ccm-70-13

TEKPLOT/CALTEK

by

G. Rose

December, 1970

OSU

COMPUTER CENTER

Oregon State University
Corvallis, Oregon 97331

TEKPLOT/CALTEK

A Subroutine Package for the Tektronix T-4002
Graphics Display Terminal

and

A Companion Subroutine Package for the
Calcomp Plotter

ccm-70-13

by
G. Rose

Computer Center
Oregon State University
Corvallis, Oregon 97331

December 1970

Printed by the Oregon State University
Department of Printing

TEKPLOT/CALTEK

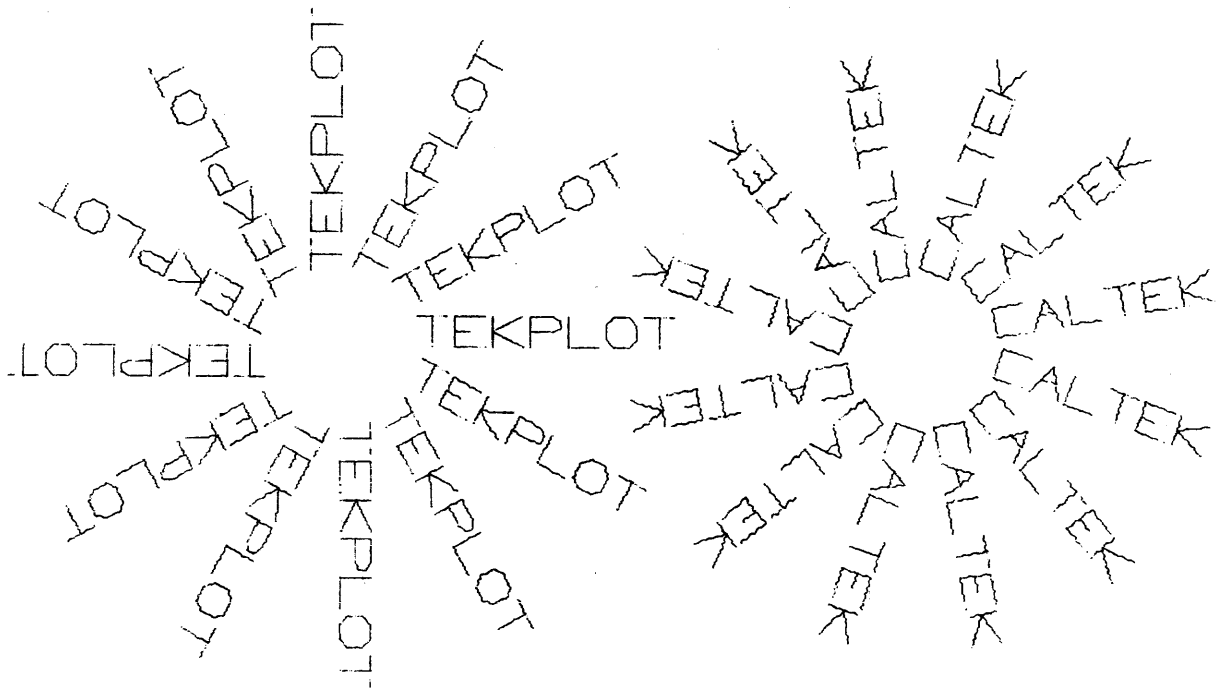


TABLE OF CONTENTS

Page

I. Introduction 1

II. Operation of the Tektronix T-4002 2

III. Use of TEKPLOT and CALTEK 5

IV. Description of TEKPLOT/CALTEK Subroutines 7
4.1 Utility routines 9
1) ERASE 9
2) PAGE 9
3) TEKPAUSE 9
4) BYENOW 10
5) PLOTLUN (J) 10
6) DOUBLE 10
7) ITALICS 11
8) NORMAL 11
4.2 Mode routines 11
9) ALPHAS 12
10) VECTORS 12
11) POINTS 13
4.3 Axis manipulation routines 13
12) SCALE (KXFACT, KYFACT, KXORG, KYORG) 13
13) FSCALE (XFACT, YFACT, XORG, YORG) 14
14) SIZE (XMAG, YMAG) 14
15) ROTATE (DEG) 15
16) AXIS (XLOW, XLNG, YLOW, YLNG, XTIC, YTIC) 15
4.4 Character I/O routines 17
17) KHOUT (J) 18
18) KHIN (N) 18
4.5 Plotting routines 18
19) PLOT (X, Y, IPEN, MARK) 19
20) IPLOT (JX, JY, IPEN, MARK) 20
21) DELTA (DELX, DELY, IPEN, JRETURN) 20
22) IDELTA (JDELX, JDELY, IPEN, JRETURN) 21
23) INCPLOT (JDIR, IPEN, NUM, KRETURN) 21
4.6 Graphics Input Routines 23
24) GRIN (X, Y, KHAR) 24
25) IGRIN (JX, JY, KHAR) 25
26) GRIDSET (J, K, IPEN) 26
27) GRID (JROW, KCOL, KHAR) 27
4.7 Software Character Generation 30
28) WRITEX (STRING1, STRING2,...,STRINGJ) 30
29) WRITEY (FACT, DEG, STRING1,...,STRINGJ) 31
30) XLATE (ICON, BCDCON) 33

V. Selected Examples 34

Appendix I - USASCII Control Characters 41

Appendix II - Summary of Routines 42

I. INTRODUCTION

The development of the direct view, bi-stable cathode ray storage tube has ushered in a family of new computer terminals. Characterized by high resolution, high speed, and comparatively low cost graphics, such terminals are significantly affecting the field of computer graphics.

The Tektronix T-4002 graphics display terminal utilizes a direct view storage tube. The terminal permits both graphics input and graphics output in addition to usual alphanumeric operation.

Section II of this manual includes a brief description of the functional characteristics of the T-4002. The remainder of the manual describes two FORTRAN compatible subroutine packages, TEKPLOT and CALTEK. These subroutine packages are tailored to the operation of the T-4002, and they are intended to provide a basis for graphics applications written in the FORTRAN language.

II. OPERATION OF THE TEKTRONIX T-4002

In this section some of the salient functional characteristics of the T-4002 are briefly discussed. No attempt is made here to cover the physical operation of the unit in detail. For additional information, the reader is referred to the T4002/4802 Graphic Computer Terminal User's Manual published by Tektronix, Inc.

The computer terminal provides information exchange between the user and the computer through a keyboard and a visual display. The keyboard includes the full USASCII character set with control-shifted characters isolated into a separate keyboard cluster. A table that lists the T-4002 control keys and their teletypewriter control-shifted equivalents is given in Appendix I.

The visual display is 8.25 inches by 6.5 inches. In alphanumeric mode, the display has 39 lines of 85 symbols each; in graphics mode the display has 1024 by 762 addressable points. A flickering cursor indicates the user's current location while in alphanumeric mode. This cursor is not visible during graphics output.

In addition to the keyboard, the terminal has two rows of control indicators. The setting of pertinent indicators is described here:

1) Power Off/On - this key should be left in the on position, even when the terminal is not in use.
2) On-line/local - must be on-line for communication with the operating system.
3) ASCII/TTY - normally set to TTY for OS-3 operation. If full ASCII is designated, the same conventions that hold for a model 37 Teletype obtain.

4) Keyboard/Aux - set to keyboard for graphics output only; set to both keyboard and Aux for graphics input and output.

5) Display/Aux - set to Display.

6) Page Full - an attempt to enter data beyond the last line of the display causes the T-4002 to send a break to the computer and lights the Page Full indicator. Operation may be resumed by depressing this indicator; at this time the terminal will be forced into alphanumeric mode with the alpha cursor positioned in the upper left corner of the screen.

7) Margin Shift - depressing Margin Shift changes the effective left-hand margin to the next of four equally spaced margin positions.

8) Comm/Error - a status light for parity error detection. As OS-3 does not keep parity, this indicator may be ignored.

9) Data Received - lights when a character is transmitted to the terminal. Depressing this indicator resets it.

10) Interrupt - the equivalent of the break key on a Teletype.

11) Direct/Compose - Direct causes each character to be sent to the computer as it is typed. Compose causes each character to be sent to a one-line scratch pad memory where it is displayed on the refreshed, non-storing area at the base of the display screen. From this display, the terminal user can edit one complete line before sending it to the computer.

12) Full/Clear - lights when the scratch pad memory is full. Depressing this control clears the scratch pad.

13) Send - depressing Send while the Compose indicator is selected causes the scratch pad memory to be transmitted to the computer. The scratch pad is not cleared by this operation.

14) Home - the Home control places the terminal in alphanumeric mode with the alpha cursor positioned to the upper left-hand corner of the screen (home position). The directional controls, two on either side of Home, are used to locally position the alpha cursor.

15) View/Hold - if the T-4002 is inactive for 90 seconds, the Hold indicator lights and the image on the visual display dims. Depressing View or any keyboard character causes the screen to revert to proper viewing intensity.

16) Erase - allows manual erasure of the screen.

In addition to graphics output, the T-4002 also allows graphics input. In this mode of operation, a full screen crosshair cursor appears. The cursor may be positioned by manipulating the joystick adjacent to the terminal. When the cursor is situated at the desired location, its coordinates are transmitted to the computer by depressing any key. Additional information about graphics input is included in the section on graphics input subroutines.

III. USE OF TEKPLOT AND CALTEK

The TEKPLOT and CALTEK subroutine packages are sets of FORTRAN compatible subroutines that are tailored to the operation of the T-4002. TEKPLOT is used for plotting directly on the terminal while CALTEK plots on the Calcomp plotter. Both packages have identical calls.

The TEKPLOT package is stored in a public file named *TEKPLOT; CALTEK is stored in *CALTEK. To use these routines in a program, they must be loaded with the FORTRAN binary deck. For example, if a user source program is stored in a file named YOURPROG, the the following statements would serve to compile and load the program together with the TEKPLOT graphics routines:

```
#FORTRAN,I=YOURPROG,X
#LOAD,56,*TEKPLOT
RUN
```

An efficient way of working is to debug your program at the terminal using *TEKPLOT. When the program is debugged, a hard copy may be obtained by loading the program with *CALTEK. In this latter case, any plotting that was previously done on the terminal will now be output to the Calcomp plotter. Continuing the preceding example, after program debugging the FORTRAN binary deck is presumably still stored on logical unit 56. To obtain a hard copy of the plot:

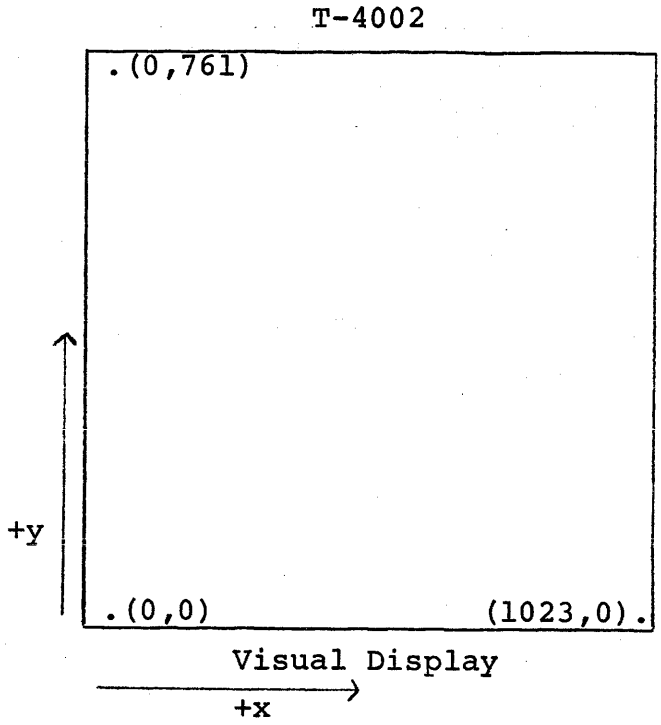
```
#LOAD,56,*CALTEK
RUN
```

This method of working allows rapid debugging of plotted output at the terminal and assures that Calcomp plotted output need be run only once. Of course, there is no inherent reason why the user must output his plot to the Calcomp plotter at all; it is merely a convenient way to obtain a hard copy of terminal output.

In the event that the user wishes to load both the TEKPLOT and the CALTEK routines simultaneously, a variant edition of the latter is stored in *CALTEKV. CALTEK and CALTEKV are identical with one exception: all subroutine names in CALTEKV have an appended V as the last character, unless the name was already eight characters long. In this case, the V suffix replaces the final character.

IV. DESCRIPTION OF TEKPLOT/CALTEK SUBROUTINES

Points on the terminal range from 0-1023 for X and 0-761 for Y. Point (0,0) is located at the bottom left of the screen.



In the following description, the term tekpoint refers to an (x,y) point in this range. The term scaled point refers to an (x,y) point that will be scaled, translated, and rotated prior to plotting. Scale factors and axis mappings are established by the user's program with the scaling and axis manipulation routines.

Vectors drawn on the screen are straight line segments that connect the current position to the (x,y) point specified as a parameter in any of the plotting routines. This latter (x,y) point then becomes the new current position.

It is necessary to stress that while tekpoint (0,0) is located at the lower left corner of the screen, the "home" position in alphanumeric mode is situated at the top left corner. This double convention allows the screen to be viewed as a page of textual information in alphanumeric mode, whereas in graphics mode the positive directions of the axes conform to the usual conventions for the first quadrant. (It will be seen in the ensuing description that the user is free to establish new axis conventions as he wishes.) In keeping with this double convention, the TEKPLOT routines distinguish between plotting and printing in the following way: the current position in graphics mode is not affected by any intervening printing that occurs in alphanumeric mode.

If the preceding conventions are not at once clear, a few moments experimentation with the terminal will make them seem obvious and natural.

Output directed to the Calcomp plotter is drawn within the confines of an implicit screen that bears the same limits as the visual display of the T-4002. That is, Calcomp output is drawn in a plane Cartesian point lattice 1024x762 points. However, since resolvable points on the T-4002 are approximately .008 inches apart while resolvable points on the Calcomp plotter are .01 inches apart, the Calcomp copy will be slightly larger. Moreover, it is possible to magnify the implicit screen size of Calcomp output using the subroutine SIZE. In this way, the user can take advantage of the increased resolution available on the Calcomp plotter.

The individual subroutines are described in the following paragraphs.

4.1 Utility routines

The utility routines are used to manipulate the environment in which plotting will take place. They include:

(1) ERASE

TEKPLOT	CALTEK
The ERASE subroutine erases the screen, places the terminal in alphanumeric mode, and positions the alpha cursor to the home position. ERASE does not affect the current graphics position.	The ERASE subroutine positions to the next implicit screen on the Calcomp plotter if any plotting has occurred since the last call to ERASE. In either case the pen is positioned to the home position. The current graphics position is not affected.

(2) PAGE

TEKPLOT	CALTEK
PAGE places the terminal in alphanumeric mode at the home position. The current graphics position is not affected.	PAGE positions the pen to the home position on the current implicit screen. The current graphics position is not affected.

(3) TEKPAUSE

TEKPLOT	CALTEK
The TEKPAUSE routine is a convenient way for a user to I/O bound himself in a time-sharing environment. TEKPAUSE places the terminal in alphanumeric mode at home position, then reads one character from the terminal, discards that character, and exits. The routine is intended for a	TEKPAUSE positions the pen to home position. The current graphics position is not affected.

user who wishes to pause at some point in his program in order to inspect his plotted output, and then resume execution by typing any character. TEKPAUSE does not affect the current graphics position.

(4) BYENOW

TEKPLOT	CALTEK
BYENOW may be used as the last executable statement in the user's FORTRAN program. Calling BYENOW suppresses the FORTRAN message END OF FORTRAN EXECUTION and returns the user to control mode.	BYENOW must be used as the last executable statement in the user's FORTRAN program. Calling BYENOW empties the user's plot buffer, positions the plotter pen out of harm's way, suppresses the message END OF FORTRAN EXECUTION and returns the user to control mode.

(5) PLOTLUN (J)

TEKPLOT	CALTEK
PLOTLUN is a no-op in TEKPLOT. That is, calling PLOTLUN has no effect whatsoever.	PLOTLUN establishes the logical unit to which plotter output will be directed. CALTEK defaults to logical unit 1. It is assumed that the user will equip and label his plot lun prior to loading *CALTEK. When called, PLOTLUN empties the current plot buffer and lifts the pen prior to assigning the new logical unit, if any plotting has occurred on the previous lun.

(6) DOUBLE

TEKPLOT	CALTEK
After calling DOUBLE, all subsequent plotting will be double intensity (i.e., broader vectors and points) and alphanumeric characters will be twice as large.	DOUBLE is a no-op for CALTEK.

While a single call to DOUBLE will suffice for all further plotting, the carriage return that occurs after each FORTRAN WRITE or PRINT statement resets this operational mode of the T-4002. For this reason, a call to DOUBLE must precede each WRITE statement in order to achieve double character size. It should be noted that when double size characters are printed on the T-4002, a space must be inserted between every character to prevent overlap.

(7) ITALICS

TEKPLOT	CALTEK
After calling ITALICS, all further alphanumeric characters will be italicized.	ITALICS is a no-op for CALTEK.

(8) NORMAL

TEKPLOT	CALTEK
Calling NORMAL resets the mode in which double intensity graphics and double size and/or italicized characters occur.	NORMAL is a no-op for CALTEK.

4.2 Mode routines

The mode routines are used to determine the mode in which all subsequent printing or plotting will take place. Five physical modes are available in the T-4002. These include:

a) alphanumeric
b) vector plot
c) point plot
d) incremental plot
e) graphics input

The TEKPLOT/CALTEK mode routines govern only the first three of these modes. Incremental plot mode is dealt with as a special case by the subroutine INCPLOT, and graphics input mode is treated separately by the graphics input routines.

The T-4002 functions as a Teletype in alphanumeric mode. The terminal is automatically placed in this mode when it is turned on, and a carriage return, an SOH, or the Home button will all reset to this mode. When in alphanumeric mode, the current character position is signaled by the location of the flickering cursor.

Vector mode is used to draw a smooth line between the current graphics position and the (x,y) point specified as a parameter to any of the plotting routines. This latter point then becomes the new current position.

Point plot mode causes a point to appear at the (x,y) point specified as a parameter in a call to any of the plotting routines. This (x,y) point then becomes the new current position.

(9) ALPHAS

TEKPLOT	CALTEK
Calling ALPHAS places the terminal in alphanumeric mode. The alpha cursor will appear at the current (x,y) position.	Calling ALPHAS has no visible effect in CALTEK. However, some internal status manipulation does take place.

(10) VECTORS

Calling VECTORS places the T-4002 or Calcomp in vector mode and positions to the current graphics location. All subsequent plotting in this mode will be in the form of smooth lines connecting each specified point.

(11) POINTS

Calling POINTS places the terminal or Calcomp in point plot mode and positions to the current graphics location. All subsequent plotting in this mode will be in the form of dots that are drawn at each specified point.

4.3 Axis manipulation routines

The axis manipulation routines are used to accomplish scaling, axis translation, axis transformation, and to draw a visible axis.

(12) SCALE (KXFACT, KYFACT, KXORG, KYORG)

A call to SCALE determines scale factors and establishes an origin for any subsequent plotting. If SCALE is not called, then the indicated default values are assumed.

Parameters are:

KXFACT - the number of tekpoints per scaled point on the x-axis. The sign of KXFACT determines the direction of the x-axis, i.e., if KXFACT is negative, the positive direction of the x-axis is to the left. Default value=1.

KYFACT - the number of tekpoints per scaled point on the y-axis. The sign of KYFACT determines the direction of the y-axis. Default value=1.

KXORG - the origin on the x-axis expressed in tekpoints. Default value=0.

KYORG - the origin on the y-axis expressed in tekpoints. Default value=0.

SCALE may be called whether or not an explicit axis is drawn. KXFACT and KYFACT act as multiplicative factors that determine the conversion from tekpoints to scaled points. KXORG and KYORG act as linear displacements from tekpoint (0,0).

For example, the physical origin of the T-4002 is tekpoint (0,0) at the lower left corner of the screen. To establish a new origin at the center of the screen, the user may

```
CALL SCALE (1,1,512,380)
```

This new origin then obtains for all subsequent plotting or until the next call to SCALE. Further examples are included in Section V.

(13) FSCALE (XFACT, YFACT, XORG, YORG)

FSCALE is identical to SCALE with floating-point arguments.

(14) SIZE (XMAG, YMAG)

TEKPLOT

SIZE is no-op for TEKPLOT

CALTEK

SIZE may be called to establish a ratio between the size of the T-4002 visual display and the size of the implicit screen for Calcomp output. Parameters are:

- XMAG - a magnification factor,>0, that establishes the number of T-4002 screen widths per implicit screen width of Calcomp output. Default value=1.
- YMAG - a magnification factor,>0, that establishes the number of T-4002 screen lengths per implicit screen length of Calcomp output. Default value=1.

For example,

```
CALL SIZE (2.,2.)
```

would yield Calcomp plotted output twice the size of the T-4002 screen.

While the magnitude of YMAG is limited only by the length of a roll of plotter paper, XMAG will be constrained by CALTEK so as not to exceed 2.75 due to the physical width of the plotter bed.

The user can always allow the width of his plot to grow very large, while constraining the length, by rotating the plot ninety degrees.

SIZE may be thought of as a special kind of scaling routine. It scales the current scaling factors, origin, and screen limits. Customarily, a call to SIZE would precede any actual plotting in the user's program.

(15) ROTATE (DEG)

By calling ROTATE, the user declares himself to be working in a new transformed axis that is rotated the specified number of degrees with respect to the Cartesian system of the screen. The center of rotation is the user's origin, scaled point (0,0).

Parameters are:

DEG - the number of degrees (not radians) of rotation from the orthogonal system of the screen. Default value=0. Positive rotation is in a counterclockwise direction.

After calling ROTATE, all further plotting will occur in the new transformed coordinate system until the next call to ROTATE.

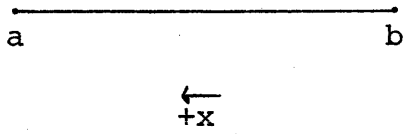
(16) AXIS (XLOW, XLNG, YLOW, YLNG, XTIC, YTIC)

The AXIS routine is used to draw X and Y axes together with tic marks if desired. After a call to AXIS, the user will be left in vector mode, positioned at his origin.

Parameters are:

XLOW - the starting position on the X-axis, expressed in scaled, rotated points. XLOW must be the smallest X value that the axis line will assume. If, for

example, an X-axis is to be drawn from point a to point b on the figure below,



and if the user has reversed the positive direction of the axis (by calling SCALE with negative JXFACT), then b in the figure is the starting position.

XLNG - the total length of the X-axis, expressed in scaled points.

YLOW - the starting position on the Y-axis, expressed in scaled, rotated points.

YLNG - the total length of the Y-axis, expressed in scaled points.

XTIC - the distance between tic marks on the X-axis, expressed in scaled points. Set XTIC=0.0 if no tic marks are desired.

YTIC - the distance between tic marks on the Y-axis, expressed in scaled points. Set YTIC=0.0 if no tic marks are desired.

AXIS converts the user's starting points, XLOW and YLOW, to their equivalent in tekpoints to determine the correct starting values. XLNG and YLNG are converted to tekpoints and added to the starting points to determine the correct terminating values. If the user has called ROTATE, then the AXIS routine will draw an appropriately rotated axis.

While it is usually the case that the axis lines pass through the origin, AXIS does not require this to happen. Each axis line will be drawn only the specified length from the indicated starting position.

Example: The user wishes to establish an origin at the center of the screen. He wishes to draw axes that extend the length of the screen in either direction. Tic marks are to be 20 tekpoints apart on the X-axis and 40 tekpoints apart on the Y-axis. The following statements would suffice:

```
CALL SCALE (10,10,512,380)
CALL AXIS (-51.,102.,-38.,76.,2.,4.)
```

4.4 Character I/O routines

The character I/O routines are not intended for use by the typical applications programmer. They are provided because the ASCII control characters needed to drive the T-4002 are not included in the FORTRAN character set and, hence, may not be generated using WRITE statements.

Since the TEKPLOT package does provide the necessary control characters as required, the user need never concern himself with details of this type.

It should be noted that if a user elects to effect his own control of the terminal using character I/O, then the TEKPLOT routines cannot keep accurate track of current terminal status. For this reason, the user is advised to call ALPHAS and then reinitialize his current graphics position after character I/O has occurred.

(17) KHOUT (J)

KHOUT is a character output routine.

Parameters are:

J - the ASCII value of a character to be output to the T-4002.

For example,

```
CALL KHOUT (207B)
```

will result in a 'ding'.

(18) KHIN (N)

KHIN is a character input routine. Unlike the other routines in TEKPLOT/CALTEK, it is written as a function.

Parameters are:

N - the number of spaces to be output prior to reading the character.

For example,

```
JJ=KHIN(10)
```

will output 10 spaces and then read a character from the T-4002. The ASCII value of that character will be assigned to JJ.

4.5 Plotting routines

The plotting routines are used to plot data in the currently selected mode (vectors or points); the mode must be set prior to calling a plotting routine. Thus, a call to PLOT while in vector mode will cause a line to be drawn from the current graphics position to the (x,y) point specified as a parameter to PLOT. This latter (x,y) then becomes the new current position. In a

similar way, a call to PLOT while in point plot mode causes a point to be drawn at the (x,y) position specified in the call; this (x,y) point then becomes the new current position. (INCPLOT is an exception to the above rules, as will be seen.)

Two types of plotting routines are provided. PLOT and IPLOT take as parameters an absolute (x,y) point in the user's established coordinate system. DELTA and IDELTA take as parameters a (Δx,Δy) relative to the current position. Both PLOT and DELTA commence at the current graphics position, and both update the current position after plotting has occurred.

(19) PLOT (X, Y, IPEN, MARK)

The PLOT subroutine is used to plot to an absolute coordinate position in the current mode.

Parameters are:

X - an X value, expressed in scaled, rotated points.

Y - a Y value, expressed in scaled, rotated points.

IPEN - determines whether the pen is up or down (Z-axis off or on). If IPEN=0, then the pen is up. If IPEN$\neq$0, then the pen is down.

MARK - the number of a data mark taken from the following list:

0 - no data mark
1 - small x
2 - small down arrow
3 - small up arrow
4 - small square
5 - small triangle
6 - small asterisk
11 - large x
12 - large down arrow
13 - large up arrow
14 - large square
15 - large triangle
16 - large asterisk

Any other number other than one of those from the above list is equivalent to no data mark.

The PLOT routine converts the parameters X and Y to tekpoints and rotates them appropriately prior to plotting. If PLOT has been called with an X or Y that, after scaling and rotation, turns out to be outside the screen limits, then the routine contrains plotting to the nearest point that is still on screen.

Calling PLOT while in alpha mode is a legal operation, providing that the pen is up. The effect is to position the pen and then return the user to alpha mode. However, calling PLOT while in alpha mode with the pen down will yield an undesirable result.

After calling PLOT, the current graphics position is appropriately updated.

(20) IPLOT (JX, JY, IPEN, MARK)

Identical to PLOT with integer arguments for X and Y.

(21) DELTA (DELX, DELY, IPEN, JRETURN)

The DELTA routine is used for local plotting, relative to the current graphics position. Plotting takes place in the current mode.

Parameters are:

DELX - a quantity, ΔX, relative to the current X position, expressed in scaled, rotated points.

DELY - a quantity, ΔY, relative to the current Y position, expressed in scaled, rotated points.

IPEN - a pen value that governs whether the pen is up or down. If IPEN=0, then the pen is up. If IPEN$\neq$0, then the pen is down.

JRETURN - determines whether DELTA will return to the original graphics position after plotting an

increment of (Δx,Δy). If JRETURN=0, then DELTA does not plot back. If JRETURN≠0, then DELTA does plot back again.

DELTA converts the parameters DELX and DELY to tekpoints and adds them to the current graphics position. DELTA then plots to this position in the current mode. If JRETURN≠0, then DELTA plots back to the starting point again.

The same conventions that obtain for PLOT also apply to DELTA, viz. off screen plotting is constrained to the nearest point still on screen; calling DELTA with pen down while in alpha mode will produce wierd effects; and after calling DELTA the current graphics position is appropriately updated.

(22) IDELTA (JDELX, JDELY, IPEN, JRETURN)

Identical to DELTA with integer arguments for DELX and DELY.

NOTE: The manufacturer's recommended maximum vector length for the T-4002 is two inches. However, the user will find that good results are possible with longer vectors when using the preceding plotting routines. In general, full screen vectors can be used except for a vector that immediately precedes a change in mode. In this instance, the final vector may not be completely drawn if it is quite long. A way to circumvent this problem and still draw long vectors is to plot the last vector twice.

(23) INCPLOT (JDIR, IPEN, NUM, KRETURN)

INCPLOT is an incremental plotting routine that utilizes the incremental plot mode of the terminal (see Section 4.2). It is treated as a special case with respect to the user's current mode. In particular, the user may call INCPLOT while in any mode; the routine will perform incremental plotting as

specified and then return the user to his current mode.

Parameters are:

JDIR - a Hollerith variable or constant that specifies one of eight possible directions for the plotted increment. Directions are N, NE, E, SE, S, SW, W, or NW. If any other Hollerith string is supplied apart from those listed, INCPLOT exits without plotting.

IPEN - determines whether the pen is up or down. If IPEN=0, then the pen is up. If IPEN≠0, then the pen is down.

NUM - the number of increments to be plotted in the specified direction.

KRETURN - determines whether INCPLOT will plot back to the original position after the increment has been plotted. If KRETURN=0, then INCPLOT will not plot back again. If KRETURN≠0, then INCPLOT does plot back.

INCPLOT does not update the current graphics position. The routine will commence incremental plotting at the current alpha position if the user is in alpha mode, or at the current graphics position if the user is in vector or point plot mode. In the latter case, the user is returned to his current graphics position after incremental plotting has been completed.

Unlike the other routines, INCPLOT does not constrain off-screen plotting to the closest point that is still on screen. Off-screen plotting on the T-4002 will "wrap around" to the other side; off-screen plotting on the Calcomp will shift the user's implicit screen a distance equal to the number of increments plotted beyond the current screen margin. In either case, after off-screen plotting has occurred, the user's position will be consistent with the previous conventions, relative to his current screen.

Example: The user wishes to write the line

CASE 1: X^2+C

at the top of his plot, and then to plot a curve of some sort. The program might begin as follows:

```
      PROGRAM SUPRSCPT
      CALL ERASE
      WRITE(61,100)
100   FORMAT('*CASE 1:X +C')
      CALL INCPLOT(1HN,0,7,0)
C     The preceding statement positions the alpha
C     cursor to write a superscript.
      WRITE(61,200)
200   FORMAT(8X,'2')
      Call vectors
          .
          .
          .
```

4.6 Graphics Input Routines

The graphics input routines enable the user to input graphic information from the T-4002 to his program. When a graphic input routine is called, a full screen crosshair cursor appears on the screen. This cursor may be positioned using the joy-stick adjacent to the terminal. After the cursor is positioned, the user causes position coordinates to be passed back to the graphics input routine by striking any key on the keyboard. When the coordinates have been sent to the computer, the graphics cursor disappears.

Since any key will cause position coordinates of the graphics cursor to be transmitted, the keys may be thought of as special function buttons for graphics input mode. The user might choose one of several subroutines depending upon which key was depressed.

It should be noted that since OS-3 is a half-duplex system, an echo will occur for most of the keys. This echo results in a character printed on the screen at the location of the graphics cursor. Characters that do not cause any printing include space, delete, bell, etc. Carriage return, US, and SOH should be avoided since they reset graphics input mode for the T-4002 and position coordinates will be suppressed.

(24) GRIN (X, Y, KHAR)

TEKPLOT

GRIN is the basic graphics input routine. When called, GRIN causes the graphics cursor to appear. The user then positions the cursor and depresses a key.

Parameters are:

X - the location of the x-axis corresponding to the cursor position, converted to scaled, rotated points.

Y - the location on the y-axis corresponding to the cursor position, converted to scaled, rotated points.

KHAR - an eight-bit ASCII character code corresponding to the key that was depressed.

GRIN reads the coordinate position of the cursor and converts it to a scaled point (x,y). These scaled coordinates are then assigned to the user's X and Y parameter locations. The ASCII value of the key that was depressed is assigned to parameter location KHAR.

CALTEK

Since the Calcomp does not allow graphic input, GRIN is a no-op for CALTEK.

GRIN may be called while the user is in any mode. After cursor coordinates have been input, the routine will return the user to his current mode.

A note about the operation of the joystick is in order. When properly aligned, the joystick confines the graphic cursor to a tether coincident with the margins of the screen. In practice, alignment problems do occur. Slight misalignment can affect the user by allowing the crosshair to be positioned off screen. In this event, the T-4002 will not digitize the crosshair coordinates. Should this happen, merely move the joystick to bring the crosshair back onto the screen.

(25) IGRIN (JX, JY, KHAR)

TEKPLOT

Identical to GRIN with integer arguments.

CALTEK

IGRIN is a no-op for CALTEK.

(26) GRIDSET (J, K, IPEN)

TEKPLOT

GRIDSET is intended for use with the GRID routine. Calling GRIDSET partitions the screen into a grid comprised of J*K rectangles.

Parameters are:

J - the number of regions along the x-axis. J≥0.

K - the number of regions along the y-axis. K≥0.

IPEN - if IPEN=0, then GRIDSET values will be retained, but no explicit grid will be drawn. If IPEN ≠0, then GRIDSET values will be retained and an explicit J*K grid will be drawn on the screen.

It is often convenient to draw an explicit grid while debugging a program. When the program is debugged, IPEN can be set to zero and the drawn grid will be suppressed.

GRIDSET does some internal rounding in the event that J or K do not partition the axes uniformly. However, for best results with

CALTEK

GRIDSET partitions the implicit screen into J*K rectangles as specified by the TEKPLOT version of this routine. However, GRIDSET has a limited use for CALTEK since the GRID routine is a no-op.

large J or K, the user should choose a number of regions that is an integral divisor of the axis in question.

GRIDSET may be called from any mode. The user is returned to his current mode by the routine.

The parameters J and K are not affected by the user's current scaling or rotation.

(27) GRID (JROW, KCOL, KHAR)

TEKPLOT

GRID is a graphics input routine. It must be used in conjunction with GRIDSET. When called, GRID causes the graphics cursor to appear. The user then positions the cursor and depresses a key. GRID returns a row and columns coordinate that is determined by the cursor location within the grid specified by the last call to GRIDSET.

Parameters are:

JROW - the row of the current grid corresponding to the cursor position.

KCOL - the column of

CALTEK

GRID is a no-op for CALTEK.

the current grid corresponding to the cursor position.

KHAR - an eight-bit ASCII character code corresponding to the key that was depressed.

GRID allows the user to graphically input a region coordinate in lieu of a point coordinate. It is useful for menu-picking or selecting areas of interest in graphs.

GRID reads the coordinate position of the cursor and converts this value to a (row, column) coordinate based upon the grid size specified in the user's last call to GRIDSET. If the cursor falls exactly at the dividing line between two adjacent cells in the grid, the larger coordinate is selected.

Row and column coordinates returned by GRID range from 1 to J and K respectively. That is, the first row is nominally called row 1; the first column is called column 1. An exception to this indexing scheme occurs if GRIDSET was called with J or K=0. In this case, GRID will return a zero

row or column coordinate corresponding to the axis that had zero paritions.

GRID may be called from any mode. The user is returned to his current mode by this routine.

The parameters JROW and KCOL are not affected by current scaling or rotation.

Example: The user wishes to write N lines of descriptive information on the screen. Each line corresponds to a choice. When the program is executed, an option will be selected by positioning the graphics cursor anywhere on the chosen line.

Assuming the user has written the code to print N options, one per line, the code to govern menu-picking might look as follows:

```
C     The T-4002 has 39 lines
C     in alphanumeric mode.
      CALL GRIDSET(0,39,0)
      CALL GRID(JCHOICE,NULL,
             KHAR)
C     At this point, JCHOICE
```

```
C     will have a value equal to
C     the index of the chosen
C     option.
```

4.7 Software Character Generation

While the T-4002 has an alphanumeric mode of operation, the Calcomp plotter does not. Hence, characters written on the screen in alpha mode using TEKPLOT will still appear on the screen when using CALTEK.

If the user wishes to annotate plotted output such that alphanumeric annotation will be carried over to Calcomp output, he must draw his characters. To this end, a software character generator is provided.

(28) WRITEX (STRING1, STRING2,...,STRINGJ)

WRITEX is a software character generator. The characters to be written are supplied as parameters to this routine.

Parameters are:

STRING - a sequence of two word Hollerith variables or constants, separated by commas. Each Hollerith parameter may contain up to eight characters to be drawn. A semi-colon (;) will terminate the current character string, but further parameters will still be processed.

All characters are drawn in vector mode. The lower left corner of each character commences at the current graphics position. Upon completion, the current position will be updated to the lower left corner of the next character to be drawn.

WRITEX may be called from any mode. The routine will print characters as specified and then return the user to his current mode.

Normal character size is 8x8; normal print direction is horizontally left to right. WRITEX leaves a 1/4 character width between adjacent characters and a 1/2 character height between adjacent lines. The routine is affected by current scaling and rotation. Thus, it is possible to write smaller, larger, taller, or fatter characters, print at an angle, etc.

The WRITEX character set includes:

A through Z
0 through 9
space, -, /, =, ., +, [,]
A carriage return is represented by a $
A line feed is represented by a #
An underline is represented by a (
A backspace is represented by a <

Any character not included in the 48-character set listed above will be ignored, except the semi-colon which terminates the current parameter string.

Example: The user wishes to use WRITEX to print

GUTENBERG IS THE MESSAGE.

at the top of his screen. He wishes to use double character size.

```
      PROGRAM JOHANN
      CALL ERASE
      CALL IPLOT(10,700,0,0)
C     Character generation begins at the current graphics
C     position.
C     It is permissible to plot in alpha mode with pen up.
      CALL SCALE(2,2,0,0)
C     Double the character size.
      CALL WRITEX(8HGUTENBER,8HG IS THE,8H MESSAGE,4H.;  )
      CALL BYENOW
      END
```

(29) WRITEY (FACT, DEG, STRING1,...,STRINGJ)

WRITEY is a software character generator similar to WRITEX. However, WRITEY allows the user to supply local scaling and rotation parameters.

Parameters are:

FACT - a local scaling factor that scales the normal character size.

DEG - a local rotation factor that determines the number of degrees of rotation from the orthogonal system of the screen. Positive rotation is in a counterclockwise direction.

STRING - a sequence of two-word Hollerith variables or constants, separated by commas. The character string is the same as the string for WRITEX.

All characters are drawn in vector mode, starting at the current graphics position. Unlike WRITEX, however, WRITEY does not update the current position.

The only scaling factor that affects the character size for WRITEY is FACT. Character size is not influenced by the user's current SCALE or SIZE.

The only rotation factor that affects WRITEY is DEG. The center of rotation will be the user's current position. This is dissimilar to the convention adopted for WRITEX, where the center of rotation is the user's origin.

WRITEY may be called from any mode. The routine will draw characters as specified and then return the user to his current mode.

The WRITEY character set and character spacing conventions are identical to those established for WRITEX.

Example: The user wishes to use WRITEY to draw a rotated label such as the one that occurs in the frontispiece of this manual.

```
      PROGRAM LABEL
      CALL ERASE
      CALL IPLOT(512,380,0,0)
C     Establish current position at center of the screen
      DO 100 I=0,330,30
      DEG=I
100   CALL WRITEY(2.,DEG,8H  CALTEK)
C     WRITEY does not update the current position
      CALL PAGE
      CALL BYENOW
      END
```

(30) XLATE (ICON, BCDCON)

XLATE converts the binary representation of a number to the equivalent BCD representation of that number. The routine can be used to convert an integer variable to a form suitable for use with WRITEX.

Parameters are:

ICON - an integer variable or constant that is to be converted to its BCD equivalent.

BCDCON - this parameter will be assigned the BCD equivalent of ICON.

XLATE suppresses leading zeros and left justifies the BCD string within BCDCON. If ICON is positive, the sign position of BCDCON will be a blank; if negative, the sign position will be a BCD minus. Finally, a semi-colon is inserted after the last BCD digit if the translation requires less than eight characters. The form of BCDCON is then suitable for input to WRITEX.

V. SELECTED EXAMPLES

In this section several examples are presented. These are intended to clarify the use of routines described in the preceding pages.

(a) Example 1 uses PLOT to draw a simple sine curve. Note the use of the loop variable to initialize the current graphics location.

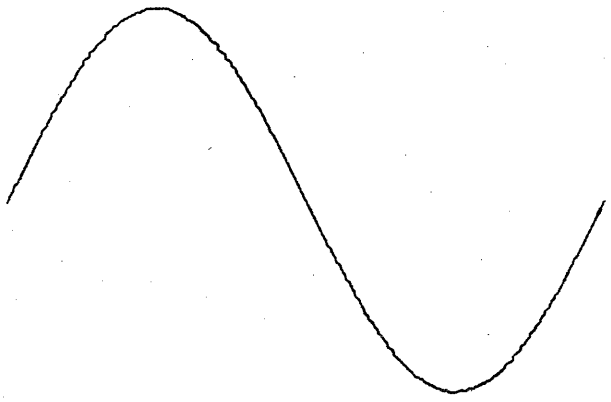
(b) Example 2 uses AXIS to draw an axis. DELTA is then employed to label the axis in alpha mode. Since the Calcomp plotter has no hardware character generator, this technique may not be used if a Calcomp copy of the labeled axis is desired.

(c) Example 3 uses AXIS and WRITEY to generate a labeled axis that can be carried over to Calcomp output. Note the use of XLATE to translate the loop variable to a BCD string for WRITEY.

(d) Example 4 shows how GRIN may be used to graphically input a set of (x,y) points together with pen values from the T-4002. In this example the points are both plotted on the screen and saved in a file named REMEMBER.

(e) Example 5 recalls (x,y) points and pen values from a file named REMEMBER and plots these. The program then allows the user to select a point of interest and a magnification factor. The graph is then replotted with a new origin chosen to displace the plot such that the point of interest coincides with the center of the screen. The graph is appropriately magnified upon replotting. This process demonstrates the use of GRIN to focus in on selected parts of a complicated plot.

```
      PROGRAM SINE
C     PROGRAM TO PLOT A SINE CURVE.
      CALL ERASE
      CALL VECTORS
      CALL SCALE(50,100,512,380)
C     SET ORIGIN TO MIDDLE OF SCREEN, AND
C     SCALE Y FOR AN AMPLITUDE OF 100.
      DO 100 I=0,630,10
      X=I/100.
      Y=SIN(X)
100   CALL PLOT(X,Y,I,0)
C     NOTE TECHNIQUE OF USING THE LOOP VARIABLE, I,
C     TO INITIALIZE THE CURRENT GRAPHICS LOCATION
C     FOR THE FIRST VECTOR.
      CALL PAGE
      CALL BYENOW
      END
```



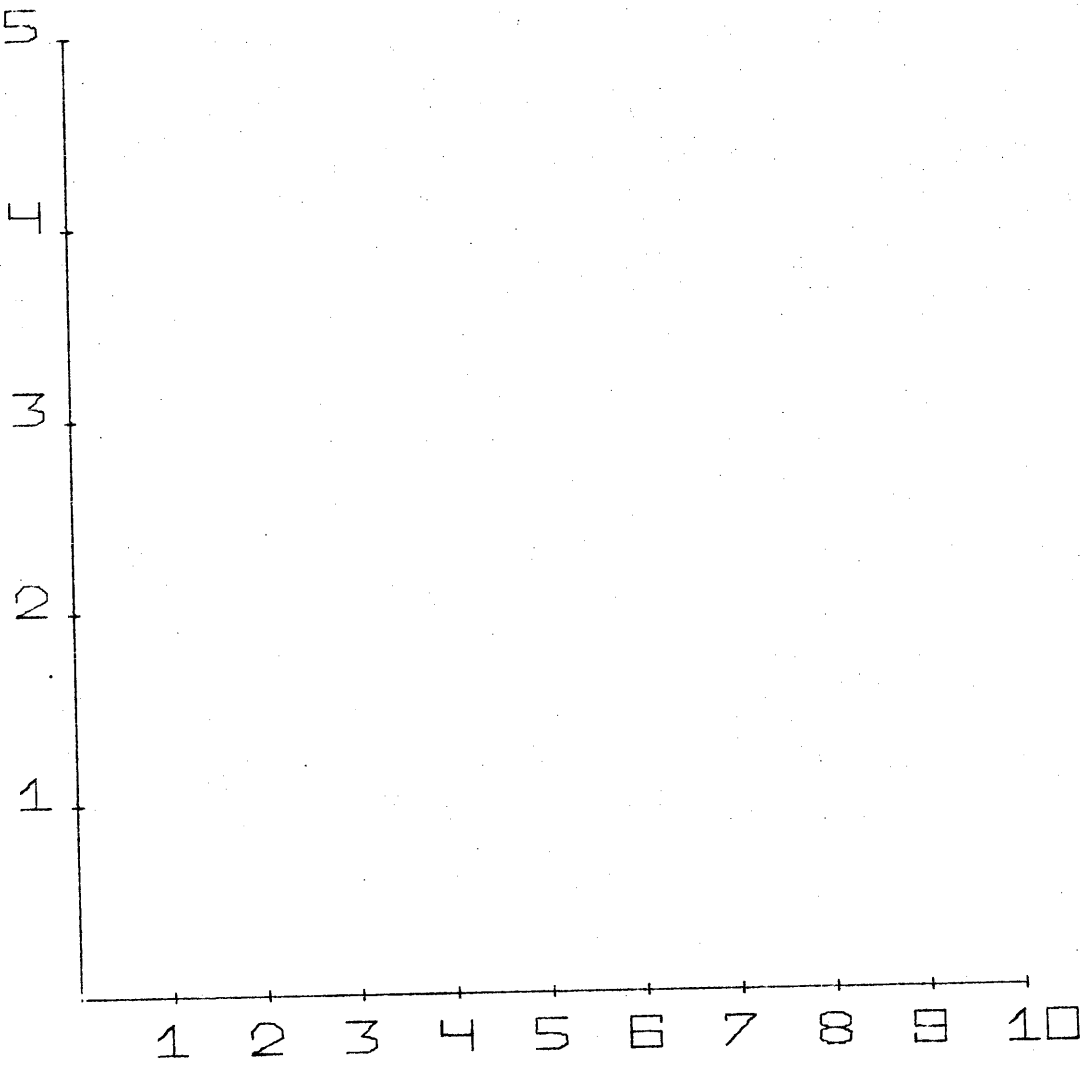
Example 1

```
      PROGRAM LABEL1
C     PROGRAM TO LABEL AN AXIS.
C     USES THE ALPHA MODE OF THE T-4002.
C     HENCE, LABELS WILL NOT APPEAR ON CALCOMP OUTPUT.
      CALL ERASE
      CALL FSCALE(10.,10.,200.,200.)
C     10 TEKPOINTS/SCALED POINT
      CALL AXIS(0.,50.,0.,50.,5.,10.)
C     DRAW THE AXIS
C     AXIS ROUTINE LEAVES TERMINAL IN VECTOR MODE,
C     POSITIONED AT THE ORIGIN.
      CALL ALPHAS
      CALL IDELTA(-1,-2,0,0)
C     DELTA TO SOUTH OF X AXIS.
C     NOW LABEL THE X AXIS
      DO 100 I=1,10
      CALL IDELTA(5,0,0,0)
C     DELTA OVER TO NEXT TIC MARK ON X AXIS..
100   WRITE(61,1) I
C     WRITE THE LABEL.
1     FORMAT(1X,I2)
C     NOTE THAT THE CARRIAGE RETURN AND LINE FEED
C     DO NOT AFFECT THE CURRENT GRAPHICS LOCATION FOR
C     THE NEXT DELTA.
C     NOW LABEL THE Y AXIS.
      CALL IPLOT(0,0,0,0)
C     PLOT BACK TO THE ORIGIN.
      CALL IDELTA(-2,0,0,0)
C     PLOT TO WEST OF THE Y AXIS.
      DO 200 I=1,5
      CALL IDELTA(0,10,0,0)
200   WRITE(61,2) I
2     FORMAT(1X,I1)
      CALL PAGE
      CALL BYENOW
      END
```

Example 2

```
      PROGRAM LABEL2
C     PROGRAM TO LABEL AN AXIS.
C     USES THE SOFTWARE CHARACTER GENERATOR.
C     HENCE, LABELS WILL BE CARRIED OVER TO CALCOMP OUTPUT
      CALL ERASE
      CALL FSCALE(10.,10.,200.,200.)
C     10 TEKPOINTS/SCALED POINT
      CALL AXIS(0.,50.,0.,50.,5.,10.)
C     DRAW THE AXIS
C     AXIS ROUTINE LEAVES TERMINAL IN VECTOR MODE,
C     POSITIONED AT THE ORIGIN.
      CALL ALPHAS
      CALL IDELTA(-1,-3,0,0)
C     DELTA TO SOUTH OF X AXIS.
C     NOW LABEL THE X AXIS
      DO 100 I=1,10
      CALL IDELTA(5,0,0,0)
C     DELTA OVER TO NEXT TIC MARK ON X AXIS..
      CALL XLATE(I,BCDCON)
100   CALL WRITEY(2.,0.,2H<;,BCDCON)
C     WRITE THE LABEL.
C     NOW LABEL THE Y AXIS.
      CALL IPLOT(0,0,0,0)
C     PLOT BACK TO THE ORIGIN.
      CALL IDELTA(-3,0,0,0)
C     PLOT TO WEST OF THE Y AXIS.
      DO 200 I=1,5
      CALL IDELTA(0,10,0,0)
      CALL XLATE(I,BCDCON)
200   CALL WRITEY(2.,0.,2H<;,BCDCON)
      CALL PAGE
      CALL BYENOW
      END
```

Example 3



Example 3 Illustration

```
      PROGRAM DIGITIZE
C     USE GRAPHICS INPUT TO DIGITIZE OR DRAW A PICTURE.
C     POINTS AND PEN VALUES ARE SAVED IN A FILE.
C     GRAPHICALLY INPUT POINTS ARE ECHOED BACK TO THE
C     T-4002 IN VECTOR MODE.
C        SPACE ECHOES AN INTENSIFIED VECTOR (PEN DOWN).
C        DEL ECHOES A DARK VECTOR (PEN UP).
C        BEL ENDS THE PROGRAM.
      DIMENSION JX(250),JY(250),JIPEN(250)
      CALL ERASE
      CALL EQUIP(10,8HREMEMBER)
C     SAVE ALL POINTS IN A FILE NAMED REMEMBER.
C     REWIND 10
      JCT=0
      CALL SCALE(1,1,512,380)
      CALL VECTORS
100   CALL IGRIN(J,K,KHAR)
C     INPUT A POINT.
      IF(KHAR.EQ.207B) GO TO 200
C     END THE PROGRAM IF CHARACTER WAS A BELL.
      IPEN=1$IF(KHAR.EQ.377B) IPEN=0
      CALL IPLOT(J,K,IPEN,0)
C     ECHO THE INPUT BACK TO THE T-4002.
      JCT=JCT+1$IF(JCT.GT.250) GO TO 200
      JX(JCT)=J$JY(JCT)=K$JIPEN(JCT)=IPEN
      GO TO 100
200   DO 300 I=1,JCT
300   WRITE(10,400) JX(I),JY(I),JIPEN(I)
C     WRITE THE POINTS OUT ON THE FILE.
400   FORMAT(2(I4),I1)
      END FILE 10
      CALL UNEQUIP(10)
      CALL PAGE
      CALL BYENOW
      END
```

Example 4

```
      PROGRAM MAGNIFY
C     PROGRAM TO READ A SET OF (X,Y) POINTS
C     FROM A FILE.
C     MAGNIFY PLOTS THE POINTS IN VECTOR MODE AND
C     THEN ALLOWS THE USER TO GRAPHICALLY INPUT A
C     POINT OF INTEREST TOGETHER WITH A MAGNIFICATION
C     FACTOR.
C     THE PROGRAM DISPLACES THE PLOT SUCH THAT THE
C     SELECTED POINT OF INTEREST COINCIDES WITH THE
C     CENTER OF THE SCREEN, AND THEN REPLOTS THE
C     MAGNIFIED SET OF POINTS.
C     IN THIS EXAMPLE, A MAXIMUM OF 250 POINTS CAN
C     BE PLOTTED.
      DIMENSION JPLOT(250,3)
      CALL EQUIP(10,8HREMEMBER)
      REWIND 10
      DO 100 I=1,250
      READ(10,1) J,K,IPEN
1     FORMAT(2(I4),I1)
      IF(EOF(10)) GO TO 200
      JPLOT(I,1)=J$JPLOT(I,2)=K$JPLOT(I,3)=IPEN
100   JCT=I
C     POINTS NOW IN CORE.
200   CALL UNEQUIP(10)
      XORG=512.$YORG=380.
      XFACT=1.$YFACT=1.
C     SET INITIAL SCALING AND ORIGIN
300   CALL FSCALE(XFACT,YFACT,XORG,YORG)
      CALL ERASE
      CALL VECTORS
      DO 400 I=1,JCT
400   CALL IPLOT(JPLOT(I,1),JPLOT(I,2),JPLOT(I,3),0)
C     PLOT THE POINTS WITH CURRENT SCALING AND ORIGIN.
      CALL GRIN(X,Y,KHAR)
C     GRAPHICALLY INPUT A POINT OF INTEREST.
      CALL PAGE
      A=TTYIN(4HMAGN,4HIFIC,4HATIO,4HN=   )
C     READ THE MAGNIFICATION FACTOR
      XFACT=XFACT*A$YFACT=YFACT*A
C     DETERMINE NEW SCALING FACTORS.
      XORG=512.-X*XFACT$YORG=380.-Y*YFACT
C     DETERMINE THE ORIGIN REQUIRED TO DISPLACE
C     THE POINT OF INTEREST SO THAT IT COINCIDES
C     WITH THE PHYSICAL CENTER OF THE SCREEN.
      GO TO 300
C     REPEAT THE PLOT WITH DESIRED MAGNIFICATION.
      END
```

Example 5

APPENDIX I - USASCII CONTROL CHARACTERS

Teletype Control Key	T-4002 Control Key	Octal Code
A	SOH	201
B	STX	202
C	ETX	203
D	EOT	204
E	ENQ	205
F	ACK	206
G	BEL	207
H	BS	210
I	HT	211
J	LF	212
K	VT	213
L	FF	214
M	CR	215
N	SO	216
O	SI	217
P	DLE	220
Q	DC1	221
R	DC2	222
S	DC3	223
T	DC4	224
U	NAK	225
V	SYN	226
W	ETB	227
X	CAN	230
Y	EM	231
Z	SUB	232

Shifted Teletype Control Key and	T-4002 Control Key	Octal Code
K	ESC	233
L	FS	234
M	GS	235
N	RS	236
O	US	237
P	NUL	200

Teletype Key	T-4002 Control Key	Octal Code
ALT MODE	}	375
RUB OUT	DEL	377

APPENDIX II - SUMMARY OF ROUTINES

The following summary lists the routines included in TEKPLOT/CALTEK together with their parameters. In addition, each routine is categorized with respect to four categories:

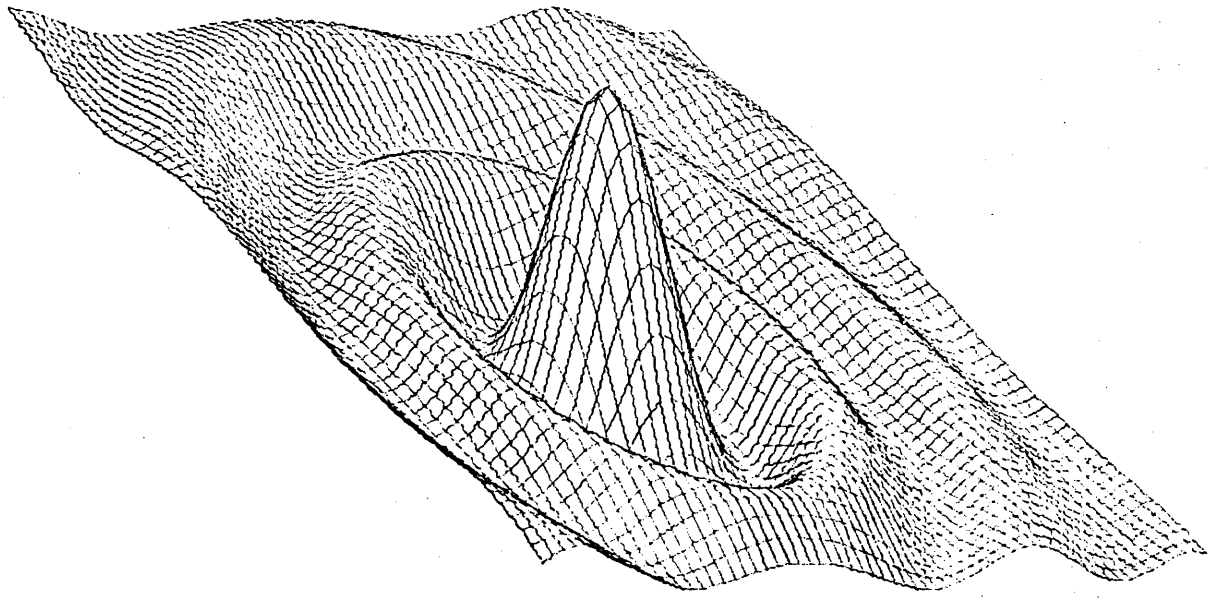
1) TEKPLOT - a Y in this column means that the routine accomplishes meaningful work.
an N in this column means that the routine is a no-op for TEKPLOT.

2) CALTEK - a Y in this column means that the routine accomplishes meaningful work.
an N in this column means that the routine is a no-op for CALTEK.

3) Mode - an A in this column means that the routine may be called from any mode, and the T-4002 will be left in alpha mode upon exit from the routine.
an I in this column means that the routine is mode independent: the routine sets its own mode internally, and returns the user to his current mode upon exiting.
a blank in this column means that the routine is mode dependent as described in the manual.

4) Location - a Y in this column means that the routine updates the current graphics location.
an N in this column means that the routine does not affect the current graphics location.

	TEKPLOT	CALTEK	MODE	LOCATION
Utility Routines				
ERASE	Y	Y	A	N
PAGE	Y	Y	A	N
TEKPAUSE	Y	Y	A	N
BYENOW	Y	Y	-	-
PLOTLUN (J)	N	Y	I	N
DOUBLE	Y	N	I	N
ITALICS	Y	N	I	N
NORMAL	Y	N	I	N
Mode Routines				
ALPHAS	Y	Y		N
VECTORS	Y	Y		N
POINTS	Y	Y		N
Axis Routines				
SCALE (KXFACT, KYFACT, KXORG, KYORG)	Y	Y	I	N
FSCALE (XFACT, YFACT, XORG, YORG)	Y	Y	I	N
SIZE (XMAG, YMAG)	N	Y	I	N
ROTATE (DEG)	Y	Y	I	N
AXIS (XLOW, XLNG, YLOW, YLNG, XTIC, YTIC)	Y	Y		Y
Character I/O				
KHOUT (J)	Y	Y	I	N
KHIN (N)	Y	Y	I	N
Plotting Routines				
PLOT (X, Y, IPEN, MARK)	Y	Y		Y
IPLOT (JX, JY, IPEN, MARK)	Y	Y		Y
DELTA (DELX, DELY, IPEN, JRETURN)	Y	Y		Y
IDELTA (JDELX, JDELY, IPEN, JRETURN)	Y	Y		Y
INCPLOT (JDIR, IPEN, NUM, KRETURN)	Y	Y	I	N
Graphics Input				
GRIN (X, Y, KHAR)	Y	N	I	N
IGRIN (JX, JY, KHAR)	Y	N	I	N
GRIDSET (J, K, IPEN)	Y	Y	I	N
GRID (JROW, KCOL, KHAR)	Y	N	I	N
Software Characters				
WRITEX (STRING1, STRING2, ..., STRINGJ)	Y	Y	I	Y
WRITEY (FACT, DEG., STRING1,...,STRINGJ)	Y	Y	I	N
XLATE (ICON, BCDCON)	Y	Y	I	N



TEKPLOT/CALTEK (revised)

In the previous version of TEKPLOT and CALTEK, off-screen plotting was constrained to the closest point still on-screen. In the current version, off-screen plotting is suppressed entirely; only that portion of a plot that is on-screen will be plotted. The INCPLOT routine is still an exception to these rules, as noted in the manual.

With this new convention, TEKPLOT/CALTEK allows the user to think of the display screen as a 'window' into two dimensional space. The TEKPLOT/CALTEK routines allow this window to be translated to any particular area of interest, rotated any arbitrary number of degrees and used as a magnifier.

In addition to the change in off-screen plotting conventions, two new subroutines have been added. WINDOW and WRITEZ are described in the following pages.

Finally, the conventions for loading TEKPLOT/CALTEK, as described on page 5 of the manual, have been changed. TEKPLOT and CALTEKV are now included in the standard FORTRAN library. CALTEK is still stored in *CALTEK.

Following the example on page 5, to load your program together with routines from TEKPLOT and/or CALTECV, you may type:

```
#FORTRAN,I=YOURPROG,R
```

To load CALTEK routines in lieu of the corresponding TEKPLOT routines, type:

```
#FORTRAN,I=YOURPROG,X
#LOAD,56,L=*CALTEK
RUN
```

NOTE:

An error has been noted on page 43 of the manual. Calling ROTATE does update the user's current graphics location. Also, calling SCALE or FSCALE will update the current location if current rotation is non-zero.

WINDOW (XMIN,YMIN,XMAX,YMAX,IPEN)

The WINDOW subroutine is used to set the window size for all subsequent plotting. Any plotting external to the current window will be clipped at the window margin.

Parameters are:

XMIN - the minimum x value of a rectangular window. XMIN should be given in scaled, rotated units. Default value=tekpoint 0.

YMIN - the minimum y value of the window. YMIN should be given in scaled, rotated units. Default value= tekpoint 0.

XMAX - the maximum x value of the window, specified in scaled, rotated units. Default value=tekpoint 1023.

YMAX - the maximum y value of the window, specified in scaled, rotated units. Default value=tekpoint 761.

IPEN - an optional parameter. If IPEN ≠ 0, then a border will be drawn about the current window. If IPEN = 0, then the border will be omitted.

WINDOW converts the user's parameters to tekpoints and saves them in the terminal status area for future use. If the XMAX parameter turns out to be smaller than XMIN, then these are interchanged. Similarly, if YMAX is less than YMIN, then these are interchanged.

If the WINDOW parameters specify a window that is larger than the screen size, then the designated window is collapsed to the screen size.

WINDOW may be used to shrink the effective screen size in order to create protected areas on the screen. The routine may be used to display multiple plots on the same screen or bring together disparate parts of a plot for comparison.

Since the window parameters are specified in terms of scaled units, parameters may be set by visual inspection of a plot using the GRIN routine.

```
CALL GRIN (XMIN,YMIN,KHAR)
CALL GRIN (XMAX,YMAX,KHAR)
CALL WINDOW (XMIN,YMIN,XMAX,YMAX,1)
```

WRITEZ (FACT,DEG,NUM,ARRAY)

WRITEZ is a software character generator similar to WRITEY. However, WRITEZ allows the user to supply his character string to an array instead of in the calling parameters.

Parameters are:

FACT - A local scaling factor that scales the normal character size.

DEG - A local rotation factor that determines the number of degrees of rotation from the orthogonal system of the screen. Positive rotation is in a counter-clockwise direction.

NUM - The effective dimension of ARRAY.

ARRAY - A floating array that contains the BCD codes of the characters to be generated, at most, eight characters per array element.

WRITEZ will process NUM array elements in ARRAY; each array element may contain up to eight BCD codes. If a semi-colon is encountered in the BCD string at the jth element in ARRAY, then no further characters will be generated from ARRAY(J). However, if NUM>J, then WRITEZ will proceed to ARRAY(J+1). This is an extension of the semi-colon terminator convention used by WRITEX and WRITEY.

All characters are drawn in vector mode, starting at the current graphics position. Like WRITEY, WRITEZ does not update the current position.

The only scaling factor that affects the character size for WRITEZ is FACT. Character size is not influenced by the user's current SCALE of SIZE.

The only rotation factor that affects WRITEZ is DEG. The center of rotation will be the user's current graphics position, not the origin.

WRITEZ may be called from any mode. The routine will draw characters as specified and then return the user to his current mode.

The WRITEZ character set and character spacing conventions are identical to those established for WRITEX.