SHARE PROGRAM LIBRARY AGENCY

PROGRAM NUMBER

086002

University of Miami

1365 MEMORIAL DRIVE - CORAL GABLES, FLORIDA
(305) - 284-6257

360D-08.6.002

INTERFACE BETWEEN PL/I USER PROGRAMS

AND CALCOMP ROUTINES

Direct Inquiries To:

Eldon H. Remy
Management System Development Corp.
Eastman Kodak Company
Rochester, New York 14650

ABSTRACT

The OS/360 plotting routines provided by Calcomp are written in Fortran and Assembler and are designed to be used by a Fortran program.

These routines may be called by a PL/I program but such use doesn't permit some commonly used PL/I features such as character strings. It is also an inconvernience for a PL/I programmer to adhere to Fortran linkage conventions such as avoiding passing dope vectors.

To permit the PL/I programmer to use straight-forward PL/I statements in plot programs this interface has been written to intercept the linkage between the users program and the routines provided by Calcomp.

Linkage editor 'change' cards are used to permit the interface to have entry points with the same names as the routines provided by Calcomp.

The JCL may be modified to conform to an installations procedures and naming conventions.

It is suggested that an installation provide the facility for the declare statements to be entered by an include compile-time statement.

Deck Key

150 cards Basic

Using Calcomp Plotter Routines With PL/I

Introduction

The PL/I Plot Interface Program can be useful to users of the F level PL/I compiler who have a CALCOMP plotter and the S/360 subroutines provided by California Computer Products.

The plotting device is off-line to the IBM 360 Computer. The plotting routines create a file on a reel of 9 track magnetic tape. At the completion of a job this tape is removed and mounted on a control unit attached to the plotter to create the actual drawing. The programmer need have no knowledge of tape processing but must include a single standardized control card with his input deck to make the tape available to the computer operating system. The card must be punched as follows starting in column 1:

```
//GO.PLOTTAPE DD UNIT=(2400,,DEFER),LABEL=(,NL),DISP=(,KEEP)
```

The Calcomp 563 is an incremental plotter which is capable of moving a pen in steps of .01 inches in the +X or +Y directions and the 45° combinations of X and Y at a speed of 200 steps per second. The plots are drawn on chart paper which is available in rolls 31 inches wide and 120 feet long. The pen can travel 29.5 inches in the +Y direction. The standard paper available in the Computing Center is either plain white or printed over a 29 inch width with 20 divisions per inch in the X and Y directions.

All information pertaining to pen location and size of the plot must be in terms of inches.

The Five Basic Routines

There are five plotting subroutines provided by California Computer Products called PLOT, LINE, SCALE, SYMBOL and AXIS. They are written in Assembler and Fortran. Because these basic routines do not make use of the PL/I capabilities such as character string data and require the Fortran library, an interface for PL/I users is available. This interface is a program which includes the basic plotter routines and the needed Fortran routines and permits PL/I users to plot using straight forward PL/I statements. The PL/I programmer merely uses various entry points of the interface to invoke the basic routines. The interface also performs for the PL/I programmer

three housekeeping functions which have to be done by the Fortran programmer on his own:

1. Calls PLOTS (an entry point to PLOT) to set up a buffer area for output to the tape.
2. Performs a CALL PLOT (0, -32, -3) to set a point of origin for future pen position references.
3. Issues a message at the start of the job that a plot program is starting. It also prints a message at the end of the program indicating the 'normal end' has been reached and that the tape should be saved.

Using the Interface

To plot, the program calls the interface and enters at appropriate entry points. The DCL statements shown with each entry point need not be included in your program if the arguments passed agree exactly with the attributes on the DCL statement. If you specify constants in the CALL statements the attributes probably do not agree (remember a fixed binary number is written with 0's and 1's followed by a B). If the DCL statements are included automatic conversion will be performed. The following entry points correspond in function to the identically named basic routines used by Fortran programmers.

(1) PLOT (X,Y,IPEN);
DCL PLOT ENTRY (FLOAT,FLOAT,FIXED BIN);
The pen is moved from its present location to point X,Y which has been defined by the users program.

IPEN = 2 is the pen down code for drawing a line.
IPEN = 3 is the up code for moving the pen without drawing a line.

A negative value for IPEN causes the point X, Y to be a new reference origin for any subsequent plotted data, i.e., X, Y now becomes 0, 0.

Note: The last plot statement in a users program must be CALL PLOT (X,Y,999); (include DCL for PLOT or write 999 in binary). This causes an end of plot record on the output tape. If IPEN = 999, the end of job typewriter message is given reminding the operator to save the tape.

(2) CALL FACTOR (FAC);
DCL FACTOR ENTRY (FLOAT);

This entry causes a multiplicative factor (FAC) to be applied to all subsequent plotted data. It has the effect of

enlarging or reducing the size of a plot without the user having to change any other values. The factor applies to everything plotted, axes, alphabetic data, etc. Normally FAC = 1 unless specifically changed by use of this entry point.

(3) CALL WHERE (X,Y,FAC);
DCL WHERE ENTRY (FLOAT,FLOAT,FLOAT);

X,Y,FAC are all values returned from WHERE when this entry point is used. X,Y are the current pen locations and FAC is the current multiplying factor. The use of this entry appears to be unnecessary for most applications.

(4) CALL SCALE (X,S,N,K,XMIN,DX);
DCL SCALE ENTRY ((*)FLOAT,FLOAT,FIXED BIN, FIXED BIN,FLOAT,FLOAT);

This entry is used to scale an array of values over some number of inches specified by the user. X is the name of the single dimensioned array which is to be scaled. Its values must already be defined by the users program.

S is the range in inches over which the user desires the array to be scaled. This value should be consistent with the length of the axis to be used for plotting the array.

N is the number of points from the array to be scaled. N may be less than or equal to the dimension of the array.

K is the skip factor for the values to be scaled. Every Kth value will be scaled. Normally K=1. Be sure that the value N*K is less than or equal to the dimension of the array.

XMIN is the scaled minimum value of the array and is an argument returned by SCALE. It is normally used to label the first tickmark of an axis.

DX is the scaled increment for the array and is an argument returned by SCALE. It is normally used as an axis increment. DX is original array units per inch.

The SCALE subroutine is computational and does no actual plotting. It returns a minimum and increment which can be used for drawing a normal axis with reasonable values. The increment itself is always 1, 2, 4, 5 or 8×10^{1} depending on the range of the original values. The scaled values replace the original array values. The use of N and K as arguments provide all means of scaling mixed arrays, i.e., arrays whose values don't all belong to the same set.

(5) CALL AXIS(X,Y,LABEL,N,S,A,XMIN,DX);
DCL AXIS ENTRY (FLOAT,FLOAT,CHAR(*),FIXED BIN,FLOAT,FLOAT, FLOAT,FLOAT);

This entry produces a single completely defined axis with a tickmark every inch and a label obtained from a character string passed as an argument.

X,Y are the coordinates of the starting point of the axis.

LABEL is a character string containing the axis label.

N is used in Fortran to provide the length of the label but in PL/I the value of N is ignored except for the sign. If N is positive all labeling is done above the axis. If negative the label appears below the axis. The PL interface obtains the label length from the character string dope vector via LENGTH function.

S is the length of the axis in inches. If the SCALE entry was used for data scaling, S should be consistent with the range over which the data was scaled.

A is the angle in degrees at which the axis is to be drawn. 0° draws the axis in a +X direction on the plotter. 90° draws the axis in the +Y direction. Any angle $-360° \leq A \leq 360°$ is acceptable.

XMIN is the value to be printed at the first tickmark on the axis. It is the value returned by the SCALE routine if it was used to scale the array to be plotted.

DX is the increment for the tickmark labels, also returned by the SCALE routine if it was used.

If SCALE was not used, the XMIN and DX values must be established by the user's program.

(6) CALL LINE (X,Y,N,K,J,L);
DCL LINE ENTRY((*)FLOAT,(*)FLOAT,FIXED BIN,FIXED BIN, FIXED BIN,FIXED BIN);

The LINE entry may be used to plot a function represented by two single dimensioned arrays. Options are available for a continuous line plot, a point plot with a particular symbol at each point or at regular intervals, or both of these. The LINE entry assumes the arrays are properly scaled for the plotter (i.e., the values are in terms of inches on the plot paper).

X is a single dimensioned array of abcissa values properly scaled.

Y is a similar dimensioned array of ordinate values.

N is the number of points to be plotted. N≤dimension of X,Y.

K is a skip factor for the values to be plotted. Every Kth point is plotted. Normally K=1. Be sure that N*K is less than or equal to the dimension of the arrays. A negative K will put the pen down when moving to point X(1), Y(1).

J is a code indicating what type of line plot is desired. If J<0 a point plot only is created and only every J*Kth point in the arrays is indicated by a symbol. If J=0 a continuous line plot is made with no symbols plotted. If J>0 a continuous line plot is created with every J*Kth point indicated by a symbol.

L is a code indicating the type of symbol to be drawn by the LINE routine. See the attached list of codes for L.

(7) SYMBOL (X,Y,H,LABEL,A,N);
DCL SYMBOL ENTRY(FLOAT,FLOAT,FLOAT,CHAR(*),FLOAT,FIXED BIN);

SYMBOL is used to draw the characters contained in a string.

X Y are the coordinates of the lower left hand corner of the first character to be drawn.

H is the height of the characters in inches. Normally .05≤H≤1.0.

LABEL is character string to be drawn.

A is the angle in degrees at which LABEL is to be drawn -360°≤A≤360°.

N is used to specify the number of characters to be drawn. In PL/I the length of the character string is gotten by the LENGTH built in function. Consequently the value of N is ignored except the sign which normally should be plus. If N is minus one (-1), the contents of LABEL is used as a code indicating the type of symbol to be drawn at the X,Y location. See the attached sheet for list of codes (same as L in LINE). The code need not be enclosed in quotes even though its position in the list indicates string. Conversion occurs automatically.

Note: The Fortran system has a routine called NUMBER. There is no PL/I counterpart for NUMBER because the same function can be handled by SYMBOL with automatic conversion from numeric to character string.

Eldon H. Remy

EHRemy:sma
September 1, 1966

CHARACTER CODE TABLE.

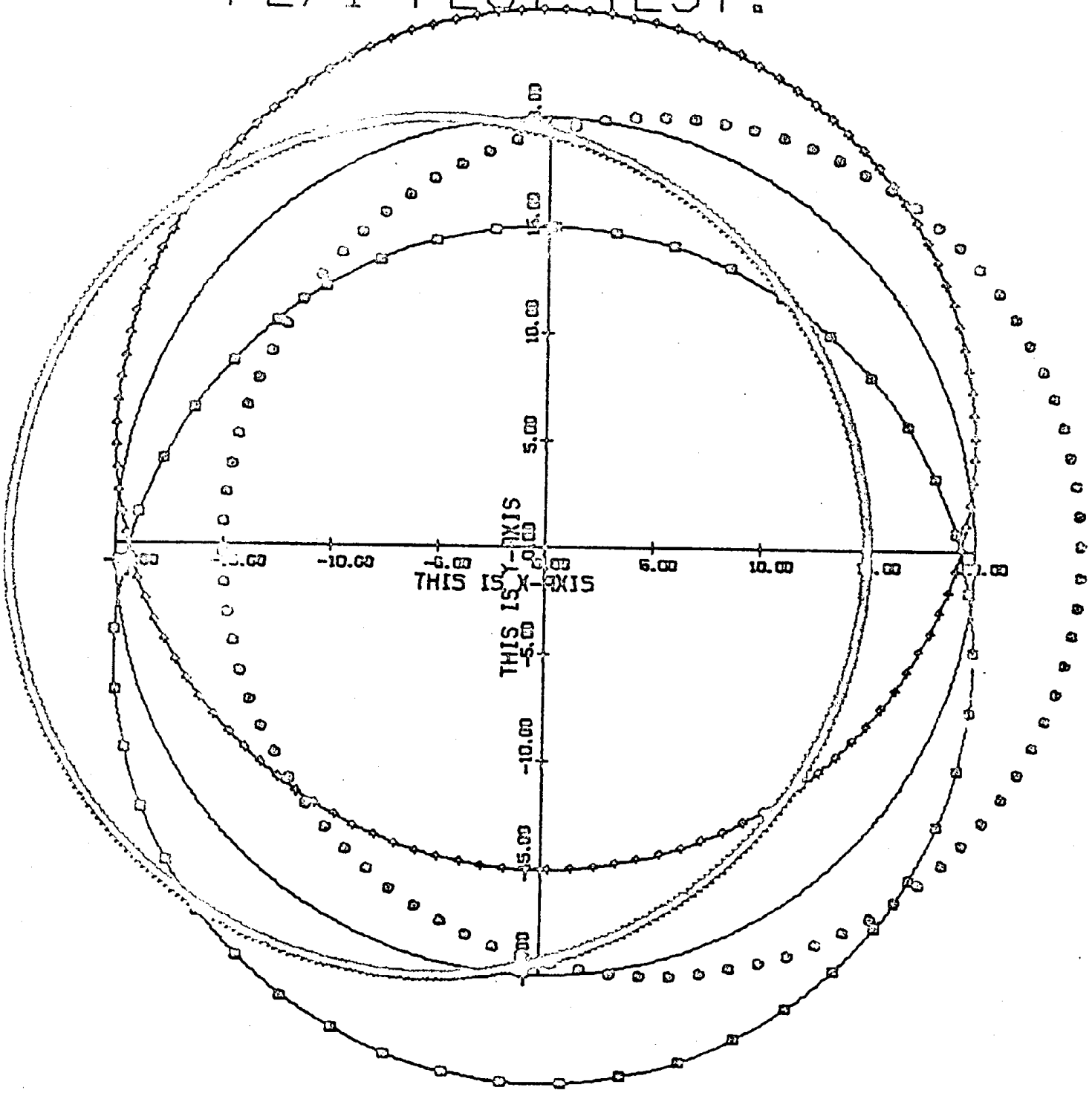
THE FOLLOWING ARE A LIST OF NUMERIC CODES WHICH IF USED AS A CONSTANT OR VARIABLE TO BE PLOTTED BY 'SYMBOL' OR 'LINE' ROUTINES YIELD THE INDICATED CHARACTER.

VALUE	SYMBOL	VALUE	SYMBOL	VALUE	SYMBOL
0	□	50	≤	100	U
1	○	51	≥	101	V
2	△	52	∆	102	W
3	+	53	[	103	X
4	×	54	]	104	Y
5	◇	55	/	105	Z
6	▽	56	↑	106	0
7	⋈	57	⌊	107	.
8	N	58	↑	108	%
9	Y	59	≠	109	-
10	⌘	60	←	110	>
11	⋈	61	×	111	?
12	⊠	62	↑	112	@
13	—	63	↓	113	1
14	☆	64		114	2
15	-	65	A	115	3
16	-	66	B	116	4
17		67	C	117	5
18	^	68	D	118	6
19	≡	69	E	119	7
20	←	70	F	120	8
21		71	G	121	9
22	⇌	72	H	122	:
23	±	73	I	123	#
24	-	74	⊕	124	@
25	▪	75		125	'
26	▪	76	<	126	=
27	⌐	77	(	127	"
28	∪	78	+		
29	∨	79	|		
30	∠	80	&		
31	∠	81	J		
32	)	82	K		
33	(	83	L		
34	⊏	84	M		
35	⊐	85	N		
36	⊕	86	O		
37	⊙	87	P		
38	⊹	88	Q		
39	×	89	R		
40	⊎	90	!		
41	⋋	91	$		
42	⊓	92	*		
43	⊕	93	)		
44	∈	94	;		
45	?	95	_		
46		96	`		
47		97	/		
48	Σ	98	S		
49	⊹	99	T		

```
PLAT:PROC OPTIONS(MAIN);


 1              PLAT:PROC OPTIONS(MAIN);
 2             ON ERROR SNAP CALL HELP;
 4           DCL  PLOT ENTRY (FLOAT,FLOAT,FIXED BIN);
 5           DCL  FACTOR ENTRY (FLOAT);
 6           DCL  SCALE ENTRY ( (*) FLOAT,FLOAT,FIXED BIN,FIXED BIN,FLOAT,FLOAT);
 7           DCL  AXIS ENTRY(FLOAT,FLOAT,CHAR(*),FIXED BIN,FLOAT,FLOAT,FLOAT,FLOAT);
 8           DCL  LINE ENTRY ( (*) FLOAT,(*) FLOAT,FIXED BIN,FIXED BIN,FIXED BIN,
                                                                  FIXED BIN);
 9           DCL  SYMBOL ENTRY (FLOAT,FLOAT,FLOAT,CHAR(*),FLOAT,FIXED BIN);
10           DCL  WHERE ENTRY (FLOAT,FLOAT,FLOAT);
11              THETA=0;  DCL X(360),Y(360);
13              DO I=1 TO 360;
14                    THETA=THETA+.017453293;
15                    X(I)= 20.*COS(THETA);
16                    Y(I)= 20.*SIN(THETA);
17                END;
18                 CALL  FACTOR(.75);
19              CALL  PLOT (10,15,-3);
20              CALL  SCALE (X,8,360,1,XMIN,XDELTA);
21              CALL SCALE (Y,8,360,1,YMIN,YDELTA);
22              CALL  AXIS (0,4,'THIS IS X-AXIS',-14,8,0,XMIN,XDELTA);
23              CALL  AXIS (4,0,'THIS IS Y-AXIS',14,8,90,YMIN,YDELTA);
24                 CALL  LINE(X,Y,360,1,0,1);  /* DRAW CONTINUOUS LINE*/
25                 X=X+1;
26                 CALL  LINE(X,Y,360,1,-4,1); /*DRAW POINT PLOT EVERY 4TH*/
27                 X=X-2;
28                 CALL  LINE(X,Y,360,1,-1,11);/*DRAW POINT PLOT EVERY POINT*/
29                 X=X+1;
30                 Y=Y+1;
31                 CALL  LINE(X,Y,360,1,3,6);/*LINE WITH POINT EVERY 3RD*/
32                 Y=Y-2;
33                 CALL  LINE(X,Y,180,2,4,0);/*LINE EVERY 2ND,POINT EVERY 8*/
34           CALL  SYMBOL (1,9,.4,'PL/1 PLOT TEST',0,14);
35           CALL WHERE(B,C,D);
36              CALL SYMBOL (B+.25,C,.4,75,0,-1);   /*PUTS IN A PERIOD */
37           CALL SYMBOL (1,-1.4,.2,'FACTOR='||D,0,13);
38           CALL PLOT (B+11,-31,999);
39         END PLAT;
```

PL/1 PLOT TEST.



FACTOR= 7.50000E-01

```
//CARA       JOB K2-5631-CC,'REMY',MSGLEVEL=1
//              EXEC PL1FCLN
//PL1           EXEC PGM=IEMAA                                                 00020000
//SYSLIB        DD    DSNAME=CTOS.PL1SOURC,DISP=OLD,                          X00040000
//                  DCB=(RECFM=F,BLKSIZE=80,LRECL=80)                          00060000
//SYSLIN        DD    DSNAME=LIN,DISP=OLD,DCB=BLKSIZE=400                      00080000
//SYSUT1        DD    DSNAME=CTOS.UT3,DISP=OLD                                 00100000
//SYSUT3        DD    DSNAME=CTOS.UT4,DISP=OLD                                 00120000
//SYSPRINT      DD    SYSOUT=A                                                 00140000
//SYSPUNCH      DD    UNIT=SYSCP                                               00160000
//SYSIN         DD    DDNAME=SOURCE                                            00180000
//PL1.SOURCE DD *
IEF236I ALLOC. FOR CARA      PL1
IEF237I SYSLIB    ON 191
IEF237I SYSLIN    ON 190
IEF237I SYSUT1    ON 191
IEF237I SYSUT3    ON 291
IEF237I SYSPUNCH ON 00D
IEF237I SYSIN     ON 00C
```



```
VERSION 2  REL 10                         OS/360 PL/I COMPILER (F)                              PAGE
                                                                                                DATE 67.124

THE COMPLETE LIST OF OPTIONS USED DURING THIS COMPILATION IS--   EBCDIC
                                                                   CHAR60
                                                                 NOMACRO
                                                                   SOURCE2
                                                                   COMP
                                                                   SOURCE
                                                                   ATR
                                                                   XREF
                                                                 NOEXTREF
                                                                 NOLIST
                                                                   LOAD
                                                                 NODECK
                                                                   FLAGW
                                                                   STMT
                                                                   SIZE=159744
                                                                   LINECNT=050
                                                                   OPT=00
                                                                   SORMGIN=(002,072)
```

```
1          PLPLOT:PROC;
2                DCL LABEL CHAR(80) VARYING;
3                DCL XXX###Y(50) STATIC;
4          DCL (X,Y,FAC,A(*),AMIN,ADELTA,ANGLE,ABS(*),ORD(*),HEIGHT,L) FLOAT;
5          DCL (CODE,N,K,M,MM) FIXED BINARY;
6                DCL CONT LABEL (L1,L2);
7                DCL XXX###X(0:2047) STATIC;
8                DCL LL FIXED BINARY INIT(9192) STATIC;
9                DCL SWITCH FIXED BINARY INIT(1) STATIC;
10               DCL (I1,I2,I3) FIXED BIN STATIC;
11               DCL I STATIC FIXED BIN (31,0);
12             FIRSTM:PROC;
13                   CALL FPLOTS(XXX###X(0),LL);
14                    SWITCH=0;
15                   CALL FPLOT (0E+0,-33E+0,-118);
16            DISPLAY ('PL/1 PLOT IS STARTING***');
17                    END FIRSTM;
18           PLOTS:ENTRY (A,N);
19                 PUT LIST('CALL PLOTS NOT USED IN PL/1 PLOT INTERFACE');
20                 RETURN;
21           PLOT:ENTRY (X,Y,CODE);
22                 IF SWITCH=1 THEN CALL FIRSTM;
23                CALL FPLOT (X,Y,CODE);
24                 IF CODE=999 THEN DISPLAY('NORMAL END,PLEASE SAVE TAPE***');
25                 RETURN;
26           FACTOR:ENTRY (FAC);
27                 IF SWITCH=1 THEN CALL FIRSTM;
28                CALL FFACTOR(FAC);
29                 RETURN;
30           WHERE:ENTRY (X,Y,FAC);
31                 IF SWITCH=1 THEN CALL FIRSTM;
32                CALL FWHERE (X,Y,FAC);
33                 RETURN;
34           SCALE:ENTRY (A,L,N,K,AMIN,ADELTA);
35                 IF SWITCH=1 THEN CALL FIRSTM;
36                 I=LBOUND(A,1);
37                CALL FSCALE (A(I),L,N,K,AMIN,ADELTA);
38                 RETURN;
39           AXIS:ENTRY (X,Y,LABEL,CODE,L,ANGLE,AMIN,ADELTA);
40                 IF SWITCH=1 THEN CALL FIRSTM;
41                 CONT=L1;
42             L3: I2=CEIL(LENGTH(LABEL)/4);
43                 I3=-3;
44           UNSPCLP:DO I1=1 TO I2;
45                 I3=I3+4;
46                 UNSPEC(XXX###Y(I1))=UNSPEC(SUBSTR(LABEL,I3,4));
47                 END UNSPCLP;
```



```
48                 GO TO CONT;
49         DCL FAXIS ENTRY(,,,FIXED BIN,,,,);
50           L1:CALL FAXIS (X,Y,XXX###Y(1),(LENGTH(LABEL)*SIGN(CODE)),L,
51                          ANGLE,AMIN,ADELTA);
52                 RETURN;
53           LINE:ENTRY (ABS,ORD,N,K,M,MM);
54                 IF SWITCH=1 THEN CALL FIRSTM;
55                 I=LBOUND(ABS,1);
56                CALL FLINE (ABS(I),ORD(I),N,K,M,MM);
57                 RETURN;
58           SYMBOL:ENTRY (X,Y,HEIGHT,LABEL,ANGLE,CODE);
59                 IF SWITCH=1 THEN CALL FIRSTM;
60                 CONT=L2;
61                 IF CODE < 0 THEN DO;
62                                  I=LABEL;
63                                  CALL FSYMBOL (X,Y,HEIGHT,I,ANGLE,-1B);
64                                  END;
65                             ELSE GO TO L3;
66                 RETURN;
67              L2:I=LENGTH(LABEL);
68              CALL FSYMBOL (X,Y,HEIGHT,XXX###Y(1), ANGLE,I);
69                 RETURN;
70            END PLPLOT;
```

ATTRIBUTE AND CROSS-REFERENCE TABLE

DCL NO.	IDENTIFIER	ATTRIBUTES AND REFERENCES
4	A	(*),PARAMETER,DECIMAL,FLOAT(SINGLE) 18,38,41,42
4	ABS	(*),PARAMETER,DECIMAL,FLOAT(SINGLE) 58,61,62
4	ADELTA	PARAMETER,DECIMAL,FLOAT(SINGLE) 38,42,44,56
4	AMIN	PARAMETER,DECIMAL,FLOAT(SINGLE) 38,42,44,56
4	ANGLE	PARAMETER,DECIMAL,FLOAT(SINGLE) 44,56,64,71,76
44	AXIS	ENTRY,DECIMAL,FLOAT(SINGLE)
	CEIL	GENERIC,BUILT-IN FUNCTION 48
5	CODE	PARAMETER,BINARY,FIXED(15,0) 21,24,25,44,56,64,68
6	CONT	AUTOMATIC,LABEL 47,54,67
4	FAC	PARAMETER,DECIMAL,FLOAT(SINGLE) 28,31,33,36
2	FACTOR	ENTRY,DECIMAL,FLOAT(SINGLE)
[illegible]	FAXIS	EXTERNAL,ENTRY,DECIMAL,FLOAT(SINGLE) 56
	FFACTOR	EXTERNAL,ENTRY,DECIMAL,FLOAT(SINGLE) 31
	FIRSIM	ENTRY,DECIMAL,FLOAT(SINGLE) 23,30,35,40,46,60,66
	FLINE	EXTERNAL,ENTRY,DECIMAL,FLOAT(SINGLE)

D	NO.	IDENTIFIER	ATTRIBUTES AND REFERENCES
			62
		FPLOT	EXTERNAL,ENTRY,DECIMAL,FLOAT(SINGLE) 15,24
		FPLOTS	EXTERNAL,ENTRY,DECIMAL,FLOAT(SINGLE) 13
		FSCALE	EXTERNAL,ENTRY,DECIMAL,FLOAT(SINGLE) 42
		FSYMBOL	EXTERNAL,ENTRY,DECIMAL,FLOAT(SINGLE) 71,76
		FWHERE	EXTERNAL,ENTRY,DECIMAL,FLOAT(SINGLE) 36
4		HEIGHT	PARAMETER,DECIMAL,FLOAT(SINGLE) 64,71,76
11		I	STATIC,BINARY,FIXED(31,0) 41,42,61,62,62,70,71,75,76
10		I1	STATIC,BINARY,FIXED(15,0) 50,52
10		I2	STATIC,BINARY,FIXED(15,0) 48,50
10		I3	STATIC,BINARY,FIXED(15,0) 49,51,51,52
5		K	PARAMETER,BINARY,FIXED(15,0) 38,42,58,62
4		L	PARAMETER,DECIMAL,FLOAT(SINGLE) 38,42,44,56
56		L1	STATEMENT LABEL CONSTANT 47
75		L2	STATEMENT LABEL CONSTANT 67

PLPLOT:PROC;

DCL NO.	IDENTIFIER	ATTRIBUTES AND REFERENCES
49	L3	STATEMENT LABEL CONSTANT 73
2	LABEL	PARAMETER,STRING,CHARACTER,VARYING 44,48,52,56,64,70,75
	LBOUND	GENERIC,BUILT-IN FUNCTION 41,61
	LENGTH	GENERIC,BUILT-IN FUNCTION 48,56,75
58	LINE	ENTRY,BINARY,FIXED(15,0)
8	LL	STATIC,INITIAL,BINARY,FIXED(15,0) 13
5	M	PARAMETER,BINARY,FIXED(15,0) 58,62
5	MM	PARAMETER,BINARY,FIXED(15,0) 58,62
5	N	PARAMETER,BINARY,FIXED(15,0) 18,38,42,58,62
4	ORD	(*),PARAMETER,DECIMAL,FLOAT(SINGLE) 58,62
21	PLOT	ENTRY,DECIMAL,FLOAT(SINGLE)
18	PLOTS	ENTRY,DECIMAL,FLOAT(SINGLE)
1	PLPLOT	ENTRY,DECIMAL,FLOAT(SINGLE)
38	SCALE	ENTRY,DECIMAL,FLOAT(SINGLE)
	SIGN	GENERIC,BUILT-IN FUNCTION 56
	SUBSTR	GENERIC,BUILT-IN FUNCTION 52
9	SWITCH	STATIC,INITIAL,BINARY,FIXED(15,0)

PLPLOT:PROC;

DCL NO.	IDENTIFIER	ATTRIBUTES AND REFERENCES
		14,22,29,34,39,45,59,65
64	SYMBOL	ENTRY,DECIMAL,FLOAT(SINGLE)
	SYSPRINT	FILE,EXTERNAL 19
50	UNSPCLP	STATEMENT LABEL CONSTANT
	UNSPEC	GENERIC,BUILT-IN FUNCTION 52,52
33	WHERE	ENTRY,DECIMAL,FLOAT(SINGLE)
4	X	PARAMETER,DECIMAL,FLOAT(SINGLE) 21,24,33,36,44,56,64,71,76
7	XXX###X	(*),STATIC,DECIMAL,FLOAT(SINGLE) 13
3	XXX###Y	(*),STATIC,DECIMAL,FLOAT(SINGLE) 52,56,76
4	Y	PARAMETER,DECIMAL,FLOAT(SINGLE) 21,24,33,36,44,56,64,71,76

```
        PLPLOT:PROC:                                                                    PAGE

 COMPILER DIAGNOSTIC MESSAGES

THE FOLLOWING WARNING MESSAGES HAVE BEEN GENERATED.

IBM0841I      54   USE OF LABEL VARIABLE CONT MAY RESULT IN AN ILLEGAL BRANCH IN STATEMENT NUMBER 54

IBM0227I           NO FILE/STRING OPTION SPECIFIED IN ONE OR MORE GET/PUT STATEMENTS. SYSIN/SYSPRINT HAS BEEN
                   ASSUMED IN EACH CASE.


END OF COMPILER DIAGNOSTIC MESSAGES.

TIME FOR THIS COMPILATION WAS          .96 MINUTES
```



```
IEF285I   CTOS.PL1SOURC                                   KEPT
IEF285I   VOL SER NOS= MSD003.
IEF285I   LIN                                             KEPT
IEF285I   VOL SER NOS= MSD001.
IEF285I   CTOS.UT3                                        KEPT
IEF285I   VOL SER NOS= MSD003.
IEF285I   CTOS.UT4                                        KEPT
IEF285I   VOL SER NOS= MSD004.
IEF285I   SYSOUT                                          SYSOUT
IEF285I   VOL SER NOS=
* CARA      PL1       STEP COST=  $3.04  RUN TIME=00.01.06  DATE=67.124  TIME=16.11.56 MSDD-MDD50-D510 *
//USERLIBS EXEC PGM=CTOSLMOD,COND=(9,LT,PL1)   DELETE LMOD                    00200000
//LKED     EXEC PGM=IEWL,PARM=(MAP,LIST,LET,NCAL),COND=(9,LT,PL1)            00220000
//SYSLIB   DD   DSNAME=CTOS.PL1SUB,DISP=OLD                                   00240000
//         DD   DSNAME=SYS1.PL1LIB,DISP=OLD                                   00260000
//         DD   DSNAME=SYS1.FORTLIB,DISP=OLD                                  00280000
//         DD   DSNAME=CTOS.PLOTLIB,DISP=OLD                                  00300000
//         DD   DSNAME=CTOS.SUBLIB,DISP=OLD                                   00320000
//         DD   DSNAME=CTOS.SCISUBPK,DISP=OLD                                 00340000
//SYSUT1   DD   DSNAME=CTOS.UT3,DISP=OLD                                      00360000
//SYSLMOD  DD   DDNAME=LOAD                                                   00380000
//SYSPRINT DD   SYSOUT=A                                                      00400000
//SYSLIN   DD   DSNAME=*.PL1.SYSLIN,DISP=OLD                                  00420000
//         DD   DDNAME=SYSIN  LINKAGE EDITOR NAME STATEMENT VIA SYSIN         00440000
//LKED.LOAD DD  DSNAME=CTOS.PL1SUB,DISP=OLD,VOLUME=REF=CTOS.PL1SUB
//LOAD     DD   SPACE=(TRK,(25,10,1)),DSNAME=LMOD(PLIMAIN),                  X00460000
//              DISP=(NEW,PASS),VOLUME=REF=SYS1.SVCLIB                        00480000
//LKED.PRIVLIB DD DSNAME=CTOS.PLOTLIB,DISP=OLD
//LKED.SYSIN DD *
IEF236I ALLOC. FOR CARA     LKED
IEF237I SYSLIB   ON 191
IEF237I          ON 190
IEF237I          ON 190
IEF237I          ON 191
IEF237I          ON 191
IEF237I          ON 191
IEF237I SYSUT1   ON 191
IEF237I SYSLMOD  ON 191
IEF237I SYSLIN   ON 190
IEF237I          ON 00C
IEF237I PRIVLIB  ON 191
```

```
IEW0461 IHEVPRA
IEW0461 IHEERRB
IEW0461 IHEVTRA
IEW0461 IHEVPCA
IEW0461 IHEVPDA
IEW0461 IHEVPHA
IEW0461 IHEVFAA
IEW0461 IHEVFCA
IEW0461 IHEVFDA
IEW0461 IHEVFEA
IEW0461 IHEVKBA
IEW0461 IHEVKCA
IEW0461 IHEVKFA
IEW0461 IHEVSGA
IEW0461 IHEUPRA
IEW0461 SIN
IEW0461 IHEUPRB
IEW0461 IHESADA
IEW0461 IHESAOA
IEW0461 IHESAFD
IEW0461 IHERSME
IEW0461 IHECSMV
IEW0461 IHEL00A
```



```
IEW0461 IHESPRT
IEW0461 IHEF00A
IEW0461 IHEOSPA
IEW0461 IHESAFC
IEW0461 IHESAFB
IEW0461 IHESAFA
IEW0461 IHECSS3
IEW0461 COS
IEW0461 ALOG10
IEW0461 ALOG
IEW0461 FRXPR#
IEW0461 FRXPI#
IEW0461 SQRT
IEW0000 PLOT      NOW REPLACED IN DATA SET
IEW0000 LINE      IS AN ALIAS FOR THIS MEMBER
IEW0000 SYMBOL    IS AN ALIAS FOR THIS MEMBER
IEW0000 AXIS      IS AN ALIAS FOR THIS MEMBER
IEW0000 SCALE     IS AN ALIAS FOR THIS MEMBER
```

---- DIAGNOSTIC MESSAGE DIRECTORY ----

IEW0461 WARNING - SYMBOL PRINTED IS AN UNRESOLVED EXTERNAL REFERENCE, NCAL WAS SPECIFIED.

---- MODULE MAP ----

CONTROL SECTION

NAME	ORIGIN	LENGTH
IHEVFR	00	D8
IHEVPA	D8	144
IHEVPE	220	2C
IHEDMA	250	F8
IHEVSC	348	AC
IHEUPA	3F8	90
IHEVPE	488	24P
IHEVPG	700	229
IHEDCN	930	28E
PLPLOT	BF0	A68
*PLPLOTA	1658	22CC
IHESPRT	3928	3P

ENTRY

NAME	LOCATION	NAME	LOCATION	NAME	LOCATION	NAME	LOCATION
IHEVFRA	00						
IHEVPAA	D8						
IHEVPEA	220						
IHEDMAA	250						
IHEVSCA	348						
IHEUPAA	3F8	IHEUPAR	44A				
IHEVPEA	488						
IHEVPGA	700						
IHEDCNA	930	IHEDCNB	932				
PLOTS	C50	PLOT	C90	FACTOR	CD8	WHERE	D10
SCALE	D58	AXIS	DAB	LINE	E28	SYMBOL	EBA