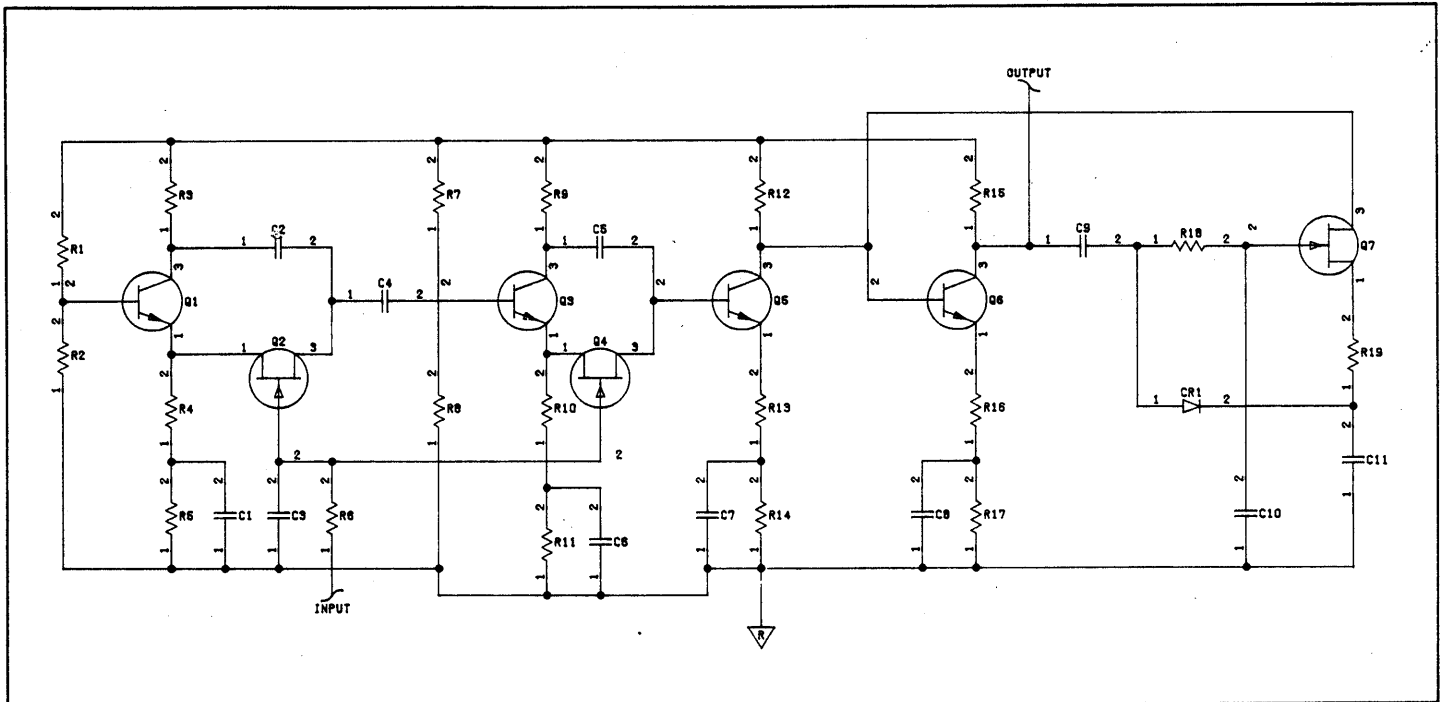# Programming
CalComp
Electromechanical
Plotters

PROGRAMMING
CALCOMP
ELECTROMECHANICAL
PLOTTERS

CONTENTS

ILLUSTRATIONS

SOFTWARE/HARDWARE

INTRODUCTION

This manual provides an introductory technical description of CalComp's Basic Software Package for Electromechanical Plotters. It includes the information that a FORTRAN-oriented programmer needs to write graphic computer programs. CalComp's Basic Software is supported for a large number of computers and operating systems; your local CalComp representative can supply specific details on availability.

TYPES OF SOFTWARE

Calcomp provides graphic software in three general categories:

- Applications Programs

- Functional Software

- Basic Software

The highest level of graphic software is the Applications Program. An Application Program is a complete problem solver. A user need only supply data and select among program options to obtain the desired graphical output — no programming is required on the user's part. Applications Programs are written in a higher level language, usually FORTRAN IV, and are available for a variety of computers.

Typical CalComp applications programs are GPCP (General Purpose Contouring Program) and AUTONET.

Appendix B contains a short description of the applications software packages currently available from CalComp. The Applications Program normally uses a Basic Software Package and may also make use of CalComp Functional Software.

Functional Software is an intermediate level of software which relieves the user of programming many commonly used graphic functions. Functional Software is further subdivided into host-computer resident and controller-resident type software. Host-computer resident functional software is usually written in FORTRAN and grouped into packages by general application:

- BUSINESS

- DRAFTING

- SCIENTIFIC

- GENERAL

Each package consists of a set of subroutines to perform particular operations. For example, one of the subroutines in the General Functional Package is CIRCLE. This subroutine plots a circle, arc or spiral. Some more sophisticated functionals are offered individually e.g., FLOCT, an automatic flow chart generator.

A controller-resident functional software product is a program written in assembly language for a CalComp programmable controller, which permits the user to obtain graphical output more easily and efficiently. Appendix C contains a list of current Functional Software.

The lowest level of software used in producing a plot is termed Basic Software. Depending upon the plotting system, basic software may be supplied only for the host computer or for both the host computer and the controller. The controller basic software is the operating software for the controller and is normally of little concern to the programmer. Host Computer Basic Software (HCBS) consists of a set of subroutines which allow the programmer to perform elementary plotting operations such as drawing lines or annotation, selecting a pen, scaling the plot, etc. The basic software generates the commands necessary to perform the specified operation and transmits them to the plotter if it is on-line, or writes the commands on an appropriate medium (e.g., magnetic tape) for subsequent plotting off-line.

The plot commands generated by the basic software may be the actual codes necessary to move the plotter, but usually they are codes which represent the input data in a highly compact and efficient format. These codes are then interpreted by hardware (in some systems) but normally by software (on a CalComp controller) which then produces the actual codes necessary to drive the plotter.

This manual is intended to provide an explanation of the subroutines included in the Host Computer Basic Software package and how they are used to produce graphical output.

Figure 1-1 shows the relationship between the types of software and gives the overall flow of data in an off-line graphics system.

```
                    ┌──────────┐
                    │CONTROLLER│
                    │BASIC     │
                    │SOFTWARE  │
                    └────┬─────┘
                         │
                         ▼
┌──────┐  ┌───────────┐  ┌───────────┐  ┌──────────┐  ┌──────────┐      ┌─────┐        ┌──────────┐  ┌──────────┐  ┌──────────┐
│USER  │  │APPLICATION│  │FUNCTIONAL │  │BASIC     │  │HOST      │      │PLOT │        │MAGNETIC  │  │CALCOMP   │  │CALCOMP   │
│DATA  │  │SOFTWARE   │  │SOFTWARE   │  │SOFTWARE  │  │COMPUTER  │      │TAPE │        │TAPE UNIT │  │GRAPHIC   │  │PLOTTER   │
└──────┘  └───────────┘  └───────────┘  └──────────┘  └──────────┘      └─────┘        └──────────┘  │CONTROLLER│  └──────────┘
                                                                                                     └──────────┘
                                                                                          ┌──────────┐
                                                                                          │CONTROLLER│
                                                                                          │FUNCTIONAL│
                                                                                          │SOFTWARE  │
                                                                                          └──────────┘
```

HOST COMPUTER   GRAPHIC CONTROLLER
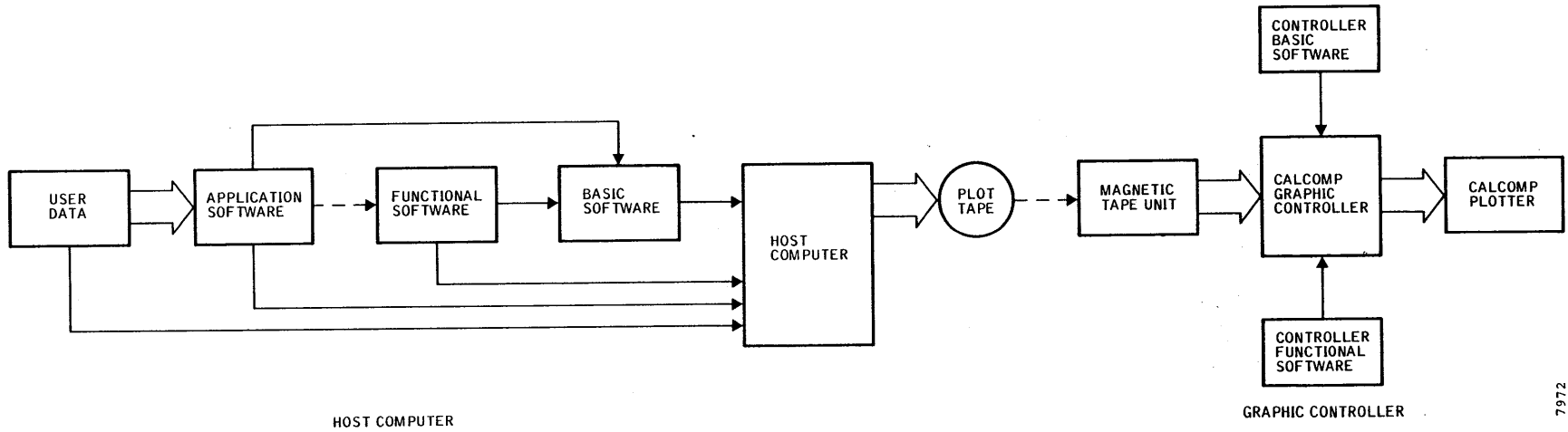
Figure 1-1. Graphic System Operation

COMPATIBILITY FEATURES

The modular design of CalComp hardware/software allows the user to upgrade or modify most CalComp plotting systems without the extensive conversion problems usually associated with hardware modifications. A change in the user's computer or plotting system usually requires little or no modification to operational applications programs.

CALCOMP SOFTWARE SUPPORT POLICY

CalComp software is an integral and essential part of our product line. Like individual hardware items, each software package is given a product number and is supported with appropriate literature and documentation. Because of the proprietary nature of CalComp software, the packages are leased rather than sold outright. The lease agreement allows unrestricted use of the software with your CalComp system and the lease price covers this usage for as long as you retain the system.

CalComp guarantees that its Basic Software will perform according to specifications in effect at the time of installation. For certain on-line systems, CalComp will also assist in the installation and checkout of the Basic Software at your facility.

A SAMPLE PLOTTING PROGRAM

To illustrate the use of CalComp Host Computer Basic Software, a sample program is given which will produce the graph shown in Figure 1-2. The only assumptions made are that: 1) The 24 pairs of TIME and VOLTAGE data values are contained in a file of 24 records; and 2) the plotting pen is initially positioned at the extreme -Y side of the plotter. Notice that only 11 executable statements are required to complete the graph.
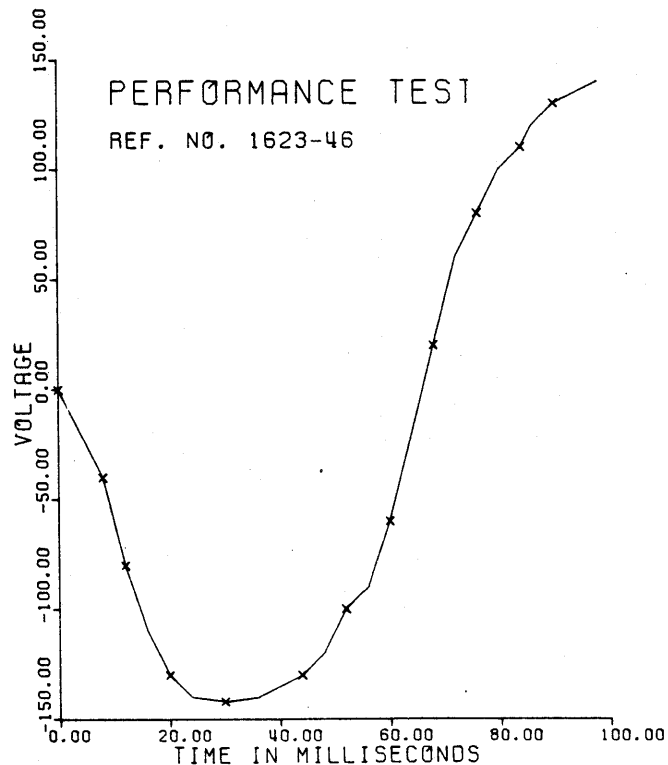
Figure 1-2. Sample Graph Produced on a CalComp Electromechanical Plotter

4

```
      DIMENSION XARRAY (26), YARRAY (26)
          Reserve space for 24 data values plus two additional locations required by the
          SCALE, AXIS, and LINE subroutines.

10    CALL PLOTS (0, 0, 6)
          Initialize the PLOT subroutine with the logical number of the output device.

20    READ 25, (XARRAY (I), YARRAY (I), I = 1, 24)

25    FORMAT (2F6.2)
          Read 24 pairs of TIME and VOLTAGE from an input file into two arrays with
          names XARRAY and YARRAY.

30    CALL PLOT (0.0, 0.5, -3)
          Establish a new origin one-half inch higher than the point where the pen was
          initially placed by the operator so that the annotation of the TIME axis will fit
          between the axis and the edge of the plotting surface.

40    CALL SCALE (XARRAY, 5.0, 24, 1)
          Compute scale factors for use in plotting the TIME values within a five-inch
          plotting area.

50    CALL SCALE (YARRAY, 6.0, 24, 1)
          Compute scale factors for use in plotting the VOLTAGE data values within a
          six-inch plotting area (i.e., the data pairs of TIME, VOLTAGE will plot
          within a five-by-six-inch area).

60    CALL AXIS (0.0, 0.0, 20HTIME IN MILLISECONDS, -20, 5.0, 0.0, XARRAY
                (25), XARRAY (26))
          Draw the TIME axis (5 inches long), using the scale factors computed in state-
          ment 40 to determine the milliseconds at each inch along the TIME axis.

70    CALL AXIS (0.0, 0.0, 7HVOLTAGE, 7, 6.0, 90.0, YARRAY (25), YARRAY (26))
          Draw the VOLTAGE axis (6 inches long), using the scale factors computed in
          statement 50 to determine the voltage at each inch along the VOLTAGE axis.

80    CALL LINE (XARRAY, YARRAY, 24, 1, 2, 4)
          Plot VOLTAGE vs TIME, drawing a line between each of the 24 scaled points
          and a symbol X at every other point.

90    CALL SYMBOL (0.5, 5.6, 0.21, 16HPERFORMANCE TEST, 0.0, 16)
          Plot the first line of the graph title.

100   CALL SYMBOL (0.5, 5.2, 0.14, 16HREF. NO. 1623-46, 0.0, 16)
          Plot the second line of the graph title.

110   CALL PLOT (12.0, 0.0, 999)
          Advance the pen beyond the current plotting area, write a terminating record,
          and close the plot output device.

120   STOP
          Terminate program execution.
      END
```

PLANNING YOUR GRAPH

Graphs and plots, like computer listings, require some planning to achieve a pleasing and effective format. The following checklist of plotting conventions is letter-keyed to Figure 1-3 to help you in your planning.

   Ⓐ   The initial position of the pen when the plotting operation begins is taken as the current logical origin (X=0, Y=0). All pen movements are defined in X and Y coordinates relative to the current origin. Subsequent origins can be established at other positions by appropriate programming (see PLOT).

        If more than one plot is to be drawn, an origin for each should be established, with care taken that the plots do not overlap.

   Ⓑ   On a drum plotter, the X-axis lies parallel to the side with the + X direction toward the back. On a flatbed plotter, the X-axis lies parallel to the front, with the + X direction to the right.
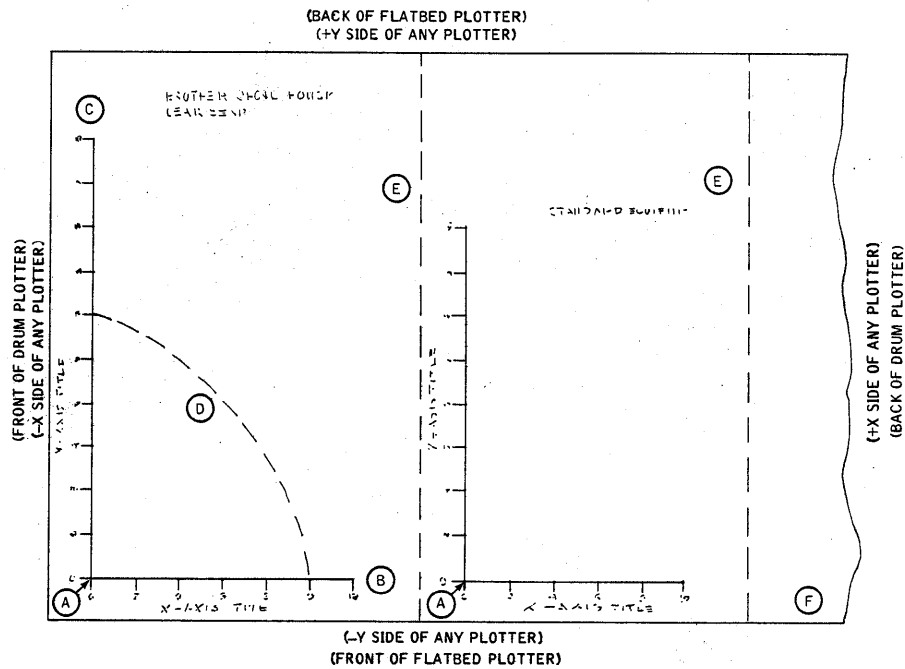
Figure 1-3. Pen Plotting Conventions

The maximum width of a plot in the X-direction is the length of a roll of paper (usually 120 feet) on a drum plotter. On a flatbed plotter, it is normally the length of the bed (from 34 inches to 82 inches, depending on the model). It is possible to produce a plot longer than the bed by plotting it in segments, if special software is employed.

Ⓒ The Y axis of both drum and flatbed plotters is parallel to the beam which carries the pen(s). The maximum height of a plot ranges from 11 inches for small drum plotters to 59 inches for a large flatbed plotter.

Ⓓ The angle of rotation about any point is determined by a vector. When this vector is extended in the + X direction, it represents 0 degrees. Any angle argument used in a calling sequence may be stated in plus or minus degrees relative to the X-axis.

Ⓔ When drawing several plots in one program, it may be desirable to draw trim lines for the operator's convenience in separating the plots.

Ⓕ After the last plot has been drawn, the pen should be moved to a position that permits easy removal of all plots. On drum plotters, this would be several inches beyond the end margin to allow the paper to be torn off the roll. On flatbed plotters, this position would be the extreme -X, -Y corner of the bed.

STANDARD SOFTWARE MAKES IT EASY

The following sections describe in detail CalComp's Host Computer Basic Software (HCBS). This software package consists of a set of subroutines written in FORTRAN and/or assembly language, which control elementary operations of the plotter and provide certain aids in plotting graphs. These subroutines are called by CalComp (and user) written applications programs and host computer functional software. All output to the plotting system should go through the basic software package.

Although the subroutines included in the HCBS package are basically the same for all plotting system configurations, some minor modifications or additional subroutines are provided in certain cases. These changes or additions are fully described in short supplements for particular computer/plotting system combinations. Obtain these supplements from your CalComp representative.

6

The HCBS subroutines are callable directly from FORTRAN programs, and, by observing some special considerations, may also be called from assembly language, PL/I, or COBOL programs.

Standard FORTRAN name conventions are used throughout this manual and in the HCBS package. In cases where the package is written in assembly language, standard FORTRAN linkage conventions are used.

The subroutines included as part of the standard HCBS package are as follows:

PLOT   Plot straight line between two points; establish plot origin; generate search addresses. Also contains four entry points for auxiliary functions:

        PLOTS — Initialization

        FACTOR — Scale entire plot

        WHERE — Returns current pen location

        NEWPEN — Selects pen

SYMBOL   Plots annotation and special symbols.

NUMBER   Plots the decimal equivalent of an internal floating point number.

SCALE   Determines starting value and scale for an array of data to be plotted on a graph.

AXIS   Draws an annotated axis line for a graph.

LINE   Scales and plots a set of data points defined by X and Y coordinate arrays.

Appendix A contains a summary of the standard calling sequences for these subroutines.

PLOT SUBROUTINE

Many graphic applications require the generation of X-Y graphs to show the relationship between two or more sets of data. Usually these graphs can be produced easily and quickly by a suitably programmed combination of the five supporting subroutines SCALE, AXIS, LINE, SYMBOL and NUMBER. These subroutines do not directly produce plotter commands; they only compute appropriate arguments that define pen positions, and then call the PLOT subroutine, which generates the actual plotter commands.

When plotting requirements cannot be satisfied by using the supporting subroutines, the user can call the PLOT subroutine, which gives him direct control of pen movement (to any X, Y coordinate position), pen status (up or down), and generation of Search records. (See SEARCH RECORDS under OPERATING CONSIDERATIONS.)

Four other functions are closely associated with the PLOT operation as follows:

PLOTS   Performs initialization.

FACTOR   Adjusts the overall size of a plot.

WHERE   Returns the current pen location.

NEWPEN   Selects pens.

In some basic software packages, these functions are programmed as separate entry points within the PLOT subroutine. In other software packages, they are separate subroutines which call PLOT with a special value of IPEN to perform the necessary operation. In either case, the calling sequence and function is the same. The discussion here assumes that they are entry points in the PLOT subroutine.

Some basic software packages contain an additional entry point (or subroutine) called BUFF. The BUFF subroutine manages the buffer into which generated plot commands are placed and then written to the output device as necessary. This subroutine should *not* be called directly by the user.

PLOT ENTRY

The PLOT entry to the PLOT subroutine is used to move the pen in a straight line to a new position, with the pen either up or down during the movement. It converts the arguments to the appropriate sequence of plotter commands and outputs these to the attached device (tape, disc, drum or plotter controller). Observe that the PLOTS entry must be called before any other entries are called.

The calling sequence has three arguments:

CALL PLOT   (XPAGE, YPAGE, ±IPEN)

XPAGE, YPAGE   are the X, Y coordinates, in inches from the current reference point (origin), of the position to which the pen is to be moved. An origin (where both X, Y equal zero) may be established anywhere on (or off) the plotting surface, as explained below for negative IPEN values. For metric plotting systems, the X, Y coordinates are in centimeters.

±IPEN   is a signed integer which controls pen status (up or down), origin definition and the generation of Search records.

If IPEN = 2, the pen is *down* during movement, thus drawing a visible line.

If IPEN = 3, the pen is *up* during movement.

9

If IPEN = -2 or -3, a new origin is defined at the terminal position after the movement is completed as if IPEN were positive. The logical X, Y coordinates of the new pen position are set equal to zero. This position is the reference point for succeeding pen movements. All of the plotter commands accumulated in the output buffer are transmitted to the output device.

If the plotting system is off-line, a Search record with the next sequential Search address is also produced. (See SEARCH RECORDS under OPERATING CONSIDERATIONS.)

If IPEN = 999, the effects are the same as if IPEN = -3, except that a Search record with Search address 999 is written (9999 on some systems), and the output device is closed.

The examples in Figure 2-1 show the pen movements that result from a series of calls to the PLOT subroutine. The initial call to PLOTS, as well as a call to FACTOR, are included. Opposite each call is shown the Search-record address that would be produced for an off-line plotting system.

PLOTS ENTRY

The PLOTS entry is used to initialize the PLOT subroutine. It must be called only once (before any other call to PLOT, SYMBOL, NUMBER, AXIS or LINE is made). This entry sets up certain constants and opens the plot output device by performing standard file-opening procedures through the computer's operating system. If the output device is off-line, the first Search record (Search address No. 001) is written. Figure 2-1 includes an example of the use of PLOTS.

The calling sequence has three arguments:

CALL PLOTS   (0, 0, LDEV)

LDEV   is the logical output device number, i.e., the device number of the plotter if it is on-line or the logical number of an output device for off-line plotting.

The first two arguments of the PLOTS subroutine are not used in all systems. Check the appropriate SUPPLEMENT.
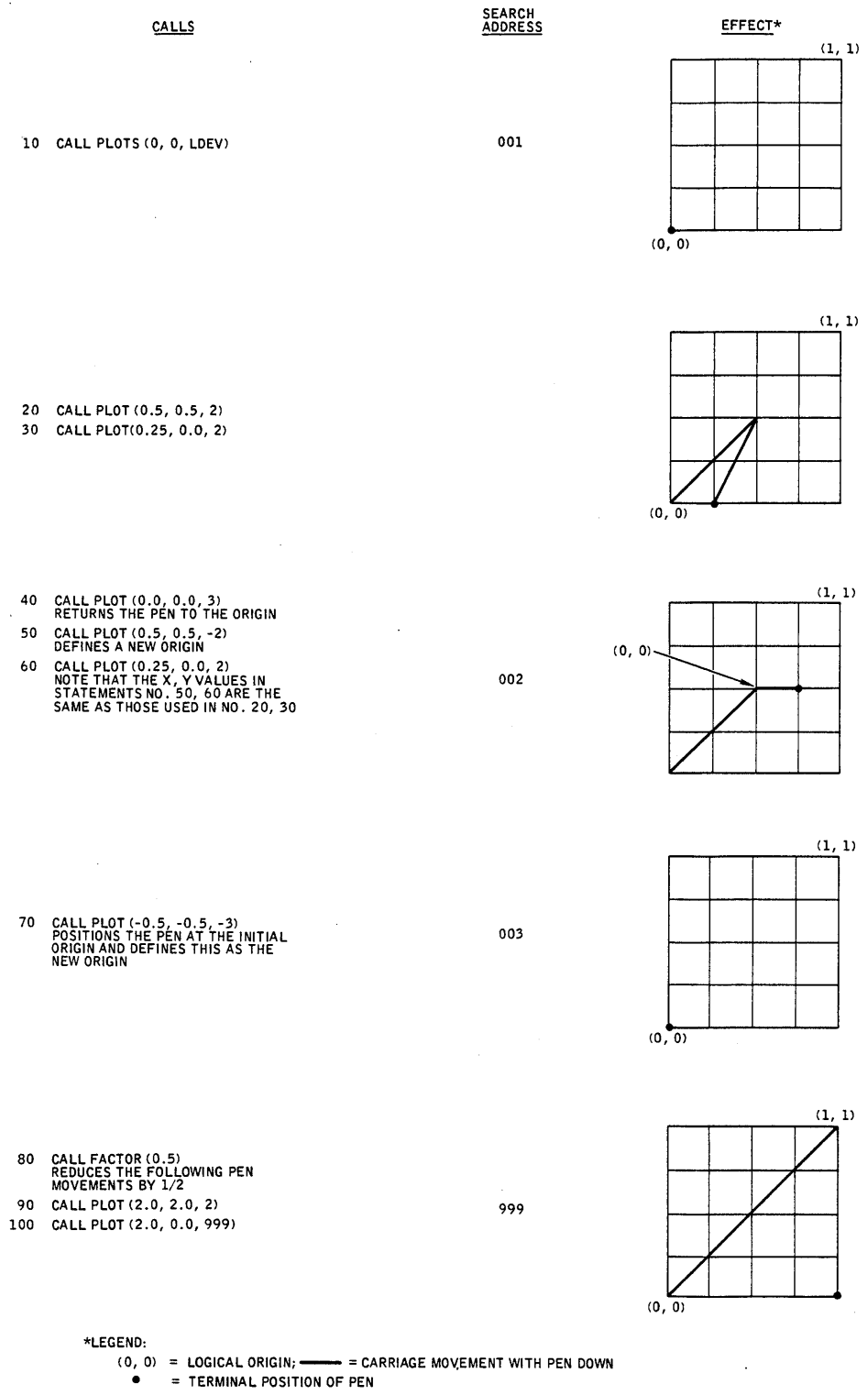
FACTOR ENTRY

The FACTOR entry to the PLOT subroutine enables the user to enlarge or reduce the size of the entire plot by changing the ratio of the desired plot size to the normal plot size. A sample FACTOR statement is shown in Figure 2-1.

CALL FACTOR   (FACT)

FACT   is the ratio of the desired plot size to the normal plot size. For example, if FACT = 2.0, all subsequent pen movements will be twice their normal size. When FACT is reset to 1.0, all plotting returns to normal size. During the debugging of a plotting application program, computer and plotting time can be saved by reducing the size of the entire plot output. This is done by calling FACTOR with a value less than 1.0, after calling PLOTS. When debugging is completed, this call statement can be removed.

WHERE ENTRY

The WHERE entry sets the three variables designated in the calling sequence to the pen's current position coordinates and the plot-sizing factor (that are in use by the PLOT subroutine). This permits user-written subroutines to know the pen's current location for optimizing pen movement.

CALLS	SEARCH ADDRESS	EFFECT*

| | | (1, 1) |

10   CALL PLOTS (0, 0, LDEV)   001

(0, 0)

| | | (1, 1) |

20   CALL PLOT (0.5, 0.5, 2)
30   CALL PLOT(0.25, 0.0, 2)

(0, 0)

| | | (1, 1) |

40   CALL PLOT (0.0, 0.0, 3)
    RETURNS THE PEN TO THE ORIGIN
50   CALL PLOT (0.5, 0.5, -2)
    DEFINES A NEW ORIGIN
60   CALL PLOT (0.25, 0.0, 2)   002
    NOTE THAT THE X, Y VALUES IN
    STATEMENTS NO. 50, 60 ARE THE
    SAME AS THOSE USED IN NO. 20, 30

(0, 0)

| | | (1, 1) |

70   CALL PLOT (-0.5, -0.5, -3)   003
    POSITIONS THE PEN AT THE INITIAL
    ORIGIN AND DEFINES THIS AS THE
    NEW ORIGIN

(0, 0)

| | | (1, 1) |

80   CALL FACTOR (0.5)
    REDUCES THE FOLLOWING PEN
    MOVEMENTS BY 1/2
90   CALL PLOT (2.0, 2.0, 2)   999
100 CALL PLOT (2.0, 0.0, 999)

(0, 0)

*LEGEND:
   (0, 0) = LOGICAL ORIGIN; ——— = CARRIAGE MOVEMENT WITH PEN DOWN
    ●   = TERMINAL POSITION OF PEN

7997

Figure 2-1. Sample Calls to PLOT

11

The calling sequence has three arguments:

CALL WHERE   (RXPAGE, RYPAGE, RFACT)

RXPAGE,   are variables that will be set to the pen's current position coordinates
RYPAGE   resulting from the previous call to PLOT (which may have been
called by SYMBOL, NUMBER , AXIS or LINE).

RFACT   is set to the current plot-sizing factor; i.e., the value last supplied by
a call to FACTOR or 1.0 if FACTOR has not been called.

NOTE

*A call to WHERE made after a call to SYMBOL returns different values
depending upon the plotting system being used. These values include: 1)
the coordinates of the starting point of the character string; 2) the coor-
dinates of the last point actually plotted; 3) the coordinates of the start-
ing point of the next character to be plotted.*

NEWPEN ENTRY

The NEWPEN entry enables program selection of any of the available pens for multipen plot-
ters. If a call to NEWPEN is made for a single pen plotter, it is ignored. Pen No. 1 is the
bottommost of the pens, which are spaced 0.6 inch apart in the Y-axis direction. Pen No. 1 is
initially selected when the PLOTS entry is called.

The calling sequence is as follows:

CALL NEWPEN   (INP)

INP   specifies the number of the pen to be selected. The old pen is raised
and the new pen is moved to the same physical location at which the
old pen was positioned. The user must guard against the situation
where the new pen cannot be moved to the previous pen's position.
This can happen if the distance between the two positions is greater
than the distance from the old pen's position to either Y-axis limit
(top or bottom edge of the plotter).

The NEWPEN entry can also be used with optional attachments to CalComp flatbed plotters.
These attachments include CalComp Film Cutters. For these attachments, INP specifies the
number of the cutting blade to be selected.

SYMBOL SUBROUTINE

The SYMBOL subroutine produces plot annotation at any angle and in practically any size. There are two SYMBOL call formats: 1) the "standard" call, which can be used to draw text such as titles, captions, and legends; and 2) the "special" call, which is used to draw special centered symbols such as a box, octagon, triangle, etc., for plotting data points.

The standard characters that are drawn by SYMBOL include the letters A-Z, digits 0-9, and certain special characters. See your computer-oriented supplement to this manual for other characters available in your particular SYMBOL subroutine. Characters available in a given SYMBOL subroutine will vary depending upon the host computer and the CalComp plotting system being used.

Both forms of the SYMBOL calling sequences have six arguments.

The "standard" call is:

CALL SYMBOL (XPAGE, YPAGE, HEIGHT, IBCD, ANGLE, +NCHAR)

XPAGE, YPAGE	are the coordinates, in inches, of the lower left-hand corner (before character rotation) of the first character to be produced. The pen is up while moving to this point.
	Annotation may be continued from the position following that at which the last annotation ended. Continuation occurs when XPAGE and/or YPAGE equals 999.0, and may be applied to X or Y independently. (Calling WHERE to obtain the current pen position and using RXPAGE, RYPAGE in another call to SYMBOL would not give the same results as using 999.)
HEIGHT	is the height, in inches, of the character to be plotted. The width of a character, including spacing, is normally the same as the height (e.g., a string of 10 characters 0.14 inch high is 1.4 inches wide).
IBCD	is the text, in internal computer representation (usually BCD or A-type format), to be used as annotation. The character(s) must be left-justified and contiguous in: a single variable, an array, or a Hollerith literal (if the compiler permits). Blanks in the text do not cause any pen movement until the next nonblank character is started.
	The text must be right-justified in IBCD if a single character is desired and NCHAR = 0.
ANGLE	is the angle, in degrees from the X-axis, at which the annotation is to be plotted. If ANGLE = 0.0, the character(s) will be plotted right side up and parallel to the X-axis.
+NCHAR	is the number of characters to be plotted from IBCD. If NCHAR >0, the data must be left-justified in the first element of IBCD.
	If NCHAR = 0, one alphanumeric character is produced, using a single character which is right-justified in the first element of IBCD.

Some examples of using the "standard" call to SYMBOL are shown in Figure 3-1.

The second form is the "special" call, which produces only a single symbol based on the index value of INTEQ — not on the BCD representation of a character.

13

```
DIMENSION IBCD (4)
CALL SYMBOL (X,Y,.28,IBCD,0.,16)
NOTE: IBCD is a 4-word array containing 16 characters.
```

A SAMPLE TITLE

```
CALL SYMBOL  (X,Y,.14,IBCD,0.,16)
CALL SYMBOL  (999.,999.,.14,IBCD,90.,16)
CALL SYMBOL  (999.,999.,.14,IBCD,180.,16)
CALL SYMBOL  (999.,999.,.14,IBCD,270.,16)
NOTE: Two spaces follow "A SAMPLE TITLE"
```

A SAMPLE TITLE

A SAMPLE TITLE

A SAMPLE TITLE

A SAMPLE TITLE

Figure 3-1. Sample Calls to SYMBOL

The "special" call is:

CALL SYMBOL (XPAGE, YPAGE, HEIGHT, INTEQ, ANGLE, —ICODE)

XPAGE, YPAGE, and ANGLE	are the same as described for the "standard" call. If the symbol to be produced is one of the centered symbols (i.e., if INTEQ is less than 14), XPAGE, YPAGE represent the geometric center of the character produced.
HEIGHT	is the height (and width), in inches, of the centered symbol to be drawn. Preferably, it should be a multiple of four times the plotter's increment size.
INTEQ	is the integer equivalent of the desired symbol. Valid integers and their symbols are listed in the Symbol Table of the applicable computer supplement. If INTEQ is 0 through 13, a centered symbol is produced. (See Figure 4-1.)
—ICODE	is negative and determines whether the pen is up or down during the move to XPAGE, YPAGE.

When —ICODE is:

—1, the pen is up during the move, after which a single symbol is produced.

—2, or less, the pen is down during the move, after which a single symbol is produced.

Figure 3-2 shows the symbols available when a CalComp 925 Controller is used. The table shows the integer equivalents for each symbol which are used in the "special" call. When a "standard" call to SYMBOL is made, the host computer's internal characters are translated to the appropriate characters from this table.

For other plotting systems, refer to the appropriate supplement to this manual for the characters available, or run the test program TABLE supplied with the basic software. TABLE generates the table of characters available for a particular plotting system.

NOTE

Some SYMBOL subroutines have a restriction on the maximum height allowed. The maximum height varies depending upon the host computer and the CalComp plotting system being used. The maximum height can be as small as 0.9 inches. If the maximum is exceeded, either the maximum or the previous value of HEIGHT will be used.

ASCII CHARACTERS AVAILABLE WITH THE SYMBOL ROUTINE
CODE NEXT TO EACH SYMBOL IS INTEGER CODE USED IN SPECIAL SYMBOL CALL.

Figure 3-2. 925 Controller Standard Symbol Set

NUMBER SUBROUTINE

NUMBER converts a floating-point number to the appropriate decimal equivalent so that the number may be plotted in the FORTRAN F-type format.

The NUMBER calling sequence has six arguments:

CALL NUMBER (XPAGE, YPAGE, HEIGHT, FPN, ANGLE, ±NDEC)

XPAGE, YPAGE, HEIGHT, and ANGLE	are the same as those arguments described for subroutine SYMBOL. The continuation feature, where XPAGE or YPAGE equals 999.0, may also be used.
FPN	is the floating-point number that is to be converted and plotted.
±NDEC	controls the precision of the conversion of the number FPN. If the value of NDEC > 0, it specifies the number of digits to the right of the decimal point that are to be converted and plotted, after proper rounding. For example, assume an internal value (perhaps in binary form) of —0.12345678 x 10^3. If NDEC were 2, the plotted number would be —123.46.

If NDEC = 0, only the number's integer portion and a decimal point are plotted, after rounding.

If NDEC = —1, only the number's integer portion is plotted, after rounding. (The above example would be plotted as —123 with no decimal point.)

If NDEC < —1, |NDEC| —1 digits are truncated from the integer portion, after rounding.
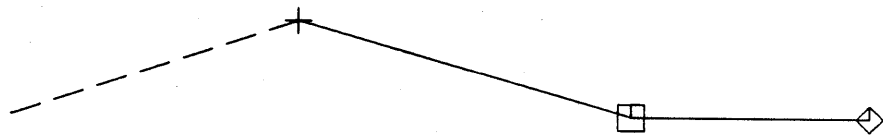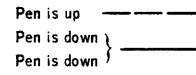
The magnitude of NDEC should not exceed 9.

Figure 4-1 illustrates various uses of SYMBOL and NUMBER.

17

A: Centered Symbols

□ ⊕ △ + ✕ ◇ ⬀ ⊠ ⟋Z Y ⬚ ✳ ⋈ |
0 1 2 3 4 5 6 7 8 9 10 11 12 13
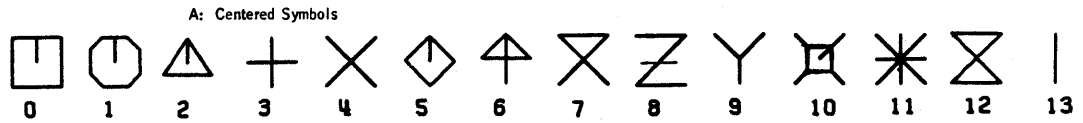
B: Plotting Data Points

CALL SYMBOL (1.5,0.5,.14,3,0.,-1)
CALL SYMBOL (3.25,0.,.14,0,0.,-2)
CALL SYMBOL (4.5,0.0,.14,5,0.,-2)

Pen is up — — — —
Pen is down ⎫
Pen is down ⎭ ———

C: Combining SYMBOL and NUMBER and Drawing a Superscript

CALL SYMBOL (X,Y,.14,10HVALUE OF X,0.,10)
CALL SYMBOL (999.,Y+.1,.07,2H2 ,0.,2)
CALL SYMBOL (999.,Y,.14,2H = ,0.,2)
CALL NUMBER (999.,999.,.14,VALUE,0.,3)

Superscript ,

$$VALUE\ OF\ X^2 = 12.123$$

D: Drawing Text and Numbers at Various Angles

DO 10 I=0,315,45
ANGLE = I
CALL SYMBOL (X,Y,.1,7H ANGLE = ,ANGLE,7)
10   CALL NUMBER (999.,999.,.1,ANGLE,ANGLE,-1)

ANGLE=90
ANGLE=135
ANGLE=45
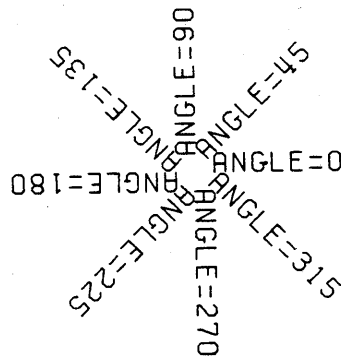ANGLE=180
ANGLE=0
ANGLE=225
ANGLE=270
ANGLE=315

Figure 4-1. Sample Uses of SYMBOL and NUMBER

18

SCALE SUBROUTINE

Typically, the user's program will accumulate plotting data in two arrays:

- An array of independent variables, X_i

- An array of dependent variables, $Y_i = f(X_i)$

It would be unusual if the range of values in each array corresponded exactly with the number of inches available in the actual plotting area. For some problems the range of data is predictable. The programmer can predetermine suitable conversion factors for use in drawing the axis scale values and plotting the data points on the graph. Usually, however, these factors are not known in advance.

Therefore, the SCALE subroutine is used to examine the data values in an array and to determine a starting value (minimum or maximum) and a scaling factor (positive or negative) such that: 1) The scale annotation drawn by the AXIS subroutine at each division will properly represent the range of real data values in the array; and 2) The data points, when plotted by the LINE subroutine, will fit in a given plotting area. These two values are computed and stored by SCALE at the end of the array.

The scaling factor (DELTAV) that is computed represents the number of data units per inch of axis, but is adjusted so that it is always an interval of 1, 2, 4, 5, or 8×10^n (where n is an exponent consistent with the original unadjusted scaling factor). Thus, an array may have a range of values from 301 to 912, to be plotted over an axis of 10 inches. The unadjusted scaling factor is $(912\text{-}301)/10 = 61.1$ units/inch. The adjusted DELTAV would be $8 \times 10^1 = 80$.

The starting value (FIRSTV), which will appear as the first annotation on the axis, is computed as some multiple of DELTAV that is equal to or outside the limits of the data in the array. For the example given above, if a minimum is wanted for FIRSTV, 240 would be chosen as the best value. If a maximum is desired instead, 960 would be selected. In some instances, FIRSTV is selected as a downward-rounded value of the lowest actual data, and in other instances, the DELTAV is adjusted upwards. An attempt is then made to center the data.

There are four arguments in the calling sequence:

CALL SCALE (ARRAY, AXLEN, NPTS, ±INC)

ARRAY	is the first element of the array of data points to be examined.
AXLEN	is the length of the axis to which the data is to be scaled. Its value must be greater than 1.0 inch.
NPTS	is the number of data values to be scanned in the array. The FORTRAN DIMENSION statement should specify at least two elements more than the number of values being scanned, to allow room for SCALE to store the computed starting value and scaling factor at the end of the array.
±INC	is an integer whose magnitude is used by SCALE as the increment with which to select the data values to be scanned in the array. Normally $|INC| = 1$; if it is 2, every other value is examined.

If INC is positive, the selected starting value (FIRSTV) approximates a minimum, and the scale factor (DELTAV) is positive.

If INC is negative, the selected starting value (FIRSTV) approximates a maximum, and the scaling factor (DELTAV) is negative.

If INC = ±1, the array must be dimensioned at least two elements larger than the actual number of data values it contains. If the magnitude of INC > 1, the computed values are stored at (INC) elements and (2*INC) elements beyond the last data point. The subscripted element for FIRSTV is ARRAY (NPTS*INC+1); for DELTAV it is ARRAY (NPTS*INC+INC+1).
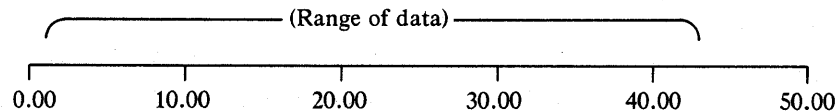
Generally, SCALE is called to examine each array to be plotted, as shown earlier in A SAMPLE PLOTTING PROGRAM. If the user knows the range of his data values, he does not have to call SCALE for that array so long as he supplies an appropriate FIRSTV and DELTAV when AXIS and LINE are called.

The following examples illustrate some typical uses of SCALE:

Example 1 — Given an array of 24 data values to be plotted over a 5-inch axis, assume the minimum value in the array is 1.00 and the maximum is 42.00. The statement

CALL SCALE (ARRAY, 5.0, 24, +1) would give the following results:

Units/inch = (42.00 -1.00)/5.0 = 8.2
DELTAV (next higher interval) = 10.0
FIRSTV (minimum multiple) = 0.00

FIRSTV value is stored in ARRAY (25)
DELTAV value is stored in ARRAY (26)
Using these values, AXIS would draw the following axis line:

```
       ┌──────────────── (Range of data) ────────────────┐
   ┌────────┬────────┬────────┬────────┬────────┬
  0.00    10.00    20.00    30.00    40.00    50.00
```

Example 2 — Assume that the array in Example 1 is to be plotted on a 4-inch axis, from maximum to minimum.

CALL SCALE (ARRAY, 4.0, 24, -1) would give these results.

DELTAV = (1.00 -42.00)/4.0 = -10.25, which is adjusted to -20.
Minimum multiple = 0.00; FIRSTV = Minimum + (AXLEN * |DELTAV|) = 80.00.
In this example, the following axis would be drawn:

```
          ┌──── (Range of data) ────┐
   ┌────────┬────────┬────────┬────────┬
 80.00    60.00    40.00    20.00    0.00
```
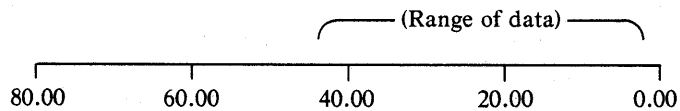
Example 3 — Assume 100 points are to be plotted on a 4-inch axis from maximum (+22) to minimum (-9), using *every other* data value in the array. The DIMENSION statement should specify ARRAY (204) and the calling sequence is

CALL SCALE (ARRAY, 4.0, 100, -2).

Initial DELTAV = (-9 -22)/4 = -7.75, adjusted to -8.
Initial FIRSTV = +16.00; last value on axis = -16.00.

Axis range is inadequate for the data range, so FIRSTV is revised so the last value on the axis is the minimum data value truncated to the same significance as DELTAV.
Revised DELTAV = -8.00, stored in ARRAY (203).
Revised FIRSTV = 23.00 stored in ARRAY (201).

The resulting axis would appear as follows:

```
       ┌──────────────── (Range of data) ────────────────┐
   ┌────────┬────────┬────────┬────────┬
 23.00    15.00    7.00    -1.00    -9.00
```
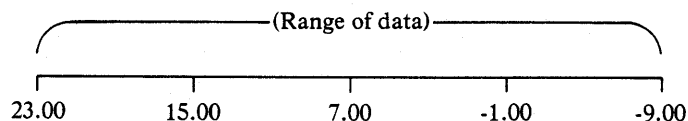
AXIS SUBROUTINE

Most graphs require axis lines and scales to indicate the orientation and values of the plotted data points. The most common type of scaled axis is produced by the AXIS subroutine which draws any length line at any angle, divides the line into one-inch segments, annotates the divisions with appropriate scale values and labels the axis with a centered title. When both the X and Y axes are needed, AXIS is called separately for each one.

The eight arguments in the calling sequence are as follows:

CALL AXIS (XPAGE, YPAGE, IBCD, ±NCHAR, AXLEN, ANGLE, FIRSTV, DELTAV)

|	
XPAGE, YPAGE	are the coordinates, in inches, of the axis line's starting point. The entire line should be at least one-half inch from any side to allow space for the scale annotation and axis title. Usually, both the X and Y axes are joined at the origin of the graph, where XPAGE and YPAGE equal zero; but other starting points can be used. When using the LINE subroutine to plot data on an axis, at least one of the coordinates must be 0, i.e., for an X-axis, XPAGE = 0, and for a Y-axis, YPAGE = 0.
IBCD	is the title, which is centered and placed parallel to the axis line. This parameter may be an alphameric array, or it may be a Hollerith literal if the FORTRAN compiler being used permits it. The characters have a fixed height of 0.14 inch (about seven characters per inch).
±NCHAR	The magnitude specifies the number of characters in the axis title, and the sign determines on which side of the line the scale (tick) marks and labeling information shall be placed. Since the axis line may be drawn at any angle, the line itself is used as a reference.
	If the sign is *positive*, all annotation appears on the positive (counterclockwise) side of the axis. This condition is normally desired for the Y-axis.
	If the sign is *negative*, all annotation appears on the negative (clockwise) side of the axis. This condition is normally desired for the X-axis.
AXLEN	is the length of the axis line, in inches.
ANGLE	is the angle in degrees (positive or negative), at which the axis is drawn. The value is 0° for the X-axis and 90° for the Y-axis.
FIRSTV	is the starting value (either minimum or maximum) which will appear at the first tick mark on the axis. This value may either be computed by the SCALE subroutine and stored at subscripted location ARRAY (NPTS*INC+1), or the value may be determined by the user and stored at any location.
	This number and scale value along the axis is drawn with two decimal places. Since the digit size is 0.105 inch (about 10 characters per inch), and since a scale value appears every inch, no more than six digits and a sign should appear to the left of the decimal point.
DELTAV	represents the number of data units per inch of axis. This value (increment or decrement), which is added to FIRSTV for each succeeding one-inch division along the axis, may either be computed by SCALE and stored beyond FIRSTV at ARRAY (NPTS*INC+INC+1), or the value may be determined by the user and stored at any location.
	In order to use a standard format of two decimal places, the size of DELTAV is adjusted to less than 100, but not less than 0.01. As a result, the decimal point may be shifted left or right in the scale values as drawn, and the axis title is then followed by "$*10^n$", where n is the power-of-ten adjustment factor. (See X-axis example in Figure 6-1.)

21
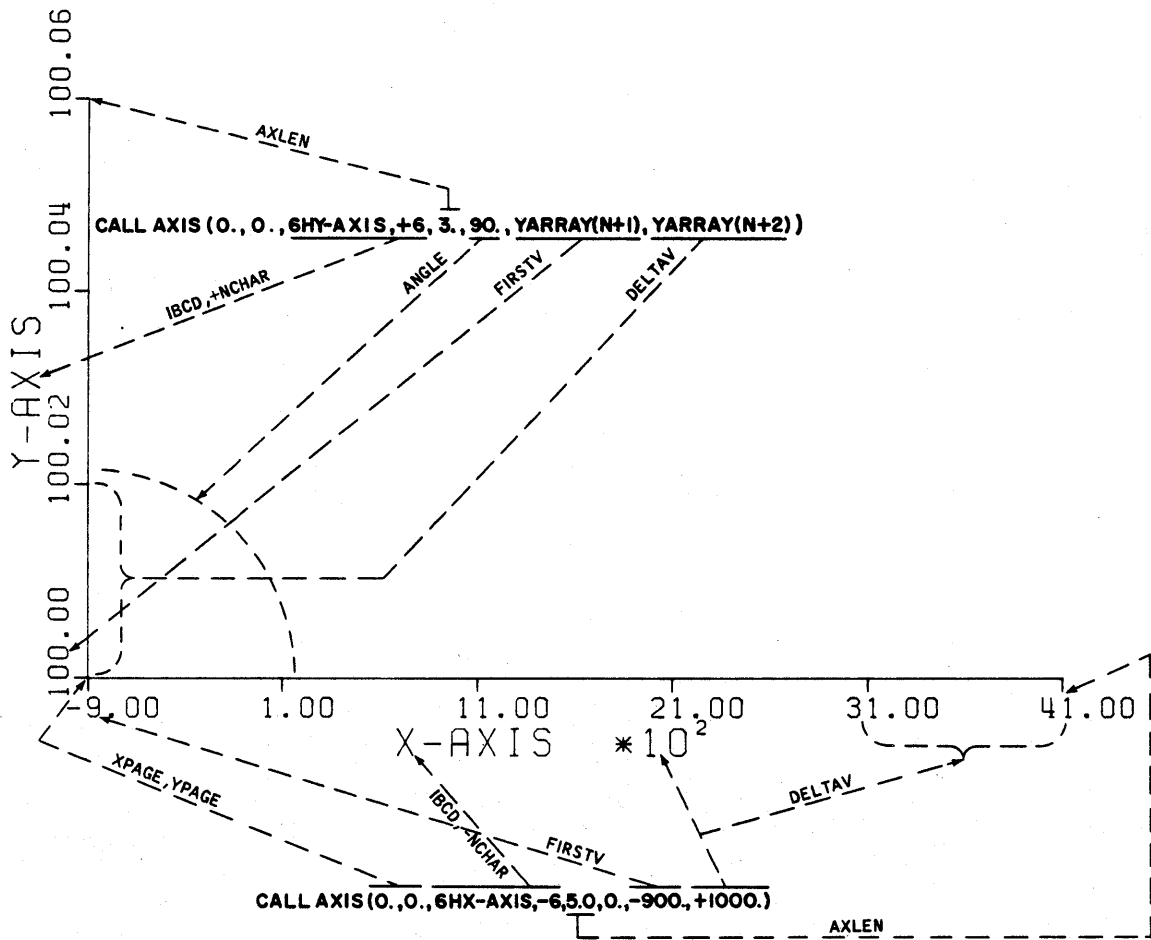
Figure 6-1. Typical X and Y Axis Elements Controlled by AXIS

NOTE

Figure 6-1 illustrates X and Y axis elements controlled by the arguments of AXIS. In the metric version of the AXIS subroutine, the tick marks are drawn at each centimeter, but only every other centimeter is annotated.

LINE SUBROUTINE

The LINE subroutine produces a line plot of the pairs of data values in two arrays (X and Y). LINE computes the page coordinates of each plotted point according to the data values in each array and the respective scaling parameters. The data points may be represented by centered symbols and/or connecting lines between points.

The scaling parameters corresponding to FIRSTV and DELTAV (see SCALE) must immediately follow each array. If these parameters have not been computed by the SCALE subroutine they must be supplied by the user.

The calling sequence has six arguments:

CALL LINE (XARRAY, YARRAY, NPTS, INC, ±LINTYP, INTEQ)

XARRAY   is the name of the array containing the abscissa (X) values and the scaling parameters for the X-array.

YARRAY   is the name of the array containing the ordinate (Y) values and the scaling parameters for the Y-array.

NPTS   is the number of data points in each of the two arrays just mentioned. The number does not include the extra two locations for the scaling parameters. The number of points in each array must be the same.

INC   is the increment that the LINE subroutine is to use in gathering data from the two arrays, as described previously for the SCALE subroutine.

±LINTYP   is a control parameter which describes the type of line to be drawn through the data points. The magnitude of LINTYP determines the frequency of plotted symbols, e.g., if LINTYP = 4, a special symbol (denoted by INTEQ) is plotted at every fourth data point.

                 If LINTYP is *zero*, the points are connected by straight lines but no symbols are plotted.

                 If LINTYP is *positive*, a straight line connects every data point defined in the array. (The pen is up when moving from its current position to the first point.)

                 If LINTYP is *negative*, no connecting lines are drawn; only the symbols are plotted.

INTEQ   is the integer equivalent of the special plotting symbol centered at each data point. This value normally can be 0 through 13, and has meaning only when LINTYP is not zero. Part A of Figure 4-1 lists the symbols that are available. Some of these symbols are as follows: box, octagon, triangle, plus, X, diamond, and asterisk.

                 Figure 7-1 illustrates the types of lines drawn by various combinations of LINTYP and INTEQ. The dummy axes shown are for reference only.

A. CALL LINE (XARRAY, YARRAY, 33,1,0,0)

B. CALL LINE (XARRAY,YARRAY,33,1,1,2)
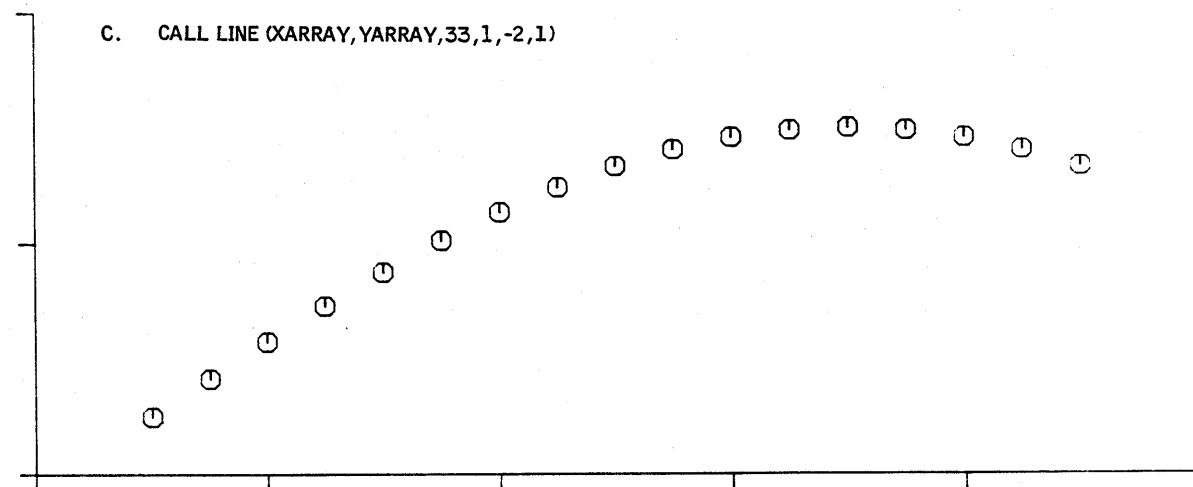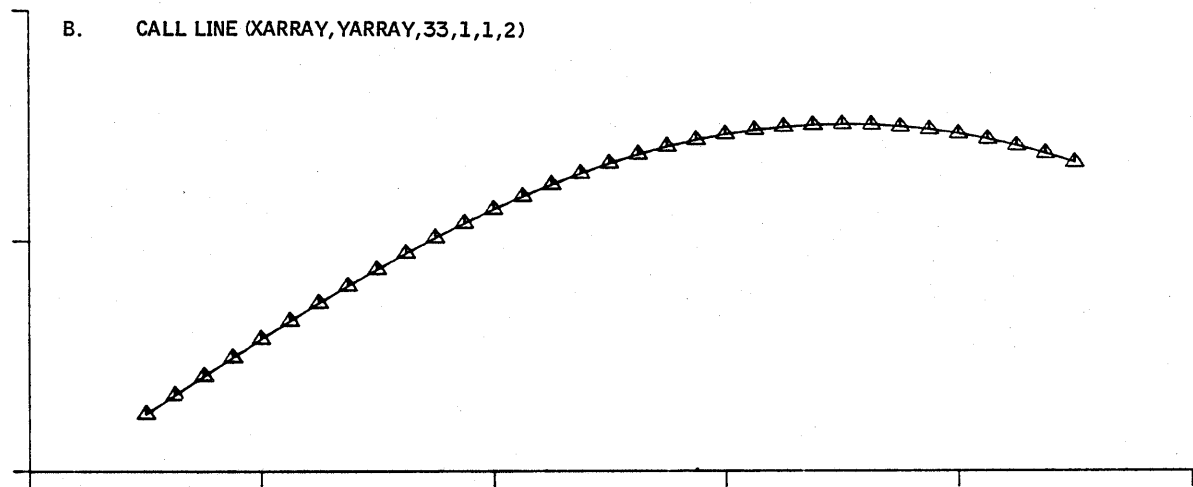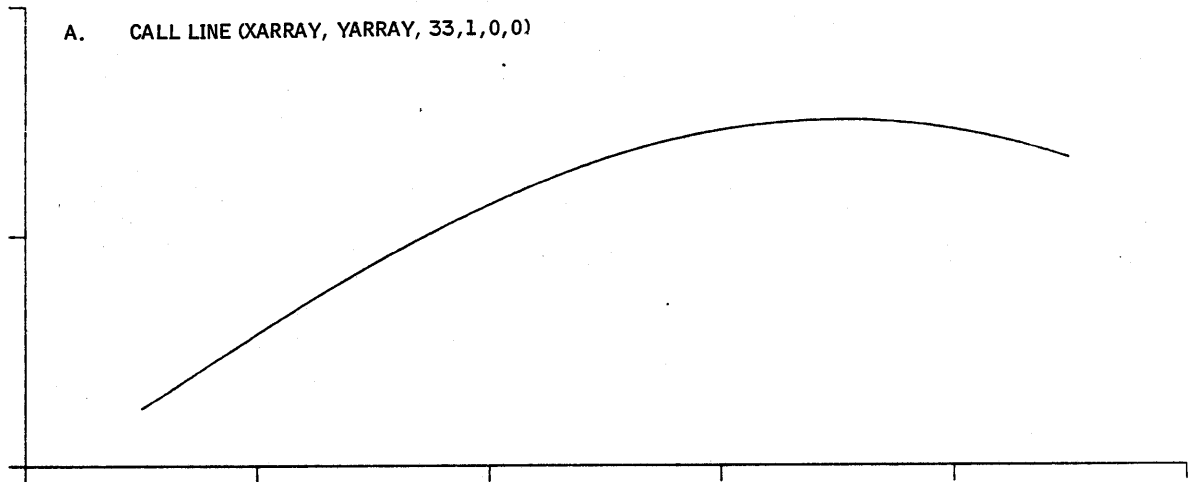
C. CALL LINE (XARRAY,YARRAY,33,1,-2,1)

Figure 7-1. Sample Lines Drawn by LINTYP and INTEQ

24

OPERATING CONSIDERATIONS

An understanding of some of the operating characteristics of CalComp plotting systems is desirable when you are planning your application program. The quality of appearance of the finished plot depends on the combination of pen, paper and ink. The accuracy of the plot can be affected by the initial setting of the pen and the travel of the pen with respect to the limit switches.

LIMIT SWITCHES

The plotting pen carriage is driven by motors. In order to avoid damage to a motor when the pen is given commands to move beyond its maximum travel, limit switches are placed at the sides of the plotter. When a limit switch is actuated, the circuits that impulse the motor to drive in the direction of the switch are disabled. All further commands to move in that direction are ignored until commands of the opposite direction move the pen away from the limit switch. Drum plotters have limit switches at the ends of the Y-axis only; flatbed and beltbed plotters have limit switches in both the X and Y axes.

Plotter operation varies when a limit switch is activated, depending on the plotting system. Plotting may continue with no indication given, or plotting may continue with a warning indication given, or plotting may be halted. In some on-line systems, an interrupt is generated when a limit switch is activated. "Hitting a limit switch" indicates that the plot has lost origin and must be restarted in a position where it will not activate any limit switch. If the plot is too large for the plotter, it must be scaled down. This can be done at the controller in some systems, or it can be done by rerunning the job and calling FACTOR in other systems. System operation manuals for each plotting system give detailed limit switch information.

INITIAL ORIENTATION AND ORIGIN CONTROL

When PLOTS is called, a logic origin is defined by setting the current X, Y coordinates maintained in the PLOT routine to (0, 0). When actual plotting begins, this logical origin corresponds to the current physical position of pen number 1, which is established as the physical origin. Before a plot is started, the operator should move the plotter to a position which is suitable for the origin of the plot. Typically, it is the lower left-hand corner of the area to be occupied by the plot, but it will depend upon the way the plot was programmed.

If gridded paper or special forms are being used, the operator must align the form properly before starting to plot. This can be done by replacing pen 1 with the crosshair reticle provided in the pen kit and using the manual pushbuttons on the plotter to position the reticle over the desired point. The pen is then replaced so that the plot can start from that position.

PLOT TIME OPTIONS

Several plot-time options are available for CalComp's programmable controllers used in off-line systems. These options allow the plot data to be transformed or even modified before plotting. These features are generally not included with a plotting system, but are available as extra-cost options.

- **Scaling**   The scaling feature allows the operator to scale the size of the entire plot up or down before plotting. On some systems, the amount of scaling is limited (for example, to 1/8, 1/4, 1/2, 1, 2, 4 or 8 times normal sizes). For other systems, any scale may be specified.

- **Windowing**   This feature allows from one to four separate rectangular "windows" of a plot to be defined. Only the part of the plot inside the defined area (inclusive window) or outside the defined area (exclusive window) is subsequently plotted. This feature also includes the general scaling option.

- **CAL-EDIT**   CAL-EDIT is a special software package which allows the operator much flexibility in plotting. It requires a special HCBS package and uses a sentence-structured, absolute-coordinate format. Using a teletype attached to the controller, the operator can specify independent X or Y scale factors, rotation, mirror imaging, or any combination of these. He can also specify an inclusive or exclusive window; plot or replot any part of the drawing (forwards or backwards); edit data prior to plotting; and plot straight lines (solid or dashed), curves, and circles by direct input.

SEARCH ADDRESSES

Search addresses are special codes placed on tapes to be plotted off-line. These codes contain sequential numbers starting with 1. Search Address Numbers are used to separate individual plots produced by a single job, to provide breakpoints in long plotting jobs, and for programmer convenience.

Search address 1 is placed on the plot tape when PLOTS is called. Thereafter, each call to PLOT with a -IPEN value causes the previous plot data to be written and a new tape record is started with a search address code containing the next sequential number. Finally, when PLOT is called with IPEN = 999, a search address of 9999 is written and the output file is closed. The plot data records on the tape between search address N and N + 1 constitute "search address N", "block address N", or "plot number N".

Most CalComp controllers allow the operator to specify which search addresses to plot and whether to plot them one at a time or continuously. This is accomplished by a set of thumbwheel switches and pushbuttons for SEARCH, SINGLE PLOT, and MULTIPLE PLOT on the controllers. The operator searches to the search address number (usually 1) at which he wishes to begin plotting, then sets the thumbwheel switches to the search address at which plotting is to stop (usually 9999), and presses either the SINGLE PLOT or MULTIPLE PLOT pushbutton to being plotting. Most CalComp controllers can handle multireel plot files and multiple plot jobs on a single reel. The operating instructions for each plotting system contain this detailed information.

Newer CalComp controllers allow the operator to enter a single command (i.e., PLOT from 40 to 100) to both search and plot.

SUMMARY OF STANDARD CALLING SEQUENCES

CALL PLOT (XPAGE, YPAGE, ±IPEN)
CALL PLOTS (0, 0, LDEV)
CALL FACTOR (FACT)
CALL WHERE (RXPAGE, RYPAGE, RFACT)
CALL NEWPEN (INP)

XPAGE, YPAGE	are the X, Y coordinates, in inches from the current origin, of a pen movement's terminal position.
IPEN	specifies the pen up/down status during movement (up = 3, down = 2). If negative, establishes a new origin at the new position.
LDEV	is the logical number of the plot output device.
FACT	is a scale factor that determines the enlargement or reduction of the entire plot.
RXPAGE, RYPAGE, RFACT	are the locations that will contain the current values of XPAGE, YPAGE, and FACT after WHERE is called.
INP	is the number of the selected pen.

CALL SYMBOL (XPAGE, YPAGE, HEIGHT, IBCD, ANGLE, ±NCHAR)
CALL SYMBOL (XPAGE, YPAGE, HEIGHT, INTEQ, ANGLE, —ICODE)
CALL NUMBER (XPAGE, YPAGE, HEIGHT, FPN, ANGLE, ±NDEC)

XPAGE, YPAGE	define the relative origin of the character string (usually the lower left corner of the first character position).
HEIGHT	is the height (and width), in inches, of a character position.
IBCD	is the location of a character string.
ANGLE	is the angle at which the character string is to be plotted.
NCHAR	is the number of characters in IBCD.
INTEQ	is the integer equivalent of a special centered plotting symbol.
ICODE	specifies the pen up/down status during movement to the relative origin.
FPN	is the location of a real (floating-point) value.
NDEC	specifies the number of decimal places to be printed.

CALL SCALE (ARRAY, AXLEN, NPTS, ±INC)
CALL AXIS (XPAGE, YPAGE, IBCD, ±NCHAR, AXLEN, ANGLE, FIRSTV, DELTAV)
CALL LINE (XARRAY, YARRAY, NPTS, INC, LINTYP, INTEQ)

ARRAY	names an array of data values.
AXLEN	is the length of the axis line.
NPTS	is the number of entries in an array.
INC	is the increment between entries in an array.
XPAGE, YPAGE	is the relative origin of the axis line.
IBCD	is the location of the alphameric axis title.
NCHAR	is the number of characters in IBCD.
ANGLE	is the angle of the axis line.
FIRSTV	is the first scale value printed along the axis.
DELTAV	is the increment between scale values on the axis.
XARRAY, YARRAY	contain the pairs of data values to be plotted.
LINTYP	specifies the type of line to be drawn through the data points.
INTEQ	is the integer equivalent of a special centered plotting symbol.

27

CALCOMP APPLICATIONS SOFTWARE

THREE-D	Plots three dimensional and perspective views of a function of two independent variables. Will automatically grid and smooth data, draw and annotate axes.
GPCP	A General Purpose Contouring Program to plot contour maps under control of a variety of options.
FLOWGEN	A program which automatically produces and plots a flowchart of any ANS FORTRAN IV program directly from the source deck.
AUTONET	Three versions of AUTONET which enable users to plot the output from various levels of network scheduling programs e.g., CPM, PERT, PMS, and PCS.
SAMPS	Creates subdivision maps complete with bearings and distances and calculates lot areas.
DATAGRAPH	Produces management information charts and graphs directly from user data on cards, tape or disc.
PROVE	A program which produces verification plots of APT/ADAPT cutter location files for numerically controlled (NC) machine tools.
CONTOUR	A basic contouring package for small machines not holding or needing the flexibility of GPCP.
AUTOGANT	An Automatic Gantt Bar Chart Display Program useful in conjunction with AUTONET.

CALCOMP FUNCTIONAL SOFTWARE

HOST COMPUTER RESIDENT FUNCTIONAL SOFTWARE

General Category

CIRCLE	Draws a circle or spiral.
DASHL	Draws dashed lines connecting a series of data points.
DASHP	Draws a dashed line to a specified point.
ELIPS	Draws an ellipse or elliptical arc.
FIT	Draws a curve through three points.
GRID	Draws a linear grid.
POLY	Draws an equilateral polygon.
RECT	Draws a rectangle.

Drafting Category

AROHD	Draws various types of arrowheads.
ARROW	Draws a line terminated with an arrowhead through a set of data points.
CNTRL	Draws a "center line" through a set of data points.
DIMEN	Draws annotated dimension lines with extension lines.
LABEL	Draws annotation centered between two points with control over symbol placement.

Business Category

AXISB	Draws an axis with "business" annotation.
AXISC	Draws an axis with calendar month annotation.
BAR	Draws bars, for bar graph plotting.
LBAXS	Draws a logarithmic axis with business annotation.
LGLIN	Plots data in either log-log or semi-log mode.
SHADE	Draws shading between lines.
SCALG	Performs scaling for logarithmic plotting.

Scientific Category

CURVX	Plots a function of X over a given range.
CURVY	Plots a function of Y over a given range.
FLINE	Draws a smooth curve through a set of data points.
LGAXS	Draws a logarithmic axis with annotation.
LGLIN	Plots data in either log-log or semi-log mode.
POLAR	Plots data points using polar coordinates.
SCALG	Performs scaling for logarithmic plotting.
SMOOT	Draws a smooth curve through sequential data points.

Miscellaneous Category

CVPLT	Polynomial Curve Fitting Routine, *accompanied by a driver to allow use as a complete program*, which determines and plots a polynomial of given degree which best fits the data points. The least squares approximation technique is used. Data points also are plotted.
FLOCT	Flowchart production program which plots and annotates a flowchart defined by input data cards.
FORGN	Tape and Card Forms Generator *which draws the blank form with headings, for tape formats or card forms*, as defined by input data cards.

PAINT	Polygon painting software for photo plotter.
SPECIAL SYMBOL SETS	Drafting Letter Set, Block Letter Symbol Set, Upper Case/Lower Case Symbol Set.
WORLD MAP	A set of subroutines used to plot and annotate various map projects.

GRAPHIC CONTROLLER RESIDENT FUNCTIONAL SOFTWARE

MMS	Microfiche Management Software produces titled and indexed microfiche directly from host computer standard print tape.
CAL-EDIT	Provides enhanced operator control at plot time: scaling, rotation, mirror image, windowing and editing.
WINDOWING FEATURE	Provides from 1 to 4 inclusive or exclusive windows and pen control for a standard plot tape.
PRINTISM	Print standard host computer print tape on a CalComp COM system.
GRAPHISM	Simulate SC4020 or SC4060, or SC4360 on CalComp COM systems.
SIMULATORS	Various programs to plot tapes generated for CalComp nonprogrammable controllers.
SCALING	Provides single-valued scaling or independent X and Y axis scaling along with rotation and mirroring.
SIZING	Provides automatic scaling and translation of data to fit on preprinted forms.
CIRCLE	Provides for quick generation of circles or circular arcs.