

64015 C.1



GRAPH III

A Digitizing and Graph Plotting Program

Colin B. Selleck

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550
for the United States Department of Energy
under Contract DE-AC04-76DP00789

01167001

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof or any of their contractors or subcontractors.

Printed in the United States of America
Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161

NTIS price codes
Printed copy: A04
Microfiche copy: A01

Distribution
Category UC-32

SAND86-0525
Unlimited Release
Printed March 1986

GRAPH III

A Digitizing and Graph Plotting Program

Colin B. Selleck
Computational Physics and
Mechanics Division I
Sandia National Laboratories
Albuquerque, New Mexico

ABSTRACT

GRAPH is an interactive program that allows the user to perform two functions. The first is to plot two dimensional graphs and the second is to digitize graphs or plots to create data files of points. The program is designed to allow the user to get results quickly and easily. It is written in RATFIV (a FORTRAN preprocessor) and is currently in use at Sandia under VMS on a VAX computer and CTSS on a Cray supercomputer. The program provides graphical output through all of the Sandia Virtual Device Interface (VDI) graphics devices.

Contents

1	Introduction	11
1.1	Access to the Program	11
1.1.1	GRAPH Under VMS	12
1.1.2	GRAPH Under CTSS	13
1.2	Command Summary	14
1.3	Software Text	16
1.4	Classified	17
1.4.1	Classified Computing under VMS	17
1.4.2	Classified Computing under CTSS	20
1.5	Digitizing	20
2	GRAPH Commands	21
	ANGLE	21
	APPEND	22
	ARC	22
	ARROW	23
	AUTOERASE	23
	AXIS	24
	BELL	25
	CIRCLE	25

CLOSE	25
COLOR	26
CURVE	27
DECIMAL	27
DIGITIZE	28
DRAW	28
ECHO	29
EXIT	29
EXPONENT	29
FONT	30
HARDCOPY	30
HELP	30
ID	30
LENGTH	31
LOGO	31
LSTYLE	32
MINOR	33
OPEN	33
ORIGIN	33
OSC	34
PAGE	34
PATH	35
PAUSE	35
PLOT	36
RECTANGLE	37
REPLOT	37
RESET	38

SCALE	38
SCIENTIFIC	39
SET	39
SIZE	40
STATUS	41
SYMBOL	42
TABLET	42
TITLE	43
TRANSLATE	44
TYPE	45
VECTOR	45
WAIT	46
WIDTH	46
WRITE	47
YNUMBER	47
2.1 Program Defaults	48
2.1.1 Miscellaneous Parameters	48
2.1.2 Graphing Parameters	48
2.1.3 Digitizing Parameters	49
3 Examples	51
3.1 Example 1—Digitizing a Graph	51
3.2 Example 2—Digitizing Points	52
3.3 Example 3—Plotting Graphs	53
3.4 Example 4—Plotting Graphs with Changes in Default Settings	53
3.5 Example 5—Plotting Graphs with User Scaling	53
Appendix A. Obsolete Commands	59
References	61

List of Figures

3.1	Default Plot	55
3.2	Plot with Changes in Default Settings	56
3.3	Plot with User Scaling	57

List of Tables

1.1	Currently Supported Devices under VMS	14
1.2	English and Greek Characters in Roman and Stick Fonts	18
1.3	Escape Sequences for Software Text	19

Chapter 1

Introduction

GRAPH was written utilizing the PLT plotting package developed by the author under the ARIES project in Department 1530.¹ PLT utilizes the Sandia Virtual Device Interface (VDI) developed by Division 2644,² so graphical output is available on numerous devices. This SAND report documents version 10.0 of **GRAPH** and replaces two earlier SAND reports (SAND 83-0464 and SAND 84-0302). Appendix A documents obsolete commands from these earlier versions of **GRAPH**.

GRAPH takes the command-oriented approach toward interactive programming: the user is required to give commands in response to a prompt. This requires the user to know a basic subset of commands, but the commands are English-language based and are quite easy to remember. There is a lot of flexibility built into the program, *e.g.* the user can tailor the plot however he chooses through the use of different commands. However, all of these tailoring features default to some value at program initialization. Thus, the user can generate plots and digitize graphs with little knowledge of the entire set of commands.

GRAPH supports both English and Greek characters as well as many special symbols such as the Sandia logo, circles, squares, *etc.* **GRAPH** is available for use under VMS on Digital Equipment Corporation VAX computers and under CTSS on Cray computers.

1.1 Access to the Program

GRAPH is available under two operating systems: VMS on VAX computers and CTSS on Cray computers. Execution of **GRAPH** under VMS differs somewhat from execution under CTSS. Under VMS, the executable images for each supported device are prelinked by the author and stored in a system directory. Under CTSS, the executable image is linked by the user and stored in his directory. Therefore, under VMS, the graphics device that you are using may not be available; the **GRAPH** command

will inform you of this. The only option left is to contact the author and request that support for your device be added.

1.1.1 GRAPH Under VMS

GRAPH is supported under the UCODES feature on most VAXes at Sandia National Laboratories in Albuquerque. To get information on how to use this program, give the command:

```
UHELP GRAPH
```

This command will tell you that to use GRAPH under VMS, the following line is added to your LOGIN.COM file (you may opt to give this command interactively).

```
$ @SNL$UCODES:GRAPH
```

This assumes that your VAX supports UCODES. If this is not the case, contact the author for the program.

Then, to execute GRAPH, give the following command.

```
$ GRAPH dev [file] [plotfile]
```

The square brackets indicate that the argument is optional; the brackets must not be present in the command. The three arguments are defined as follows.

dev The three character VDI device code supported by Division 2644. The currently supported devices under VMS are shown in Table 1.1 on page 14. If you leave off the device code on the GRAPH command, you will get a list of the available devices. If Table 1.1 shows the device type to be BATCH, then an output graphics file will be created; see the following description on *plotfile*.

file An optional file containing GRAPH commands. If this argument is present, GRAPH will execute the commands in *file* and then exit. This is useful for subprocesses and batch jobs.

plotfile An optional file specification for the name of the output file if you are using a batch device. The name of the output file for batch devices is determined in the following way.

1. If you specify a filename with the optional third argument, GRAPH will of course use that with a default extension equal to the device code (*e.g.* .I30). For example, if you say "GRAPH I30 COPPER CU", GRAPH will create an output file named CU.I30.

2. If you do not specify an output file but do specify an input file of **GRAPH** commands, then the output file will have the same name as your input file with an extension equal to the device code. For example, if you say "GRAPH I30 LEAD", **GRAPH** will create an output file named LEAD.I30.
3. If you specify neither an input nor output filename, then the output file will be named GRAPH.*dev* where *dev* is the device code. For example if you say "GRAPH I30", **GRAPH** will create an output file named GRAPH.I30.

The file SNL\$UCODES:GRAPH.TXT is a listing of the help file. It will always have the latest updates in it.

Under VMS, you can define a default plot device for **GRAPH** to use. If the symbol PLOT_DEVICE is defined (in your LOGIN.COM file for instance) to be one of the valid devices, then, to use **GRAPH**, all you have to do is give the command **GRAPH**. The program will automatically use the plot device specified by PLOT_DEVICE. If you wish to use a different device, give the command **GRAPH dev**. You can define the symbol PLOT_DEVICE under VMS with a command similar to the following.

```
$ PLOT_DEVICE == "LS5"
```

You should replace "LS5" with the appropriate device code.

1.1.2 GRAPH Under CTSS

GRAPH is built under CTSS by using CCL (Common Command Language). To use **GRAPH** under CTSS, give the following commands while using CCL. You may opt to put these commands in your CCLLIBC file; this is the file of CCL commands that is executed every time you run CCL.

```
mass get /e00022647/commands/cbslib  
select lib='cllibc cbslib'
```

The first command retrieves a library file of commands from the IFS. The second command selects your CCLLIBC file and this library file as files to search when CCL is looking for a command. You can add other files in the SELECT command if needed. Then, to run **GRAPH**, give the command:

```
GRAPH dev [comfil]
```

Here, *dev* is the VDI device code. *comfil* is an optional file of **GRAPH** commands to be executed; **GRAPH** will exit after these commands are executed. The square brackets indicate that the argument is optional; the brackets must not be present in the command.

Table 1.1: Currently Supported Devices under VMS

Code	Type	Description
24L	Batch	24X Legal Fiche
35C	Batch	Dicomed 35mm Color Slides
DT4	Interactive	Datamedia DT80/Tektronix 4014 combination
HC1	Batch	Versatec V-80
I10	Batch	Imagen Imprint 10
I30	Batch	Imagen 300 dpi Laser Printers
JP1	Interactive	Jupiter 1024 (will work on AED 1024 also)
JP7	Interactive	Jupiter 7 (will work on AED 767 also)
LS5	Interactive	Lear Siegler 7105 or 7107 or Envision
MET	Batch	Metafile
QMS	Batch	QMS laser printer
R25	Interactive	Raster Technologies 1/25 (serial)
RET	Interactive	Retrographics
T05	Interactive	Tektronix 4105
T15	Interactive	Tektronix 4115, 4125
TEK	Interactive	Tektronix 4010 (also 4010 emulators— <i>e.g.</i> IBM PC
TK4	Interactive	Tektronix 4014/4015
V25	Interactive	Digital VT125

In addition to running **GRAPH**, this command will build a version of **GRAPH** if it is not already present in your directory. Also, all necessary files to build and run **GRAPH** will be retrieved from the IFS if they are not already present in your directory. The files **GRAPHdev**, **STKFNT**, **ROMFNT**, and **GRAMES** will be left in your directory and are required to execute **GRAPH**. For a list of supported VDI devices on CTSS give the command **HELP GCS DEVICES** while logged into CTSS.

If you specify a batch device under CTSS, the output file will be named **TAPE77**. It is up to you to plot this file.

The IFS file **/ARIES/GRAPH/GRATXT.S** is a listing of the help file in **STEXT** format. It will always have the latest updates in it. You may use the **MASS** utility to retrieve this file and then use the **NTEXT** utility to convert it to your computers format.

1.2 Command Summary

The commands are described in Chapter 2 of this report. Some general notes on how the commands are interpreted and some special features are described here.

The program has extensive internal help similar to the HELP command on the VAX. Just give the command HELP and a list of available commands will be displayed.

Upon startup, some messages may be printed on your screen. These messages will inform you of current developments in GRAPH. You may see these messages again by giving the command HELP MESSAGES. Also, the changes to GRAPH since the publication of this report can be seen by giving the command HELP CHANGES.

Some of the commands require dimensions. There are two types of coordinates: device and graph. The device coordinates range from 0.0 to 1.0 in the x direction and 0.0 to 0.75 in the y direction. Graph coordinates are the coordinate system on the graph axes. When specifying numbers to GRAPH, the decimal point is optional.

An interrupt character is available that returns control to the prompt level. This is useful for breaking out of prompts, for breaking out of command files, or when you are plotting a graph that takes a long time and you decide to terminate it. Under VMS, the interrupt is control-c (hold down the control key and hit a "c"). Under CTSS, the interrupt is control-e,i,<CR> (hold down the control key and hit an "e", release the control key, hit an "i", and then hit carriage return).

The commands can all be shortened to whatever makes them unique, but must be at least two characters long. They may be either upper or lower case. GRAPH will prompt the user for all missing information.

GRAPH uses a command parser to interpret commands. A space or comma is a delimiter between words. If, while typing a command line, the user needs to include spaces or commas in a text string, such as a plot title, the user must enclose this string in double quotes (""). This forces GRAPH to interpret the quoted string as a single word. This must also be done to force a number to be a word.

Under VMS, GRAPH first searches for a file equivalenced to the logical name GRAPH_COMMAND (this defaults to GRAPH.CMD but can be changed by the user with an ASSIGN command). Under CTSS, GRAPH first searches for a file named GRACMD. If this file is present under either operating system, the GRAPH commands in that file are executed. This is useful if you desire to change some of the default settings of GRAPH.

You can create command files with GRAPH commands in them and then execute these files from within GRAPH. The format of this command is the "@" character followed by the file specification. For example:

```
@LARGE.COM
```

The file LARGE.COM is opened and the GRAPH commands in it are executed. Control is then returned to the terminal. Nesting of command files is not allowed. Under VMS, the default file extension for command files is .CMD. This is especially useful for creating hardcopy output on a laser printer. The user first creates a command file with

an editor and then uses an interactive terminal to get everything just right (the **OSC** command is good for this). Then, the user executes **GRAPH**, specifying a laser printer as the device code. Then, the command file is executed and hardcopy is produced.

Data files for **GRAPH** can have up to 50 columns of data in them. Furthermore, the columns do not have to be the same length. The data is read in free format, and thus can be separated by commas, tabs, or any number of spaces. Under VMS, the default file extension for data files is `.DAT`.

The “!” character is a comment character. If this character is encountered anywhere on a command line or in a data file, the rest of the line is ignored. This is especially useful for commenting command and data files.

The use of a graphics cursor is supported by **GRAPH**. The cursor is typically a horizontal and a vertical line or a small cross or X. The user can move the cursor around with thumbwheels or a joystick. The user hits any key on the keyboard except carriage return to indicate that the cursor is at the desired place on the screen.

1.3 Software Text

GRAPH uses escape sequences in text strings. An escape sequence is an embedded command in a text string. It is delineated by an escape character followed by some number of characters. The escape character is a backslash (“\”). A single space is expended after all escape sequences except for “\^”, “_”, and “_”. However, escape sequences can be concatenated together without spaces between them. The following example is a valid use of escape sequences.

This is a line.\cr\lf This is the next line.

Since `\cr` is a carriage return and `\lf` is a linefeed, this would result in two lines of text, the second below the first.

You can include superscripts and subscripts in titles by using the following escape sequences.

\^ — superscripts
_ — normal
_ — subscripts

For example, the text

`x_ij_ + y\^2`

would produce

$x_{ij} + y^2$

Note that these escape sequences act as switches and you must turn off superscripts or subscripts with a “\-”.

Greek characters can be used by using the following escape sequences.

\gr — Greek characters
\eng — English characters

Like the superscript and subscript escape sequences, these act as switches. For example,

These are \gr greek \eng characters.

would result in the following.

These are $\gamma\rho\epsilon\epsilon\kappa$ characters.

The correspondence between English and Greek characters is shown in Table 1.2 on page 18. For example, an “a” will give α , a “b” will give β , *etc.* This table also shows the two fonts available to the user—Roman and Stick. The first four columns are the upper case characters and the second four columns are the lower case characters.

Table 1.3 on page 19 shows all of the escape sequences supported by GRAPH. The escape sequences that draw the various line styles (\ddline, *etc.*) are useful for drawing legends.

1.4 Classified

1.4.1 Classified Computing under VMS

Under VMS, GRAPH will run from a classified account. However, no security markings are put on the plots. It is up to you to make sure all of your output is marked appropriately. The only difference between running GRAPH unclassified or classified is the need for a classification level. GRAPH does this by translating the logical name \$CATEGORY. The valid categories are:

RD — Restricted Data
FRD — Formerly Restricted Data
NSI — National Security Information

You do not have to assign this logical name; GRAPH will prompt you for the category. To prevent this query, you can assign this logical name in your LOGIN.COM file or with an interactive DCL command with a command similar to the following:

```
$ ASSIGN RD $CATEGORY
```

You should replace RD with FRD or NSI when appropriate.

Table 1.2: English and Greek Characters in Roman and Stick Fonts

Roman English	Stick English	Roman Greek	Stick Greek	Roman English	Stick English	Roman Greek	Stick Greek
A	A	A	A	a	a	α	α
B	B	B	B	b	b	β	β
C	C	H	H	c	c	γ	γ
D	D	Δ	Δ	d	d	δ	δ
E	E	E	E	e	e	ε	ε
F	F	Φ	Φ	f	f	φ	φ
G	G	Γ	Γ	g	g	γ	α
H	H	X	X	h	h	χ	χ
I	I	I	I	i	i	ι	ι
J	J	J	J	j	j	ε	ε
K	K	K	K	k	k	κ	κ
L	L	Λ	Λ	l	l	λ	λ
M	M	M	M	m	μ	μ	μ
N	N	N	N	n	ν	ν	ν
O	O	O	O	o	ο	ο	ο
P	P	Π	Π	p	ρ	π	π
Q	Q	Θ	Θ	q	ρ	θ	θ
R	R	P	P	r	ρ	ρ	ρ
S	S	Σ	Σ	s	σ	σ	σ
T	T	T	T	t	τ	τ	τ
U	U	V	V	u	υ	υ	υ
V	V	Ω	Ω	v	φ	φ	φ
W	W	Ξ	Ξ	w	ω	ω	ω
X	X	Ψ	Ψ	x	ξ	ξ	ξ
Y	Y			y	ψ	ψ	ψ
Z	Z			z	ζ	ζ	ζ

Table 1.3: Escape Sequences for Software Text

Escape Sequence	Function
\\	backslash character
\cci	centered circle
\ccs	centered cross
\cdi	centered diamond
\cdo	centered dot
\cl	carriage return and linefeed
\clo	centered logo
\ci	circle
\cr	carriage return
\cs	cross
\csq	centered square
\ctr	centered triangle
\cx	centered X
\di	diamond
\ddline	dot-dashed line
\dline	dotted line
\do	dot
\lf	linefeed
\ldline	long dashed line
\lo	logo
\mdline	medium dashed line
\plusmin	plus and minus sign
\sdline	short dashed line
\sline	solid line
\sq	square
\tr	triangle
\x	X

1.4.2 Classified Computing under CTSS

Classified computing is not supported on CTSS as CTSS is not classified.

1.5 Digitizing

GRAPH allows you to digitize either points or graphs by giving the **DIGITIZE** command. If you are digitizing points, they are multiplied by a scale factor and added to a translation factor that you can set with the **SCALE** and **TRANSLATE** command. If you are digitizing graphs, you are asked to digitize a point on both the x and y axes as well as the origin of the graph. As you digitize each point, you are asked to enter the graph coordinates of the point just digitized. In this manner, **GRAPH** automatically scales and translates the digitized points to your graph's coordinate system. Note that the orientation of the graph does not matter. All digitized points are written to all currently open files. Thus, the first command given when digitizing is typically an **OPEN** or **APPEND**. These commands open the files that the digitized points are written to. **APPEND** writes to the end of an existing file; **OPEN** creates a new file.

Chapter 2

GRAPH Commands

This chapter defines all of the **GRAPH** commands. The command defaults are designated by a (D) where appropriate throughout this chapter. When the format of a command is described, the **GRAPH** command is set in boldface, while command qualifiers to be supplied by the user are set in italics. A command qualifier may be either parameters supplied by the user (*e.g.* a filename) or a **GRAPH** keyword (*e.g.* **SOLID**). Anything enclosed in square brackets ([]) is optional; the brackets must not be included in the command line.

ANGLE

Sets the angle of various options.

FORMAT: **ANGLE** *keyword angle*

PARAMETERS:

angle The new angle in degrees.

KEYWORDS:

ARROWHEAD Sets the angle between the arrowhead and the arrow body to *angle*; defaults to 15 degrees.

CHARACTERS Sets the angle with which to plot characters within a text string to *angle*; defaults to 0 degrees.

TEXT Sets the angle with which to plot the entire text string to *angle*; defaults to 0 degrees.

APPEND

Opens an existing file for appending digitized points to.

FORMAT: **APPEND** *filename*

PARAMETERS:

filename The file you want to append to. Under VMS, the default file extension is .DAT.

ARC

Draws an arc in the current foreground color.

FORMAT: **ARC** [*keyword*₁] [*keyword*₂] [*x*_{*c*}, *y*_{*c*} *x*_{*e*}, *y*_{*e*} *A*_{*b*}, *A*_{*e*}]

PARAMETERS:

*x*_{*c*}, *y*_{*c*} The center of the arc.

*x*_{*e*}, *y*_{*e*} A point on the edge of the arc.

*A*_{*b*}, *A*_{*e*} The beginning and ending points of the arc, measured in degrees counterclockwise from the positive *x* axis.

KEYWORDS:

CURSOR Puts the cursor on the screen so the user can input the coordinates.

DEVICE Denotes that the coordinates are in device coordinates.

FILL Fills in the arc; this is only supported on terminals that have hardware polygon fill, such as most raster devices. If this keyword is left off, only the outline of the arc is drawn.

GRAPH Denotes that the coordinates are in graph coordinates (D).

RADIAL Draws the radial lines. For arcs that are not filled in you can specify whether or not you want the lines from the arc center to each arc endpoint drawn.

The arc is drawn counterclockwise from the beginning angle to the ending angle.

ARROW

Draws an arrow in the current foreground color.

FORMAT: **ARROW** [*keyword*] [x_0, y_0 x_1, y_1]

PARAMETERS:

x_0, y_0 x_1, y_1 The coordinates of the arrow; the head will be at x_1, y_1 .

KEYWORDS:

CURSOR Puts the cursor on the screen so the user can input the endpoints.

DEVICE Denotes that the endpoints are in device coordinates.

GRAPH Denotes that the endpoints are in graph coordinates (D).

The size of the arrowhead can be set with the **SIZE** command and the angle between the arrowhead and the arrow can be set with the **ANGLE** command.

AUTOERASE

Controls the automatic erasing of the screen when the **PLOT** command is given.

FORMAT: **AUTOERASE** *keyword*

KEYWORDS:

ON Enables erasing of the screen when the **PLOT** command is given (D).

OFF Disables erasing of the screen when the **PLOT** command is given.

The screen is always erased when the **REPLOT** command is given. The user can erase the screen at any time with the **PAGE** command. The default is **AUTOERASE ON**.

AXIS

Sets axis parameters for drawing graphs.

FORMAT: **AXIS** *keyword*

KEYWORDS:

FULL Draws a full box for axes (D).

LINEAR Sets linear x and y axes (D).

LOG X Sets logarithmic x axis.

LOG Y Sets logarithmic y axis.

LOG XY Sets logarithmic x and logarithmic y axes.

NOFULL Draws only bottom and left axes.

SCALE DIFFERENT Allows different x and y axes scaling (D).

SCALE EXACT [$x_m, x_M, x_f, i_x, m_x, y_m, y_M, y_f, i_y, m_y$] Allows the user to set exact scale. This scaling type is useful as the axes do not have to start with a major tick. x_m, x_M refer to the minimum and maximum x coordinate on the x axis. x_f refers to the first number on the x axis to start labeling. i_x is the interval between major ticks on the x axis. m_x is the number of minor ticks (ticks between major ticks) on the x axis. The rest of the parameters refer to the y axis. For logarithmic axes, the first number, the interval, and the number of minor intervals must be specified, but are ignored.

SCALE SAME Forces same x and y axes scaling.

SCALE USER [$x_m, x_M, y_m, y_M, M_x, m_x, M_y, m_y$] Allows the user to set the axis scale. x_m, x_M refer to the minimum and maximum x coordinate on the x axis. M_x, m_x refer to the number of major (numbered) ticks and minor ticks (ticks between major ticks) on the x axis. The rest of the parameters refer to the y axis. For logarithmic axes, the number of major and minor ticks must be specified, but are ignored.

You can specify partial decades on logarithmic axes. Minor tick labeling on logarithmic axes can be turned on or off with the **MINOR LABEL** command. You can choose the type of labeling (scientific or normal) with the **SCIENTIFIC** command. If you do not specify any numbers to the **AXIS SCALE USER** or the **AXIS SCALE EXACT** command, **GRAPH** will prompt you for the numbers, allowing you to use the current value if you wish. The defaults are full, linear, scaled differently axes.

BELL

Controls the ringing of the terminal bell after each digitized point.

FORMAT: **BELL** *keyword*

KEYWORDS:

ON Enables ringing of the terminal bell for each digitized point (D).

OFF Disables ringing of the terminal bell for each digitized point.

The default is **BELL ON**.

CIRCLE

Draws a circle in the current foreground color.

FORMAT: **CIRCLE** [*keyword*₁] [*keyword*₂] [*x*_c, *y*_c *x*_e, *y*_e]

PARAMETERS:

*x*_c, *y*_c The coordinates of the center of the circle.

*x*_e, *y*_e The coordinates of an edge of the circle.

KEYWORDS:

CURSOR Puts the cursor on the screen so the user can input the coordinates.

DEVICE Denotes that the coordinates are in device coordinates.

FILL Fills in the circle; this is only supported on terminals that have hardware polygon fill, such as most raster devices. If this keyword is left off, only the outline of the circle is drawn.

GRAPH Denotes that the coordinates are in graph coordinates (D).

CLOSE

Closes the file *filename*.

FORMAT: **CLOSE** *filename*

PARAMETERS:

filename The file you want to close. Under VMS, the default file extension is .DAT. If *filename* is **ALL**, closes all files.

COLOR

Colors different parts of the graph.

FORMAT: **COLOR** *object color*

PARAMETERS:

object The object to color; the valid objects are:

AXIS Colors the axes.

BACKGROUND Colors the background; the screen must be erased before the background color will change.

CURVE Colors the curve.

FOREGROUND Colors the foreground; this is how you change the color of arcs, circles, rectangles, titles, *etc.*

LABEL Colors the axis labels.

MAJGRID Colors the major grid.

MINGRID Colors the minor grid.

NUMBERS Colors the axis numbers.

SYMBOLS Colors the symbols.

TICKS Colors the axis ticks.

ZLINE Colors the zero lines.

color The color to make *object*. The valid colors are:

BLACK	BLUE	CYAN	DKGRAY
GRAY	GREEN	LIME	LTBLUE
LTGRAY	MAGENTA	ORANGE	PINK
RED	VIOLET	WHITE	YELLOW

Zero lines are lines drawn through 0.0 parallel to the coordinate axes. **LT** stands for light and **DK** stands for dark. All colors default to white except the background which defaults to black. This command is ignored on black and white devices.

CURVE

Draws a curve on existing axes.

FORMAT: **CURVE** *filename* [c_x, c_y] [p_1, p_2]

PARAMETERS:

filename The file you want to plot; the file format is ASCII. The maximum number of points per file is 10000.

c_x, c_y The columns of data to read out of *filename*. If omitted, they default to 1,2. The maximum number of columns is 50.

p_1, p_2 The starting and ending positions of the data to read out of *filename*. This is essentially the starting and ending row numbers of the two columns of data you wish to read. If omitted, the default is to read all complete rows of both columns.

DECIMAL

Sets the number of decimal places in the numbers on the axes.

FORMAT: **DECIMAL** *keyword number*

PARAMETERS:

number A positive integer specifying the new number of decimal places. $0 \leq \textit{number} \leq 9$.

KEYWORDS:

X Sets the number of decimal places on the x axis to *number*.

Y Sets the number of decimal places on the y axis to *number*.

The default is one decimal place on both the x and y axes. If *number* is not an integer, it will be truncated to an integer, and if it is negative, it will be made positive.

DIGITIZE

Digitizes points or graphs.

FORMAT: **DIGITIZE** *keyword*₁ [*keyword*₂]

KEYWORDS:

GRAPH Digitizes a graph. This option expects you to have scaled axes on your graph. You are then instructed to digitize three points and give the graph coordinates for each point. **GRAPH** automatically rotates and scales the digitized points to your graph's coordinate system; thus the orientation of the graph doesn't matter. An optional keyword can be specified to indicate what type of graph you are digitizing. This keyword can be one of the following: **LINEAR**, **LOGX**, **LOGY**, and **LOGXY**. The default is **LINEAR**.

POINTS Digitizes points; points are written to currently open files after being multiplied by scale factors (see **SCALE**) and added to translation factors (see **TRANSLATE**).

SCREEN Digitizes a graph that was drawn on the terminal screen using **GRAPH**. This command puts up a cursor and allows the user to pick points off the graph and write them to a file.

The digitizing tablet to use is chosen by the **TABLET** command. Use the **OPEN** or **APPEND** command before the **DIGITIZE** command to open files to write the digitized points to.

DRAW

Controls the drawing of the digitized points as they are being digitized.

FORMAT: **DRAW** *keyword*

KEYWORDS:

ON Enables drawing of the digitized points; **ECHO** is automatically turned off.

OFF Disables drawing of the digitized points (D).

This option, when enabled, draws vectors between the first and second points, between the second and third points, *etc.* The default is **DRAW OFF**.

ECHO

Controls the printing of the coordinates of digitized points on the screen as they are being digitized.

FORMAT: **ECHO** *keyword*

KEYWORDS:

ON Enables printing of the digitized points on the screen; **DRAW** is automatically turned off (D).

OFF Disables printing of the digitized points on the screen.

The default is **ECHO ON**.

EXIT

Closes all files and exits from the program.

FORMAT: **EXIT**

EXPONENT

Sets the exponent to be labeled on the axes.

FORMAT: **EXPONENT** *keyword amount*

PARAMETERS:

amount The integer number to set the labeled exponent to.

KEYWORDS:

X Sets the *x* exponent to *amount*. If *amount* is the keyword **AUTOMATIC**, then **GRAPH** will choose the exponent; this is the default.

Y Sets the *y* exponent to *amount*. If *amount* is the keyword **AUTOMATIC**, then **GRAPH** will choose the exponent; this is the default.

If *amount* is not an integer, it will be truncated to an integer. This command is only useful if the notation type is scientific.

FONT

Chooses a new software font to be used for axis numbers, labels and titles.

FORMAT: **FONT** *keyword*

KEYWORDS:

ROMAN Enables the roman font (high resolution).

STICK Enables the stick font (low resolution).

The default font is dependent upon the resolution of the device. See Table 1.2 on page 18 for examples of the two available fonts. The roman font is more elaborate but takes longer to draw and is not very aesthetic on low resolution terminals.

HARDCOPY

Makes a hardcopy of the screen if the terminal has a hardcopy device attached and VDI supports this hardcopy device.

FORMAT: **HARDCOPY**

This command is ignored by batch devices and by terminals that do not have a supported hardcopy device attached. This command will not redraw what is currently on the screen; use the **REPLOTT** command for this.

HELP

Gives help on various **GRAPH** commands and features.

FORMAT: **HELP** [*topic*]

PARAMETERS:

topic The topic you want help on. If *topic* is omitted, then a list of topics will be listed.

ID

Prints out an identification of the program.

FORMAT: **ID**

LENGTH

Sets the length of an axis.

FORMAT: **LENGTH** *keyword amount*

PARAMETERS:

amount The new length of the axis in device coordinates.

KEYWORDS:

X Sets the length of *x* axis to *amount*. $0.0 < amount \leq 1.0$

Y Sets the length of *y* axis to *amount*. $0.0 < amount \leq 0.75$

The *x* axis length defaults to 0.75 and the *y* axis length defaults to 0.58.

LOGO

This command serves two functions: it draws a **logo** on the screen and it enables the automatic drawing of logos for every **PLOT** command.

FORMAT: **LOGO** [*keyword*] [*x, y*] [*position*]

PARAMETERS:

position An integer number (1, 2, 3, or 4) specifying where to automatically draw logos for every **PLOT** command. Used in conjunction with the **DEFAULT** keyword. *x, y* are not required.

x, y Draws a logo centered about *x, y*. *position* is not required.

KEYWORDS:

CURSOR Puts the cursor on the screen so the user can input the location of the center of the logo. *x, y* and *position* are not required.

DEFAULT Turns on drawing automatic logos on every plot; *position* is 1, 2, 3 or 4 which are defined as: 1–upper right, 2–upper left, 3–lower left, 4–lower right. *x, y* are not required.

DEVICE Denotes that *x, y* are in device coordinates.

GRAPH Denotes that *x, y* are in graph coordinates. (D)

OFF Turns off drawing automatic logos on plots. *x, y* and *position* are not required.

The size of the logo is set by the command **SIZE LOGO** *amount*. The logo is drawn in the foreground color. The default is not to draw any automatic logos.

LSTYLE

Sets line type for various options.

FORMAT: **LSTYLE** *line style*

PARAMETERS:

line The line of which to change the style. Valid lines are:

CURVE Sets the line style for curves.

MAJGRID Sets the line style for the major grid.

MINGRID Sets the line style for the minor grid.

ZLINE Sets the line style for the zero lines.

style The line style to set *line* to. Valid styles are:

BLANK No line.

DOTDASH Dot dashed line.

DOTTED Dotted line.

LDASH Long dashed line.

MDASH Medium dashed line.

SDASH Short dashed line.

SOLID Solid line line.

Zero lines are lines drawn through 0.0 parallel to the coordinate axes. Due to restrictions in VDI, it is possible that your output device does not support all these line styles. The only guaranteed line styles are **BLANK**, **DOTTED**, and **SOLID**. The defaults are for the curve to be drawn with a solid line, both major and minor grids to be drawn with blank lines (*i.e.* no grids will be drawn), and the zero lines to be drawn with dotted lines. Use the **BLANK** line style to turn off an option.

MINOR

Chooses an option for the minor ticks.

FORMAT: **MINOR** *keyword*

KEYWORDS:

LABEL OFF Turns off labeling minor logarithmic tick marks (D).

LABEL X Turns labeling of minor ticks on for the x logarithmic axis and off for the y logarithmic axis.

LABEL XY Turns on labeling of minor ticks on x and y logarithmic axes.

LABEL Y Turns labeling of minor ticks off for the x logarithmic axis and on for the y logarithmic axis.

The default is **MINOR LABEL OFF**.

OPEN

Creates a new file for writing digitized points to.

FORMAT: **OPEN** *filename*

PARAMETERS:

filename The file you want to open. The default file extension under VMS is .DAT.

ORIGIN

Sets the origin for graphs.

FORMAT: **ORIGIN** x_0, y_0

PARAMETERS:

x_0, y_0 The new origin in device coordinates. $0.0 \leq x_0 \leq 1.0$ and $0.0 \leq y_0 \leq 0.75$.

The origin is the intersection of the x and y axes at the lower left corner and defaults to 0.15,0.076. Note that an origin too close to 0.0,0.0 will not leave enough room for the axes labels and numbers.

OSC

Allows the user to issue a operating system command from within GRAPH.

FORMAT: **OSC** *command*

PARAMETERS:

command Any valid operating system command.

This command allows the user to escape to the operating system to execute *command*. *command* can be an operating system command like DIRECTORY or EDIT. Control is then returned to GRAPH. This command is especially useful in editing command and data files.

PAGE

Erases the screen.

FORMAT: **PAGE**

PATH

Draws a series of vectors in any of the line styles.

FORMAT: **PATH** [*keyword*₁] [*keyword*₂] [*x*₀, *y*₀ *x*₁, *y*₁ . . . *x*₁₀, *y*₁₀]

PARAMETERS:

*x*₀, *y*₀ *x*₁, *y*₁ . . . *x*₁₀, *y*₁₀ The coordinates of the vectors. You can specify up to 10 pairs of coordinates.

KEYWORDS:

CURSOR Puts the cursor on the screen so the user can input the endpoints. The path is terminated by hitting a "Q".

DEVICE Denotes that the endpoints are in device coordinates.

DOTDASH Draw the vectors in dot dashed lines.

DOTTED Draw the vectors in dotted lines.

GRAPH Denotes that the endpoints are in graph coordinates (D).

LDASH Draw the vectors in long dashed lines.

MDASH Draw the vectors in medium dashed lines.

SDASH Draw the vectors in short dashed lines.

SOLID Draw the vectors in solid lines (D).

If no line style is specified, solid lines are used.

PAUSE

Forces the program into a wait state; useful in command files.

FORMAT: **PAUSE**

This command puts the cursor on the screen. To continue, hit any character on the keyboard except the carriage return. This command is ignored by batch devices.

PLOT

Draws axes and plots points.

FORMAT: **PLOT** *filename* [*c_x*, *c_y*] [*p₁*, *p₂*] [*x_l*, *y_l*, *x_u*, *y_u*]

PARAMETERS:

filename The file you want to plot. The default file extension in VMS is .DAT. The format of *filename* is an ASCII file with up to 50 columns of data in it. The maximum number of points per file is 10000.

c_x, *c_y* The columns of data to read out of *filename*. If omitted, they default to 1,2. The maximum number of columns is 50.

p₁, *p₂* The starting and ending positions of the data to read out of *filename*. This is essentially the starting and ending row numbers of the two columns of data you wish to read. If omitted, the default is to read all complete rows of both columns.

x_l, *y_l*, *x_u*, *y_u* The *x* and *y* axis labels and units. If any of these are more than one word, put them in double quotes (").

You can leave off any or all of the labels or units and they will default to a blank. For example:

PLOT URANIUM Time,Distance

will put *x* and *y* axis labels but not units. If you wish to put just units but no labels, you will have to do the following.

PLOT PLUTONIUM "", "",seconds,meters

RECTANGLE

Draws a rectangle in the current foreground color.

FORMAT: **RECTANGLE** [*keyword*₁] [*keyword*₂] [*x*_{ll}, *y*_{ll} *x*_{ur}, *y*_{ur}]

PARAMETERS:

*x*_{ll}, *y*_{ll} The coordinates of the lower left corner of the rectangle.

*x*_{ur}, *y*_{ur} The coordinates of the upper right corner of the rectangle.

KEYWORDS:

CURSOR Puts the cursor on the screen so the user can input the coordinates.

DEVICE Denotes that the coordinates are in device coordinates.

FILL Fills in the rectangle; this is only supported on terminals that have hardware polygon fill, such as most raster devices.

GRAPH Denotes that the coordinates are in graph coordinates (D).

REPLOT

Replots all of the curves plotted since the last **PLOT** command.

FORMAT: **REPLOT** [*keyword*] [*x*_m, *x*_M, *y*_m, *y*_M, *M*_x, *m*_x, *M*_y, *m*_y]

PARAMETERS:

*x*_m, *x*_M, *y*_m, *y*_M, *M*_x, *m*_x, *M*_y, *m*_y The parameters for scaling the axes. *x*_m, *x*_M refer to the minimum and maximum *x* coordinate on the *x* axis. *M*_x, *m*_x refer to the number of major (numbered) ticks and minor ticks (ticks between major ticks) on the *x* axis. The rest of the parameters refer to the *y* axis.

KEYWORDS:

PROMPT Prompts the user for the scaling options. The user can specify the current value by hitting a carriage return. Puts **GRAPH** into **AXIS SCALE USER** mode. *x*_m, *x*_M, *etc.* are not required.

RESCALE Automatically rescales the graph so all curves fit on the graph. Puts **GRAPH** into **AXIS SCALE DIFFERENT** mode. *x*_m, *x*_M, *etc.* are not required.

The number of major and minor intervals is ignored when a logarithmic plot is specified. If no parameters or keywords are specified, then the plot is redrawn as before. There is a maximum of 30 curves that can be replotted. Line styles, colors and symbols on each curve are preserved in the replot. Titles are not redrawn.

RESET

Resets **GRAPH** to the default parameters and erases the screen.

FORMAT: **RESET**

SCALE

Sets scale factors for digitizing points and for plotting points.

FORMAT: **SCALE** [*keyword*] [*n*] *factor*

PARAMETERS:

factor The new scale factor.

n Sets scale factor for column *n*. This means column *n* will be scaled by *factor* for every **PLOT** or **CURVE** command. No keywords are required.

KEYWORDS:

X Sets scale factor in *x* direction to *factor* for use by **DIGITIZE POINTS** command. The *x* coordinate of the digitized point is multiplied by *factor* before being written to all currently open files. *n* is not required.

Y Sets scale factor in *y* direction to *factor* for use by **DIGITIZE POINTS** command. The *y* coordinate of the digitized point is multiplied by *factor* before being written to all currently open files. *n* is not required.

All scale factors default to 1.0. This command works in conjunction with the **TRANSLATE** command so that your data can be scaled and translated. The scaled and translated coordinate is calculated as $x_n = (x \times s) + t$ where x_n is the scaled and translated coordinate, *x* is the coordinate read in from the data file, *s* is the scale factor, and *t* is the translation amount.

SCIENTIFIC

Turns on scientific notation on the axes.

FORMAT: **SCIENTIFIC** *keyword*

KEYWORDS:

OFF Specifies no scientific notation on either axis.

X Enables scientific notation on x axis, no scientific notation on the y axis.

Y Enables scientific notation on y axis, no scientific notation on the x axis.

XY Enables scientific notation on x and y axes (D).

This command affects both linear and logarithmic axes. The default is **SCIENTIFIC XY**.

SET

Sets an option.

FORMAT: **SET** *keyword*

KEYWORDS:

NOVERIFY Disables printing each command in a command (D) procedure as it is being executed.

VERIFY Enables printing each command in a command procedure as it is being executed.

The default is **SET NOVERIFY**.

SIZE

Changes the size of an object.

FORMAT: **SIZE** *keyword amount*

PARAMETERS:

amount The new size.

KEYWORDS:

ARROWHEAD Sets the length of the arrowhead; defaults to 0.02. $0.0 \leq \textit{amount} \leq 1.0$.

BASELINE Sets the amount of white space between lines of software text.; defaults to 1.5. *amount* is a factor by which to multiply the character size to get the linefeed size. $0.0 \leq \textit{amount} \leq 20.0$.

LABELS Sets size of axis labels to *amount*; defaults to 5.0. $0.0 \leq \textit{amount} \leq 20.0$.

LOGO Sets size of the Sandia logo that is drawn with the **LOGO** command to *amount*; defaults to 0.05. $0.0 < \textit{amount} \leq 1.0$.

NUMBERS Sets size of axis numbers to *amount*; defaults to 5.0. $0.0 \leq \textit{amount} \leq 20.0$.

SYMBOLS Sets size of graph symbols to *amount*; defaults to 5.0. $0.0 \leq \textit{amount} \leq 20.0$.

TEXT Sets size of software text drawn with the **TITLE** command to *amount*; defaults to 0.02. $0.0 < \textit{amount} \leq 1.0$.

The label, number, and symbol size are relative. They all depend on the lengths of the *x* and *y* axes; the longer the axes, the larger they are. You can turn off the plotting of the numbers and labels on the axes by specifying the size to be 0.0.

STATUS

Shows current values of GRAPH parameters.

FORMAT: **STATUS** *key*

PARAMETERS:

key An item or set of related items that you want status on.

key is a keyword that indicates which parameter you want the current value of. The keyword may be only a few characters long (*e.g.* **AXIS**). In this case, all keywords that have **AXIS** in them will have their status listed. This makes it easy to get the status on a group of related keywords such as those that give status on colors and sizes. The one exception to this is if the keyword is a single character; all keywords that begin with this character have their status listed in this case.

The available keywords are listed below.

ARROWANGLE	ARROWSIZE	AUTOERASE
AUTOLOGO	AXISCOLOR	AXISSCALE
AXISSIZE	AXISTYPE	AXISWIDTH
BACKGROUND	BASELINE	BELL
CHARANGLE	CURVECOLOR	CURVELINE
CURVEWIDTH	DECIMAL	DRAW
ECHO	EXPONENT	FILES
FONT	FOREGROUND	FRAME
LABELCOLOR	LABELSIZE	LABELWIDTH
LINECOLOR	LOGOSIZE	MAJGRIDLINE
MAJGRIDWIDTH	MAJGRIDCOLOR	MINGRIDCOLOR
MINGRIDLINE	MINGRIDWIDTH	MINORLABEL
NOTATION	NUMBERSIZE	NUMBERWIDTH
NUMBERCOLOR	ORIGIN	SCALING
SYMBOLWIDTH	SYMBOLTYPE	SYMBOLCOLOR
SYMBOLSIZE	TABLET	TEXTANGLE
TEXTSIZE	TEXTWIDTH	TICKCOLOR
TICKWIDTH	TRANSLATE	VECTORWIDTH
VERIFY	YNUMBER	ZLINECOLOR
ZLINETYPE	ZLINEWIDTH	

SYMBOL

Sets symbol type and increment to be used in graphing.

FORMAT: **SYMBOL** *type* [*increment*]

PARAMETERS:

increment The symbol increment; this should be a positive integer. If *increment* is not an integer, it will be truncated to an integer, and if it is negative, it will be made positive. If *increment* is specified, then symbols will be plotted every n^{th} data point. If omitted, it defaults to 1.

type The symbol to use. The available symbols are:

CIRCLE	CROSS	DIAMOND
DOT	SQUARE	TRIANGLE
X		

To turn off symbol plotting, give the command **SYMBOL OFF**; this is the default.

TABLET

Chooses which digitizing tablet to use in the **DIGITIZE** command.

FORMAT: **TABLET** *keyword*

KEYWORDS:

TEK4110 Tektronix 4110 series tablet.

TEK4953 Tektronix 4953 tablet.

TEK4954 Tektronix 4954 tablet (same as **TEK4953**).

The default is **TEK4953**.

TITLE

Draws software text in the current foreground color.

FORMAT: **TITLE** [*keyword*₁] [*keyword*₂] [*x, y*] *text*

PARAMETERS:

x, y The *x* and *y* coordinate used to place the title. If omitted, the title is centered above the plot.

text The text to draw.

KEYWORDS:

CENTER Centers the text about the specified coordinates. The text is centered in both the *x* and *y* directions.

CURSOR Uses the cursor to position the text.

DEVICE Indicates that the specified coordinates are in device coordinates.

GRAPH Indicates that the specified coordinates are in graph coordinates (D).

NEXT Indicates that the text is to be positioned under the last plotted software text. Omit *x, y* in this case.

If *text* is longer than one word or is a number, enclose it in double quotes. You can set the size of the text with the **SIZE** command. You can choose from two different fonts with the **FONT** command. A number of escape sequences are available to do super- and sub-scripting, greek characters, and special symbols. See Section 1.3 on page 16 for information on escape sequences.

TRANSLATE

Sets the translation factors for digitizing points or for plotting points.

FORMAT: **TRANSLATE** [*keyword*] [*n*] *amount*

PARAMETERS:

amount The new translation amount.

n Sets translation amount for column *n* to *amount*. This means column *n* will be translated by *amount* for every **PLOT** or **CURVE** command. No keywords are required.

KEYWORDS:

X Sets translation amount in *x* direction to *amount* for use by **DIGITIZE POINTS** command. *amount* is added to the *x* coordinate of the digitized point before being written to all currently open files. *n* is not required.

Y Sets translation amount in *y* direction to *amount* for use by **DIGITIZE POINTS** command. *amount* is added to the *y* coordinate of the digitized point before being written to all currently open files. *n* is not required.

All translations default to 0.0. This command works in conjunction with the **SCALE** command so that your data can be scaled and translated. The scaled and translated coordinate is calculated as $x_n = (x \times s) + t$ where x_n is the scaled and translated coordinate, x is the coordinate read in from the data file, s is the scale factor, and t is the translation amount.

TYPE

Types a file or a text string on the terminal.

FORMAT: **TYPE** [*keyword*] *string*

PARAMETERS:

string The file or text string you want to type. If *string* is a filename, the format of the file is ASCII and, under VMS, the default file extension is .DAT.

KEYWORDS:

TEXT If this keyword is specified, then *string* is assumed not to be a filename but a text string to be typed out on the terminal screen. If *string* is longer than one word or is a number, enclose it in double quotes. This command is useful for such things as erasing the alpha screen as you can make *string* an escape sequence that the terminal understands.

If no keywords are present, *string* is assumed to be the name of a file that is to be typed out on the terminal screen.

VECTOR

Draws a vector.

FORMAT: **VECTOR** [*keyword*₁] [*keyword*₂] [*x*₀, *y*₀ *x*₁, *y*₁]

PARAMETERS:

*x*₀, *y*₀ *x*₁, *y*₁ The coordinates of the vector.

KEYWORDS:

CURSOR Puts the cursor on the screen so the user can input the endpoints.

DEVICE Denotes that the endpoints are in device coordinates.

DOTDASH Draw the vector in a dot dashed line.

DOTTED Draw the vector in a dotted line.

GRAPH Denotes that the endpoints are in graph coordinates (D).

LDASH Draw the vector in a long dashed line.

MDASH Draw the vector in a medium dashed line.

SDASH Draw the vector in a short dashed line.

SOLID Draw the vector in a solid line (D).

WAIT

Waits for a specified period of time.

FORMAT: **WAIT** *time*

PARAMETERS:

time The time to wait, specified as *hr:mi:se.cc*. Here *hr* is hours, *mi* is minutes, and *se.cc* is seconds.

WIDTH

Changes the pen width of an option.

FORMAT: **WIDTH** *keyword amount*

PARAMETERS:

amount The new pen width. $0.0 < amount \leq 10.0$.

KEYWORDS:

AXIS Sets the pen width used to draw the axes.

CURVES Sets the pen width used to draw the curve.

LABELS Sets the pen width used to draw the labels.

MAJGRID Sets the pen width used to draw the major grid.

MINGRID Sets the pen width used to draw the minor grid.

NUMBERS Sets the pen width used to draw the numbers.

SYMBOLS Sets the pen width used to draw the symbols.

TEXT Sets the pen width used to draw text.

TICKS Sets the pen width used to draw axis ticks.

VECTORS Sets the pen width used to draw vectors.

ZLINE Sets the pen width used to draw zero lines.

Zero lines are lines drawn through 0.0 parallel to the coordinate axes. The pen width is selectable only on a few devices like the laser printers. This command does not work on raster devices. All pen widths default to 1.6.

WRITE

Writes the last plotted points to a file.

FORMAT: **WRITE** *filename* [*keyword*]

PARAMETERS:

filename The file you want to write. Under VMS, the default file extension is .DAT. The format of *filename* will be ASCII.

KEYWORDS:

CLIP Clips the points written to the file against the axis limits. If this keyword is not there, all of the data points in the last plotted file are written, whether they were drawn or not.

The points last plotted with the **PLOT** or **CURVE** command will be written to *filename* in two column format. The points are written as scaled and translated by the **SCALE** and **TRANSLATE** command.

YNUMBER

Sets the plotting of the numbers on the *y* axis to be either horizontal or vertical.

FORMAT: **YNUMBER** *keyword*

KEYWORDS:

HORIZONTAL Makes the *y* axis numbers horizontal (D).

VERTICAL Makes the *y* axis numbers vertical. Numbers near the origin may overlap.

The default is for the *y* numbers to be **HORIZONTAL**.

2.1 Program Defaults

As a quick reference, the defaults upon program initialization are listed below. This may also be seen by giving the command **HELP DEFAULTS**. The current values can be seen by using the **STATUS** command.

2.1.1 Miscellaneous Parameters

Parameter	Default Value
VERIFY	OFF

2.1.2 Graphing Parameters

Parameter	Default Value
ARROWHEAD ANGLE	15
ARROWHEAD SIZE	.02
AUTOERASE	ON
AXIS SCALE	DIFFERENT XY
AXIS TYPE	LINEAR XY
BASELINE FACTOR	1.5
CHARACTER ANGLE	0
COLORS	WHITE (background color is BLACK)
CURVE LINE STYLE	SOLID
EXPONENT	both x and y exponents are AUTOMATIC
FRAME TYPE	FULL
GRID LINE STYLES	both major and minor are BLANK
LABEL SIZE	5.
LOGO	no automatic logo
LOGO SIZE	.05
NOTATION	SCIENTIFIC XY
NUMBER SIZE	5.
ORIGIN	.15,.076
PEN WIDTHS	1.6
SYMBOL SIZE	5.
SYMBOL TYPE	OFF
TEXT ANGLE	0
TEXT SIZE	.02
X,Y AXIS LENGTHS	.75,.58
X and Y DECIMAL PLACES	1
YNUMBERS	HORIZONTAL
ZLINE LINE STYLE	DOTTED

2.1.3 Digitizing Parameters

Parameter	Default Value
BELL	OFF
DRAW	OFF
ECHO	ON
SCALE X and Y	1.
TRANSLATE X and Y	0.0
TABLET	TEK4953

Chapter 3

Examples

As an aid to first time **GRAPH** users, examples are included here. They cover both digitizing and plotting graphs. **GRAPH** prompts and responses are in typewriter font and user commands are in san serif. Comments are in parenthesis.

3.1 Example 1—Digitizing a Graph

In the following example, it is assumed that the user has a scaled graph of some data that he wishes to digitize. It is best that he tape this graph onto the digitizing tablet to minimize movement which could cause errors. The user tells **GRAPH** to ring the terminal bell for each digitized point and then opens a data file called SHOT23.DAT (the .DAT is assumed). **GRAPH** prompts the user to digitize the origin and a point on each axis and to enter the graph coordinates for each point. This enables **GRAPH** to scale the digitized points to the graph's coordinate system. Note that this method makes the orientation of the graph with respect to the digitizing tablet irrelevant. The user then digitizes as many points as he wishes and terminates digitizing by hitting two carriage returns on the keyboard. The opened file is then closed.

```

GRAPH> bell on
GRAPH> open shot23
GRAPH> digitize graph
    _Digitize origin of graph
    Hit two carriage returns to use the same values as last time
    (User digitizes origin of graph at this point)
    _Enter graph coordinates of point just digitized: 0..0.
    _Digitize a point on the x axis
    (User digitizes a point on the x-axis at this point)
    _Enter graph coordinates of point just digitized: 10..0.
    _Digitize a point on the y axis
    (User digitizes a point on the y-axis at this point)
    _Enter graph coordinates of point just digitized: 0..100.
    _Begin digitizing points
    Hit two carriage returns when you are finished
    (User digitizes as many points as he desires and
    then hits two carriage returns)
GRAPH> close shot23

```

3.2 Example 2—Digitizing Points

The following example shows a user digitizing points from the tablet. The user opens a file called SSX.DAT and then tells GRAPH to scale in the x direction by 3.2 and in the y direction by 5.6. Also, GRAPH is instructed to translate in the x direction by -12.0 and in the y direction by 3.5. Remember that the digitizing tablet's maximum x - and y -coordinate is 1.0. So, a point digitized as $(.34, .56)$ would actually be written to all open files as $((.34 \times 3.2 + (-12.0)), (5.6 \times .56 + 3.5))$, or $(-10.91, 6.64)$. The opened file is then closed.

```

GRAPH> open sxx.dat
GRAPH> scale x 3.2
GRAPH> scale y 5.6
GRAPH> translate x -12
GRAPH> translate y 3.5
GRAPH> digitize points
    _Begin digitizing points
    Hit two carriage returns on the keyboard when finished
    (User digitizes as many points as he desires and
    then hits two carriage returns)
GRAPH> close sxx.dat

```

3.3 Example 3—Plotting Graphs

In the following example, the user plots a graph of columns 1 and 2 in the file TESTDATA.DAT, using all of the defaults. The output from this command is shown in Figure 3.1 on page 55.

```
GRAPH> plot testdata X,Y
```

3.4 Example 4—Plotting Graphs with Changes in Default Settings

In the following example, the user specifies the device origin of his plot, specifies the x and y axes lengths, sets the minor grid lines to be dotted, sets normal notation for both axes, chooses logarithmic axes, chooses minor labeling of the logarithmic axes and then plots columns 3 and 6 from the file TEST35.DAT. The output from this command is shown in Figure 3.2 on page 56.

```
GRAPH> origin .12..1
GRAPH> length x .7
GRAPH> length y .6
GRAPH> lstyle mingrid dotted
GRAPH> scientific off
GRAPH> axis log xy
GRAPH> minor label xy
GRAPH> axis scale user .4,2.0,10,200,1,1,1,1
GRAPH> plot test35 3,6 Time,"Normal Velocity",seconds,"Meters/second"
```

3.5 Example 5—Plotting Graphs with User Scaling

In the following example, the user chooses his own axis scaling as well as changes a number of graphing parameters. The output from these commands is shown in Figure 3.3 on page 57. Note the use of the escape sequences in the **TITLE** commands.

```
GRAPH> axis nofull
GRAPH> decimal x 0
GRAPH> decimal y 0
GRAPH> scientific off
GRAPH> font stick
GRAPH> origin .1,.05
GRAPH> length x .55
GRAPH> length y .55
GRAPH> logo default 1
GRAPH> symbol square
GRAPH> size symbol 7
GRAPH> axis scale exact 2.5,13.5,3.2,8.5,215,20,20,4
GRAPH> plot shot12 Time,"Peak Pressure", "\gr m\eng seconds", "Newtons/m\^2"
GRAPH> lstyle curve dotted
GRAPH> symbol off
GRAPH> curve shot12e
GRAPH> size text .015
GRAPH> title graph 3,200 "\dline Theoretical"
GRAPH> title next "\sline Experimental"
GRAPH> title next "\sq - Experimental Data Points"
GRAPH> size text .025
GRAPH> title "Pressure vs Time"
```

Figure 3.1: Default Plot

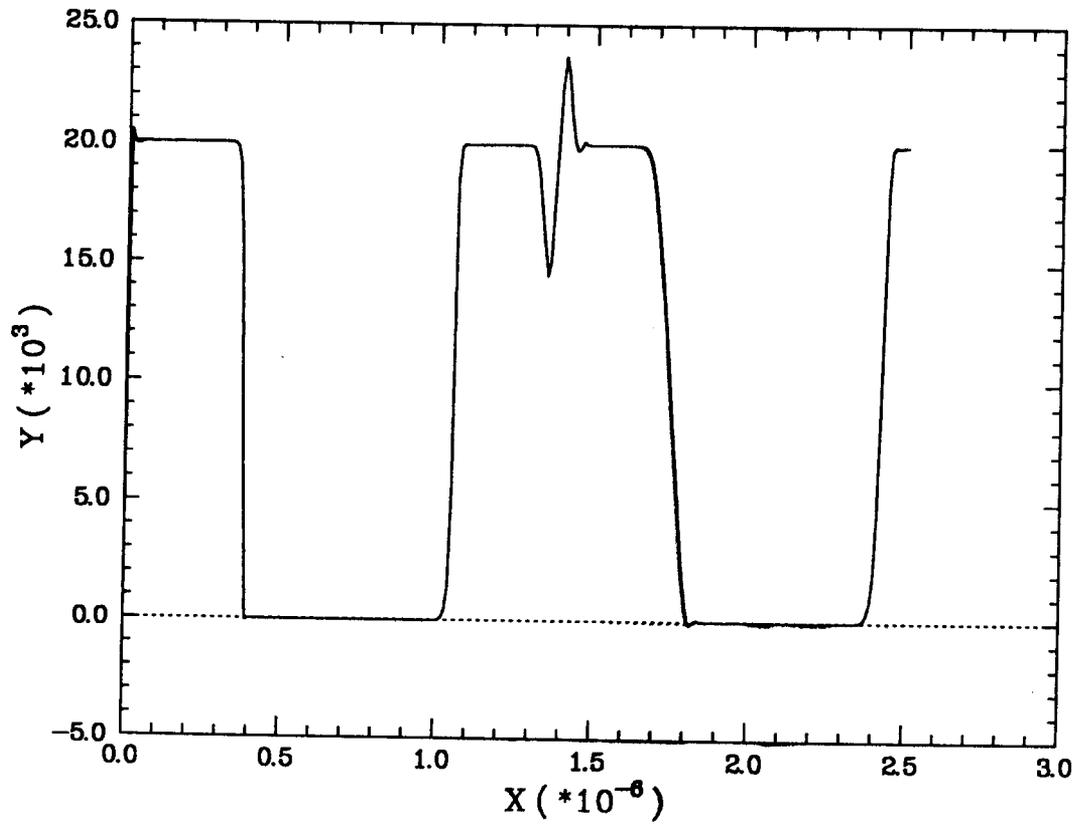


Figure 3.2: Plot with Changes in Default Settings

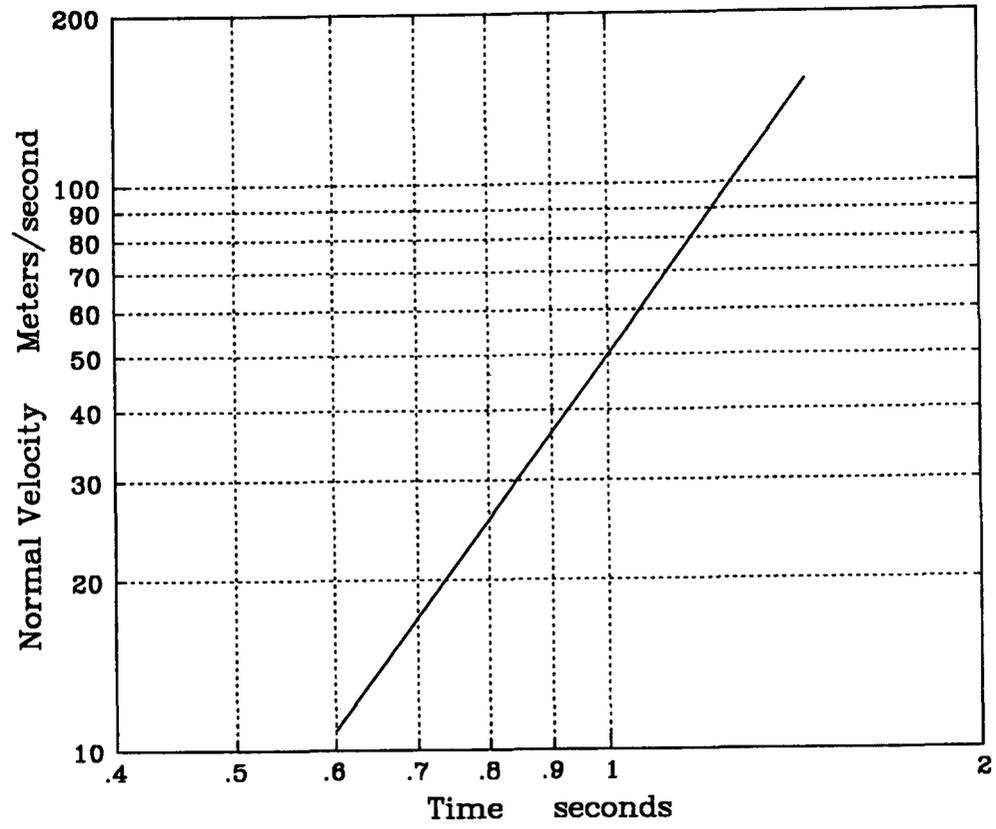
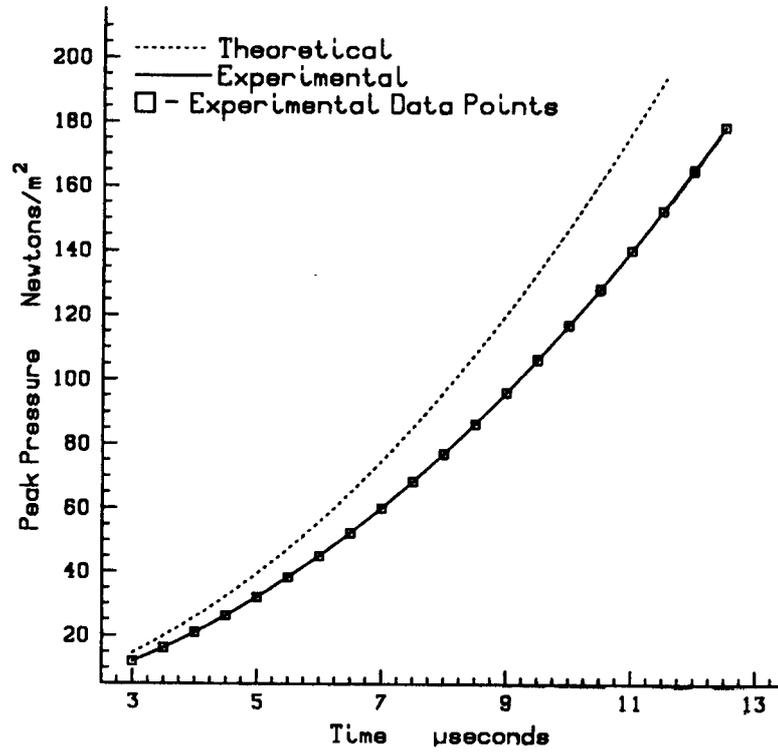


Figure 3.3: Plot with User Scaling

Pressure vs Time



Appendix A

Obsolete Commands

Following is a list of obsolete commands and the commands that replaced them. The obsolete commands continue to work. However, you should modify your command files and start using the new commands as there are no guarantees as to how long the obsolete commands will continue to work.

- The **AXIS NORMAL** command has been replaced by the **AXIS LINEAR** command.
- The **COLOR GRID** command has been replaced by the **COLOR MAJGRID** and the **COLOR MINGRID** commands.
- The **COLOR TITLE** command has been replaced by the **COLOR FOREGROUND** command.
- The **DCL** command has been replaced by the **OSC** command.
- The **NORMAL** keyword on the **DIGITIZE** command has been replaced by **LINEAR**.
- The **GRID** command has been replaced by the **LSTYLE** command.
- The **LINE** command has been replaced by the **LSTYLE** command.
- The **LIST** command has been replaced by the **STATUS FILES** command.
- The **MINOR LABEL** command has been reworked. The only remaining options are **OFF**, **X**, **XY**, and **Y**.
- The **SIZE TITLE** command has been replaced by the **SIZE TEXT** command.
- The **TEXT** command has been replaced by the **TYPE TEXT** command.
- The **ZLINE** command has been replaced by the **LSTYLE** command.

References

1. C. B. Selleck and M. P. Sears, *PLT—A Multipurpose Plotting Package*, Sandia National Laboratories Report, SAND83-0637 (1983).
2. K. M. Erickson and R. W. Simons, *Functional Specification of the Sandia Virtual Device Interface (SVDI)*, Sandia National Laboratories Report, SAND81-1900 (1981).

**Unlimited Release
Distribution:**

Applied Research Associates
4300 San Mateo NE
Suite A220
Albuquerque, NM 87110
Attn: J. C. Partch

Hines/Loyola Department of Radiotherapy
Mail Drop 114B
Hines VA Hospital
Hines, IL 60141
Attn: T. R. Canfield

Ktech Corporation
901 Pennsylvania Av. NE
Albuquerque, NM 87110
Attn: E. Young

Martin Marietta
P. O. Box 179
Mail Stop D6130
Denver, CO 80201
Attn: B. Easom

Microelectronics and Computer
Technology Corporation
9430 Research Blvd.
Eschelon Building # 1, Suite 200
Austin, TX 78759-6509
Attn: G. R. Siebers

RE/SPEC Incorporated (10)
P. O. Box 14984
Albuquerque, NM 87191
Attn: M. L. Blanford

Sandia Internal

1510 J. W. Nunziato
1520 D. J. McCloskey
1523 J. F. Holland

1530 L. W. Davison
1531 S. L. Thompson
1531 T. K. Bergstresser
1531 T. J. Burns
1531 C. M. Korbin
1531 H. S. Lauson
1531 J. S. Rottler
1531 C. B. Selleck (25)
1531 J. W. Swegle
1531 F. J. Zeigler
1533 P. Yarrington
1533 W. T. Brown
1533 W. R. Davey
1533 A. V. Farnsworth
1533 G. I. Kerley
1533 M. E. Kipp
1533 S. T. Montgomery
1533 F. R. Norwood
1533 S. L. Passman
1533 A. C. Robinson
1533 D. M. Webb
1534 J. R. Asay
1534 L. M. Barker
1534 L. C. Chhabildas
1534 D. S. Drumheller
1534 J. E. Dunn
1534 D. E. Grady
1534 T. G. Trucano
1534 J. L. Wise
1540 W. C. Luth
1541 C. R. Carrigan
1541 G. J. Elbring
1541 J. B. Rundle
1541 M. C. Walck
1542 B. M. Butcher
1542 L. S. Costin
1542 W. H. Gerstle
1542 D. J. Holcomb
1542 A. K. Jones
1542 W. A. Olsson
1542 R. H. Price
1542 L. W. Teufel
1542 W. R. Wawersik
2112 C. F. Diegert

2614 A. F. Schkade
2614 G. C. Shepherd
2644 R. E. Jones
6258 T. C. Bickel
6258 B. J. Thorne
6322 R. S. Longenbaugh
6332 T. M. Torres
6332 J. G. Arguello
7112 C. W. Smith
7125 H. C. Ogden
8243 D. J. Bammann
3141 S. A. Landenberger (5)
3151 W. A. Garner (3)
3154-3 C. Dalin (28)
(for DOE/OSTI)
8024 P. W. Dean (1)