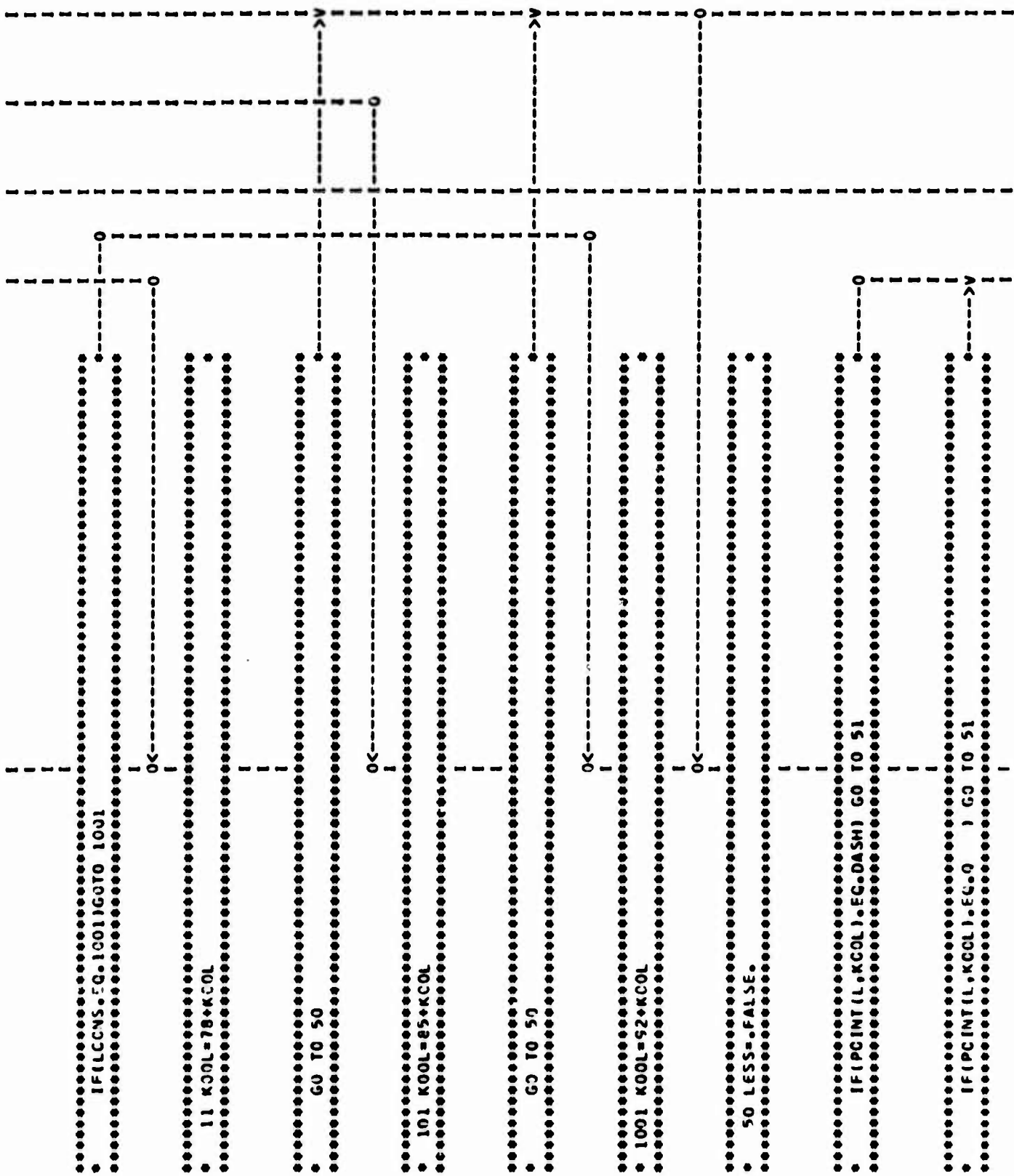
```
.....  
* IF(ANUMB1.GE.3.1)GO TO 1211  
* .....  
.....
```

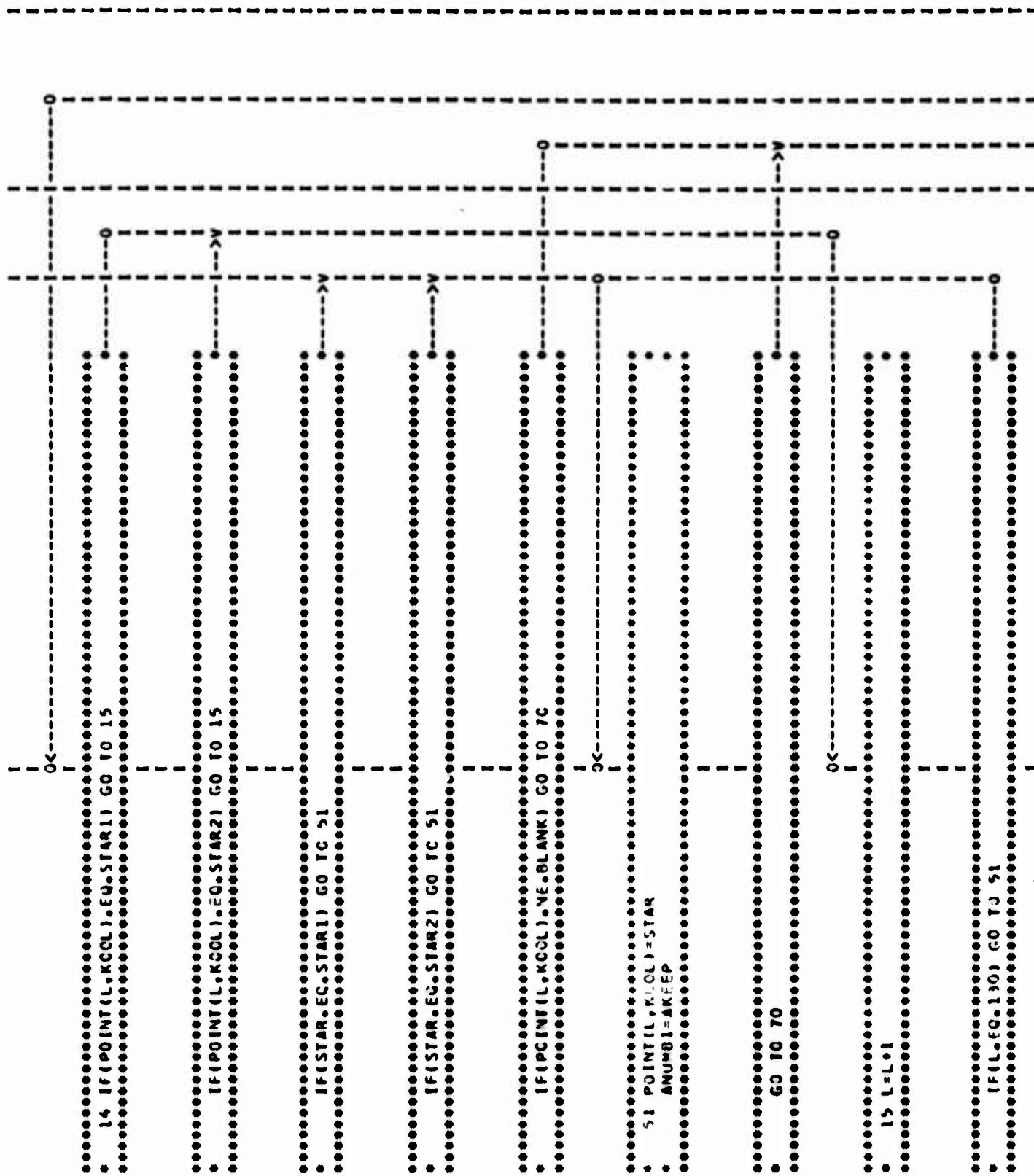
```
.....  
* IF(ANUMB1.GE.2.3)GO TO 1311  
* .....  
.....
```

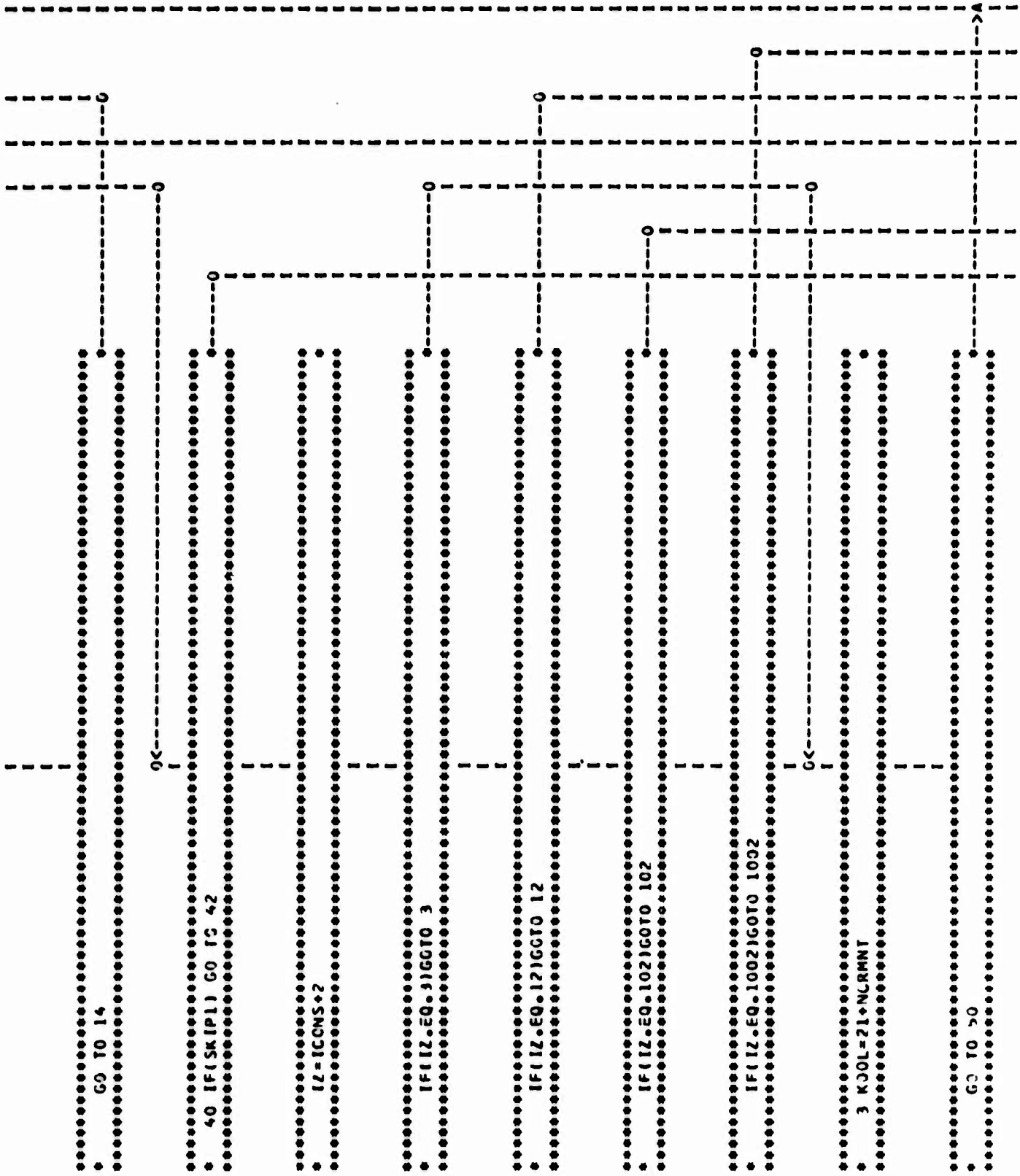


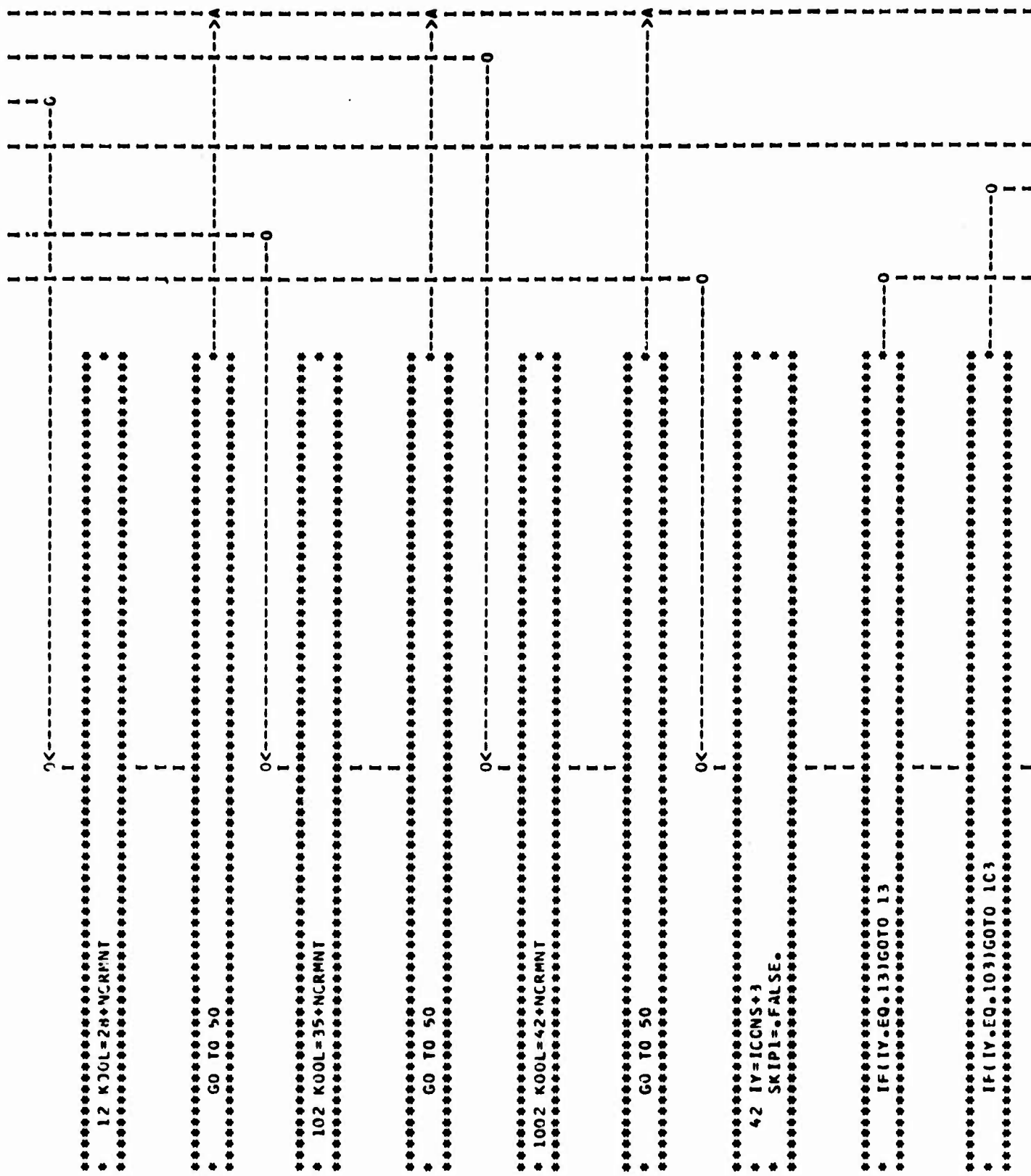


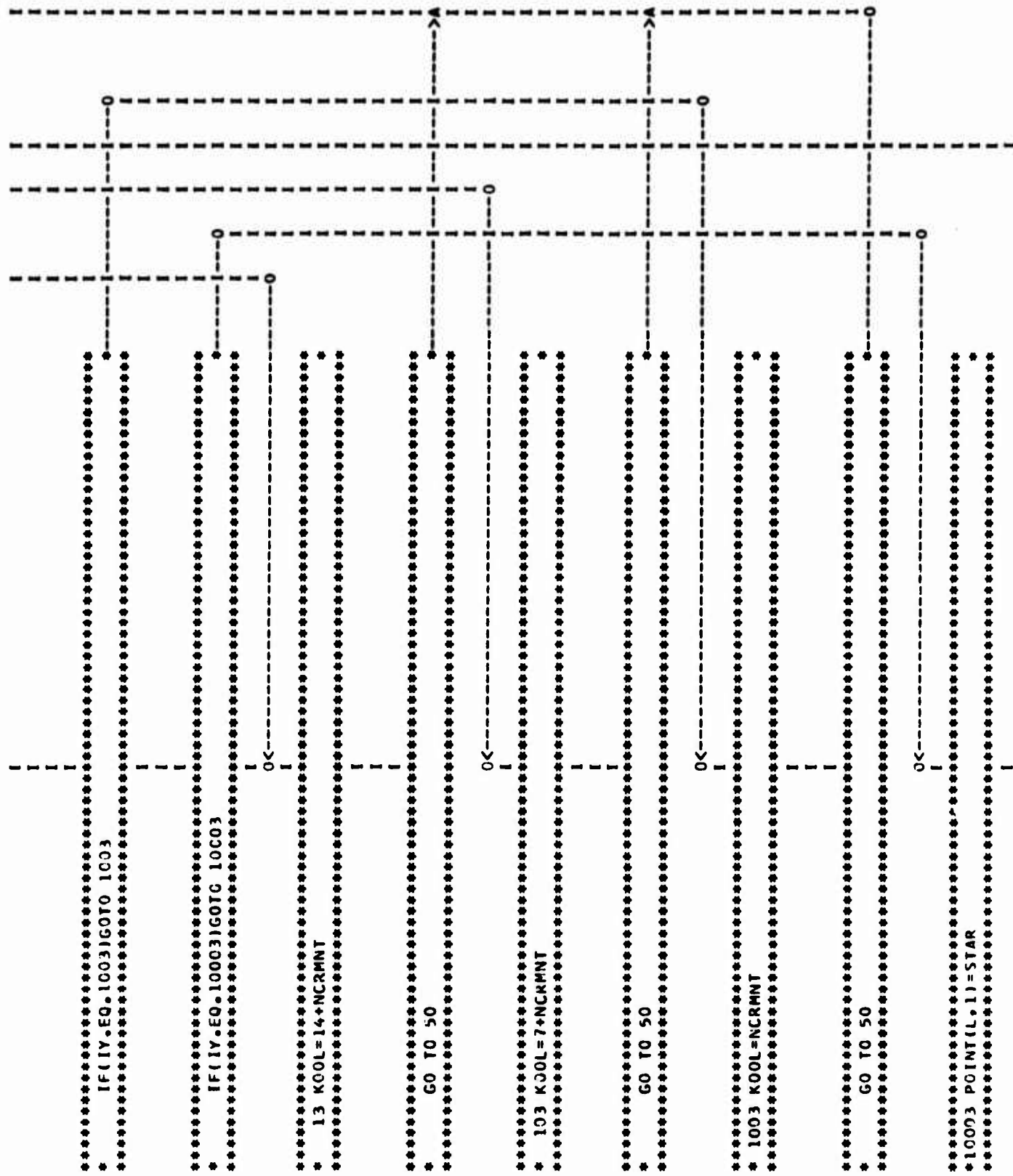


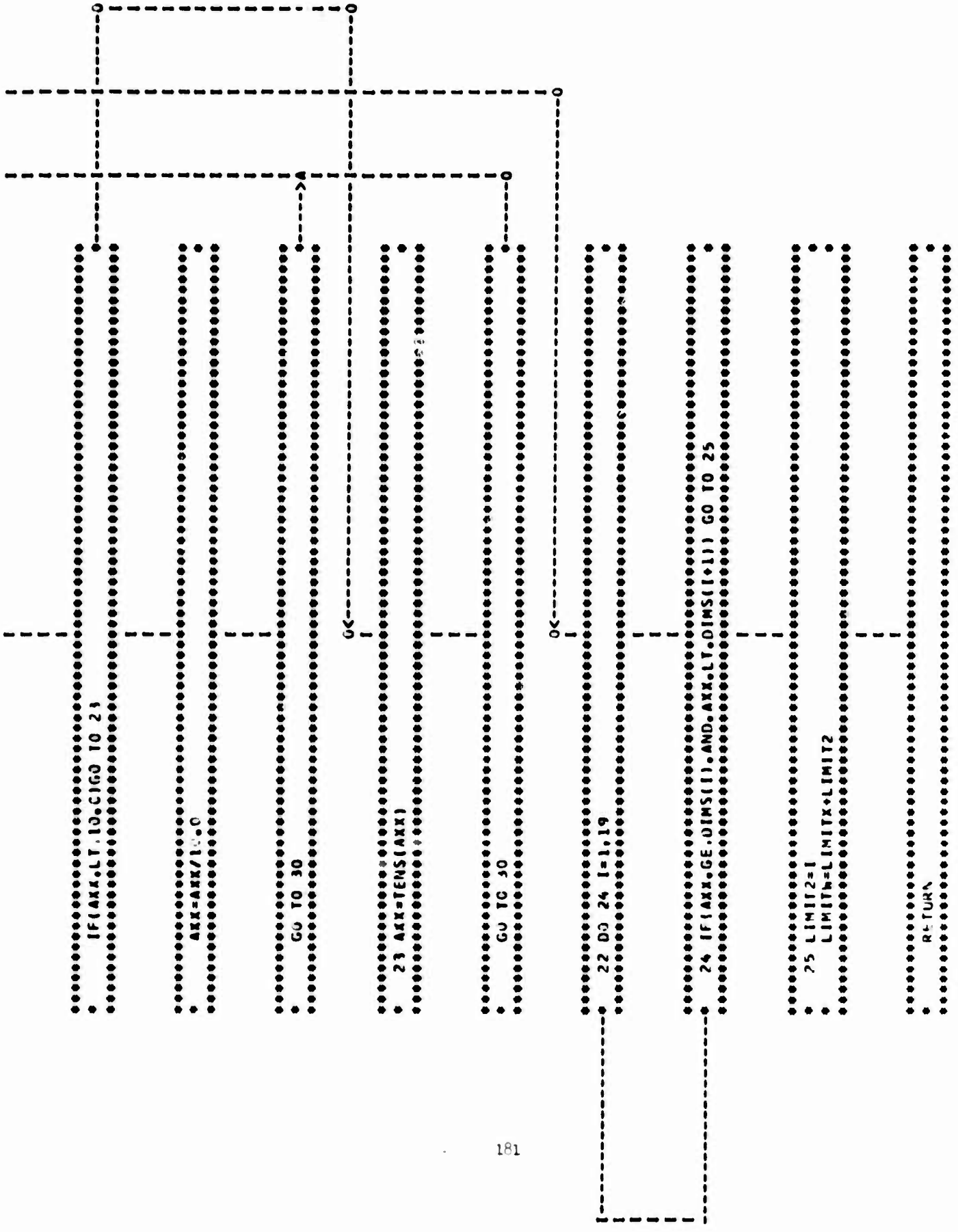












END

(ENTRANCE)

```
.....  
*C  
*C  
*C SUBROUTINE HLANKR (POINT,N1,N2,LIM1)  
*C INTEGER POINT(N1,N2),BLANK,DASH,Q  
*C COMMON /INFO4/ Q,BLANK,DASH  
*C DATA BLANK/1H /,DASH/1H-/,Q/1H/  
.....  
.....  
..... DO 51 M=1,N1  
.....  
..... DO 51 N=1,N2  
.....  
..... 51 POINT(M,N)=BLANK  
.....  
..... DO 52 M=1,N1  
.....  
..... POINT(M,1)=DASH  
.....  
..... 52 POINT(M,N2)=DASH  
.....  
..... DO 53 M=1,N2  
.....
```


(ENTRANCE)

```
*****  
*C  
*C  
* SURROUTINE NEWPLT(X,Y,Z,LET)  
* INTEGER POINT(130,100),POINTF(120,58),STAR,C  
* DIMENSION ALIM(7)  
* DIMENSION Y(3000),Z(3000)  
* DIMENSION SAVE1(100,100),SAVE2(100,100)  
* COMMON /INFO4/ Q,BLANK,DASH  
* COMMON /INFO2/ALIM  
* COMMON SAVE1,SAVE2,POINT  
* EQUIVALENCE(PCINT(1,1),POINTF(1,1))  
* DATA STAR/1H*/,Q/1H/,BLANK/1H/  
* 1 FORMAT(1H1)  
* CALL BLANKR(POINTF,120,58,120)  
*****
```

```
*****  
* DO 10 I=1,3000  
*****
```

```
*****  
* IF(Y(I).EQ.O.C) GO TO 11  
*****
```

```
*****  
* GO TO 12  
*****
```

```
*****  
* 11 IF(Z(I).EQ.O.C) GO TO 10  
*****
```

```
*****  
* 12 CALL LOGGR(Y(I),LIMITM)  
*****
```

```
*****  
* IF(LIMITM.EQ.C) GO TO 10  
*****
```

```

.....
* CALL LINEAR(X,LIMIT,Z(I))
.....
.....
* IF(LIMIT,EO.0) GO TO 10
.....
* POINT(LIMIT,LIMIT)=STAR
.....
.....
10 CONTINUE
.....
.....
DO 40 I=2,100,20
.....
DO 40 J=1,58
.....
.....
40 POINT(I,J)=Q
.....
.....
* WRITE(6,1)
* 30 FORMAT(1X,6(IPIF10.1,1X,3MR/5.6X),1PIE1C.1)
* J=X
* WRITE(6,51)J,LET,(PCINTF(1,1),I=1,120)
* 51 FORMAT(1X,14, A4,1X,12CA1)
* WRITE(6,20)((POINT(I,J),I=1,120),J=2,57)
* J=-X
* WRITE(6,51)J,LET,(POINT(I,58),I=1,120)
* 20 FORMAT(10X,12CA1)
* WRITE(6,3)JALPH
.....

```

1
1
1
1

.....
* R F T U R N *
.....

.....
* E N D *
.....

(ENTRANCE)

```
.....  
*C  
*C  
*C SUMMULTINE LINEAR(X,LIMITY,Y)  
*C DIMENSION ENCRMT(59)  
*C LIMITY=0  
*C FLIMIT=X*2.  
*C DELTA=FLIMIT/58.  
*C ENCRMT(1)=X  
.....
```

```
.....  
DO 10 I=2,58  
.....
```

```
.....  
10 ENCRMT(I)=ENCRMT(I-1)-DELTA  
.....
```

```
.....  
ENCRMT(59)=-X  
.....
```

```
.....  
DO 11 I=1,58  
.....
```

```
.....  
11 IF(Y.LE.ENCRMT(I).AND.Y.GE.ENCRMT(I+1))GO TO 20  
.....
```

```
.....  
GO TO 40  
.....
```

```
.....  
20 LIMITY=I  
.....
```

! !
! !
OK-----0

.....
* 40 RETURN
.....

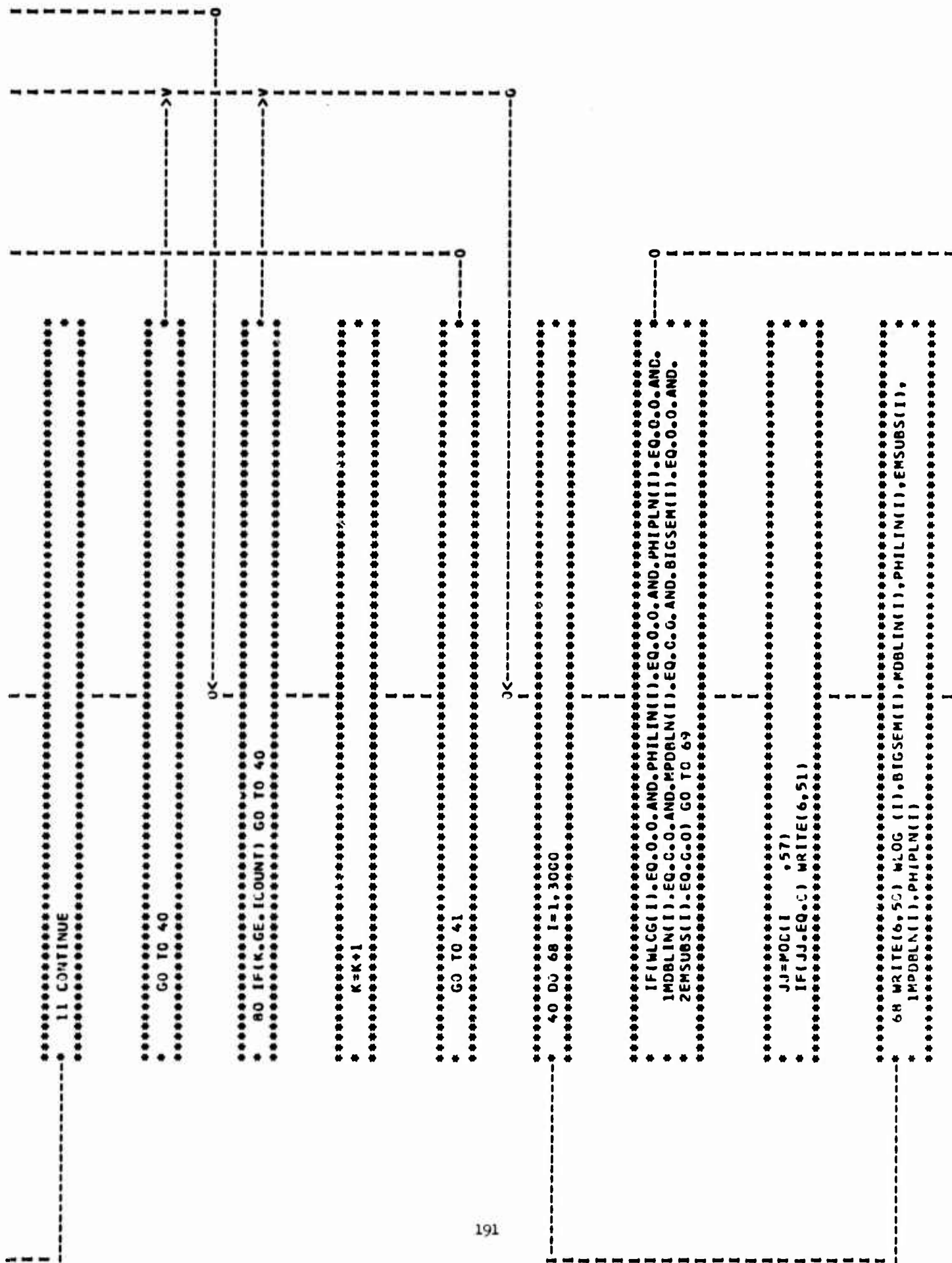
.....
* END
.....

(ENTRANCE)

```

*****
*C
*C
* SURROUTINE SWEEP(XOMEGA)
* DIMENSION OMEGA(17),SWEPER(17)
* COMMON/INFO3/OMEGA
* DATA (SWEPER(1),I=1,17) /0.01,0.05,0.1,0.2,0.3,0.4,0.5,0.6,0.7,
* 0.8,0.9,1.0,2.0,4.0,6.0,8.0,10.0/
*****
I
I
*****
DO 70 K=1,17
*****
I
I
I
I
I
I
*****
70 OMEGA(K)= SWEPER(K)*XOMEGA
*****
I
I
I
*****
* RETURN
*****
*****
* END
*****

```



I
I
O<-----O

.....
* 69 WRITE(6,14)
* CALL NEMPT(DEGREE,WLOG,PHILIN,LET2)
.....

.....
* IF(IEXP(1).EQ.0) GO TO 30
.....

.....
* JJ=IEXP(1)
.....

.....
* DO 34 I=1,JJ
.....

.....
* 34 CALL EXPLOT(WLOG,300,PHILIN,3000,DEGREE,LET2)
.....

I
I
O<-----O

.....
* 30 CALL NEMPT(DBLIM,WLOG,MSBLIM,LET1)
.....

.....
* IF(IEXP(2).EQ.0) GO TO 31
.....

.....
* JJ=IEXP(2)
.....

.....
* DO 35 I=1,JJ
.....

I
I
O<-----O

```

15 CALL EXPLT(WLOG, ICCO, MDRLIN, 3000, DRILM, LET1)
16 CALL NEPLT(DRLIM, WLOG, MPDBLM, LET1)
17 DO 31 I=1, JJ
18     WRITE(6, 15)
19     CALL NEPLT(DEGREE, WLOG, PHIPLN, LET2)
20 IF (IEXP(3).EQ.0) GO TO 32
21 JJ=IEXP(3)
22 DO 36 I=1, JJ
23     CALL EXPLT(WLOG, 3000, PHIPLN, 3000, DEGREE, LET2)
24     CALL NEPLT(DRLIM, WLOG, MPDBLM, LET1)
25     JJ=IEXP(4)
26 DO 37 I=1, JJ

```

```

31 WRITE(6, 15)
32 CALL NEPLT(DEGREE, WLOG, PHIPLN, LET2)
33 IF (IEXP(3).EQ.0) GO TO 32
34 JJ=IEXP(3)
35 DO 36 I=1, JJ
36     CALL EXPLT(WLOG, 3000, PHIPLN, 3000, DEGREE, LET2)
37     CALL NEPLT(DRLIM, WLOG, MPDBLM, LET1)
38     JJ=IEXP(4)
39 DO 37 I=1, JJ

```

```

31 WRITE(6, 15)
32 CALL NEPLT(DEGREE, WLOG, PHIPLN, LET2)
33 IF (IEXP(3).EQ.0) GO TO 32
34 JJ=IEXP(3)
35 DO 36 I=1, JJ
36     CALL EXPLT(WLOG, 3000, PHIPLN, 3000, DEGREE, LET2)
37     CALL NEPLT(DRLIM, WLOG, MPDBLM, LET1)
38     JJ=IEXP(4)
39 DO 37 I=1, JJ

```

```

31 WRITE(6, 15)
32 CALL NEPLT(DEGREE, WLOG, PHIPLN, LET2)
33 IF (IEXP(3).EQ.0) GO TO 32
34 JJ=IEXP(3)
35 DO 36 I=1, JJ
36     CALL EXPLT(WLOG, 3000, PHIPLN, 3000, DEGREE, LET2)
37     CALL NEPLT(DRLIM, WLOG, MPDBLM, LET1)
38     JJ=IEXP(4)
39 DO 37 I=1, JJ

```

```

I
I
I
-----
*****
* 37 CALL EXPLGT(MLOG,1000,MPDRLN,3000,CBLIM,LE11)
*****
I
I
I
-----
*****
* IF(IEXP(4).EQ.0) GO TO 33
*****
I
I
I
-----
*****
* 51 FORMAT(1H1,5X,5HOMEGA,13X,3H/M/,16X,6H/M/-DB,10X,3HPHI,20X,4H/M/P,
* 10X,7H/M/P-DB,14X,5HPHI-P,/)
* 50 FORMAT(2X,1PIE12.5,5X,1PIE12.5,6X,1PIE12.5,5X,1PIE12.5,11X,
* 1PIE12.5,3X,1PIE12.5,9X,1PIE12.5)
* 14 FORMAT(1H1,//////////,25X,60HTHE FOLLOWING PLOTS ARE NOR
* 5AL FREQUENCY RESPONSE DATA
* 15 FORMAT(1H1,//////////,25X,36HTHE FOLLOWING PLOTS ARE POP
* 0V DATA
*****
I
I
I
-----
*****
* 33 RETURN
*****
I
I
I
-----
*****
* END
*****

```

(ENTRANCE)

```
*****  
*C  
*C  
* SUBROUTINE FIGUR (KONT,OMEGA,KOUNT)  
* DIMENSION SAVE(100,100),SAVE2(100,100)  
* DIMENSION ACOEFF(100),RCOEFF(100),AAAAA(99),BBBBB(99)  
* DIMENSION WLOG(300),PHILN(300),MDRLN(300),MPIPLN(300),  
* MPDBLN(300),BIGSEM(300),EMSUBS(300),OMEGA(KONT)  
* REAL MDRLN,MPDRLN  
* INTEGER P,INT(130,100)  
* COMMON/FREK/ACOEFF,RCOEFF  
* EQUIVALENCE(AAAAA1),ACOEFF(2),(BBBBB1),BCOEFF(2)  
* COMMON/INFO6/EMSURS  
* COMMON SAVE1,SAVE2,POINT  
* EQUIVALENCE (WLOG1),SAVE1(1),(PHILN1),SAVE2(1),  
* 1 (SAVE1(300)),MDRLN(1),(SAVE2(300)),MPIPLN(1),  
* 2 (SAVE1(600)),MPDBLN(1),(SAVE2(600)),BIGSEM(1)  
*****
```

```
*****  
*C  
*C  
* DO 21 I=1,KONT  
*****
```

```
*****  
* CALL FIGURE(OMEGA(1),ACOEFF(1),AAAAA,EMOFAS,PHIAS)  
* CALL FIGURE(OMEGA(1),RCOEFF(1),BBBBB,EMOFAS,PHIBS)  
* BIGEM=ABS(EMCFAS/EMOFBS )  
* WLOG(KONT)=BIGEM  
* PHIENCK=PHIAS-PHIBS  
* IF(PHIEND.LT.-180.) PHIEND=PHIEND+360.  
* IF(PHIEND.GT. 180.) PHIEND=PHIEND-360.  
* PHILN(KONT)=PHIEND  
* PHIENC = PHIEND * C.0174533  
* REEL = BIGEM * CJS ( PHIEND )  
* ETMAG= BIGEM * SIN ( PHIEND )  
* EMSUBP = SORT ( ( REEL * 2 ) * ( OMEGA ( 1 ) * EIMAG ) * 2 ) )  
* EMSUBS(KONT)=EMSUBP  
* QUANZ=OMEGA(1) * EIMAG  
* PHIPEL=ATAN2(QUANZ,REEL)  
* PH:EE = PHIPEE * 57.2957795131  
* IF(PHIPEE .LT. 0.) PHIPEE = PHIPEE + 360.  
* IF(PHIPEE.LT.-180.) PHIPEE=PHIPEE+360.  
* IF(PHIPEE.GT. 180.) PHIPEE=PHIPEE-360.  
* MPIPLN(KONT)=PHIPEE  
* DEEBE=20.0*(ALOG10(BIGEM))  
* MDRLN(KONT)=DEEBE  
* DEEBEP=20.0*(ALOG10(EMSUMP))  
* MPDRLN(KONT)=DEEBEP  
* KOUNT=KONT+1  
*****
```

21 CONTINUE

RETURN

END

(ENTRANCE)

```
.....  
C  
C  
SUBROUTINE FIGURE(X,Y,Z,EMCFAS,PHIAS)  
DIMENSION Z(99)  
REFELAS=Y  
I=2  
A=-1.0  
.....
```

```
.....  
21 REELAS=(Z(1)*(X+1)+A)*REELAS  
.....
```

```
.....  
25 I=I+2  
.....
```

```
.....  
IF(I.GT.99) GO TO 22  
.....
```

```
.....  
A=-A  
.....
```

```
.....  
IF(Z(1).EQ.0.)GO TO 25  
.....
```

```
.....  
GO TO 21  
.....
```

```
.....  
27 I=1  
A=1.0  
ALMAGA=0.0  
.....
```



```

.....
* 24 AIMAGA=(Z(I)*(X**I)**A)+AIMAGA
.....
.....
* 26 I=I+2
.....
.....
* IF(I.GT.94 ) GO TO 23
.....
.....
* A=-A
.....
.....
* IF(Z(I).EQ.0.)GO TO 26
.....
.....
* GO TO 24
.....
.....
* 23 EMOFAS=SQRT((REELAS**2)+(AIMAGA**2))
* PHIAS=ATAN2(AIMAGA,REELAS)
* PHIAS = PI-IAS*57.2957795131
* IF(PHIAS.LT.0.) PHIAS=PHIAS+360.
.....
.....
* RETURN
.....

```

.....
• END •
.....

(ENTRANCE)

```
*****  
C  
C  
C SURROUTINE LGGER(XX,LIMITM)  
C DIMENSION ALIM(7),DIM5(20)  
C EQUIVALENCE(WASTE,AXX)  
C COMPON/INFC2/ALIP  
C TENSIX)=R*10.C  
C DATA (DIM5(1),1-1,20) /1.C,1.15,1.3,1.5,1.7,1.95,2.25,2.5,2.85,  
C 3.15,3.5,3.95,4.5,4.85,5.4,6.05,6.8,7.7,8.8,9.99999/  
C LIMIT1=0  
C LIMITX=0  
C LIMIT2=0  
C*****
```

```
*****  
C  
C  
C DO 20 I=1,6  
C*****
```

```
*****  
C  
C  
C AA=2*(I-1)  
C*****
```

```
*****  
C  
C  
C 20 IF(XX.LT.ALIM(I+1).AND.XX.GE.ALIM(I)) GO TO 21  
C*****
```

```
*****  
C  
C  
C RETURN  
C*****
```

```
*****  
C  
C  
C 21 LIMITX=TENSIXA)  
C WASTE=ABS(XX)  
C IF(WASTE>10.0) RETURN  
C*****
```

```
*****  
C  
C  
C 30 IF(AXX.GE.1.C.AND.AXX.LT.10.0) GO TO 22  
C*****
```

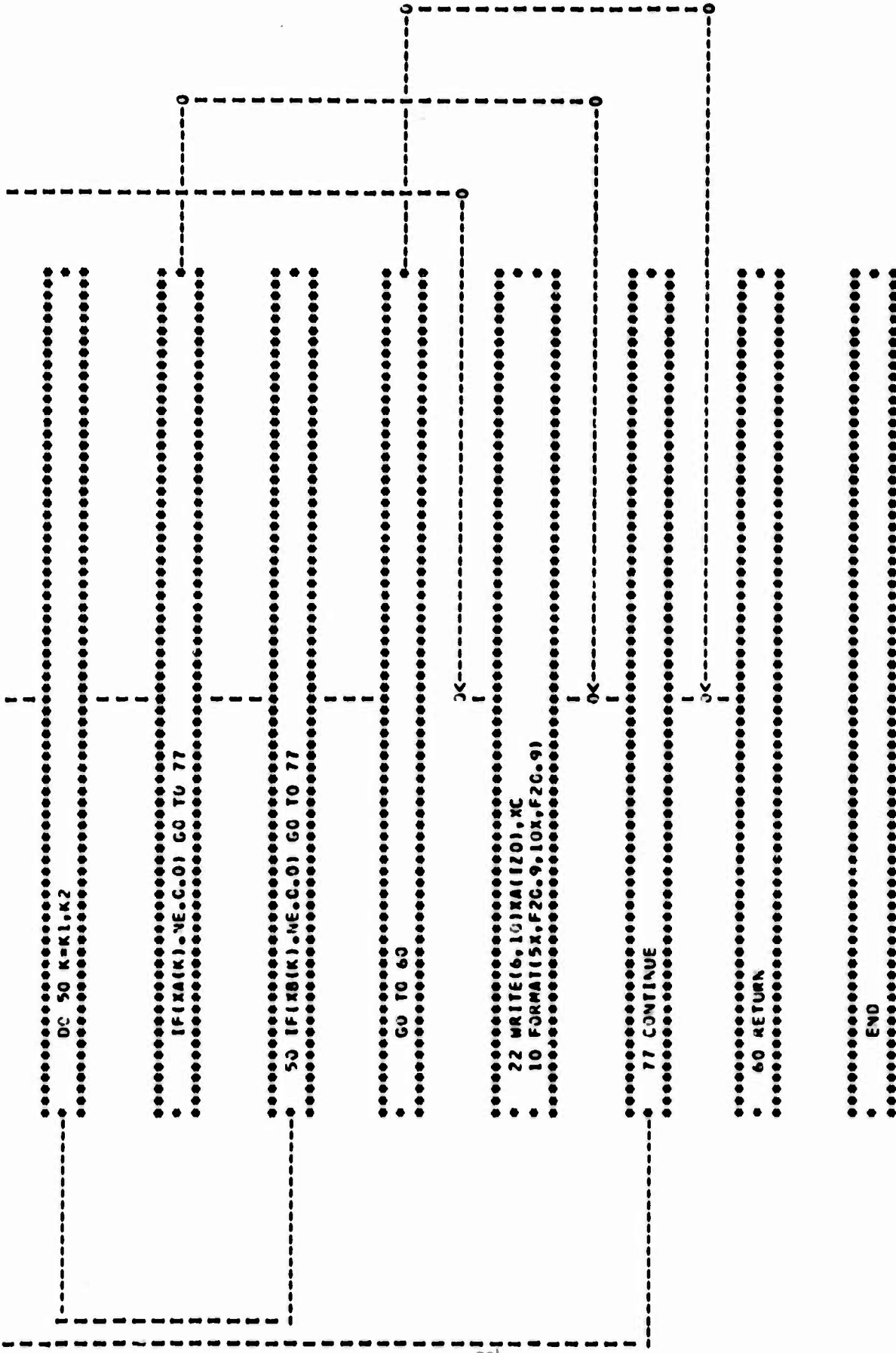
(ENTRANCE)

```
.....  
*C  
*C  
*C  
*C  
.....  
SURROUTINE PREPAR(POINT)  
INTEGER POINT(130,100),PNT  
DATA PNT /6/  
WRITE(6,1)  
1 FORMAT(1H1)  
WRITE(4,11)  
11 FORMAT(//)  
WRITE(PRT,14)  
14 FORMAT(75X,8HLOG PLOT,/58X,42HCOMPLEX FREQUENCY PLANE,LEFT HAND QU  
*ADRANT,/75X,9H(RAD/SEC) )  
WRITE(6,12)  
12 FORMAT(32X,5H1000,9X,4H1000,9X,3H100,1CX,2H10,12X,1H1,11X,2H.1,10  
*X,14H.01C---J-OMEGA,/ 18X,11MINUS SIGMA)  
13 FORMAT(32X,5H1000,9X,4H1000,9X,3H100,1CX,2H10,12X,1H1,11X,2H.1,10  
*X,14H.01C---J-OMEGA,/ 18X,11H PLUS SIGMA)  
.....  
.....  
DO 50 I=1,130  
.....  
50 POINT(I,50)=POINT(I,51)  
.....  
.....  
WRITE(PRT,10)POINT  
10 FORMAT(1X,130A1)  
WRITE(6,11)  
WRITE(PRT,15)  
15 FORMAT(75X,8HLOG PLOT,/58X,42HCOMPLEX FREQUENCY PLANE,RIGHT HAND QU  
*ADRANT,/75X,9H(RAD/SEC) )  
.....  
.....  
RETURN  
.....
```

.....
EVD
.....

(ENTRANCE)

```
.....  
*C  
*C  
*C  
*C  
*C  
.....  
SUBROUTINE WRITTT(XA,XB)  
DIMENSION KA(2000),KB(2000)  
WRITE(6,1)  
1    FORMAT(1H1)  
.....  
11  FORMAT(5X,40#THE FOLLOWING ROOTS ARE PLOTTED ON THE LOG PLOT,/  
15X,99#ROOTS AT THE ORIGIN ARE NOT PLOTTED OR PLOTTED, ROOTS ON THE  
* J-OMEGA AXIS ARE NOT PLOTTED.  
2, //16X, 5#SIGMA, 25X, 7#J-OMEGA, //)  
.....  
.....  
DO 77 IZ0=1,2000  
.....  
.....  
XC=XB(IZ0)  
IF(XC.GT.0) XC=-XC  
.....  
.....  
IF(XC.NE.0) GO TO 22  
.....  
.....  
20  IF(XA(IZ0))22,40,22  
.....  
.....  
40  K1=IZC  
    K2=IZC+12  
.....  
.....
```



(ENTRANCE)

```

.....
*
* SUBROUTINE FRE.FRS(SIGMAX,OMEGAY)
*
* DIMENSION SAVE1(-00,100),SAVE2(100,100),ACOFF(100),WCNEFF(100),
*   AAAA(19),B99(99),SIGMAX(2000),OMEGAY(2000),DEC(9),
*   OMEGAS(9),OMEGA(17),W_0G(3000),PHILIN(3000),MDBLIN(3000
*   ),PHIPLN(3000),MPDBLN(3000),ALIM(7),BIGSEFM(3000),
*   EMSUBS(3000)
*
* REAL MDBLA,MPDBLN
* INTEGER FMRPAD,IERF(4)
* INTEGER PCINT(130,100)
* COMMON/FREK/ACOFF,WCNEFF
* COMMON/TNE02/ALIM
* COMMON/TNE03/OMEGA
* COMMON/TNE06/EMSUBS
* COMMON/TNE07/ICOUNT
* COMMON SAVE1,SAVE2,PCINT
* EQUIVALENCE (W_0G(1),SAVE1(1)),(PHILIN(1),SAVE2(1)),
*   (SAVE1(3001),MDBLIN(1)),(SAVE2(3001),PHIPLN(1)),
*   (SAVE1(6001),MPDBLN(1)),(SAVE2(6001),BIGSEFM(1)),
*   (ALIM(1),XSMALL),(AAAA(1),ACOFF(2)),(B99(99)(1),
*   )PCNEFF(2))
*
* TENS(X)=X*10.0
*
* DATA(DEC(1),1,2,3,7)/1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0/
*
* DATA (EY1/4M  UB,LE12/4M DEG/
*
* K100
* KOUNTS1
.....

```

205

```

.....
*
*
*   NO 61 I=1.4
*
.....

```

```

.....
*
*   NO 49 I=1.3000
*
.....

```

```

.....
*
*   NO 06 I=0.0
*   PHILIN(1)=0.0
*   BIGSEFM(1)=0.0
*   EMSUBS(1)=0.0
*   MDBLA(1)=0.0
*   MDPLN(1)=0.0
.....

```



```

.....
I | I | I |
.....
AN N03MLN(1)30.U
.....

```

```

.....
I | I |
.....
WRITE(6,51)
READ(5,10)X$ALL,NEGEE,U9LIM
READ(5,12)PRVEXP
IF(PRVEXP.NE.0)READ(5,12)(IEXP(K),K=1,4)
12 FORMAT(4110)
19 FORMAT(3F9.0,.50A)
.....

```

```

.....
I | I |
.....
DO 90 I=2,7
.....

```

```

.....
I | I |
.....
70 A I M(I)=TEASALM(I-1)
.....

```

```

.....
I | I |
.....
DO 70 J=1,4
.....

```

```

.....
I | I |
.....
DO 71 I=1,9
.....

```

```

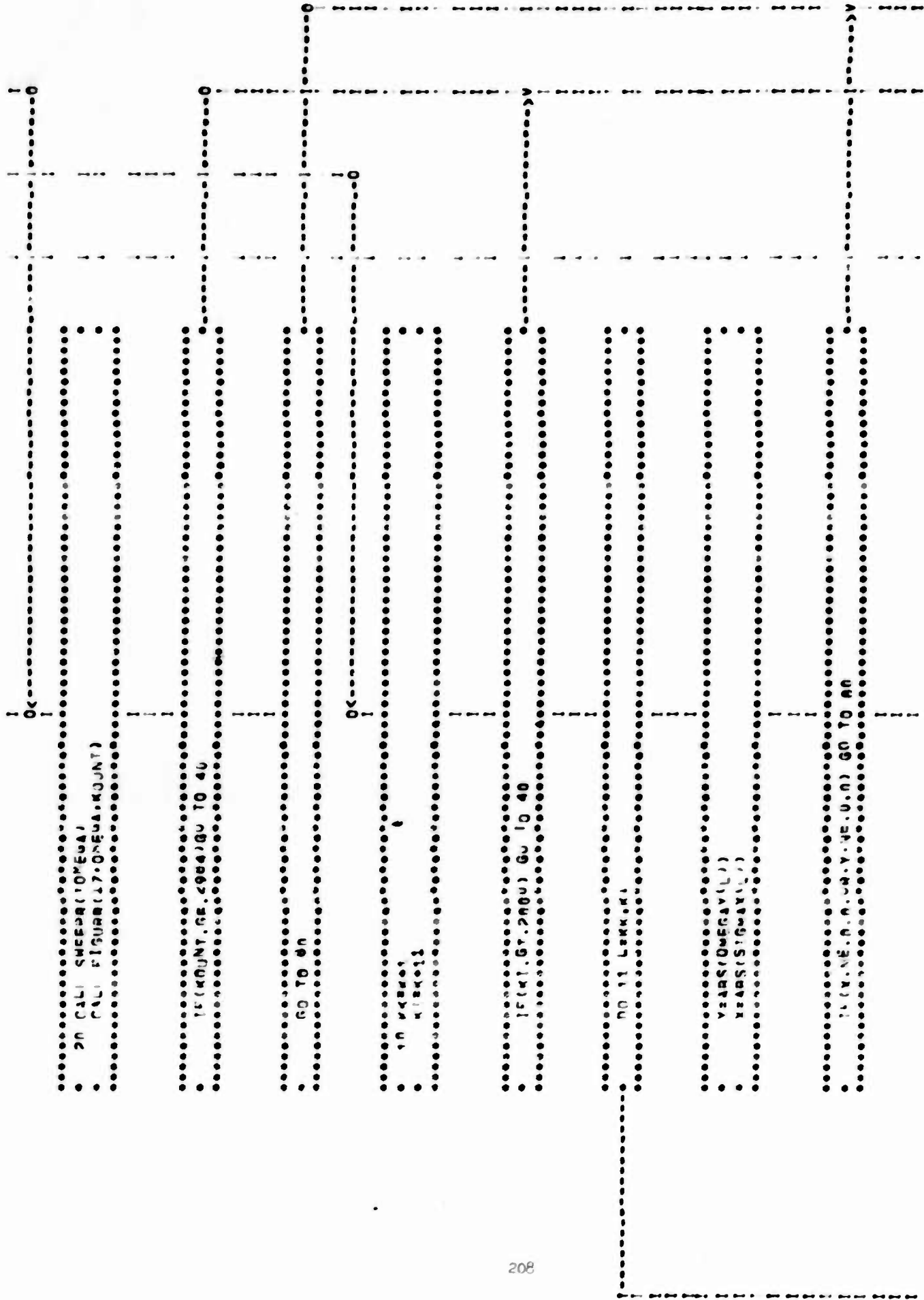
.....
I | I |
.....
WRITE(6)
.....

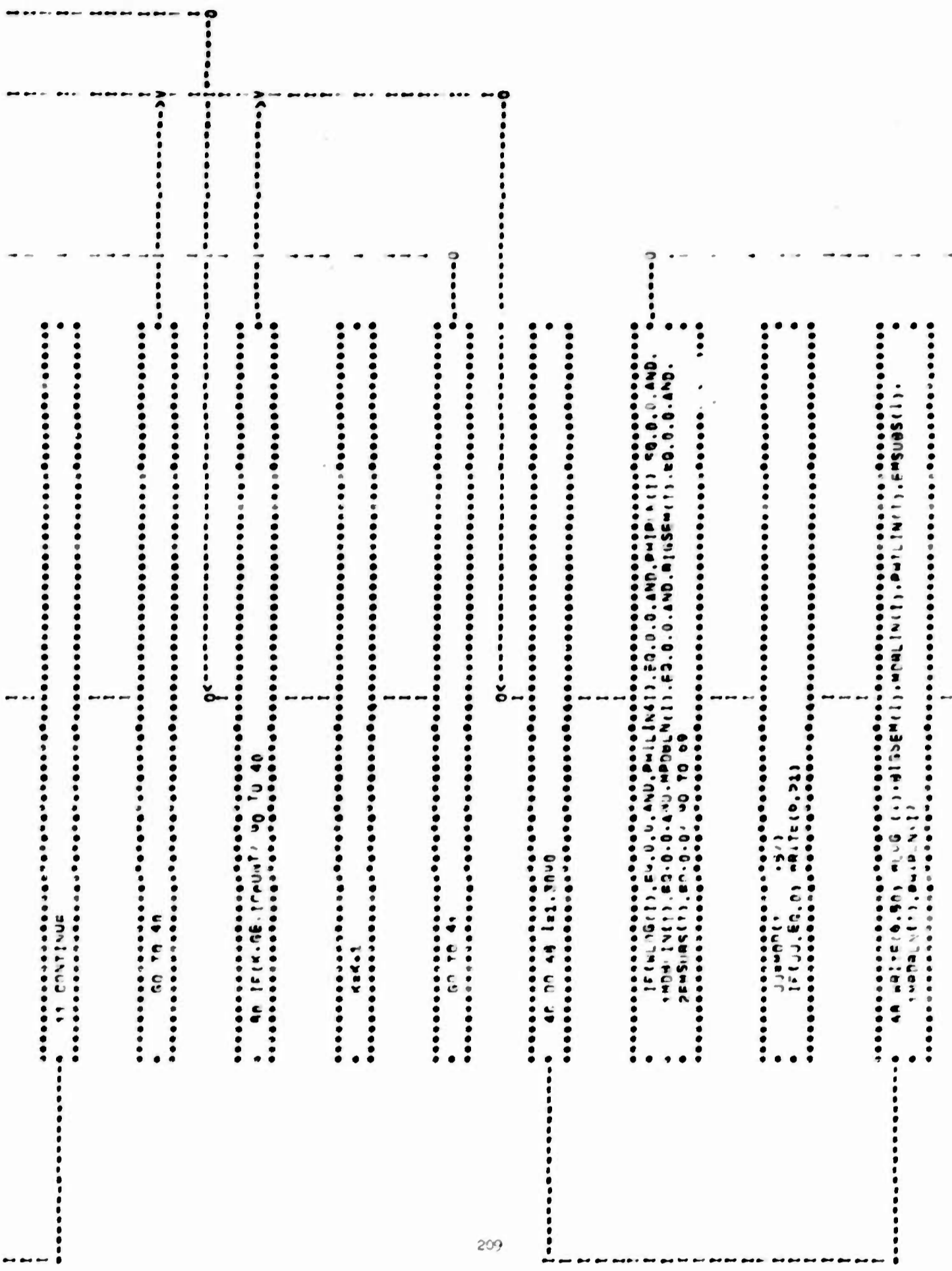
```

```

.....
I | I |
.....
79 N06GAS(N)=PEL(1)*ALM(J)
.....

```



0K-----0

.....
* 40 WRITE(6,14)
* CALL NEWP(YOURRE,LOG,MILIN,LEP2)
.....

.....
* IF(LEP12,54,0) GO TO 30
.....

.....
* JUSTEP12
.....

.....
* DO 14 INT, JJ
.....

.....
* 14 CALL EXPD(Y,LOG,3000,MILIN,3000,DEGSE,LEP2)
.....

0K-----0

.....
* 10 CALL NEWP(YOURM,LOG,MILIN,LEP1)
.....

.....
* IF(LEP12,54,0) GO TO 31
.....

.....
* JUSTEP12
.....

.....
* DO 14 INT, JJ
.....

0K-----0

.....
.. VA CALL EMPLOYING.3000, WDLIN.3000, DMLIN. LFY1)
.....

.....
.. U UNIVER(0.19)
.. CALL NEWBYTUR-14, WLOG, WDLIN, LFY2)
.....

.....
.. IF(TPXP(3), E, 0) GU 10 N2
.....

.....
.. JUBIEXP(3)
.....

.....
.. DO VA 141.00
.....

.....
.. VA CALL EMPLOYING.3000, WDLIN.3000, DEGRFF. LFY2)
.....

.....
.. V2 CALL NEWBYTUR-14, WLOG, WDLIN, LFY1)
.....

.....
.. IF(TPXP(4), E, 0) GU 10 J3
.....

.....
.. JUBIEXP(4)
.....

(ENTRANCE)

```

.....
.....
.....
.....
SUBROUTINE FLOW (KONT, OMEGA, KOUNT)
.....
DIMENSION SAVE1(10, 100), SAVE2(100, 100)
.....
DIMENSION ARBEFF(100), BCOEFF(100), AAAAAA(99), BBBBBB(100)
.....
DIMENSION KWQ1(300), PW1LN(300), MDL1N(300), MPDL1N(300), PW1PLN(300),
.....
MDPL1N(300), R12EM(300), bmsubs(300), OMEGA(KONT)
.....
REAL MODL1A, MODL1B
.....
INTEGER BAI(13), KOU
.....
COMMON/FEMR4/AUSEF, MCOEFF
.....
EQUVALENCE(AAAAAA(1), ARBEFF(2)), (BBBBBB(1), MCOEFF(2))
.....
COMMON/INFA/ERSUB
.....
COMMON SAVE1, SAVE2, KOUNT
.....
EQUVALENCE (KOU(1), SAVE1(1)), (PW1LN(1), SAVE2(1))
.....
1 (SAVE1(300), MDL1N(1)), (SAVE2(300), PW1PLN(1))
.....
2 (SAVE1(300), MPDL1N(1)), (SAVE2(300), MDPL1N(1))
.....
.....
.....
.....
DO 24 I=KOU
.....
.....
.....
CAL FLUQE(OMEGA(1), ACOEFF(1), AAAAAA, BMOFAS, BM1AQ)
.....
CAL FLQBE(PMEGA(1), BCOEFF(1)), (BBBBBB, BMOFAS, BM1AQ)
.....
R12EM(KQUAT)ZEMM
.....
KUNG(KQUAT)ZUREG(1)
.....
EDENDEEMIAS-EMIS
.....
IF (EM1END.LT.10V) / P1END=EM1END-300
.....
IF (EM1END.GT.10V) / P1END=EM1END+300
.....
PW1LN(KQUAT)ZEM1END
.....
EM1END = EM1END + 0.017433
.....
REF = ARBEFF(1) * PW1END
.....
EM1AS = ARBEFF(1) * PW1END
.....
EM1QR = SQRT ( MELEO 2 ) * ( OMEGA ( 1 ) * EM1AQ ) * ( 2 )
.....
EM1QR(KQUAT)ZEM1QR
.....
QUANZOMEGA(1) = EM1AQ
.....
PW1BEEM1A2IQUANZEM1EL
.....
PW1BE = PW1BE + 27.209705131
.....
IF (PW1BE.LT.10V) / PW1BE = PW1BE + 300
.....
IF (PW1BE.GT.10V) / PW1BE=PW1BE-300
.....
PW1LN(KQUAT)ZEM1BE
.....
MPLIN(KQUAT)ZEM1L
.....
MPLEREM(KQUAT)ZEM1L
.....
MDPLIN(KQUAT)ZEM1L
.....
MDPLEREM(KQUAT)ZEM1L
.....
KOUNTZKOU
.....
.....
.....
.....

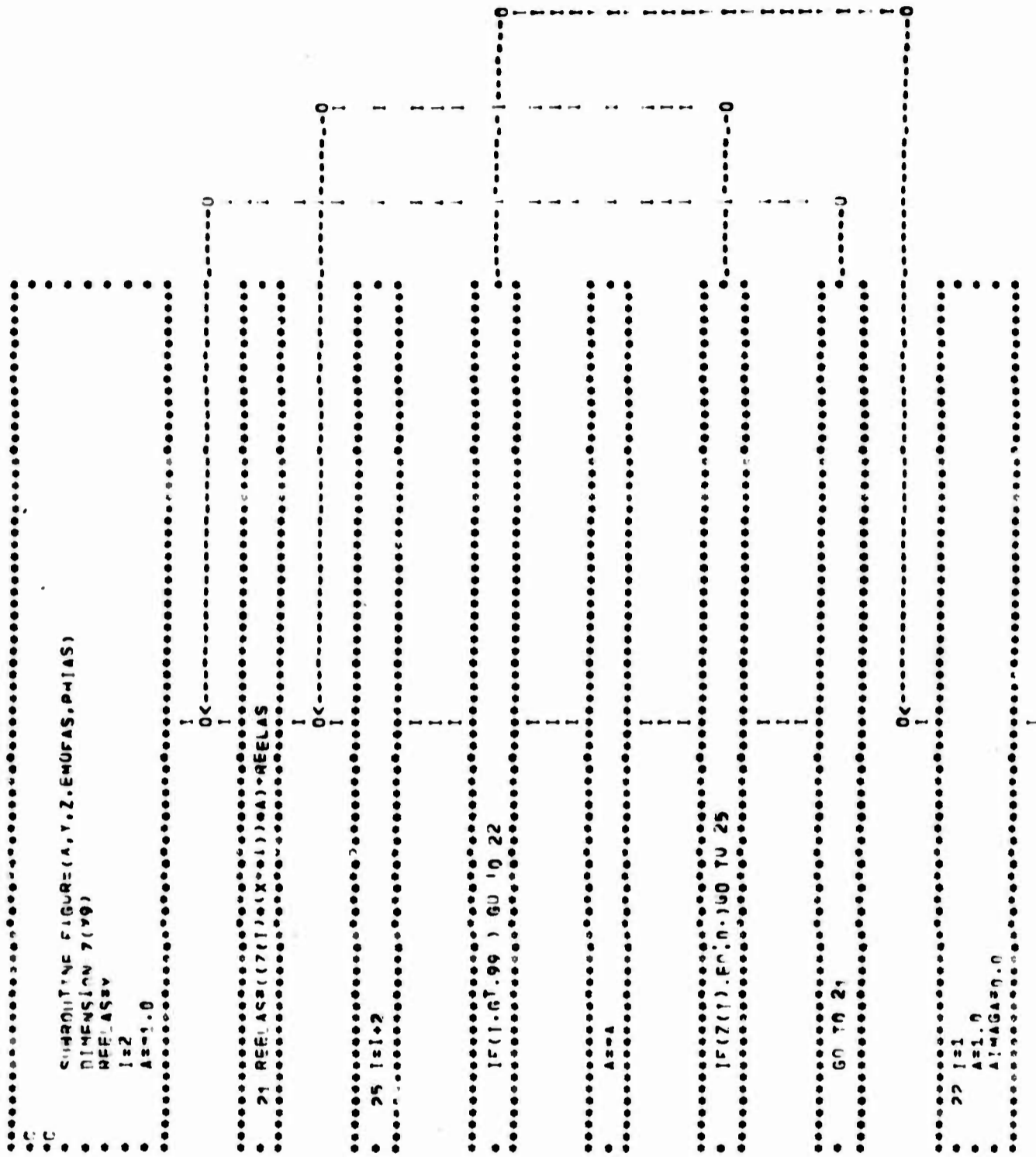
```


.....
..... 21 CONTINUE
.....

.....
..... RETURN
.....

.....
..... END
.....

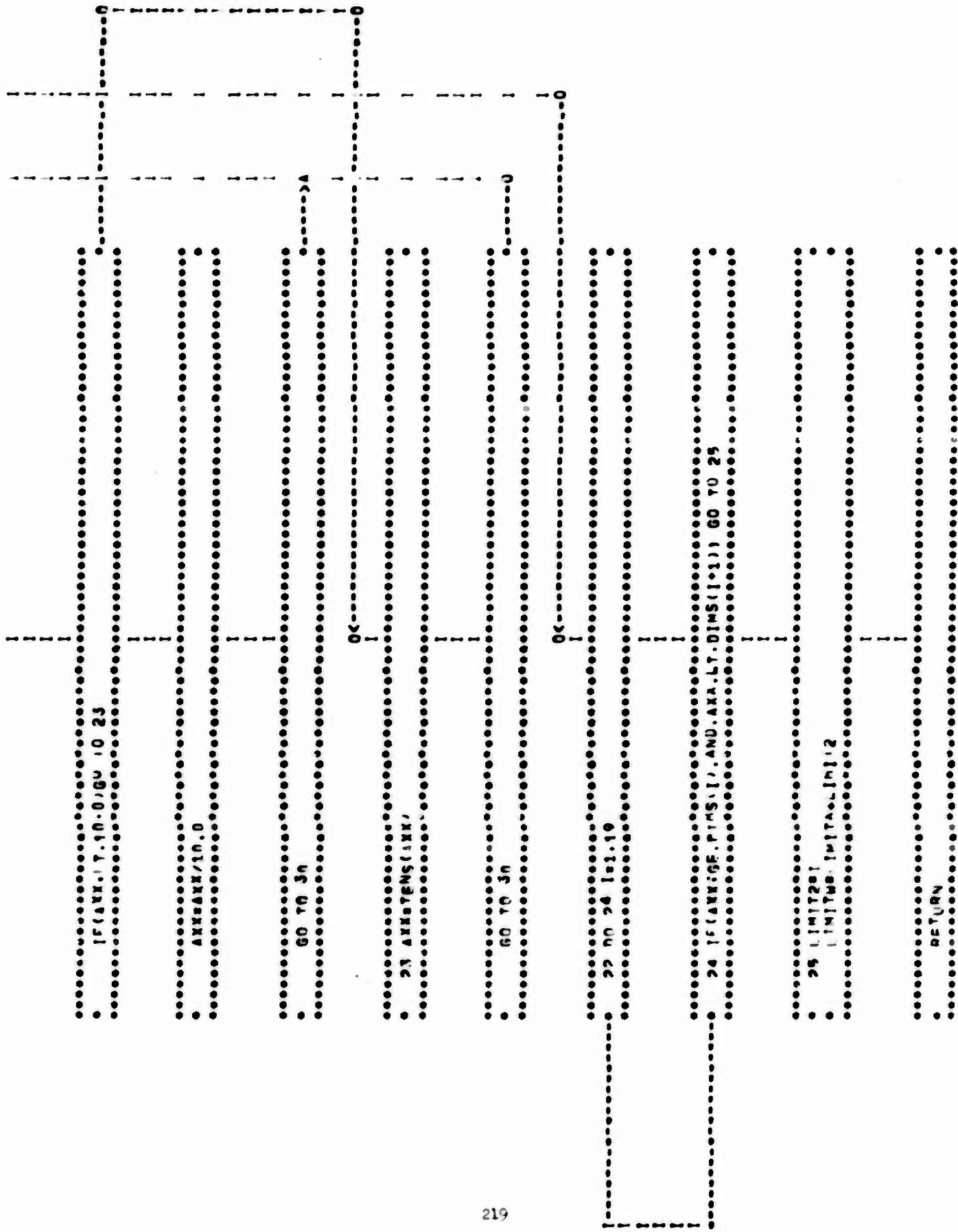
(ENTRANCE)



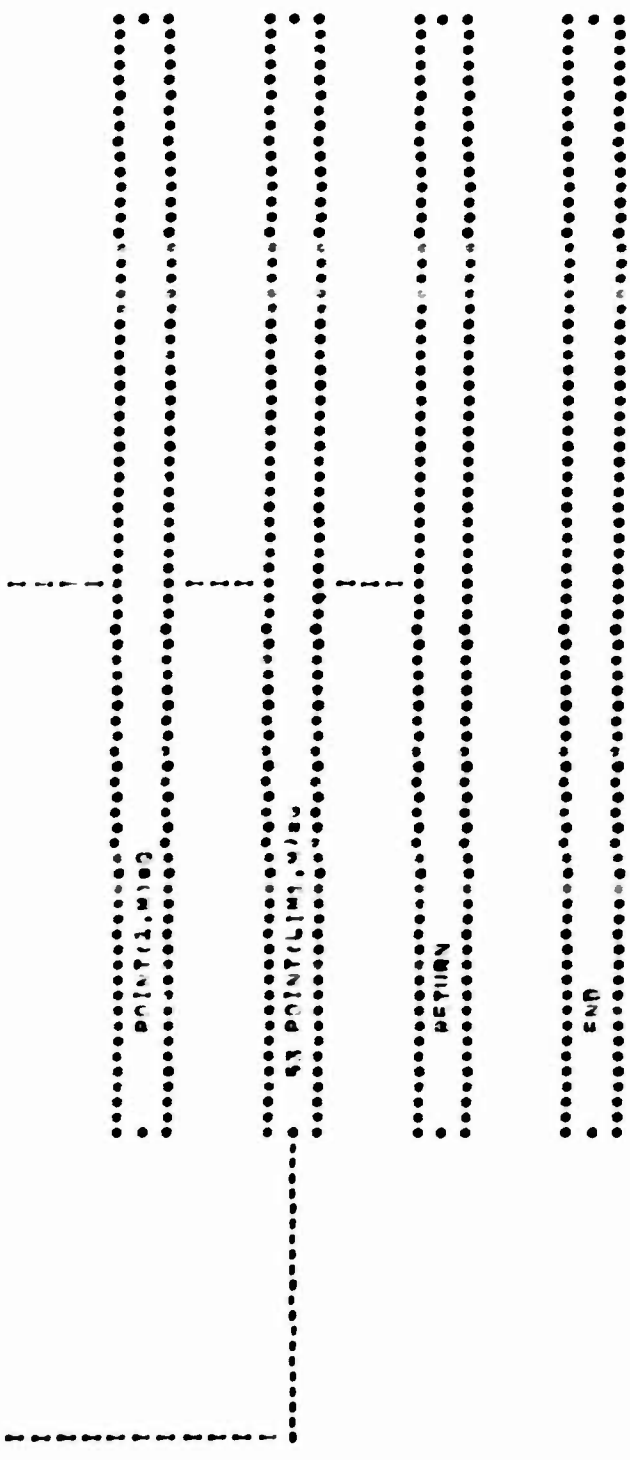
```

I
I
OK-----U
I
.....
24 AIMAGAB(Z:Y)(X=I)AIMAGA
.....
I
OK-----0
I
.....
25 I=I+2
.....
I
I
I
IF(1.GT.99) GU 10 23
.....
I
I
I
AREA
.....
I
I
I
IF(71).EQ.0)GO TU 26
.....
I
I
I
GO TO 24
.....
OK-----+-----0
I
.....
23 EMPASOSPT (REELAS*2)+(AIMAGA*2)
PHASEATAN? (AIMAGA*REELAS)
PHIAS & PHIAS*27-2797795131
IF(DMIAS.LT.U.. PHIASPHIAS-360.
.....
I
I
I
RETURN
.....

```

.....
• END
.....




```

.....
CALL LINEAR(LIMITV,Z(I))
.....
IF(LIMITV.EQ.0) GO TO 10
.....
POINTF(LIMITV,LIMITV)STAR
.....
OK
.....
10 CONTINUE
.....
DO 40 I=20,10,20
.....
DO 40 J=1,50
.....
POINTF(I,J)
.....
END

```

```

.....
WRITE(6,1)
DO FORMAT(1X,A1P1E+0.1,1X,3MM/5.6X).1P1E10.1)
J=1
WRITE(6,91)J,LET,(POINTF(I,J),I=1,120)
DO FORMAT(1X,10,44,1X,120A1)
WRITE(6,201)(POINTF(I,J),I=1,120),J=2,97)
J=J+1
WRITE(6,91)J,LET,(POINTF(I,50),I=1,120)
DO FORMAT(10X,100A1)
WRITE(6,301)ALPH
.....

```

```

.....
WRITE(6,1)
DO FORMAT(1X,A1P1E+0.1,1X,3MM/5.6X).1P1E10.1)
J=1
WRITE(6,91)J,LET,(POINTF(I,J),I=1,120)
DO FORMAT(1X,10,44,1X,120A1)
WRITE(6,201)(POINTF(I,J),I=1,120),J=2,97)
J=J+1
WRITE(6,91)J,LET,(POINTF(I,50),I=1,120)
DO FORMAT(10X,100A1)
WRITE(6,301)ALPH
.....

```

.....
RETURN
.....

.....
END
.....

1 1 0
1 1 0
0 1 1

.....
4n RETURN
.....

.....
END
.....

(ENTRANCE)

```

.....
*
*
* SUBROUTINE SWEEP(K,OMEGA)
* DIMENSION OMEGA(17),SWEPER(17)
* COMMON/INFR3/OMEGA
* DATA (SWEPER), (0.01,0.05,0.1,0.2,0.3,0.4,0.5,0.6,0.7,
* 0.8,0.9,1.0,2.0,4.0,6.0,10.0)
*
.....

```

```

.....
* DO 70 K=1,17
*
.....

```

```

.....
* 70 OMEGA(K)= SWEPER(K)/OMEGA
*
.....

```

```

.....
* RETURN
*
.....

```

```

.....
* END
*
.....

```

AD Burke, Harold H., Payne, Robert L., Jr.
 USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency
 Aberdeen Proving Ground, Md. 21005
 Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED
 Report
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC
 EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H.
 Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M1102,
 March 1970

Trace Index Terms
 Systems analyses, computer capability Phase-amplitude characteristic,
 Linear frequency response method nonlinear systems
 Nonlinear frequency response method Closed loop transfer function
 Open loop transfer function Bode plots, system transfer
 Line printer plots, root loci
 1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC
 EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS Harold H.
 Burke, Robert L. Payne, Jr.

AD Burke, Harold H., Payne, Robert L., Jr.
 USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency
 Aberdeen Proving Ground, Md. 21005
 Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED
 Report
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC
 EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H.
 Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M1102,
 March 1970

Trace Index Terms
 Systems analyses, computer capability Phase-amplitude characteristic,
 Linear frequency response method nonlinear systems
 Nonlinear frequency response method Closed loop transfer function
 Open loop transfer function Bode plots, system transfer
 Line printer plots, root loci
 1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC
 EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H.
 Burke, Robert L. Payne, Jr.

AD Burke, Harold H., Payne, Robert L., Jr.
 USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency
 Aberdeen Proving Ground, Md. 21005
 Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED
 Report
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC
 EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H.
 Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M1102,
 March 1970

Trace Index Terms
 Systems analyses, computer capability Phase-amplitude characteristic,
 Linear frequency response method nonlinear systems
 Nonlinear frequency response method Closed loop transfer function
 Open loop transfer function Bode plots, system transfer
 Line printer plots, root loci
 1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC
 EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS Harold H.
 Burke, Robert L. Payne, Jr.

AD Burke, Harold H., Payne, Robert L., Jr.
 USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency
 Aberdeen Proving Ground, Md. 21005
 Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED
 Report
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC
 EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H.
 Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M1102,
 March 1970

Trace Index Terms
 Systems analyses, computer capability Phase-amplitude characteristic,
 Linear frequency response method nonlinear systems
 Nonlinear frequency response method Closed loop transfer function
 Open loop transfer function Bode plots, system transfer
 Line printer plots, root loci
 1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC
 EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H.
 Burke, Robert L. Payne, Jr.

AD
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS
 USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005
 Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AM6AA. DDC; Project No. RDT&E 1P765801M1102, March 1970

Trace Index Terms
 Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems
 Linear frequency response method Closed loop transfer function
 Nonlinear frequency response method Bode plots, system transfer
 Open loop transfer function Popov plots, system transfer
 Line printer plots, root loci
 1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.

Burke, Harold H., Payne, Robert L., Jr.
 USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005
 Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AM6AA. DDC; Project No. RDT&E 1P765801M1102, March 1970

Trace Index Terms
 Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems
 Linear frequency response method Closed loop transfer function
 Nonlinear frequency response method Bode plots, system transfer
 Open loop transfer function Popov plots, system transfer
 Line printer plots, root loci
 1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.

AD
 Systems analyses, computer capability
 USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005
 Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AM6AA. DDC; Project No. RDT&E 1P765801M1102, March 1970

Trace Index Terms
 Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems
 Linear frequency response method Closed loop transfer function
 Nonlinear frequency response method Bode plots, system transfer
 Open loop transfer function Popov plots, system transfer
 Line printer plots, root loci
 1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.

AD
 Linear frequency response method
 USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005
 Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AM6AA. DDC; Project No. RDT&E 1P765801M1102, March 1970

Trace Index Terms
 Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems
 Linear frequency response method Closed loop transfer function
 Nonlinear frequency response method Bode plots, system transfer
 Open loop transfer function Popov plots, system transfer
 Line printer plots, root loci
 1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
 A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.

<p>AD _____</p> <p>Nonlinear frequency response method</p> <p>USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005</p> <p>Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report</p> <p>A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M102, March 1970</p> <p><u>Trace Index Terms</u></p> <p>Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems Linear frequency response method Closed loop transfer function Nonlinear frequency response method Bode plots, system transfer Open loop transfer function Popov plots, system transfer Line printer plots, root loci</p> <p>1. Systems analyses, computer capability 2. Linear frequency response method 3. Nonlinear frequency response method 4. Line printer plots, root loci</p> <p>A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.</p>	<p>Line printer plots, root loci</p> <p>USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005</p> <p>Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report</p> <p>A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M102, March 1970</p> <p><u>Trace Index Terms</u></p> <p>Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems Linear frequency response method Closed loop transfer function Nonlinear frequency response method Bode plots, system transfer Open loop transfer function Popov plots, system transfer Line printer plots, root loci</p> <p>1. Systems analyses, computer capability 2. Linear frequency response method 3. Nonlinear frequency response method 4. Line printer plots, root loci</p> <p>A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.</p>
<p>AD _____</p> <p>Nonlinear frequency response method</p> <p>USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005</p> <p>Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report</p> <p>A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M102, March 1970</p> <p><u>Trace Index Terms</u></p> <p>Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems Linear frequency response method Closed loop transfer function Nonlinear frequency response method Bode plots, system transfer Open loop transfer function Popov plots, system transfer Line printer plots, root loci</p> <p>1. Systems analyses, computer capability 2. Linear frequency response method 3. Nonlinear frequency response method 4. Line printer plots, root loci</p> <p>A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.</p>	<p>Line printer plots, root loci</p> <p>USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005</p> <p>Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report</p> <p>A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M102, March 1970</p> <p><u>Trace Index Terms</u></p> <p>Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems Linear frequency response method Closed loop transfer function Nonlinear frequency response method Bode plots, system transfer Open loop transfer function Popov plots, system transfer Line printer plots, root loci</p> <p>1. Systems analyses, computer capability 2. Linear frequency response method 3. Nonlinear frequency response method 4. Line printer plots, root loci</p> <p>A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.</p>

AD

Nonlinear frequency response method

USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005

Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report

A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M1102, March 1970

Trace Index Terms

Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems
Linear frequency response method
Nonlinear frequency response method Closed loop transfer function
Open loop transfer function Bode plots, system transfer
Line printer plots, root loci Popov plots, system transfer

1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
- A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.

AD

Line printer plots, root loci

USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005

Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report

A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M1102, March 1970

Trace Index Terms

Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems
Linear frequency response method
Nonlinear frequency response method Closed loop transfer function
Open loop transfer function Bode plots, system transfer
Line printer plots, root loci Popov plots, system transfer

1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
- A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.

AD

USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005

Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report

A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M1102, March 1970

Trace Index Terms

Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems
Linear frequency response method
Nonlinear frequency response method Closed loop transfer function
Open loop transfer function Bode plots, system transfer
Line printer plots, root loci Popov plots, system transfer

1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
- A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.

AD

USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Md. 21005

Army Materiel Systems Analysis Agency Technical Memorandum No. 69, UNCLASSIFIED Report

A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr., AMSAA. DDC; Project No. RDT&E 1P765801M1102, March 1970

Trace Index Terms

Systems analyses, computer capability Phase-amplitude characteristic, nonlinear systems
Linear frequency response method
Nonlinear frequency response method Closed loop transfer function
Open loop transfer function Bode plots, system transfer
Line printer plots, root loci Popov plots, system transfer

1. Systems analyses, computer capability 2. Linear frequency response method
 3. Nonlinear frequency response method 4. Line printer plots, root loci
- A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS, Harold H. Burke, Robert L. Payne, Jr.

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R & D

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

1. ORIGINATING ACTIVITY (Corporate author) USA Aberdeen Research and Development Center, Army Materiel Systems Analysis Agency, Aberdeen Proving Ground, Maryland 21005		2a. REPORT SECURITY CLASSIFICATION Unclassified	
		2b. GROUP	
3. REPORT TITLE A LINEAR AND NONLINEAR SYSTEMS ANALYSIS TOOL: LINE PRINTER PLOTS OF CHARACTERISTIC EQUATION ROOT LOCI, BODE AND POPOV PLOTS OF SYSTEM TRANSFER FUNCTIONS			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates)			
5. AUTHOR(S) (First name, middle initial, last name) Harold H. Burke Robert L. Payne, Jr.			
6. REPORT DATE March 1970		7a. TOTAL NO. OF PAGES 230	7b. NO. OF REFS 9
8a. CONTRACT OR GRANT NO.		8b. ORIGINATOR'S REPORT NUMBER(S)	
a. PROJECT NO. RDT&E 1P765801ML102		AMSAA TM-69	
c.		8c. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
d.			
10. DISTRIBUTION STATEMENT This document may be further distributed by the holder only with specific prior approval of the Commanding General, U.S. Army Materiel Command, ATTN: AMCPA-SA, Washington, D.C. 20315.			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY U.S. Army Materiel Command Washington, D.C.	
13. ABSTRACT A computer program to perform dynamic systems analyses and plot the results is presented. Linear systems and a large classification of nonlinear systems representing engineering, scientific, and economic disciplines can be modeled to permit application of the computer program. Two examples are given to demonstrate the capabilities of the analysis tool. The mathematical model of a missile guidance and control system is analyzed and a ratio of polynomials representing the closed loop transfer function of a high performance model follower aircraft is evaluated. Linear differential equations to the 100th order having real or complex roots can be studied. System characteristic equation root loci and system transfer functions are plotted.			

DD FORM 1473
1 NOV 66

REPLACES DD FORM 1473, 1 JAN 64, WHICH IS OBSOLETE FOR ARMY USE.

Unclassified
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Systems analysis						
Systems synthesis						
Root locus						
Bode plots						
Steady state response						
Popov plots						
Modified phase-amplitude characteristic						