AD-A033 711 ROME AIR DEVELOPMENT CENTER GRIFFISS AFB N Y F/G 15/3
SPACE SURVEILLANCE SOFTWARE SUPPORT. VOLUME II. GENERALIZED DAT--ETC(U)
OCT 76 J A MANLEY

UNCLASSIFIED RADC-TR-76-261-VOL-2 NL

1 OF 2
AD
A033711

(12)

B.S.

RADC-TR-76-261, Volume II
In-house Report
October 1976

SPACE SURVEILLANCE SOFTWARE SUPPORT

Volume II

Generalized Data Entry and Plot Program

John A. Manley

Approved for Public Release.
Distribution unlimited.

ROME AIR DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
GRIFFISS AIR FORCE BASE, NEW YORK 13441

This report has been reviewed by the Office of Information, RADC, and approved for release to the National Technical Information Service, NTIS. At NTIS, it will be available to the general public, including foreign nations.

This report has been reviewed and is approved for publication.

APPROVED:

John C. Cleary

JOHN C. CLEARY
Project Engineer

APPROVED: Rudolf C. Paltauf

RUDOLF C. PALTAUF
Lt Col, USAF
Chief, Surveillance Division

FOR THE COMMANDER: John P. Huss

JOHN P. HUSS
Acting Chief
Plans Office

Do not return this copy. Retain or destroy.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
RADC-TR-76-261, Volume II

2. GOVT ACCESSION NO.

3. ~~RECIPIENT'S~~ CATALOG NUMBER
(9)

4. ~~TITLE (and Subtitle)~~
(6) SPACE SURVEILLANCE SOFTWARE SUPPORT. Volume II.
Generalized Data Entry and Plot Program.

5. TYPE OF ~~REPORT & PERIOD~~ COVERED
~~In-house~~ Final rept.
Jan 75 — June 76.

6. ~~PERFORMING ORG.~~ REPORT NUMBER
N/A

7. ~~AUTHOR(s)~~
(10) John A. Manley

8. CONTRACT OR GRANT NUMBER(s)
N/A

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Rome Air Development Center (OCSA)
Griffiss AFB, NY 13441

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
62702F
65121205

11. CONTROLLING OFFICE NAME AND ADDRESS
Same

12. ~~REPORT DATE~~
(11) Oct~~ober~~ 1976

13. NUMBER OF PAGES
107

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
Same
(12) 117p.

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
N/A

16. DISTRIBUTION STATEMENT (of this Report)
Approved for Public Release.
Distribution Unlimited.
(16) 6512 (17) 12

17. DISTRIBUTI~~ON STATEMENT (of the abstract entered in Block 20, if different from Report)~~
Same (18) RADC (19) TR-76-261-Vol-2

18. SUPPLEMENTARY NOTES
This is Volume II of three volumes; Volume I is in two parts.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Antenna Patterns
Trajectory Curves
Waveform Response
HP Calculator Plots

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

The primary purpose of this report is to document the Hewlett-Packard (HP) 9820A calculator programs used to plot data generated by the HIS-6180 digital computing system. This was done because on-line plotting at the central computer is impossible and large volumes of data cannot be digested efficiently by the engineers concerned.

Quickly plotted data often allows an engineer to narrow down the number of computer runs by bounding a specific problem since a picture is worth a thousand

done

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

309050

1B

20.

words. For example, trajectory curves (an antenna pattern or waveform response to a target) are more than long tables of numerical listings. The objective of the plotting capability outlined in this report is to free the engineer for more analysis work and cut the required number of computer runs by bounding the problem in a timely manner. This will result in more effective use of the engineer at a cost savings to the Air Force.

In addition, this report will serve as a supplement to the HP manuals detailing the use of the HP paper tape and punched card readers.

PREFACE

This report provides the computer programs used to plot virtually any data of an (X,Y) nature. These programs are written in algebraic language for a Hewlett-Packard 9820A programmable calculator. The data must be in USA Standard Code for Information Interchange (ASCII) format in order to be handled by the HP 9820A. The data generated by the RADC HIS-6180 is in either binary or a Binary Coded Decimal (BCD) format and must be converted to ASCII format before being plotted. The procedure for obtaining the data from the HIS-6180 in the correct format and plotting will be explained and illustrated.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (CONT'D)

COMPUTER PROGRAMS

I. Introduction

These plot programs were developed in support of the Space Surveillance Software Support contract. They have also been used in the SEEK SAIL study, the Endo Exo-Atmospheric Modeling contract*, the Space Defense System (SDS) Radar Study, and the Interactive Radar Simulator contract.

The data generated under each of these studies and contracts listed above, although extremely varied, can be taken from permanent disc file storage on the HIS 6180 and a plot generated using virtually the same procedure for each case. Slight variations will occur in scaling the data and in labeling the plots in the Hewlett-Packard (HP) 9820A calculator. The foregoing procedure will be described and illustrated such that the user may generate the plots he requires for his particular application.

This report will also serve as a supplemental manual to the HP manuals detailing the use of the HP paper tape and punched card readers for data input to the HP 9820A calculator.

---

* See Appendix D

II. HP 9820A Calculator and Peripherals

The plot program was implemented on the HP 9820A calculator located at RADC/OCSA in Building 106.

Peripherals include an HP 9862A plotter, HP 9869A card reader, HP 2748B paper tape reader, and 11223A cassette memory units.

Due to the potentially large amounts of data and the relatively limited memory of the calculator, the cassette tapes are used as a temporary store for the data before plotting. The tape reader and card reader are the input devices by which data is introduced into the "HP system" described above.

III. Procedure Used to Transfer Data to Punched Cards

The program used to generate data must also store the data on a PRMFL (permanent file). The format used to store the data on PRMFL must be the same as the data points are to appear on the punched card. One card has a field of 80 possible digits for punching data, and the card reader will recognize a space or comma as a delimiter between data points*. One example of the way to store data on a PRMFL is shown below:

```
              WRITE (09,1000) (DATA (I), I = 1,5)
1000 FORMAT (2X, F10.6, 2X, F10.6,2X, F10.6, 2X, F10.6, 2X, F10.6)
```

The 09 in the WRITE statement is the number of the PRMFL where data is stored and the 1000 is the number of the FORMAT statement which determines the format of stored data. The data is stored in blocks of 5 and corresponds to the card columns shown below.

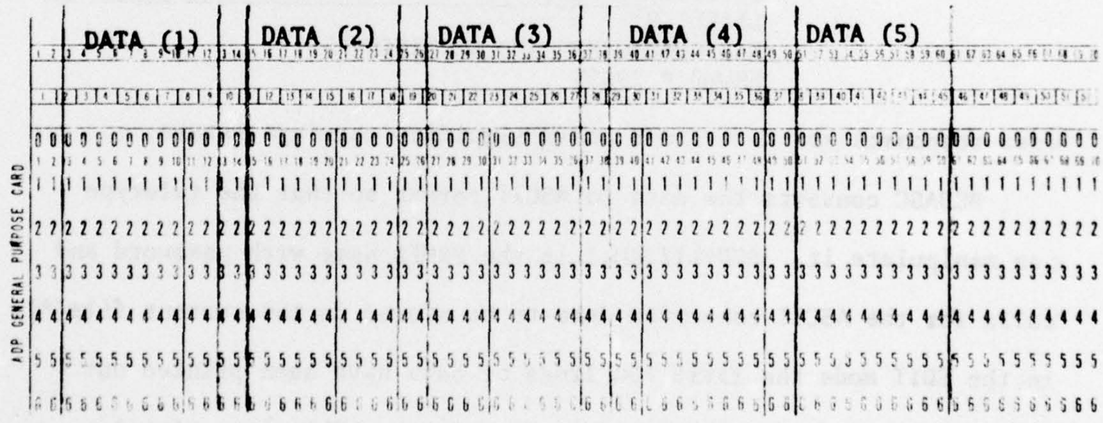


FIGURE 1. Sample Data Fields on Computer Card

---

* The card reader will treat any character as a data delimiter with the following exceptions:1) a digit from 0 to 9, 2) a plus or minus sign, 3) an exponential (5E - 05)

To transfer stored data from PRMFL to punched cards, the following time-sharing system (TSS) procedure can be used:

```
    SYSTEM  ?CARDIN
    OLD OR NEW - N
    READY
    *BCDASC
    INPUT FILES  BCDFL1$JAM;*
    LINE NUMBERS Ⓡ
    TAB CHARACTERS AND SETTINGS  Ⓡ
    *EDIT
  - P;10
    001229457545E 01
     0.459417272E 01
     0.706554240E 01
     0.965192258E 01
     0.122055005E 02
     0.144559367E 02
     0.160495934E 02
     0.167047484E 02
     0.163838296E 02
     0.152977045E 02
  - DONE
    *BPUNCH *
    $ IDENT USERID, USERS NAME, ACCOUNT NO., REMARKS
    LABELS Ⓡ
    TAB CHARACTERS AND SETTINGS Ⓡ
    SNUMB # 2985T
```

User responses are underlined and Ⓡ signifies carriage return.

BCDASC converts the data to ASCII format so that the teletype can manipulate it. BCDFL1$JAM;* is the PRMFL name with password and calls for the ASCII converted data to be stored on the current file(*). In the EDIT mode the first ten lines of data have been printed out as a check to ensure that the data is correct. This is optional. BPUNCH* causes data stored on the current file (*) to be punched on cards.

IV. Procedures Used to Transfer Data to Punched Paper Tape

Extracting data from the HIS-6180 to paper tape is similar to the process for transferring data to punched cards. The program generating the data must also store it on PRMFL using the same procedure as shown for cards in the previous section. The WRITE and FORMAT statements for storing data in BCD code on PPMFL are identical.

There is no way of obtaining punched paper tape directly (on line) from the HIS-6180. It is necessary to utilize a teletypewriter interfaced with a paper tape punch. The one currently used is the Model ASR 33 Teletype.

Once the data is stored on PRMFL in the desired format, the data can be transferred to paper tape according to the following:

```
SYSTEM?  CARDIN
OLD OR NEW - NEW
READY
* BCDASC
INPUT FILES?  FILE NAME & PASSWD; *
LINE NUMBERS              ®
TAP CHARACTERS AND SETTINGS      ®
* LIST
```

The responses typed in by the user are underlined, and ® signifies a carriage return (RETURN key). Once the word LIST has been typed in, hit the RETURN key and simultaneously turn on the tape punch. The teletypewriter will print out a hard copy and will simultaneously punch out the data onto a paper tape. This data is in ASCII code.

V. Generalized Plot Program - Data Input via Punched Cards

The plot program will be explained using punched card data for the following reasons:

1. Punched cards have been used more extensively
2. The time required to generate the data in a form compatible with the "HP system" can be excessive for paper tape vs cards.
3. When large amounts of data are involved, the amount of paper tape becomes unwieldy; i.e., the paper tape becomes extremely long. The feed mechanism on the HP 2748B paper tape reader cannot automatically draw the paper tape from the feed tray through the reader, and the operator must manually feed the tape through the reader. Large volumes of data are of the order of 500-700 data points and approximately 6 to 10 minutes are required to punch this data for each plot.

A listing of the computer program as implemented on the HP 9820A calculator follows.

The inputs required by the program are shown in Table 1. The output is primarily the plot and several examples will be shown.

Also, each register used in the program is listed in Table 2 with an explanation of the value stored in each.

```
 0:  PRT "MANLEY";FXD 4 [
 1:  ENT "PNUMB",R48,"SELECT CODE",Y,"DFILE",A [
 2:  IF R48=1;GTO +4 [
 3:  FMT "D";WRT 2;FMT *;RED 2,R7,R6,R0,R13,R41;PRT R7,
     R6,R0,R13,R41;SPC 2 [
 4:  (R41-1)R13+R0 TO R1 [
 5:  SSC Y;RCF A,R0,R41;PRT R0,R1,R6,R7,R13,R41;SPC 2;G
     TO +2 [
 6:  SSC Y;LDF A,R0;PRT R0,R1,R6,R7,R41;SPC 2 [
 7:  ENT "XLENGTH",R15,"YLENGTH",R16 [
 8:  ENT "ABSCISSA USED",R17,"ORDINATE USED",R18 [
 9:  SCL 0,R15,0,R16;AXE 1,1,.5,.5 [
10:  R15/R17 TO R42;R16/R18 TO R43 [
11:  2R17 TO R11;2R18 TO R12 [
12:  ENT "XMIN",R20,"XMAX",R21,"YMIN",R22,"YMAX",R23 [
13:  ABS (R21-R20)/R11 TO R3;R20+R3 TO R4;1 TO R40 [
14:  R21-R42(R21-R20) TO R5 [
15:  SCL R5,R21,0,R16 [
16:  FXD 0;LTR R4,.4,222;PLT R4 [
17:  1+R40 TO R40;IF R40 # R11+1;R4+R3 TO R4;GTO -1 [
18:  ABS (R23-R22)/R12 TO R8;R22+R8 TO R9;1 TO R40 [
19:  R23-R43(R23-R22) TO R10 [
20:  SCL 0,R15,R10,R23 [
21:  FXD 2;LTR .5,R9,211;PLT R9 [
22:  1+R40 TO R40;IF R40 # R12;R9+R8 TO R9;GTO -1 [
```

FIGURE 2. GENERALIZED PLOT PROGRAM - DATA INPUT VIA PUNCHED CARDS

```
23:  ENT "NO. FILES",R2,"PTS/FILE",R40,"SFILE",A [

24:  1 TO C;0 TO B [

25:  5INT (R41/5) TO R45;0 TO R47;IF R2=1;GTO "2" [

26:  IF R48=1;SSC Y;LDF A,R50;(R2-1)R40 TO R47;GTO +19 [

27:  FMT "C";WRT 2 [

28:  FMT *;RED 2,R(50+B),R(51+B),R(52+B),R(53+B),R(54+B

     ) [

29:  B+5 TO B;IF R40-1>B;GTO -2 [

30:  R47+R40 TO R47 [

31:  SSC Y [

32:  RCF A,R50,R(R40+49);PRT A [

33:  IF R47=(R2-1)R40;GTO "2" [

34:  A+1 TO A;1+C TO C [

35:  IF C # R2+1;0 TO B;GTO -8 [

36:  "2";IF R41/5=INT (R41/5);R41-R47 TO R46;GTO +3 [

37:  R45+5 TO R45;R45-R47 TO R46;IF R2=1;0 TO B;GTO +2 [

38:  0 TO B;A+1 TO A [

39:  IF R48=1;GTO +5 [

40:  FMT "C";WRT 2 [

41:  FMT *;RED 2,R(50+B),R(51+B),R(52+B),R(53+B),R(54+B

     ) [

42:  B+5 TO B;IF R46-1>B;GTO -2 [

43:  SSC Y;PRT A;RCF A,R50,R(R46+49) [

44:  A-R2+1 TO A;1 TO C;SPC 2;PRT A;LDF A,R50 [

45:  0 TO Z;R41-R47 TO R47;PRT A [
```

FIGURE 2. GENERALIZED PLOT PROGRAM - DATA INPUT VIA PUNCHED CARDS (CONT'D)

```
46:  SCL R5,R21,R10,R23 [
47:  PLT R0,R(50+Z) [
48:  IF C=R2;GTO +3 [
49:  IF R40-1 # Z=1;R0+R13 TO R0;Z+1 TO Z;GTO -2 [
50:  A+1 TO A;C+1 TO C;IF C <= R2;R0+R13 TO R0;0 TO Z;PRT A;LDF A
     ,R50;GTO -3 [
51:  IF R47-1 # Z=1;R0+R13 TO R0;Z+1 TO Z;GTO -4 [
52:  PRT "END OF DATA" [
53:  END  [
```

FIGURE 2. GENERALIZED PLOT PROGRAM - DATA INPUT VIA PUNCHED CARDS (CONT'D)

INPUTS

Variable Name	Register	Meaning
P NUMB	R48	Determine where data is stored. Set to 1 if on cassette tapes, or 0 if on cards or paper tape
SELECT CODE	Y	Selects proper cassette Unit(s)
DFILE	A	File where minimum and maximum X and Y values of data, X increment, and number of pts key in file number
Xmin*	RO	Minimum X (independent variable) value
Ymin*	R6	Minimum Y value
Ymax*	R7	Maximum Y value
Xincrement*	R12	Increment between data points of independent variable
No pts *	R41	Total number points to be plotted
XLENGTH	R15	The total horizontal length of the plotting surface
YLENGTH	R16	The total vertical length of the plotting surface
ABSCISSA USED	R17	The portion of the abscissa used for plotting, i.e., excluding portion used of labeling the axis
ORDINATE USED	R18	The portion of the ordinate used for plotting, i.e., excluding portion used for labeling the axis

TABLE 1
INPUTS FOR GENERALIZED PLOT PROGRAM

---

* These data may be calculated by the program generating the data and are punched out on the first card or they may be keyed in by the operator using an ENT statement. However, if these data are not calculated by the generating program and it is inefficient to find them from long numerical listings, it is necessary to key in only the minimum value and the increment between values of the independent variable and the number of points.

INPUTS

Variable Name	Register	Meaning
XMIN	R20	Minimum value of independent variable for scaling and labeling the X-axis
XMAX	R21	Maximum value of independent variable for scaling and labeling the X-axis
Y MIN	R22	Minimum value of dependent variable for scaling and labeling the Y-axis
Y'AX	R23	Maximum value of dependent variable for scaling and labeling the Y-axis
NO. FILES	R2	No. of data files
P TS/FILE	R40	No. of data points per file
SFILE	A	File number of first data file

TABLE 1 (CONT'D)

REGISTER TABLE

Register	Value Stored
R0	Minimum value of independent variable
R1	Maximum value of independent variable
R2	No. of files of stored data
R3	Increment between labels on abscissa
R4	Value of each label on abscissa
R5	Xmin for scale of plotting surface used for plotting, i.e., excluding portion used for labeling axes
R6	Minimum value of dependent variable
R7	Maximum value of dependent variable
R8	Increment between labels on ordinate
R9	Value of each label on ordinate
R10	Ymin for scale of plotting surface used for plotting, i.e., excluding portion used for labeling axes
R11	No. of labels on abscissa
R12	No. of labels on ordinate
R13	Increment between consecutive values of independent variable
R14	Not used
R15	Length of plotting surface in X dimension, i.e, no. major divisions
R16	Length of plotting surface in Y dimension, i.e., no. of major divisions

TABLE 2

REGISTER CROSS REFERENCE CHART FOR GENERALIZED PLOT PROGRAM

REGISTER TABLE

Register	Value Stored
R17	Portion of plotting surface in x dimension used for plotting, i.e., excluding portion used for labeling axes
R18	Portion of plotting surface in Y dimension used for plotting, i.e., excluding portion used for labeling axes
R 19	Not used
R20	Minimum value of labels on abscissa (XMIN)
R21	Maximum value of labels on abscissa (XMAX)
R22	Minimum value of labels on ordinate (YMIN)
R23	Maximum value of labels on ordinate (YMAX)
R24-R39	Not used
R40	Points per file
R41	Total number of points
R42	Ratio of total length of abscissa to portion used for plotting - R15/R17
R43	Ratio of total length of ordinate to portion used for plotting - R16/R18
R44	Not used
R45	No. of complete records of valid data (5 per record)
R46	No. of points in the last file
R47	Counter for number of points recorded on a file basis (i.e., number files X points/file)
R 48	Flag used to determine where data stored
R 50 ...	Data Storage
A	File Number

TABLE 2

REGISTER CROSS REFERENCE CHART FOR GENERALIZED PLOT PROGRAM (CONT'D)

REGISTER TABLE

Register	Value Stored
B	Register Counter
C	File Counter
X	Not used
Y	Select Code
Z	Register counter

TABLE 2

REGISTER CROSS REFERENCE CHART FOR GENERALIZED PLOT PROGRAM (CONT'D)

A detailed flow chart showing each statement with an explanation of each follows, but a general description of program operation will be helpful.

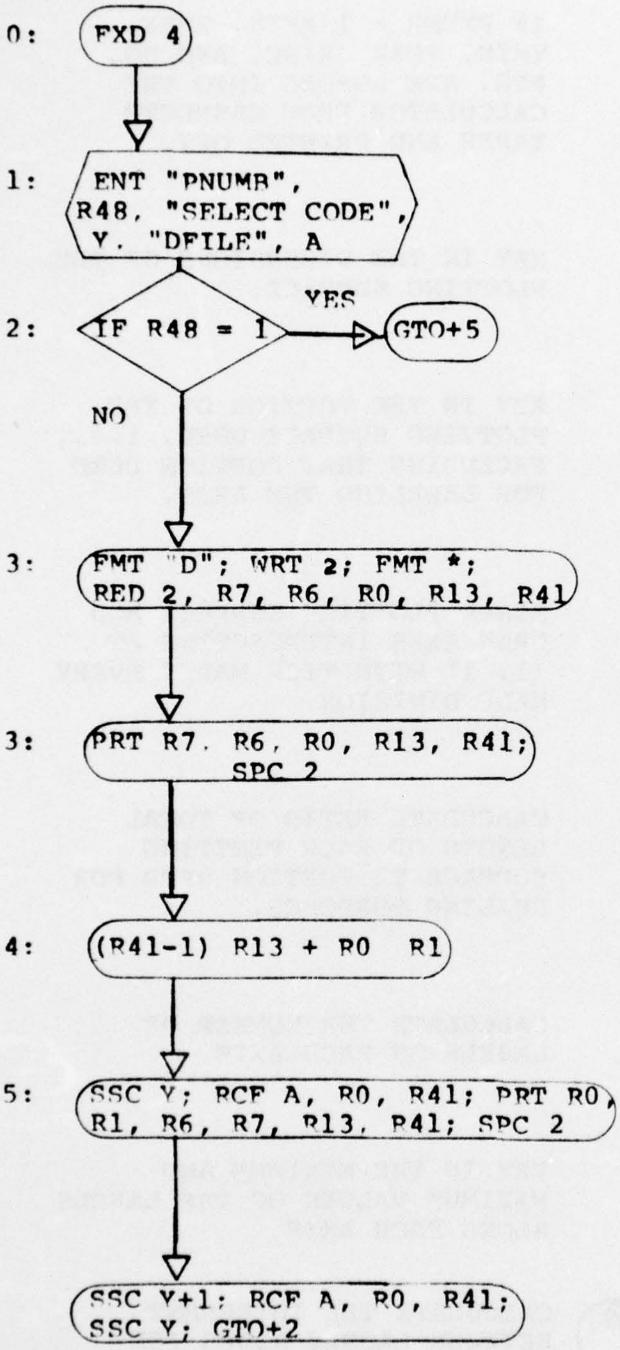
Program steps 0 - 6 determine where the input data for plotting is stored and which cassette unit(s) are used. The input data (i.e., min. and max. values, increment between values of independent variable, and total number of points) is recorded into a cassette file and each value is printed out to let the operator know where to scale the plot.

Statements 8 - 12 determine the length of the axes, the number of tick marks and draw in the intersecting axes.

Program steps 13 - 23 calculate the scales of the X and Y axes respectively and the number of labels on each axis, and write the labels on each axis.

Statements 24-43 read the data from punched cards and record it onto cassette tapes. The program determines the number of cards to be read, the number of points to be stored and the proper cassette file for data storage. There must be five data points per card in this version of the program.

Statements 44-52 reload the data stored on cassette tapes back into the calculator from the cassette tapes, beginning with the first file stored. The data is then automatically scaled and plotted.



SPECIFY ACCURACY OF PRINTED OR DISPLAY NUMBERS. FOUR DIGITS AFTER DECIMAL PT.

KEYED IN BY OPERATOR. PNUMB = 1 FOR DATA ON CARDS OR PAPER TAPE AND ANY OTHER NUMBER IF ON CASSETTE TAPES. SELECT CODE SPECIFIES CASSETTE UNIT USED. DFILE IS FILE CONTAINING PLOT DATA, i.e., # PTS., XMIN, ETC. TEST PNUMB TO DETERMINE WHERE DATA IS.

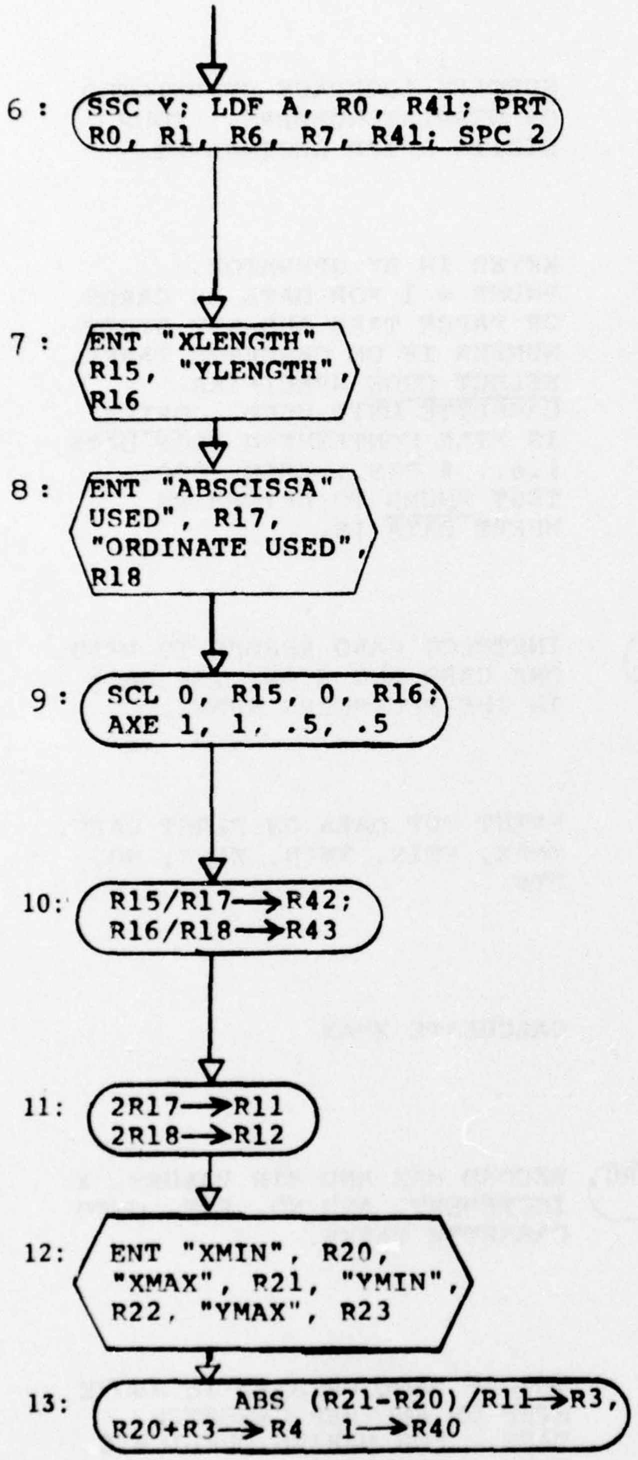
INSTRUCT CARD READER TO READ ONE CARD AND STORE THE DATA IN THE REGISTERS SHOWN.

PRINT OUT DATA ON FIRST CARD. YMAX, YMIN, XMIN, XINC, NO. PTS.

CALCULATE XMAX

RECORD MAX AND MIN VALUES, X INCREMENT, AND NO. PTS. ONTO CASSETTE TAPES.

RECORD SAME DATA AS IN ABOVE STEP ON ANOTHER CASSETTE TAPE. FOR MAKING DUPLICATE TAPE. OPTIONAL



IF PNUMB = 1 XMIN, XMAX, YMIN, YMAX, XINC, AND NO. PTS. ARE LOADED INTO THE CALCULATOR FROM CASSETTE TAPES AND PRINTED OUT.

KEY IN THE DIMENSIONS OF THE PLOTTING SURFACE.

KEY IN THE PORTION OF THE PLOTTING SURFACE USED, i.e., EXCLUDING THAT PORTION USED FOR LABELING THE AXES.

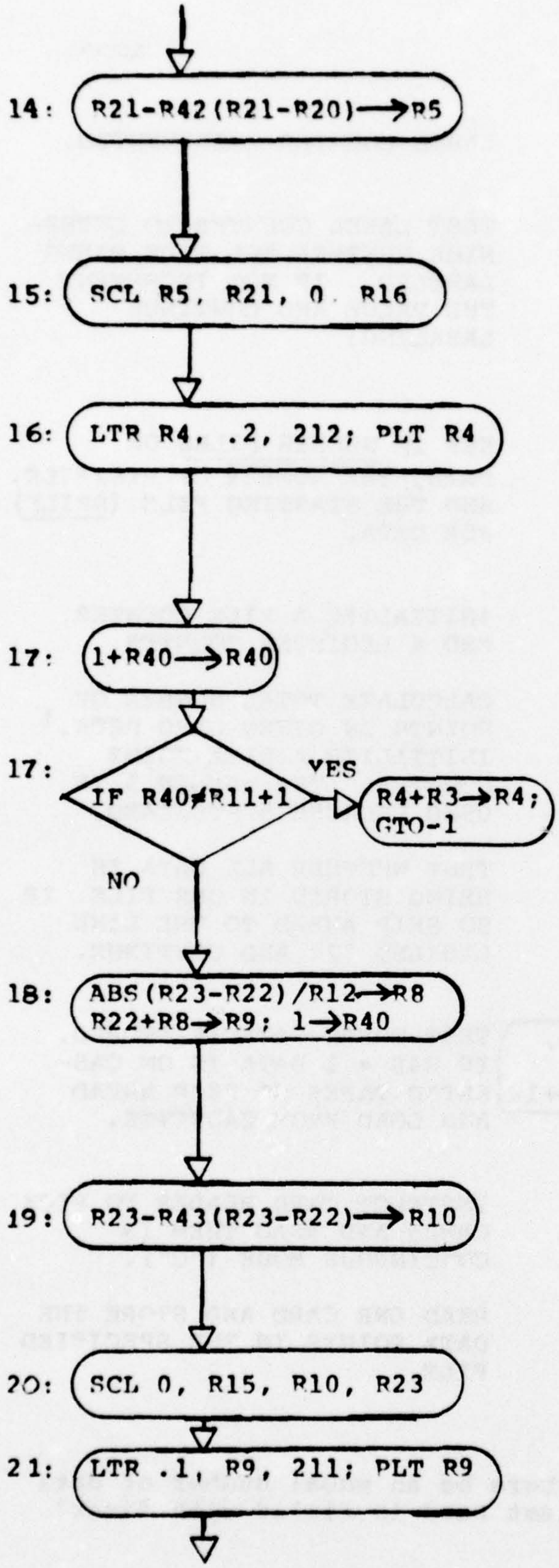
SCALE PLOTTING SURFACE AND DRAW AXES INTERSECTING AT (1, 1) WITH TICK MARKS EVERY HALF DIVISION.

CALCULATE RATIO OF TOTAL LENGTH OF EACH PLOTTING SURFACE TO PORTION USED FOR SCALING PURPOSES.

CALCULATE THE NUMBER OF LABELS ON EACH AXIS.

KEY IN THE MINIMUM AND MAXIMUM VALUES OF THE LABELS ALONG EACH AXIS.

CALCULATE THE INCREMENT BETWEEN LABELS ALONG THE X-AXIS. CALCULATE VALUE OF FIRST LABEL. INITIALIZE & LABEL COUNTER.



CALCULATE X VALUE FOR SCALING ON BASIS OF X DIMENSION USED.

SCALE PLOTTER FOR LABELING X-AXIS.

SPECIFY (X, Y) POSITION AND THE HEIGHT, WIDTH AND DIRECTION (HWD) OF LETTERING AND WRITE THE LABELS AT EACH TICK MARK.

LABEL COUNTER INCREMENTED.

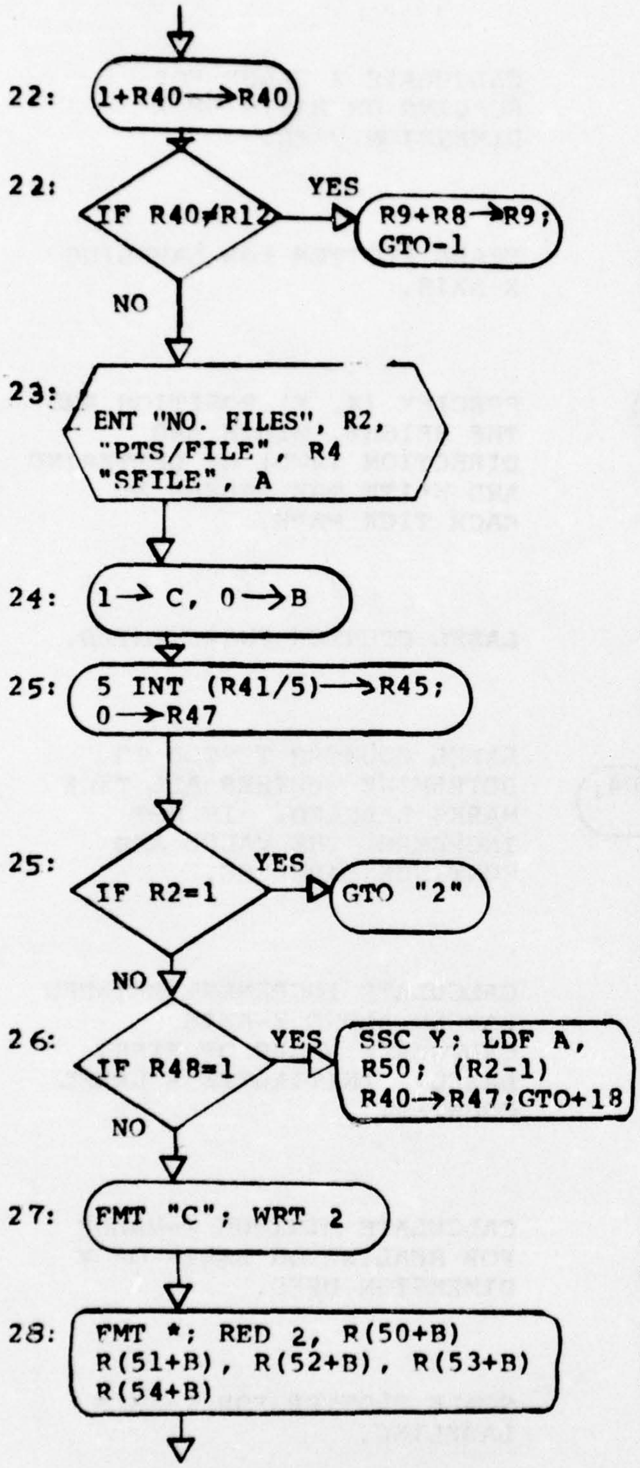
LABEL COUNTER TESTED TO DETERMINE WHETHER ALL TICK MARKS LABELED. IF NOT INCREMENT THE VALUE AND CONTINUE LABELING.

CALCULATE INCREMENT BETWEEN LABELS ALONG Y-AXIS. CALCULATE VALUE OF FIRST LABEL. INITIALIZE A LABEL COUNTER.

CALCULATE MINIMUM Y-VALUE FOR SEALING ON BASIS OF Y DIMENSION USED.

SCALE PLOTTER FOR Y-AXIS LABELING.

SPECIFY (X,Y) POSITION AND HEIGHT, WIDTH, AND DIRECTION (HWD) OF LETTERING. WRITE THE LABELS ON Y-AXIS.



LABEL COUNTER INCREMENTED.

TEST LABEL COUNTER TO DETERMINE WHETHER ALL TICK MARKS LABELED. IF NOT INCREMENT THE VALUE AND CONTINUE LABELING.

KEY IN NUMBER FILES OF DATA, THE NUMBER OF PTS/FILE, AND THE STARTING FILE (SFILE) FOR DATA.

INITIALIZE A FILE COUNTER AND A REGISTER COUNTER.

CALCULATE TOTAL NUMBER OF POINTS IN GIVEN CARD DECK.[1] INITIALIZE A DATA POINT COUNTER. THE NUMBER 5 IS USED BECAUSE 5 PTS/CARD.

TEST WHETHER ALL DATA IS BEING STORED IN ONE FILE. IF SO SKIP AHEAD TO THE LINE LABELED "2" AND CONTINUE.

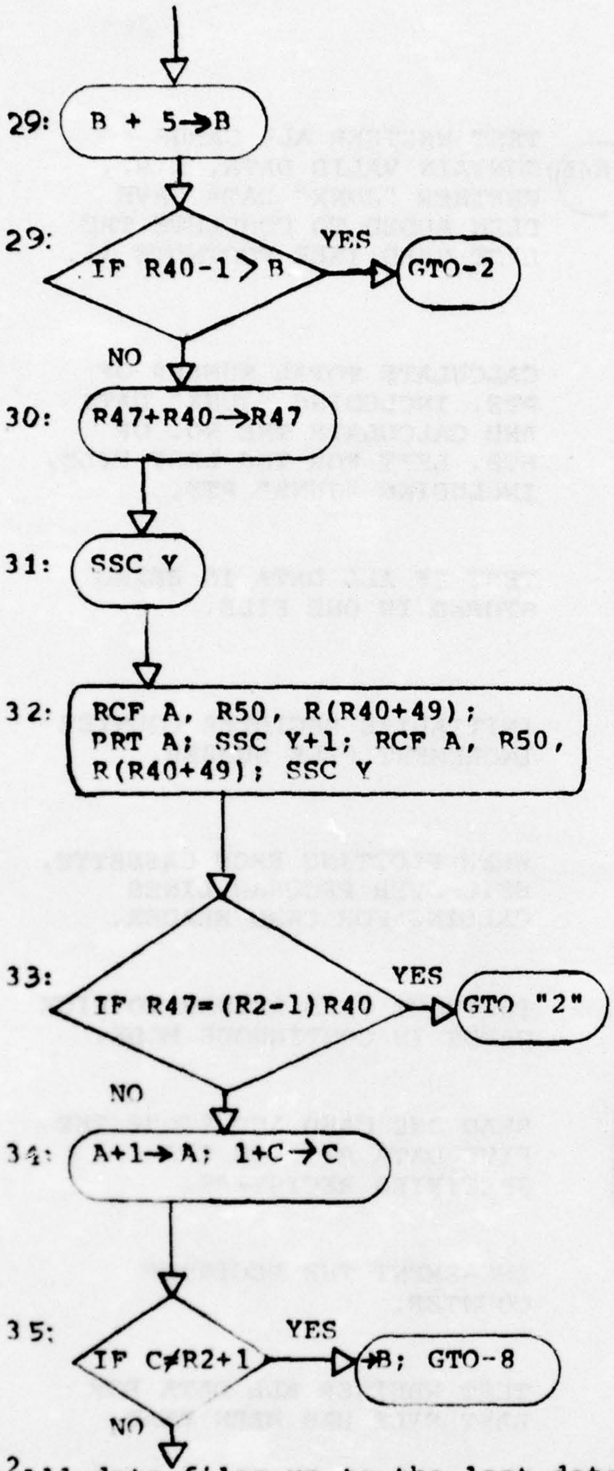
TEST WHERE DATA IS STORED. IF R48 = 1 DATA IS ON CASSETTE TAPES SO SKIP AHEAD AND LOAD FROM CASSETTE.

INSTRUCT CARD READER TO PICK CARDS AND READ THEM IN CONTINUOUS MODE ("C").

READ ONE CARD AND STORE THE DATA POINTS IN THE SPECIFIED FILE.

---

[1] The card reader requires that there be an equal number of data pts. per card; sometimes the last card is filled with "junk" pts. to satisfy this condition.



REGISTER COUNTER INCREMENTED.

TEST WHETHER ENOUGH DATA HAS BEEN READ TO FILL THE CURRENT FILE.

INCREMENT THE DATA POINT COUNTER.

CALL APPROPRIATE CASSETTE UNIT.

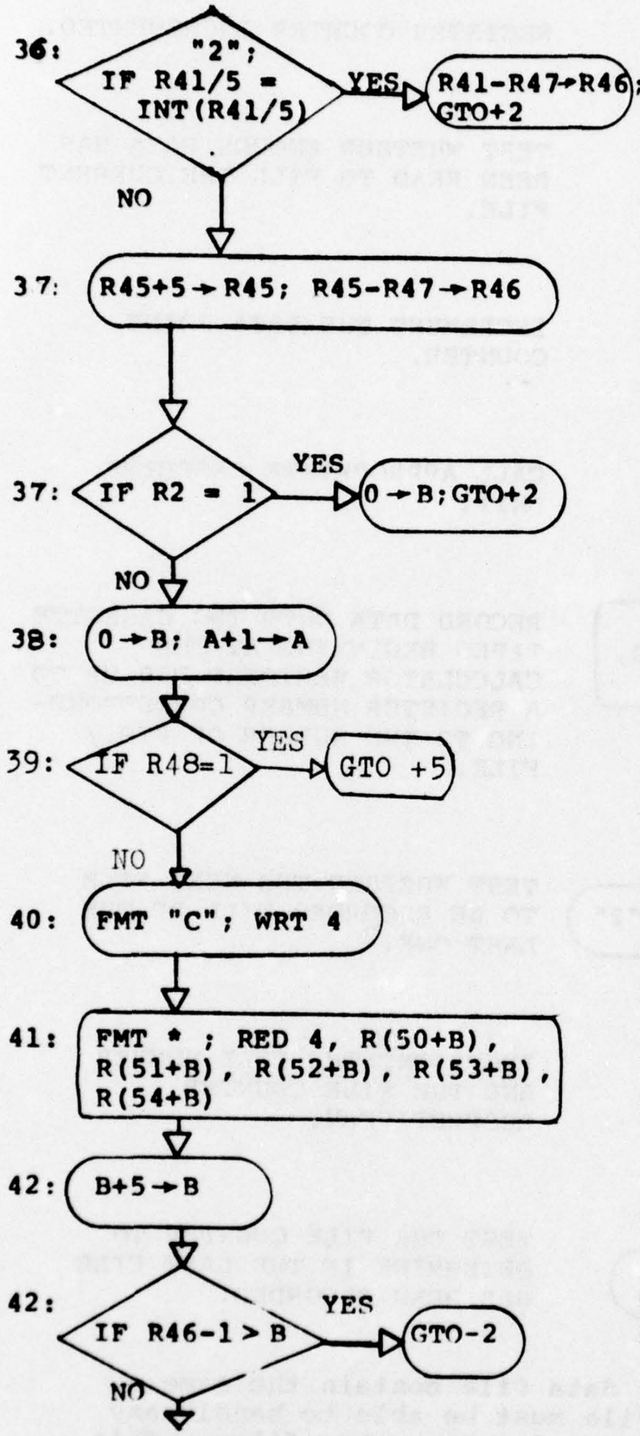
RECORD DATA ONTO TWO CASSETTE TAPES BEGINNING AT THE CALCULATOR REGISTER R50 UP TO A REGISTER NUMBER CORRESPONDING TO THE NUMBER OF PTS./FILE.

TEST WHETHER THE NEXT FILE TO BE RECORDED WILL BE THE LAST ONE.[2]

INCREMENT THE FILE NUMBER AND THE FILE COUNTER RESPECTIVELY.

TEST THE FILE COUNTER TO DETERMINE IF THE LAST FILE HAS BEEN RECORDED.

[2] All data files up to the last data file contain the same number of points. The last file must be able to handle any number of points up to the number for the other files. This requires some extra bookkeeping.



TEST WHETHER ALL CARDS CONTAIN VALID DATA, i.e., WHETHER "JUNK" DATA HAVE BEEN ADDED TO COMPLETE THE LAST CARD (SEE FOOTNOTE 1).

CALCULATE TOTAL NUMBER OF PTS. INCLUDING "JUNK" DATA AND CALCULATE THE NO. OF PTS. LEFT FOR THE LAST FILE, INCLUDING "JUNK" PTS.

TEST IF ALL DATA IS BEING STORED IN ONE FILE.

INITIALIZE REGISTER COUNTER INCREMENT FILE NUMBER.

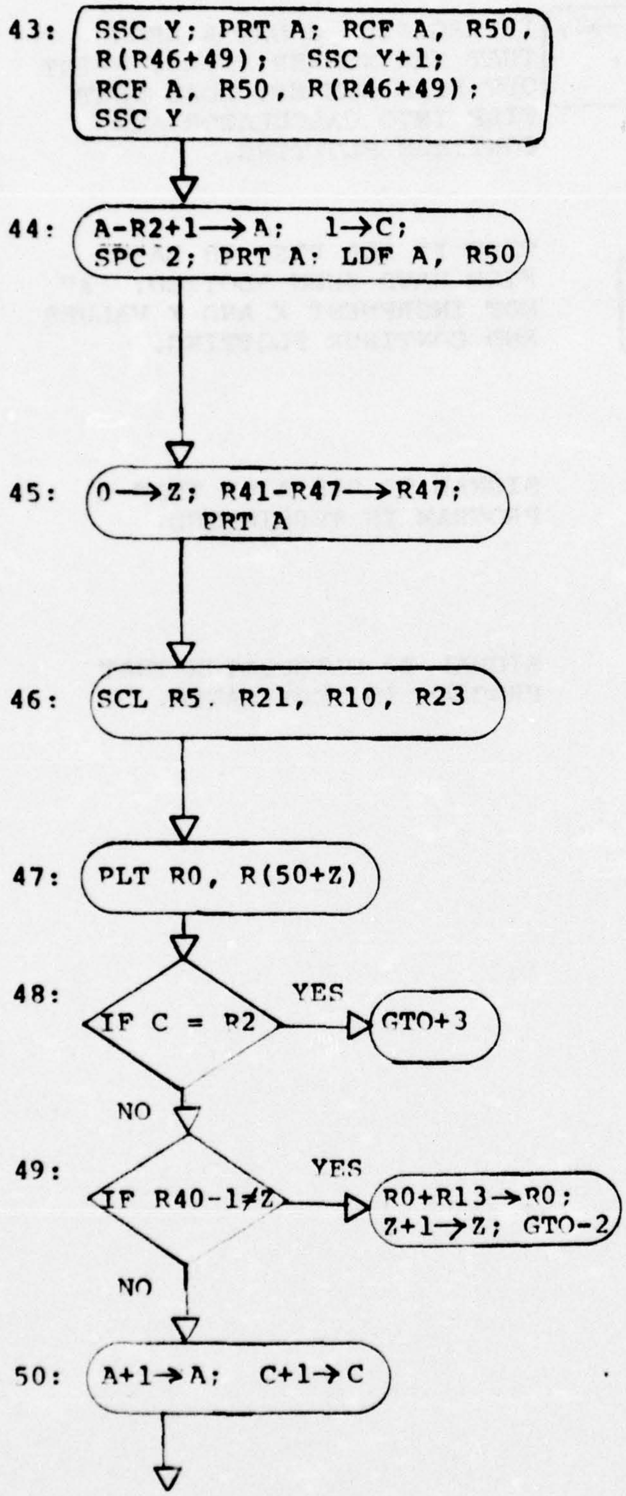
WHEN PLOTTING FROM CASSETTE, SKIP OVER PROGRAM LINES CALLING FOR CARD READER.

INSTRUCT CARD READER TO PICK CARDS IN CONTINUOUS MODE.

READ ONE CARD AND STORE THE FIVE DATA PTS. IN THE SPECIFIED REGISTERS.

INCREMENT THE REGISTER COUNTER.

TEST WHETHER ALL DATA FOR LAST FILE HAS BEEN READ.



PRINT CURRENT FILE NUMBER. RECORD THE DATA ONTO TWO CASSETTE TAPES. CAN REMOVE THAT WHICH IS INSIDE BRACKETS IF NO DUPLICATE TAPE DESIRED.

CALCULATE THE FIRST FILE RECORDED. INITIALIZE A FILE COUNTER. PRINT OUT THE FILE NUMBER AND LOAD THE DATA FROM THE FIRST FILE INTO THE CALCULATOR.

INITIALIZE A REGISTER COUNTER. CALCULATE NO. PTS. IN LAST FILE. PRINT OUT FILE NUMBER.

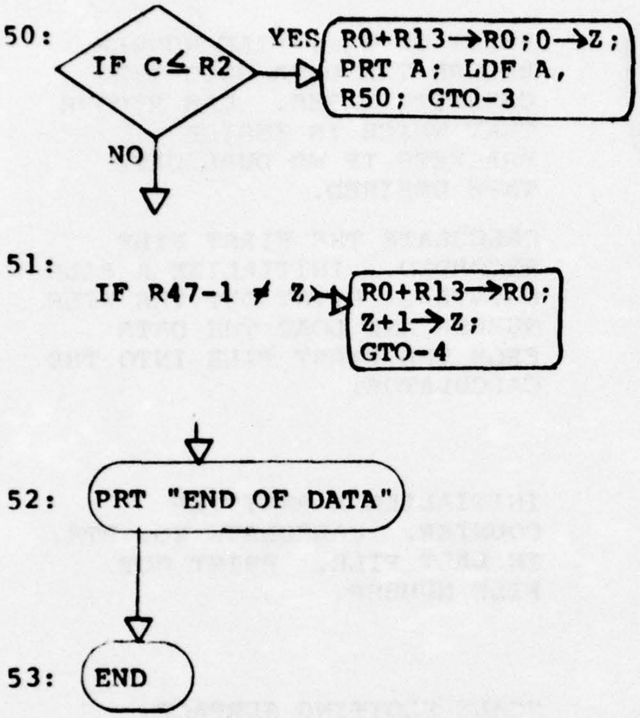
SCALE PLOTTING SURFACE ACCORDING TO PREVIOUSLY CALCULATED VALUES.

PLOT ONE DATA POINT.

TEST REGISTER COUNTER TO DETERMINE IF CURRENT FILE IS THE LAST FILE.

TEST IF ALL PTS. IN FILE HAVE BEEN PLOTTED. IF NO INCREMENT X AND Y VALUES AND CONTINUE PLOTTING.

INCREMENT FILE NUMBER AND FILE COUNTER.



IF REGISTER COUNTER LESS THAT MAX NUMBER FILES, PRINT OUT FILE NUMBER, LOAD NEXT FILE INTO CALCULATOR AND CONTINUE PLOTTING.

TEST IF ALL PTS. IN LAST FILE HAVE BEEN PLOTTED. IF NOT INCREMENT X AND Y VALUES AND CONTINUE PLOTTING.

SIGNAL TO OPERATOR THAT PROGRAM IS TERMINATED.

SIGNAL TO CALCULATOR THAT PROGRAM IS TERMINATED.

Sample output plots are contained in Figures 3 through 9. Figures 3 and 4 are Time Scanned Array Radar (TSAR) antenna transmit patterns. The values along the abscissa are angles in degrees and the values along the ordinate are in decibels (dB). Figures 5 and 6 are plots of magnitude and phase respectively of a TSAR pulse train looking through the main lobe shown in Figure 3. The abscissa of both plots is in nanoseconds and the ordinate in Figure 5 is in decibels and in degrees in Figure 6. Figures 7, 8 and 9 again show the magnitude and phase of the pulse train looking through the main lobe shown in Figure 3. Figure 7 shows the pulse train for a much longer time interval than does Figure 5. The time delay and the number of pulses required to load the array are illustrated here. Figure 8 illustrates the pulse-to-pulse phase shift and the phase gradient of the pulse train in Figure 7. Figure 9 is an expanded plot of Figure 8.

Figure 3. Sample TSAR Beam Pattern

Figure 4. Sample TSAR Beam Pattern

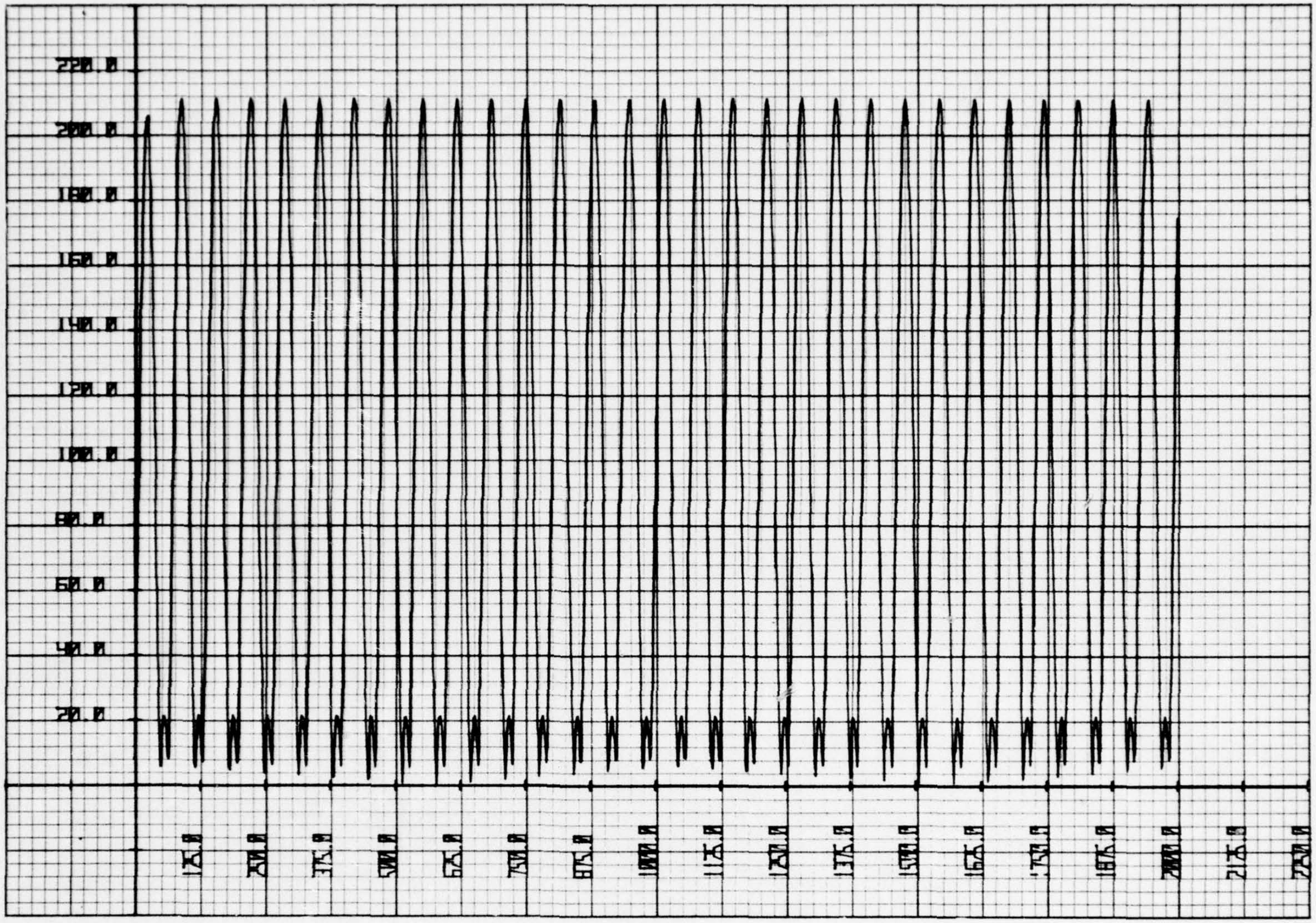


Figure 5. Magnitude of Pulse Train Leaving TSAR Main Beam

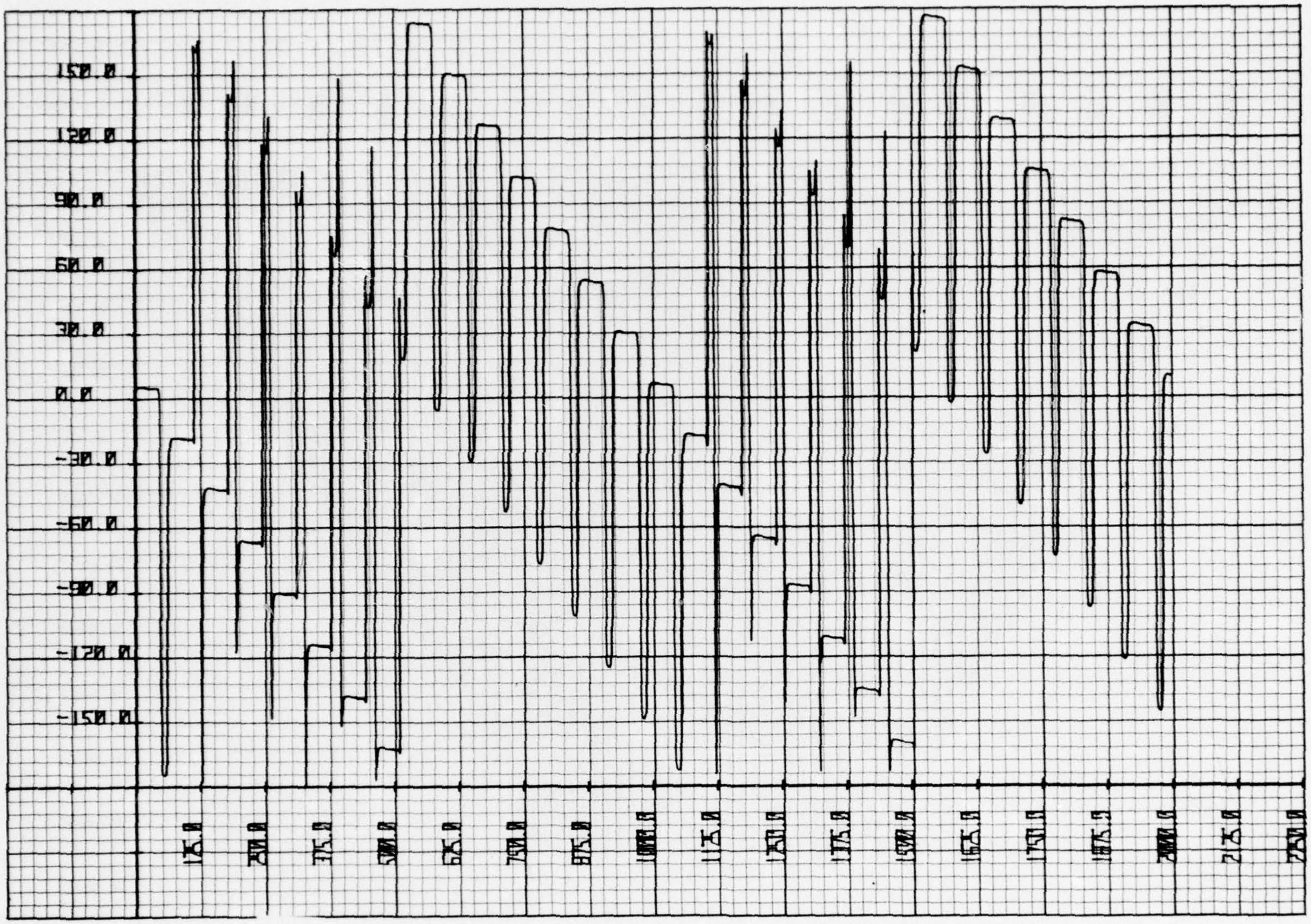


Figure 6. Phase of Pulse Train Leaving TSAR Main Beam

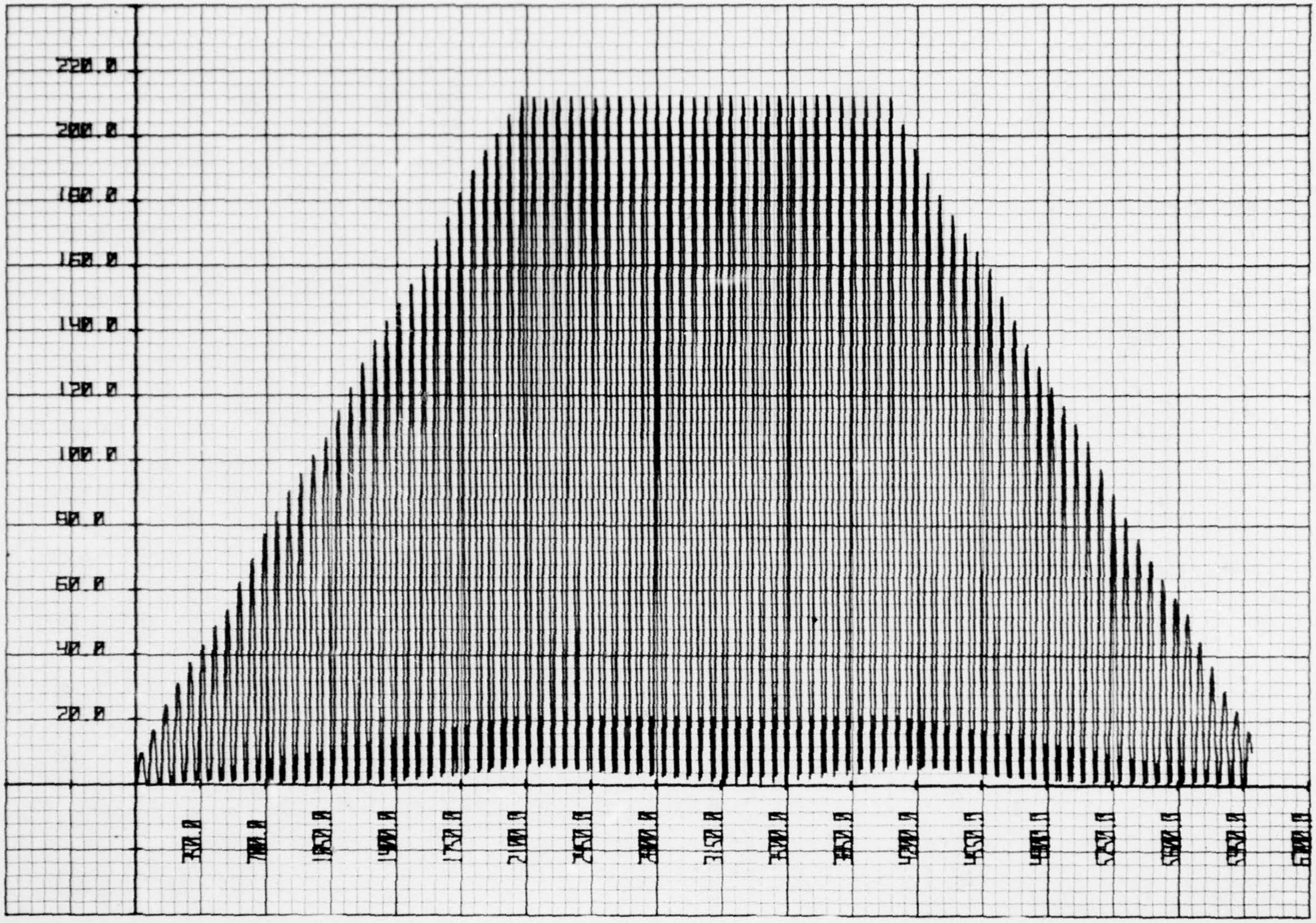


Figure 7. Magnitude of Pulse Train Filling up the Array

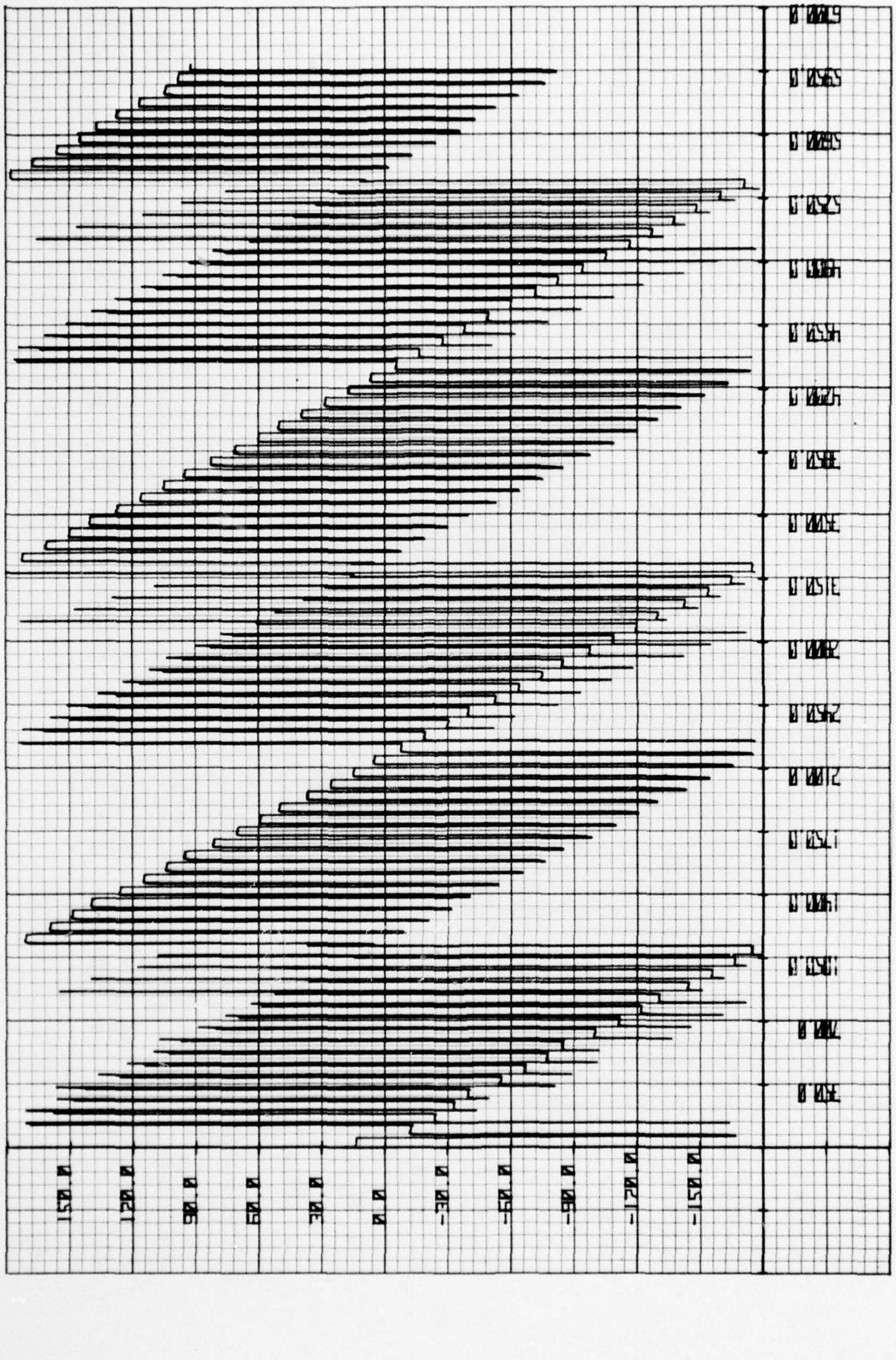


Figure 8. Phase of Pulse Train Filling up the Array

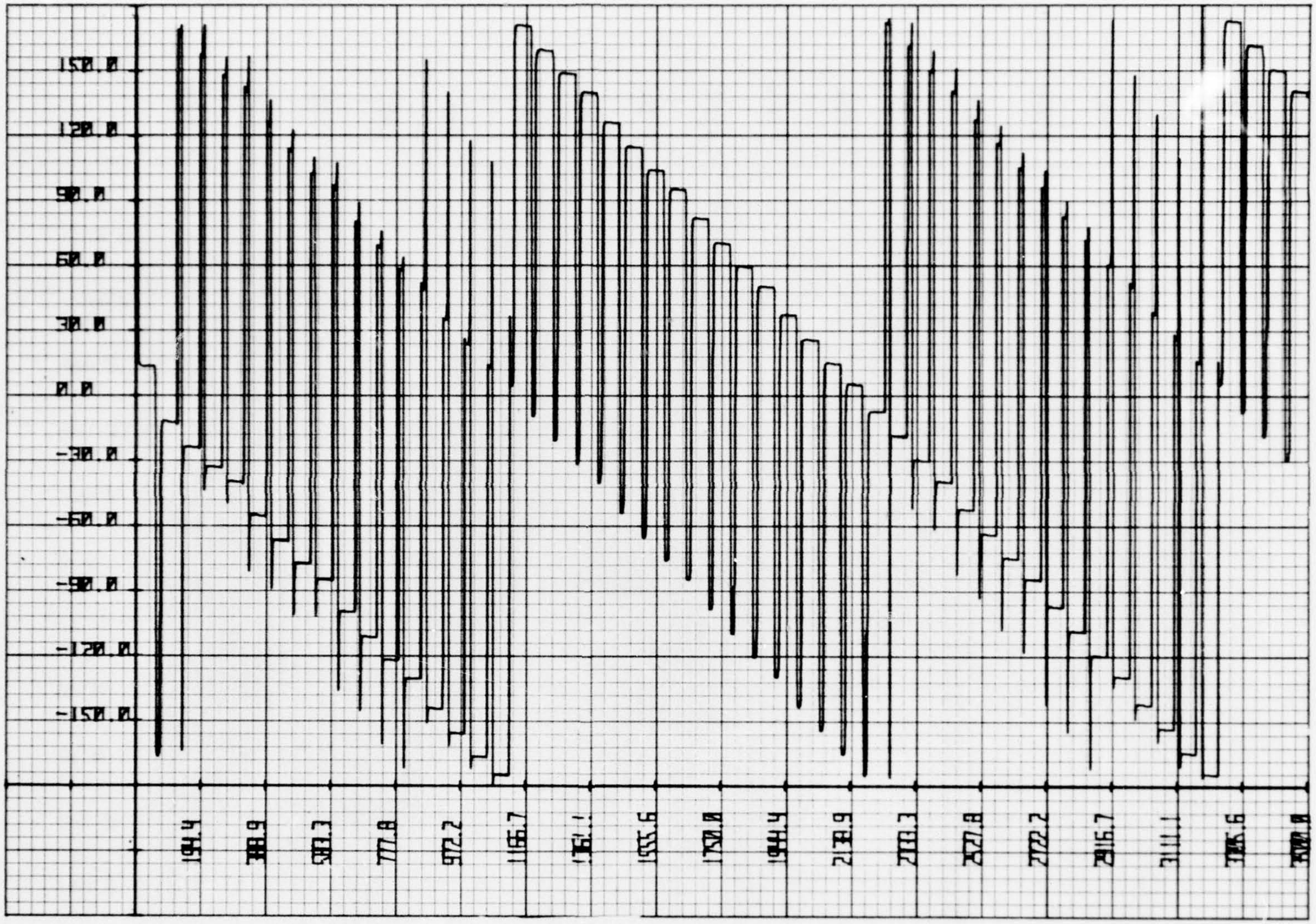


Figure 9. Expanded Plot of Phase of Pulse Train Filling up the Array

VI. RAE Plot Program

The R A E plot program (shown on page 34) will plot slant range, azimuth, and elevation-vs-time all on the same graph; or slant range rate, azimuth rate, and elevation rate-vs-time all on the same graph.* The data is read into the "HP system" from punched paper tape.

Program steps 0 and 1 specify the dimensions of the abscissa and ordinate and draw in the intersecting axes with tick marks. Steps 2 through 18 define the variable to be plotted and the units, and label the plot accordingly (see the illustrations on page 59).

Statement 19 calculates the number of labels to be written on the axes. Steps 20 through 28 define the limits of time for labeling purposes and write the proper values of time at each tick mark.

In statements 29 through 34, the limits of elevation, azimuth, and slant range are keyed in and subsequently printed out to insure that the desired values have been keyed in correctly. From Statements 35 to 57, the ordinate is labeled with three scales; i.e., elevation, azimuth, and slant range.

Data is read in and stored on cassette tapes as defined by program Steps 58 to 75, and plotted in Steps 76 through 97.

---

*This program was modified from an original program written by John Cleary. (See Appendix E.)

```
0:  ENT "XLENGTH",R27,"YLENGTH",R28,"XUSED",R29,"YUSED
    ",R30,"S1",R0 [
1:  SCL 0,R27,0,R28;AXE 1,1,.5,.5;LTR 2,0,221 [
2:  PLT "TIME (MINUTES AFTER BURNOUT)" [
3:  IF R0>0;GTO "LARD" [
4:  SPC 2;PRT "LABEL R A E" [
5:  LTR 0,9.8,221;PLT "E" [
6:  LTR 0.4,9.8,221;PLT "A" [
7:  LTR 0.8,9.8,221;PLT "R" [
8:  LTR .1,9.3,221;PLT "DEG" [
9:  LTR 0.7,9.3,221;PLT "NM" [
10: GTO "EN" [
11: "LARD";SPC 2;PRT "LABEL R A E RATES" [
12: LTR 0,9.8,221;PLT "E" [
13: LTR 0.4,9.8,221;PLT "A" [
14: LTR 0.8,9.8,221;PLT "R" [
15: LTR .1,9.6,221;PLT "DEG" [
16: LTR .1,9.3,221;PLT "SEC" [
17: LTR 0.7,9.6,221;PLT "FT" [
18: LTR 0.6,9.3,221;PLT "SEC"[
19: R27/R29 TO R42;R28/R30 TO R43;2*R29 TO R31;2*R30 TO R32 [
20: "EN";ENT "TMIN=",R1,"TMAX=",R2 [
21: ABS (R2-R1)/R31 TO R3;R1+R3 TO R14;1 TO R13 [
22: PRT "TMIN=",R1,"TMAX=",R2,"TINC",R3;FXD 1 [
23: R2-R42(R2-R1) TO R18 [
```

Figure 10. RAE Plot Program

```
24:  SCL R18,R2,0,R28 [
25:  LTR R14,0.5,212 [
26:  PLT R14 [
27:  R13+1 TO R13;R14+R3 TO R14 [
28:  IF R13 # R31+1;GTO -3 [
29:  FXD 1;ENT "EMIN=",R4,"EMAX=",R5 [
30:  ENT "AMIN=",R7,"AMAX=",R8 [
31:  ENT "RMIN=",R10,"RMAX=",R11 [
32:  PRT "EMIN=",R4,"EMAX=",R5 [
33:  PRT "AMIN=",R7,"AMAX=",R8 [
34:  PRT "RMIN=",R10,"RMAX=",R11 [
35:  ABS (R5-R4)/(R32/2) TO R6;R4+R6 TO R15;1 TO R13 [
36:  R5-R43(R5-R4) TO R19 [
37:  SCL 0,R27,R19,R5 [
38:  LTR 0,R15,211 [
39:  PLT R15 [
40:  R13+1 TO R13;R15+R6 TO R15 [
41:  IF R13 # R32-10;GTO -3 [
42:  ABS (R8-R7)/(R32/2) TO R9 [
43:  R7+R9/2 TO R16;1 TO R13 [
44:  R8-R43(R8-R7) TO R20 [
45:  SCL 0,R27,R20,R8 [
46:  LTR .4,R16,211 [
47:  PLT R16 [
48:  R13+1 TO R13;R16+R9 TO R16 [
```

FIGURE 10. RAE PLOT PROGRAM (CONT'D)

```
49:  IF R13 # R32-9;GTO -3 [

50:  FXD 1;ABS (R11-R10)/(R32/2) TO R12 [

51:  R10+R12 TO R17;1 TO R13 [

52:  R11-R43(R11-R10) TO R21 [

53:  SCL 0,R27,R21,R11 [

54:  LTR 0.6,R17,211 [

55:  PLT R17 [

56:  R13+1 TO R13;R17+R12 TO R17 [

57:  IF R13 # R32-10;GTO -3 [

58:  ENT "SELECT CODE",Y,"NO. FILES",R22,"PTS/FILE",R40

     ,"SFILE",A [

59:  ENT "NO. PTS",R41,"PNUMB",R48;IF R48=1;GTO "PT" [

60:  "1";1 TO C;0 TO Z;SSC Y [

61:  INT (R41/3) TO R45;0 TO R47 [

62:  "1READ";IF R22=3;R45 TO R46;0 TO Z;GTO +8 [

63:  "READ";RED 1,R(50+Z) [

64:  Z+1 TO Z;IF R40 # Z;GTO -1 [

65:  R47+R40 TO R47 [

66:  RCF A,R50,R(R40+49);C+1 TO C;PRT A [

67:  R45-R47 TO R46 [

68:  IF R40 <= R46;A+1 TO A;0 TO Z;GTO "READ" [

69:  A+1 TO A;0 TO Z [

70:  RED 1,R(50+Z) [

71:  Z+1 TO Z;IF R46 # Z;GTO -1 [

72:  RCF A,R50,R(R46+49);C+1 TO C;PRT A [
```

FIGURE 10. RAE PLOT PROGRAM (CONT'D)

```
73:  IF C=R22+1;GTO +3 [

74:  IF R22=3;A+1 TO A;GTO "1READ" [

75:  0 TO Z TO R47;A+1 TO A;GTO "READ" [

76:  "PT";ENT "START PLOT FILE",A,"TMIN",R23,"TMAX",R25
     ,"TINC",R24 [

77:  SSC Y;0 TO Z TO R47;1 TO C;PRT A;LDF A,R50 [

78:  SCL R18,R2,R19,R5 [

79:  R23 TO R26 [

80:  IF R22=3;R45 TO R46;GTO +6 [

81:  "PLT";PLT R26,R(50+Z) [

82:  R26+R24 TO R26;Z+1 TO Z;IF R40 # Z;GTO -2 [

83:  R47+R40 TO R47;R45-R47 TO R46 [

84:  IF R40 <= R46;A+1 TO A;C+1 TO C;LDF A,R50;GTO "PLT" [

85:  A+1 TO A;C+1 TO C;0 TO Z;PRT A;LDF A,R50 [

86:  PLT R26,R(50+Z) [

87:  R26+R24 TO R26;Z+1 TO Z;IF R46 # Z;GTO -1 [

88:  PEN ;IF C=R22;GTO +8 [

89:  IF C=R22/3;GTO +2 [

90:  IF C=2R22/3;GTO +3 [

91:  IF R22=3;R23 TO R26;SCL R18,R2,R20,R8;GTO -6 [

92:  R23 TO R26;0 TO Z TO R47;SCL R18,R2,R20,R8;A+1 TO A;C+1 TO C
     ;PRT A;LDF A,R50;GTO "PLT" [

93:  IF R22=3;R23 TO R26;SCL R18,R2,1E03R21,1E03R11;GTO -
     8 [
```

FIGURE 10. RAE PLOT PROGRAM (CONT'D)

```
94:  R23 TO R26;0 TO Z TO R47;SCL R18,R2,1E03R21,1E03R11;A+1 TO
     A;C+1 TO C;PRT A [
95:  LDF A,R50;GTO "PLT" [
96:  PRT "END OF DATA" [
97:  END  [
```

FIGURE 10. RAE PLOT PROGRAM (CONT"D)

The preceding description of program operation has been purposely made general with the intent that it be used as a summary. For further explanation, a flow chart follows which describes program operation in detail.

Table 3 lists the inputs required by the program. The display on the calculator will show the parameter listed under Variable Name in the Table. The value to be keyed in is explained, and the register where the keyed-in value is stored is shown.

In Table 4, each register used in the program is listed with an explanation of the value stored.

INPUTS

Variable Name	Register	Meaning
XLENGTH	R27	Absolute horizontal length of the plotting surface (X-axis).
YLENGTH	R28	Absolute vertical length of the plotting surface (Y-axis).
XUSED	R29	Portion of XLENGTH used for plotting; i.e., excluding the portion used for labeling the axis.
YUSED	R30	Portion of YLENGTH used for plotting; i.e., excluding the portion used for labeling the axis.
S1	R0	A switch. If $S1 > 0$ label Y-axis $\dot{E}$ $\dot{A}$ $\dot{R}$. If $S1 \leq 0$ label Y-axis E A R.
TMIN	R1	Min. value of time for labeling the X-axis.
TMAX	R2	Max. value of time for labeling the X-axis.
EMIN	R4	Min. value of elevation for labeling the Y-axis.
EMAX	R5	Max. value of elevation for labeling the Y-axis.
AMIN	R7	Min. value of azimuth for labeling the Y-axis.
AMAX	R8	Max. value of azimuth for labeling the Y-axis.
RMIN	R10	Min. value of slant range for labeling the Y-axis.
RMAX	R11	Max. value of slant range for labeling the Y-axis.
SELECT CODE	Y	Select code of the cassette unit where data is to be stored.

TABLE 3
Inputs - RAE Program

INPUTS (CONT.)

Variable Name	Register	Meaning
NO. FILES	R22	The total no. of files used to store the data.
PTS/FILE	R40	The number of pts. to be stored in each file.
SFILE	A	The first file where data is to be stored.
NO. PTS	R41	The total no. of pts.
PNUMB	R48	A switch. If PNUMB = 1 load data from cassette and plot. If PNUMB ≠ 1 read data from paper tape.
START PLOT FILE	A	The no. of the first file to be plotted.
TMIN	R23	Min. value of time (independent variable) to be plotted.
TMAX	R25	Max. value of time (indep. var.) to be plotted.
TINC	R24	Increment between values of the independent variable.

TABLE 3 (CONT'D)

REGISTER TABLE

Register	Value Stored
R0	Switch to determine whether to label the Y-axis with $\dot{E}$ $\dot{A}$ $\dot{R}$ or E A R (input).
R1	Min. time value for labeling (input).
R2	Max. time value for labeling (input).
R3	Increment between time labels on the abscissa.
R4	Min. elevation value for labeling (input).
R5	Max. elevation value for labeling (input).
R6	Increment between elevation labels on the ordinate.
R7	Min. azimuth value for labeling (input).
R8	Max. azimuth value for labeling (input).
R9	Increment between azimuth labels on the ordinate.
R10	Min. slant range value for labeling (input).
R11	Max. slant range value for labeling (input).
R12	Increment between slant range labels on the ordinate.
R13	Label counter.
R14	Value of each time label.
R15	Value of each elevation label.
R16	Value of each azimuth label.

Table 4
Register Cross-Reference Chart
for RAE Program

REGISTER TABLE (CONT.)

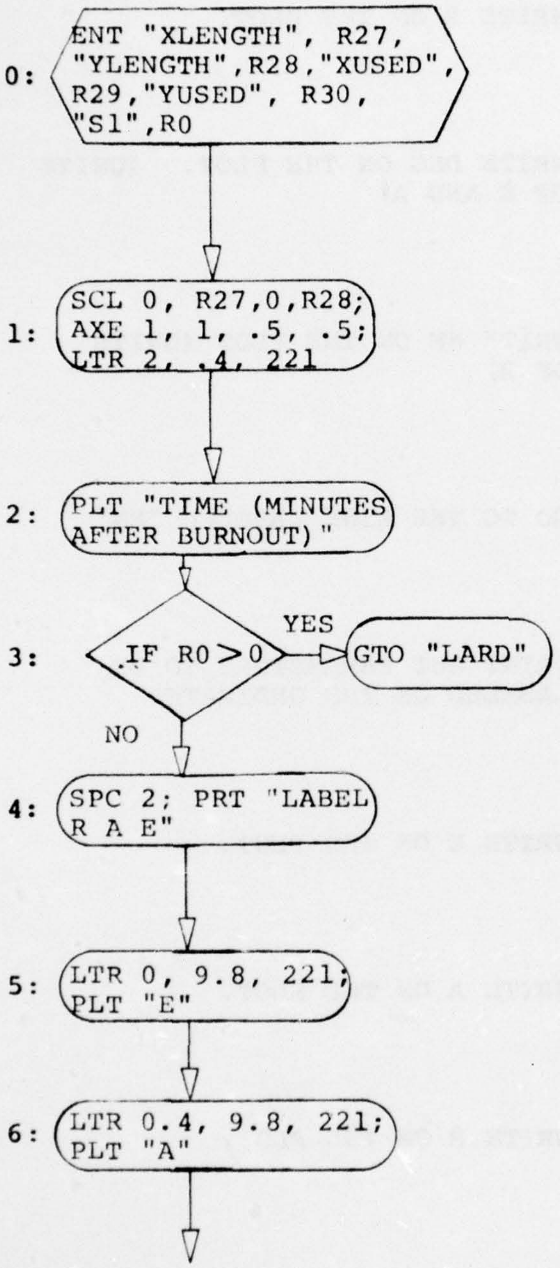
Register	Value Stored
R17	Value of each slant range label.
R18	Xmin value for scale of abscissa based on the ratio of XUSED/XLENGTH (see INPUT Table) and the max. and min. values of time.
R19	Ymin value for scale on ordinate, based on the ratio of YUSED/YLENGTH (see INPUT Table) and the max. and min. values of elevation.
R20	Ymin value for scale on ordinate, based on the ratio of YUSED/YLENGTH (see INPUT Table) and the max. and min. values of azimuth.
R21	Ymin value for scale on ordinate, based on the ratio of YUSED/YLENGTH (see INPUT Table) and the max. and min. values of slant range.
R22	Total number of files used to store data.
R23	Min. time value plotted.
R24	Increment between values of the independent variable-time.
R25	Max. time value plotted.
R26	The time value plotted.
R27	XLENGTH (see INPUT Table)
R28	YLENGTH (see INPUT Table)
R29	XUSED (see INPUT Table)
R30	YUSED (see INPUT Table)
R31	No. of labels on the abscissa
R32	No. of labels on the ordinate.

Table 4 (Cont'd)

REGISTER TABLE (CONT.)

Register	Value Stored
R33-R39	Not used.
R40	No. of pts. per file.
R41	Total no. of pts.
R42	Ratio of XUSED/XLENGTH.
R43	Ratio of YUSED/YLENGTH.
R44	Not used.
R45	No. of pts. per plot.
R46	No. of pts. left to plot.
R47	Point counter.
R48	Switch to determine where data stored (see INPUT Table).
R50 ...	Data storage.
A	File Number
B	Not Used
C	File counter
X	Not Used
Y	Select Code
Z	Register Counter

Table 4 (cont'd)



```
0: ENT "XLENGTH", R27,
   "YLENGTH",R28,"XUSED",
   R29,"YUSED", R30,
   "S1",R0
1: SCL 0, R27,0,R28;
   AXE 1, 1, .5, .5;
   LTR 2, .4, 221
2: PLT "TIME (MINUTES
   AFTER BURNOUT)"
3: IF R0>0   GTO "LARD"
4: SPC 2; PRT "LABEL
   R A E"
5: LTR 0, 9.8, 221;
   PLT "E"
6: LTR 0.4, 9.8, 221;
   PLT "A"
```

KEY IN DIMENSIONS OF PLOTTING SURFACE AND PORTION OF PLOTTING SURFACE USED (i.e. EXCLUDING THAT USED FOR AXIS LABELS). "S1" DETERMINES UNITS ON THE ORDINATE.

SCALE PLOTTING SURFACE ACCORDING TO THE DIMENSIONS SPECIFIED IN THE ENT STATEMENT ABOVE. DRAW AXES INTERSECTING AT (1, 1). SPECIFY POSITION OF THE LABEL ON THE X-AXIS.

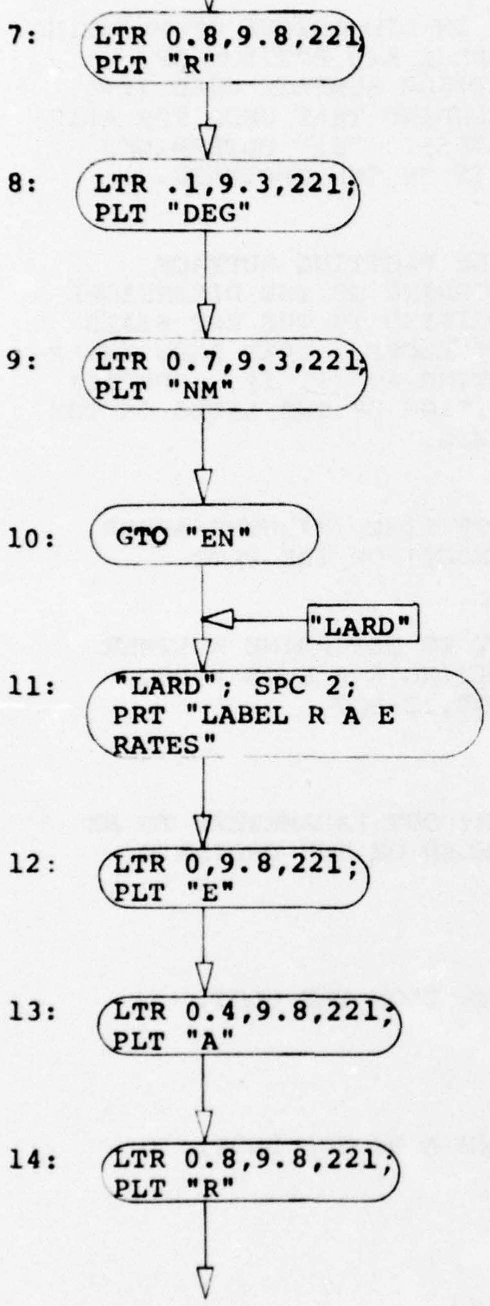
WRITE TIME (MINUTES AFTER BURNOUT) ON THE PLOT.

TEST TO DETERMINE WHETHER PLOTTING R A E OR R-DOT, A-DOT, E-DOT.

PRINT OUT PARAMETERS TO BE LABELED ON THE ORDINATE.

WRITE E ON THE PLOT.

WRITE A ON THE PLOT.



```
7:  LTR 0.8,9.8,221;
    PLT "R"
```

WRITE R ON THE PLOT.

```
8:  LTR .1,9.3,221;
    PLT "DEG"
```

WRITE DEG ON THE PLOT. (UNITS OF E AND A)

```
9:  LTR 0.7,9.3,221;
    PLT "NM"
```

WRITE NM ON THE PLOT (UNITS OF R)

```
10: GTO "EN"
```

GO TO THE LINE LABELED "EN"

```
"LARD"
```

```
11: "LARD"; SPC 2;
    PRT "LABEL R A E
    RATES"
```

PRINT OUT PARAMETERS TO BE LABELED ON THE ORDINATE.

```
12: LTR 0,9.8,221;
    PLT "E"
```

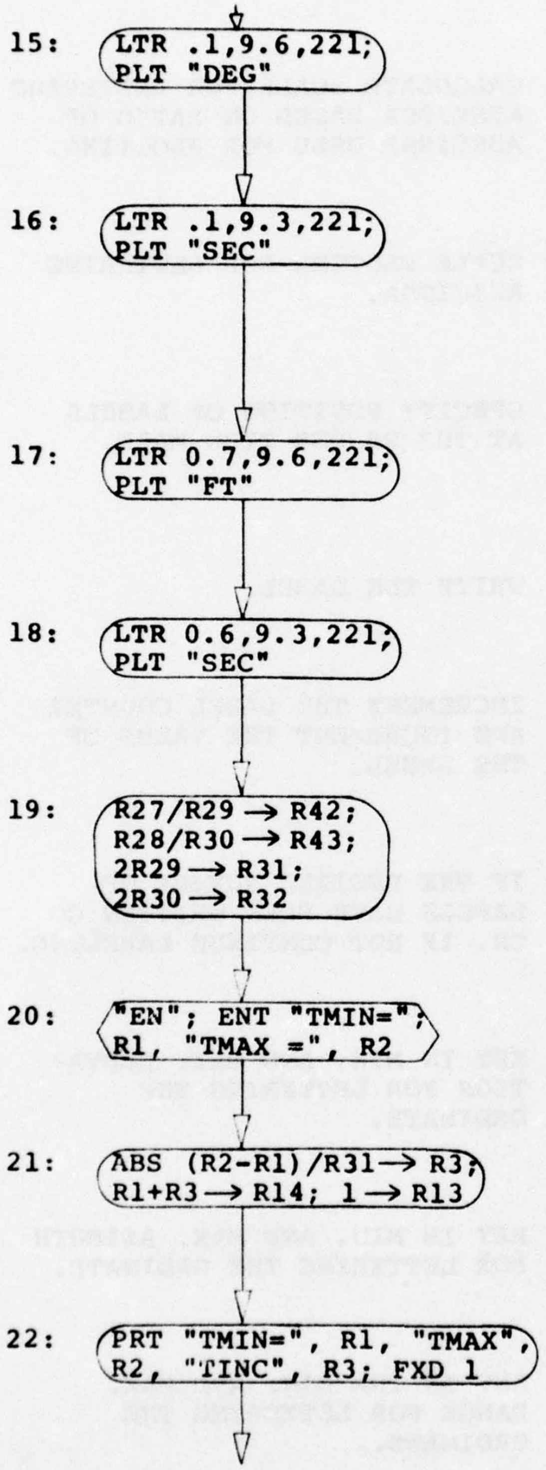
WRITE E ON THE PLOT.

```
13: LTR 0.4,9.8,221;
    PLT "A"
```

WRITE A ON THE PLOT.

```
14: LTR 0.8,9.8,221;
    PLT "R"
```

WRITE R ON THE PLOT.



```
15:  LTR .1,9.6,221;
     PLT "DEG"

16:  LTR .1,9.3,221;
     PLT "SEC"
```

WRITE DEG (DEG/SEC) ON PLOT
SEC

(UNITS OF E AND A)

```
17:  LTR 0.7,9.6,221;
     PLT "FT"

18:  LTR 0.6,9.3,221;
     PLT "SEC"
```

WRITE FT (FT/SEC) ON PLOT
SEC

(UNITS OF R)

```
19:  R27/R29 → R42;
     R28/R30 → R43;
     2R29 → R31;
     2R30 → R32
```

CALCULATE PORTION OF PLOTTING SURFACE USED FOR PLOT (EXCLUDING PORTION USED FOR LABELING AXES). PRELIMINARY CALCULATION FOR THE NUMBER OF LABELS ON EACH AXIS.

```
20:  "EN"; ENT "TMIN=";
     R1, "TMAX =", R2
```

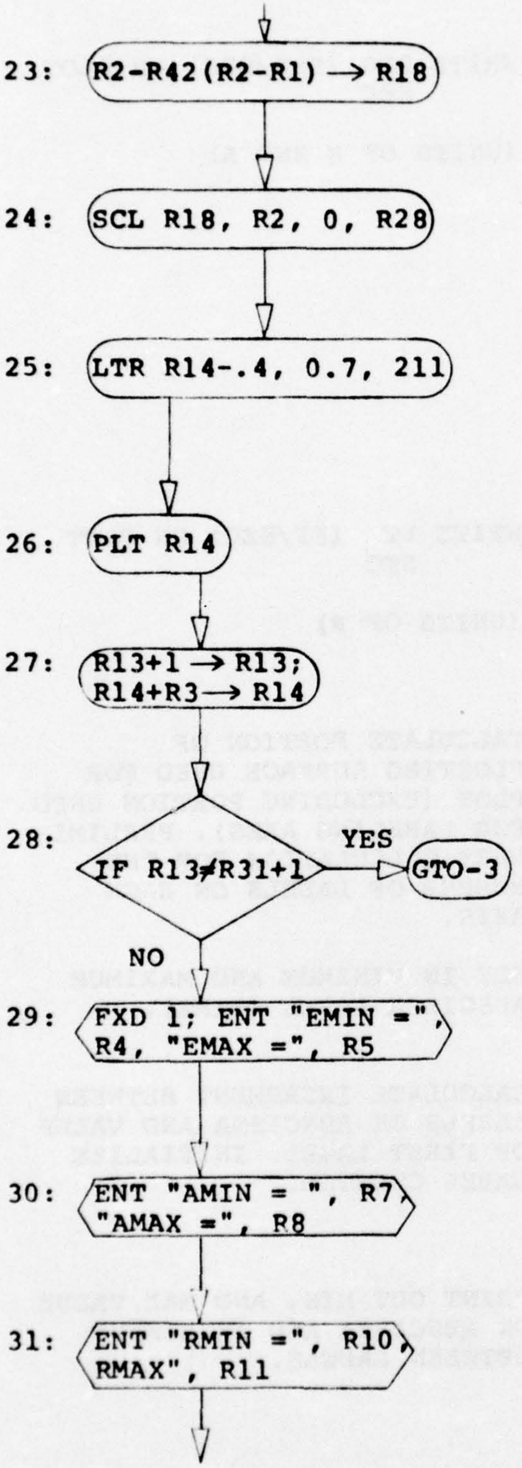
KEY IN MINIMUM AND MAXIMUM ABSCISSA LABEL (TIME).

```
21:  ABS (R2-R1)/R31 → R3;
     R1+R3 → R14; 1 → R13
```

CALCULATE INCREMENT BETWEEN LABELS ON ABSCISSA AND VALUE OF FIRST LABEL. INITIALIZE LABEL COUNTER.

```
22:  PRT "TMIN=", R1, "TMAX",
     R2, "TINC", R3; FXD 1
```

PRINT OUT MIN. AND MAX. VALUE ON ABSCISSA AND INCREMENT BETWEEN LABELS.



CALCULATE SCALE FOR LETTERING ABSCISSA BASED ON RATIO OF ABSCISSA USED FOR PLOTTING.

SCALE PLOTTER FOR LETTERING ABSCISSA.

SPECIFY POSITION OF LABELS AT THE PROPER TICK MARK.

WRITE THE LABEL.

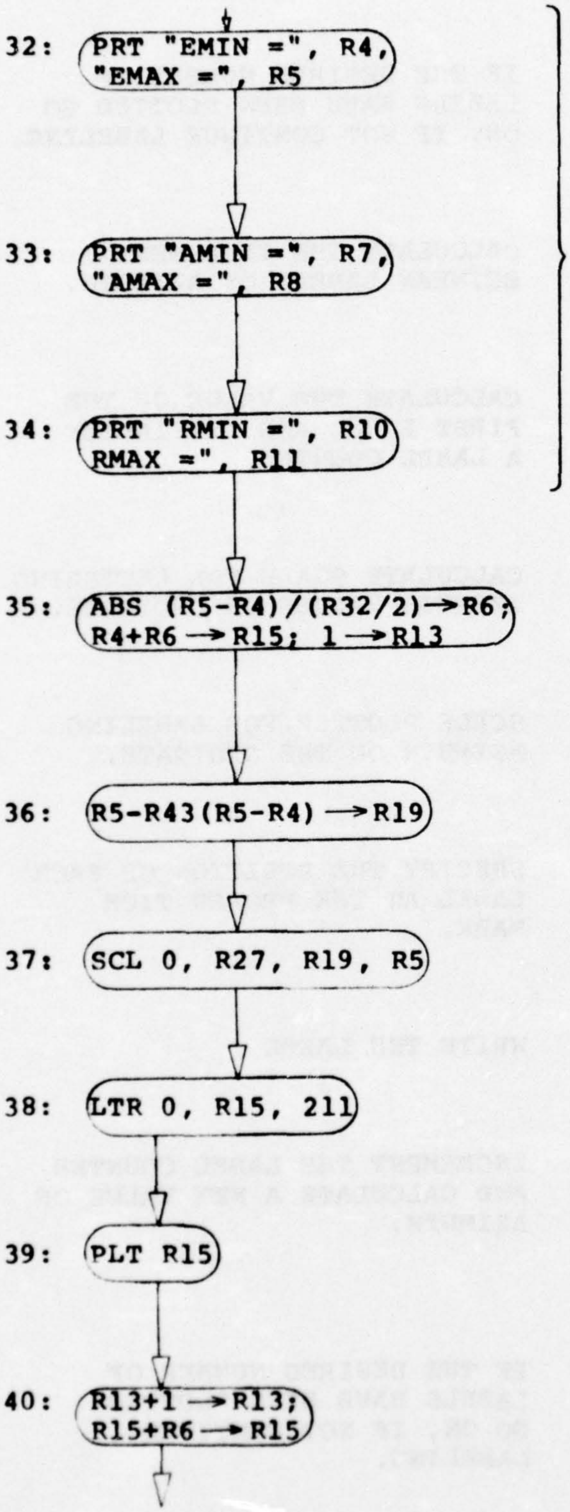
INCREMENT THE LABEL COUNTER AND INCREMENT THE VALUE OF THE LABEL.

IF THE DESIRED NUMBER OF LABELS HAVE BEEN WRITTEN GO ON, IF NOT CONTINUE LABELING.

KEY IN MIN. AND MAX. ELEVATION FOR LETTERING THE ORDINATE.

KEY IN MIN. AND MAX. AZIMUTH FOR LETTERING THE ORDINATE.

KEY IN THE MIN. AND MAX. RANGE FOR LETTERING THE ORDINATE.



PRINT OUT THE MIN. AND MAX. VALUES JUST KEYED IN

CALCULATE INCREMENT BETWEEN ELEVATION LABELS AND VALUE OF FIRST LABEL. INITIALIZE A LABEL COUNTER.

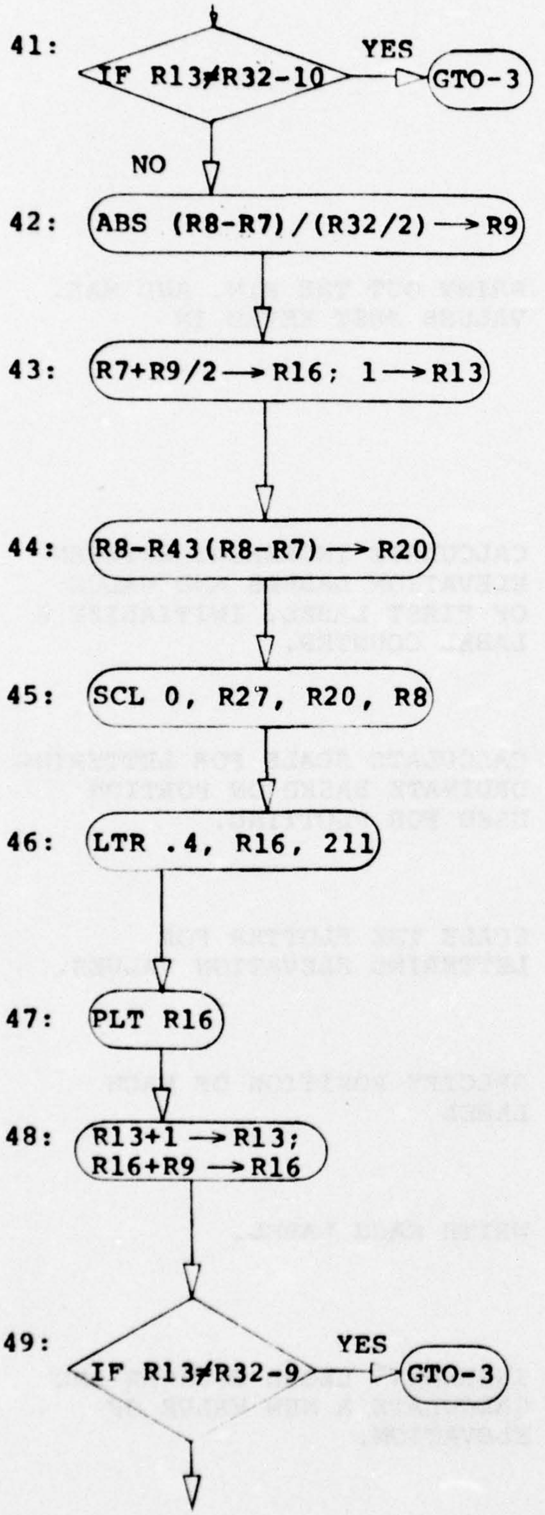
CALCULATE SCALE FOR LETTERING ORDINATE BASED ON PORTION USED FOR PLOTTING.

SCALE THE PLOTTER FOR LETTERING ELEVATION VALUES.

SPECIFY POSITION OF EACH LABEL

WRITE EACH LABEL.

INCREMENT LABEL COUNTER AND CALCULATE A NEW VALUE OF ELEVATION.



IF THE DESIRED NUMBER OF LABELS HAVE BEEN PLOTTED GO ON, IF NOT CONTINUE LABELING.

CALCULATE THE INCREMENT BETWEEN LABELS OF AZIMUTH.

CALCULATE THE VALUE OF THE FIRST LABEL AND INITIALIZE A LABEL COUNTER.

CALCULATE SCALE FOR LETTERING AZIMUTH VALUES ON ORDINATE.

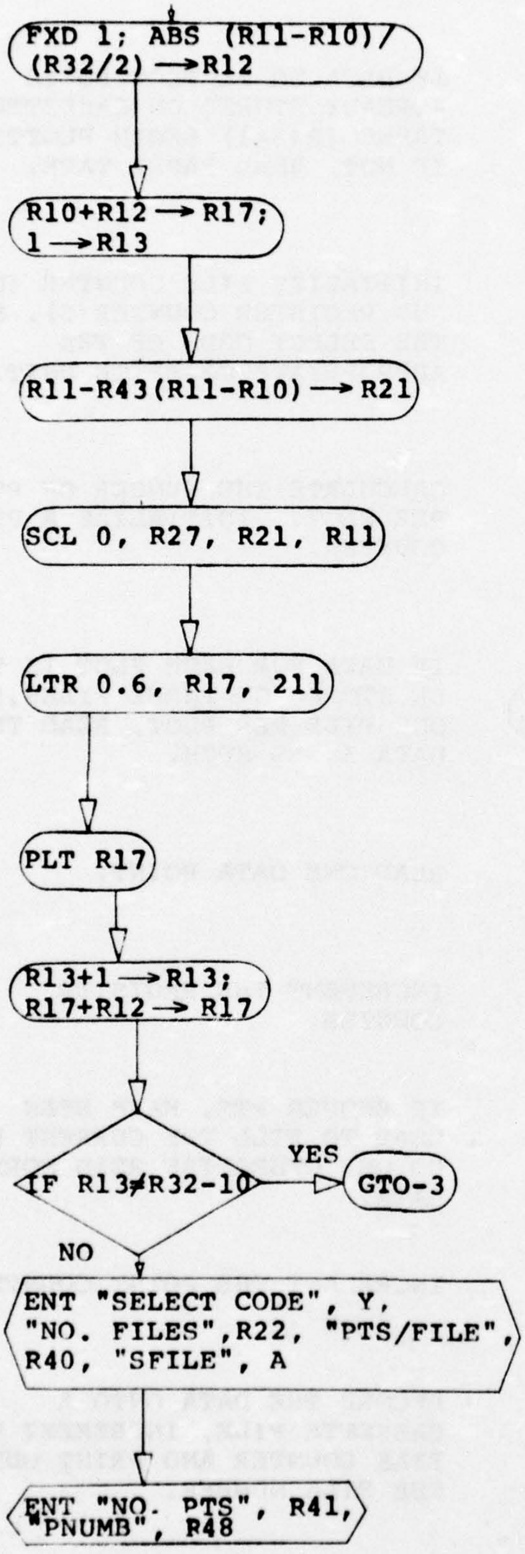
SCALE PLOTTER FOR LABELING AZIMUTH ON THE ORDINATE.

SPECIFY THE POSITION OF EACH LABEL AT THE PROPER TICK MARK.

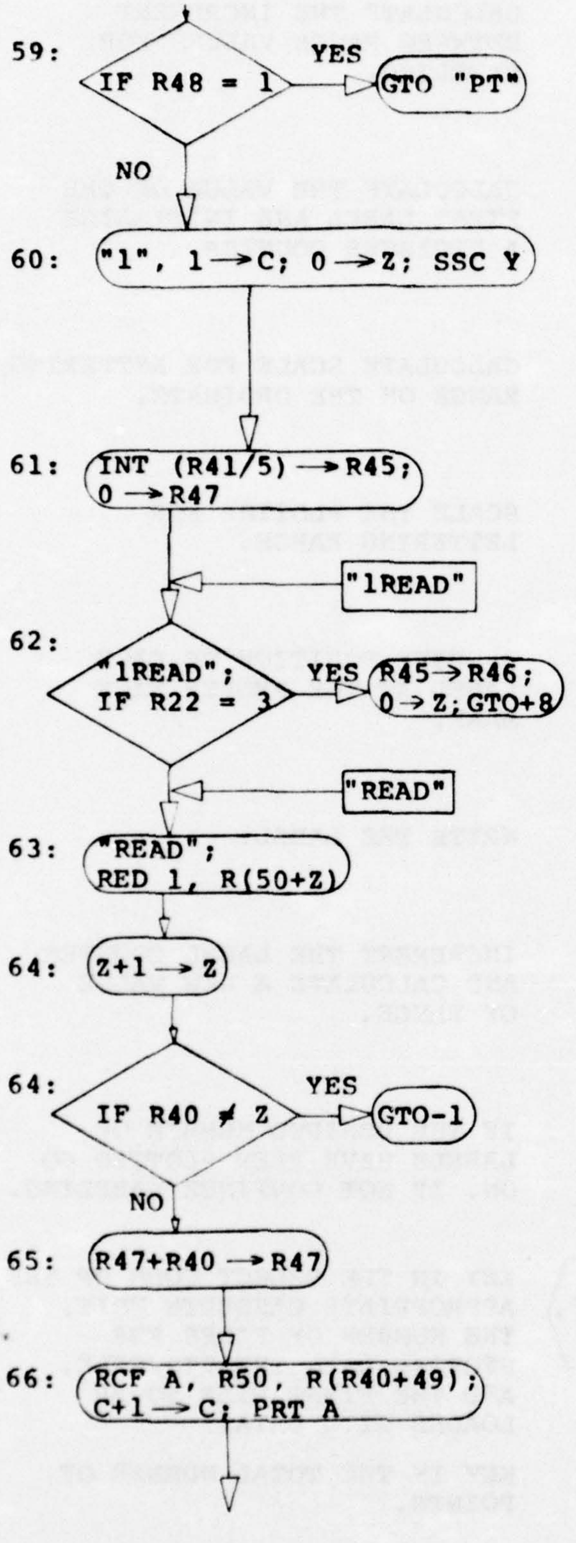
WRITE THE LABEL.

INCREMENT THE LABEL COUNTER AND CALCULATE A NEW VALUE OF AZIMUTH.

IF THE DESIRED NUMBER OF LABELS HAVE BEEN PLOTTED GO ON, IF NOT CONTINUE LABELING.



50:	FXD 1; ABS (R11-R10)/ (R32/2) →R12	CALCULATE THE INCREMENT BETWEEN RANGE VALUES FOR LABELING.
51:	R10+R12 → R17; 1 →R13	CALCULATE THE VALUE OF THE FIRST LABEL AND INITIALIZE A REGISTER COUNTER.
52:	R11-R43(R11-R10) → R21	CALCULATE SCALE FOR LETTERING RANGE ON THE ORDINATE.
53:	SCL 0, R27, R21, R11	SCALE THE PLOTTER FOR LETTERING RANGE.
54:	LTR 0.6, R17, 211	SPECIFY POSITION OF EACH LABEL AT THE PROPER TICK MARK.
55:	PLT R17	WRITE THE LABEL.
56:	R13+1 →R13; R17+R12 → R17	INCREMENT THE LABEL COUNTER AND CALCULATE A NEW VALUE OF RANGE.
57·	IF R13≠R32-10 YES → GTO-3 NO	IF THE DESIRED NUMBER OF LABELS HAVE BEEN PLOTTED GO ON, IF NOT CONTINUE LABELING.
58:	ENT "SELECT CODE", Y, "NO. FILES",R22, "PTS/FILE", R40, "SFILE", A	KEY IN THE SELECT CODE OF THE APPROPRIATE CASSETTE UNIT, THE NUMBER OF FILES FOR STORING DATA,THE PTS/FILE, AND THE FIRST FILE TO BE LOADED WITH DATA.
59:	ENT "NO. PTS", R41, "PNUMB", R48	KEY IN THE TOTAL NUMBER OF POINTS.



IF DATA TO BE PLOTTED IS ALREADY STORED ON CASSETTE TAPES (R48=1) BEGIN PLOTTING. IF NOT, READ PAPER TAPE.

INITIALIZE FILE COUNTER (C) AND REGISTER COUNTER(Z). SET THE SELECT CODE OF THE APPROPRIATE CASSETTE UNIT.

CALCULATE THE NUMBER OF PTS PER PLOT. INITIALIZE A PT COUNTER.

IF DATA FOR EACH PLOT IS TO BE STORED ON THREE FILES, i.e. ONE FILE PER PLOT, READ THE DATA IN AS SUCH.

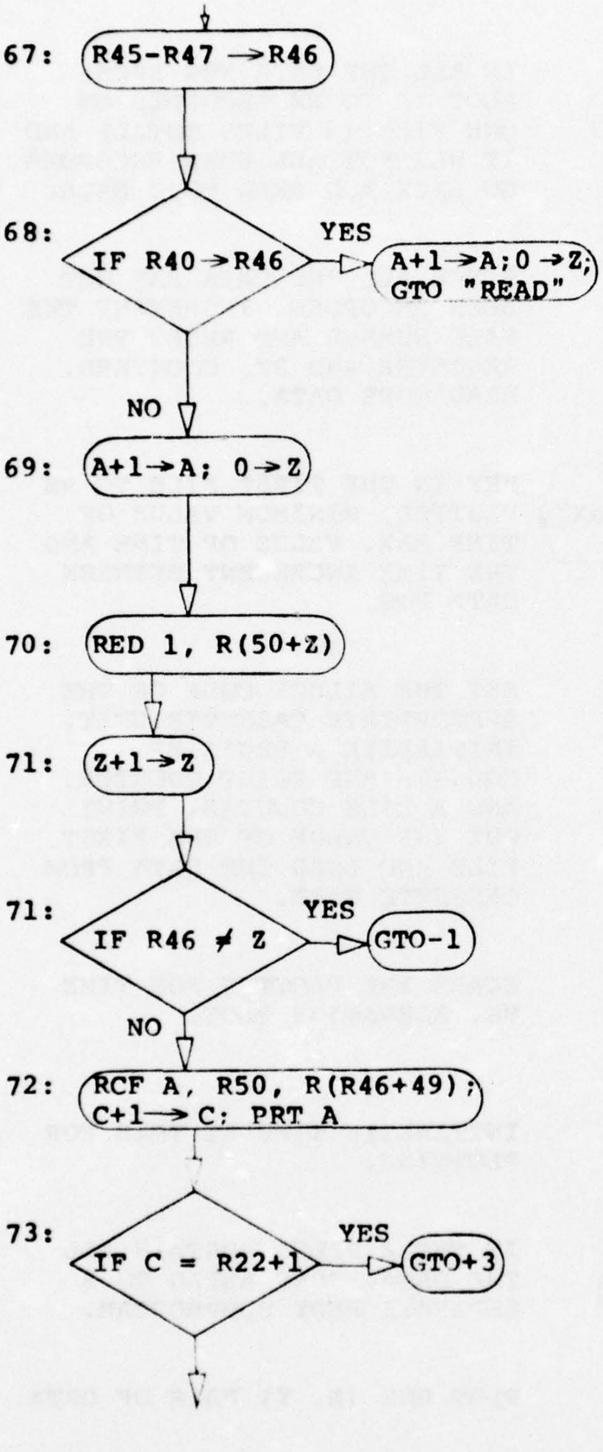
READ ONE DATA POINT.

INCREMENT THE REGISTER COUNTER.

IF ENOUGH PTS. HAVE BEEN READ TO FILL THE CURRENT FILE GO ON, OTHERWISE READ MORE DATA.

INCREMENT THE POINT COUNTER.

RECORD THE DATA ONTO A CASSETTE FILE, INCREMENT THE FILE COUNTER AND PRINT OUT THE FILE NUMBER.



CALCULATE THE NUMBER OF PTS. LEFT TO BE READ FOR ANY GIVEN PLOT.

IF THE NUMBER OF PTS LEFT TO BE RECORDED ARE MORE THAN CAN BE RECORDED ONTO ONE FILE, INCREMENT THE FILE NUMBER, GO BACK AND READ MORE DATA.

INCREMENT THE FILE NUMBER AND RESET THE REGISTER COUNTER.

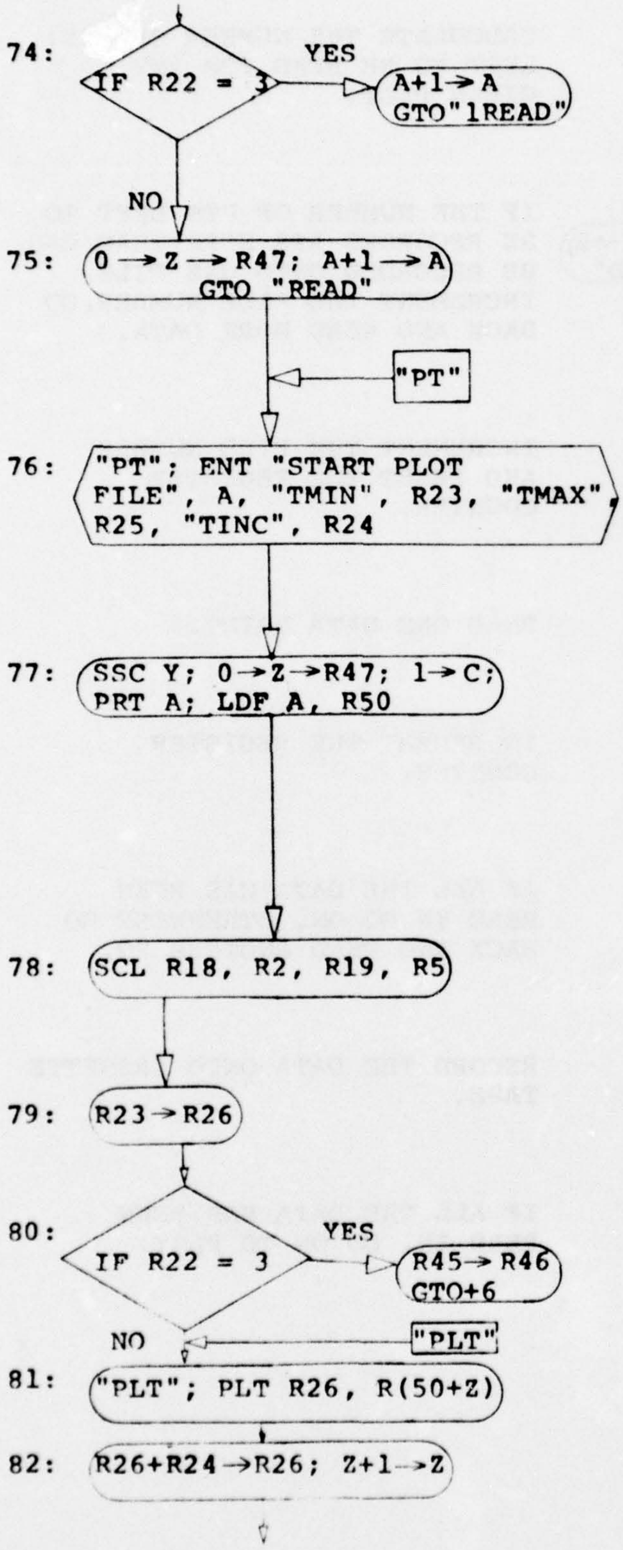
READ ONE DATA POINT.

INCREMENT THE REGISTER COUNTER.

IF ALL THE DATA HAS BEEN READ IN GO ON, OTHERWISE GO BACK AND READ ANOTHER PT.

RECORD THE DATA ONTO CASSETTE TAPE.

IF ALL THE DATA HAS BEEN READ IN, GO ON TO PLOT.



IF ALL THE DATA FOR EACH PLOT IS TO BE RECORDED ON ONE FILE (3 FILES TOTAL) AND IT HAS NOT ALL BEEN RECORDED GO BACK AND READ MORE DATA.

SINCE ALL THE DATA HAS NOT BEEN RECORDED. INCREMENT THE FILE NUMBER AND RESET THE REGISTER AND PT. COUNTERS. READ MORE DATA.

KEY IN THE FIRST FILE TO BE PLOTTED, MINIMUM VALUE OF TIME MAX. VALUE OF TIME AND THE TIME INCREMENT BETWEEN DATA PTS.

SET THE SELECT CODE OF THE APPROPRIATE CASSETTE UNIT, INITIALIZE A REGISTER COUNTER AND POINT COUNTER, AND A FILE COUNTER. PRINT OUT THE VALUE OF THE FIRST FILE AND LOAD THE DATA FROM CASSETTE TAPE.

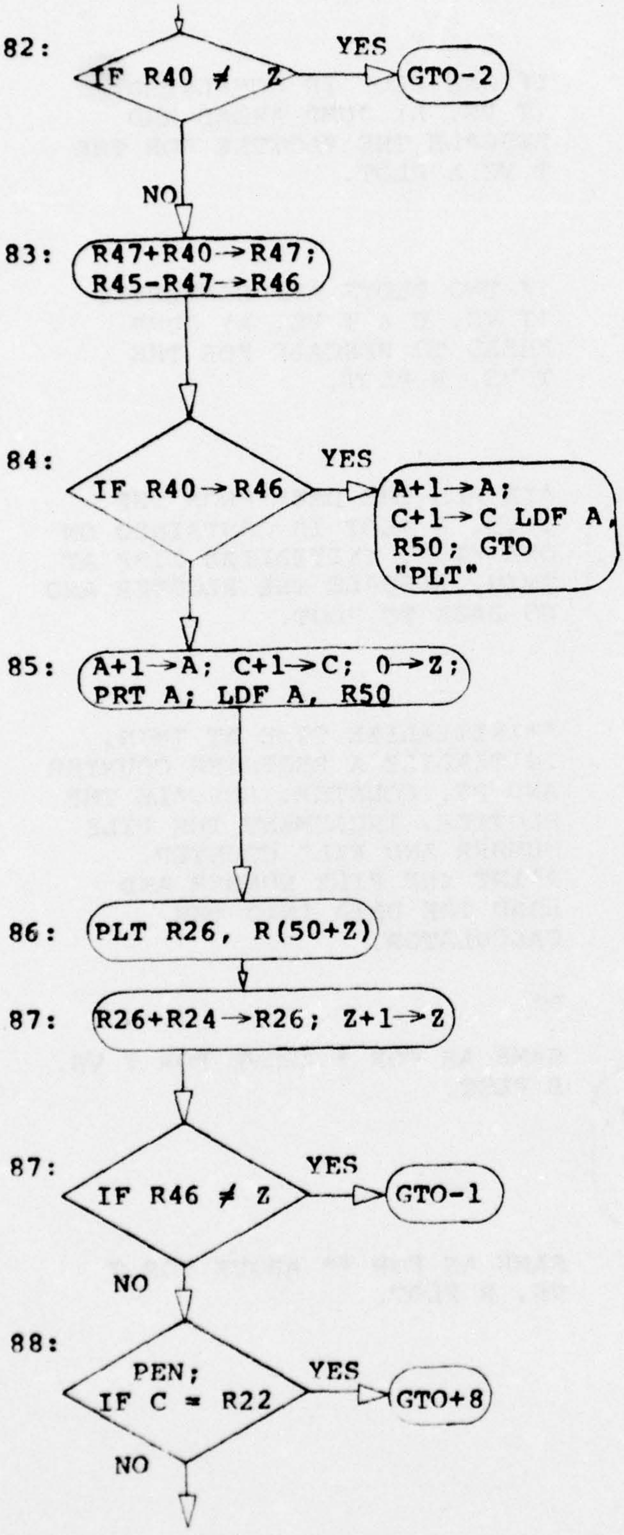
SCALE THE PLOTTER FOR TIME VS. ELEVATION PLOT.

INITIALIZE TIME AT TMIN FOR PLOTTING.

IF THREE FILES CONTAIN ALL THE DATA, JUMP AHEAD TO A SEPARATE PLOT SUBPROGRAM.

PLOT ONE (X, Y) PAIR OF DATA.

INCREMENT THE VALUE OF THE INDEPENDANT VARIABLE (TIME) AND THE REGISTER COUNTER.



IF ALL THE POINTS FROM ONE FILE HAVE BEEN PLOTTED GO ON. OTHERWISE GO BACK AND CONTINUE PLOTTING.

INCREMENT THE POINT COUNTER AND CALCULATE THE NUMBER OF PTS. LEFT TO BE PLOTTED.

IF MORE THAN ONE FILE OF DATA IS LEFT TO BE PLOTTED GO BACK AND PLOT THE NEXT FILE. OTHERWISE GO ON.

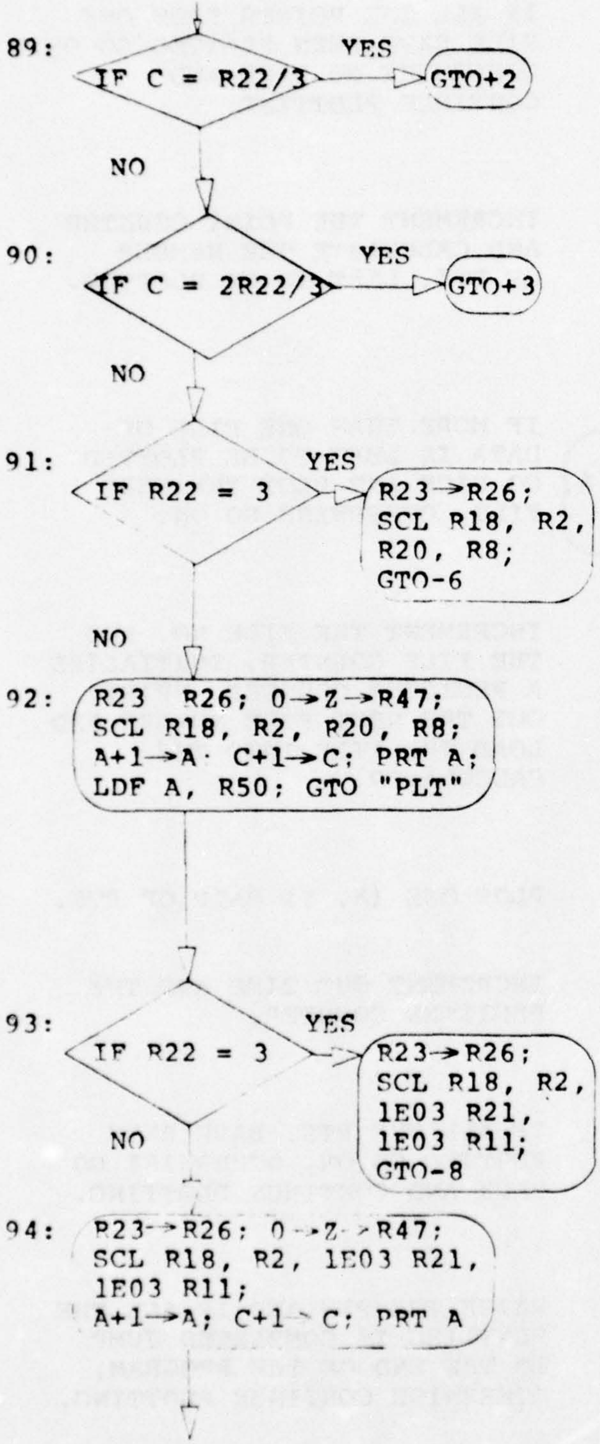
INCREMENT THE FILE NO. AND THE FILE COUNTER. INITIALIZE A REGISTER COUNTER. PRINT OUT THE NEXT FILE NUMBER AND LOAD THE FILE INTO THE CALCULATOR.

PLOT ONE (X, Y) PAIR OF PTS.

INCREMENT THE TIME AND THE REGISTER COUNTER.

IF ALL THE PTS. HAVE BEEN PLOTTED GO ON, OTHERWISE GO BACK AND CONTINUE PLOTTING.

RAISE THE PEN AND IF ALL THE PLOTTING IS COMPLETED JUMP TO THE END OF THE PROGRAM, OTHERWISE CONTINUE PLOTTING.



IF ONE PLOT IS COMPLETED (T VS. E) JUMP AHEAD AND RESCALE THE PLOTTER FOR THE T VS A PLOT.

IF TWO PLOTS ARE COMPLETED (T VS. E & T VS. A) JUMP AHEAD TO RESCALE FOR THE T VS. R PLOT.

*IF ALL THE DATA FROM THE T VS. A PLOT IS CONTAINED ON ONE FILE, INITIALIZE TIME AT TMIN, RESCALE THE PLOTTER AND GO BACK TO PLOT.

**INITIALIZE TIME AT TMIN, INITIALIZE A REGISTER COUNTER AND PT. COUNTER. RESCALE THE PLOTTER. INCREMENT THE FILE NUMBER AND FILE COUNTER. PRINT THE FILE NUMBER AND LOAD THE DATA INTO THE CALCULATOR.

SAME AS FOR * ABOVE FOR T VS. R PLOT.

SAME AS FOR ** ABOVE FOR T VS. R PLOT.

95: LDF A, R50: GTO "PLT"	LOAD THE LAST FILE AND GO BACK TO PLOT IT.
96: PRT "END OF DATA"	MESSAGE TO THE OPERATOR THAT ALL THE DATA HAS BEEN PLOTTED.
97: END	MESSAGE TO THE CALCULATOR THAT THE PROGRAM IS TERMINATED.

Sample plots are shown in Figures 11 through 13. Figures 11 and 12 are plots of elevation, azimuth, and slant range-vs-time. Figure 13 is a plot of elevation rate, azimuth rate, and slant range rate-vs-time.

Listed in Appendix F is a version of the RAE Program in which the plot parameters and input data are input from cards. The program operates on the same principle as the generalized plot program described in Section V of this report so that the flow chart and register cross-reference chart can be utilized for this program also. The range is plotted first, then azimuth and elevation-vs-time; similarly, for the rates.

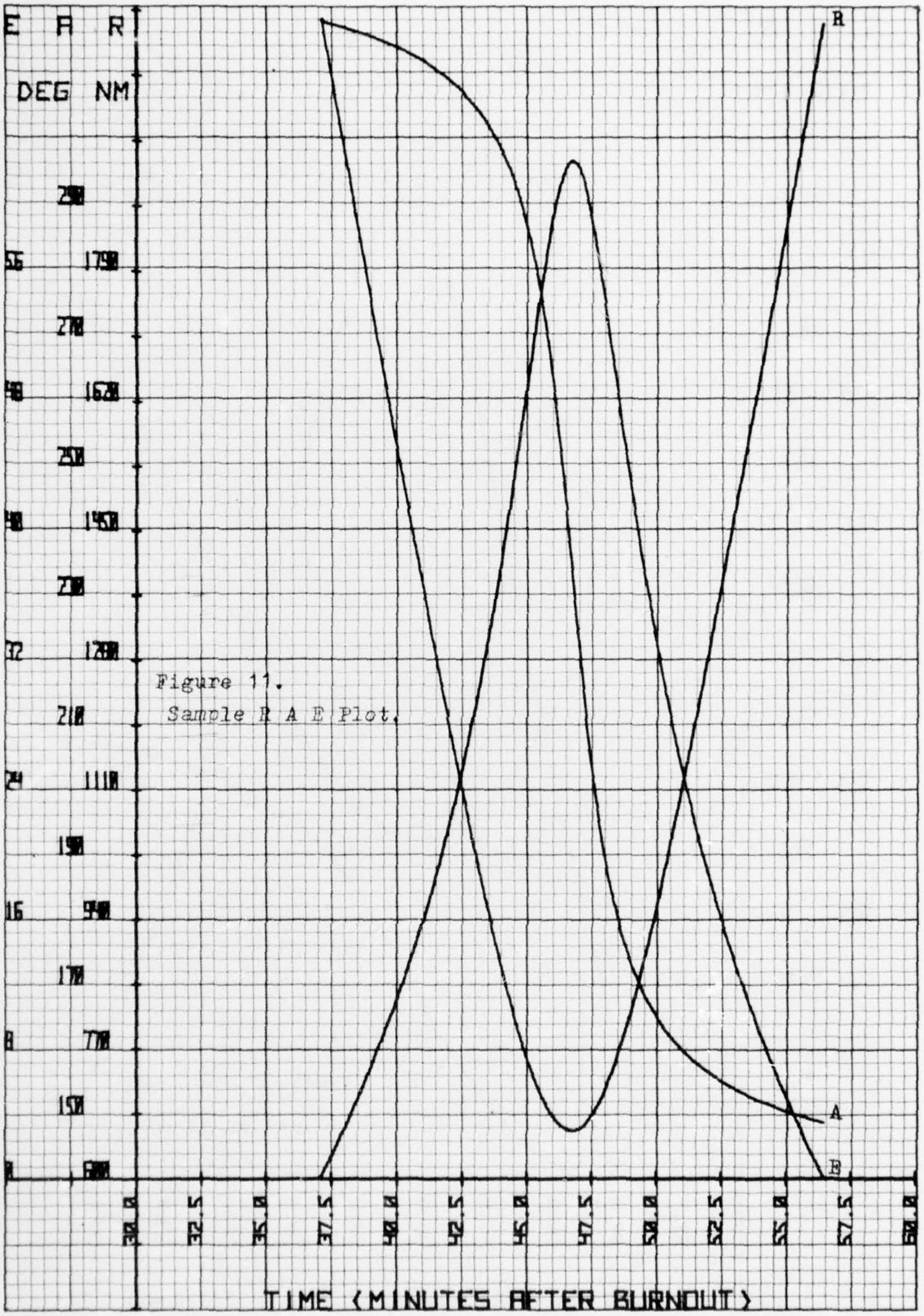
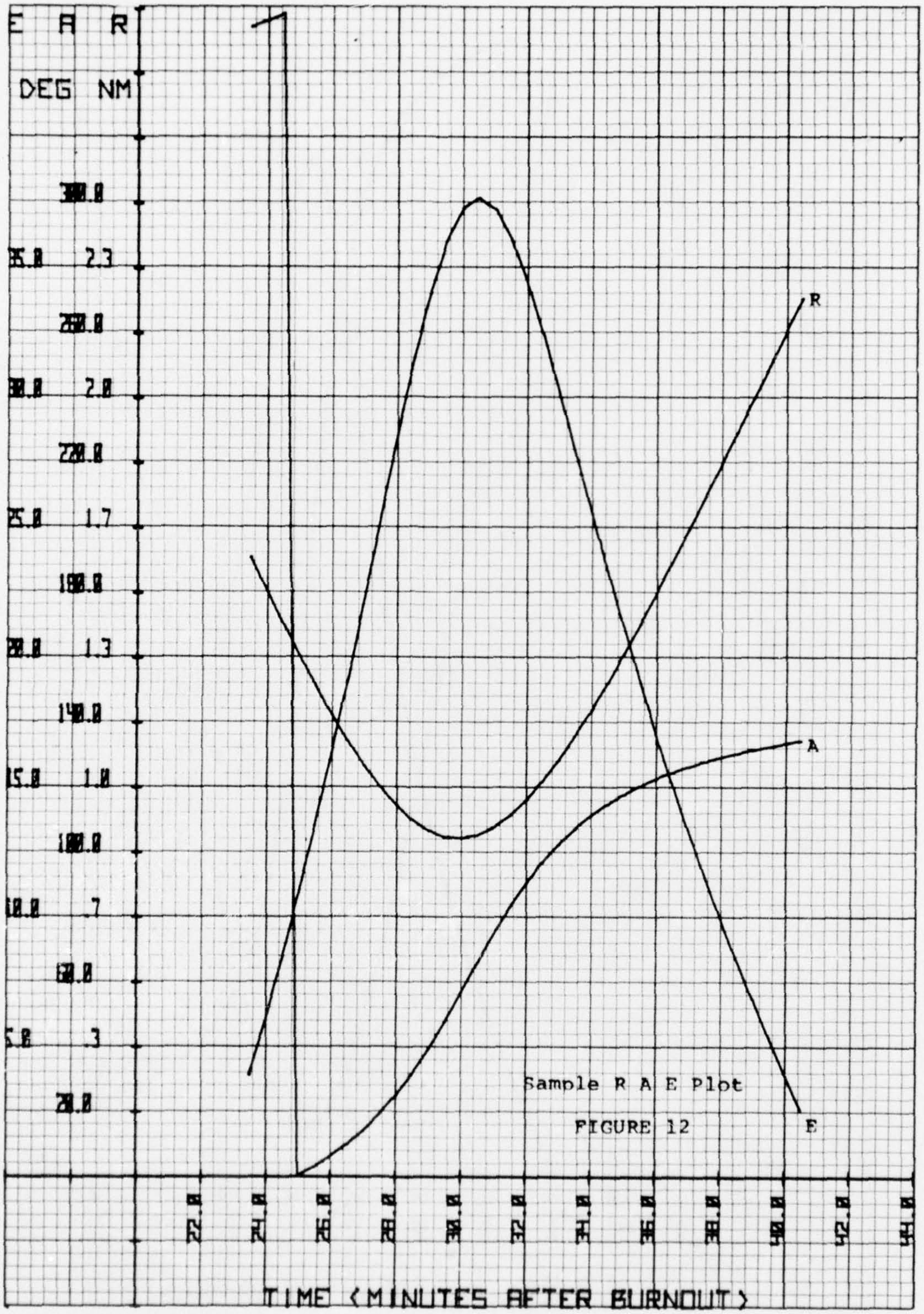


Figure 11.
Sample R A E Plot.



Sample R A E Plot

FIGURE 12

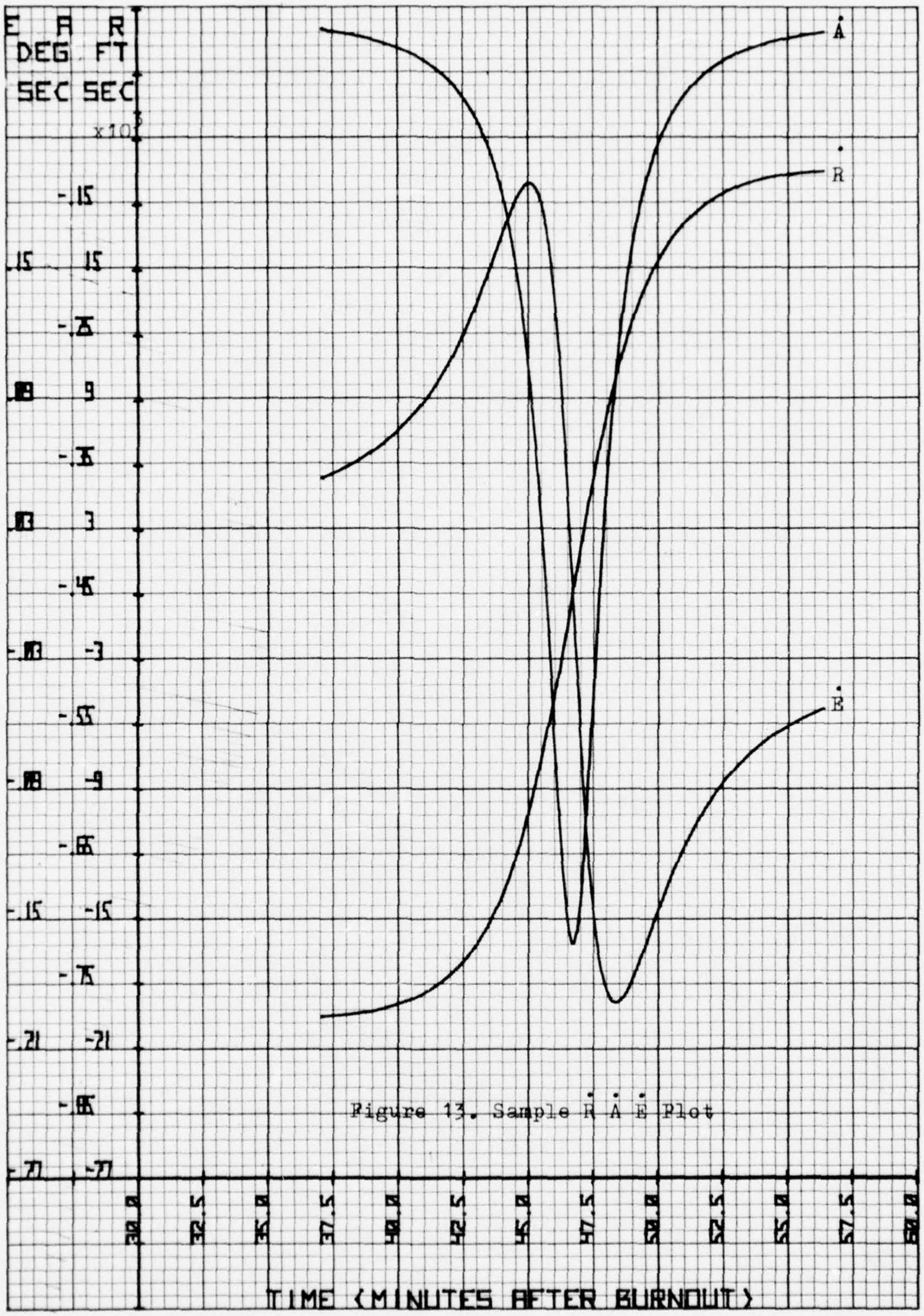


Figure 13. Sample $\dot{R}$ $\dot{A}$ $\dot{E}$ Plot

APPENDICES

The programs described in the following appendices are given to show the developmental progression leading to the general programs described in the main report. These are written for specific applications and contain definite limitations in their present form. The basic logic is the same; however, (i.e., read in the data, record it on cassette tapes, and plot) for all the programs in this report, and it may be easier to look at these in an effort to learn the general programs. Also, should it be necessary to make major modifications to the main program, or rewrite portions of it, these simplified versions may be an excellent starting point for further development.

APPENDIX A: Angle vs Intensity Plot Program

The program shown on the following page will plot the intensity of a transmit beam vs angle.

Statements 1-17 are used to label the plot and to draw and label the axes. The size and lettering can be changed by changing the LTR commands in Statements 2, 3, 4 and 5.

Statement 6 reads the first six data on paper tape and sets them equal to the Min. X value, Max. X value, X increment, Min. Y value, Max. Y value, and a number of points, respectively. If the paper tape does not contain these data, the RED command may be changed to an ENT command (similar to Statement 18) and these data may be keyed by the operator from the keyboard. The program replacing the RED command in Statement 6 with an ENT command is shown in Figure A-2. Statements 7-17 are set up to draw the axes and label the tick marks as shown in Figure A-3. The number on the axes depends on the range of values but the number of values (i.e., 18 for X and 12 for Y) is constant for this size graph paper. The proper statements can be readily modified to accommodate other sizes without changing the type of computations required or formulas used.

The part of the program which reads the paper tape and plots it from stored data consists of Statements 18 through 33. The data is read in and stored in blocks according to the following formula: (Number of points/number of files). Care must be taken when an odd number of pairs of points are to be stored. The number of points/number of files must be such that an equal number of points are recorded per file and each (x,y) pair of data points must remain intact, i.e., the x value cannot be recorded on one file and the y value on another.

The following example should help to clarify:

Suppose some data was punched on paper tape where the pairs of data points ranged from 0 to 70. This is 71 pairs of points and there is no way of dividing 71 pairs evenly among any number of files (other than 1 or 71) without splitting a data pair. Since the plotter must have an (x,y) pair to be able to plot, the pairs of points are important rather than the absolute number of points (142 in this example).

There are three possible ways of avoiding this problem depending on the data. The first method is, as implemented in the angle vs intensity program #3, where the independent variable time, is initialized at some time, tmin, and incremented by some constant Δt until a time, tmax, is reached. See Figure A-3. The data labeled "intensities at various angles (dB)" are 46 values of intensity one value for each angle from 0° to 45°: at 0°, intensity = 21.07 dB at 1°, intensity = 0.38 dB, etc. Therefore, only intensity need be recorded and the angle may be programmed.

The second method occurs when the number of points is relatively small and all the points can be stored on one file.

The two methods described above are relatively specific and require definite limitations on the data. The third method is more general and will work for all cases. If there are an odd number of data pairs, simply key in an extra data pair and store it as any other point. The keyed-in point would either be a duplicate of the first data point punched on paper tape or a duplicate of the last data point punched on paper tape. This extra point can be added to the paper tape when it is punched or it can be programmed. In either case, an even number of data pairs will result.

Table A-1 lists the registers used by the program and the value stored in each. This table also includes the inputs required and these are indicated by the word "input" enclosed in parenthesis after the explanation of the value stored.

Figure A-3 is a sample teletypewriter printout of the data punched on paper tape and Figure A-4 is a sample printout of the data printed as it was being plotted.

Figure A-5 is the plot of Angle vs Intensity.

Register Table

Register	Value Stored
R0	Min. angle (input)
R1	Max. angle (input)
R2	No. of files (input)
R3	Increment between angle labels on abscissa
R4	Value of angle label
R5	Min. x for scaling plotter
R6	Min. value of intensity (input)
R7	Max. value of intensity (input)
R8	Increment between intensity labels on ordinate
R9	Value of intensity label
R10	Min. Y for scaling plotter
R11	No. of labels on abscissa
R12	No. of labels on ordinate
R50	Label counter
R51	No. of points to be plotted (input)
R100	Data Storage
A	File no. of data file (begins with File 1)
B	Register counter
Y	Select code of cassette unit (input)
Z	Register counter

TABLE A-1

Register Table With Inputs for Angle vs Intensity Program

```
0:  FXD 5 [

1:  ENT "XLENGTH",R15,"YLENGTH",R16 [

2:  ENT "ABSCISSA USED",R17,"ORDINATE USED",R18 [

3:  SCL 0,R15,0,R16;AXE 1,1,.5,.5 [

4:  R15/R17 TO R42;R16/R18 TO R43 [

5:  2R17 TO R11;2R18 TO R12 [

6:  RED 1,R7,R6,R0,R13,R41 [

7:  (R41-1)R13+R0 TO R1;PRT R0,R1,R6,R7,R41 [

8:  ENT "XMIN",R0,"XMAX",R1,"YMIN",R6,"YMAX",R7 [

9:  ABS (R1-R0)/R11 TO R3;R0+R3 TO R4;1 TO R40 [

10: R1-R42(R1-R0) TO R5 [

11: SCL R5,R1,0,R16 [

12: LTR R4,.2,212;PLT R4 [

13: 1+R40 TO R40;IF R40 # R11+1;R4+R3 TO R4;GTO -1 [

14: ABS (R7-R6)/R12 TO R8;R6+R8 TO R9;1 TO R40 [

15: R7-R43(R7-R6) TO R10 [

16: SCL 0,R15,R10,R7 [

17: FXD 3;LTR .2,R9-.5,211;PLT R9 [

18: IF R40 <= R12;1+R40 TO R40;R9+R8 TO R9;GTO -1 [

19: ENT "SELECT CODE",Y,"NO. FILES",R2 [

20: 1 TO A [

21: 0 TO B [

22: RED 1,R(50+B);B+1 TO B;IF R41/R2 # B;JMP 0 [

23: SSC Y [

24: RCF A,R50,R(R41/R2+49);PRT A [
```

Figure A-1. Angle vs Intensity Plot Program

```
25:  A+1 TO A [
26:  IF A # R2+1;GTO -5 [
27:  1 TO A;LDF A,R50;PRT "FILE A LOADED" [
28:  0 TO Z [
29:  SCL R5,R1,R10,R7 [
30:  PLT R0,R(50+Z) [
31:  IF R41/R2-1 # Z;R0+R13 TO R0;Z+1 TO Z;GTO -1 [
32:  A+1 TO A;IF A <= R2;R13+R0 TO R0;0 TO Z;LDF A,R50;GTO -2 [
33:  PRT "END OF DATA" [
34:  END  [
```

Figure A-1. Angle vs Intensity Plot Program (Cont'd)

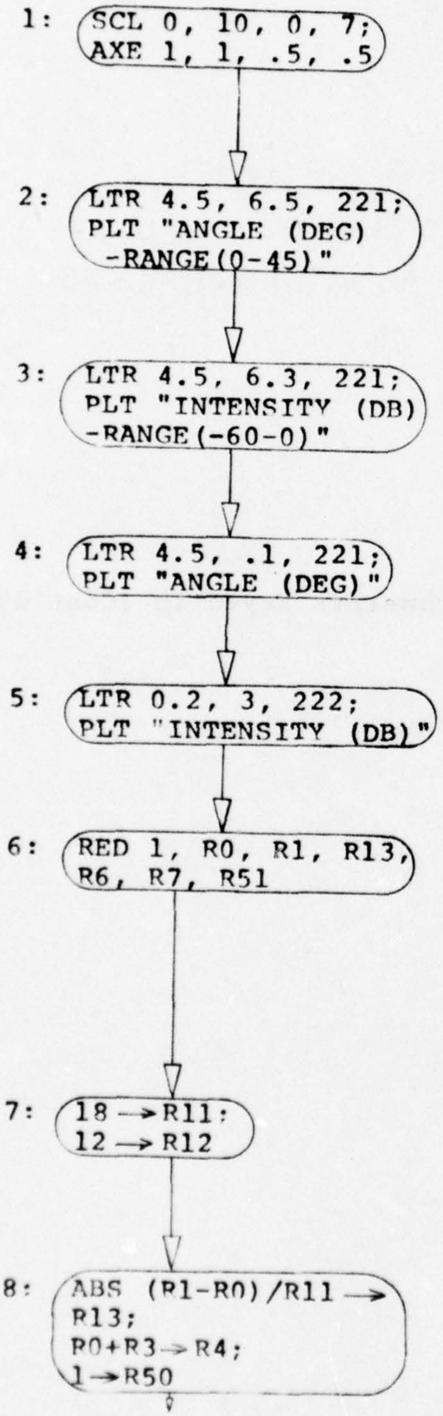
```
0:  FXD 2 [
1:  SCL 0,10,0,7;AXE 1,1,.5,.5 [
2:  LTR 4.5,6.5,221;PLT "ANGLE (DEG)-RANGE(0-45)" [
3:  LTR 4.5,6.3,221;PLT "INTENSITY(DEG)-RANGE(-60-0)" [
4:  LTR 4.5,.1,221;PLT "ANGLE (DEG)" [
5:  LTR 0.2,3,222;PLT "INTENSITY(DB)" [
6:  ENT "XMIN",R0,"XMAX",R1,"NO. LABELS",R11 [
7:  ABS (R1-R0)/R11 TO R3;R0+R3 TO R4;1 TO R50 [
8:  R1-(10/9)(R1-R0) TO R5 [
9:  SCL R5,R1,0,7 [
10: LTR R4,.6,212;PLT R4 [
11: 1+R50 TO R50;IF R50 # 19;R4+R3 TO R4;GTO -1 [
12: ENT "YMIN",R6,"YMAX",R7,"NO. LABELS",R12 [
13: ABS (R7-R6)/R12 TO R8;R6+R8 TO R9;1 TO R50 [
14: R7-(7/6)(R7-R6) TO R10 [
15: SCL 0,10,R10,R7 [
16: LTR .3,R9,211;PLT R9 [
17: IF R50 <= R12;1+R50 TO R50;R8+R9 TO R9;GTO -1 [
18: ENT "SELECT CODE",Y,"NO. FILES",R2 [
19: 1 TO A [
20: 100 TO B [
21: RED 1,RD;B+1 TO B;IF R51/R2+100 # B;JMP 0 [
22: SSC Y [
23: RCF A,R100,R(51/R2+99);PRT A [
24: A+1 TO A [
```

Figure A-2. Angle vs Intensity - Plot Parameters Keyed in

```
25:  IF A # R2+1;GTO 20 [

26:  1 TO A;LDF A,R100;PRT "FILE A LOADED" [

27:  0 TO Z [

28:  SCL R5,R1,R10,R7 [

29:  PLT R0,R(100+Z);PRT "R0=",R0,"R(100+Z)=",R(100+Z) [

30:  IF R51/R2-1 # Z;R0+R13 TO R0;Z+1 TO Z;GTO -1 [

31:  A+1 TO A;IF A # R2;1+R0 TO R0;0 TO Z;LDF A,R100;GTO -2 [

32:  PRT "END OF DATA" [

33:  END  [
```

Figure A-2. Angle vs Intensity - Plot Parameters Keyed in (Cont'd)

ANGLE VS INTENSITY



```
1: SCL 0, 10, 0, 7;
   AXE 1, 1, .5, .5
```

SCALE PLOTTER FOR ENTIRE PLOTTING SURFACE (8 1/2 x 11).

```
2: LTR 4.5, 6.5, 221;
   PLT "ANGLE (DEG)
   -RANGE(0-45)"
```

WRITE ANGLE (DEG) - RANGE (0-45) ON THE PLOT.

```
3: LTR 4.5, 6.3, 221;
   PLT "INTENSITY (DB)
   -RANGE(-60-0)"
```

WRITE INTENSITY (DB) - RANGE (-60-0) ON THE PLOT.

```
4: LTR 4.5, .1, 221;
   PLT "ANGLE (DEG)"
```

LABEL THE X AXIS.

```
5: LTR 0.2, 3, 222;
   PLT "INTENSITY (DB)"
```

LABEL THE Y AXIS

```
6: RED 1, R0, R1, R13,
   R6, R7, R51
```

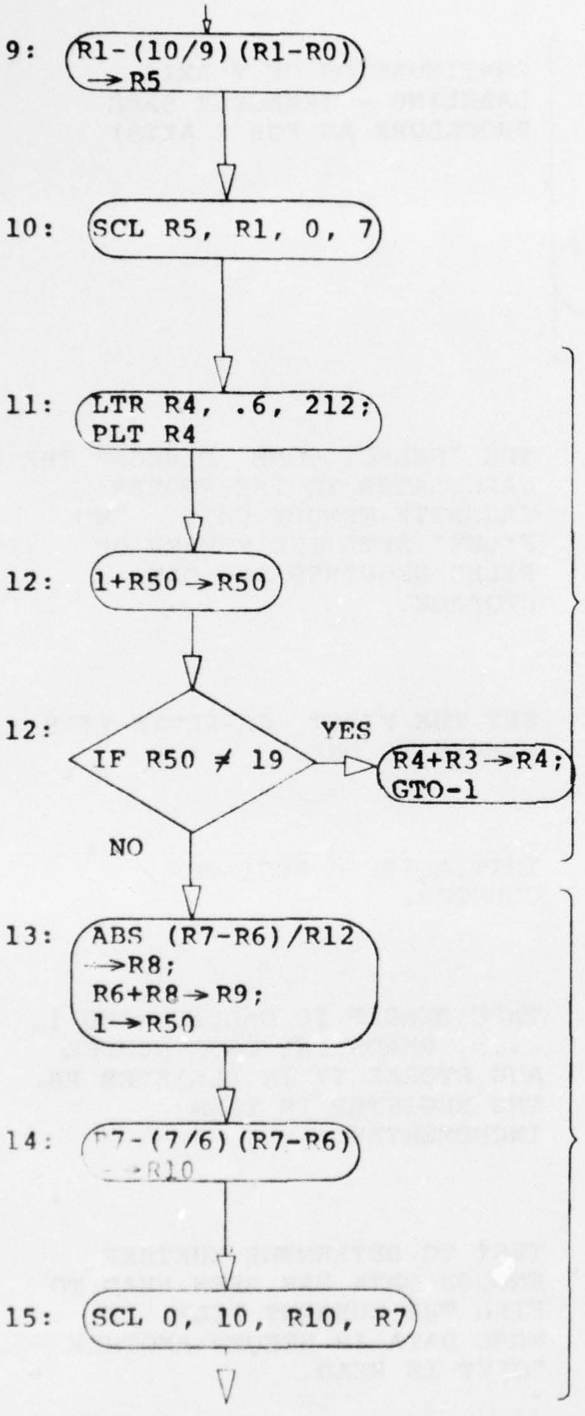
READ FIRST SIX DATA ELEMENTS ON PAPER TAPE: XMIN, XMAX, X INCREMENT, YMIN, YMAX, AND NUMBER OF POINTS RESPECTIVELY. SEE FIG.

```
7: 18 → R11;
   12 → R12
```

DEFINE THE NUMBER OF LABELS ON X AND Y AXES RESPECTIVELY.

```
8: ABS (R1-R0)/R11 →
   R13;
   R0+R3 → R4;
   1 → R50
```

CALCULATE INCREMENT BETWEEN THE LABELS ON THE X-AXIS AND THE VALUE OF THE FIRST LABEL. INITIALIZE A LABEL COUNTER.

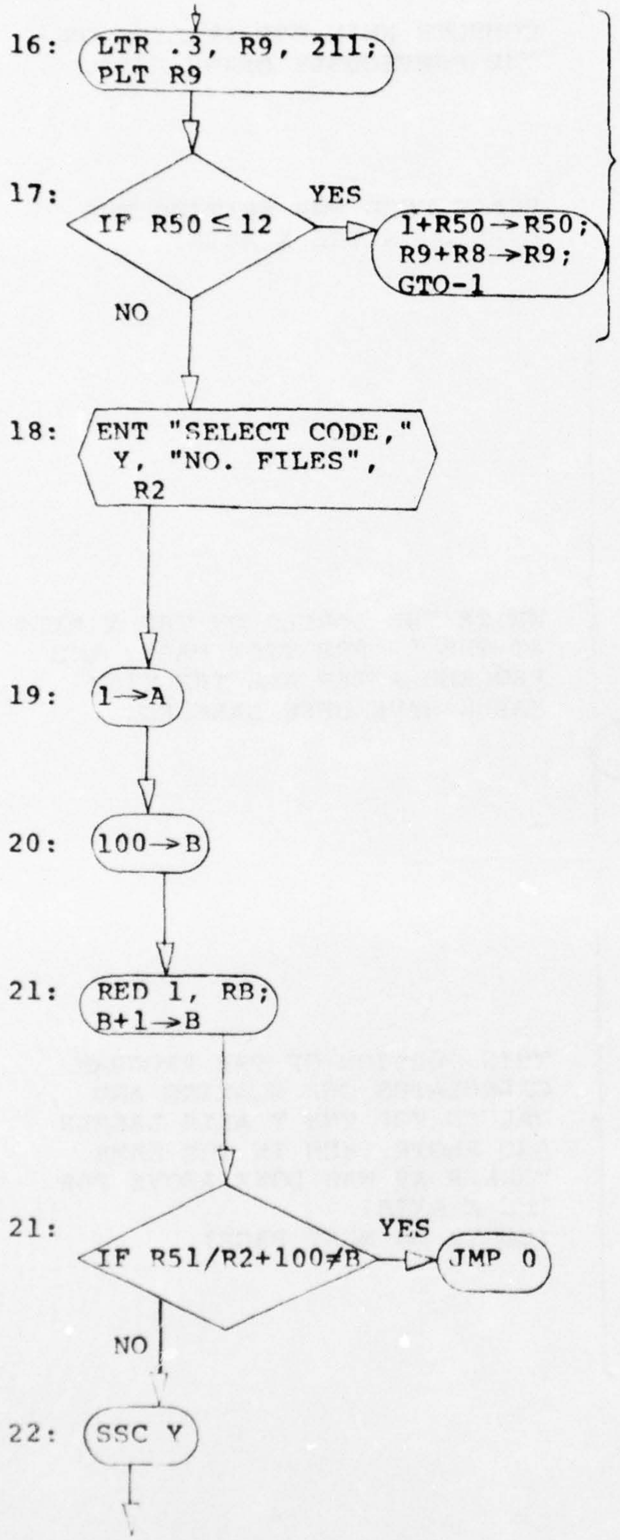


COMPUTE XMIN FOR SEALING ON THE PREVIOUSLY DRAWN AXES.

SCALE USED FOR WRITING THE LABELS ON THE X AXIS.

WRITE THE LABELS ON THE X AXIS AT THE PROPER TICK MARK, AND PROCEED AFTER ALL THE TICK MARKS HAVE BEEN LABELED.

THIS PORTION OF THE PROGRAM CALCULATES THE SCALING AND VALUES FOR THE Y AXIS LABELS AND PLOTS THEM IN THE SAME MANNER AS WAS DONE ABOVE FOR THE X AXIS.
(CONT. ON NEXT PAGE)



CONTINUATION OF Y AXIS LABELING - (EXACTLY SAME PROCEDURE AS FOR X AXIS)

THE "SELECT CODE" DIRECTS THE CALCULATOR TO THE PROPER CASSETTE MEMORY UNIT. "NO. FILES" SETS THE NUMBER OF FILES REQUIRED FOR DATA STORAGE.

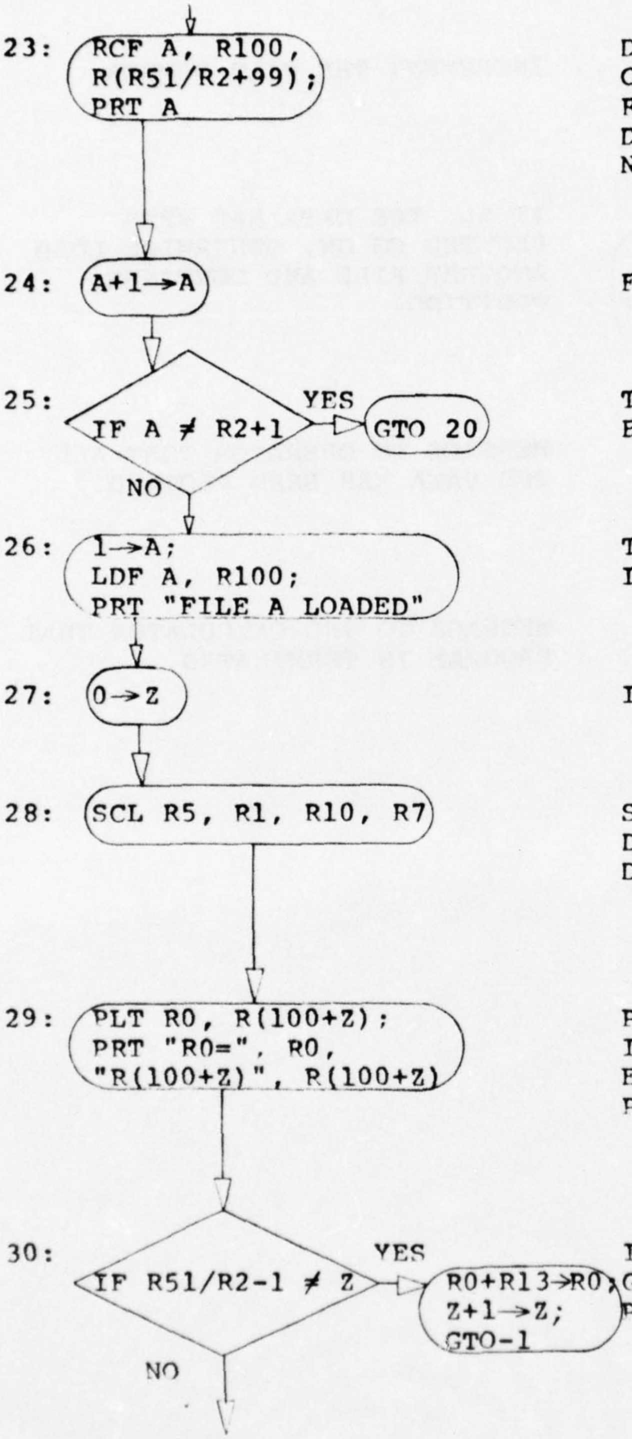
SET THE FIRST CASSETTE FILE NUMBER TO ONE.

INITIALIZE A REGISTER COUNTER.

TAPE READER IS CALLED (RED 1, ...), READS 1st DATA NUMBER AND STORES IT IN REGISTER RB. THE REGISTER IS THEN INCREMENTED.

TEST TO DETERMINE WHETHER ENOUGH DATA HAS BEEN READ TO FILL THE CURRENT FILE. IF MORE DATA IS NEEDED ANOTHER POINT IS READ.

WHEN ENOUGH DATA IS READ IN TO FILL A FILE, THE CASSETTE UNIT IS CALLED.



```
23: RCF A, R100,
    R(R51/R2+99);
    PRT A
24: A+1→A
25: IF A ≠ R2+1    YES  GTO 20
26: 1→A;
    LDF A, R100;
    PRT "FILE A LOADED"
27: 0→Z
28: SCL R5, R1, R10, R7
29: PLT R0, R(100+Z);
    PRT "R0=", R0,
    "R(100+Z)", R(100+Z)
30: IF R51/R2-1 ≠ Z    YES  R0+R13→R0;
                            Z+1→Z;
                            GTO-1
```

23: DATA IS RECORDED INTO THE CURRENT FILE BEGINNING WITH REGISTER R100 UNTIL ALL THE DATA IS RECORDED. THE FILE NUMBER PRINTED OUT.

24: FILE NUMBER IS INCREMENTED.

25: TEST WHETHER ALL THE DATA HAS BEEN READ IN AND RECORDED.

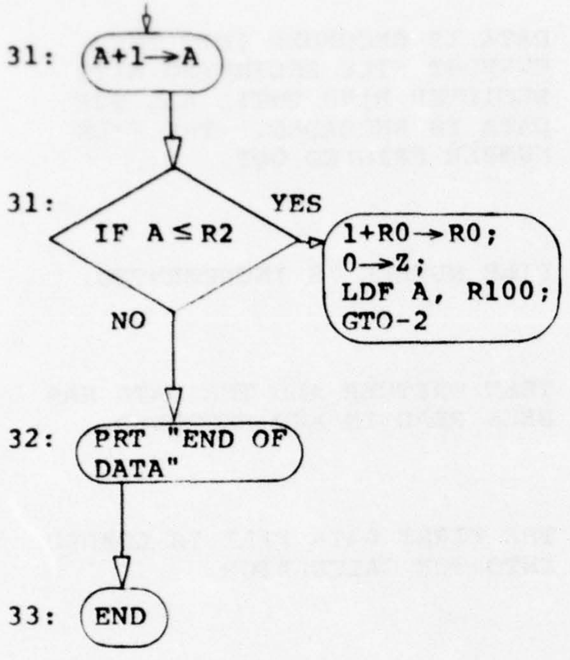
26: THE FIRST DATA FILE IS LOADED INTO THE CALCULATOR.

27: INITIALIZE A REGISTER COUNTER.

28: SCALE THE PLOTTING AREA DEFINED BY THE PREVIOUSLY DRAWN AXES.

29: PLOT ONE VALUE OF ANGLE VS INTENSITY AND PRINT OUT EACH VALUE AS IT IS BEING PLOTTED.

30: IF ONE FILE HAS BEEN PLOTTED GO ON, OTHERWISE GO BACK TO PLOT ANOTHER POINT.



INCREMENT THE FILE NUMBER.

IF ALL THE DATA HAS BEEN PLOTTED GO ON, OTHERWISE LOAD ANOTHER FILE AND CONTINUE PLOTTING.

MESSAGE TO OPERATOR THAT ALL THE DATA HAS BEEN PLOTTED.

MESSAGE TO THE CALCULATOR THAT PROGRAM IS TERMINATED.

Data As It Appears On Paper Tape

```
 Min.     Max.        Angle  Min.     Max.     No.
 Angle    Angle       Inc.   Inten.   Inten.  Points
 (deg)    (deg)       (deg)  (db)     (db)

   0.      45.00      1.00  -60.00     0.      46
-21.07   -0.38 -17.54 -28.00 -34.79 -35.47 -40.12 -30.06 -24.37
-42.60 -28.45 -25.03 -24.40 -25.26 -40.14 -18.49 -22.58 -26.09
-32.37 -37.71 -46.83 -35.12 -33.49 -34.43 -36.34 -36.40 -32.77
-21.07  -0.83 -16.50 -20.44 -26.04 -28.88 -37.33 -29.89 -29.86
-29.50 -28.16 -21.75 -32.03 -28.00 -25.11

       Intensities at various angles (db)
```

Figure A-3. Sample Hard Copy of Paper Tape Data

```
RO=
                  0.00
R(100+Z)=
                -21.07
RO=
                  1.00
R(100+Z)=
                -  .38
RO=
                  2.00
R(100+Z)=
                -17.54
RO=
                  3.00
R(100+Z)=
                -28.00
RO=
                  4.00
R(100+Z)=
                -34.79
```

Figure A-4. Sample Calculator Printout of Plotted Data

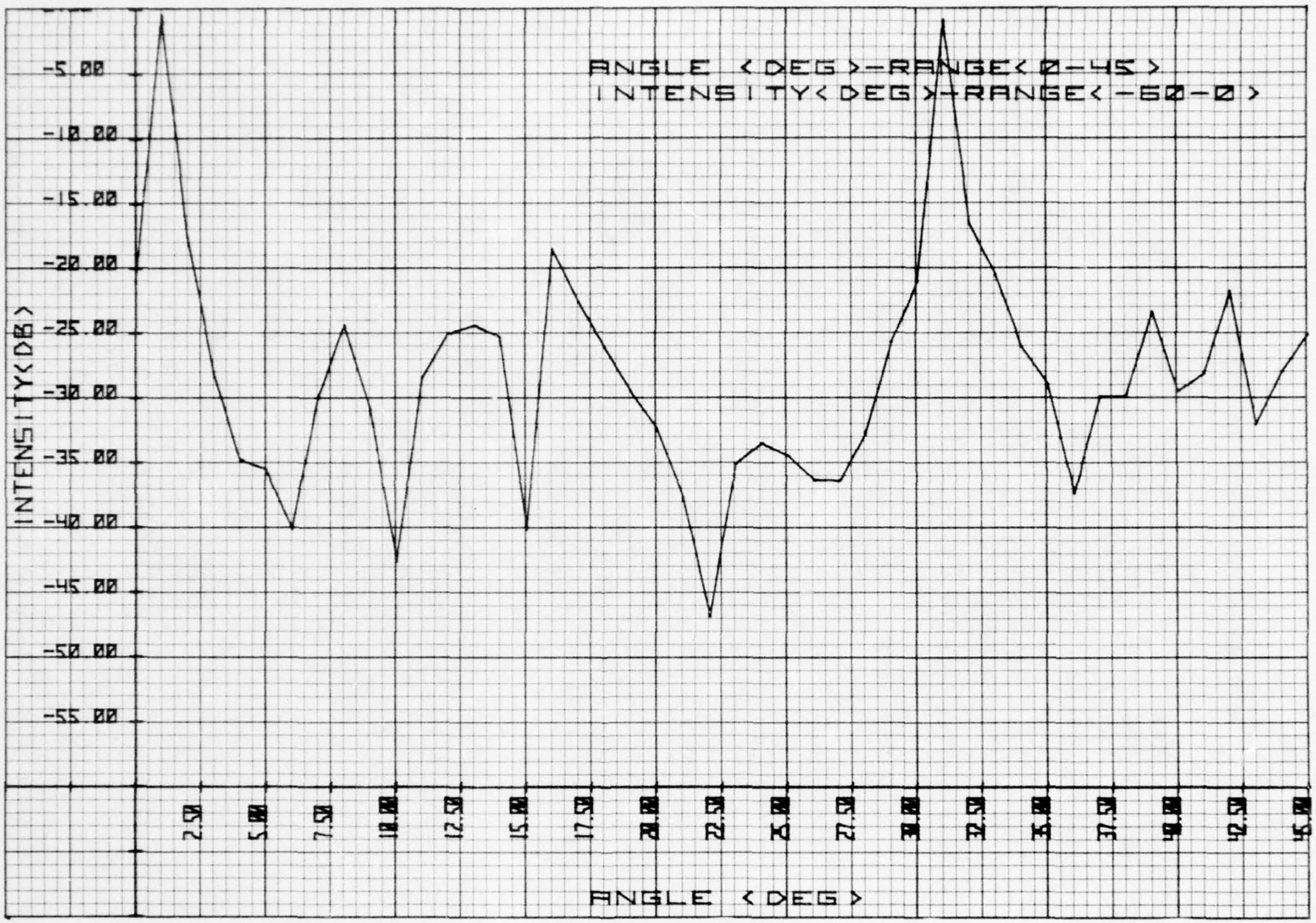


Figure A-5. Sample Angle vs Intensity Plot

APPENDIX B.: TRAJECTORY GROUND TRACE ON A MERCATOR PROJECTION

The next program plots the ground trace of a trajectory on a Mercator Projection. Each value of latitude undergoes a calculation which scales it to Mercator latitude. The calculation is of the form

$$Y= \ln[1/\{(\tan \frac{(90- \Phi_p)}{2})(\frac{1+ e\cos(90 - \Phi_p)}{1- e\cos(90 - \Phi_p)})^{e/2}\}]$$

Y is the Mercator latitude.

A listing of the program is shown in Figure B-1. Statements 4-9 perform the calculation for Mercator latitude.

Program Statements 11-14 record data onto cassette tapes and Statements 15-22 reload the data and plot it.

Table B-2 lists the registers used by the program, defines the value stored in each and which are inputs that are keyed-in by the operator.

Figure B-2 is a sample ground trace plotted on a Mercator Projection.

Register Table

Register	Value Stored
R0 & R1	Not used
R2	No. of data files (input)
R3	Intermediate values in calculation of the Mercator latitude
R4	
R51	No. of points (input)
R100 ...	Data storage
A	File Number of Data File
B	Register counter
Y	Select code of cassette unit (input)
Z	Register counter

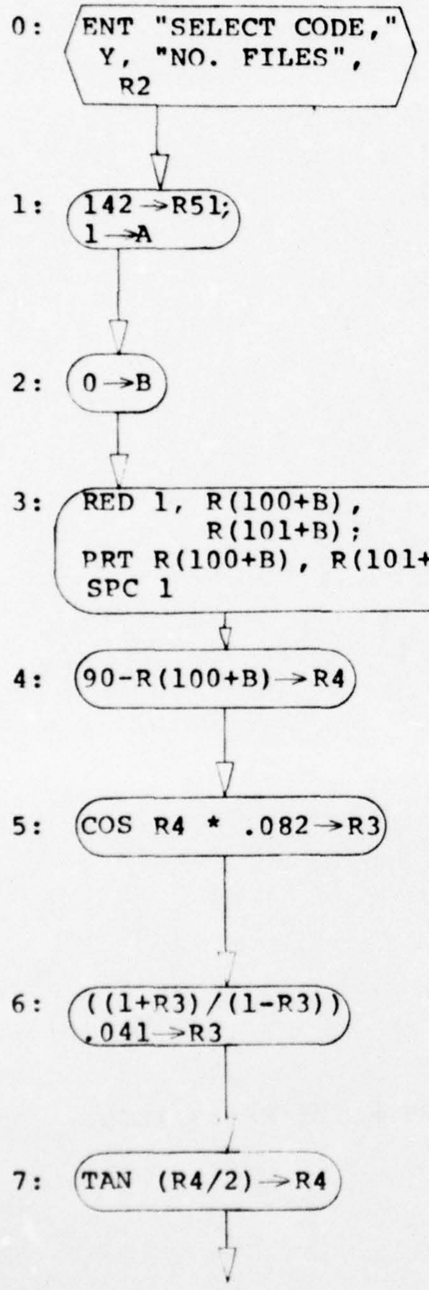
Table B-1

Register Table with Inputs for Trajectory Ground Trace Program

```
0:  ENT "SELECT CODE",Y,"NO. FILES",R2 [

1:  142 TO R51;1 TO A [

2:  0 TO B [

3:  RED 1,R(100+B),R(101+B) [

4:  90-R(100+B) TO R4 [

5:  COS R4*.082 TO R3 [

6:  ((1+R3)/(1-R3))↑.041 TO R3 [

7:  TAN (R4/2) TO R4 [

8:  R4*R3 TO R3 [

9:  LOG (1/R3) TO R(100+B) [

10: B+2 TO B;IF R51/R2>B;GTO 3 [

11: SSC Y [

12: RCF A,R100,R(R51/R2+99);PRT A;SPC 2 [

13: A+1 TO A [

14: IF A # R2+1;GTO 2 [

15: 1 TO A;LDF A,R100,R(R51/R2+99) [

16: 0 TO Z [

17: SCL 20,360,-.7509317,.7509317 [

18: PLT R(101+Z),R(100+Z) [

19: IF R51/R2-2 # Z;Z+2 TO Z;GTO -1 [

20: A+1 TO A;IF A <= R2;0 TO Z;LDF A,R100,R(R51/R2+99);GTO -
    2 [

21: PRT "END OF DATA" [

22: END  [
```

Figure B-1. Program to Plot Trajectory Ground Trace on a Mercator Projection (Cont'd)

GROUND TRACE ON MERCATOR PROJECTION



```
0: ENT "SELECT CODE,"
    Y, "NO. FILES",
    R2
```

SELECT APPROPRIATE CASSETTE UNIT AND DEFINE MAX. NUMBER OF FILES FOR DATA STORAGE.

```
1: 142→R51;
   1→A
```

NUMBER OF POINTS DEFINED AND FILE ONE IS DESIGNATED TO BE THE FIRST FILE LOADED.

```
2: 0→B
```

INITIALIZE A REGISTER COUNTER.

```
3: RED 1, R(100+B),
          R(101+B);
   PRT R(100+B), R(101+B);
   SPC 1
```

READ DATA FROM PAPER TAPE AND STORE IT IN APPROPRIATE REGISTER. PRINT OUT VALUES AS THEY ARE READ IN.

```
4: 90-R(100+B)→R4

5: COS R4 * .082→R3

6: ((1+R3)/(1-R3))
   .041→R3

7: TAN (R4/2)→R4
```

CALCULATE ORDINATE (LATITUDE) OF MERCATOR PROJECTION ACCORDING TO:

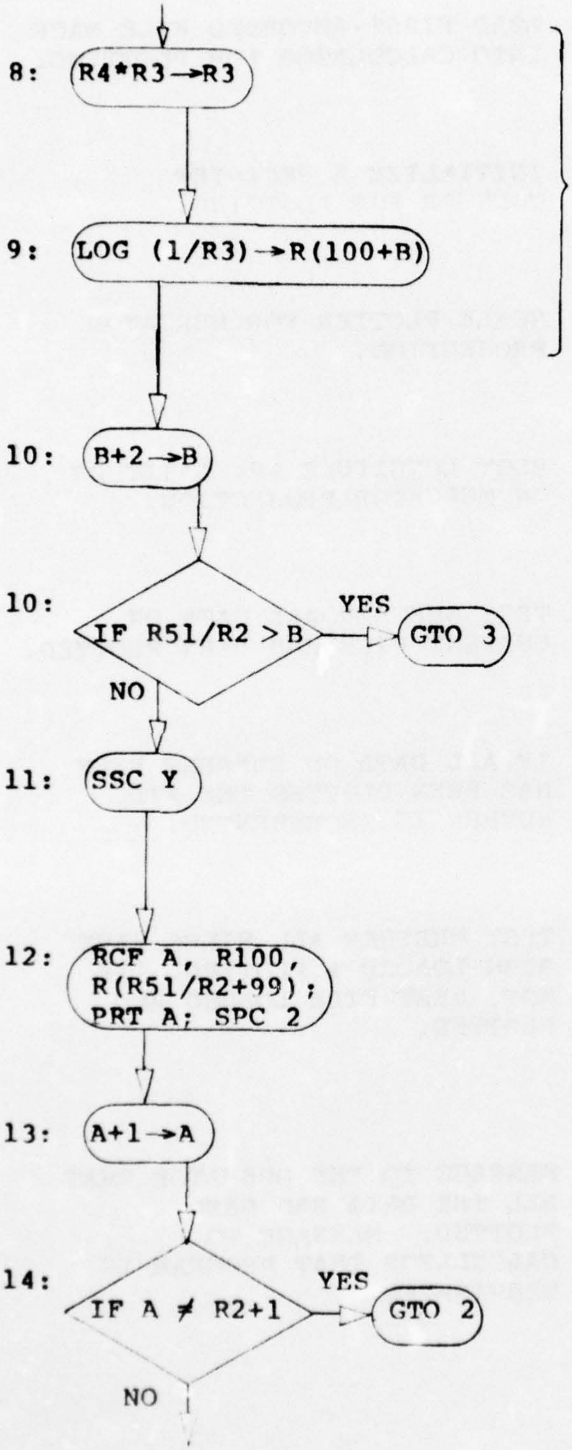
$$Y = \ln\left[1/\left\{\left(\frac{\tan(90-\Phi_p)}{2}\right)\left(\frac{1+e\cos(90-\Phi_p)}{1-e\cos(90-\Phi_p)}\right)\frac{e}{2}\right\}\right]$$

v = ORDINATE OF MERCATOR PROJECTION

e = EARTH'S ECCENTRICITY

Φ_p = LATITUDE

$90-\Phi_p$ = COLATITUDE



CONTINUATION OF CALCULATION OF LATITUDE FOR MERCATOR PROJECTION.

INCREMENT REGISTER COUNTER.

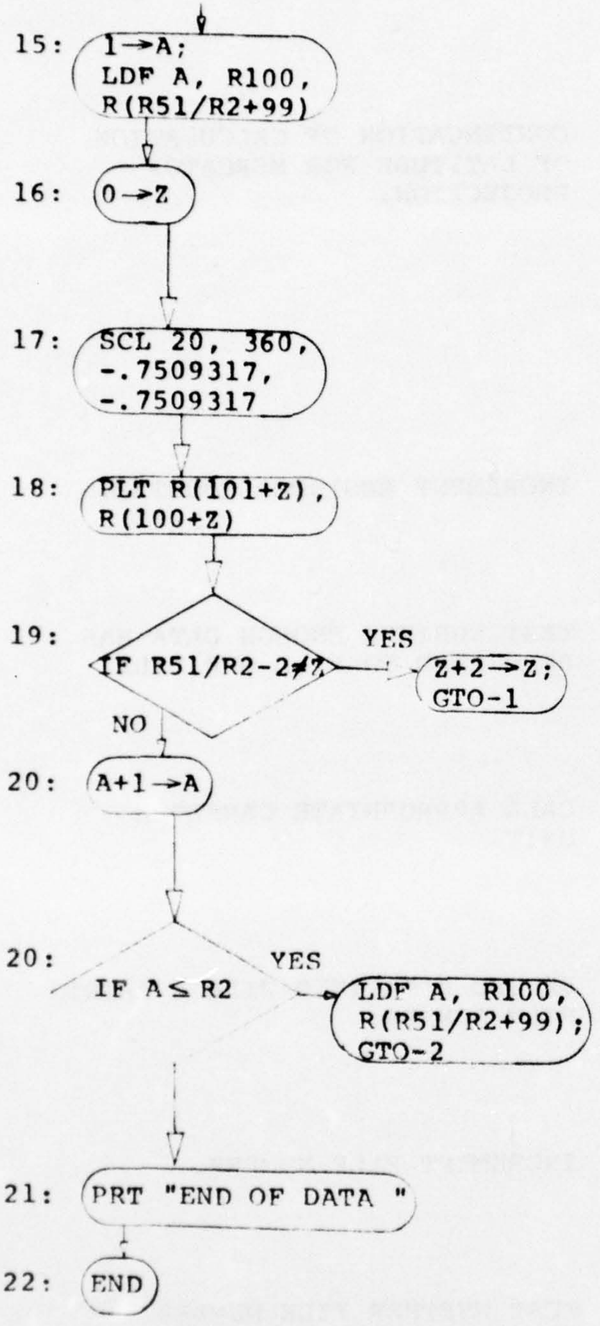
TEST WHETHER ENOUGH DATA HAS BEEN READ TO FILL ONE FILE.

CALL APPROPRIATE CASSETTE UNIT.

RECORD DATA INTO FILE & PRINT FILE NUMBER.

INCREMENT FILE NUMBER.

TEST WHETHER FILE NUMBER EXCEEDS MAX. FILE NUMBER. IF CURRENT FILE NO. EQUALS MAX. NO. THEN ALL DATA HAS BEEN RECORDED.



LOAD FIRST RECORDED FILE BACK INTO CALCULATOR FOR PLOTTING.

INITIALIZE A REGISTER COUNTER FOR PLOTTING.

SCALE PLOTTER FOR MERCATOR PROJECTION.

PLOT LONGITUDE VS. LATITUDE ON MERCATOR PROJECTION.

TEST WHETHER ALL DATA ON CURRENT FILE HAS BEEN PLOTTED.

IF ALL DATA ON CURRENT FILE HAS BEEN PLOTTED THE FILE NUMBER IS INCREMENTED.

TEST WHETHER ALL FILES HAVE BEEN LOADED & PLOTTED. IF NOT, NEXT FILE LOADED AND PLOTTED.

MESSAGE TO THE OPERATOR THAT ALL THE DATA HAS BEEN PLOTTED. MESSAGE TO CALCULATOR THAT PROGRAM IS TERMINATED.

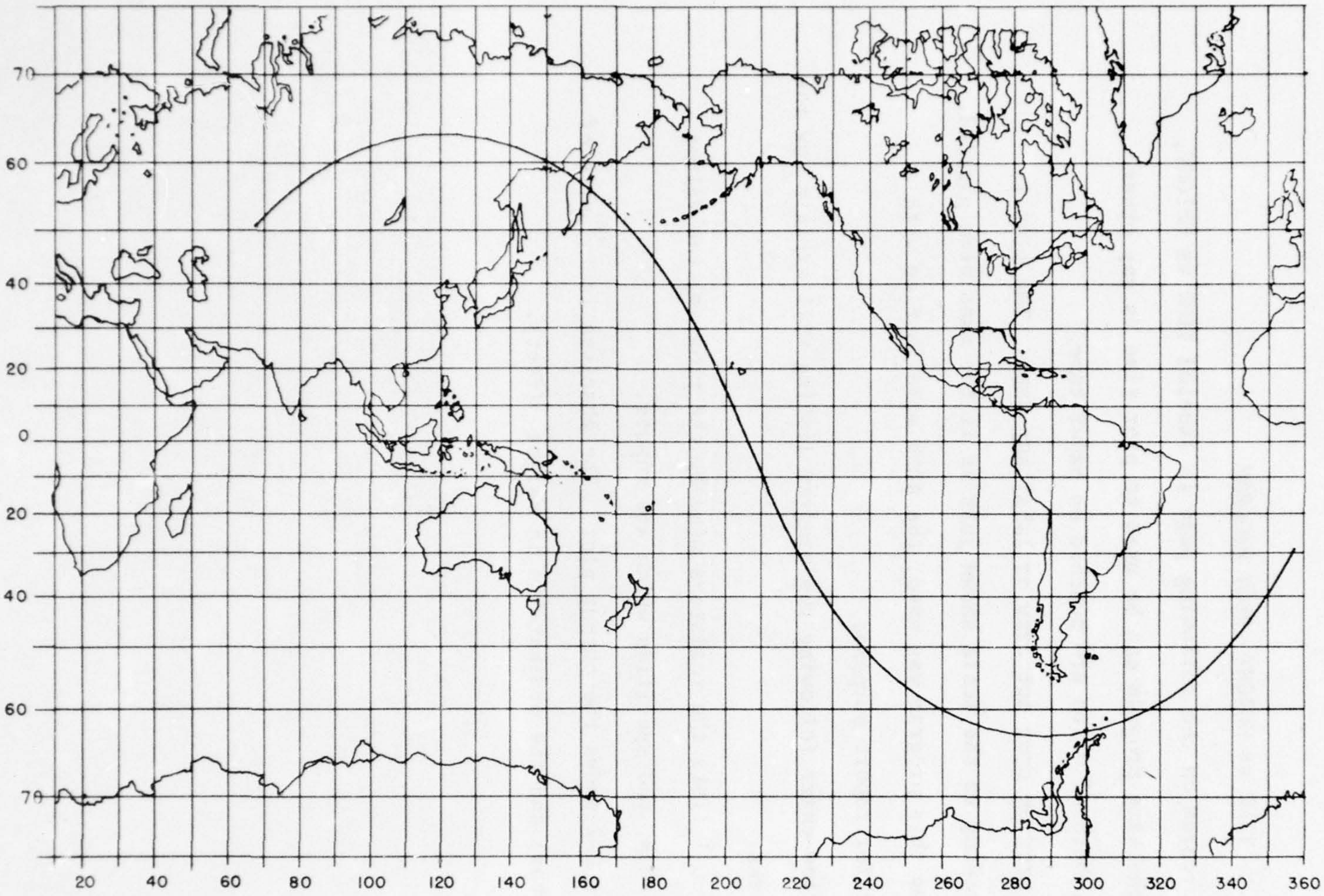


Figure B-2. Sample Ground Trace on Mercator Projection

APPENDIX C: TIME vs HEIGHT PLOT PROGRAM

The program on the following page is labeled TIME vs HEIGHT, however, the same program can be used to plot time vs any other parameter, both of which are punched on paper tape.

This program does not draw or label any axes. The plot is drawn according to the scale shown just as if the axes were plotted. At the time this program was used, the axes were labeled with a typewriter for report purposes.

The flow chart following the program listing will explain how the program works.

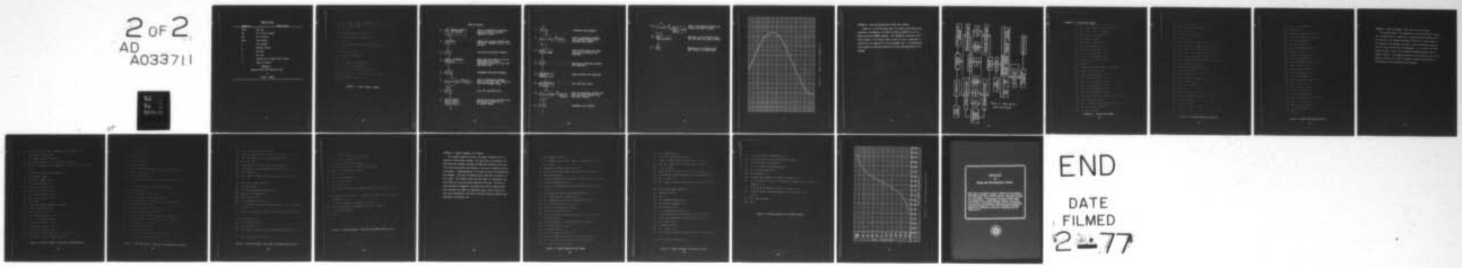
Table C-1 lists the registers used by the program, the values stored in each, and specifies which are inputs.

Figure C-2 shows the output plot. The abscissa is in units of time (minutes) and the ordinate is in height (feet).

AD-A033 711 ROME AIR DEVELOPMENT CENTER GRIFFISS AFB N Y F/G 15/3
SPACE SURVEILLANCE SOFTWARE SUPPORT. VOLUME II. GENERALIZED DAT--ETC(U)
OCT 76 J A MANLEY

UNCLASSIFIED RADC-TR-76-261-VOL-2 NL

2 OF 2
AD
A033711



END
DATE
FILMED
2-77

Register Table

Register	Value Stored
RO-R1	Not used
R2	No. of files (input)
R51	No. of points
R100 ...	Data storage
A	File counter
B	Register counter
C	Not used
X	Not used
Y	Select code of cassette unit (input)
Z	Register counter

Table C-1

Register Table with Inputs for Time

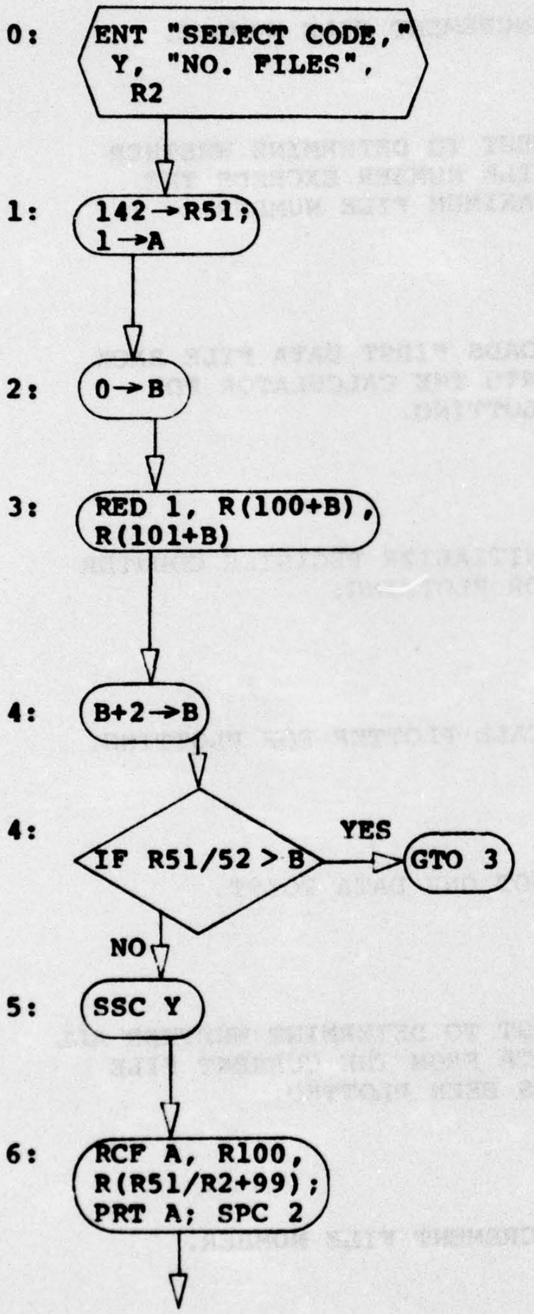
vs

Height Program

```
 0:  ENT "SELECT CODE",Y,"NO. FILES",R2 [
 1:  142 TO R51;1 TO A [
 2:  0 TO B [
 3:  RED 1,R(100+B),R(101+B) [
 4:  B+2 TO B;IF R51/R2>B;GTO 3 [
 5:  SSC Y [
 6:  RCF A,R100,R(R51/R2+99);PRT A;SPC 2 [
 7:  A+1 TO A [
 8:  IF A # R2+1;GTO 2 [
 9:  1 TO A;LDF A,R100 [
10:  0 TO Z [
11:  SCL 500000,2500000,0,70 [
12:  PLT R(101+Z),R(100+Z) [
13:  IF R51/R2-2 # Z;Z+2 TO Z;GTO -1 [
14:  A+1 TO A;IF A <= R2;0 TO Z;LDF A,R100;GTO -2 [
15:  PRT "END OF DATA" [
16:  END  [
```

FIGURE C-1. Time vs Height Program

TIME VS HEIGHT



SELECT APPROPRIATE CASSETTE UNIT AND DEFINE MAXIMUM NUMBER OF FILES.

NUMBER OF POINTS ASSIGNED AND THE FIRST CASSETTE FILE IS SET TO ONE.

INITIALIZE REGISTER COUNTER.

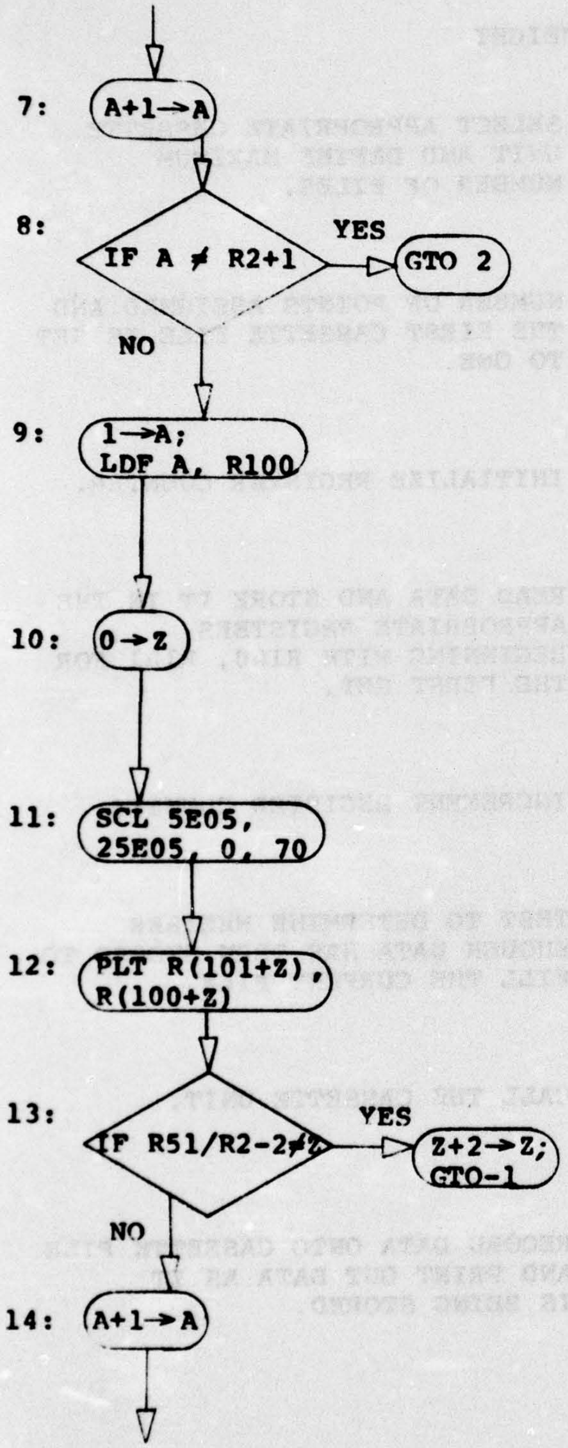
READ DATA AND STORE IT IN THE APPROPRIATE REGISTERS BEGINNING WITH R100, R101 FOR THE FIRST SET.

INCREMENT REGISTER COUNTER.

TEST TO DETERMINE WHETHER ENOUGH DATA HAS BEEN STORED TO FILL THE CURRENT FILE.

CALL THE CASSETTE UNIT.

RECORD DATA ONTO CASSETTE FILE AND PRINT OUT DATA AS IT IS BEING STORED.



INCREMENT FILE NUMBER.

TEST TO DETERMINE WHETHER FILE NUMBER EXCEEDS THE MAXIMUM FILE NUMBER.

LOADS FIRST DATA FILE BACK INTO THE CALCULATOR FOR PLOTTING.

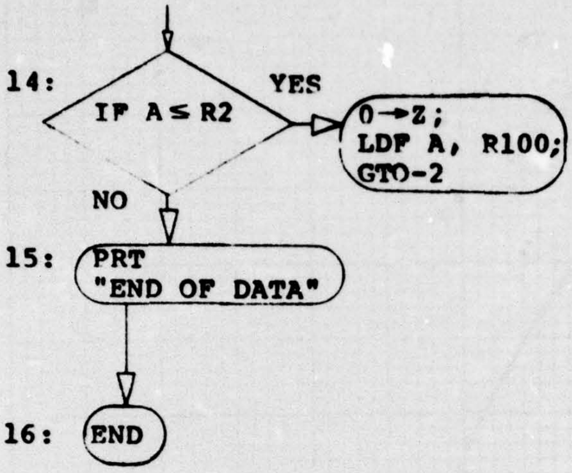
INITIALIZE REGISTER COUNTER FOR PLOTTING.

SCALE PLOTTER FOR PLOTTING.

PLOT ONE DATA POINT.

TEST TO DETERMINE WHETHER ALL DATA FROM THE CURRENT FILE HAS BEEN PLOTTED.

INCREMENT FILE NUMBER.



TEST TO DETERMINE WHETHER ALL FILES HAVE BEEN PLOTTED.

MESSAGE TO THE OPERATOR THAT ALL THE DATA HAS BEEN PLOTTED.

MESSAGE TO CALCULATOR THAT THE PROGRAM IS TERMINATED.

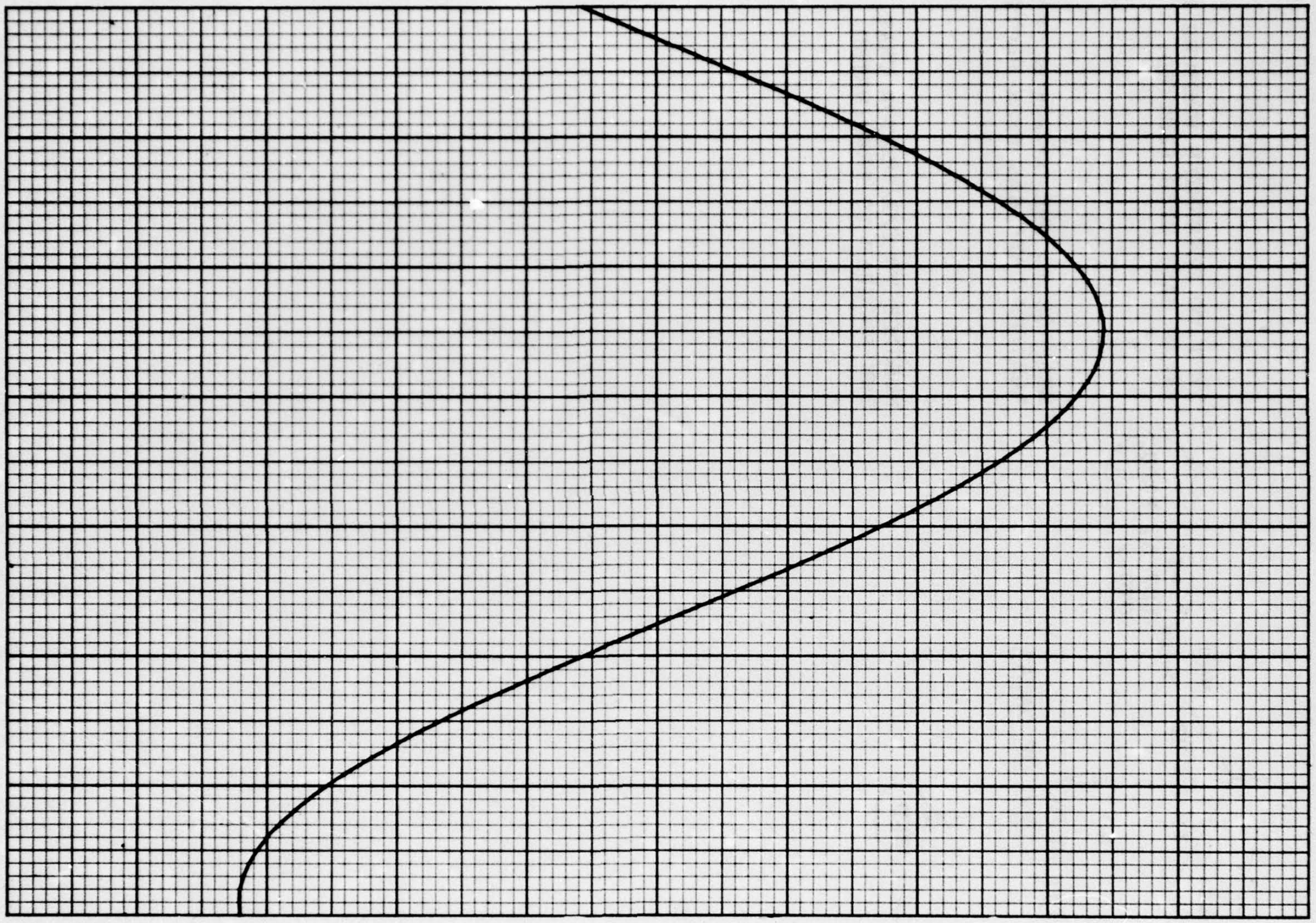


Figure C-2. Sample Time vs Height Plot

APPENDIX D: Radar System Simulation Model Block Diagram

Figure F-1, on the following page, is a sample of the major radar subsystem, environmental, and signal processing capabilities of the Radar Simulation (RADSIM) program. The information contained in this block diagram is by no means complete since it would be impossible to include all the capabilities on any meaningful chart. The generalized plot program was exercised extensively to plot data generated by this program.

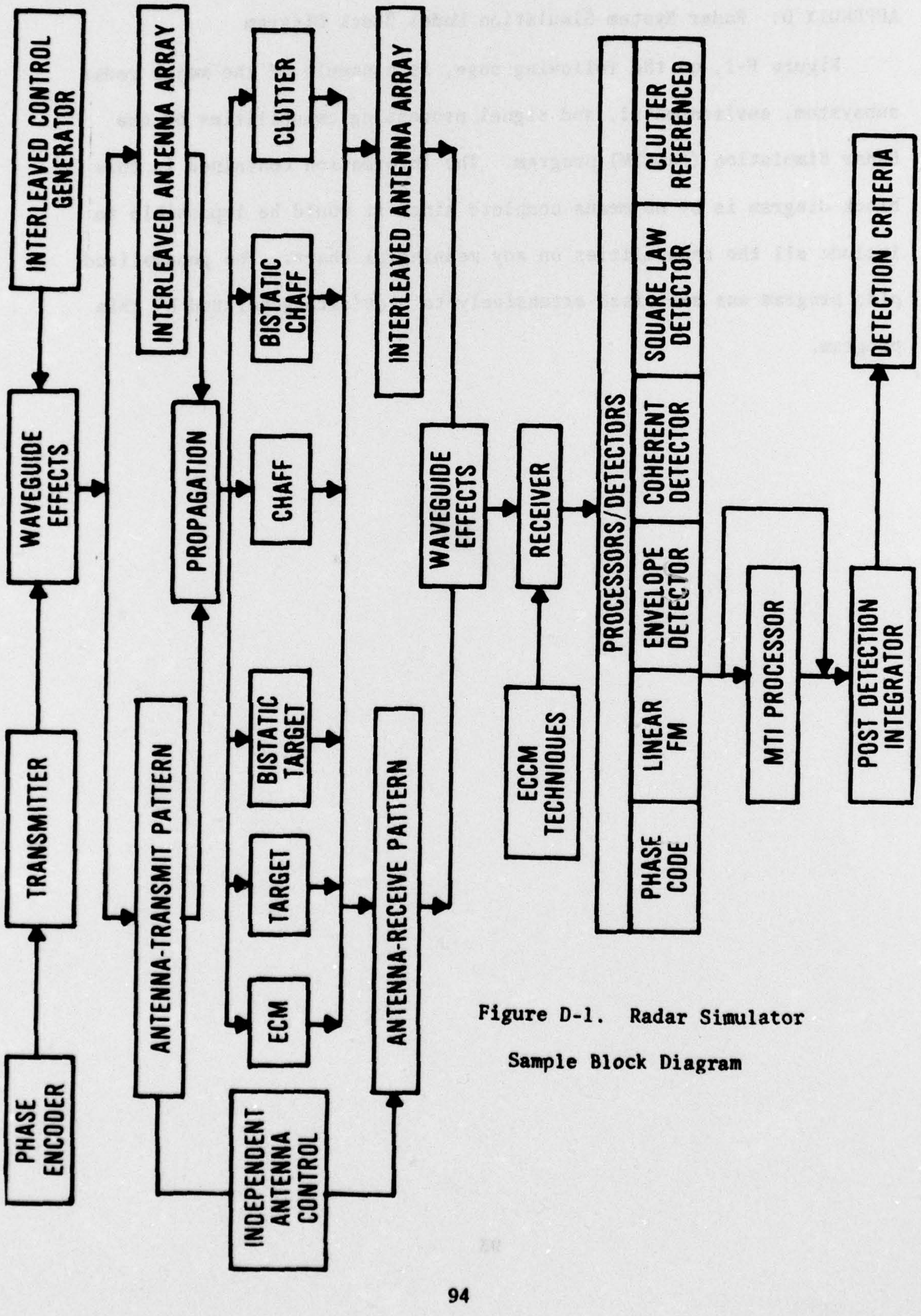
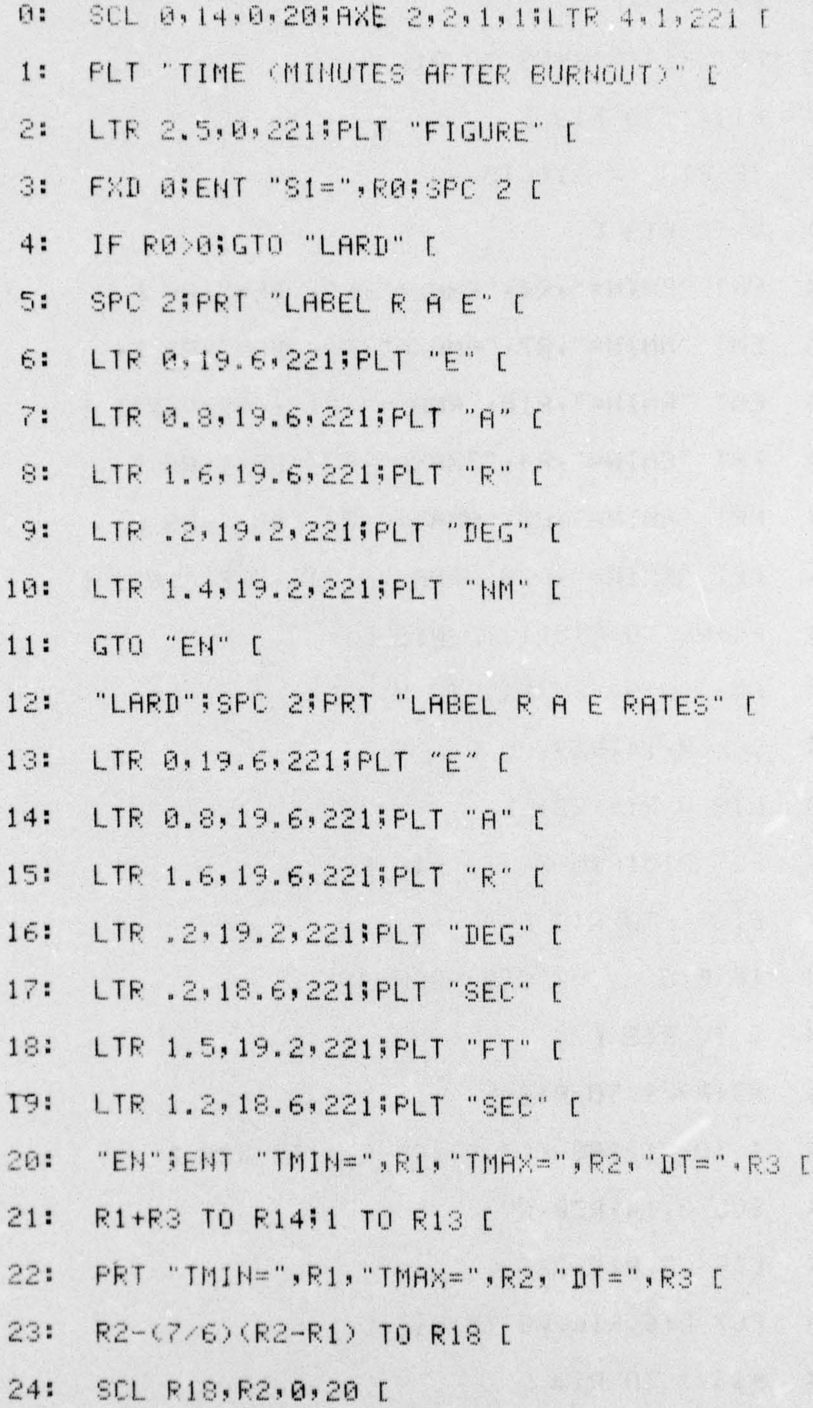


Figure D-1. Radar Simulator Sample Block Diagram

APPENDIX E: Original RAE Program



```
0:   SCL 0,14,0,20;AXE 2,2,1,1;LTR 4,1,221 [
1:   PLT "TIME (MINUTES AFTER BURNOUT)" [
2:   LTR 2.5,0,221;PLT "FIGURE" [
3:   FXD 0;ENT "S1=",R0;SPC 2 [
4:   IF R0>0;GTO "LARD" [
5:   SPC 2;PRT "LABEL R A E" [
6:   LTR 0,19.6,221;PLT "E" [
7:   LTR 0.8,19.6,221;PLT "A" [
8:   LTR 1.6,19.6,221;PLT "R" [
9:   LTR .2,19.2,221;PLT "DEG" [
10:  LTR 1.4,19.2,221;PLT "NM" [
11:  GTO "EN" [
12:  "LARD";SPC 2;PRT "LABEL R A E RATES" [
13:  LTR 0,19.6,221;PLT "E" [
14:  LTR 0.8,19.6,221;PLT "A" [
15:  LTR 1.6,19.6,221;PLT "R" [
16:  LTR .2,19.2,221;PLT "DEG" [
17:  LTR .2,18.6,221;PLT "SEC" [
18:  LTR 1.5,19.2,221;PLT "FT" [
I9:  LTR 1.2,18.6,221;PLT "SEC" [
20:  "EN";ENT "TMIN=",R1,"TMAX=",R2,"DT=",R3 [
21:  R1+R3 TO R14;1 TO R13 [
22:  PRT "TMIN=",R1,"TMAX=",R2,"DT=",R3 [
23:  R2-(7/6)(R2-R1) TO R18 [
24:  SCL R18,R2,0,20 [
```

FIGURE E-1. Original RAE Program

```
25:  LTR R14,1.5,221 [
26:  PLT R14;R14+R3 TO R14 [
27:  R13+1 TO R13 [
28:  IF R13 <= 11;GTO -3 [
29:  0 TO R13 [
30:  ENT "EMIN=",R4,"EMAX=",R5,"DE=",R6 [
31:  ENT "AMIN=",R7,"AMAX=",R8,"DA=",R9 [
32:  ENT "RMIN=",R10,"RMAX=",R11,"DR=",R12 [
33:  PRT "EMIN=",R4,"EMAX=",R5,"DE=",R6 [
34:  PRT "AMIN=",R7,"AMAX=",R8,"DA=",R9 [
35:  PRT "RMIN=",R10,"RMAX=",R11,"DR=",R12 [
36:  R4+R6 TO R15;1 TO R13 [
37:  R5-(10/9)(R5-R4) TO R19 [
38:  SCL 0,14,R19,R5 [
39:  LTR 0,R15,221 [
40:  PLT R15;R15+R6 TO R15 [
41:  R13+1 TO R13 [
42:  IF R13 <= 07;GTO -3 [
43:  0 TO R13 [
44:  R7+R9/2 TO R16 [
45:  1 TO R13;R8-(10/9)(R8-R7) TO R20 [
46:  SCL 0,14,R20,R8 [
47:  LTR .8,R16,221 [
48:  PLT R16;R16+R9 TO R16 [
49:  R13+1 TO R13 [
```

Figure E-1. Original RAE Program (Cont'd)

```
50:  IF R13 <= 08;GTO -3 [
51:  0 TO R13 [
52:  R10+R12 TO R17;1 TO R13 [
53:  R11-(10/9)(R11-R10) TO R21 [
54:  SCL 0,14,R21,R11 [
55:  LTR 1.5,R17,221 [
56:  PLT R17;R17+R12 TO R17 [
57:  R13+1 TO R13 [
58:  IF R13 <= 07;GTO -3 [
59:  SCL R18,R2,R19,R5 [
60:  FXD 2;SPC ;CFG 13;PRT "T","E" [
61:  ENT "T=",X,"E=",Y [
62:  PRT X,Y;PLT X,Y;SPC  [
63:  IF FLG 13=0;GTO -2 [
64:  SCL R18,R2,R20,R8 [
65:  CFG 13;PEN ;SPC 2;PRT "T","A" [
66:  ENT "T=",X,"A=",Y [
67:  PRT X,Y;PLT X,Y;SPC  [
68:  IF FLG 13=0;GTO -2 [
69:  SCL R18,R2,R21,R11 [
70:  CFG 13;PEN ;SPC 2;PRT "T","R" [
71:  ENT "T=",X,"R=",Y [
72:  PRT X,Y;PLT X,Y;SPC  [
73:  IF FLG 13=0;GTO -2 [
74:  END  [
```

Figure E-1. Original RAE Program (Cont'd)

APPENDIX F: RAE Plot Program - Data Input Via Punched Cards

The program listed in this appendix is used to plot range, azimuth, and elevation as well as range rate, azimuth rate and elevation rate in the same way as the program described in Section VI of this report. In this version of the program, the data is input from punched cards and the first card contains the plot parameters; i.e., the max y and min y values, the min. x value, the increment between data points, and the number of points. The program operation is very similar to that of the program in Section VI so that the register tables and flow charts in that section can also be used for this program.

```
 0:  PRT "MANLEY";SPC ;ENT "XLENGTH",R15,"YLENGTH",R16,
     "XUSED",R17,"YUSED",R18 [
 1:  ENT "RAE=0 RATES=1",R0;FXD 4 [
 2:  SCL 0,R15,0,R16;AXE 1,1,.5,.5;LTR 2,0,221 [
 3:  R15/R17 TO R42;R16/R18 TO R43;2*R17 TO R11;2*R18 TO R12;1 TO
     R49 [
 4:  PLT "TIME (MINUTES AFTER BURNOUT)" [
 5:  IF R0>0;GTO "LARD" [
 6:  SPC 2;PRT "LABEL R A E" [
 7:  LTR 0,9.8,221;PLT "E" [
 8:  LTR 0.4,9.8,221;PLT "A" [
 9:  LTR 0.8,9.8,221;PLT "R" [
10:  LTR .1,9.3,221;PLT "DEG" [
11:  LTR 0.7,9.3,221;PLT "NM" [
12:  GTO "HEAD" [
13:  "LARD";SPC 2;PRT "LABEL R A E RATES" [
14:  LTR 0,9.8,221;PLT "E" [
15:  LTR 0.4,9.8,221;PLT "A" [
16:  LTR 0.8,9.8,221;PLT "R" [
17:  LTR .1,9.6,221;PLT "DEG" [
18:  LTR .1,9.3,221;PLT "SEC" [
19:  LTR 0.7,9.6,221;PLT "FT" [
20:  LTR 0.6,9.3,221;PLT "SEC" [
21:  "HEAD";ENT "PNUMB",R48,"SSC",Y,"DFILE",A [
22:  TBL 6;ENT "R",R50,"A",R51,"E",R52 [
```

Figure F-1. RAE Plct Program - Data Input via Punched Cards

```
23:  IF R49>1;SFG 4 [
24:  IF R50=1;SFG 1 [
25:  IF R51=1;SFG 2 [
26:  IF R52=1;SFG 3 [
27:  IF R48=1;GTO +4 [
28:  FMT "D";WRT 2;FMT *;RED 2,R7,R6,R0,R13,R41;PRT R7,
     R6,R0,R13,R41;SPC 2 [
29:  (R41-1)R13+R0 TO R1 [
30:  SSC Y;RCF A,R0,R41;PRT R0,R1,R6,R7,R41;SPC 2;GTO +
     2 [
31:  SSC Y;LDF A,R0;PRT R0,R1,R6,R7,R41;SPC 2 [
32:  IF FLG 4;GTO +2 [
33:  ENT "TMIN",R20,"TMAX",R21 [
34:  ENT "YMIN",R22,"YMAX",R23,"FXD",X [
35:  ABS (R21-R20)/R11 TO R3;R20 TO R4;1 TO R40 [
36:  R21-R42(R21-R20) TO R5 [
37:  SCL R5,R21,0,R16;IF FLG 4;GTO +3 [
38:  FXD 1;LTR R4,.5,212;PLT R4 [
39:  1+R40 TO R40;IF R40 # R11+2;R4+R3 TO R4;GTO -1 [
40:  ABS (R23-R22)/R12 TO R8;R22 TO R9;1 TO R40 [
41:  R23-R43(R23-R22) TO R10 [
42:  SCL 0,R15,R10,R23;FXD X [
43:  IF FLG 2;R9+R8 TO R9 [
44:  "YLAB";IF FLG 1;LTR .6,R9,211;PLT R9 [
45:  IF FLG 2;LTR .4,R9,211;PLT R9 [
```

Figure F-1. RAE Plot Program - Data Input via Punched Cards (Cont'd)

```
46:  IF FLG 3;LTR 0,R9,211;PLT R9 [
47:  1+R40 TO R40;IF R40 # R12-9;R9+2R8 TO R9;GTO "YLAB" [
48:  ENT "NO. FILES",R2,"PTS/FILE",R40,"SFILE",A [
49:  1 TO C;0 TO B [
50:  5INT (R41/5) TO R45;0 TO R47;IF R2=1;GTO "2" [
51:  IF R48=1;SSC Y;LDF A,R50;(R2-1)R40 TO R47;GTO +19 [
52:  FMT "C";WRT 2 [
53:  FMT *;RED 2,R(50+B),R(51+B),R(52+B),R(53+B),R(54+B
     ) [
54:  B+5 TO B;IF R40-1>B;GTO -2 [
55:  R47+R40 TO R47 [
56:  SSC Y;RCF A,R50,R(R40+49);PRT A [
57:  IF R47=(R2-1)R40;GTO "2" [
58:  A+1 TO A;1+C TO C [
59:  IF C # R2+1;0 TO B;GTO -8 [
60:  "2";IF R41/5=INT (R41/5);SFG 5;R41-R47 TO R45 TO R46;G
     TO +3 [
61:  R41-R47 TO R46;5INT (R46/5) TO R45;IF R2=1;0 TO B;GTO +2
      [
62:  0 TO B;A+1 TO A [
63:  IF R48=1;GTO "PLT" [
64:  FMT "C";WRT 2 [
65:  FMT *;RED 2,R(50+B),R(51+B),R(52+B),R(53+B),R(54+B
     ) [
66:  B+5 TO B;IF R45-1>B;GTO -2 [
```

Figure F-1. RAE Plot Program - Data Input via Punched Cards (Cont'd)

```
67:  IF FLG 5;GTO +3 [

68:  FMT "C";WRT 2;FMT *;RED 2,R(50+B) [

69:  B+1 TO B;IF R46>B;GTO -1 [

70:  PRT A;SSC Y;RCF A,R50,R(R46+49) [

71:  "PLT";A-R2+1 TO A;1 TO C;PRT A;LDF A,R50 [

72:  0 TO Z TO X;R41-R47 TO R47 [

73:  ENT "SF",B [

74:  SCL R5,R21,R10,R23 [

75:  PLT R0,R(50+Z)/B [

76:  IF C=R2;GTO +3 [

77:  IF (R40-1 # Z)(R41 # X);R0+R13 TO R0;Z+1 TO Z;1+X TO X;GTO -
     2 [

78:  A+1 TO A;C+1 TO C;IF C <= R2;R0+R13 TO R0;0 TO Z;PRT A;LDF A
     ,R50;GTO -3 [

79:  IF R47-1 # Z;R0+R13 TO R0;Z+1 TO Z;GTO -4 [

80:  IF R49 # 3;R49+1 TO R49;PEN ;FXD 4;GTO "HEAD" [

81:  PRT "END OF DATA" [

82:  END  [
```

Figure F-1. RAE Plot Program - Data Input via Punched Cards (Cont'd)

APPENDIX G: Doppler Frequency Plot Program

This program computes and plots the Doppler frequency shift vs satellite time in radar coverage. The input data is generated by the RADC Trajectory Program, available in RADC/OCSA, where the range rate of the missile/satellite with respect to the sensor is used as input to this program. Program operation is the same as that of the Generalized Plot Program - Data Input Via Punched Cards, described in Section V of this report. The register tables and flow charts are applicable; however, there are four more inputs required of the user. The first is radar frequency in megahertz, the second and third are FXDX and FXDY which determine the number of significant digits used to label the x and y axes respectively, and lastly, SF, which scales the plotted data from hertz to kilohertz, etc.

```
0:  PRT "MANLEY";FXD 4 [
1:  ENT "PNUMB",R48,"SELECT CODE",Y,"DFILE",A,"FREQ (M
    HZ)",R30 [
2:  R30*10↑6 TO R30;IF R48=1;GTO +5 [
3:  FMT "D";WRT 2;FMT *;RED 2,R7,R6,R0,R13,R41;PRT R7,
    R6,R0,R13,R41;SPC 2 [
4:  (R41-1)R13+R0 TO R1;9.84252E8 TO R29;2R30/R29 TO R31 [
5:  R6R31 TO R6;R7R31 TO R7 [
6:  SSC Y;RCF A,R0,R41;PRT R0,R1,R6,R7,R13,R41;SPC 2;G
    TO +2 [
7:  SSC Y;LDF A,R0;PRT R0,R1,R6,R7,R41;SPC 2 [
8:  ENT "XLENGTH",R15,"YLENGTH",R16 [
9:  ENT "ABSCISSA USED",R17,"ORDINATE USED",R18 [
10: SCL 0,R15,0,R16;AXE 1,1,.5,.5 [
11: R15/R17 TO R42;R16/R18 TO R43 [
12: 2R17 TO R11;2R18 TO R12 [
13: ENT "XMIN",R20,"XMAX",R21,"FXDX",B,"YMIN",R22,"YMA
    X",R23,"FXDY",Z [
14: ABS (R21-R20)/R11 TO R3;R20 TO R4;1 TO R40 [
15: R21-R42(R21-R20) TO R5 [
16: SCL R5,R21,0,R16 [
17: FXD B;LTR R4,.4,222;PLT R4 [
18: 1+R40 TO R40;IF R40 # R11+2;R4+R3 TO R4;GTO -1 [
19: ABS (R23-R22)/R12 TO R8;R22 TO R9;1 TO R40 [
20: R23-R43(R23-R22) TO R10 [
```

FIGURE G-1. Doppler Frequency Plot Program

```
21:  SCL 0,R15,R10,R23 [

22:  FXD Z;LTR .5,R9,211;PLT R9 [

23:  1+R40 TO R40;IF R40 # R12+1;R9+R8 TO R9;GTO -1 [

24:  ENT "NO. FILES",R2,"PTS/FILE",R40,"SFILE",A [

25:  1 TO C;0 TO B [

26:  5INT (R41/5) TO R45;0 TO R47;IF R2=1;GTO "2" [

27:  IF R48=1;SSC Y;LDF A,R50;(R2-1)R40 TO R47;GTO +19 [

28:  FMT "C";WRT 2 [

29:  FMT *;RED 2,R(50+B),R(51+B),R(52+B),R(53+B),R(54+B
     ) [

30:  B+5 TO B;IF R40-1>B;GTO -2 [

31:  R47+R40 TO R47 [

32:  SSC Y [

33:  RCF A,R50,R(R40+49);PRT A [

34:  IF R47=(R2-1)R40;GTO "2" [

35:  A+1 TO A;1+C TO C [

36:  IF C # R2+1;0 TO B;GTO -8 [

37:  "2";IF R41/5=INT (R41/5);R41-R47 TO R46;GTO +3 [

38:  R45+5 TO R45;R45-R47 TO R46;IF R2=1;0 TO B;GTO +2 [

39:  0 TO B;A+1 TO A [

40:  IF R48=1;GTO +5 [

41:  FMT "C";WRT 2 [

42:  FMT *;RED 2,R(50+B),R(51+B),R(52+B),R(53+B),R(54+B
     ) [

43:  B+5 TO B;IF R46-1>B;GTO -2 [
```

Figure G-1. Doppler Frequency Plot Program (Cont'd)

```
44:   SSC Y;PRT A;RCF A,R50,R(R46+49) [

45:   A-R2+1 TO A;1 TO C;SPC 2;PRT A;LDF A,R50 [

46:   0 TO Z;R41-R47 TO R47;PRT A [

47:   ENT "SF",R33;SCL R5,R21,R10,R23 [

48:   PLT R0,R(50+Z)R31/R33 [

49:   IF C=R2;GTO +3 [

50:   IF R40-1 # Z=1;R0+R13 TO R0;Z+1 TO Z;GTO -2 [

51:   A+1 TO A;C+1 TO C;IF C <= R2;R0+R13 TO R0;0 TO Z;PRT A;LDF A
      ,R50;GTO -3 [

52:   IF R47-1 # Z=1;R0+R13 TO R0;Z+1 TO Z;GTO -4 [

53:   IF R33 # 1;SCL 0,R15,0,R16;LTR .3,6.7,211;PLT "X";PL
      T R33 [

54:   PRT "END OF DATA" [

55:   END  [
```

Figure G-1. Doppler Frequency Plot Program (Cont'd)

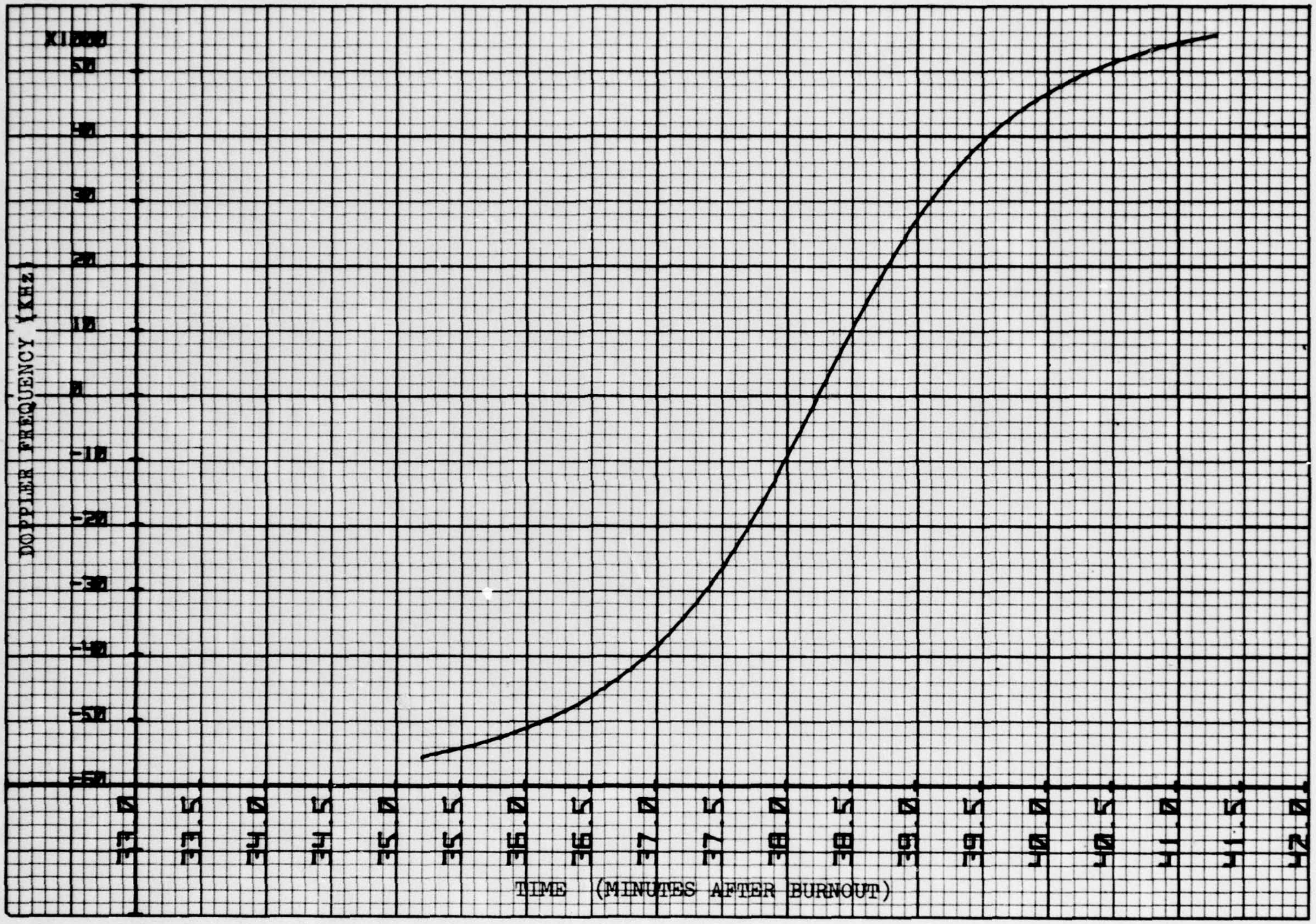


Figure G-2. Sample Doppler Frequency Plot

MISSION of Rome Air Development Center

RADC plans and conducts research, exploratory and advanced development programs in command, control, and communications (C^3) activities, and in the C^3 areas of information sciences and intelligence. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.

AMERICAN REVOLUTION BICENTENNIAL 1776-1976