NASA TECHNICAL NOTE

NASA TN D-2175

NASA TN D-2175

AN OPTIMIZED PRINTER PLOTTING SYSTEM CONSISTING OF COMPLEMENTARY 7090 (FORTRAN) AND 1401 (SPS) SUBROUTINES

PART II -- SYSTEMS PROGRAMMERS MANUAL

by Lois T. Dellner and Betty Jo Moore

Lewis Research Center
Cleveland, Ohio

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • APRIL 1964

AN OPTIMIZED PRINTER PLOTTING SYSTEM CONSISTING OF COMPLEMENTARY 7090 (FORTRAN) AND 1401 (SPS) SUBROUTINES

PART II - SYSTEMS PROGRAMMERS MANUAL

By Lois T. Dellner and Betty Jo Moore

Lewis Research Center
Cleveland, Ohio

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

For sale by the Office of Technical Services, Department of Commerce,
Washington, D.C. 20230 -- Price $2.50

CONTENTS

	Page
SUMMARY	1
INTRODUCTION	1
GENERAL DESCRIPTION	2
PLOTXY - (FORTRAN II)	4
SUBROUTINE PLOTXY	9
PLOTXY - BLOCK DIAGRAM	14
PLOTXY - DICTIONARY	15
PLOTMY - (FORTRAN II)	20
SUBROUTINE PLOTMY	26
PLOTMY - BLOCK DIAGRAM	34
PLOTMY - DICTIONARY	36
PISTUG - (FORTRAN II)	44
SUBROUTINE PISTUG	45
PISTUG - BLOCK DIAGRAM	48
PISTUG - DICTIONARY	49
SORTXY - (FORTRAN II)	52
SUBROUTINE SORTXY	53
SCALE - (FORTRAN II)	54
SUBROUTINE SCALE	55
PLOT - (1401 SPS)	56
SUBROUTINE PLOT	61
PLOT - BLOCK DIAGRAM	72
APPENDIXES	
A - SAMPLE PLOTS	79
B - NOTES	104
I. Machine Configuration	104
II. Storage Requirements	104
III. Operating and Programming Conventions	105
IV. FORTRAN II	105
V. Perforation Test (PISTUG)	106
VI. Range Test (PISTUG)	106
VII. Range Assumptions (PLOT)	107
VIII. Range Assumptions (PISTUG)	107
IX. Direction Assumptions (PLOTMY)	107
X. Duplicate Points	107
XI. Multi-Record Titles and Legends	107
XII. Error Write-Outs	108
REFERENCES	110

AN OPTIMIZED PRINTER PLOTTING SYSTEM CONSISTING OF COMPLEMENTARY 7090 (FORTRAN) AND 1401 (SPS) SUBROUTINES

PART II - SYSTEMS PROGRAMMER MANUAL

by Lois T. Dellner and Betty Jo Moore

Lewis Research Center

SUMMARY

Complementary subroutines, easy for the FORTRAN programmer to use, provide printed plots as part of the normal off-line output. These subroutines are simplified so that no choices must be made and generalized so that choices may be made, by the programmer, of the plotting characters, the scales, the appearance of the grid, the number of curves, and other options. The 7090 routines write information-packed records that are decoded and used by the 1401 subroutine to produce the plot at full printer speed (6 sec/page). Part I (Technical Note D-2174) of this presentation is an instruction manual for the users of the system. Part II is intended to simplify the work of the systems programmer who implements, and probably must modify, the system, and then must help debug users' programs.

INTRODUCTION

The OPP system (Optimized Printer Plotting), a set of complementary subroutines for the IBM 7090 and 1401, is a simple-to-use system that requires minimum storage, reduces 7090 execution time (by a factor of more than 10 from that required by presently available machine plotting methods), and produces printed plots at full printer speed (6 sec/page) as part of normal off-line output. This optimization is achieved by combining three techniques: (1) performing most of the requisite arithmetic on the 7090, (2) packing the results on the output tape, (3) having the 1401 generate the grid and the grid-labels, as well as position the points in a subroutine of the standard tape-to-printer program.

The system permits, but does not require, the programmer to choose the plotting characters, the scales, the grid-line spacing, etc. It handles single or multiple curves, prints true scales, and permits printing of titles above and below the plot.

The chief advantages of OPP system plotting over manual plotting are the complete elimination of elapsed time and the reduction of the cost per plot by a

factor of at least 50. Even when the limitations of the printer - 10 characters per inch horizontally (to a minimum of 100) and 6 characters per inch vertically - make a printed plot inadequate for a particular application, the time required to produce a manual plot from the same data is reduced by at least 5 minutes, because the printed plot supplies guidance in the choice of scales and the minimum number of points required to define the curve.

More than 150 programmers are now using this system effectively and more than 500 plots are printed weekly. The complementary subroutine PLOT for the 1401 was written for a 4K machine.

GENERAL DESCRIPTION

The OPP system offers the programmer plotted output with a minimum of effort. After writing a title for his plot on the output tape, he writes CALL PLOTXY or CALL PLOTMY (for multiple curves). The arguments, or call list, include the names of the arrays to be plotted and specify the number of points per curve and the number of curves. The programmer then writes a legend to be printed at the bottom of the plot.

The plot or plots are printed as part of the regular output listing with no delay. No changes in his card-handling procedures nor special instructions for the operators are required.

If he is using PLOTXY, the values of the variable to be plotted in the x-direction must be in sequence. If they are not, the subroutine SORTXY is supplied to be used before calling PLOTXY. For either PLOTXY or PLOTMY, if the size of the elements in (or the total range of) any array is not known to be within certain limits, the programmer calls the subroutine SCALE for each array before calling the plotting subroutine. SCALE will transform the array to suit PLOTXY and PLOTMY only if it is necessary.

In addition to the minimum-effort use just described, the programmer may choose to use one or more of several options that permit him to control, for example, the appearance of the grid (by specifying the frequency of the grid-lines in either direction), the scale for either variable (by specifying the scale-factor and a starting-value), the plotting character, etc.

This presentation of the OPP system is in two sections. Part I (ref. 1) is a manual for the FORTRAN programmer who wants printed plots as part of his normal output. The general objective of Part II is to simplify the work of the systems programmer who implements, and probably must modify, the OPP system, and then must help debug users' programs. To such a programmer, the system consists of a set of FORTRAN II subroutines (PLOTXY, PLOTMY, PISTUG, SCALE, and SORTXY) and one 1401 (SPS) subroutine (PLOT) incorporated into a standard tape-to-printer program. PLOTXY and PLOTMY write information-packed records on the output tape, and PLOT decodes these records and generates the plot, one line at a time, preparing the next line as the current line is being printed, maintaining full printer speed the while.

A brief discussion and a complete listing are included for all routines. For PLOTXY, PLOTMY, and PLOT, detailed specifications of input and output and comprehensive block diagrams are also provided. For PLOTXY, PLOTMY, and PISTUG, dictionaries of the FORTRAN variables are supplied. These are intended as systems debugging aids. Each is designed for use primarily as a cross-reference with the corresponding FORTRAN listing and block diagram. Within each dictionary, each variable name is identified in terms of its contribution to the program. Throughout the text, a subroutine name is printed at the bottom of each page for assistance in rapid cross-referencing.

Appendix A consists of a set of sample plots each of which is accompanied by a representation of the associated tape records written by the 7090 and used by the 1401 to produce it. The sources of some difficulties are discussed in appendix B, and corrective actions suggested where feasible.

PLOTXY - (FORTRAN II)

To plot one curve, the programmer will usually use PLOTXY. His call statement lists the names of the arrays to be plotted, the number KODE, (an indicator of which options are being used (see I.-D.), and finally, an array whose first element is the number of points to be plotted and whose other elements are any additional data required by the choice of option.

PLOTXY may call an auxiliary subroutine, PISTUG, (p. 44) to compute scaling parameters, then it writes one INFO record specifying the appearance of the grid to the 1401 subroutine PLOT. This is followed by one or more packed DATA records that specify the location of each point to be plotted in terms of KX, the number of lines down from the first x grid-line, and KY, the number of print positions to the right of the first y grid-line.

I. Input to this routine is supplied via the four arguments of the CALL, as follows:

CALL PLOTXY (XDOWN, YACROS, KODE, P)

A. XDOWN and YACROS are floating-point variable array names. In general, the values in both arrays must lie between -10^6 and 10^6. (See appendix B, VI., for precise limitations.) The subroutine SCALE (p. 54) is supplied to handle arrays whose element size is not known. In this case, SCALE should be called by the programmer before calling PLOTXY.

The elements of the XDOWN array must be in either decreasing or increasing sequence. The subroutine SORTXY (p. 52) is supplied to rearrange the arrays when the XDOWN array is not in order. It should be called by the programmer before calling PLOTXY.

B. KODE is a fixed-point variable or constant whose value is restricted by $0 \leq \text{KODE} \leq 127$. Seven different options are offered to the programmer. Each has a number associated with it. The sum of the numbers corresponding to the options used is the KODE.

C. P is a floating-point variable array name. P(1), the first value in the P array, is the number of points to be plotted. The other values to be supplied in the P array are determined by the particular options being used.

D. Options. The sample plots in appendix A illustrate the effects of using the various options.

NUMBER	OPTION
1	Choice of plotting character. - The plotting character will be an asterisk unless the programmer uses this option. When he does, he must specify the plotting character by placing its BCD code in the leftmost six bits of P(2). He may use any character in the set of all FORTRAN characters except the minus sign "-".
2	Choice of x grid-line frequency. - An x grid-line will be printed every 10 line-spaces below the first one unless the programmer uses this option. When he does, the desired frequency (M) of x grid-lines must be supplied to PLOTXY in P(3). If M = 0, only the first and final x grid-lines will be printed.
4	Choice of y grid-line frequency. - A y grid-line will be printed every 10 print positions to the right of the first (leftmost), unless this option is used. If it is, the desired frequency (N) of y grid-lines must be supplied in P(4). If N = 0, only the first y grid-line will be printed.
16	Choice of x scale. - PLOTXY will supply the three scaling parameters that specify the x scale starting-value, the scale-factor (value of one line space), and the x grid-labels, unless this option is used. When it is used, the programmer must specify his own scale by supplying the three parameters KSX, FX, and DX in P(6), P(7), and P(8), respectively. These parameters are defined by: FX is the desired starting-value multiplied by 10^U. DX is the desired scale-factor for one line-space multiplied by 10^U. U must be chosen so that FX and DX are integers and $0 \leq U \leq 6$. KSX is equal to 6 - U.

NUMBER	OPTION
32	Choice of y scale. - PLOTXY will supply the three scaling parameters that specify the y-scale starting-value, the scale-factor (value of one print position), and the y grid-labels, unless the programmer uses this option. If he does, he must specify his own scale by supplying the three parameters, KSY, FY, and DY in P(9), P(10), and P(11), respectively. These parameters are defined by: FY is the desired starting-value multiplied by 10^V. DY is the desired scale-factor for one print position multiplied by 10^V. V must be chosen so FY and DY are integers and $0 \leq V \leq 6$. KSY is equal to 6 - V.
64	Choice of printing coordinates at left of plot. - The 29 print positions to the left of the first y grid-line are usually blank except for x grid-labels every tenth line in positions 20 to 28. If option 64 is used, the coordinates of each point are printed immediately to the left[1] of the grid-labels on the same line as the point.
8	Choice of printing message at left of plot. - The 29 print positions to the left of the first y grid-line are usually blank except for x grid-labels printed in positions 20 to 28. When option 8 is used, the programmer must supply a 6. in P(5) and the information to be printed in P(12), P(13), ... The 6 BCD characters in P(12) are printed immediately to the left[1] of the grid-labels on the same line on which the first point is printed, the six characters in P(13) are printed with the second point, etc. Specifically, the 6 BCD characters in P(I+11) are printed on the same line on which the I^{th} value of the XDOWN array is plotted.

[1]If options 8 and 64 are used together, the BCD characters will be printed to the left of the coordinates.

II. Output from this routine consists of one INFO record and one or more DATA records. Sample tape records appear in appendix A.

A. INFO record.

The contents of the INFO record are the information needed by the 1401 subroutine PLOT to determine the construction of the grid, the construction of the grid-labels, the plotting character, and the format of the succeeding DATA records. Specifically, the INFO record consists of the two characters PI, followed by 10 fields, whose contents are:

(Field width is shown in parentheses.)

a. (1) This is the plotting character for all points in the DATA records.

b. (1) This digit is the decimal-point-shift code, KSX, for OFX and ODX. (Items d. and f. in this list.)

c. (1) This digit is the decimal-point-shift code, KSY, for OFY and ODY. (Items e. and g. in this list.)

d. (9) The number (OFX) in this field is in the form ±0.xxxxxx and represents the x-scale starting-value.

e. (9) The number (OFY) in this field is in the form ±0.xxxxxx and represents the y-scale starting-value.

f. (9) The number (ODX) in this field is in the form ±0.xxxxxx and represents the x scale-factor (value of one line-space).

g. (9) The number (ODY) in this field is in the form ±0.xxxxxx and represents the y scale-factor (value of one print position).

h. (3) This integer is the frequency, M, of the x grid-lines.

i. (3) This integer is the frequency, N, of the y grid-lines.

j. (3) This integer specifies the length, NF, of the first field in each block of information in the following DATA record. NF may be zero.

B. DATA records.

The contents of a DATA record (after the first two characters PD) are successive blocks of the information needed by the 1401 subroutine PLOT to position one plotting character and print the left-side field on the same line. Within a record, only complete blocks are permitted; one record may hold as many as 21 blocks. Within a block, the location in which the plotting character for one point is to be printed is specified by two positive integers, KX (the number of line-spaces below the first x grid-line) and KY (the number of print positions to the right of the first y grid-line). Specifically, the format of one block in the DATA record is:

a. The first NF (see j. preceding) characters are those to be printed in the left-side field (to the left of the first y grid-line) unless NF is zero; in this case, the characters to be printed are blanks and will be generated by the 1401 subroutine.

b. The next three characters are KX.

c. The next (and last) three characters are KY.

III. After "initializing" switches, the routine inspects the input variable KODE bit by binary bit, using Boolean statements and octal masks. It branches on each bit to set necessary output switches, and either supplies standard values or picks up programmer-provided specifications from the P array. If necessary, it calls an auxiliary routine (PISTUG) for scaling either or both coordinate arrays. The INFO record (II.-A.) is then written.

The routine then uses the scaling parameters to calculate KX (the number of line-spaces down from the first x grid-line) and KY (the number of print positions to the right of the first y grid-line) for each point, which together determine where the plotting character should be placed. As soon as enough points have been processed to fill a DATA record, it is written out, and this continues until all the points have been processed. Control is then returned to the calling program.

```
      SUBROUTINE PLOTXY(X,Y,BK,P)

      SUBROUTINE PLOTXY(X,Y,BK,P)
C                                          THIRD ARGUMENT IS A BOOLEAN VARIABLE
C
      DIMENSION X(100),Y(100),P(111),KX(21),KY(21),XOW(10),YOW(10),
     1 POW(10)
C                                          IT IS ALSO USED AS AN INTEGER
      EQUIVALENCE (   BKK,KK)
C
  100 N=P(1)+SIGNF(.5,P(1))
  101 BKK=BK
  102 KSW=0
  103 KHDG=2
C                                          GO TO 140 IF KODE IS NOT ZERO
  104 IF(KK) 140,106,140
  106 A1=1H*
  107 M=10
  108 NY = 10
  109 NF1=0
C                                          SCALE X AND TO 190 OR ERROR EXIT
  120 K=3
  121 CALL PISTUG(K,N,X,DX,FX,KSX,KHARX)
  122 KDX=DX
  124 IF(KDX) 190,900,190
C                                          SCALE Y AND TO 220 OR ERROR EXIT
  126 K=2
  128 CALL PISTUG(K,N,Y,DY,FY,KSY,KHARY)
  130 KDY=DY
  132 IF(KDY) 220,900,220
C
C
C                                          OPTION 1 - USE * OR P(2)
B 140 IF(BKK*1000000) 148,148,144
  144 A1=P(2)
  146 GO TO 150
  148 A1=1H*
C                                          OPTION 2 - USE 10 OR P(3)
B 150 IF(BKK*2000000) 158,158,154
  154 M=P(3)+SIGNF(.5,P(3))
  156 GO TO 160
  158 M=10
C                                          OPTION 4 - USE 10 OR P(4)
B 160 IF(BKK*4000000) 168,168,164
  164 NY = P(4)+SIGNF(.5,P(4))
  166 GO TO 170
  168 NY = 10
C                                          OPTION 8 - ALTER KSW AND NF1
B 170 IF(BKK*10000000) 178,178,174
  174 NF1=P(5)+SIGNF(.5,P(5))
  176 KSW=1
  177 GO TO 200
  178 NF1=0
C                                          OPTION 64 - ALTER KSW AND NF1
B 200 IF(BKK*100000000) 180,180,204
  204 NF1=NF1+13
  206 KSW=KSW+2
  208 KHDG=1
```

```
      SUBROUTINE PLOTXY(X,Y,BK,P)

C                                          OPTION 16 - GO TO 120 OR USE P(6-8)
B 180 IF(BKK*20000000) 120,120,184
  184 KSX=P(6)+SIGNF(.5,P(6))
  185 FX=P(7)
  186 DX=P(8)
  187 RFX=FX*(10.**(KSX-6))
      RDX=DX*(10.**(KSX-6))
      PFX=RFX+100.*RDX
      XMAX=MAX1F(ABSF(RFX),ABSF(PFX))
      KHARX=XINTF(LOGF(XMAX)/2.302585+40.)-40
C                                          OPTION 32 - GO TO 126 OR USE P(9-11)
B 190 IF(BKK*40000000) 126,126,194
  194 KSY=P(9)+SIGNF(.5,P(9))
  196 FY=P(10)
  198 DY=P(11)
  199 RFY=FY*(10.**(KSY-6))
      RDY=DY*(10.**(KSY-6))
      PFY=RFY+100.*RDY
      YMAX=MAX1F(ABSF(RFY),ABSF(PFY))
      KHARY=XINTF(LOGF(YMAX)/2.302585+40.)-40
C                                          GO TO 250 OR PRINT SPECIAL HEADING
  220 GO TO (230,250),KHDG
  230 KPWRX=3-KHARX
  232 KPWRY=3-KHARY
  234 F10X=10.**KPWRX
  236 F10Y=10.**KPWRY
  238 WRITE OUTPUT TAPE 6,500,KPWRX,KPWRY
  250 ODX = DX* 1.E-6+SIGNF(.0000005,DX )
      OFX = FX* 1.E-6+SIGNF(.0000005,FX )
      ODY = DY*1 .E-6+SIGNF(.0000005,DY )
      OFY = FY* 1.E-6+SIGNF(.0000005, FY)
C                                          WRITE INFO RECORD
  251 WRITE OUTPUT TAPE 6,502,A1,KSX,KSY,OFX,OFY,ODX,ODY,M,NY,NF1
C
C
C                                          INITIALIZE FOR OUTPUT LOOPING
  300 L=130/(6+NF1)
      J=1
C                                               FOR ALL POINTS
  310 DO 498 I=1,N
C                                          COMPUTE COUNT DOWN AND ACROSS
  320 C6 = X(I)*(10.**(6-KSX))/DX
  322 C7=INTF(C6+SIGNF(.5,C6))
  324 KX(J)=(DX*C7-FX)/DX+.5
  330 C6 = Y(I)*(10.**(6-KSY))/DY
  332 C7=INTF(C6+SIGNF(.5,C6) )
  334 KY(J)=(DY*C7-FY)/DY+.5
C                                          GO TO 346 OR PICKUP COORDINATES
  338 KBR=KSW/2+1
  340 GO TO (346,342),KBR
  342 XOW(J)=F10X*X(I)
  344 YOW(J)=F10Y*Y(I)
C                                          GO TO 350 OR PICKUP P ARRAY MESSAGE
  346 KBR2=KSW+1
  347 GO TO(350,348,350,348),KBR2
  348 POW(J)=P(I+11)
```

```
      SUBROUTINE PLOTXY(X,Y,BK,P)

C                                          GO TO 360 UNLESS READY TO WRITE RECORD
  350 IF(J-L) 360,352,360
  352 J=0
  354 GO TO 400
C                                          GO TO 498 IF ALL POINTS NOT DONE
  360 IF(I-N) 498,362,498
  362 L=J
C                                          WRITE A DATA RECORD
  400 KBRM=KSW+1
  402 GO TO (410,420,430,440),KBRM
  410 WRITE OUTPUT TAPE 6,510,(KX(J),KY(J),J=1,L)
  412 GO TO 498
  420 WRITE OUTPUT TAPE 6,520,(POW(J),KX(J),KY(J),J=1,L)
  422 GO TO 498
  430 WRITE OUTPUT TAPE 6,530,(XOW(J),YOW(J),KX(J),KY(J),J=1,L)
  432 GO TO 498
  440 WRITE OUTPUT TAPE 6,540,(POW(J),XOW(J),YOW(J),KX(J),KY(J),J=1,L)
  498 J=J+1
C                                          END OF DO THAT STARTED AT  310
  900 RETURN
C
  500 FORMAT(2HPT/2HPT/2HPT,8X,3HX*E,I2,5H  Y*E,I2)
  502 FORMAT(2HPI,A1,2I1,4F9.6,3I3)
  510 FORMAT (2HPD,42I3)
  520 FORMAT (2HPD,10(A6,I3,I3))
  530 FORMAT (2HPD,6(F6.0,F7.0,    I3,I3))
  540 FORMAT (2HPD,5(A6,F6.0,F7.0,    I3,I3))
      END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

SUBROUTINE PLOTXY(X,Y,BK,P)

STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
875	01553	32561	77461

STORAGE LOCATIONS FOR VARIABLES APPEARING IN DIMENSION AND EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
BKK	874	01552	KK	874	01552	KX	873	01551	KY	852	01524	POW	811	01453
XOW	831	01477	YOW	821	01465									

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENT

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
A1	801	01441	C6	800	01440	C7	799	01437	DX	798	01436	DY	797	01435
F10X	796	01434	F10Y	795	01433	FX	794	01432	FY	793	01431	I	792	01430
J	791	01427	KBR2	790	01426	KBRM	789	01425	KBR	788	01424	KDX	787	01423
KDY	786	01422	KHARX	785	01421	KHARY	784	01420	KHDG	783	01417	KPWRX	782	01416
KPWRY	781	01415	K	780	01414	KSW	779	01413	KSX	778	01412	KSY	777	01411
L	776	01410	M	775	01407	NF1	774	01406	N	773	01405	NY	772	01404
ODX	771	01403	ODY	770	01402	OFX	769	01401	OFY	768	01400	PFX	767	01377
PFY	766	01376	RDX	765	01375	RDY	764	01374	RFX	763	01373	RFY	762	01372
XMAX	761	01371	YMAX	760	01370									

SYMBOLS AND LOCATIONS FOR SOURCE PROGRAM FORMAT STATEMENTS

	EFN	LOC		EFN	LOC		EFN	LOC		EFN	LOC		EFN	LOC
8)FK	500	01353	8)FM	502	01344	8)FU	510	01340	8)G8	520	01336	8)GI	530	01332
8)GS	540	01325												

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
1)	751	01357	2)	687	01257	3)	700	01274	6)	714	01312	9)	748	01354
B)	696	01270	C)G1	754	01362	C)G2	755	01363	C)G3	756	01364	C)G4	757	01365
C)G5	758	01366	C)G6	759	01367	D)21G	682	01252	D)31G	681	01251	E)R	545	01041
E)T	559	01057	E)10	575	01077	E)15	605	01135	E)18	626	01162	E)1B	649	01211

LOCATIONS OF NAMES IN TRANSFER VECTOR

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
EXP(2	1	00001	LOG	2	00002	PISTUG	0	00000	(FIL)	4	00004	(STH)	3	00003

ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

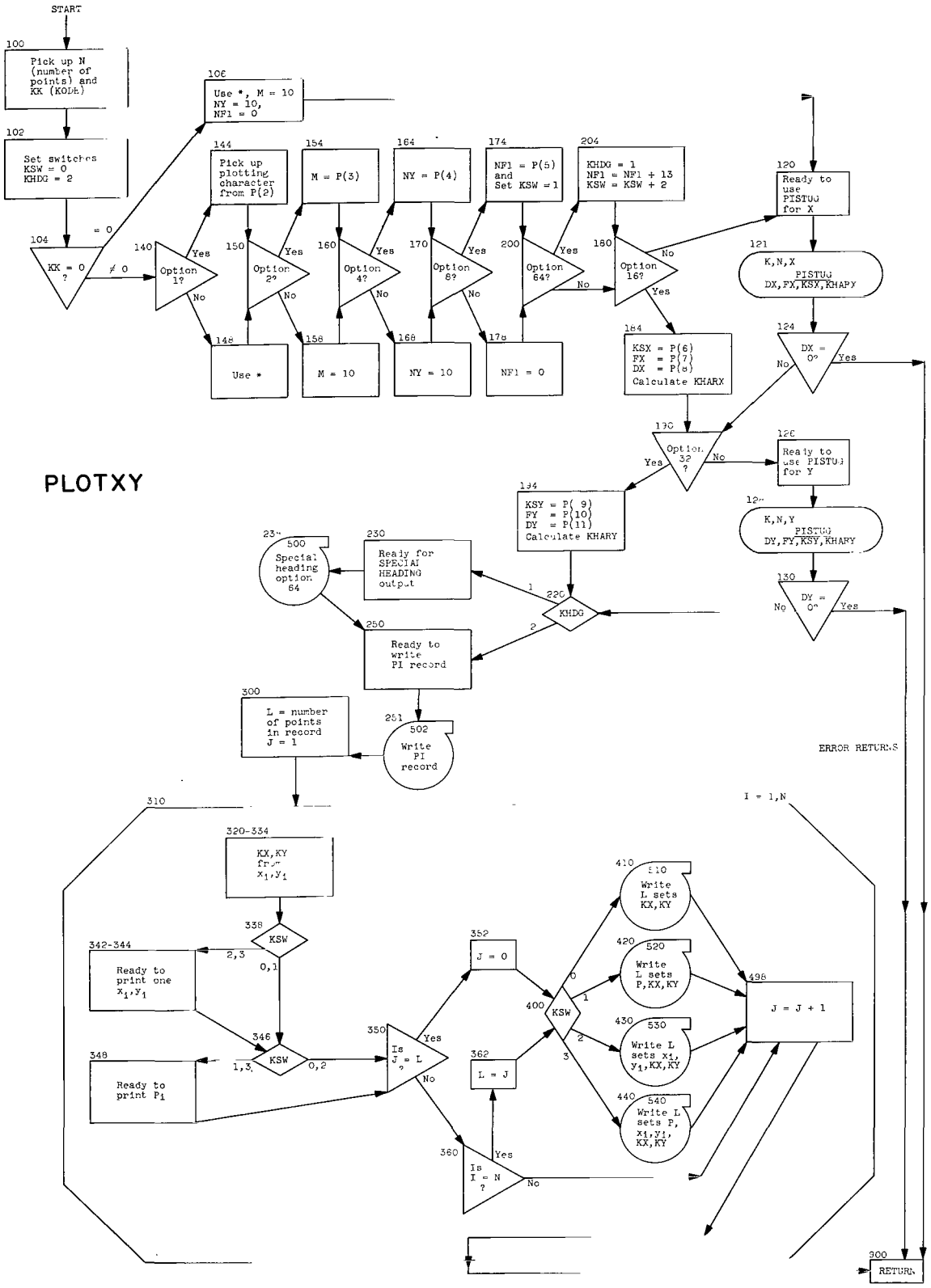
EXP(2 LOG PISTUG (FIL) (STH)

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
100	11	00072	101	12	00104	102	13	00106	103	14	00110	104	15	00114
106	16	00120	107	17	00122	108	18	00124	109	19	00126	120	20	00130
121	21	00132	122	23	00144	124	24	00153	126	25	00157	128	26	00161
130	28	00173	132	29	00202	140	30	00206	144	31	00213	146	32	00215
148	33	00216	150	34	00220	154	35	00225	156	36	00237	158	37	00240

```
      SUBROUTINE PLOTXY(X,Y,BK,P)
```

160	38	00242	164	39	00247	166	40	00261	168	41	00262	170	42	00264
174	43	00271	176	44	00303	177	45	00305	178	46	00306	200	47	00310
204	48	00315	206	49	00320	208	50	00323	180	51	00327	184	52	00334
185	53	00346	186	54	00350	187	55	00352	190	60	00437	194	61	00444
196	62	00456	198	63	00460	199	64	00462	220	69	00547	230	70	00551
232	71	00554	234	72	00557	236	73	00565	238	74	00573	250	76	00603
251	80	00643	300	82	00673	310	84	00710	320	85	00715	322	86	00732
324	87	00741	330	88	00755	332	89	00772	334	90	01001	338	91	01015
340	92	01027	342	93	01032	344	94	01035	346	95	01042	347	96	01047
348	97	01054	350	98	01060	352	99	01065	354	100	01071	360	101	01072
362	102	01100	400	103	01102	402	104	01107	410	105	01113	412	110	01134
420	111	01136	422	116	01161	430	117	01163	432	122	01210	440	123	01212
498	128	01241	900	129	01253									



PLOTXY

PLOTXY

PLOTXY - DICTIONARY

Superscripts are statement numbers in the FORTRAN listing (p. 9).
The notation $\overline{z}$ means option z is being used.

A1 The plotting character. It is either supplied[144] in P(2) by the programmer if $\overline{1}$ or specified[106,148] to be an asterisk. It is written out[251] in the INFO record.

BK The third argument in the call list of this subroutine. Its mode (floating point) differs from the mode the programmer uses (fixed point) in his CALL statement, because it is used as a Boolean variable. It is renamed[101] BKK to avoid excessive address modification.

BKK The Boolean variable name for the number KODE (the sum of the option numbers). It is "equivalenced" to KK. Branching on single bit positions of BKK, each of which represents a different option number being used, occurs.[140,150,160,170,180,190,200]

C6 An intermediate answer in the calculation of KX (the line count) from x_i[320], and KY (the print position count) from y_i.[330]

C7 C6 rounded and truncated to an integer.[322,332]

DX The scaling parameter for the x scale-factor. It is either supplied[186] in P(8) by the programmer if $\overline{16}$, or calculated[121] in PISTUG.

DY The scaling parameter for the y scale-factor. It is either supplied[198] in P(11) by the programmer if $\overline{32}$, or calculated[128] in PISTUG.

F1OX The value of 10^{KPWRX}. It is computed[234] and used[342] to prepare x_i for printing at the left of the first y grid-line if $\overline{64}$.

F1OY The value of 10^{KPWRY}. It is computed[236] and used[344] to prepare y_i for printing at the left of the first y grid-line if $\overline{64}$.

FX The scaling parameter for the first x grid-label. It is either calculated[121] in PISTUG or supplied[185] in P(7) by the programmer if $\overline{16}$.

FY The scaling parameter for the first y grid-label. It is either calculated[128] in PISTUG or supplied[196] in P(10) by the programmer if $\overline{32}$.

I The index of the loop[310-498] that processes all pairs of coordinates. I is compared[360] to N to provide a branch for the condition that L is not a factor of N.

J — The counter for the number of blocks ready to be written. It is initialized[300+1] to 1, stepped[498] by 1 after each pair of coordinates is processed, tested[350] against L (the maximum number of points in a DATA record), and reset[352] to zero when it is equal to L. It replaces[362] the value of L if the last DATA record is to be less than L blocks long. J is also used as the index of the loop within each output statement[410,420, 430,440] list. See appendix B, IV.-E.

K — A switch variable set[120] to 3 and sent[121] to PISTUG when scaling parameters for the X array are to be found; set[126] to 2 and sent[128] to PISTUG when scaling parameters for the Y array are to be found.

KBR — A manipulation[338] of KSW to create a two-way branch used[340] to control picking up X_i and Y_i for printing in the left-side field if 64.

KBR2 — A manipulation[346] of KSW to create a two-way branch used[347] to control picking up P_i if 8.

KBRM — A manipulation[400] of KSW to create a four-way switch variable used[402] to control branching to the correct one of four possible output statements.[410,420,430,440]

KDX — A fixed-point name for DX[122]. It is tested[124] against zero to recognize an error message from PISTUG.

KDY — A fixed-point name for DY[130]. It is tested[132] against zero to recognize an error message from PISTUG.

KHARX — An integer representing the range of the X array. It is always calculated, either by PLOTXY[187+4] if 16 or by PISTUG[121]. It is used[230] to compute KPWRX if 64.

KHARY — An integer representing the range of the Y array. It is always calculated, either by PLOTXY[199+4] or by PISTUG[128] if 32. It is used[232] to compute KPWRY if 64.

KHDG — A switch variable initialized[103] to 2, and set[208] to 1 if 64. It is used[220] to control branching to write a special heading over the left-side field print-out of the coordinates.

KK — The fixed-point name for KODE (the sum of the option numbers). It is "equivalenced" to BKK and tested[104] for zero.

KPWRX — The value of 3 - KHARX. If 64, it is computed,[230] used[234] to compute FlOX, and written out[238] in a special heading.

KPWRY — The value of 3 - KHARY. If 64, it is computed,[232] used[236] to compute FlOY, and written out[238] in a special heading.

KSW A basic switch variable. It is initialized[102] to zero, set[176] to 1 if 8, and increased[206] by 2 if 64; hence, it may have the value 0, 1, 2, or 3.

KBR = KSW/2 + 1 is computed[338] and used[340] to control branching to prepare X_i and Y_i for print out if 64.

KBR2 = KSW + 1 is computed[346] and used[347] to control branching to pick up an element of the P array if 8.

KBRM = KSW + 1 is computed[400] and used[402] to control branching to the proper output statement.

KSX One of three scaling parameters for the x scale. It is either calculated[121] in PISTUG or supplied[184] in P(6) by the programmer if 16. It is used[320] to compute C6, an intermediate step in the calculation of KX. It is written out[251] in the INFO record. It is the number of places the decimal point must be moved to the right in the values of OFX and ODX by the 1401 subroutine PLOT.

KSY One of three scaling parameters for the y scale. It is either calculated[128] in PISTUG or supplied[194] in P(6) by the programmer if 32. It is used[330] to compute C6, an intermediate step in the calculation of KY. It is written out[251] in the INFO record. It is the number of places the decimal point must be moved to the right in the values of OFY and ODY by the 1401 subroutine PLOT.

KX The array in which the values of KX (number of lines down) are stored[324] until they are written out in a DATA record by one of four output statements.[410,420,430,440]

KY The array in which the values of KY (number of print positions to the right) are stored[334] until they are written out in a DATA record by one of four output statements.[410,420,430,440]

L The maximum number of blocks possible in one DATA record. It is computed[300] and used[350] to test J and to write out[410,420,430,440] the DATA record. If there are less than the maximum number of blocks to be written in the last DATA record, L is set[362] equal to J before the write-out.

M The frequency of the x grid-lines. It is either supplied[154] in P(3) by the programmer if 2 or set[107,158] equal to 10. It is written out[251] in the INFO record.

N The number of points to be plotted. It is supplied[100] by the programmer in P(1) and used in both calls to PISTUG.[121,128] It is the total number of executions of the loop[310-498] that processes all points and writes out the DATA records. It is tested[360] against I to cause branching on the condition that L is not a factor of N.

NF1 The length of the first field in one block of a DATA record. The NF1 characters in a block are to be printed to the left of the first y grid-line on the same line with the point whose KX and KY values are in the block. If NF1 = 0, blanks are to be printed. NF1 is initialized[109,178] to zero, set[174] equal to 6 if 8, and increased[204] by 13 if 64. It is used[300] to compute L, the maximum number of blocks per record, and is written[251] out in the INFO record.

NY The frequency of the y grid-lines. It is either set[108,168] to 10 or supplied[164] in P(4) by the programmer if 4, and it is written out[251] in the INFO record.

ODX A number written out[251] representing the x scale-factor. It is computed[250] by multiplying the scaling parameter DX (always an integer) by 10^{-6} to put it in the form required for the INFO record.

ODY A number written out[251] representing the y scale-factor. It is computed[250+2] by multiplying the scaling parameter DY (always an integer) by 10^{-6} to put it in the form required for the INFO record.

OFX A number written out[251] representing x-scale starting-value. It is computed[250+1] by multiplying the scaling parameter FX (always an integer) by 10^{-6} to put it in the form required for the INFO record.

OFY A number written out[251] representing the y-scale starting-value. It is computed[250+3] by multiplying the scaling parameter FY (always an integer) by 10^{-6} to put it in the form required for the INFO record.

P An array name, the fourth argument in the call list of this subroutine. The number of points to be plotted must be the first element of this array and the other elements are any data required by the particular options being used. (See ref. 1, p.7)

PFX An intermediate value found[187+2] in the calculation of KHARX.

PFY An intermediate value found[199+2] in the calculation of KHARY.

POW The array that has as its i^{th} element the six BCD characters associated with X_i. It is supplied[348] in P(I+11) by the programmer if 8 and written out in the DATA record.[420,440]

RDX The true value of the x scale-factor. It is computed and used[187 et seq.] to compute KHARX if 16.

RDY The true value of the y scale-factor. It is computed and used[199 et seq.] to compute KHARY if 32.

RFX The true value of the x-scale starting-value. It is computed and used[187 et seq.] to compute KHARX if 16.

RFY The true value of the y-scale starting-value. It is computed and used[199 et seq.] to compute KHARY if 32.

X An array name, the first argument in the call list of this subroutine, containing the values of the variable to be plotted on the vertical scale. It is sent[121] to PISTUG for scaling, unless 16. The elements must be in sequence, either increasing or decreasing.

XMAX An intermediate value found[187 et seq.] during the calculation of KHARX.

XOW The array in which the transformed values of X_i are held[342] until they are written out[430,440] in the DATA record if 64.

Y An array name, the second argument in the call list of this subroutine, containing the values of the variable to be plotted on the horizontal scale. It is sent[128] to PISTUG for scaling, unless 32.

YMAX An intermediate value found[199 et seq.] during the calculation of KHARY.

YOW The array in which the transformed values of Y_i are held[344] until they are written out[430,440] in the DATA record if 64.

PLOTMY - (FORTRAN II)

Although more than one curve can be plotted on the same grid using PLOTXY, they will all use the same plotting character. When a different plotting character for each curve is desired, the programmer uses PLOTMY. The CALL statement lists the names of the arrays to be plotted; an array containing the number KODE, the number of curves, and possibly other information - depending upon the options (PLOTXY I.-D.) used and the Variation (PLOTMY I.-C.) chosen; and finally an array whose first element indicates the Variation chosen and whose other elements are any additional data required for the options being used.

PLOTMY may call an auxiliary subroutine PISTUG to compute scaling parameters, then it writes an INFO record specifying the appearance of the grid to the 1401 subroutine PLOT. This is followed by one or more packed DATA records that specify the plotting character as well as the location of each point in terms of KX (the number of lines down from the first x grid-line) and KY (the number of print positions to the right of the first y grid-line).

I. Input to this routine is supplied via the four arguments of the CALL, as follows:

CALL PLOTMY (XDOWN, YACROS, KKK, P)

A. XDOWN and YACROS are floating-point variable array names. In general, the values in both arrays must be between -10^6 and 10^6. (See appendix B,VI., for precise limitations.) The subroutine SCALE is supplied to handle arrays whose element size is not known. (SCALE should be called before calling PLOTMY.) There is no restriction on the order of the elements; however, the elements of the array XDOWN are destroyed. The minimum DIMENSIONS of these arrays must be as shown in reference 1, page 11.

B. KKK is a fixed-point variable array name, whose minimum DIMENSIONS must be either 14, or two more than twice the number of curves to be plotted, whichever is larger. The elements in this array are determined by the choice of options (I.-E.) and Variation (see I.-C.) made by the programmer.

KKK(1) is the KODE. Seven different options are offered to the programmer. Each has a number associated with it. The sum of the option numbers is the KODE, and $0 \leq \text{KODE} \leq 127$.

KKK(2) = KN, the number of curves to be plotted. If KN > 6, option 1 must be used.

KKK(3) is supplied as specified by the Variation selected by the programmer.

KKK(4), KKK(6), . . . are supplied only if option 1 is used.

KKK(5), KKK(7), . . . are supplied only if Variation III is selected.

C. Variations

Frequently a set of curves to be plotted is defined by several sets of ordinates associated with the same set of abscissas (each value of an independent variable determining one value of each of several dependent variables). Variation I is for such multiple curves, with the single set of abscissas placed in the XDOWN array and the multiple sets of ordinates in YACROS. For complete flexibility, Variation II permits the interchange of axes; that is, the single set of abscissas in YACROS, the ordinates in XDOWN. In these two Variations each curve has the same number of points (NPTS), and it is supplied to the subroutine in KKK(3).

Variation III is for general multiple curves, in which each curve may have any number of points. The number of points for the first curve is supplied in KKK(3), for the second curve in KKK(5); in general, the number of points for the J^{th} curve must be in KKK(2*J+1).

D. P is a floating-point variable array name:

P(1) = 1. for Variation I.

P(1) = 3. for Variation II.

P(1) = 5. for Variation III.

The DIMENSION and remaining elements of the P array are determined by the particular options being used (see ref. 1, p. 11).

E. Options. The sample plots in appendix A illustrate the effects of using the various options.

NUMBER	OPTION
1	Choice of plotting characters. - The plotting characters *, +, O, x, =, C will be used for the first six curves unless the programmer uses this option. When he does, he must supply the desired plotting characters (which may be any FORTRAN character except the minus sign) in the KKK array. The octal representation of the plotting character for the first curve must be in the leftmost 6 bits of KKK(4), for the second curve in KKK(6) and, in general, for the J^{th} curve in KKK(2*J+2). If more than six curves are to be plotted, the programmer must use option 1, and must supply all the plotting characters.
2	Choice of x grid-line frequency. - An x grid-line will be printed every 10 line-spaces below the first one unless the programmer uses this option. When he does, the desired frequency (M) of x grid-lines must be supplied to PLOTMY in P(3). If M = 0, only the first and final x grid-lines will be printed.
4	Choice of y grid-line frequency. - A y grid-line will be printed every 10 print positions to the right of the first unless this option is used. If it is, the desired frequency (N) of y grid-lines must be supplied in P(4). If N = 0, only the first (leftmost) y grid-line will be printed.
16	Choice of x scale. - PLOTMY will supply the three scaling parameters that specify the x-scale starting-value, the scale-factor (value of one line-space), and the x grid-labels unless this option is used. If it is used, the programmer must specify his own scale by supplying the three parameters KSX, FX, and DX in P(6), P(7), and P(8), respectively. These parameters are defined by: FX is the desired starting-value for the x scale multiplied by 10^U. DX is the desired scale-factor for one line-space multiplied by 10^U. DX must be positive. U must be chosen so that FX and DX are integers and $0 < U < 6$. KSX is equal to 6 - U.

NUMBER	OPTION
32	Choice of y scale. - PLOTMY will supply the three scaling parameters that specify the y-scale starting-value, the scale-factor (value of one print position) and the y grid-labels unless the programmer uses this option. If he does, he must specify his own scale by supplying the three parameters, KSY, FY, and DY in P(9), P(10), and P(11), respectively. These are defined by: FY is the desired starting-value multiplied by 10^V. DY is the desired scale-factor for one print position multiplied by 10^V. V must be chosen so that FY and DY are integers and $0 \leq V \leq 6$. KSY is equal to 6 - V.
64	Choice of printing x-coordinate at the left of the plot. - The 29 print positions to the left of the first y grid-line are usually blank except for x grid-labels every tenth line in positions 20 to 28. If option 64 is used, the x-ordinate of each point is printed immediately to the left[1] of the grid-labels on the same line as the point.
8	Choice of printing message at left of plot. - The 29 print positions to the left of the first y grid-line are usually blank except for x grid-labels every tenth line in positions 20 to 28. If option 8 is used, the 6 BCD characters in P(12) are printed immediately to the left[1] of the x grid-labels on the same line on which the first point is printed, the six characters in P(13) are printed with the second point, etc. Specifically, the 6 characters in P(I+11) are printed on the same line on which the I^{th} value of the XDOWN array is plotted.

[1]If option 8 and 64 are used together, the BCD characters will print to the left of the ordinate.

II. Output from this routine consists of one INFO record and one or more packed DATA records. Sample tape records appear in appendix A. As a result of the execution of this routine, the elements of the array XDOWN are destroyed.

A. INFO record.

The contents of the INFO record are the information needed by the 1401 subroutine PLOT to determine the construction of the grid, the construction of the grid-labels, and the format of the DATA records that follow. Specifically, the contents of the INFO record consist of the two characters, PI, followed by 10 fields, whose contents are:

(Field width is shown in parentheses.)

a. (1) This character is always a minus sign, and signifies that the plotting characters to be associated with each point will be found in the DATA records.

b. (1) This digit is the decimal-point-shift code, KSX, for OFX and ODX. (Items d. and f. in this list.)

c. (1) This digit is the decimal-point-shift code, KSY, for OFY and ODY. (Items e. and g. in this list.)

d. (9) The number (OFX) in this field is in the form ±0.xxxxxx and represents the x-scale starting-value.

e. (9) The number (OFY) in this field is in the form ±0.xxxxxx and represents the y-scale starting-value.

f. (9) The number (ODX) in this field is in the form ±0.xxxxxx and represents the x scale-factor (value of one line-space).

g. (9) The number (ODY) in this field is in the form ±0.xxxxxx and represents the y scale-factor (value of one print position).

h. (3) This integer is the frequency, M, of the x grid-lines.

i. (3) This integer is the frequency, N, of the y grid-lines.

j. (3) This integer specifies the length, NF, of the first field in each block of information in the following DATA records. NF may be zero.

B. DATA records.

The contents of a DATA record (after the first two characters PD) are successive blocks of information needed by the 1401 subroutine PLOT to position one or more plotting characters in the same line and print the left-side field of that line. Within one record, only complete blocks are permitted, and each block is terminated by an end-of-block character or delimiter - the letter E. One record may hold as many as 16 blocks. Within one block, the first NF characters are those to be printed in the left-side field (to the left of the first y grid-line) unless NF = 0; in this case, blanks are to be generated by the 1401. The next three characters are KX, the number of line-spaces below the first x grid-line at which all the points in that block are to be printed. The next three characters are a value of KY, the number of print positions to the right of the first y grid-line, in which a point is to be printed. The plotting character itself follows KY. Since additional points may fall on the same line, further values of KY, each with its associated plotting character, may follow the first. The plotting character for the last value of KY in the block is followed by the delimiter E.

III. After initializing switches, the routine inspects the input variable KODE, bit by binary bit, branching on each of the seven bit positions to set necessary output switches, and either supplies standard values or picks up programmer-provided specifications from the KKK and P arrays. It branches also on Variation number for additional initializing. It may call an auxiliary routine PISTUG (p. 44) for scaling either or both coordinate arrays. The INFO record (II.-A.) is written, and the routine constructs the output FORMAT statement for the packed DATA records.

The XDOWN array is searched for a minimum value and the corresponding value of KX (number of lines down) is computed and stored in an output buffer area. KY's (the number of print positions across) are computed from corresponding YACROS values; each is sent to the buffer with its associated plotting character; the last is followed by the delimiter. The minimum element of the XDOWN array is then overwritten with a large number, and a new search for a minimum is made. Whenever the output buffer area is filled, a DATA record is written. This process continues until all points have been processed; control is then returned to the calling program.

```
      SUBROUTINE PLOTMY(X,Y,K,P)

      SUBROUTINE PLOTMY(X,Y,K,P)
C
      EQUIVALENCE (BKK,KK),(A,KPC),(FUSE(5),FUSEM),(B,KB)
      DIMENSION X(1),Y(1),K(1),P(1),A(6),KPC(6),FUSE(21),FUSEM(16),B(64)
     1,KB(64)
C
C                                          RENAME INPUT
  100 KK=K(1)
  101 KN=K(2)
  102 NPTS=K(3)
C                                          SET UP FORMAT STATEMENT FOR DATA OUTPUT
B 103 FUSE(1)=740230472473
B 104 FUSE(2)= 020074606060
B 105 FUSE(3) = 606060606060
B 106 FUSE(4) = 603103736060
  107 DO 108 I=1,16
  108 FUSEM(I) = FUSE(3)
  109 FUSE(21)= 6H A1))
C                                          INITIALIZE SWITCHES,COUNTERS,CONSTANTS
  110 KEXIT=1
  111 KTL=1
  112 KSW64=1
  113 KSW8=1
  114 KSW1=1
  116 KSW3=1
  117 KWADD=2
  118 KLADD=4
  119 NF1=0
      M = 10
      NY=10
C
C                                          OPTION 1'GET STD. P.C.S OR GO TO 150
B 120 IF(BKK*1000000)122,122,150
  122 A(1)= 1H*
      A(2)= 1H+
      A(3)= 1HO
      A(4)=1HX
      A(5)= 1H=
      A(6)= 1HO
      K(4)= KPC(1)
      K(6)= KPC(2)
      K(8)= KPC(3)
      K(10)=KPC(4)
      K(12)=KPC(5)
      K(14)=KPC(6)
C                                          GO TO 200 IF KODE IS ZERO
  140 IF(KK) 150,200,150
C                                          INITIALIZE FOR OPTION 64 OR GO TO 160
B 150 IF(BKK*100000000) 160,160,152
  152 KSW64=2
B 154 FUSE(3)=602607330073
  156 NF1=7
  158 KLADD=KLADD+7
  159 KWADD=KWADD+1
C
C
```

```
      SUBROUTINE PLOTMY(X,Y,K,P)
C                                          INITIALIZE FOR OPTION 8 OR GO TO 170
B 160 IF(BKK*10000000) 170,170,162
  162 KSW8=2
B 164 FUSE(2)=020074210673
  166 NF1=NF1+6
  168 KLADD=KLADD+6
  169 KWADD=KWADD+1
C                                          IF  OPTION  2, USE P(3)
B 170 IF(BKK*2000000) 180,180,172
  172 M=P(3)+SIGNF(.5,P(3))
C                                          IF OPTION 4, USE P(4)
B 180 IF(BKK*4000000) 200,200,182
  182 NY=P(4)+SIGNF(.5,P(4))
C                                          INITIALIZE FOR V.I OR GO TO 210
  200 IF (P(1)-2.5) 202,210,210
  202 KTL=KN
  204 KSW1=2
  208 GO TO 228
C                                          INITIALIZE FOR V.III OR GO TO 220
  210 IF(P(1)-4.) 220,220,214
  214 NPTST=0
  216 DO 218 I=1,KN
  218 NPTST=NPTST+K(2*I+1)
  219 GO TO 230
C                                          INITIALIZE FOR V.II
  220 KTIMES=KN-1
  221 DO 225 I=1,KTIMES
  222 MM=NPTS*I
 1222 K(2*I+3) = NPTS
  223 DO 225 II=1,NPTS
  224 L=MM+II
  225 Y(L)=Y(II)
  226 KSW3=2
C                                          FOR V.I  AND V.II
  228 NPTST=KN*NPTS
C                                          IF OPTION 16,USE P(6-8)
B 230 IF(BKK*20000000)250,250,231
  231 KSX=P(6)+SIGNF(.5,P(6))
  232 FX=P(7)
  233 DX=P(8)
  234 RFX=FX*(10.**(KSX-6))
  235 RDX=DX*(10.**(KSX-6))
  236 PFX=RFX+100.*RDX
  237 XMAX=MAX1F(ABSF(RFX),ABSF(PFX))
  238 KHARX=XINTF(LOGF(XMAX)/2.302585+40.)-40
C                                          IF OPTION 32, USE P(9-11)
B 240 IF(BKK*40000000) 270,270,242
  242 KSY=P(9)+SIGNF(.5,P(9))
  244 FY=P(10)
  246 DY=P(11)
  248 GO TO 300
C                                          NSX DIFFERS FOR V.I
  250 GO TO (256,252),KSW1
  252 NSX=NPTS
  254 GO TO 260
  256 NSX=NPTST
```

```
      SUBROUTINE PLOTMY(X,Y,K,P)
C                                                SCALE X AND TO 240 OR ERROR EXIT
  260 CALL PISTUG( 1 ,NSX,X,DX,FX,KSX,KHARX)
  262 KDX=DX
  264 IF(KDX) 240,900,240
C                                                NSY DIFFERS FOR V.II
  270 GO TO (276,272),KSW3
  272 NSY=NPTS
  274 GO TO 280
  276 NSY=NPTST
C                                                SCALE Y AND TO 300 OR ERROR EXIT
  280 CALL PISTUG (2,NSY,Y,DY,FY,KSY,KHARY)
  282 KDY=DY
  284 IF(KDY) 300,900,300
C                                                WRITE OPT64 HEADING OR GO TO 310
  300 GO TO (310,302),KSW64
  302 KPWRX=3-KHARX
  304 F10X=10.**KPWRX
  306 WRITE OUTPUT TAPE 6,500,KPWRX
  500 FORMAT(2HPT/2HPT/2HPT,15X,3HX*E,I2)
C                                                WRITE THE INFO RECORD
  310 RD= 5.E-7
  311 SFT=1.E-6
  312 ODX=DX*SFT+SIGNF(RD,DX)
  313 OFX=FX*SFT+SIGNF(RD,FX)
  314 ODY=DY*SFT+SIGNF(RD,DY)
  315 OFY=FY*SFT+SIGNF(RD,FY)
  316 XMF=(10.**(6-KSX))/DX
  317 YMF=(10.**(6-KSY))/DY
  318 WRITE OUTPUT TAPE 6,502,KSX,KSY,OFX,OFY,ODX,ODY,M,NY,NF1
C
  502 FORMAT(3HP1-,2I1,4F9.6,3I3)
C
C                                                INITIALIZE FOR DATA OUTPUT
C
C                                                  FINISH FORMAT STATEMENT
  320 DO 322 I = 1,KTL
B 322 FUSEM(I) = 310373210173
C                                                  COMPUTE BUFFER CONSTANTS AND SET CTRS
  330 KLC=KLADD+4*KTL
  332 NBR=130/KLC
  334 KW =KWADD+2*KTL
  336 NBW=KW*NBR
  337 J = 1
  338 KBW=1
  339 KBBC=1
  340 KLAST=2*KN+1
C                                                  ILIM DIFFERS FOR V.I
  342 GO TO (348,344),KSW1
  344 ILIM=NPTS
  346 GO TO 350
  348 ILIM=NPTST
C                                                  READY TO FIND SMALLEST X
  350 XMIN=1.E15
  352 IMIN=1
C
C
```

```
      SUBROUTINE PLOTMY(X,Y,K,P)
C                                          FIND SMALLEST X AND ITS SUBSCRIPT IMIN
  360 DO 368 I=1,ILIM
  362 IF(XMIN-X(I)) 368,368,364
  364 XMIN=X(I)
  366 IMIN=I
  368 CONTINUE
C                                          IF OPTION 8,MOVE P(IMIN+11) TO BUFFER
  370 GO TO (380,372),KSW8
  372 B(KBW)=P(IMIN+11)
  374 KBW=KBW+1
C                                          IF OPTION 64,MOVE X(IMIN) TO BUFFER
  380 GO TO (390,382),KSW64
  382 B(KBW)=F10X*XMIN
  384 KBW=KBW+1
C                                          COMPUTE KX AND MOVE TO BUFFER
  390 C6  =XMIN*XMF
  392 C7  =INTF(C6+SIGNF(.5,C6))
  394 KB(KBW)=(DX*C7-FX)/DX+.5
  396 KBW=KBW+1
C                                          OVERWRITE SMALLEST X
  398 X(IMIN) = 1.E15
C
C                                          ASSOCIATED KYS AND P.C.S GO TO BUFFER
C
  400 DO 446 IM=1,KTL
  402 LL=IMIN+(IM-1)*NPTS
  404 YS=Y(LL)
  406 C6=YS*YMF
  408 C7=INTF(C6+SIGNF(.5,C6))
  410 KB(KBW)= (DY*C7-FY)/DY+.5
  412 KBW=KBW+1
C                                            IF NOT V.I,SEARCH FOR P.C.
  420 GO TO (422,440),KSW1
  422 IK=0
  424 DO 430 IL=3,KLAST,2
  426 IK=IK+K(IL)
  428 IF (IK-IMIN) 430,436,436
  430 CONTINUE
C                                            IF NOT FOUND,ERROR EXIT
  432 WRITE OUTPUT TAPE 6,600,IMIN
  600 FORMAT  (2HPL, 42H NPTS IS MISSING OR WRONG IN KARRAY.IMIN= ,I3)
  434 GO TO 900
C                                            P.C. FOUND FOR V.II AND V.III
  436 KPCI = K(IL+1)
  438 GO TO 442
C                                            P.C. FOR V.I
  440 KPCI = K(2*IM+2)
C                                            MOVE PLOT CHAR. TO BUFFER
  442 KB (KBW)=KPC I
  444 KBW=KBW+1
  446 CONTINUE
C                                          DELIMITER TO BUFFER
  448 B(KBW)=1HE
  449 KBW=KBW+1
C                                          GO TO 460 IF BUFFER FILLED
  450 IF(NBW-KBW) 460,452,452
```

```
      SUBROUTINE PLOTMY(X,Y,K,P)

C                                                  GO TO 470 IF ALL POINTS COMPUTED
  452 IF(ILIM-J) 470,470,454
C                                                  STEP COUNTERS AND GO BACK TO 350
  454 J=J+1
  456 KBBC=KBBC+1
  458 GO TO 350
C                                                  WRITE A DATA RECORD
  460 KBL=KW*KBBC
  462 WRITE OUTPUT TAPE 6,FUSE,(B(N),N=1,KBL)
C                                                  RESET BUFFER COUNTERS
  464 KBW=1
  466 KBBC=0
C                                                  GO TO 452 OR QUIT
  468 GO TO (452,900), KEXIT
C                                                  IF ALL RECORDS WRITTEN,QUIT
  470 IF(KBBC) 471,900,471
C                                                  IF NOT,SET EXIT TO QUIT AND GO TO 460
  471 KEXIT=2
  472 GO TO 460
C
  900 RETURN
      END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

SUBROUTINE PLOTMY(X,Y,K,P)

STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
1125	02145	32561	77461

STORAGE LOCATIONS FOR VARIABLES APPEARING IN DIMENSION AND EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
A	1123	02143	BKK	1124	02144	B	1096	02110	FUSEM	1113	02131	FUSE	1117	02135
KB	1096	02110	KK	1124	02144	KPC	1123	02143						

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENT

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
C6	1032	02010	C7	1031	02007	DX	1030	02006	DY	1029	02005	F10X	1028	02004
FX	1027	02003	FY	1026	02002	II	1025	02001	IK	1024	02000	ILIM	1023	01777
IL	1022	01776	IMIN	1021	01775	IM	1020	01774	I	1019	01773	J	1018	01772
KBBC	1017	01771	KBL	1016	01770	KBW	1015	01767	KDX	1014	01766	KDY	1013	01765
KEXIT	1012	01764	KHARX	1011	01763	KHARY	1010	01762	KLADD	1009	01761	KLAST	1008	01760
KLC	1007	01757	KN	1006	01756	KPCI	1005	01755	KPWRX	1004	01754	KSW1	1003	01753
KSW3	1002	01752	KSW64	1001	01751	KSW8	1000	01750	KSX	999	01747	KSY	998	01746
KTIMES	997	01745	KTL	996	01744	KWADD	995	01743	KW	994	01742	LL	993	01741
L	992	01740	MM	991	01737	M	990	01736	NBR	989	01735	NBW	988	01734
NF1	987	01733	NPTS	986	01732	NPTST	985	01731	NSX	984	01730	NSY	983	01727
NY	982	01726	ODX	981	01725	ODY	980	01724	OFX	979	01723	OFY	978	01722
PFX	977	01721	RD	976	01720	RDX	975	01717	RFX	974	01716	SFT	973	01715
XMAX	972	01714	XMF	971	01713	XMIN	970	01712	YMF	969	01711	YS	968	01710

SYMBOLS AND LOCATIONS FOR SOURCE PROGRAM FORMAT STATEMENTS

	EFN	LOC		EFN	LOC		EFN	LOC		EFN	LOC		EFN	LOC
8)FK	500	01657	8)FM	502	01652	8)IO	600	01646						

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
1)	951	01667	2)	869	01545	3)	895	01577	6)	919	01627	9)	944	01660
8)	879	01557	C)G2	954	01672	C)G3	955	01673	C)G5	956	01674	C)G6	957	01675
C)G7	958	01676	C)G8	959	01677	C)G9	960	01700	C)GA	961	01701	C)GB	962	01702
C)GC	963	01703	C)GD	964	01704	C)GF	965	01705	C)200	966	01706	C)201	967	01707
D)10J	280	00430	D)11H	653	01215	D)11S	784	01420	D)11T	788	01424	D)11V	814	01456
D)121	828	01474	D)126	861	01535	D)206	186	00272	D)20C	239	00357	D)20J	282	00432
D)20N	316	00474	D)200	322	00502	D)21K	706	01302	D)30N	315	00473	D)300	321	00501
D)408	207	00317	D)410	444	00674	D)526	860	01534	E)1H	655	01217	E)1J	668	01234
E)1L	746	01352	E)20	820	01464	E)326	863	01537						

LOCATIONS OF NAMES IN TRANSFER VECTOR

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
EXP(2	0	00000	LOG	1	00001	PISTUG	2	00002	(FIL)	4	00004	(STH)	3	00003

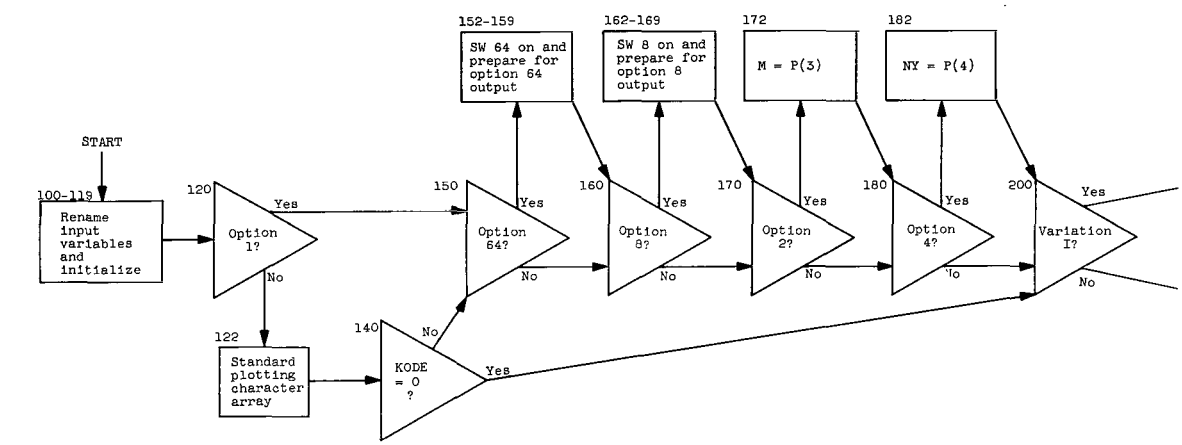
ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

SUBROUTINE PLOTMY(X,Y,K,P)

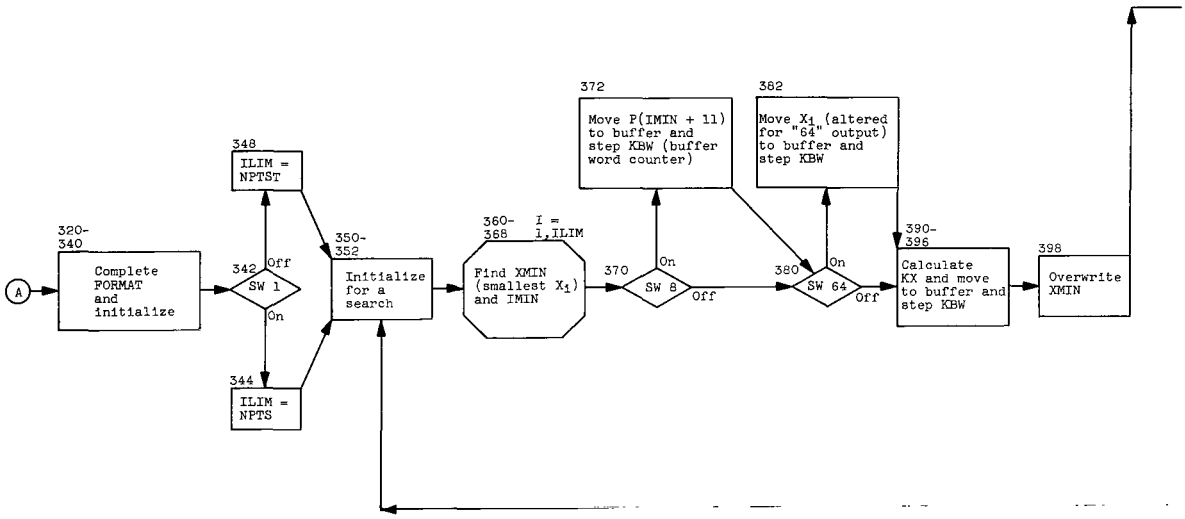
EXP(2 LOG PISTUG (FIL) (STH)

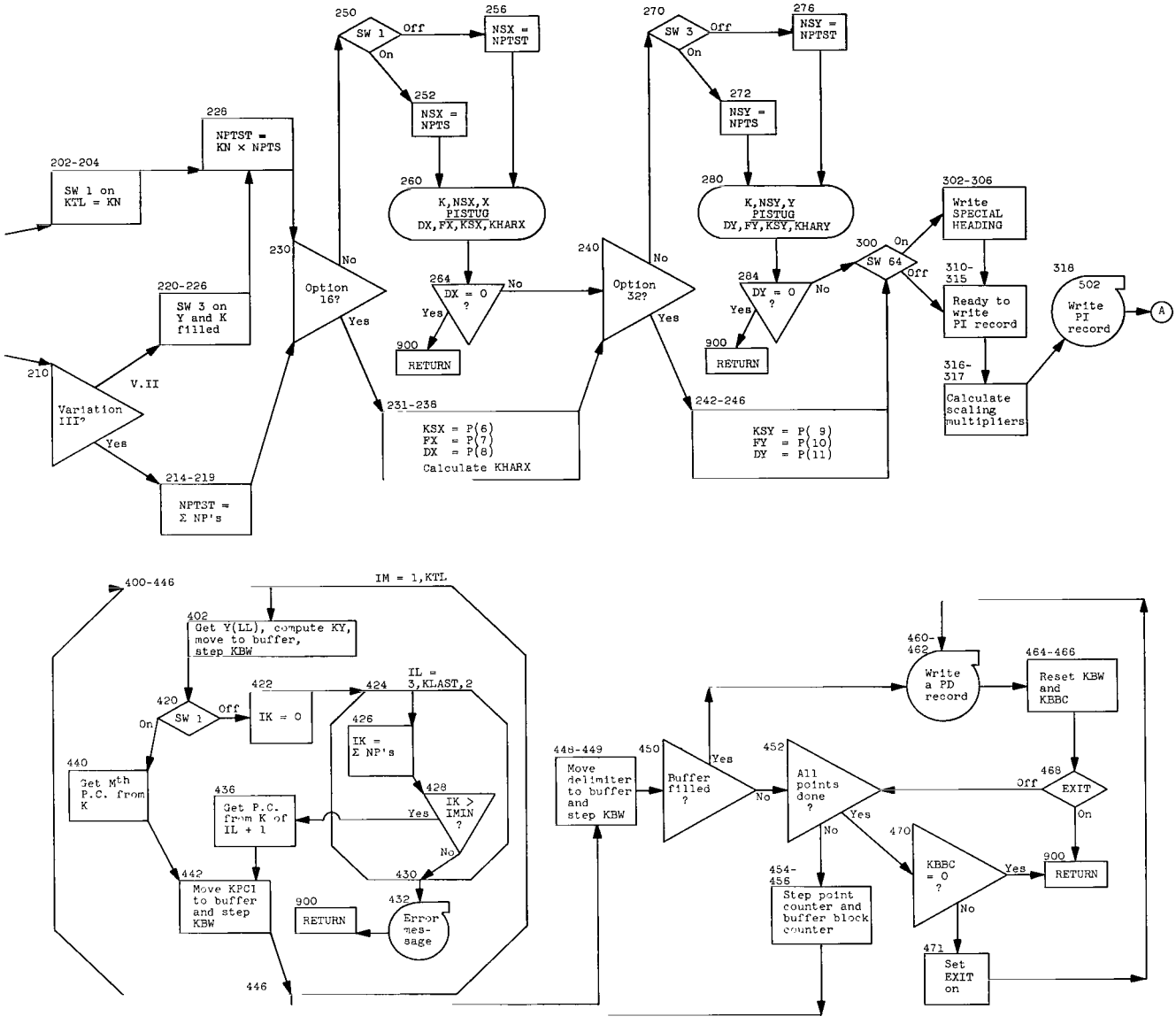
EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
100	8	00123	101	9	00125	102	10	00127	103	11	00131	104	12	00133
105	13	00135	106	14	00137	107	15	00141	108	16	00142	109	17	00146
110	18	00150	111	19	00154	112	20	00156	113	21	00162	114	22	00166
116	23	00172	117	24	00176	118	25	00200	119	26	00202	120	29	00210
122	30	00214	140	42	00244	150	43	00246	152	44	00253	154	45	00257
156	46	00261	158	47	00263	159	48	00266	160	49	00273	162	50	00300
164	51	00304	166	52	00306	168	53	00311	169	54	00314	170	55	00320
172	56	00325	180	57	00337	182	58	00344	200	59	00360	202	60	00364
204	61	00366	208	62	00372	210	63	00373	214	64	00400	216	65	00402
218	66	00407	219	67	00414	220	68	00415	221	69	00420	222	70	00433
1222	71	00437	223	72	00441	224	73	00446	225	74	00453	226	75	00466
228	76	00475	230	77	00503	231	78	00510	232	79	00522	233	80	00524
234	81	00526	235	82	00542	236	83	00556	237	84	00562	238	85	00575
240	86	00613	242	87	00620	244	88	00632	246	89	00634	248	90	00636
250	91	00637	252	92	00642	254	93	00644	256	94	00645	260	95	00647
262	97	00661	264	98	00670	270	99	00675	272	100	00700	274	101	00702
276	102	00703	280	103	00705	282	105	00721	284	106	00730	300	107	00732
302	108	00735	304	109	00740	306	110	00746	310	112	00754	311	113	00756
312	114	00760	313	115	00770	314	116	01000	315	117	01010	316	118	01020
317	119	01032	318	120	01044	320	122	01072	322	123	01075	330	124	01101
332	125	01106	334	126	01114	336	127	01121	337	128	01125	338	129	01127
339	130	01133	340	131	01135	342	132	01142	344	133	01145	346	134	01147
348	135	01150	350	136	01152	352	137	01154	360	138	01160	362	139	01165
364	140	01172	366	141	01174	368	142	01200	370	143	01204	372	144	01207
374	145	01211	380	146	01221	382	147	01224	384	148	01227	390	149	01235
392	150	01240	394	151	01247	396	152	01263	398	153	01270	400	154	01272
402	155	01303	404	156	01314	406	157	01316	408	158	01321	410	159	01330
412	160	01344	420	161	01350	422	162	01353	424	163	01355	426	164	01361
428	165	01364	430	166	01377	432	167	01402	434	169	01414	436	170	01415
438	171	01417	440	172	01421	442	173	01425	444	174	01427	446	175	01433
448	176	01442	449	177	01444	450	178	01451	452	179	01457	454	180	01465
456	181	01470	458	182	01473	460	183	01475	462	184	01501	464	189	01516
466	190	01521	468	191	01523	470	192	01526	471	193	01530	472	194	01533
900	195	01541												


START
100-119
Rename input variables and initialize
120
Option 1?
Yes
No
122
Standard plotting character array
140
KODE = 0 ?
No
Yes
150
Option 64?
Yes
No
152-159
SW 64 on and prepare for option 64 output
160
Option 8?
Yes
No
162-169
SW 8 on and prepare for option 8 output
170
Option 2?
Yes
No
172
M = P(3)
180
Option 4?
Yes
No
182
NY = P(4)
200
Variation I?
Yes
No

PLOTMY


A
320-340
Complete FORMAT and initialize
342
SW 1
Off
On
348
ILIM = NPTST
344
ILIM = NPTS
350-352
Initialize for a search
360-368
I = 1,ILIM
Find XMIN (smallest X_1) and IMIN
370
SW 8
On
Off
372
Move P(IMIN + 11) to buffer and step KBW (buffer word counter)
380
SW 64
On
Off
382
Move X_1 (altered for "64" output) to buffer and step KBW
390-396
Calculate KX and move to buffer and step KBW
398
Overwrite XMIN


202-204
SW 1 on
KTL = KN
226
NPTST =
KN × NPTS
220-226
SW 3 on
Y and K
filled
210
Variation
III?
V.II
Yes
214-219
NPTST =
Σ NP's
230
Option
16?
No
Yes
250
SW 1
Off
On
256
NSX =
NPTST
252
NSX =
NPTS
260
K,NSX,X
PISTUG
DX,FX,KSX,KHARX
264
DX = 0
?
Yes
No
900
RETURN
231-238
KSX = P(6)
FX = P(7)
DX = P(8)
Calculate KHARX
240
Option
32?
No
Yes
270
SW 3
Off
On
276
NSY =
NPTST
272
NSY =
NPTS
280
K,NSY,Y
PISTUG
DY,FY,KSY,KHARY
284
DY = 0
?
Yes
No
900
RETURN
242-246
KSY = P(9)
FY = P(10)
DY = P(11)
300
SW 64
On
Off
302-306
Write
SPECIAL
HEADING
310-
315
Ready to
write
PI record
316-
317
Calculate
scaling
multipliers
318
502
Write
PI
record
A
400-446
IM = 1,KTL
402
Get Y(LL), compute KY,
move to buffer,
step KBW
420
SW 1
On
Off
422
IK = 0
424
IL =
3,KLAST,2
426
IK =
Σ NP's
428
IK >
IMIN
?
Yes
No
430
432
Error
mes-
sage
900
RETURN
436
Get P.C.
from K of
IL + 1
440
Get Mth
P.C. from
K
442
Move KPC1
to buffer
and step
KBW
446
448-449
Move
delimiter
to buffer
and
step KBW
450
Buffer
filled
?
Yes
No
452
All
points
done
?
Yes
No
454-
456
Step point
counter and
buffer block
counter
460-
462
Write
a PD
record
464-466
Reset KBW
and
KBBC
468
EXIT
Off
On
900
RETURN
470
KBBC
= 0
?
Yes
No
471
Set
EXIT
on

PLOTMY

Superscripts are statement numbers in the FORTRAN listing (p. 26).
The notation $\underline{z}$ means option z is being used.

A — An array that holds the six standard plotting characters. It is equivalenced to KPC. The array is set up as A and used as KPC[122] et seq unless $\underline{1}$.

B — The array name of the output buffer area (or buffer) in which DATA records are assembled. It is equivalenced to KB. The name B is used[372,382,448] to move the left-side field characters from the P array if $\underline{8}$, x_i if $\underline{64}$, and the delimiter E (end-of-block character). The B array is written out[462] into the DATA record.

BKK — The Boolean variable name for KODE (the sum of the option numbers). It is "equivalenced" to KK. Branching on single bit positions, each of which represents a different option number being used, occurs.[120,150, 160,170,180,230,240]

C6 — An intermediate answer in the calculation of KX[390] from x_i and KY[406] from y_i.

C7 — C6 rounded and truncated[392,408] to an integer.

DX — The scaling parameter for the x scale-factor. It is either supplied[233] in P(8) by the programmer if $\underline{16}$ or calculated[260] in PISTUG. It is used to compute XMF[316] and each value of KX.[394] If $\underline{16}$, it is used[235] to compute RDX. See also KDX, ODX.

DY — The scaling parameter for the y scale-factor. It is either supplied[246] in P(11) by the programmer if $\underline{32}$, or calculated[280] in PISTUG. It is used to compute YMF[317] and each value of KY.[410] See also KDY, ODY.

F10X — The value[304] of 10^{KPWRX}. It is used to prepare[382] X_i for printing at the left of the first y grid-line if $\underline{64}$.

FUSE — The array (DIMENSION is 21) containing the FORMAT statement used[462] to write the DATA records. It is initialized[103-109] to:

(2HPD,	20(		I3,		· · ·	A1))
1	2	3	4	5		21

The second word is replaced[164] by 20(A6, if $\underline{8}$, the third is replaced[154] by F7.0, if $\underline{64}$. KTL words (beginning with the fifth) are replaced[320-322] by I3,A1,, using the array FUSEM, which is "equivalenced" to FUSE(5) for convenience.

FUSEM An array (DIMENSION is 16) "equivalenced" to FUSE(5) for convenience. It is initialized[107-108] to all blanks, and KTL elements are replaced[320-322] by [I3,A1].

FX The scaling parameter for the first x grid-label. It is either calculated[260] in PISTUG or supplied[232] in P(7) by the programmer if 16. It is used[313,394] to compute OFX and all values of KX. If 16, it is used[234] to compute RFX.

FY The scaling parameter for the first y grid-label. It is either calculated[280] in PISTUG or supplied[244] in P(10) by the programmer if 32. It is used[315,410] to compute OFY and all values of KY.

I The index of:

(a) The loop[107-108] that initializes FUSEM.

(b) The loop[216-218] that computes the total number of points in Variation III.

(c) The outer loop[221-225] that initializes the Y array in Variation II.

(d) The loop[320-322] that completes FUSE.

(e) The loop[360-368] that searches for the minimum value in the X array.

II The index of the inner loop[223-225] that initializes the Y array in Variation II.

IK A counter initialized[422] to zero and used in the search loop[424-430] that locates the correct plotting character for Variations II and III.

IL The index of the loop[424-430] that locates the correct plotting character for Variations II and III.

ILIM The number of values of X_i to be processed. It is the number of times the loop[360-368] is executed to find a value of XMIN. ILIM is set[344] equal to NPTS for Variation I, otherwise[348] to NPTST. It is also compared to J to cause branching[452] when all points have been processed.

IM The index of the loop[400-446] that produces all KY's and their associated plotting characters for one value of the X array.

IMIN The subscript of the minimum value in the X array after the search loop[360-368] for XMIN is completed. (It is initialized[352] to 1.) It is used[402] to locate the value or values of Y_I associated with the X_I. For Variations II and III, IMIN is used in the search loop[424-430] that locates the correct plotting character. IMIN is written out[432] in an error message if the plotting character is not found.

J A point counter. It is initialized[337] to 1, increased[454] by 1, and tested[452] for equality to ILIM to cause branching when all X_I's have been processed.

K An array name, the third argument in the call list of this subroutine. The first three elements are always supplied by the programmer: the first is KODE (see I.-B.); the second is KN (the number of curves); the third is a number of points to be plotted, as specified by the Variation selected (ref. 1, p. 11). Succeeding even-numbered elements of K are the plotting characters, supplied by the programmer if I or by this subroutine[122+6] et seq. Succeeding odd-numbered elements are:

(a) Blank if Variation I is selected.

(b) Set[1222] equal to K(3) if Variation II is selected.

(c) Supplied by the programmer if Variation III is selected and used[216-218] to compute NPTST.

KB The output buffer area (or buffer) in which DATA records are assembled. KB is "equivalenced" to B. The name KB is used to move[394,410,442] KX, KY, and the plotting character, KPCI.

KBBC The buffer block counter. It is initialized[339] to 1, stepped[456] by 1 each time the point counter J is stepped, used[460] to compute KBL (the length of the output array to be written from the buffer), and reset[466] to zero each time the buffer is emptied. It is also tested[470] against zero as a simple way of recognizing the special case when NBR (the number of blocks in a record) is a factor of ILIM (the total number of points).

KBL The length of the array to be written[462] out of the buffer. It is the product[460] of KW (the number of words in a buffer block) and KBBC (the current total in the buffer block counter).

KBW The buffer word counter. It is used as a subscript to move words into the buffer. Initialized[338] to 1, it is stepped[374,384,396,412,444,449] by 1 each time a word is sent to the buffer, compared[450] to NBW to see if the buffer is filled, and reset[464] to 1 when the buffer is emptied.

KDX A fixed-point name[262] for DX. It is tested[264] against zero to identify error message from PISTUG.

KDY A fixed-point name[282] for DY. It is tested[284] against zero to identify error message from PISTUG.

KEXIT A two-way switch variable. Initialized[110] to 1, its normal sequencing permits the problem to continue. When all points have been calculated, it is set[471] to branch[468] to the RETURN of PLOTMY.

KHARX An integer representing the range of the X array. It is calculated by PISTUG[260] or by PLOTMY[238] if 16. It is used[302] to compute KPWRX if 64.

KHARY An integer representing the range of the Y array. It is calculated[280] by PISTUG for use by PLOTXY and is never used by PLOTMY.

KK The fixed-point name for KODE. It is moved[100] from K(1) to permit "equivalencing" it to BKK for option branching. As KK, it is tested[140] for zero.

KLADD The number of characters, exclusive of KY's and plotting characters, in one block of a DATA record. It is initialized[118] to 4 (three for KX, one for the delimiter), increased by six[168] if 8, and by seven[158] if 64. The total is then used[330] to compute KLC (the total number of characters in one block of a DATA record).

KLAST The subscript of the second last position of the K array. It is computed[340] and used as the limit of the search loop[424-430] that locates the address of the correct plotting character for Variations II and III.

KLC The total number of characters in one block of a DATA record. It is the sum of KLADD and four times KTL. It is computed[330] and used[332] to compute NBR, the total number of blocks possible in one DATA record (of no more than 132 characters).

KN The number of curves to be plotted. It is always supplied[101] by the programmer in K(2), and:

(a) It is used[340] to compute KLAST.

(b) It replaces[202] KTL for Variation I.

(c) It is used[220] to compute KTIMES for Variation I.

(d) It is used[228] to compute NPTST for Variations I and II.

(e) It is used[216] to compute NPTST for Variation III.

KPC An array from which the standard plotting characters *, +, 0, X, =, 0, are moved[122] et seq. into the K array if not 1. It is equivalenced to A.

KPCI The plotting character to be moved to the buffer. It is determined either by a search loop[424-430] which establishes the subscript of its position in the K array, or by direct calculation[440] of that subscript (possible only for Variation I).

KPWRX The value of 3 - KHARX. If $\overline{64}$, it is computed,[302] used[304] to compute FlOX, and written out[306] in a special heading.

KSW1 A two-way switch variable. Initialized[114] to 1 and set[204] to 2 for Variation I, it is used[250,342,420] to control:

(a) The value of NSX when PISTUG is called[260] to scale the X array.

(b) The value of ILIM.

(c) The method of locating the correct plotting character, KPCI.

KSW3 A two-way switch variable. It is initialized[116] to 1, and set[226] to 2 for Variation II. It is used to control[270] the value of NSY when PISTUG is called[280] to scale the Y array.

KSW8 A two-way switch variable. It is initialized[113] to 1 and set[162] to 2 if $\overline{8}$. It is used[370] to branch to move the left-side field characters from the P array to the buffer.

KSW64 A two-way switch variable. It is initialized[112] to 1, and set[152] to 2 if $\overline{64}$. It is used to branch[300] to the preparation and output of a special heading for the left-side field, and also to branch[380] to move the value of an X_i to the buffer.

KSX One of three scaling parameters for the x scale. It is either calculated[260] by PISTUG or supplied[231] in P(6) by the programmer if $\overline{16}$. It is used[316] to compute XMF and is written out[318] in the INFO record. It is the number of places the decimal point must be moved to the right in the values of ODX and OFX by the 1401 subroutine PLOT.

KSY One of three scaling parameters for the y scale. It is either calculated[280] by PISTUG or supplied[242] in P(9) by the programmer if $\overline{32}$. It is used[317] to compute YMF and is written out[318] in the INFO record. It is the number of places the decimal point must be moved to the right in the values of ODY and OFY by the 1401 subroutine PLOT.

KTIMES The value of KN-1. It is computed[220] and used only for Variation II as the limit of the loop[221-225] that initializes the Y array.

KTL The number of Y_i's (elements in the Y array) associated with each element in the X array. It is initialized[111] to 1 and replaced[202] by the value of KN only for Variation I. It is the number of times the loop[400-446] is executed for each X_i. In each execution one value of KY is found and moved to the buffer with its associated plotting character. It is the total number of executions of the loop[320-322] that completes the FORMAT statement FUSE. KTL is also used[330,334] to compute KLC and KW.

KW The number of buffer words required to store KLC characters. It is the sum of KWADD and two times KTL. It is computed[334] and used[336,460] to compute NBW and KBL.

KWADD The number of buffer words, exclusive of those required to store KY's and plotting characters, needed to store one block. It is initialized[117] to 2 (one for KX and one for the delimiter), and increased by one[159] if $\overline{8}$, and by one[169] if $\overline{64}$. The total is then used[334] to compute KW.

L A subscript computed[224] for use in the inner loop[223-225] that initializes the Y array for Variation II.

LL A subscript computed[402] for use in the loop[400-446] that produces the KY's and their associated plotting characters.

M The frequency of the x grid-lines. It is set[119+1] to 10, and, if $\overline{2}$, it is replaced[172] by the value supplied in P(3) by the programmer. It is written out[318] in the INFO record.

MM A value computed[222] for use in the loop[223-225] that initializes the Y array for Variation II.

NBR The maximum number of blocks that can be written in one DATA record (of no more than 132 characters). It is computed[332] by dividing 130 (the number of characters in a DATA record following the "PD" required by 1401 subroutine PLOT) by KLC (the length of a block). It is used[336] to compute NBW (the length of the buffer).

NBW The number of buffer words required to hold the total number of characters to be written out in one DATA record. It is computed[336] and used[450] to test KBW (the buffer word counter) to tell if the buffer is full.

NF1 This is the number of characters within a block preceding the three characters of KX. It is initialized[119] to zero, and is increased[156] by seven if $\overline{8}$, and by six[166] if $\overline{64}$. The total is written out[318] in the INFO record.

NPTS The number of values in the X array for Variation I or in the Y array for Variation II. It is supplied[102] by the programmer in K(2). For Variation I, it establishes[344] the value of ILIM and is the length[252] of the X array sent to PISTUG[260] for scaling. For Variation II, it is stored[1222] in K(5), K(7), . . . K(2*KN + 1), used[222,223] to initialize the Y array, used[228] to compute NPTST, and is the length[272] of the Y array sent to PISTUG[280] for scaling.

NPTST The number of points in the X array for Variations II and III. It is computed[228] by multiplying KN (the number of curves) by NPTS for Variation II, and by summing[214-218] the numbers of points for each curve supplied in K(3), K(5), . . . K(2*KN + 1) by the programmer for Variation III. For Variation II, it is the length[276] of the Y array sent to PISTUG for scaling; for both II and III, it is the length[256] of the X array sent to PISTUG for scaling, as well as the value[348] of ILIM.

NSX The length of the array sent[260] to PISTUG to have the three scaling parameters for the x scale calculated. For Variation I it is equal[252] to NPTS and for Variations II and III it is equal[256] to NPTST.

NSY The length of the array sent[280] to PISTUG to have the three scaling parameters for the y scale calculated. For Variation II it is equal[272] to NPTS and for Variations I and III it is equal[276] to NPTST.

NY The frequency of the y grid-lines. It is set[119+2] to 10, and if $\overline{4}$, it is replaced[182] by the value supplied in P(4) by the programmer. It is written out[318] in the INFO record.

ODX A number written out[318] representing the x scale-factor. It is computed[312] by multiplying the scaling parameter DX, (always an integer) by 10^{-6} to put it in the form required for the INFO record.

ODY A number written out[318] representing the y scale-factor. It is computed[314] by multiplying the scaling parameter DY, (always an integer) by 10^{-6} to put it in the form required for the INFO record.

OFX A number written out[318] representing the x-scale starting-value. It is computed[313] by multiplying the scaling parameter FX, (always an integer) by 10^{-6} to put it in the form required for the INFO record.

OFY A number written out[318] representing the y-scale starting-value. It is computed[315] by multiplying the scaling parameter FY, (always an integer) by 10^{-6} to put it in the form required for the INFO record.

P An array name, the fourth argument in the call list of this subroutine. The first element specifies the Variation selected; other elements are supplied by the programmer as required by the options chosen.

PFX An intermediate value found[236] and used[237] in the calculation of KHARX.

RD A constant established[310] equal to 5.E-7 and used[312-315] for rounding.

RDX The true value of the x scale-factor. If 16, it is computed[235] and used[236] to compute KHARX.

RFX The true value of the x scale starting-value. If 16, it is computed[234] and used[236,237] to compute KHARX.

SFT A constant established[311] to equal 1.E-6 and used[312-315] for altering the scaling parameters to be written out[318] in the INFO record.

X An array name, the first argument in the call list of this subroutine. This array contains the values of the variable to be plotted on the vertical scale. It is sent[260] to PISTUG for scaling unless 16. Its current minimum value is found (and held as XMIN) each time the search loop[360-368] is executed. That element of the array is then replaced[398] by a large number (10^{15}) preceding the search for a new minimum.

XMAX An intermediate value found[237] during the calculation of KHARX.

XMF An intermediate value in the calculation of KX. It is computed[316] once and used[390] to compute each KX.

XMIN Initialized[350] to a large number, this holds[364] the current minimum value of the X array at the end of the search loop.[360-368] It is used[390] to compute KX. If 64, it is multiplied[382] by F10X before being moved to the buffer to prepare it for printing in the left-side field.

Y An array name, the second argument in the call list of this subroutine. It contains the values of the variable to be plotted on the horizontal scale. It is sent[280] to PISTUG for scaling unless 32. When Variation II is selected, the Y array is initialized[221-225] by filling it with repetitions of the given set of values.

YMF An intermediate value in the calculation of KY. It is computed[317] once and used[406] to compute each KY.

YS The value from the Y array currently being transformed[404-406] into a KY.

PISTUG - (FORTRAN II)

This routine is internal to the OPP system and is used by PLOTXY and PLOTMY. The calling statement is:

CALL PISTUG (KXY, NPTS, ARRAY, DELTA, FIRST, KSHIFT, KHAR)

PISTUG accepts as input; an array in any order, its length (NPTS), and a switch setting (KXY, which indicates whether scaling is to be done for the x or y scale and whether or not the array is in order). It returns as arguments: suitably chosen scaling parameters (FIRST, DELTA, and KSHIFT, which together define a starting-value and a scale-factor), and an integer (KHAR, which is required in the calling routine to create a special heading when option 64 is used).

PISTUG uses the minimum and maximum values of the array to compute the total range. When scaling parameters are being computed for the y scale, 101 print positions are available to cover the range. For the x scale, an arbitrarily (see appendix B, V.) chosen number of lines is assigned. In either case, the scale factor computed is always a value of $D \times 10^n$. D is a member of the set [2, 2.5, 5, 10].

If the array is such that the field width allotted for the printed grid-labels (9 characters) will be exceeded (see appendix B, VI.), PISTUG writes an error message, sets a signal to terminate the plot, and returns control to the calling routine.

```
      SUBROUTINE PISTUG(KXY,NPTS,ARRAY,DELTA,FIRST,KSHIFT,KHAR)

      SUBROUTINE PISTUG(KXY,NPTS,ARRAY,DELTA,FIRST,KSHIFT,KHAR)
C
      DIMENSION ARRAY(1)
C                                          RENAME INPUT
  100 N=NPTS
  102 K=KXY
C                                          FIND MIN AND MAX OF ARRAY
  104 X1=ARRAY(1)
  106 XN=ARRAY(N)
  107 GO TO(302,302,110),K
C
  302 DO 306 J=1,N
  304 X1 = MIN1F(X1,ARRAY(J))
  306 XN  = MAX1F(XN,ARRAY(J))
C                                          TLIN = F(NPTS) OR 101
  110 GO TO (112,116,112),K
C
  112 NP=N/35+1
  113 TLIN=FLOATF(NP)*55.
  114 GO TO 118
C
  116 TLIN=101.
C                                          COMPUTE TOTAL RANGE
  118 C3 = XN-X1
C                                          CALCULATE MEASURE OF SIZE AND TEST
  120 C4=C3/TLIN
  122 KCDX = XINTF(LOGF(ABSF(C4))/2.302585 +40.) -40
C
C                                          TO ERROR EXIT IF N.G.
  124 IF(25-KCDX*KCDX) 900,140,140
C                                          CALC MAX POSSIBLE DELTA
  140 TENCDX=10.**KCDX
  142 C5=C4/TENCDX
  144 AC5=ABSF(C5)
C                                          CHOOSE BEST VALUE OF D
  150 IF(2.5-AC5) 154,204,152
  152 IF(2.0-AC5) 204,200,200
  154 IF(5.0-AC5) 212,208,208
C
  200 D=2.
  202 GO TO 214
  204 D=2.5
  206 GO TO 214
  208 D=5.
  210 GO TO 214
  212 D=10.
C                                          CALC  SCALE FACTOR
  214 DX=SIGNF(D,C5)*TENCDX
C                                          IF SCALING X,PLOT ENDS TOO NEAR TOP OF
C                                          PAGE,CHANGE TOTAL LINES AND GO BACK
  216 GO TO (220,230,220),K
C
  220 IF( (2.*C3/DX-TLIN)/TLIN-.2)222,230,230
C
  222 TLIN=TLIN+35.
  224 GO TO 120
```

```
      SUBROUTINE PLSTUG(KXY,NPTS,ARRAY,DELTA,FIRST,KSHIFT,KHAR)
C                                          CALC STARTING VALUE
  230 ADX=ABSF(DX)
  232 INT=ABSF(X1)/ADX
  234 BELOW=ADX*FLOATF(INT)
  236 IF(DX*X1+1.E-20) 244,240,240
  240 F=BELOW
  242 GO TO 246
  244 F=BELOW+ADX
  246 FX=SIGNF(F,X1)
C                                          IF SCALE VALUES INTEGRAL,GO TO 270
  250 IF(KCDX) 256,256,252
  252 KSX=6
  254 GO TO 270
C                                          IF NOT,ALTER AND CALC SHIFT CODE
  256 KSFT=XABSF(KCDX)+1
  258 TEMP=10.**KSFT
  260 DX=DX*TEMP
  262 FX=FX*TEMP
  264 KSX = 6-KSFT
C                                          RENAME OUTPUT AND CALC KHAR
  270 DELTA = INTF(DX+SIGNF(.5,DX))
  272 FIRST = INTF(FX+SIGNF(.5,FX))
  274 KSHIFT=KSX
  276 KHAR = XINTF(LOGF(MAX1F(ABSF(X1),ABSF(XN)))/2.302585 +40.)-40
  280 RETURN
C
C
C                                          ERROR MESSAGE
  900       WRITE OUTPUT TAPE 6,500,K,  KCDX
  500   FORMAT(30HPL PLOT HALTED BY ARRAY NUMBER,I2,26H VALUES OUT OF RA
 5001NGE. KCD=,I4)
  902 DX=0.
  904 GO TO 270
      END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

SUBROUTINE PISTUG(KXY,NPTS,ARRAY,DELTA,FIRST,KSHIFT,KHAR)

STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
375	00567	32561	77461

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENT

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
AC5	374	00566	ADX	373	00565	BELOW	372	00564	C3	371	00563	C4	370	00562
C5	369	00561	D	368	00560	DX	367	00557	F	366	00556	FX	365	00555
INT	364	00554	KCDX	363	00553	K	362	00552	KSFT	361	00551	KSX	360	00550
NP	359	00547	N	358	00546	TEMP	357	00545	TENCDX	356	00544	TLIN	355	00543
X1	354	00542	XN	353	00541									

SYMBOLS AND LOCATIONS FOR SOURCE PROGRAM FORMAT STATEMENTS

	EFN	LOC		EFN	LOC		EFN	LOC		EFN	LOC		EFN	LOC
8)FK	500	00531												

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
1)	348	00534	2)	309	00465	3)	314	00472	6)	327	00507	9)	346	00532
C)G1	351	00537	C)G2	352	00540	E)4	64	00100	E)6	85	00125	E)8	140	00214
E)D	150	00226	E)I	178	00262	E)K	213	00325	E)O	232	00350	E)Q	296	00450
E)10D	150	00226	E)20J	184	00270									

LOCATIONS OF NAMES IN TRANSFER VECTOR

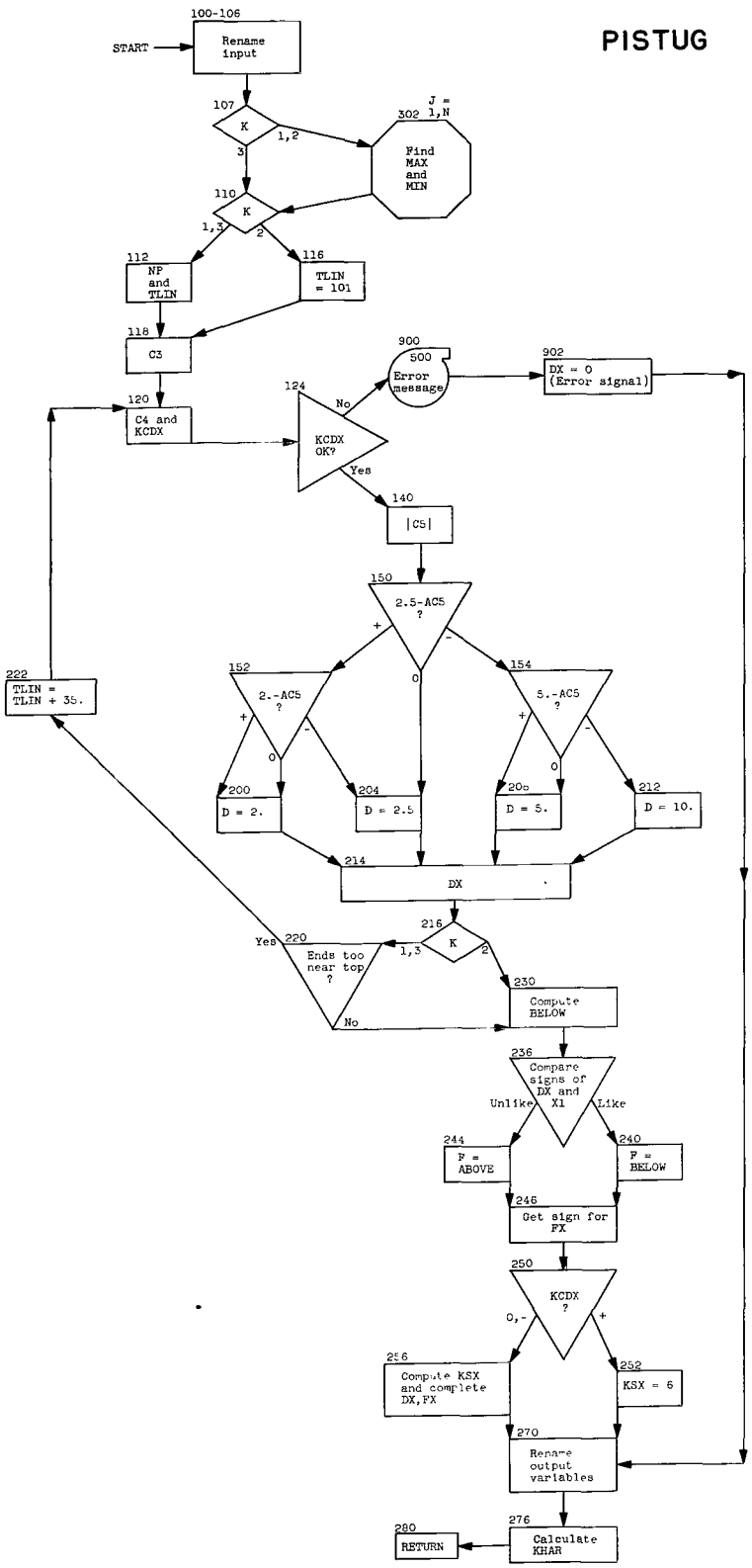
	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
EXP(2	1	00001	LOG	0	00000	(FIL)	3	00003	(STH)	2	00002			

ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

EXP(2 LOG (FIL) (STH)

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
100	5	00042	102	6	00045	104	7	00050	106	8	00052	107	9	00054
302	10	00057	304	11	00062	306	12	00070	110	13	00102	112	14	00105
113	15	00115	114	16	00124	116	17	00127	118	18	00131	120	19	00134
122	20	00137	124	21	00156	140	22	00166	142	23	00174	144	24	00177
150	25	00202	152	26	00207	154	27	00216	200	28	00223	202	29	00225
204	30	00230	206	31	00232	208	32	00233	210	33	00235	212	34	00236
214	35	00240	216	36	00246	220	37	00251	222	38	00264	224	39	00267
230	40	00272	232	41	00275	234	42	00307	236	43	00317	240	44	00327
242	45	00331	244	46	00332	246	47	00335	250	48	00341	252	49	00345
254	50	00347	256	51	00352	258	52	00356	260	53	00364	262	54	00367
264	55	00372	270	56	00375	272	57	00404	274	58	00413	276	59	00415
280	60	00444	900	62	00452	902	64	00462	904	65	00464			


PISTUG
100-106
Rename input
START
107
K
1,2
3
302 J = 1,N
Find MAX and MIN
110
K
1,3
2
112
NF and TLIN
116
TLIN = 101
118
C3
120
C4 and KCDX
900
500
Error message
902
DX = 0 (Error signal)
124
No
KCDX OK?
Yes
140
|C5|
150
2.5-AC5 ?
+
0
-
152
2.-AC5 ?
154
5.-AC5 ?
222
TLIN = TLIN + 35.
200
D = 2.
204
D = 2.5
206
D = 5.
212
D = 10.
214
DX
216
K
1,3
2
Yes
220
Ends too near top ?
No
230
Compute BELOW
236
Compare signs of DX and X1
Unlike
Like
244
F = ABOVE
240
F = BELOW
246
Get sign for FX
250
KCDX ?
0,-
+
256
Compute KSX and complete DX, FX
252
KSX = 6
270
Rename output variables
276
Calculate KHAR
280
RETURN

Superscripts are statement numbers in the FORTRAN listing (p. 45).
The notation z means option z is being used.

AC5 When the maximum possible scale-factor, C4, is expressed as x.xxxx multiplied by 10^n, AC5 is[144] the absolute value of the x.xxxx factor. It is used[150-154] to select D, the next larger value from the set of permissible values [2, 2.5, 5, 10].

ADX The absolute value of DX. It is used[232-246] in the calculation of the scaling parameter FX.

ARRAY Input to PISTUG, the third argument in the call list. Its minimum and maximum values are found in a loop[302-306] if they are not already known (as is the case when PISTUG is scaling an X array for PLOTXY).

BELOW An intermediate step in the calculation[230-246] of FX.

C3 The total range of the array being scaled. It is computed[118] from the minimum and maximum values of the array and used[120] to compute C4 (the maximum possible scale-factor). When an X array is being scaled, C3 is used[220] to decide if TLIN should be modified.

C4 The value of the maximum possible scale-factor. It is found[120] by dividing the range, C3, by TLIN and used[122] to compute KCDX.

C5 When the maximum possible scale-factor C4 is expressed as x.xxxx multiplied by 10^n, C5 is the x.xxxx factor. It is computed[142] and used[144] to find the value of AC5.

D This is found[150-212] as the smallest member of the set [2, 2.5, 5, 10] that is larger than AC5. It establishes[214] the first value of the scaling parameter DX.

DELTA Output from PISTUG, the fourth argument in the call list. It is the scaling parameter that represents the scale-factor. It is established as DX and renamed[270] to avoid excessive address modification.

DX The scaling parameter that represents the scale-factor. It is originally[214] the true value of the scale-factor; if KCDX $\leq$ 0, DX is multiplied[260] by a suitably chosen power of 10 and is still named DX. DX is set[902] to zero as an error signal to the calling routine.

F An intermediate step in the calculation[230-246] of FX.

FIRST Output from PISTUG, the fifth argument in the call list. It is the scaling parameter that represents the starting-value. It is established as FX and renamed[272] to avoid excessive address modification.

FX The scaling parameter that represents the starting-value. It is originally[246] the true value with which the first grid-line will be labeled; if KCDX $\leq$ 0, FX is multiplied[262] by a suitably chosen power of 10 and is still named FX.

INT An intermediate step in the calculation[232] of FX.

K A switch variable, supplied[102] (as KXY) by the calling routine. It will have the value:

(a) 1 when CALL is to scale X from PLOTMY.

(b) 2 when CALL is to scale Y from either PLOTXY or PLOTMY.

(c) 3 when CALL is to scale X from PLOTXY.

It is used[107,110,216] to control branching:

(a) To find the minimum and maximum of the array.

(b) To determine TLIN.

(c) To reevaluate TLIN if necessary.

KCDX This is the characteristic of $\log_{10}$ C4. It is computed[122] and tested[124] to see if it is equal to or between -5 and +5, and if it is not, an error message is written[900] and an error signal (DX=0) returned to the calling routine. It is also used[140] to compute TENCDX. If KCDX is zero or negative[250], the scaling parameters FX and DX are multiplied by 10^{KSFT}. (KSFT = $|KCDX| + 1$)

KHAR Output from PISTUG, the seventh argument of the call list. It is calculated[276] by finding the characteristic of the $\log_{10}$ of either the maxi- or the minimum of the array, whichever has the greater absolute value.

KSFT An intermediate value calculated[256] whenever KCDX $\leq$ 0. It is used[258] as the power of 10 by which the scaling parameters FX and DX are multiplied.

KSHIFT Output from PISTUG, the sixth argument in the call list. This scaling parameter is the decimal-point-shift code applicable to the other two scaling parameters DELTA and FIRST, which it accompanies. It is computed as KSX and renamed[274] to avoid excessive address modification.

KSX The scaling-parameter representing the decimal-point-shift code for DX and FX. If KCDX is greater than zero, KSX is set[252] equal to six. If KCDX is less than or equal to zero, KSX is set[256-264] equal to 5 - |KCDX|.

KXY Input to PISTUG, the first argument in the call list. It is renamed[102] K to avoid excessive address modification.

N The number of points in the array being scaled. It is supplied[100] by the calling routine (as NPTS) in the call list. It is used as the limit of the loop[302-306] that locates the minimum and maximum values of the array and in the original calculation[112-114] of TLIN when scaling an X array.

NP An intermediate value computed[112] in the calculation of TLIN.

NPTS Input to PISTUG, second argument in the call list. It is renamed[100] N to avoid excessive address modification.

TEMP Temporary storage.

TENCDX The value of 10^{KCDX}. It is computed[140] and used to compute C5[142] and DX.[214]

TLIN When PISTUG is scaling a Y array, TLIN is set[116] equal to 101. When an X array is being scaled, TLIN is found[113] first as an empirical function of the length of the array, later it may be modified.[220-222] (See appendix B, V.) In either case, TLIN is used[120-122] to compute KCDX.

X1 The minimum or maximum value of the array being scaled. It is set[104] equal to the first element of the array when PISTUG is scaling an X array for PLOTXY, otherwise it contains[304] the minimum value of the array after the search loop[302-306] is completed. It is used[118] to compute C3, the range of the array, and enters[232,236,246] several times into the calculation of FX, the scaling parameter representing the starting-value. It is also used[276] to compute KHAR.

XN The minimum or maximum value of the array being scaled. It is set[106] equal to the last element of the array when PISTUG is scaling an X array for PLOTXY, otherwise it contains[306] the maximum value of the array after the search loop[302-306] is completed. It is used[118] to compute the range of the array and in the calculation[276] of KHAR.

SORTXY - (FORTRAN II)

The calling statement is:

CALL SORTXY (V, W, NPTS)

This subroutine rearranges the NPTS elements of the V array in order of increasing size. The elements of the W array are moved to maintain the original pair-relation; that is, if the fifth element of the V array is moved to the first position of V, the fifth element of W is moved to the first position of W.

```
      SUBROUTINE SORTXY(X,Y,NPTS)

      SUBROUTINE SORTXY(X,Y,NPTS)
     DIMENSION X(100),Y(100)
100  N=NPTS
102  NN=N-1
104  DO 140 KT=1,NN
     XMIN=X(KT)
     JAD=KT
     JKL=KT+1
112  DO 120 JK=JKL,N
114  IF (XMIN-X(JK)) 120,120,116
116  XMIN=X(JK)
118  JAD=JK
120  CONTINUE
122  YMIN=Y(JAD)
     X(JAD)= X(KT)
     Y(JAD)= Y(KT)
     X(KT)= XMIN
     Y(KT)= YMIN
140  CONTINUE
      RETURN
     END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```


STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
97	00141	32561	77461

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENT

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
JAD	96	00140	JKL	95	00137	JK	94	00136	KT	93	00135	NN	92	00134
N	91	00133	XMIN	90	00132	YMIN	89	00131						

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
2)	80	00120	6)	81	00121	9)	87	00127	C)G2	88	00130			

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
100	4	00031	102	5	00033	104	6	00036	112	10	00054	114	11	00061
116	12	00066	118	13	00070	120	14	00074	122	15	00077	140	20	00111

SCALE - (FORTRAN II)

The calling statement is:

```
CALL SCALE (NPTS, A, KRSTR)
```

This subroutine finds the largest absolute value of the NPTS elements of A, then computes the characteristic of its $\log_{10}$. If the characteristic K is $-2 \leq K \leq 4$, KRSTR is set to zero and control returns to the calling program. If $K > 4$ or < -2, each element of A is multiplied by a power (KRSTR) of 10 to transform the array to suit PLOTXY and PLOTMY.

KRSTR is output to permit the user to ReSToRe the array or record how it has been altered.

```
      SUBROUTINE SCALE(NPTS,X,KRSTR)

      SUBROUTINE SCALE(NPTS,X,KRSTR)
      DIMENSION X(1)
  100 N=NPTS
  102 XMAX=X(1)
  104 DO 106 J=1,N
  106 XMAX=MAXIF(ABSF(XMAX),ABSF(X(J)))
  108 KHAR = XINTF(LOGF(XMAX)/2.302585+40.)-40
  116 IF((4-KHAR)*(KHAR+2))120,118,118
  118 KPWR=0
  119 GO TO 130
  120 KPWR=3-KHAR
  122 FACT=10.**KPWR
  126 DO 128 J=1,N
  128 X(J)=X(J)*FACT
  130 KRSTR=KPWR
  140 RETURN
      END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)



      SUBROUTINE SCALE(NPTS,X,KRSTR)

                                   STORAGE NOT USED BY PROGRAM

        DEC   OCT                DEC   OCT
        118 00166              32561 77461

          STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENT

       DEC   OCT                DEC   OCT                DEC   OCT                DEC   OCT                DEC   OCT
FACT   117 00165        KHAR   116 00164        KPWR   115 00163          N   114 00162        XMAX   113 00161

                        LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

       DEC   OCT                DEC   OCT                DEC   OCT                DEC   OCT                DEC   OCT
  1)   110 00156          2)    93 00135          3)    99 00143          6)   102 00146          9)   108 00154

                                LOCATIONS OF NAMES IN TRANSFER VECTOR

       DEC   OCT                DEC   OCT                DEC   OCT                DEC   OCT                DEC   OCT
EXP(2    1 00001        LOG      0 00000

                            ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

EXP(2       LOG

                     EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

 EFN   IFN   LOC          EFN   IFN   LOC          EFN   IFN   LOC          EFN   IFN   LOC          EFN   IFN   LOC
 100     4 00026          102     5 00030          104     6 00032          106     7 00035          108     8 00052
 116     9 00070          118    10 00103          119    11 00105          120    12 00106          122    13 00111
 126    14 00117          128    15 00122          130    16 00127          140    17 00131
```

PLOT - (1401 SPS)

Control is transferred to PLOT from the Lewis 1401 tape-to-printer program whenever a record has been read whose first character is a P. PLOT recognizes as input, and processes differently, four types of records; those whose first two characters are:

PT - hereinafter called TITLE records,

PI " " INFO "

PD " " DATA "

PL " " LEGEND "

The result of this processing is a printed plot consisting of x and y grid-lines, x and y grid-labels, plotted points, and possibly printed information at the top, bottom, and to the left of the grid. Sample tape records and the corresponding plotted output are in appendix A.

I. The input to this routine is any and all BCD records whose first character is P. The routine recognizes and processes any such record whose second character is T, I, D, or L. If any other character is in the second position of the record, the record is printed and control is returned to the tape-to-printer program.

A. TITLE records.

The remaining contents of a TITLE record (starting with the third character) are printed above the plot, starting in print position 1. The PT is not printed. The total length of a TITLE record is limited to 132 characters.

B. INFO records.

The remaining contents of an INFO record (starting with the third character) are information needed by PLOT to determine the construction of the grid, the construction of the grid-labels, the plotting character or characters, and the format of the DATA records that follow. Specifically, the PI record consists of the characters PI followed by 10 fields whose contents are:

(Field width is shown in parentheses.)

a. (1) This is the plotting character for all points unless it is a "-". In this case, the plotting characters will be supplied in the following DATA records.

b. (1) This digit is the decimal-point-shift code, PXSH, for XSTART and DX. (Items d. and f. in this list.)

c. (1) This digit is the decimal-point-shift code, PYSH, for YSTART and DY. (Items e. and g. in this list.)

d. (9) The number (XSTART) in this field is in the form ±0.xxxxxx and represents the first x grid-label.

e. (9) The number (YSTART) in this field is in the form ±0.xxxxxx and represents the first y grid-label.

f. (9) The number (DX) in this field is in the form ±0.xxxxxx and represents the x scale-factor (value of one line-space).

g. (9) The number (DY) in this field is in the form ±0.xxxxxx and represents the y scale-factor (value of one print position).

h. (3) This field contains an integer, XGD, specifying the frequency of the x grid-lines. They will be printed every XGD line-spaces below the first. If XGD = 0, only the first and final x grid-lines will be printed.

i. (3) This field contains an integer, YGD, specifying the frequency of the y grid-lines. They will be printed every YGD positions to the right of the first. If YGD = 0, only the first will be printed.

j. (3) This is an integer, FM1, specifying the length of the first field in each block of information in the following DATA record.

C. DATA records.

The remaining contents of a DATA record starting with the third character are successive blocks of information needed by PLOT to position the plotting characters and to print the left-side field. Only complete blocks are permitted within a record. Within blocks, the location in which a plotting character is to be printed is specified by two positive integers, X, the number of line-spaces below the first x grid-line, and Y, the number of print positions to the right of the first y grid-line. Specifically, the format of a block is:

(a) The first FM1 characters are the left-side field.

(b) The next three characters are X.

(c) (1) If the first character in the INFO record (after PI) is not a minus sign, the next three characters are Y and the block is complete.

(2) If it is a minus sign, the next three characters are Y and the following character is the plotting character for that point; since additional points may fall on the same line, additional Y's and their associated plotting characters may follow the first. The end of a block is indicated by a one-character delimiter, the letter E.

D. LEGEND records.

The remaining contents of a LEGEND record starting with the third character are printed below the plot starting in print position 1. The PL is not printed. The total length of a LEGEND record is limited to 132 characters.

II. The output from this routine consists of:

a. The contents of the first TITLE record printed at the top of a new page. The contents of succeeding TITLE records are printed single space, followed by a single space.

b. A line of y grid-labels. They are constructed using YSTART, DY, and PYSH. (See I.-B.-c.,e.,g.) The first label is centered over the first y grid-line, and the others are centered successively around every tenth print position to the right. The maximum length of a label is nine print positions; seven digits, decimal point, and sign.

c. A set of y grid-lines. The first is in print position 30 and the others are in every YGDth print position to the right (I.-B.-i.). The y grid-line character is "1". If a plotted point falls on the grid-line, the "1" is replaced by the appropriate plotting character.

d. A set of x grid-labels. They are constructed using XSTART, DX, and PXSH (I.-B.-b.,d.,f.). The first is printed with the first x grid-line to the immediate left of the first y grid-line. The others are printed every 10 line-spaces below this until the plot is terminated. The maximum label length is nine print positions; seven digits, decimal point, and sign.

e. A set of x grid-lines. The first is printed immediately below the line of y grid-labels (described in II.-B.) and the others every XGD line-spaces below the first until the last point has been printed (I.-B.-h.). If it did not lie in a grid line, the plot is continued until the next x grid-label is printed and a final x grid-line is then printed with the label. The x grid-line character is "-". If a plotted point falls on the grid-line, the "-" is replaced by the appropriate plotting character.

f. Points. Plotting characters are positioned on the page using X and Y (I.-C.).

g. Up to 19 characters to the left of the first y grid-line on any line on which a point is plotted. This is called the left-side field.

h. A line of y grid-labels (similar to II.-b.) is printed below the final x grid-line.

1. A double space follows the last line of grid-labels before the contents of the first LEGEND record are printed. Successive LEGEND records are printed single space.

III. PLOT is an SPS subroutine called by the Lewis 1401 tape-to-printer program that processes 7090 output tapes. An input record is expected to be in core, starting at RECORD, each time PLOT is called.

Each plot is started at the top of a new page. One or more TITLE records are printed, a single space occurs, and a line of y grid-labels (centered around every 10 print positions) is printed.

The first line of the plot is then constructed in the print area. It consists of the first x grid-label, the first x grid-line, the y grid-line character (every 10 print positions), and the plotting characters for any points with a line count of zero. Once all of the points whose line count is zero have been processed, the line is printed. Succeeding lines of the plot are constructed in the same manner with the following exceptions:

1. The x grid-labels are printed only on every tenth line.

2. The x grid-line frequency of printing is controlled by the input value XGD.

3. The spacing of the y grid-line characters is controlled by the y grid-line frequency, YGD.

When the first LEGEND record is received (indicating no more data points), the plot is completed in the following manner. If the line ready in the print area contains an x grid-line, it is printed and followed by a line of y grid-labels. If there is no x grid-line in the print area, then, after the current line is printed, the y grid-lines are continued until the line for the next x grid-label is reached. A final x grid-line accompanies this label and is followed by a line of y grid-labels. A double space occurs before the waiting LEGEND record is printed, and any futher LEGEND records are printed single space.

After the routine has processed any one record beginning with P, control is returned to the calling program.

THIS ROUTINE IS WRITTEN IN AN ASSEMBLY LANGUAGE SIMILAR TO 1401 SPS. THE ASSEMBLER IS A 7090 FORTRAN II PROGRAM WHICH HAS BEEN DISTRIBUTED THROUGH SHARE (NUMBER 1080).

THIS LISTING WAS PREPARED BY A SPECIAL 1401 PROGRAM WHICH COMBINED COMMENTS CARDS WITH THE ASSEMBLER OUTPUT.

NO SYMBOL TABLE DICTIONARY IS INCLUDED FOR THIS SUBROUTINE, SINCE THE DATA DEFINITIONS AND THE ALPHABETIC CROSS-REFERENCED LABEL TABLE WHICH FOLLOW SERVE THE SAME PURPOSE.

PLOT

NO.	CC	LABEL	OP	A-ADD.	B-ADD.		D	COMMENTS	LOCATION
1	00		ORG	1000				ALL EXITS FROM PLOT	
2	07	PEXIT	MCW	EDIT	EDX			WHEN A PLOT IS FINISHED (OR STOPPED WITH AN ERROR MESSAGE), THE	1000
3	01		MCW					EDIT WORDS MUST BE CLEARED,SWITCH FSTREC SET FOR FIRST-TITLE	1007
4	07		SW	FSTREC	LSTLAB			AND SWITCH LSTLAB SET FOR NOT-FIRST LEGEND - - BEFORE CONTROL	1008
5	04	START	B	*				IS RETURNED TO TTP THROUGH RETURN STORED AT *START*.	1015
								ENTRY TO SUBROUTINE PLOT	
6	04	ENTRY	SBR	START +3				STORE RETURN AT *START*.	1019
7	04		SW	PPGUP				SET SKIP-TO-NEW-PAGE-SWITCH (PPGUP) FOR USE BY TTP.	1023
8	08		B	DATA	REC	+1	D	IF DATA RECORD, GO TO *DATA*. IF INFO RECORD, GO TO *FORMAT*. IF	1027
9	08		B	FORMAT	REC	+1	I	LEGEND RECORD, GO TO *LABEL*. IF NONE OF THESE, TREAT AS A	1035
10	08		B	LABEL	REC	+1	L	TITLE RECORD.	1043

```
NO.  CC  LABEL   OP   A-ADD.      B-ADD.      D  - - - - - - - - COMMENTS - - - - - - - -  LOCATION

                                                                  PROCESS TITLE RECORDS

 11  01           NOP                                 FSTREC=FIRST-TITLE SWITCH.                          1051
 12  02  FSTREC  CC                              1        IF FIRST TITLE,  SKIP TO NEW PAGE.               1052

 13  07  TITLEP  CW   FSTPT       FSTREC                  SET SWITCH FSTPT FOR FIRST-DATA AND SWITCH FSTREC FOR  1054
                                                          NOT-FIRST TITLE.
 14  04           CS   336                           CLEAR PRINT AREA.                                   1061
 15  01           CS                                                                                     1065
 16  07           MCM  REC   +2   201                 MOVE TITLE RECORD(OMITTING 1ST 2 CHARACTERS-PT-) INTO PRINT AREA.  1066
 17  07           MCW  EDIT  -8   336                 MOVE X GRID-LINE CHARACTER (-) INTO 336.            1073
 18  04           W    START                          PRINT THE TITLE RECORD AND EXIT TO ITP THRU *START*. 1080

                                                                  PROCESS DATA RECORDS.

 19  07  DATA     SBR  A           REC   +1           INITIALIZE INDEX A FOR STEPPING THROUGH DATA RECORD. 1084

 20  01           NOP                                 FSTPT=FIRST-DATA SWITCH                             1091
 21  04  FSTPT    B    ZX                                 IF NOT-FIRST DATA, GO TO *ZX*.                  1092
 22  04           SW   FSTPT                              IF FIRST-DATA, SET SWITCH FSTPT FOR NOT-FIRST DATA. 1096
 23  04  INTX     S    LASTX                          CLEAR LINE COUNTER(LASTX) TO ZERO.                  1100

 24  07  MIYRET   C    XLBLTS      TEN                 IF IT IS TIME FOR AN X GRID-LABEL, (EVERY 10TH LINE), GO TO  1104
 25  05           B    XGL                       S        SUBROUTINE *XGL* TO POSITION LABEL IN PRINT AREA.  1111
 26  07  XLRET    C    YGDTST      XGD                 IF IT IS NOT TIME FOR AN X GRID-LINE,(EVERY-XGD-LINES),  1116
 27  05           B    YGDR                      /        GO TO *YGDR*.                                   1123

 28  04  YGRID    SW   PLZERO                              IF IT IS, PREPARE TO PUT X GRID-LINE CHARACTER INTO PRINT  1128
 29  07           SBR  YGDTST      0                      AREA. CLEAR COUNTER YGDTST TO ZERO. FILL PRINT AREA (EXCEPT  1132
 30  07  NOYGD    MCW  MINUS       335                    LEFT-SIDE FIELD) WITH MINUS SIGNS.                1139

 31  04  YGDR     B    VTGRID                         GO TO SUBROUTINE *VTGRID* TO POSITION Y GRID-LINE CHARACTERS IN  1146
                                                          PRINT AREA IN EVERY YGD PRINT POSITIONS.

 32  04  ZX       S    Y                              CLEAR WORKING AREA FOR X.                           1150
 33  04           MCW  Y                              CLEAR WORKING AREA FOR Y.                           1154
 34  07  XADD     A    0           A X                MOVE X IN FROM DATA RECORD.                         1158
 35  07  YADD     A    0           A Y                MOVE Y IN FROM DATA RECORD.                         1165

 36  07  XST01    C    X             LASTX            IF X=LASTX(LINE COUNTER), POINT IS TO BE PRINTED ON THIS LINE,SO  1172
 37  05  TNXTX    B    PLOT                       S        GO TO PLOT.                                     1179
 38  07           S    LASTX         X                IF NOT, FORM (X-LASTX). IF THIS VALUE IS NEGATIVE, GO TO  1184
 39  08  DXTST    BWZ  STOP          X          K        ERROR MESSAGE VIA *STOP*.                       1191
 40  07           A    LASTX         X                   IF OK, RESTORE VALUE OF X.                      1199
 41  01           W                                   PRINT A LINE.                                       1206
 42  04           CS   332                            CLEAR PRINT AREA AND INITIALISE  FOR NEXT LINE.     1207
 43  01           CS                                                                                     1211
 44  07           A    ONE           XLBLTS               STEP COUNTER FOR X GRID-LABELS (XLBLTS) BY 1.   1212
 45  01           A                                       STEP LINE COUNTER (LASTX) BY 1.                 1219
 46  01           A                                       STEP COUNTER FOR X GRID-LINES (YGDTST) BY 1.    1220
 47  07           A    DX            XSTART               STEP VALUE OF XSTART (USED FOR X GRID-LABELS) BY DX.  1221
 48  04  CHGSIG   B    MIYRET                         GO TO *MIYRET*.                                     1228
```

```
NO.  CC  LABEL   OP   A-ADD.      B-ADD.      D  - - - - - - - - COMMENTS - - - - - - - -  LOCATION

 49  04  PLOT    SW   1         A                  SET WORD-MARK IN DATA RECORD TO END MOVE.          1232
 50  07  PLOTA   MCW  0         A 219              MOVE LEFT-SIDE FIELD TO PRINT AREA.               1236
 51  04           CW   1         A                  CLEAR WORD-MARK.                                  1243
 52  07  YST01   MN   Y           B                STORE Y IN INDEX B (OMITTING SIGN).               1247
 53  01           MCW                                                                               1254
 54  07  YST02   C    B           YLAST            IF Y IS TOO BIG (GREATER THAN 103), GO TO *TOBIG*. 1255
 55  05           B    TOBIG                  T                                                    1262

 56  01           NOP                           MCBRA=MULTIPLE-CURVE SWITCH.                      1267
 57  04  MCBRA    B    MANYY                         IF MULTIPLE CURVES, GO TO *MANYY*.            1268
 58  07           MCW  PLCHAR     PLZERO    B         IF SINGLE CURVE,MOVE PLOT CHARACTER INTO PRINT AREA B 1272
                                                      POSITIONS TO THE RIGHT OF THE FIRST Y GRID-LINE.

 59  07  PLOTBK   A    TWONE       A                STEP A TO NEXT BLOCK OF DATA RECORD.              1279

 60  08  ONWARD   B    START       6          A ‡   END OF RECORD TEST.                               1286
 61  01           B                                    IF ONE OF THE NEXT SIX CHARACTERS IN THE DATA RECORD 1294
 62  01      •    B                                    IS A GROUP MARK, GO TO *START*. IF NOT, GO    1295
 63  01           B                                    TO *ZX* TO PROCESS NEXT BLOCK.                1296
 64  01           B                                                                                  1297
 65  01           B                                                                                  1298
 66  04           B    ZX                                                                            1299

 67  01  MANYY    NOP                           MANYY=FIRST-Y-IN-BLOCK SWITCH.                    1303
 68  04           B    MODA                          IF THIS IS FIRST Y, BRANCH TO *MODA* AND RETURN. 1304

 69  04  MUPLC    SW   PLZERO    B                      SET WORD-MARK FOR PLOT CHARACTER AND MOVE PLOT CHARACTER INTO 1308
 70  07           MCW  1         A PLZERO    B         PRINT AREA B POSITIONS TO THE RIGHT OF THE FIRST Y GRID-LINE. 1312

 71  07  UPA      SBR  A           1          A     STEP INDEX A BY 1.                                1319

 72  08  LTYTST   BBE  ENDTST      1          A +   IF NEXT CHARACTER IS THE DELIMITER, GO TO *ENDTST*. 1326
 73  07           SBR  A           3          A     IF NOT, STEP INDEX A FOR ADDRESS OF NEXT Y.       1334
 74  04           S    Y                            CLEAR WORKING STORAGE Y.                      1341
 75  07           A    0         A Y                MOVE NEXT Y IN FROM DATA RECORD.              1345
 76  04           B    YST01                      GO BACK TO *YST01*.                               1352

 77  07  ENDTST   SBR  A           1          A     STEP INDEX A TO NEXT BLOCK OF DATA RECORD.        1356
 78  04           SW   MANYY +1                    SET SWITCH MANYY FOR FIRST-Y.                     1363
 79  04           B    ONWARD                     GO TO *ONWARD*.                                   1367

 80  04  MODA     CW   MANYY +1                    SET SWITCH MANYY FOR NOT-FIRST-Y.                 1371
 81  07           A    TWONE       A              STEP INDEX A TO LOCATE FIRST PLOT CHARACTER IN DATA BLOCK. 1375
 82  04           B    MUPLC                      GO TO *MUPLC*.                                    1382

 83  07  TOBIG    MCW  RM          332            PUT A RECORD MARK AT RIGHT END OF PRINT AREA TO SHOW Y TOO BIG. 1386
 84  08           BWZ  UPA         MCBRA      1   IF SWITCH MCBRA IS SET FOR MULTIPLE CURVES, GO TO *UPA*. IF FOR 1393
 85  04           B    PLOTBK                        SINGLE CURVE, GO TO *PLOTBK*.                 1401
```

```
NO.  CC  LABEL   OP   A-ADD.        B-ADD.      D   - - - - - - - - COMMENTS - - - - - - - -   LOCATION

                                                                    SUBROUTINE *NXREC*
                                                        CONSTRUCTS AND PRINTS LINE OF Y GRID-LABELS
 86  04  NXREC   SBR  ENDNR +3                        STORE RETURN AT *ENDNR*.                    1405
 87  04          CS   332                            CLEAR PRINT AREA.                           1409
 88  01          CS                                                                             1413
 89  07  LABOUT  MCW  YSTART       TEMP1             MOVE YSTART TO TEMP1.                       1414
 90  07          SBR  A            0                 CLEAR INDEX A.                              1421

 91  07  NXTLAB  SBR  B            0                 CLEAR INDEX B. IF FIRST 5 POSITIONS OF YSTART (NOW IN TEMP1) ARE  1428
 92  07          C    TEMP1 -1     ZTST  -1               NOT ZERO, GO TO *EDYST*. IF THEY ARE ZERO, MOVE LAST DIGIT  1435
 93  05          B    EDYST                  /           (WITHOUT SIGN), AND IF IT IS ALSO ZERO, GO TO  *YZROK*.      1442
 94  07  YZRTST  MN   TEMP1        PLZERO    A                                                  1447
 95  08          B    YZROK        PLZERO    A 0                                                1454

 96  07  EDYST   LCA  EDY          TEMP2             LOAD  Y EDIT WORD INTO TEMP2 AND EDIT TEMP1.  1462
 97  07          MCE  TEMP1        TEMP2                                                        1469

 98  08  TSTB    BBE  NONB         TEMP2 -6  B ‡     LOOP TO LOCATE FIRST NON-BLANK POSITION IN YSTART (NOW IN TEMP2).  1476
 99  07          SBR  B            1         B           EXIT TO *NONB* WITH INDEX B = NUMBER OF BLANKS.  1484
100  04          B    TSTB                                                                     1491

101  04  NONB    SW   TEMP2 -7                       SET LENGTH OF LABEL FIELD = 8. MOVE SIGN INTO BTH POSITION.  1495
102  07          MZ   TEMP2 -8     TEMP2 -7  B                                                1499

103  08  UPB     B    MVLAB        B           6     THE EFFECT OF THIS LOOP IS TO BUILD A CONSTANT IN INDEX A, TO BE  1506
104  07          SBR  A            1         A           USED IN CENTERING A Y GRID-LABEL, AND A CONSTANT IN*TEN* TO  1514
105  07          SBR  B            1         B           BE USED IN *YZROK*  WHICH ADVANCES INDEX A READY FOR CENTER-  1521
106  07          A    I99          TEN                   ING THE NEXT LABEL.                    1528
107  04          B    UPB                                                                      1535

108  07  MVLAB   MCW  TEMP2        PLZERO-2  A       MOVE Y GRID-LABEL TO PRINT AREA USING CONSTANT IN INDEX A.  1539

109  07  YZROK   A    TEN          A                 INCREASE INDEX A BY CONSTANT IN *TEN*.      1546
110  07          SBR  TEN          10                RESTORE *TEN* = 10.                         1553
111  07          A    DY           TEMP1             ADD DY TO THIS Y GRID-LABEL TO GET NEXT LABEL.  1560
112  07          C    ONETEN       A                 IF PRINT LINE IS NOT COMPLETED, GO BACK TO *NXTLAB*. WHEN IT IS  1567
113  05          B    ENDNR                  U           COMPLETED, PRINT THE LINE OF LABELS AND EXIT THROUGH RETURN  1574
114  04          B    NXTLAB                             STORED AT *ENDNR*.                     1579
115  04  ENDNR   W    *                                                                        1583

                                                                              ERROR EXIT.
                                                                  WRITE ERROR MESSAGE AND HALT THIS PLOT.
116  01  STOP    W                                  PRINT A LINE.                               1587
117  04          CS   332                            CLEAR PRINT AREA.                           1588
118  01          CS                                                                             1592
119  07          LCA  Y            332               MOVE CURRENT VALUE OF Y INTO PRINT AREA.    1593
120  01          LCA                                 MOVE CURRENT VALUE OF(X-LASTX),WHICH IS NEGATIVE, INTO PRINT AREA.  1600
121  01          LCA                                 MOVE MESSAGE NAMED XOFORD INTO PRINT AREA.  1601
122  01          W                                   PRINT ERROR MESSAGE.                        1602

123  08  SKREC   MCW  (U3           REC         R     READ A TAPE RECORD.                         1603
124  08          B    SKPON        REC         P           IF FIRST CHARACTER IS P, GO TO *SKPON*.  1611

125  05  PNEX    CU   (U3                        B           IF NOT, BACKSPACE TAPE ONE RECORD AND RETURN TO TTP THROUGH  1619
126  04          B    PEXIT                               *PEXIT*.                               1624

127  08  SKPON   B    PNEX         REC   +1     T     IF SECOND CHARACTER IS T, GO TO *PNEX*.     1628
128  04          B    SKREC                              IF NOT, GO TO *SKREC*.                 1636
```

```
NO.  CC  LABEL   OP   A-ADD.      B-ADD.      D  - - - - - - - - COMMENTS - - - - - - - -  LOCATION

                                                PROCESS INFO RECORDS
                                                (ENTRY AT *FORMAT*)

129  04  MULCV   SW   MCBRA                        SET SWITCH MCBRA FOR MULTIPLE CURVES.          1640
130  04           B    MCBACK                      GO TO *MCBACK*.                                1644
131  07  FORMAT   MCM  REC   +2   IREC             MOVE INFO RECORD (OMITTING 1ST 2 CHARACTERS -PI-) TO IREC.  1648
132  07           CW   MCBRA      LSTLAB           SET SWITCH MCBRA FOR SINGLE CURVES AND SWITCH LSTLAB FOR FIRST-  1655
133  08           B    MULCV      PLCHAR     -          LEGEND. IF PLOT CHARACTER IS (-), GO TO *MULCV* AND RETURN.  1662

134  07  MCBACK   SBR  A          0                  CLEAR INDEX A.                                 1670
135  07           MCW  PXSH       A                  CONSTRUCT THE EDIT WORDS, EDX AND EDY, FOR X AND Y GRID-LABELS,  1677
136  07           MCW  PERZ       EDX   -6   A            USING DECIMAL-POINT-SHIFT CODES IN INDEX A TO POSITION THE  1684
137  07           MCW  PYSH       A                        POINTS.                                1691
138  07           MCW  PERZ       EDY   -6   A                                                    1698
139  07           MZ   XSTART-8   XSTART             MOVE SIGNS OF XSTART AND YSTART FROM HIGH TO LOW ORDER POSITIONS.  1705
140  07           MZ   YSTART-8   YSTART                                                         1712
141  07           MCW  ZTST  -4   XSTART-6           REPLACE THE DECIMAL POINT IN XSTART WITH A ZERO.  1719
142  07           MCW  DY         DYT   -1           MULTIPLY DY BY 10 BY MOVING INTO AND OUT OF DYT.  1726
143  07           MCW  DYT        DY                                                             1733
144  07           MZ   DX    -8   DX                 MOVE SIGNS OF DX AND DY FROM HIGH TO LOW ORDER POSITIONS.  1740
145  07           MZ   DY    -8   DY                                                             1747
146  07  DXPL     SBR  A          0                  CLEAR INDEX A.                                 1754
147  07           A    ZTST  +3   YGD                REPLACE LEADING BLANKS IN XGD AND YGD WITH ZEROES BY CHAINED  1761
148  01           A                                       ADDING OF A SIGNED CONSTANT ZERO.         1768

149  07           C    XGD        A                  TEST X GRID-LINE FREQUENCY.                    1769
150  05           B    GRDX                    /          IF NON-ZERO, GO TO *GRDX*.                1776
151  07           MCW  NINS       XGD                     IF ZERO, REPLACE BY 999.                  1781

152  07  GRDX     C    YGD        A                  TEST Y GRID-LINE FREQUENCY.                    1788
153  05           B    GRDY                    /          IF NON-ZERO, GO TO *GRDY*.                1795
154  07           MCW  NINS       YGD                     IF ZERO, REPLACE BY 999.                  1800

155  07  GRDY     A    FM1        A                  ADD LEFT-SIDE-FIELD LENGTH TO INDEX A.         1807
156  07           MN   A          PLOTA +3           MODIFY ADDRESS OF *PLOTA*.                     1814
157  01           MN                                                                             1821
158  07           C    A          ZTST  -4           TEST LEFT-SIDE-FIELD LENGTH.                   1822
159  05           B    MVNOP                   S          IF ZERO, GO TO *MVNOP*.                   1829
160  07           MCW  *     -6   PLOTA                   IF NON-ZERO, MAKE *PLOTA* A MOVE COMMAND.  1834

161  07  MNVR     SBR  A          3          A       ADD 3 TO INDEX A. USE NEW TOTAL TO MODIFY *XADD*.  1841
162  07           MN   A          XADD  +3                                                       1848
163  01           MN                                                                             1855
164  07           SBR  A          3          A       AGAIN ADD 3 TO INDEX A AND MODIFY ADDRESS OF *YADD*.  1856
165  07           MN   A          YADD  +3                                                       1863
166  01           MN                                                                             1870
167  07           SBR  TWONE      0          A       STORE INDEX A AT TWONE (6 + LENGTH OF LEFT-SIDE-FIELD).  1871
168  05           CC   NXREC                 J       DOUBLE SPACE AND GO TO *NXREC* TO PRINT A LINE OF Y GRID-LABELS.  1878
169  04           SW   PLZERO                         SET A WORD-MARK IN PRINT AREA AT FIRST Y GRID-LINE (PLZERO). CLEAR  1883
170  01           CS                                       ALL POSITIONS TO THE LEFT AND PUT THE X GRID-LINE CHARACTER  1887
171  07           MCW  MINUS      335                      (MINUS SIGN) IN ALL POSITIONS TO THE RIGHT.  1888
172  07           SBR  MODXGD+3   TEN                FORCE FIRST X GRID-LINE AND LABEL, AND Y GRID-CHARACTERS (EVERY 10  1895
173  07           MCW  TEN        XL8LTS                   POSITIONS), BY PRE-SETTING *MODXGD*,XL8LTS,AND YGDTST.  1902
174  07           MCW  XGD        YGDTST                                                         1909
175  04           B    START                         RETURN TO TTP THROUGH *START*.                 1916
```

```
NO.  CC  LABEL   OP   A-ADD.      B-ADD.     D  - - - - - - - - COMMENTS - - - - - - - -  LOCATION

                                                                  PROCESS LEGEND RECORDS

176  01  LABEL   NOP                                LSTLAB=FIRST-LEGEND SWITCH.                1920
177  04  LSTLAB  B    LAB2                              IF NOT-FIRST LEGEND, GO TO *LAB2*.      1921
                                                        IF FIRST-LEGEND,ASK IF LINE WAITING IN PRINT AREA IS A
178  08          BBE  LABP        331           ‡       GRID-LINE. IF IT IS, GO TO *LABP*.      1925
179  01          W                                      IF NOT, PRINT IT.                       1933
180  04          CS   332                            CLEAR PRINT AREA.                          1934
181  01          CS                                                                             1938
182  07  CPYGD   A    DX          XSTART              INCREASE XSTART (VALUE USED FOR X GRID-LABELS) BY DX.  1939
183  07          A    ONE         XLBLTS              STEP X GRID-LABEL COUNTER (XLBLTS) BY 1.   1946
184  07          C    XLBLTS      TEN                 TEST X GRID-LABEL COUNTER TO SEE IF IT IS TIME FOR
                                                        A LABEL (EVERY TENTH LINE).
185  05          B    ENDGD                    S         IF TIME TO PRINT AN X GRID-LABEL, GO TO *ENDGD*.  1960
186  04          B    VTGRID                             IF NOT, GO TO SUBROUTINE *VTGRID* FOR Y GRID-LINES.  1965
187  04          W    CPYGD                              CONTINUE PRINTING GRID-LINES UNTIL IT IS TIME FOR NEXT X
                                                        GRID-LABEL.                             1969

188  04  ENDGD   SW   PLZERO                          SET A WORD MARK IN PRINT AREA AT FIRST Y GRID-LINE (PLZERO). PUT  1973
189  07          MCW  MINUS       335                    X GRID-LINE CHARACTER (-) IN ALL POSITIONS TO THE RIGHT.  1977
190  07          SBR  MODXGD+3    TEN                 SET A-ADD. OF *MODXGD*=10.                 1984
191  04          B    VTGRID                         GO TO SUBROUTINE *VTGRID* TO PUT Y GRID-LINE CHARACTERS INTO
                                                        EVERY TENTH POSITION IN LAST X GRID-LINE.  1991

192  04  LABP    B    XGL                            GO TO SUBROUTINE *XGL* FOR X GRID-LABEL.PRINT LAST X GRID-LINE AND  1995
193  04          W    NXREC                              GO TO SUBROUTINE *NXREC* FOR LINE OF Y GRID-LABELS. DOUBLE
                                                        SPACE.                                  1999
194  02          CC                               L                                             2003
195  04  LAB2    CS   332                            CLEAR PRINT AREA.                          2005
196  01          CS                                                                             2009
197  07          MCM  REC   +2    201                MOVE LEGEND RECORD TO PRINT AREA. PRINT AND RETURN TO TTP VIA  2010
198  04          W    PEXIT                              *PEXIT*.                               2017

                                                                  PATCH FOR PROCESS INFO

199  07  MVNOP   MCW  FSTREC-1    PLOTA               MAKE *PLOTA* A NOP COMMAND.                2021
200  04          B    MNVR                           GO TO *MNVR*.                              2028
```

```
NO.  CC  LABEL    OP   A-ADD.        B-ADD.      D   - - - - - - - - COMMENTS - - - - - - - -   LOCATION

                                                          SUBROUTINE *VTGRID*
                                                INSERTS Y GRID-LINE CHARACTERS (1) INTO PRINT AREA.

201  04  VTGRID   SBR  YGXIT +3                          STORE RETURN AT *YGXIT*.                  2032
202  07           SBR  B           0                    CLEAR INDEX B                             2036
203  07  XGRID    MCW  VERT        PLZERO      B        MOVE Y GRID-LINE CHARACTER INTO PRINT AREA USING INDEX B. ADD   2043
204  07  MODXGD   A    XGD         B                         TO INDEX B THE QUANTITY SPECIFIED IN A-ADD. OF *MODXGD*.  2050
205  07           C    ONETEN      B                         CONTINUE UNTIL B = 105.                   2057
206  05  NOXGDA   B    XGRID                       T                                                  2064
207  07           SBR  MODXGD+3    YGD                  RESET A-ADD. OF *MODXGD* TO Y GRID-LINE FREQUENCY.  2069
208  04  YGXIT    B    *                                RETURN                                    2076

                                                          SUBROUTINE *XGL*
                                                CONSTRUCTS AND EDITS X GRID-LABEL USING XSTART

209  04  XGL      SBR  RETXL +3                         STORE RETURN                              2080
210  07           C    XSTART-1    ZTST  -2             IF FIRST SIX DIGITS OF X START ARE NOT ZERO, GO TO *XGLA*.  2084
211  05           B    XGLA                    /                                                  2091
212  07  XZRTST   MN   XSTART      PLZERO-3                  MOVE 7TH DIGIT (WITHOUT SIGN). IF IT IS ZERO, GO TO*RETXL*-7.  2096
213  08           B    RETXL -7    PLZERO-3    0                                                  2103

214  07  XGLA     LCA  EDX         PLZERO-2             LOAD X EDIT WORD INTO PRINT AREA AND EDIT XSTART.  2111
215  07           MCE  XSTART      PLZERO-2                                                        2118
216  07           SBR  XLBLTS      0                    SET X GRID-LABEL COUNTER = 0 AND RETURN.  2125
217  04  RETXL    B    *                               RETURN                                    2132
```

```
NO.  CC  LABEL   OP   A-ADD.          - - - - - - - - DATA DEFINITIONS - - - - - - - -                    LOCATION
218  01  PPGUP   DC   *               SKIP-TO-NEW-PAGE SWITCH. A WORD MARK IS SET AT PPGUP BY PLOT EACH    2136
                                          TIME IT IS CALLED, FOR LATER USE BY TTP TO START NON-PLOT
                                          OUTPUT ON NEW PAGE.
219  02  PERZ    DCW  *       0.      CONSTANT USED IN CONSTRUCTING THE EDIT WORDS EDX AND EDY.            2138

                                                    EDY,EDX,EDIT
                                               THE NEXT THREE FIELDS MUST BE ADJACENT AND IN
                                               THE ORDER SHOWN.(USED IN A CHAINED MOVE.)
220  09  EDY      DCW  *       -       EDIT WORD FOR THE Y GRID-LABELS.                                    2147
221  09  EDX      DCW  *       -       EDIT WORD FOR THE X GRID-LABELS.                                    2156
222  09  EDIT     DCW  *       -       CONSTANT USED TO INITIALIZE THE EDIT WORDS EDX AND EDY.              2165

                                                    XOFORD,X,Y
                                               THE NEXT THREE FIELDS MUST BE ADJACENT AND
                                               IN THE ORDER SHOWN.(USED IN CHAINED MOVE.)
223  15  XOFORD   DCW  *       X OUT OF ORDER  ERROR MESSAGE.                                              2180
224  03  X        DCW  *       000     INPUT - THE LINE COUNT FOR A POINT OR SET OF POINTS.                2183
225  03  Y        DCW  *       000     INPUT - THE PRINT POSITION COUNT FOR A POINT.                       2186
226  03  NINS     DCW  *       999     CONSTANT REPLACES XGD OR YGD IF EITHER IS ZERO.                     2189

                                                  YGDTST,LASTX,XLBLTS
                                               THE NEXT THREE FIELDS MUST BE ADJACENT AND
                                               IN THE ORDER SHOWN.(USED IN A CHAINED ADD)
227  03  YGDTST   DCW  *       +00+    THE X GRID-LINE COUNTER.IT IS STEPPED BY ONE WHENEVER A LINE IS     2192
                                          PRINTED AND COMPARED TO XGD. WHENEVER THEY ARE EQUAL, AN X
                                          GRID-LINE IS PRINTED AND THE COUNTER IS RESET TO ZERO.
228  03  LASTX    DCW  *       +00+    THE LINE COUNTER. IT IS INITIALIZED TO ZERO WHEN THE FIRST DATA     2195
                                          RECORD IS PROCESSED AND STEPPED BY ONE EACH TIME A LINE IS
                                          PRINTED.
229  03  XLBLTS   DCW  *       +00+    THE X GRID-LABEL COUNTER. IT IS SET TO TEN TO FORCE THE FIRST X     2198
                                          LABEL TO PRINT, THEN RESET TO ZERO.EACH TIME A LINE IS PRINT-
                                          ED, IT IS INCREASED BY ONE AND COMPARED TO TEN TO DETERMINE
                                          IF A LABEL SHOULD BE PRINTED. WHEN EQUAL TO TEN A LABEL IS
                                          PRINTED AND THE COUNTER RESET TO ZERO.

230  07  ZTST     DCW  *       0000000 CONSTANT ZERO.                                                      2205
231  09  DYT      DCW  *       000000000       FIELD USED TO MULTIPLY DY BY 10 BY SHIFTING  IT LEFT ONE POSITION. 2214
232  03  TEN      DCW  *       010     CONSTANT. ALSO USED AS A COUNTER IN CENTERING Y GRID-LABELS.        2217

                                                    VERT,      ,ONE
                                               THE NEXT THREE FIELDS MUST BE ADJACENT AND
                                               IN THE ORDER SHOWN.(USED IN A CHAINED ADD)
233  01  VERT     DCW  *       1       THE X GRID-LINE CHARACTER.                                          2218
234  03            DCW  *       001                                                                      2221
235  03  ONE      DCW  *       001     CONSTANT.                                                           2224

236  03  ONETEN   DCW  *       105     CONSTANT USED TO STOP PRINTING Y GRID-LINES.                        2227
237  01  RM       DCW  *       ‡       CONSTANT.  IF Y IS GREATER THAN 102 THIS CHARACTER IS PRINTED IN    2228
                                          PRINT POSITION 132.
238  07  TEMP1    DCW  *               TEMPORARY STORAGE USED IN COMPUTING Y GRID-LABELS.                  2235
239  03  I99      DCW  *       I9I     CONSTANT USED IN CENTERING THE Y GRID-LABELS.(COMPLEMENT OF I)      2238
240  03  YLAST    DCW  *       103     CONSTANT. IF Y IS GREATER THAN YLAST A RECORD MARK (‡) IS PRINTED   2241
                                          IN PRINT POSITION 132.
241  03  TWONE    DCW  *       000     COMPUTED DURING THE PROCESSING OF THE INFO RECORD,THIS IS 6 + FM1.  2244
                                          IT IS USED TO LOCATE THE NEXT DATA BLOCK IN THE DATA RECORD
                                          FOR SINGLE CURVES AND THE FIRST PLOTTING CHARACTER IN THE
                                          BLOCK FOR MULTIPLE CURVES.
```

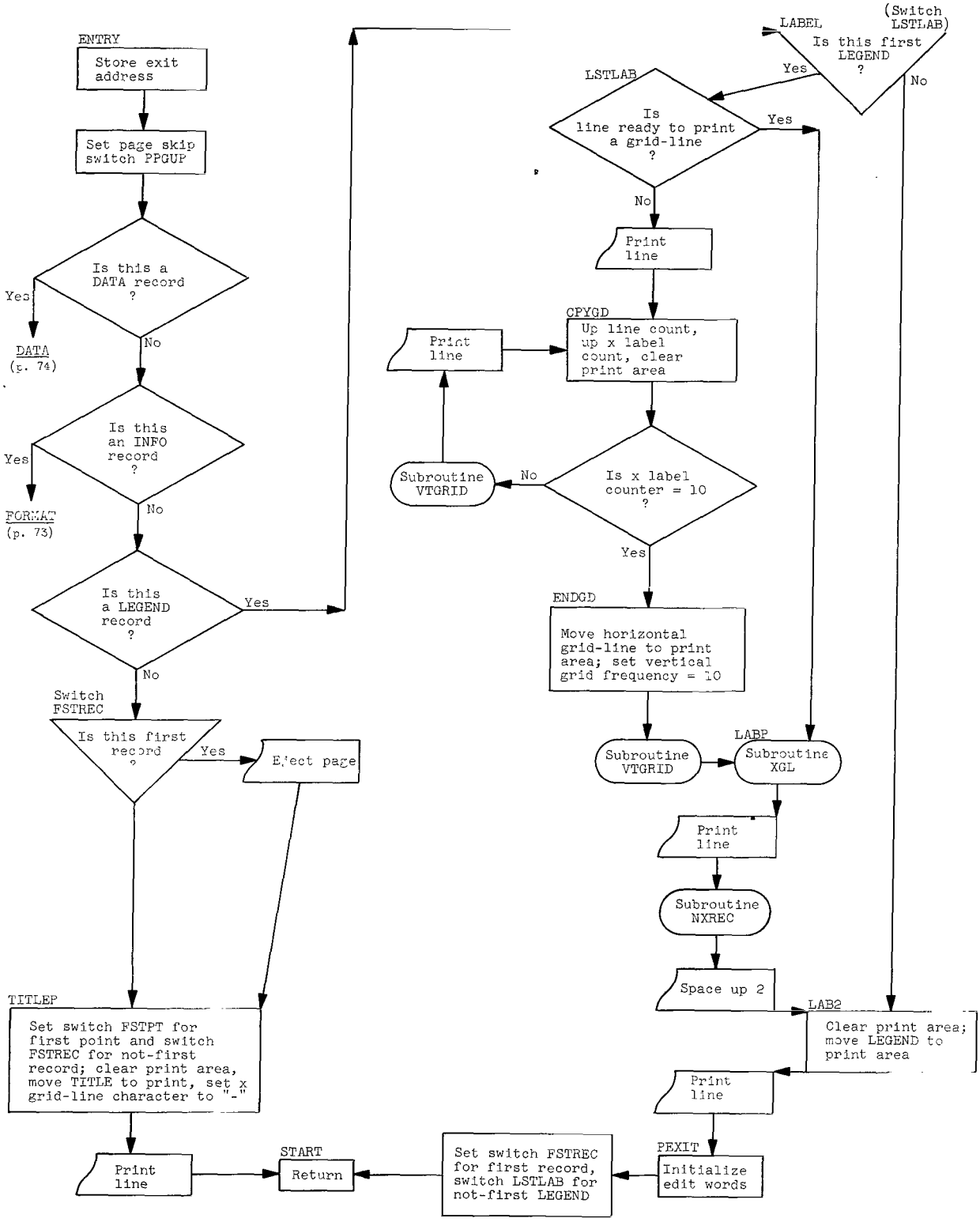
NO.	CC	LABEL	OP	A-ADD.	- - - - - - - DATA DEFINITIONS - - - - - - -	LOCATION
242	01	PLCHAR	DCW	*	INPUT -THE PLOTTING CHARACTER FOR SINGLE CURVES, A MINUS SIGN FOR MULTIPLE CURVES. IT IS THE FIRST POSITION OF IREC.	2245
243	01	PXSH	DCW	*	INPUT - THE SHIFT CODE FOR THE X GRID-LABELS. THE DECIMAL POINT IS MOVED PXSH POSITIONS FROM THE LEFT END OF THE EDIT WORD EDX.	2246
244	01	PYSH	DCW	*	INPUT - THE SHIFT CODE FOR THE Y GRID-LABELS. THE DECIMAL POINT IS MOVED PYSH POSITIONS FROM THE LEFT END OF THE EDIT WORD EDY.	2247
245	02		DCW	*		2249
246	07	XSTART	DCW	*	INPUT - THE NUMBER REPRESENTING THE FIRST X GRID-LABEL.	2256
247	03		DCW	*		2259
248	06	YSTART	DCW	*	INPUT - THE NUMBER REPRESENTING THE FIRST Y GRID-LABEL.	2265
249	03		DCW	*		2268
250	06	DX	DCW	*	INPUT - THE SCALING PARAMETER FOR ONE LINE-SPACE, USED TO COMPUTE THE X GRID-LABELS.	2274
251	03		DCW	*		2277
252	06	DY	DCW	*	INPUT - THE SCALING PARAMETER FOR ONE PRINT POSITION. IT IS REPLACED BY 10DY, THEN USED TO COMPUTE THE Y GRID-LABELS.	2283
253	03	XGD	DCW	*	INPUT - THE X GRID-LINE FREQUENCY.	2286
254	03	YGD	DCW	*	INPUT - THE Y GRID-LINE FREQUENCY.	2289
255	03	FM1	DCW	*	INPUT - THE LENGTH OF THE LEFT-SIDE-FIELD IN THE DATA RECORDS.	2292
256	00	TEMP2	DS	109	TEMPORARY STORAGE USED IN EDITING THE Y GRID-LABELS.	109
257	00	IREC	DS	PLCHAR	THE FIRST POSITION OF THE FIELD IN WHICH THE INFO RECORD IS SAVED.	2245
258	00	PLZERO	DS	230	POSITION (30) OF THE FIRST Y GRID-LINE IN THE PRINT AREA.	230
259	01	MINUS	DC	336	KEEPING THE X GRID-LINE CHARACTER IN THIS POSITION PERMITS A ROW OF MINUS SIGNS TO BE MOVED INTO THE PRINT AREA WITH ONE MOVE COMMAND.	336
260	00	A	DS	89	INDEX REGISTER 1	89
261	00	B	DS	94	INDEX REGISTER 2	94
262	00	C	DS	99	INDEX REGISTER 3	99
263	01	RECORD	DCW	*	THE FIRST POSITON OF THE INPUT BUFFER (LENGTH 630 CHARACTERS).	2293
264	99		DS	*		2392
265	99		DS	*		2491
266	99		DS	*		2590
267	99		DS	*		2689
268	99		DS	*		2788
269	99		DS	*		2887
270	35		DS	*		2922
271	00	REC	DS	RECORD	REC IS EQUIVALENT TO RECORD	2293
272	01	GRPMK	DCW	* ‡	THIS GROUP-MARK WORD-MARK SIGNALS THE END OF THE INPUT BUFFER	2923

LABEL TABLE

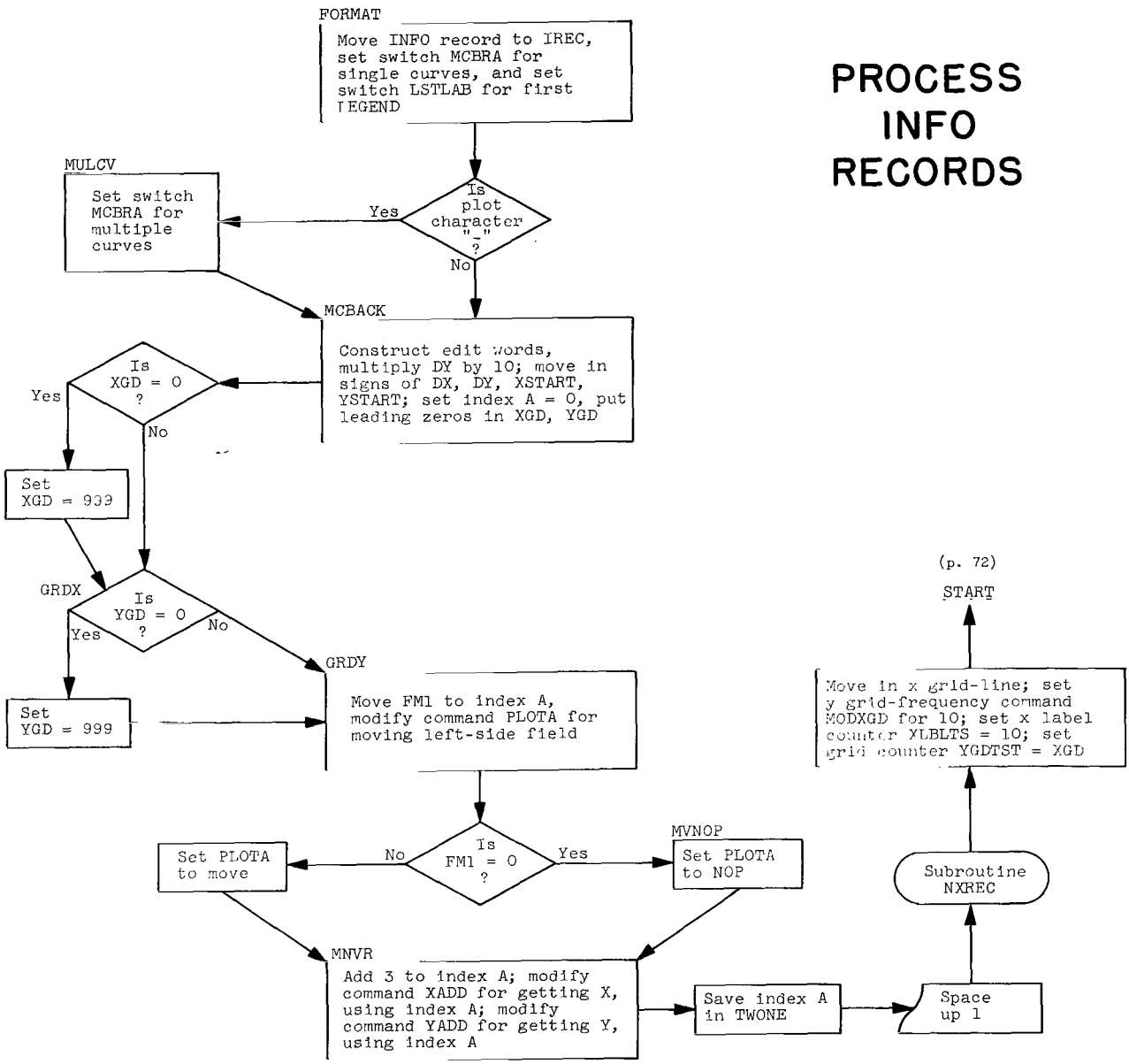
000089	A	1084	1279	1319	1334	1356	1375	1421	1514	1546	1567	1670	1677
		1691	1754	1769	1788	1807	1814	1822	1841	1848	1856	1863	
000094	B	1247	1255	1428	1484	1506	1521	2036	2050	2057			
001228	CHGSIG	0											
001939	CPYGD	1969											
000099	C	0											
001084	DATA	1027											
001754	DXPL	0											
002274	DX	1221	1740	1740	1939								
001191	DXTST	0											
002283	DY	1560	1726	1733	1747	1747							
002214	DYT	1726	1733										
002165	EDIT	1000	1073										
002156	EDX	1000	1684	2111									
002147	EDY	1462	1698										
001462	EDYST	1442											
001973	ENDGD	1960											
001583	ENDNR	1405	1574										
001356	ENDTST	1326											
001019	ENTRY	0											
002292	FM1	1807											
001648	FORMAT	1035											
001092	FSTPT	1054	1096										
001052	FSTREC	1008	1054	2021									
001788	GRDX	1776											
001807	GRDY	1795											
002923	GRPMK	0											
002238	I99	1528											
001100	INTX	0											
002245	IREC	1648											
001414	LABOUT	0											
002005	LAB2	1921											
001920	LABEL	1043											
001995	LABP	1925											
002195	LASTX	1100	1172	1184	1199								
001921	LSTLAB	1008	1655										
001326	LTYTST	0											
001303	MANYY	1268	1363	1371									
001670	MCBACK	1644											
001268	MCBRA	1393	1640	1655									
000336	MINUS	1139	1888	1977									
001104	MIYRET	1228											
001841	MNVR	2028											
001371	MODA	1304											
002050	MODXGD	1895	1984	2069									
001640	MULCV	1662											
001308	MUPLC	1382											
001539	MVLAB	1506											
002021	MVNOP	1829											
001139	NOYGD	0											
002189	NINS	1781	1800										
001495	NONB	1476											
002064	NOXGDA	0											
001405	NXREC	1878	1999										
001428	NXTLAB	1579											
002224	ONE	1212	1946										
002227	ONETEN	1567	2057										

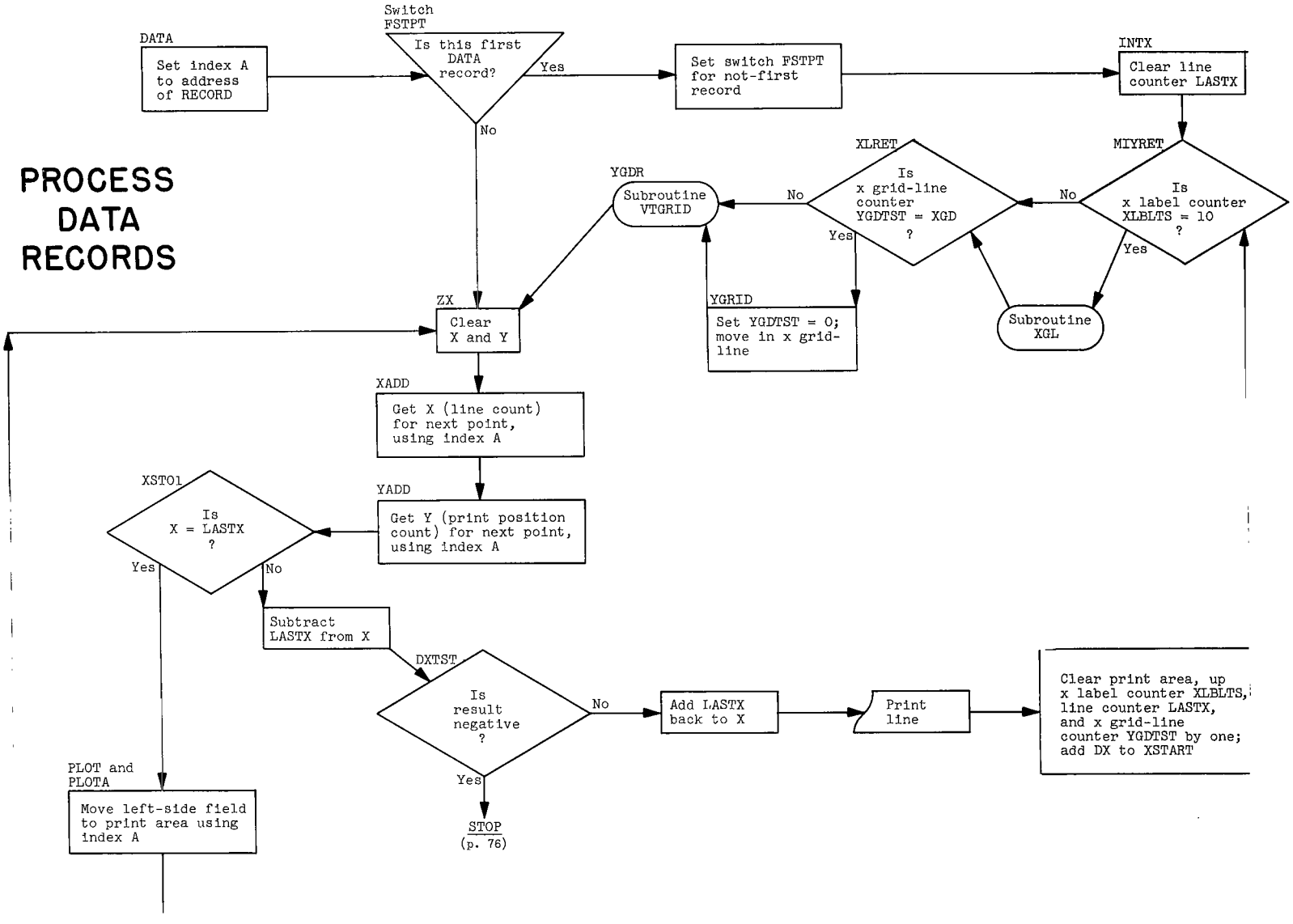
001286	ONWARD	1367											
002138	PERZ	1684	1698										
001000	PEXIT	1624	2017										
002245	PLCHAR	1272	1662										
001236	PLOTA	1814	1834	2021									
001279	PLOTBK	1401											
001232	PLOT	1179											
000230	PLZERO	1128	1272	1308	1312	1447	1454	1539	1883	1973	2043	2096	2103
		2111	2118										
001619	PNEX	1628											
002136	PPGUP	1023											
002246	PXSH	1677											
002247	PYSH	1691											
002293	RECORD	0											
002293	REC	1027	1035	1043	1066	1084	1603	1611	1628	1648	2010		
002132	RETXL	2080	2103										
002228	RM	1386											
001628	SKPON	1611											
001603	SKREC	1636											
001015	START	1019	1080	1286	1916								
001587	STOP	1191											
002235	TEMP1	1414	1435	1447	1469	1560							
000109	TEMP2	1462	1469	1476	1495	1499	1499	1539					
002217	TEN	1104	1528	1546	1553	1895	1902	1953	1984				
001054	TITLEP	0											
001179	TNXTX	0											
001386	TOBIG	1262											
001476	TSTB	1491											
002244	TWONE	1279	1375	1871									
001319	UPA	1393											
001606	UPB	1535											
002218	VERT	2043											
002032	VTGRID	1146	1965	1991									
001158	XADD	1848											
002286	XGD	1116	1769	1781	1909	2050							
002111	XGLA	2091											
002080	XGL	1111	1995										
002043	XGRID	2064											
002198	XLBLTS	1104	1212	1902	1946	1953	2125						
001116	XLRET	0											
002180	XOFORD	0											
002183	X	1158	1172	1184	1191	1199							
001172	XST01	0											
002256	XSTART	1221	1705	1705	1719	1939	2084	2096	2118				
002096	XZRTST	0											
001165	YADD	1863											
001146	YGDR	1123											
002289	YGD	1761	1788	1800	2069								
002192	YGDTST	1116	1132	1909									
001128	YGRID	0											
002076	YGXIT	2032											
002241	YLAST	1255											
002186	Y	1150	1154	1165	1247	1341	1345	1593					
001247	YST01	1352											
001255	YST02	0											
002265	YSTART	1414	1712	1712									
001546	YZROK	1454											
001447	YZRTST	0											
002205	ZTST	1435	1719	1761	1822	2084							
001150	ZX	1092	1299										

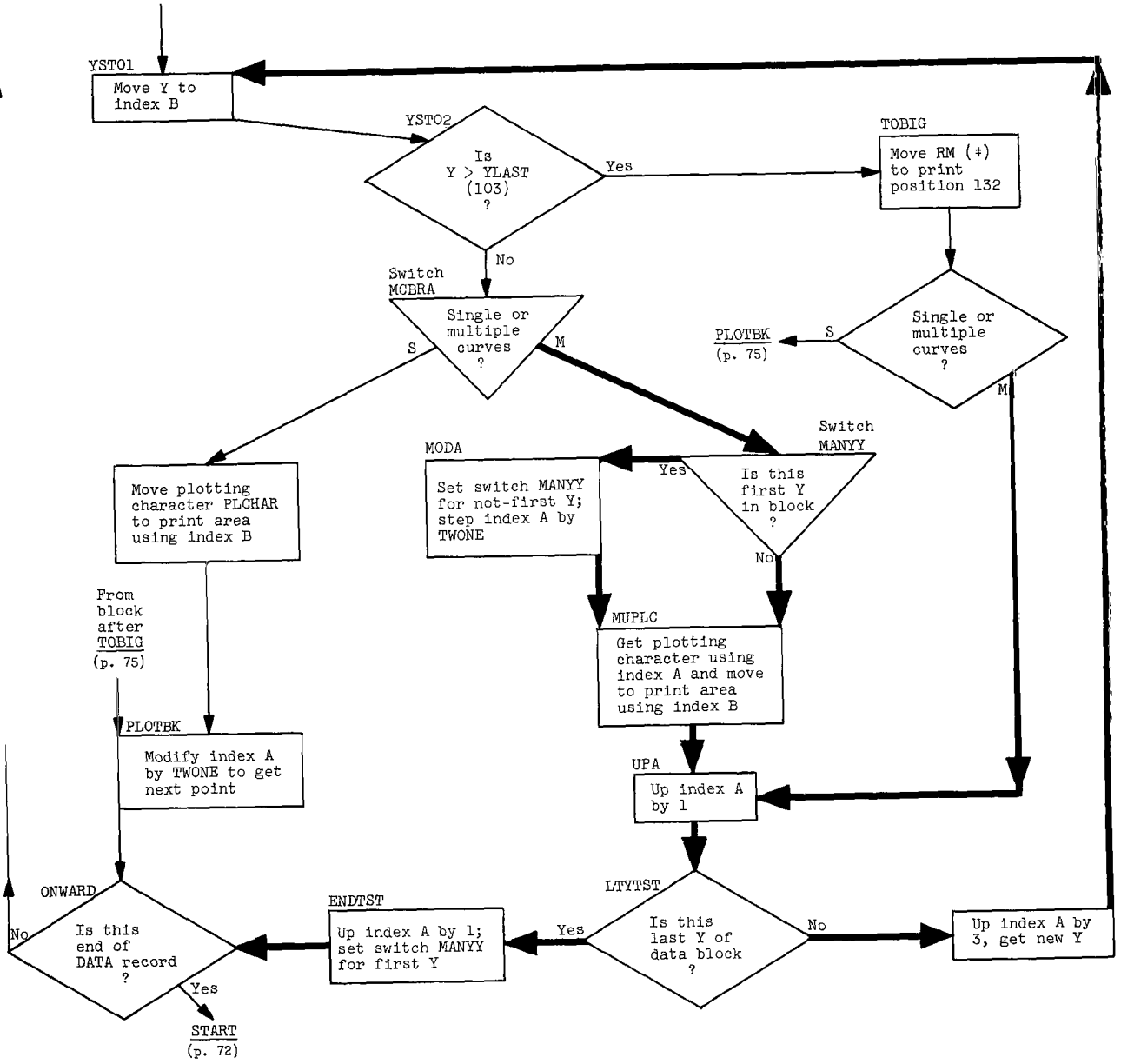
PLOT


ENTRY
Store exit address
Set page skip switch PPGUP
Is this a DATA record ?
Yes
DATA (p. 74)
No
Is this an INFO record ?
Yes
FORMAT (p. 73)
No
Is this a LEGEND record ?
Yes
No
Switch FSTREC
Is this first record ?
Yes
Eject page
TITLEP
Set switch FSTPT for first point and switch FSTREC for not-first record; clear print area, move TITLE to print, set x grid-line character to "-"
Print line
START
Return
Set switch FSTREC for first record, switch LSTLAB for not-first LEGEND
PEXIT
Initialize edit words
LABEL
(Switch LSTLAB)
Is this first LEGEND ?
Yes
No
LSTLAB
Is line ready to print a grid-line ?
Yes
No
Print line
CPYGD
Up line count, up x label count, clear print area
Print line
Is x label counter = 10 ?
No
Subroutine VTGRID
Yes
ENDGD
Move horizontal grid-line to print area; set vertical grid frequency = 10
Subroutine VTGRID
LABP
Subroutine XGL
Print line
Subroutine NXREC
Space up 2
LAB2
Clear print area; move LEGEND to print area
Print line

PROCESS INFO RECORDS

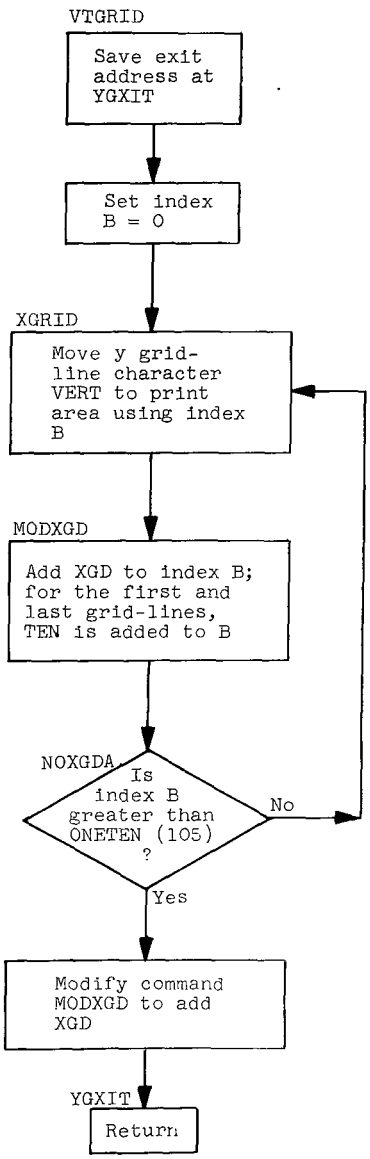



PROCESS DATA RECORDS
DATA
Set index A to address of RECORD
Switch FSTPT
Is this first DATA record?
Yes
No
Set switch FSTPT for not-first record
INTX
Clear line counter LASTX
MIYRET
Is x label counter XLBLTS = 10 ?
No
Yes
Subroutine XGL
XLRET
Is x grid-line counter YGDTST = XGD ?
No
Yes
YGDR
Subroutine VTGRID
YGRID
Set YGDTST = 0; move in x grid-line
ZX
Clear X and Y
XADD
Get X (line count) for next point, using index A
YADD
Get Y (print position count) for next point, using index A
XSTO1
Is X = LASTX ?
Yes
No
Subtract LASTX from X
DXTST
Is result negative ?
No
Yes
Add LASTX back to X
Print line
Clear print area, up x label counter XLBLTS, line counter LASTX, and x grid-line counter YGDTST by one; add DX to XSTART
PLOT and PLOTA
Move left-side field to print area using index A
STOP
(p. 76)


YSTO1
Move Y to index B
YSTO2
Is Y > YLAST (103) ?
Yes
TOBIG
Move RM (‡) to print position 132
No
Switch MCBRA
Single or multiple curves ?
S
M
PLOTBK (p. 75)
S
Single or multiple curves ?
M
Switch MANYY
MODA
Set switch MANYY for not-first Y; step index A by TWONE
Yes
Is this first Y in block ?
No
Move plotting character PLCHAR to print area using index B
From block after TOBIG (p. 75)
MUPLC
Get plotting character using index A and move to print area using index B
PLOTBK
Modify index A by TWONE to get next point
UPA
Up index A by 1
ONWARD
No
Is this end of DATA record ?
Yes
START (p. 72)
ENDTST
Up index A by 1; set switch MANYY for first Y
Yes
LTYTST
Is this last Y of data block ?
No
Up index A by 3, get new Y
Indicates program path used exclusively for multiple curves.

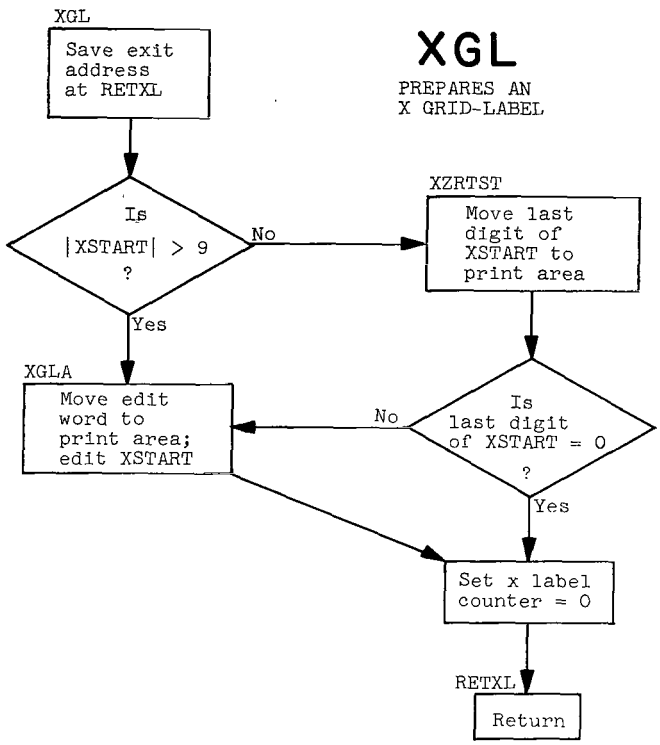
VTGRID

INSERTS VERTICAL GRID-LINES

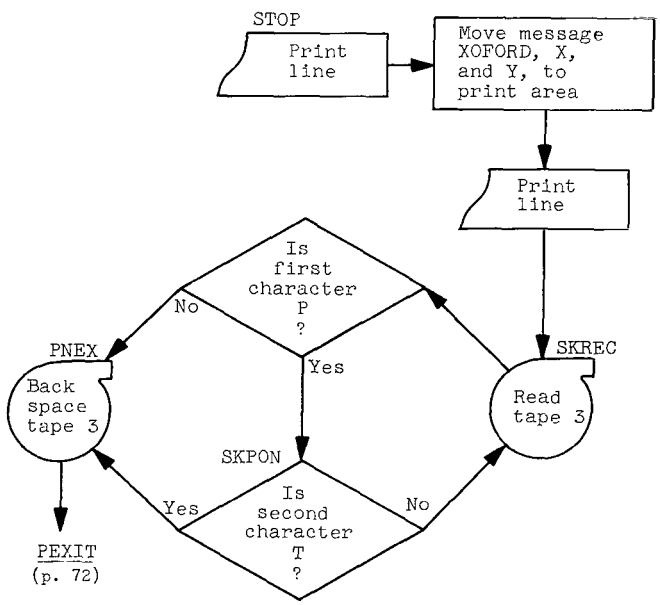


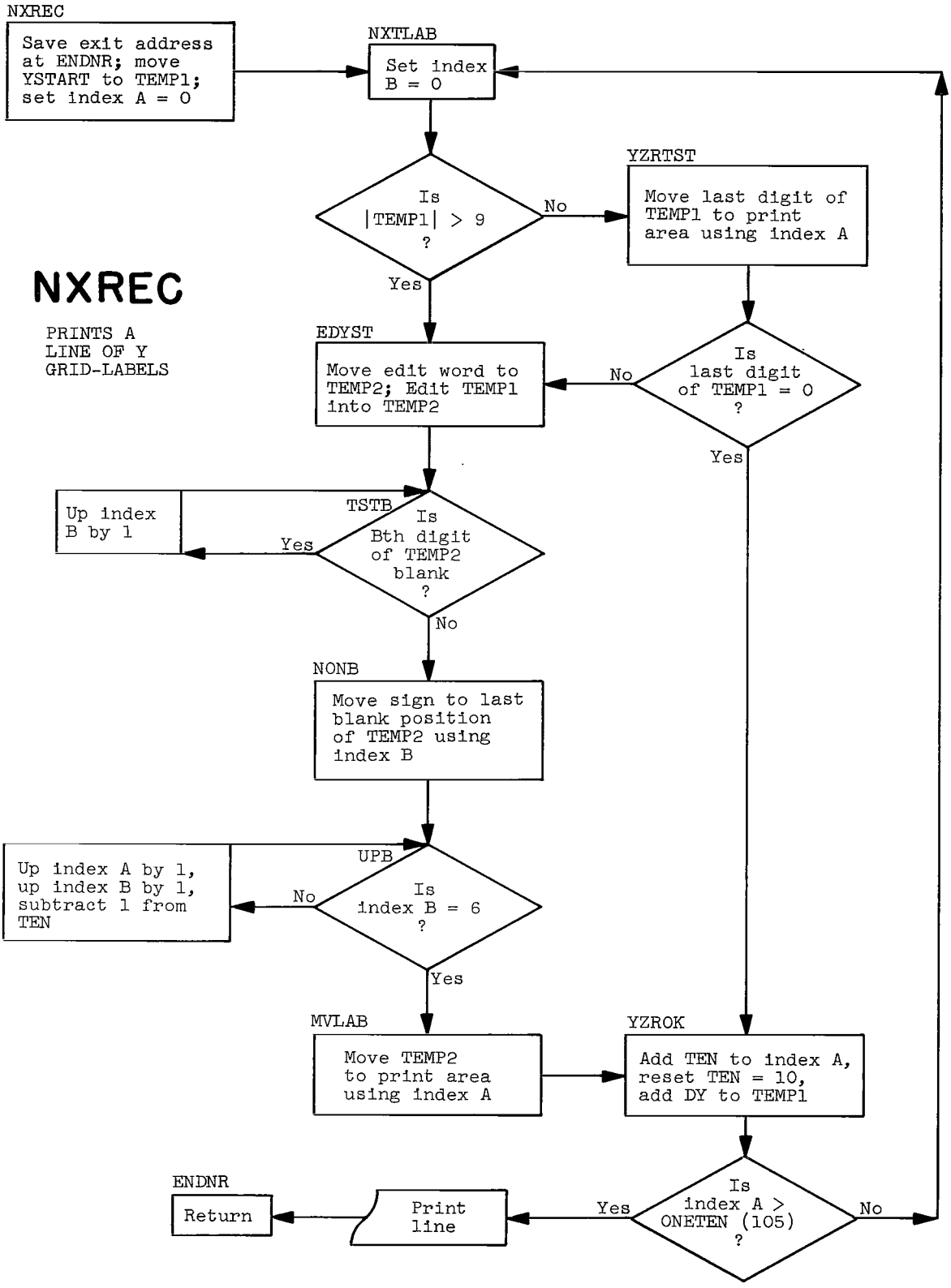
XGL

PREPARES AN X GRID-LABEL



ERROR EXIT



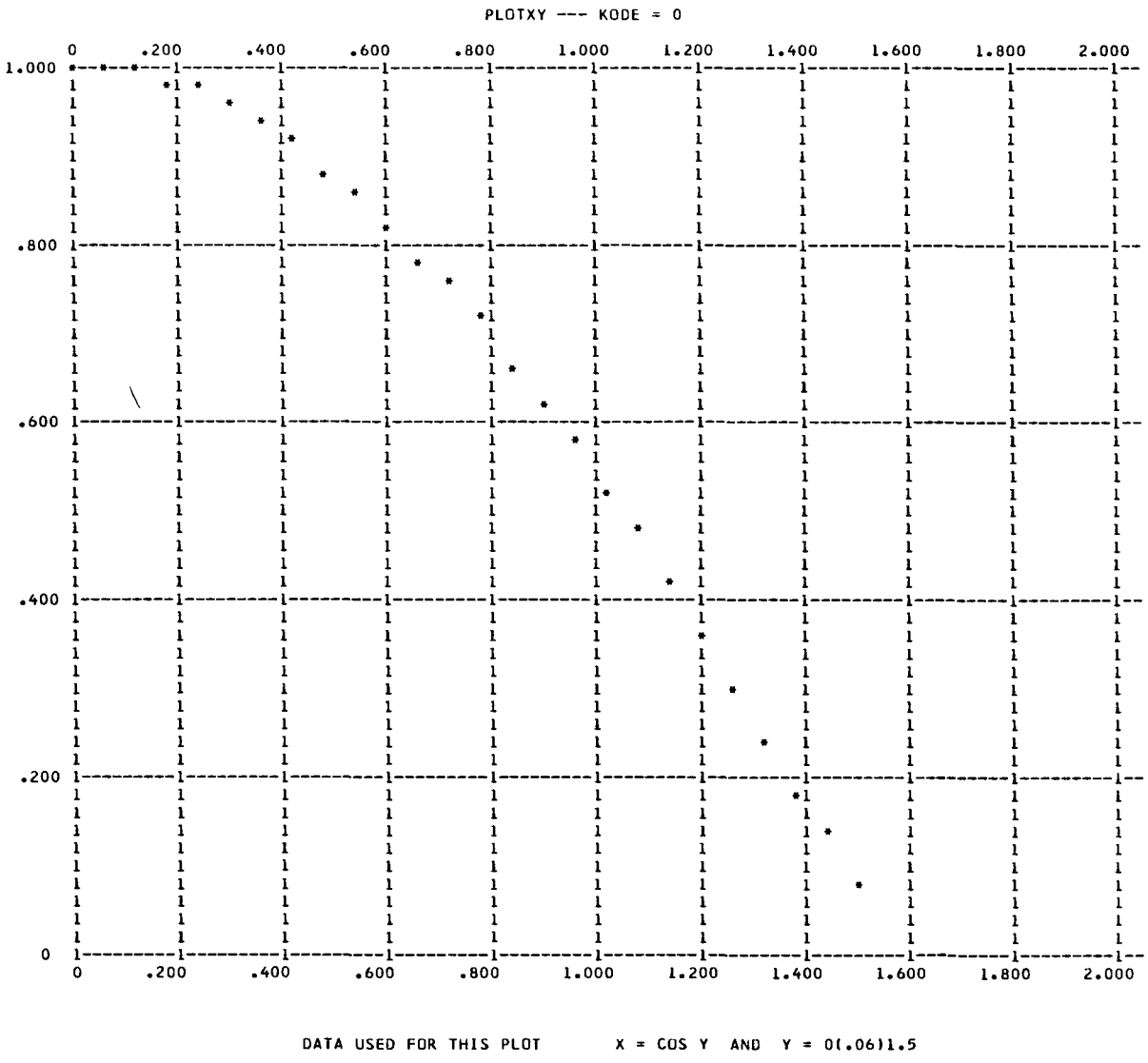

NXREC
Save exit address at ENDNR; move YSTART to TEMP1; set index A = 0
NXTLAB
Set index B = 0
Is |TEMP1| > 9 ?
No
YZRTST
Move last digit of TEMP1 to print area using index A
Yes
NXREC
PRINTS A LINE OF Y GRID-LABELS
EDYST
Move edit word to TEMP2; Edit TEMP1 into TEMP2
No
Is last digit of TEMP1 = 0 ?
Yes
TSTB
Up index B by 1
Yes
Is Bth digit of TEMP2 blank ?
No
NONB
Move sign to last blank position of TEMP2 using index B
UPB
Up index A by 1, up index B by 1, subtract 1 from TEN
No
Is index B = 6 ?
Yes
MVLAB
Move TEMP2 to print area using index A
YZROK
Add TEN to index A, reset TEN = 10, add DY to TEMP1
ENDNR
Return
Print line
Yes
Is index A > ONETEN (105) ?
No

APPENDIX A

SAMPLE PLOTS

The plots in figures 1 to 12 on the following pages were produced by specially written FORTRAN programs to illustrate some of the possibilities of the OPP system and to provide sample tape records for the 1401 systems programmer. The same 7090 output tape read by the 1401 when the plots were generated was also read by a specially written tape-dump program to present the associated tape records exactly as they appear on the tape, except for the insertion of blank lines between records for easier reading.

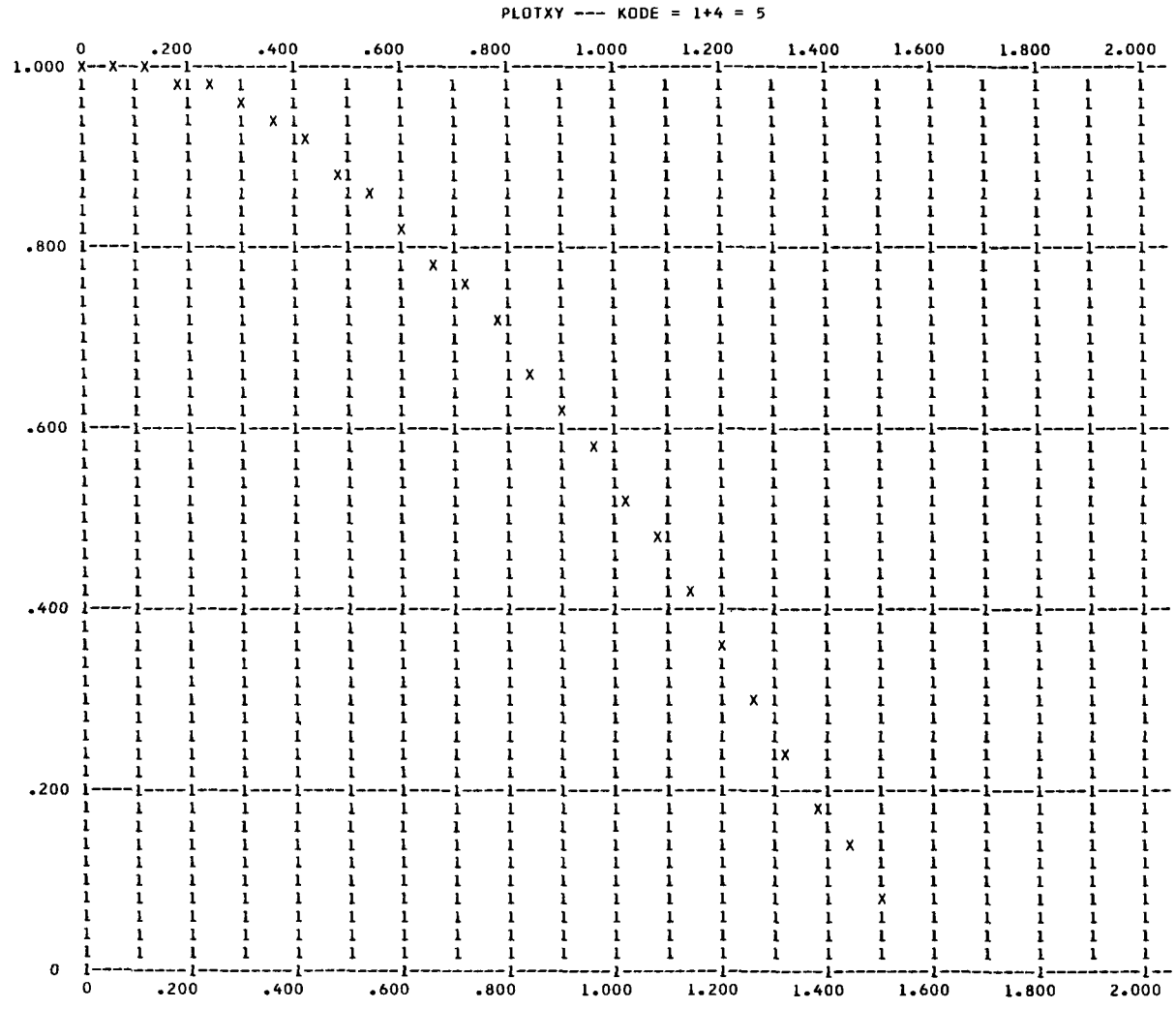
FIGURE 1.


PLOTXY --- KODE = 0
0
.200
.400
.600
.800
1.000
1.200
1.400
1.600
1.800
2.000
1.000
.800
.600
.400
.200
0
DATA USED FOR THIS PLOT X = COS Y AND Y = 0(.06)1.5

```
PT   FIGURE   1.  *
PT                                                    PLOTXY --- KODE = 0

PI*33 0.001000 0.000000-0.000020 0.000020 10 10  0    *

PD  0  0  0  3  0  6  1  9  1 12  2 15  3 18  4 21  6 24  7 27  9 30 11 33 12 36 14 39 17 42 19 45 21 48 24 51 26 54 29 57 32 60
PD 35 63 38 66 41 69 43 72 46 75    *

PL                                          DATA USED FOR THIS PLOT       X = COS Y  AND  Y = 0(.06)1.5            *
```

Figure 1. - This plot illustrates using PLOTXY with KODE = 0. The subroutine provides 10 × 10 grid-line spacing, chooses both scales, and uses an asterisk as the plotting character. The number of points (26.) was placed in P(1).

FIGURE 2.

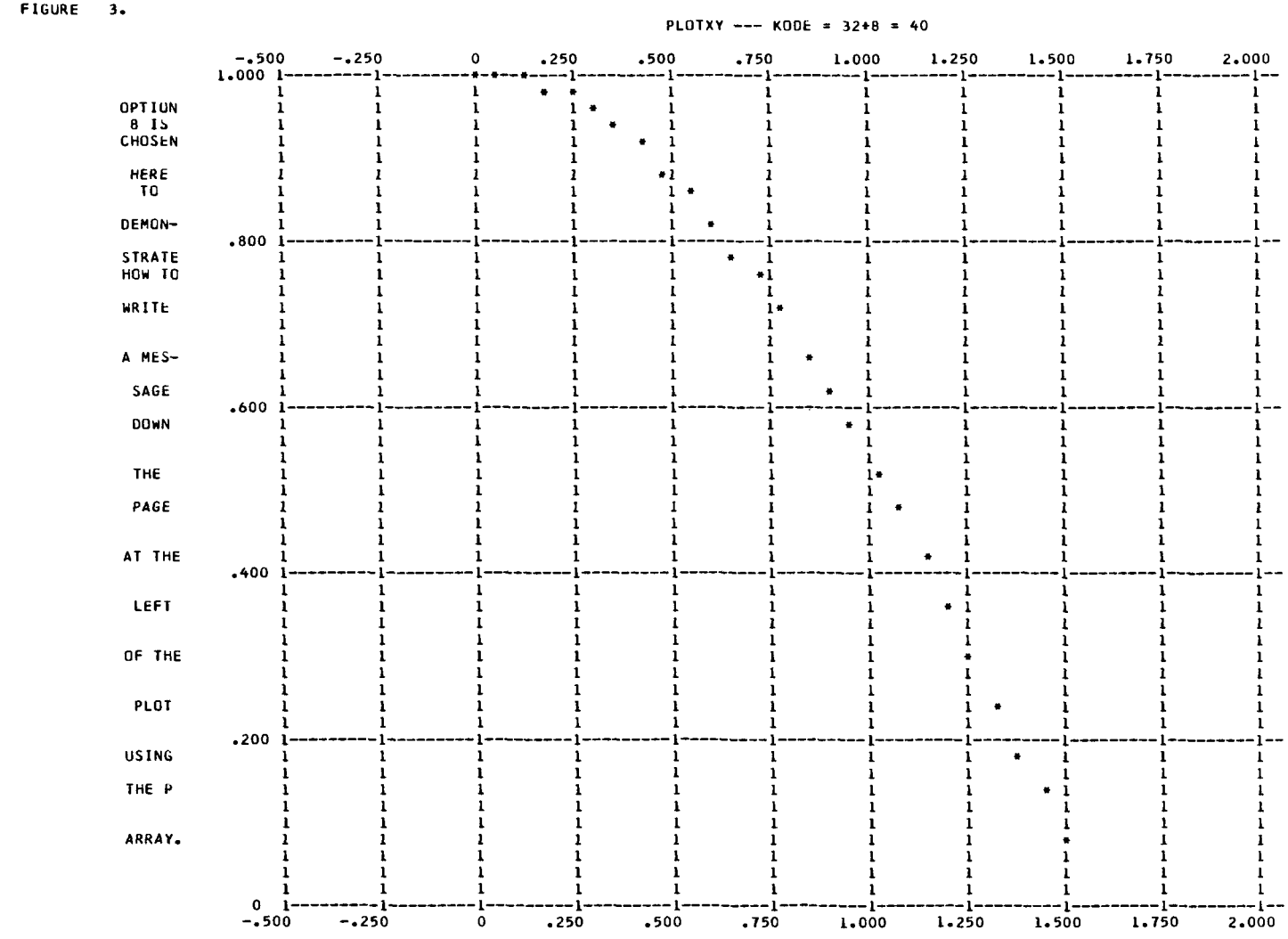


(SAME DATA AS IN FIGURE 1.)

```
PT   FIGURE   2.  *
PT                                                    PLOTXY --- KODE = 1+4 = 5      *

PIX33 0.001000 0.000000-0.000020 0.000020 10  5  0    *

PD  0  0  0  3  0  6  1  9  1 12  2 15  3 18  4 21  6 24  7 27  9 30 11 33 12 36 14 39 17 42 19 45 21 48 24 51 26 54 29 57 32 60
PD 35 63 38 66 41 69 43 72 46 75    *

PL                                                    (SAME DATA AS IN FIGURE 1.)      *
```

Figure 2. - This plot illustrates using option 1 - choice of plotting character. The desired plotting character, X, was placed in P(2). Option 4 - choice of y grid-line frequency - was also used. The desired frequency (5.) was placed in P(4). Notice that the frequency of y grid-labels is not affected.

FIGURE 3.



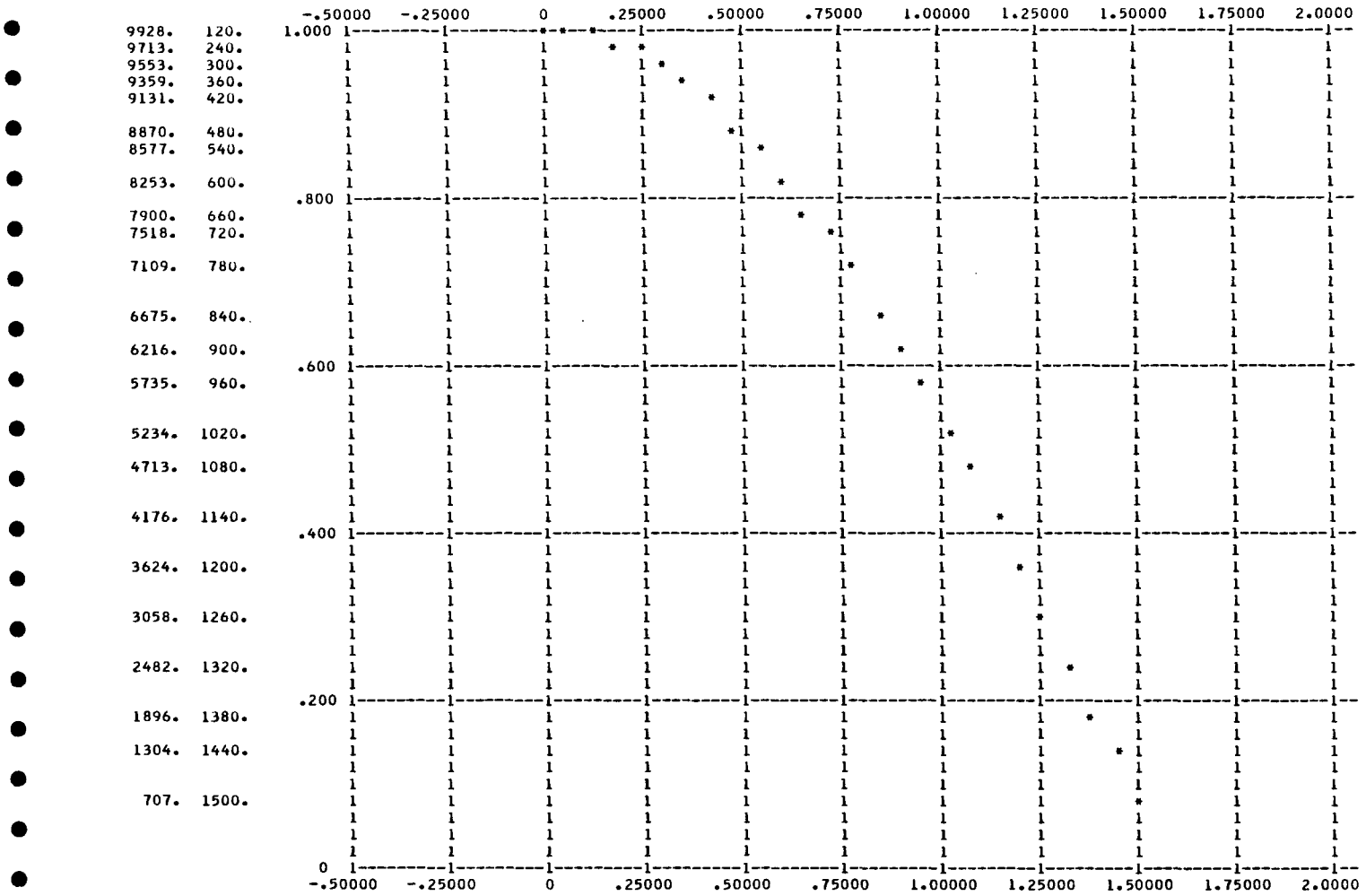
(SAME DATA AS IN FIGURE 1.)

```
PT   FIGURE   3.  *
PT                                                    PLOTXY --- KODE = 32+8 = 40    *

PI*33 0.001000-0.000500-0.000020 0.000025 10 10  6    *

PD        0 20        0 22        0 25        1 27        1 30OPTION  2 32 8 IS   3 34CHOSEN  4 37 HERE   6 39  TO    7 42    *
PDDEMON-  9 44STRATE 11 46HOW TO 12 49WRITE  14 51A MES- 17 54 SAGE  19 56 DOWN  21 58 THE   24 61 PAGE  26 63AT THE 29 66    *
PD LEFT  32 68OF THE 35 70 PLOT  38 73USING  41 75THE P  43 78ARRAY. 46 80    *

PL                                                    (SAME DATA AS IN FIGURE 1.)       *
```

Figure 3. - This plot illustrates using option 32 - choice of y scale. The scale chosen was specified to PLOTXY by placing KSY = 3. in P(9), FY = -500. in P(10), and DY = 25. in P(11). The maximum possible value of KSY was used. (Fig. 4 displays the effect of using a smaller value of KSY.) Also illustrated is the use of option 8 - choice of printing left-side field. The message was placed in the P array; the characters bbbbbb in P(12) through P(15), the characters OPTION in P(16); the characters b8bISb in P(17); etc.

FIGURE 4.

PLOTXY --- KODE = 64+32 = 96

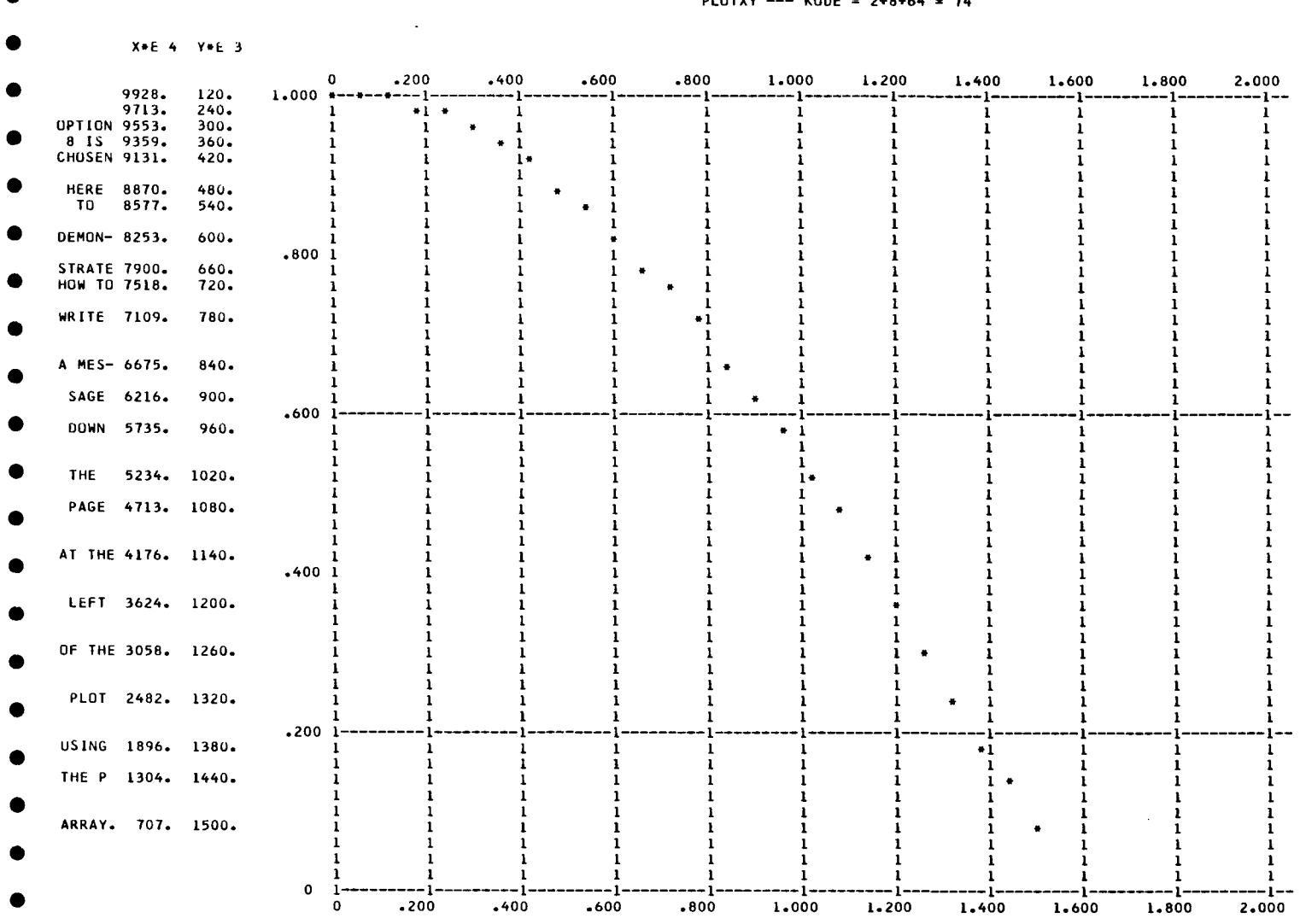


(SAME DATA AS IN FIGURE 1.)

```
PT   FIGURE   4.  *
PT                                                    PLOTXY --- KODE = 64+32 = 96   *
PT                *
PT                *
PT        X*E 4  Y*E 3  *

PI*31 0.001000-0.050000-0.000020 0.002500 10 10 13    *

PD 9990.     0.  0 20 9982.    60.  0 22 9928.   120.  0 25 9838.   180.  1 27 9713.   240.  1 30 9553.   300.  2 32   *
PD 9359.   360.  3 34 9131.   420.  4 37 8870.   480.  6 39 8577.   540.  7 42 8253.   600.  9 44 7900.   660. 11 46   *
PD 7518.   720. 12 49 7109.   780. 14 51 6675.   840. 17 54 6216.   900. 19 56 5735.   960. 21 58 5234.  1020. 24 61   *
PD 4713.  1080. 26 63 4176.  1140. 29 66 3624.  1200. 32 68 3058.  1260. 35 70 2482.  1320. 38 73 1896.  1380. 41 75   *
PD 1304.  1440. 43 78  707.  1500. 46 80  *

PL                                                    (SAME DATA AS IN FIGURE 1.)      *
```

Figure 4. - This plot illustrates using option 64 - choice of printing coordinates at left. Because of the limited field size allotted to this print-out, PLOTXY transforms the coordinates to preserve as many significant figures as possible, and indicates in a special heading what has been done. This plot also illustrates using option 32 - choice of y scale. The same starting-value and scale-factor were wanted as in figure 3, but the values KSY = 1. were placed in P(9), FY = -50000. in P(10), and DY = 2500. in P(11). Notice the corresponding change in the appearance of the y grid-labels.

FIGURE 5.

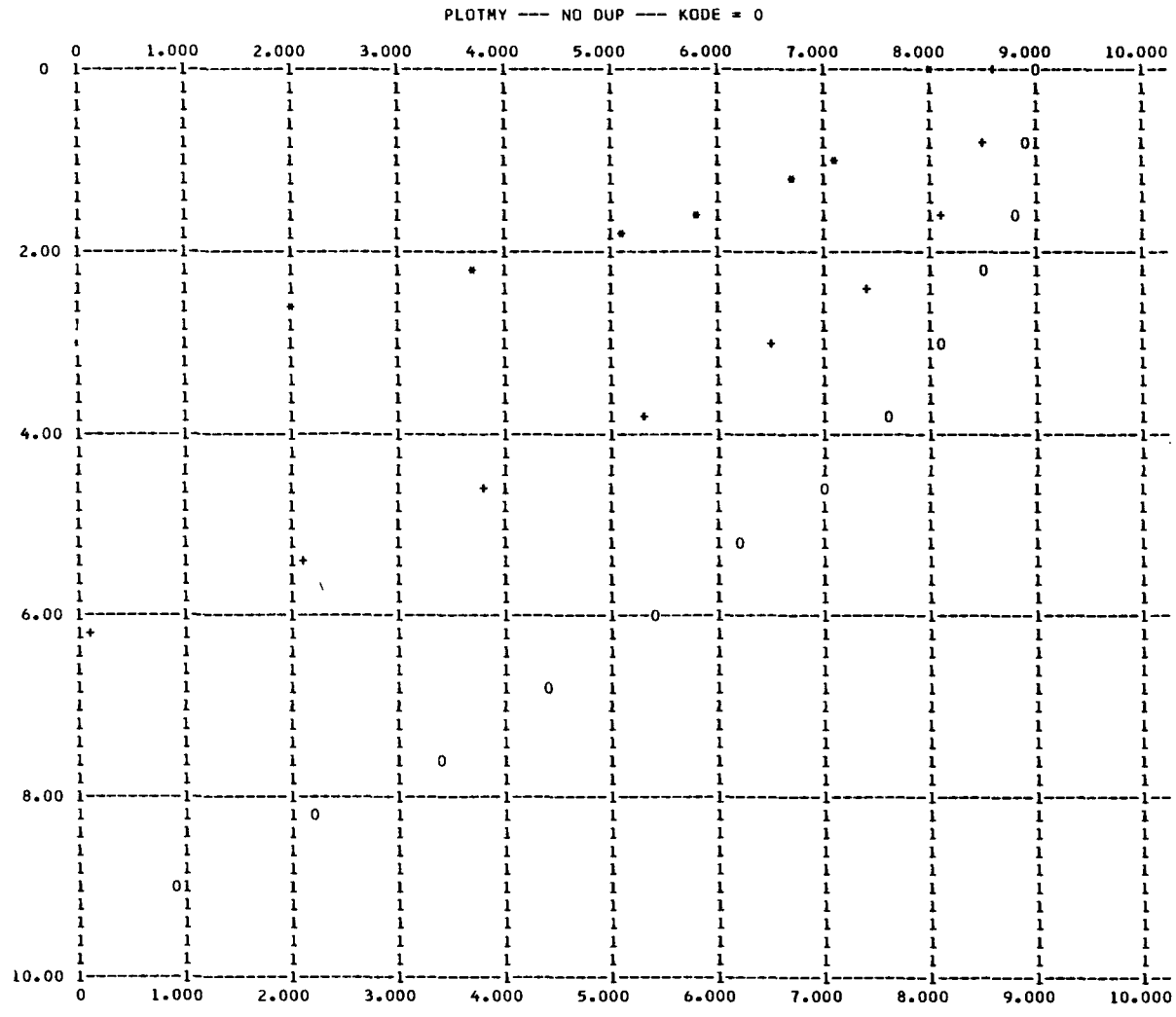


(SAME DATA AS IN FIGURE 1.)

```
PT   FIGURE   5.  *
PT                                                    PLOTXY --- KODE = 2+8+64 = 74  *
PT                *
PT                *
PT        X*E 4  Y*E 3  *


PI*33 0.001000 0.000000-0.000020 0.000020 20 10 19


PD       9990.     0.  0  0       9982.    60.  0  3       9928.   120.  0  6       9838.   180.  1  9       9713.   240.  1 12
PDOPTION 9553.   300.  2 15 8 IS  9359.   360.  3 18CHOSEN 9131.   420.  4 21 HERE  8870.   480.  6 24  TO   8577.   540.  7 27
PDDEMON- 8253.   600.  9 30STRATE 7900.   660. 11 33HOW TO 7518.   720. 12 36WRITE  7109.   780. 14 39A MES- 6675.   840. 17 42
PD SAGE  6216.   900. 19 45 DOWN  5735.   960. 21 48 THE   5234.  1020. 24 51 PAGE  4713.  1080. 26 54AT THE 4176.  1140. 29 57
PD LEFT  3624.  1200. 32 60OF THE 3058.  1260. 35 63 PLOT  2482.  1320. 38 66USING  1896.  1380. 41 69THE P  1304.  1440. 43 72
PDARRAY.  707.  1500. 46 75  *


PL                                                    (SAME DATA AS IN FIGURE 1.)       *
```

Figure 5. - This plot illustrates using option 2 - choice of x grid-line frequency. The desired frequency (20.) was placed in P(3). Notice that the frequency of the x grid-labels is not affected. This plot also demonstrates that after the last point has been plotted, the grid is only continued to the next grid-label, not to the next grid-line. The result of using options 8 and 64 simultaneously should be compared with figures 3 and 4.

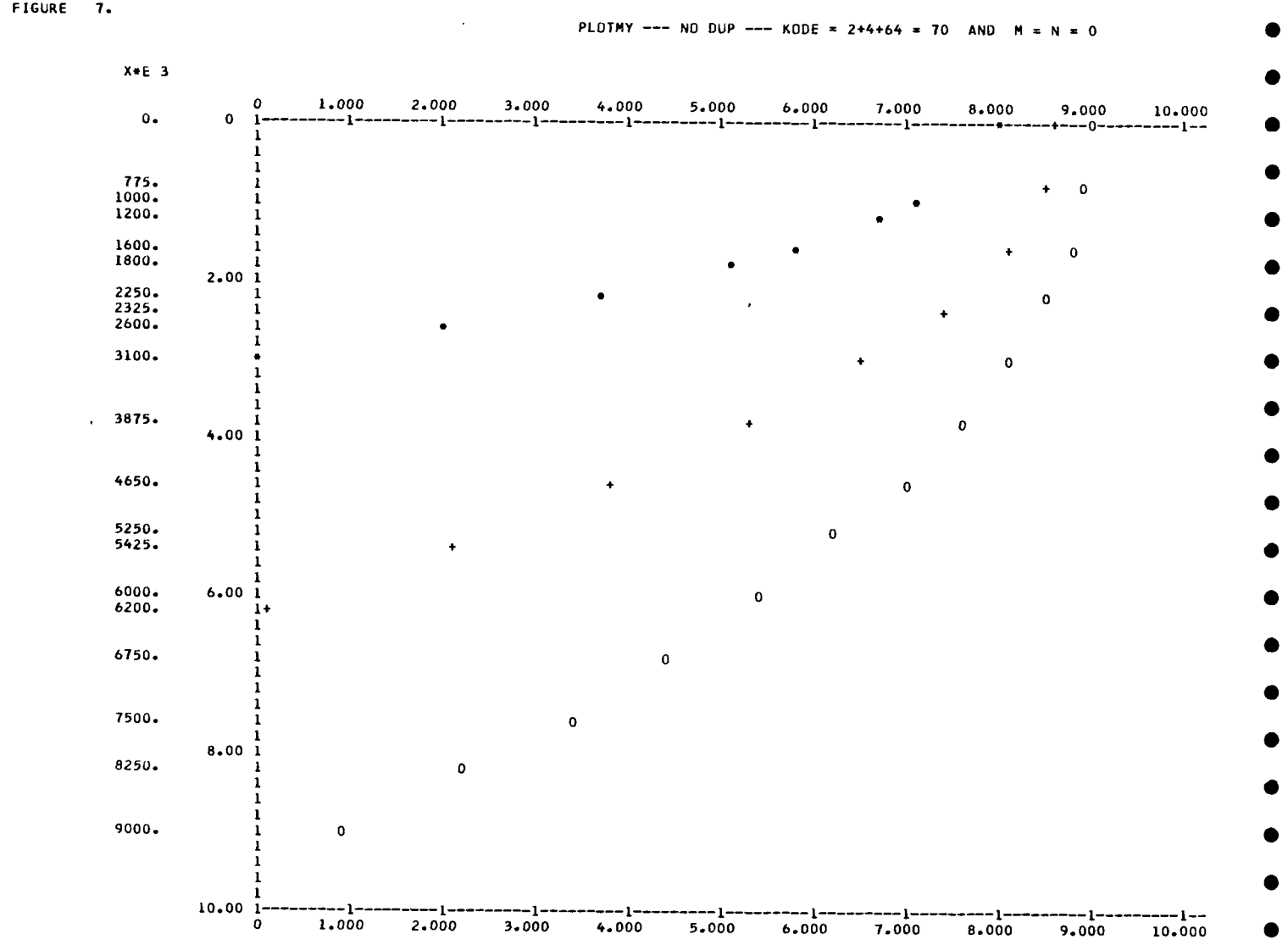
FIGURE 6.


PLOTMY --- NO DUP --- KODE = 0
0
1.000
2.000
3.000
4.000
5.000
6.000
7.000
8.000
9.000
10.000
2.00
4.00
6.00
8.00
10.00

```
PT   FIGURE   6.  *
PT                                                    PLOTMY --- NO DUP --- KODE = 0      *

PI-43 0.000000 0.000000 0.000020 0.000100 10 10  0    *

PD  0 80*E  0 86+E  0 900E  4 890E  4 85+E  5 71*E  6 67*E  8 880E  8 81+E  8 58*E  9 51*E 11 37*E 11 850E 12 74+E 13 20*E 15  0*E
PD 15 810E 15 65+E 19 760E 19 53+E 23 700E 23 38+E 26 620E 27 21+E 30 540E 31  1+E 34 440E 38 340E 41 220E 45  90E*

PL                *
```

Figure 6. - This plot illustrates using PLOTMY - Variation III - NO DUP with KKK(1) = KODE = 0. As in PLOTXY, the grid-line spacing 10 × 10 is supplied by the routine, as are the scales and the plotting characters. The number of curves (3) was placed in KKK(2), the number of points for the first curve in KKK(3), for the second in KKK(5), and for the third in KKK(7). Variation III was specified by placing 5. in P(1). Notice that several points fall on the grid-lines and compare with figure 7.

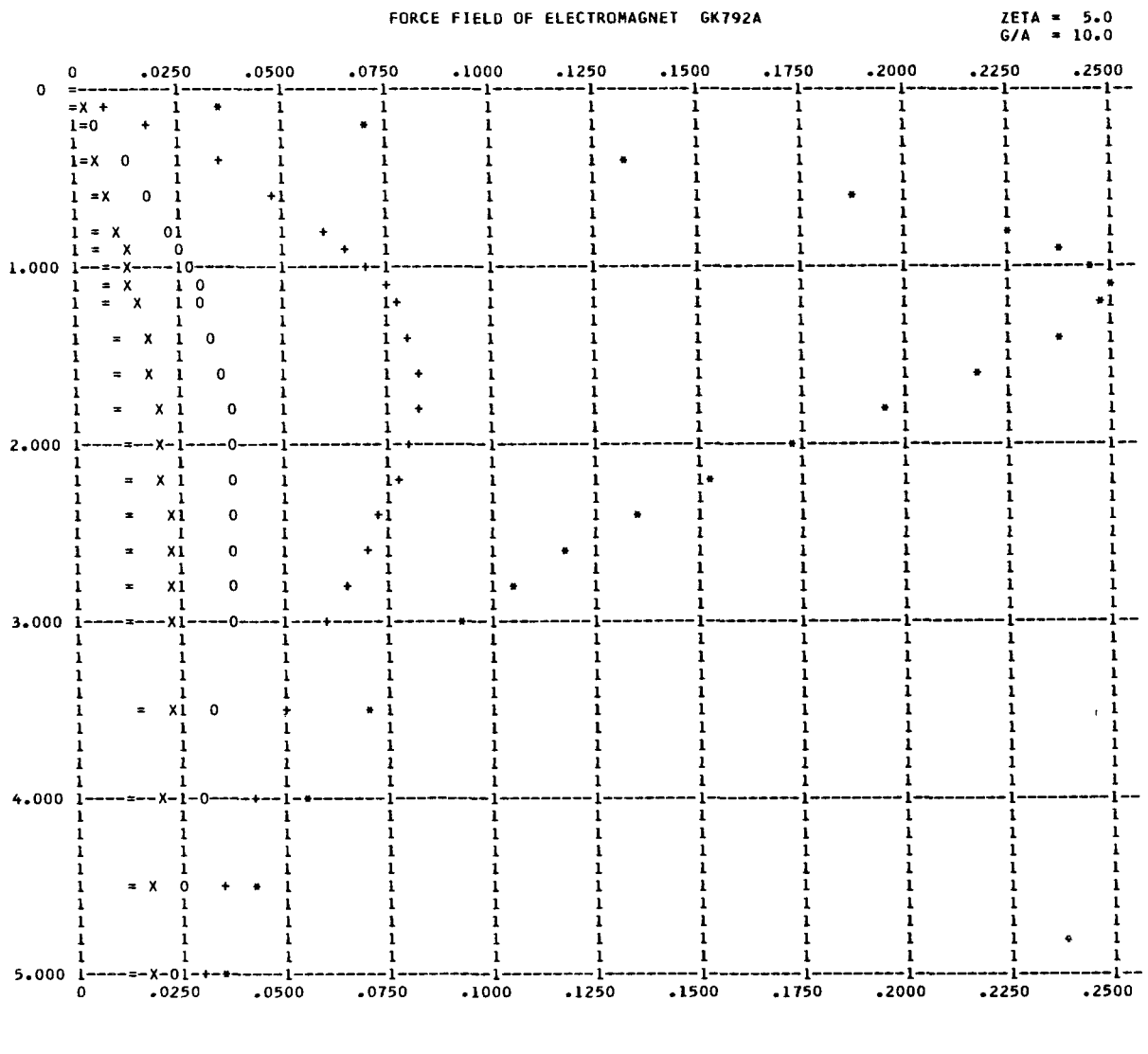
FIGURE 7.


PLOTMY --- NO DUP --- KODE = 2+4+64 = 70 AND M = N = 0
X*E 3
0.
775.
1000.
1200.
1600.
1800.
2250.
2325.
2600.
3100.
3875.
4650.
5250.
5425.
6000.
6200.
6750.
7500.
8250.
9000.
0
2.00
4.00
6.00
8.00
10.00
0 1.000 2.000 3.000 4.000 5.000 6.000 7.000 8.000 9.000 10.000

```
PT   FIGURE   7.  *
PT                                                    PLOTMY --- NO DUP --- KODE = 2+4+64 = 70  AND  M = N = 0    *
PT                *
PT                *
PT               X*E 3  *


PI-43 0.000000 0.000000 0.000020 0.000100  0  0  7    *

PD      0.  0 80*E      0.  0 86+E      0.  0 900E    750.  4 890E    775.  4 85+E   1000.  5 71*E   1200.  6 67*E   1500.  8 880E    *
PD   1550.  8 81+E   1600.  8 58*E   1800.  9 51*E   2200. 11 37*E   2250. 11 850E   2325. 12 74+E   2600. 13 20*E   3000. 15  0*E    *
PD   3000. 15 810E   3100. 15 65+E   3750. 19 760E   3875. 19 53+E   4500. 23 700E   4650. 23 38+E   5250. 26 620E   5425. 27 21+E    *
PD   6000. 30 540E   6200. 31  1+E   6750. 34 440E   7500. 38 340E   8250. 41 220E   9000. 45  90E    *


PL                *
```

Figure 7. - This plot illustrates using options 2 and 4 to remove all but the first, last, and left-most grid-lines, by supplying the frequencies = 0 in P(3) and P(4). This effect is possible in PLOTXY also. This plot also illustrates the use of option 64 with PLOTMY. The x ordinate only is printed, not both, as in PLOTXY (see fig. 4).

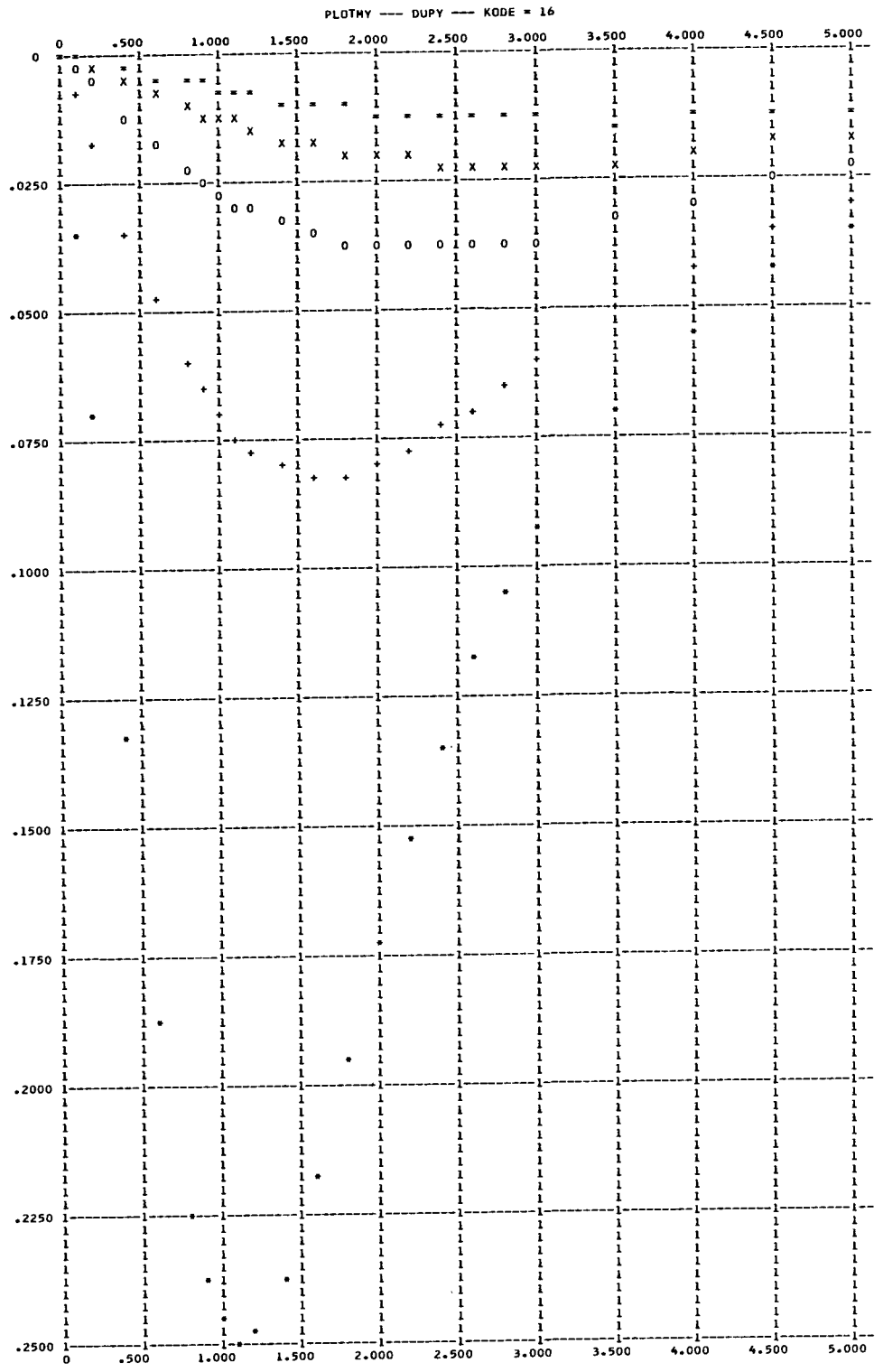
FIGURE 8.


FORCE FIELD OF ELECTROMAGNET GK792A
ZETA = 5.0
G/A = 10.0
0
.0250
.0500
.0750
.1000
.1250
.1500
.1750
.2000
.2250
.2500
1.000
2.000
3.000
4.000
5.000
S/B VS. FORCE PARAMETER FOR CONSTANT VALUES OF T/ZZ
* - 1.0 + - 0.8 0 - 0.6
X - 0.4 = - 0.2

```
PT   FIGURE   8.  *
PT                                                                                                    ZETA =  5.0
PT                                          FORCE FIELD OF ELECTROMAGNET  GK792A                      G/A  = 10.0


PI-32 0.000000-0.000000 0.000100 0.000025 10 10  0    *


PU  0  0*  0+  00  0X  0=E  1 14*  3+  10  1X  0=E  2 28*  7+  20  1X  1=E  4 53* 14+  50  2X  1=E  6 75* 19+  70  3X  2=E    *
PU  8 90* 24+  90  4X  2=E  9 95* 26+ 100  5X  2=E 10 98* 28+ 110  5X  3=E 11100* 30+ 120  5X  3=E 12 99* 31+ 120  6X  3=E    *
PU 14 95* 32+ 130  7X  4=E 16 87* 33+ 140  7X  4=E 18 78* 33+ 150  8X  4=E 20 69* 32+ 150  8X  5=E 22 61* 31+ 150  8X  5=E    *
PU 24 54* 29+ 150  9X  5=E 26 47* 28+ 150  9X  5=E 28 42* 26+ 150  9X  5=E 30 37* 24+ 150  9X  5=E 35 28* 20+ 130  9X  6=E    *
PU 40 22* 17+ 120  8X  5=E 45 17* 14+ 100  7X  5=E 50 14* 12+  90  7X  5=E    *


PL                                                    S/B VS. FORCE PARAMETER FOR CONSTANT VALUES OF T/2Z*
PL                                  * - 1.0   + - 0.8   0 - 0.6      *
PL                                  X - 0.4   = - 0.2    *
```

Figure 8. - This plot illustrates using PLOTMY - Variation I - DUP X with KKK(1) = KODE = 0. A single set of x values was placed in the x array, and five sets of y values one after another in the y array. The number of curves was placed in KKK(2) and the number of x values in KKK(3). Variation I was specified by placing 1. in P(1).

FIGURE 9.

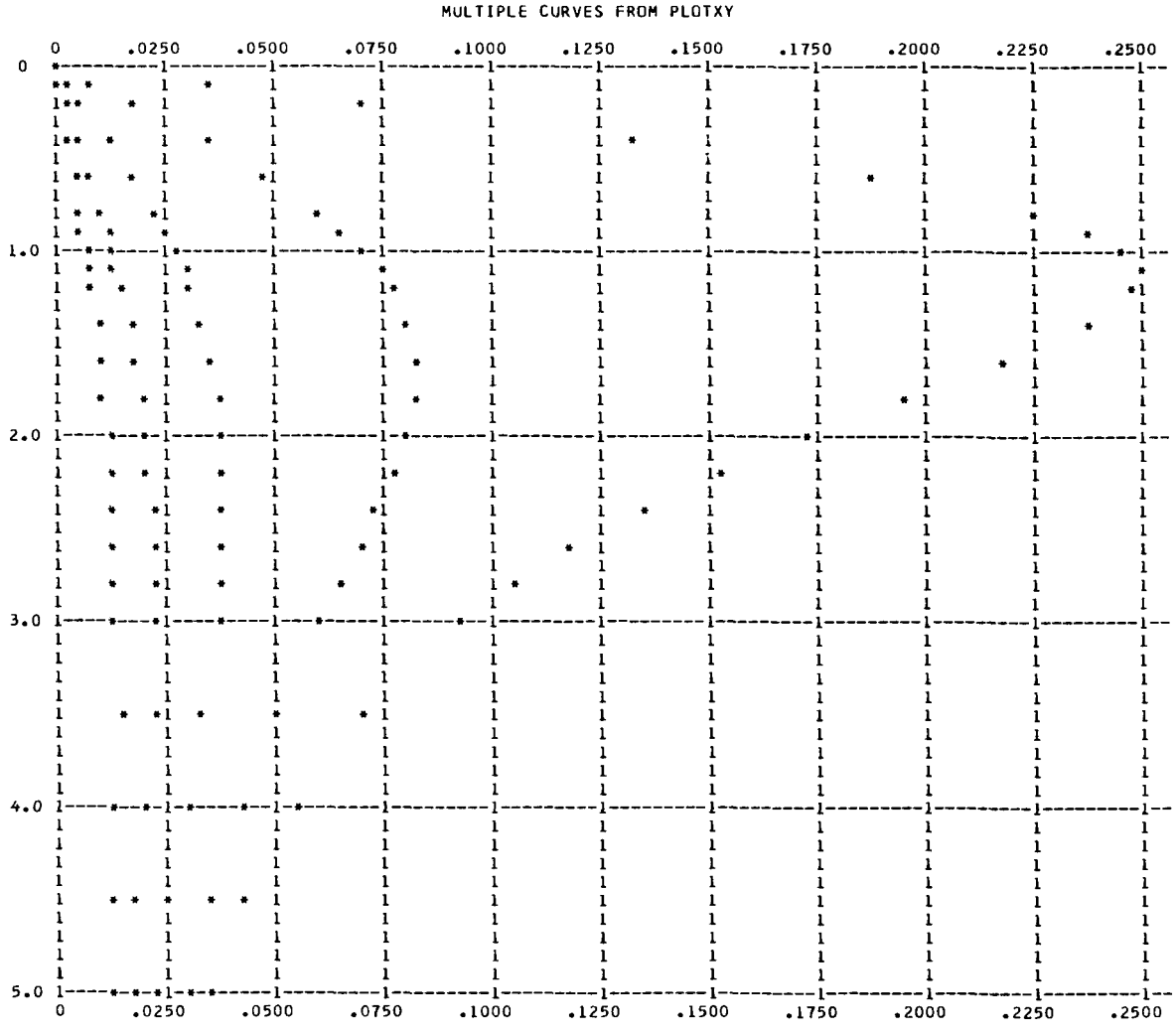


(SAME DATA AS IN FIGURE 8.)

```
PT   FIGURE   9.  *
PT                                                    PLOTMY --- DUPY --- KODE = 16  *

P1-23 0.000000 0.000000 0.000025 0.000050 10 10  0    *

PD  0  0*E  0  0+E  0  00E  0  0XE  0  0=E  0  2=E  1  2XE  1  4=E  1  4XE  1  8=E  1  20E  2 12=E  2  8XE  2 16=E  2  40E  2 18=E
PD  3 20=E  3 22=E  3 24=E  3 12XE  3  2+E  4 28=E  4 32=E  4 16XE  4 36=E  5 40=E  5 18XE  5  80E  5 44=E  5100=E  5 20XE  5 48=E
PD  5 52=E  5 90=E  5 56=E  5 60=E  5 22XE  5 80=E  6 70=E  6 24XE  7 28XE  7100XE  7  4+E  7 120E  7 32XE  7 90XE  8 36XE  8 80XE
PD  8 40XE  8 44XE  9 70XE  9 48XE  9 52XE  9 60XE  9 56XE  9 160E  91000E 10 180E 10 900E 11 200E 12 220E 12100+E 12 800E 12 240E
PD 13 700E 13 280E 14  8+E 14  2*E 14100*E 14 90+E 14 320E 15 600E 15 560E 15 360E 15 520E 15 400E 15 480E 15 440E 17 80+E 17 90*E
PD 19 12+E 20 70+E 22 80*E 24 60+E 24 16+E 26 56+E 26 18+E 28  4*E 28 52+E 28 20+E 28 70*E 29 48+E 30 22+E 31 44+E 31 24+E 32 40+E
PD 32 28+E 33 36+E 33 32+E 37 60*E 42 56*E 47 52*E 53  8*E 54 48*E 61 44*E 69 40*E 75 12*E 78 36*E 87 32*E 90 16*E 95 28*E 95 18*E
PD 98 20*E 99 24*E100 22*E    *

PL                                                    (SAME DATA AS IN FIGURE 8.)      *
```

Figure 9. - This plot illustrates using PLOTMY - Variation II - DUP Y with KKK(1) = KODE = 16. The x and y arrays used for figure 8 were interchanged and the x scale specified by placing KSX = 2. in P(6) and DX = 25. in P(8). Otherwise, PLOTMY's choice of scale (which is based on the length of the x array, not on the number of points in the y array supplied in KKK(3)) would have spread the plot over four pages. Variation II was specified by placing 3. in P(1). Compare with figure 8.

FIGURE 10.

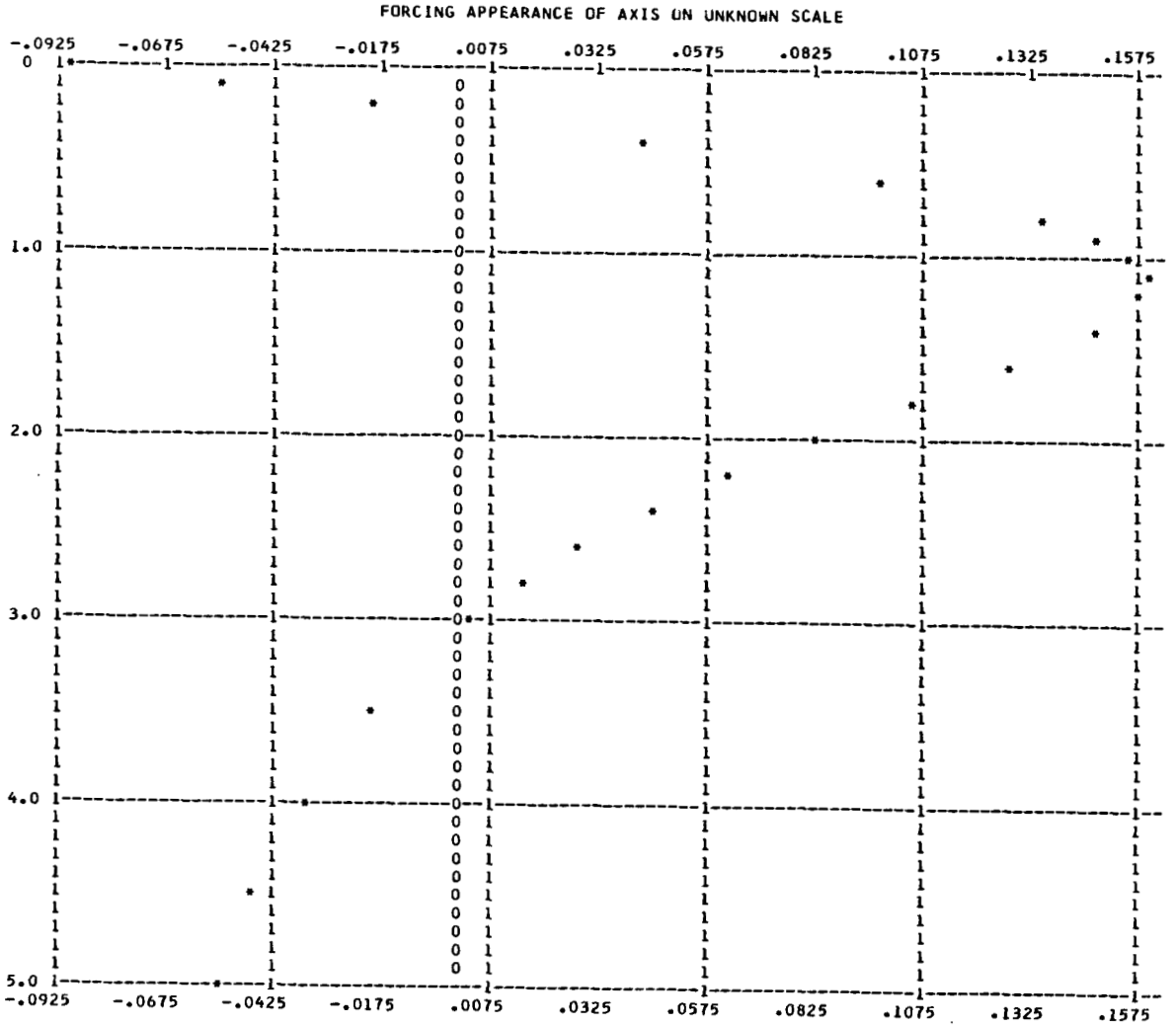


(SAME DATA AS IN FIGURE 8.)

```
PT   FIGURE  10.  *
PT                                                    MULTIPLE CURVES FROM PLOTXY        *

P1*52 0.000000-0.000000 0.000001 0.000025 10 10  0    *

PD  0  0  0  0  0  0  0  0  0  0  1 14  1  3  1  1  1  1  1  0  2 28  2  7  2  2  2  1  2  1  4 53  4 14  4  5  4  2  4  1  6 19
PD  6  7  6  3  6 75  6  2  8 24  8  9  8 90  8  4  8  2  9 26  9 95  9 10  9  5  9  2 10 28 10 11 10  5 10  3 10 98 11 12 11  5
PD 11 30 11  3 11100 12 12 12 31 12  6 12 99 12  3 14 13 14 95 14  7 14  4 14 32 16 14 16  7 16  4 16 87 16 33 18  8 18 15 18 78
PD 18 33 18  4 20  8 20 32 20  5 20 15 20 69 22  8 22 31 22 61 22  5 22 15 24  9 24  5 24 54 24 15 24 29 26 15 26  5 26  9 26 28
PD 26 47 28 26 28  5 28 42 28  9 28 15 30  5 30  9 30 24 30 15 30 37 35  6 35  9 35 20 35 28 35 13 40 22 40 12 40 17 40  8 40  5
PD 45 10 45  5 45 17 45  7 45 14 50 12 50 14 50  9 50  7 50  5    *

PL                                                    (SAME DATA AS IN FIGURE 8.)        *
```

Figure 10. - This plot illustrates the use of PLOTXY for multiple curves. The data used for figure 8 were supplemented by repeating the set of x values in the x array, then SORTXY was called. Option 16 was used and the x scale specified by placing KSX = 5. in P(6) and DX = 1. in P(8) to avoid spreading the plot over four pages. (PLOTXY would have chosen a scale based on the number of points (115.) supplied in P(1).) Compare with figure 8.

FIGURE 11.


FORCING APPEARANCE OF AXIS ON UNKNOWN SCALE
-.0925
-.0675
-.0425
-.0175
.0075
.0325
.0575
.0825
.1075
.1325
.1575
0
1.0
2.0
3.0
4.0
5.0

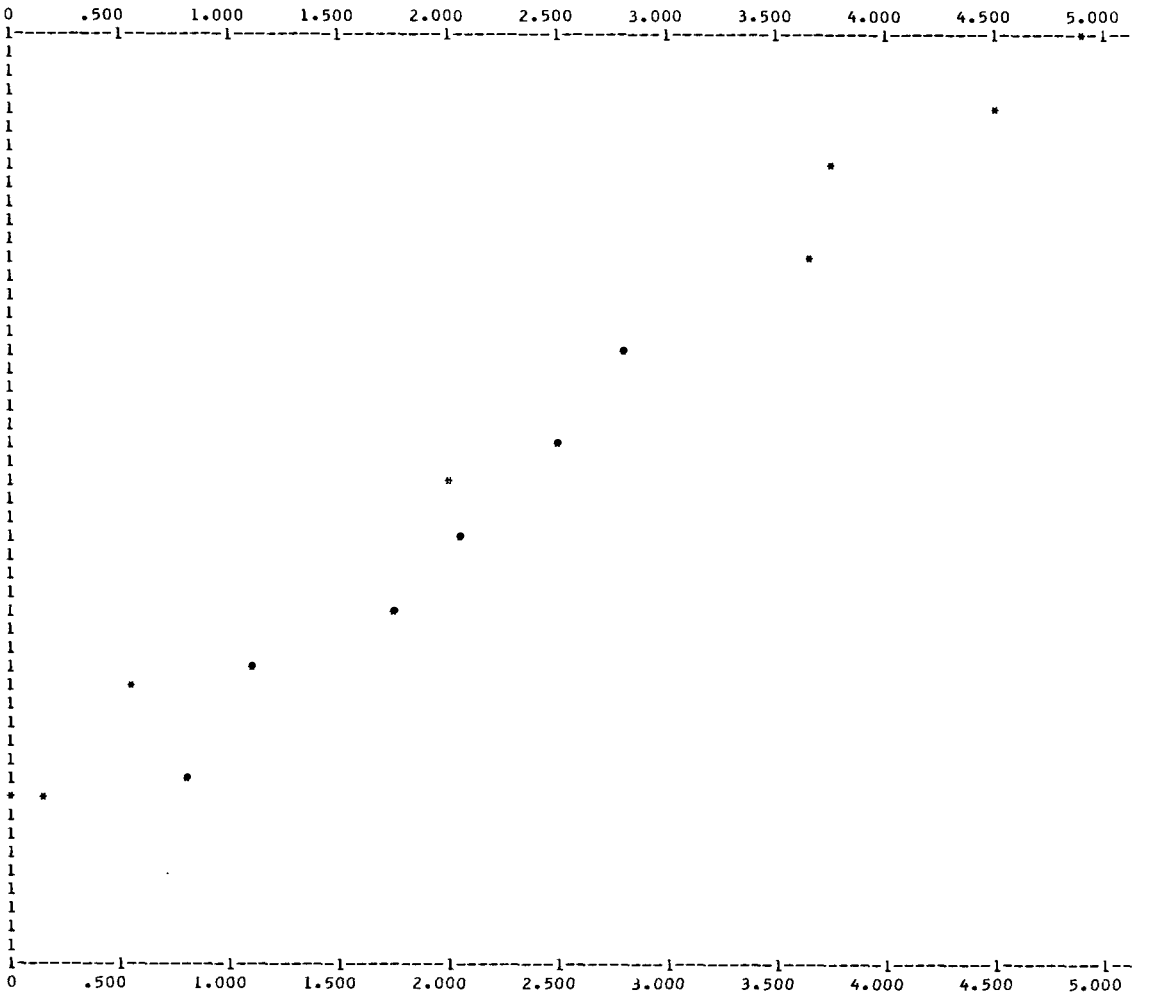
```
PT   FIGURE  11.  *
PT                                                  FORCING APPEARANCE OF AXIS ON UNKNOWN SCALE    *

PI-52 0.000000-0.000925 0.000001 0.000025 10 20  0    *

PU  0  1*E  1 15*E  1 370E  2 29*E  2 370E  3 370E  4 54*E  4 370E  5 370E  6 76*E  6 370E  7 370E  8 91*E  8 370E  9 96*E  9 370E
PU 10 99*E 10 370E 11101*E 11 370E 12100*E 12 370E 13 370E 14 96*E 14 370E 15 370E 16 88*E 16 370E 17 370E 18 79*E 18 370E 19 370E
PU 20 70*E 20 370E 21 370E 22 62*E 22 370E 23 370E 24 55*E 24 370E 25 370E 26 48*E 26 370E 27 370E 28 43*E 28 370E 29 370E 30 38*E
PU 30 370E 31 370E 32 370E 33 370E 34 370E 35 29*E 35 370E 36 370E 37 370E 38 370E 39 370E 40 23*E 40 370E 41 370E 42 370E 43 370E
PU 44 370E 45 18*E 45 370E 46 370E 47 370E 48 370E 49 370E 50 15*E*

PL                *
```

Figure 11. - When either PLOTXY or PLOTMY chooses a scale, the value zero does not necessarily fall on a grid-line. If it is desirable to see intersections at a glance, (in this case, with the x-axis), this is readily done. For this plot, PLOTMY - Variation III - NO DUP was used, and a faked curve generated whose x values were .1(.1)4.9, (the range of x was known) and whose y values were zero. Option 1 was used in order to choose a zero for the plotting character for the second curve; option 16 was used to confine the curve to one page; option 4 was used with P(4) = 20. to lighten the appearance of the grid.

FIGURE 12.

TESTING HYPOTHESIS THAT THE RELATIONSHIP BETWEEN THE GIVEN SETS OF VALUES FOR FV AND V IS OF THE FORM FV = E**(A*V). LOG (BASE 10) OF FV IS PLOTTED DOWN THE PAGE. V IS PLOTTED ACROSS THE PAGE.

FV	LOG FV	V
	X*E 3	Y*E 3
0.2	-700.	4900.-
0.5	-301.	4500.
1.0	1.	3750.
3.0	477.	3650.
10.0	999.	2800.
30.2	1480.	2500.
50.0	1699.	2000.
99.8	1999.	2050.
248.0	2394.	1750.
495.0	2695.	1100.
698.0	2844.	550.
2002.0	3301.	800.
2687.0	3429.	0.



```
PT   FIGURE  12.  *
PT                                TESTING HYPOTHESIS THAT THE RELATIONSHIP BETWEEN THE GIVEN SETS OF VALUES FOR FV AND V IS OF THE
PT                                FORM FV = E**(A*V). LOG (BASE 10) OF FV IS PLOTTED DOWN THE PAGE. V IS PLOTTED ACROSS THE PAGE.
PT                  *
PT   FV  LOG FV   V     *
PT                  *
PT                  *
PT        X*E 3  Y*E 3  *

PI*33-0.000700 0.000000 0.000100 0.000050  0   0 19     *

PD   0.2 -700.  4900.  0 98    0.5 -301.  4500.  4 90    1.0     1.  3750.  7 75    3.0  477.  3650. 12 73   10.0   999.  2800. 17 56
PD  30.2 1480.  2500. 22 50   50.0 1699.  2000. 24 40   99.8 1999.  2050. 27 41 248.0 2394.  1750. 31 35 495.0 2695.  1100. 34 22
PD 698.0 2844.   550. 35 112002.0 3301.   800. 40 162503.0 3398.   150. 41  32687.0 3429.     0. 41  0*

PL                  *
```

Figure 12. - This plot simulates a plot made on semilog paper. The values in the y array were those that would be plotted on the linear scale, while those in the x array were the logarithms of the values that would be plotted on the logarithmic scale. The original values are displayed using option 8.

APPENDIX B

NOTES

I. Machine Configuration

A. The FORTRAN subroutines have no hardware requirements other than those required for FORTRAN II (see IV.); however, they do rely on the Lewis Monitor Error Package (see III.-B.), which makes use of divide-by-zero-trap hardware.

B. The SPS subroutine PLOT was written for a 4K-1401, modified[1] to alter address-validity and instruction-length checking, and with the following "Special Features" (ref. 2):

Print Storage

Additional Print Control

Sign Control Left

Decimal Control

Indexing

Store Address Register

Move Record

High-Low-Equal Compare

Column Binary

II. Storage Requirements

PLOTXY - 875 words

PLOTMY - 1125 words

PISTUG - 375 words

SORTXY - 97 words

SCALE - 118 words

PLOT - 1924 characters

[1] IBM-RPQ F-93971.

III. Operating and Programming Conventions

A. The Lewis Monitor System requires all output to be written on tape 6. WRITE OUTPUT TAPE 6 statement numbers are:

PLOTXY - 238, 251, 410, 420, 430, 440

PLOTMY - 306, 318, 432, 462

PISTUG - 900

B. The Lewis Monitor System includes the Lewis Monitor Error Package, which is responsive to overflows, major and minor underflows, division by zero, and the misuse of library subroutines, such as, square root of negatives, logarithm of zero (see XII., Error Write-Outs).

C. The Lewis 1401 tape-to-printer program expects input records on tape 3. In subroutine PLOT, instruction *NXREC* reads tape 3, and *PNEX* backspaces tape 3.

D. The Lewis 1401 tape-to-printer program loads the word marks for the three index registers; hence subroutine PLOT does not.

E. The Lewis 1401 tape-to-printer program tests a switch, *PPGUP*, set by subroutine PLOT at *ENTRY*+7, to control skipping to a new page.

F. Subroutine PLOT moves records with a Move Record command; hence all TITLE, INFO, and LEGEND records must be terminated with a record mark or a group mark word mark.

G. If the printing of a plot is halted by machine error or operator intervention, the subroutine PLOT must be reloaded. Reloading will supply the necessary initialization that occurs when a plot ends normally.

IV. FORTRAN II

All FORTRAN subroutines were written in 32K 709/7090 FORTRAN (ref. 3).

A. There are Boolean statements in:

PLOTXY - 140, 150, 160, 170, 180, 190, 200

PLOTMY - 103, 104, 105, 106, 120, 150, 154, 160, 164, 170, 180,

230, 240, 322

B. The standard FORTRAN library subroutines for LOG and EXP are used by PLOTXY, PLOTMY, and PISTUG.

C. The standard FORTRAN library subroutine IOH, modified to write 132 character records, is used by PLOTXY and PLOTMY.

D. The standard FORTRAN library subroutine IOH will not generate a record less than three words (18 BCD characters) long, and it will supply blanks if necessary to attain that length. When the last PD record written contains only one point, and the left-side field length is zero, the end-of-record test made by PLOT fails, and the blanks are treated as the line count (zero) for another point. Although all points will have been plotted correctly, the plot will terminate with an X OUT OF ORDER error message and will not be properly ended. This situation can occur in PLOTXY only when conditions (1), (4), and (5) are simultaneously true, and in PLOTMY only when (2), (3), (4), and (5) are simultaneously true:

(1) The number of points to be plotted is one more than an integral multiple of 21.

(2) The number of points to be plotted is one more than an integral multiple of 16.

(3) Either Variation II (DUPY) or III (NO DUP) is selected.

(4) Option 8 is not being used.

(5) Option 64 is not being used.

These combinations of conditions are relatively rare; each has occurred only once in over a year of use.

E. The simultaneous use of a symbol as a fixed-point variable and as the index of a loop within the list of an output statement is accepted and compiled correctly by both 704 FORTRAN and 32K 709/7090 FORTRAN. The simplest change for other versions of FORTRAN is the substitution of JJ for each appearance of J in the output statements 410, 420, 430, and 440, in PLOTXY.

V. Perforation Test (PISTUG)

When an x array is being scaled by PISTUG, the maximum number of lines permitted for the plot is computed by dividing the total number of points by 35 (using integer arithmetic), adding 1, and multiplying the result by 55. (The constants 35 and 55 were determined empirically.) After the scaling parameter for the scale-factor has been computed, a check is made to determine whether, on a two-page plot, the final x grid-line will fall within 12 lines of the top of the second page. If it will, an arbitrary 35 is added to the maximum number of lines permitted, and the parameter is recomputed.

VI. Range Test (PISTUG)

The absolute value of the range (maximum element minus minimum element) of any array being scaled by PISTUG must be greater than 10^{-3}, and the larger of the absolute values of these two elements must fall between 10^{-3} and 10^{7}. This restriction is dictated by the length of the grid-label field on the 1401 and is enforced by testing the size of KCDX in PISTUG (see

XII.-D.). If scaling is done by the programmer, no check is made to enforce this requirement, and the programmer is responsible for avoiding a choice of scale that will lead to impossible grid-labels.

VII. Range Assumptions (PLOT)

A. PLOT assumes the line count, KX, is limited by $0 \leq KX \leq 999$, and if PISTUG does the scaling, it is. If the programmer uses option 16 and chooses scaling parameters that lead to negative line counts, program may be overwritten, with unpredictable results. Values of KX greater than 999 also cause trouble, but any programmer with 15 pages of plot has trouble already.

B. PLOT assumes the print position count, KY, is limited by $0 \leq KY \leq 102$, and if PISTUG is doing the scaling, it is. PLOT tests KY, and any value of KY such that $103 \leq KY \leq 999$ will cause a record mark to be printed in position 103. If KY is negative and a single digit, the point will be printed as if it had been positive and the programmer will never know it is wrong; larger negative values may overwrite program inviting catastrophe.

VIII. Range Assumptions (PISTUG)

Theoretically PISTUG can scale an array of any length correctly, but if scaling an X array of more than 595 points, scaling parameters leading to line-counts greater than 999 may be chosen. This limitation is on the total number of points, and would not apply, for instance, to using PLOTMY - Variation I - DUP X for six curves of 200 points each.

IX. Direction Assumptions (PLOTMY)

PLOTMY assumes that the scale-factor in the x-direction is positive, and if PISTUG is doing the scaling, it will be. If the programmer uses option 16 and supplies a negative scaling parameter, DX, the plot will be incorrect. If the programmer really wants a decreasing x scale he should multiply the X array by -1. and manually insert negative signs and alter "-" to "+" on the x grid-labels as required.

X. Duplicate Points

Even though PLOTMY may supply two or more plotting characters to be printed in the same location, PLOT will print only the last character received. No over-printing is done. Similarly, of multiple left-side fields, only the last received by PLOT will be printed.

XI. Multi-record Titles and Legends

The most common programming error to date is the omission of the required 2HPT or 2HPL for the second TITLE or LEGEND record from the FORMAT statements. Such omission also occurs when slashes (///) are used in the

FORMAT statement to get line spaces following, or between, printed lines. This does not work. Use 2HPT/2HPT/2HPT, etc.

XII. Error Write-Outs

A. PLOTXY and PLOTMY have error returns. If the value of either of the scaling parameters DX or DY is returned from PISTUG as zero, the routines branch immediately to RETURN and no INFO or DATA records are written. However, this situation has yet to occur without some message having already been written during the execution of PISTUG.

B. PLOTMY will write an error message

NPTS IS MISSING OR WRONG IN THE KARRAY. IMIN = nnn

if the search loop (statement number 424 to 430), which finds the correct plotting character, is unsuccessful. Information missing or misplaced in the K array, or "dimensioning" the K array less than 14, has caused this write-out to occur.

C. The programming error of specifying DX (or DY) = 0 is caught by the Lewis Monitor Error Package, either as division by zero in statement number 324 (or 334) in PLOTXY, and in 394 (or 410) in PLOTMY; or, if FX (or FY) is zero (this is permissible) also, DX (or DY) = 0 is caught earlier as LOGF(0.0) at 187+4 (or 199+4) in PLOTXY, or at 238 in PLOTMY.

D. PISTUG writes an error message if an array fails the grid-label-size test made on KCDX,

PLOT HALTED BY ARRAY NUMBER n VALUES OUT OF RANGE. KCD=mmmm

and branches to RETURN with DX (or DY) = 0, which prevents further calculation in PLOTXY or PLOTMY.

E. The programming error of supplying an array (to be scaled by PISTUG) whose elements are all equal (usually zero, from using a wrong array name) is caught by the Lewis Monitor Error Package as LOGF(0.0) at statement number 122 in PISTUG. Sending valid data and forgetting to supply the number of points has exactly the same effect, as does supplying the number of points in fixed point rather than in floating point. If the programmer chooses both his scales and neglects to supply the number of points, there will be no error write-outs, but his plot will consist entirely of one horizontal grid-line correctly labeled.

F. PLOT writes the error message

X OUT OF ORDER mmmnnn

whenever the value of a difference it forms (by subtracting the current total in the line counter from the current value of X) is negative. This

will always occur if PLOTXY has been called with the elements of the X array not in sequence. Also written out are this difference (mmmm) and the corresponding value of Y(nnn). PLOT then bypasses the remaining PD and PL records without using them, and control is returned to the tape-to-printer routine.

REFERENCES

1. Dellner, Lois T., and Moore, Betty Jo: An Optimized Printer Plotting System Consisting of Complementary 7090 (FORTRAN) and 1401 (SPS) Subroutines. I - Instructions for Users. NASA TN D-2174, 1964.

2. Anon.: IBM Reference Manual - 1401 Data Processing System. A24-1403-3, Int. Business Machines Corp., June 1961.

3. Anon.: IBM Reference Manual - 709/7090 FORTRAN Programming System. C28-6054-2, Int. Business Machines Corp., Jan. 1961.