

SLAC COMPUTATION GROUP  
Stanford, California

CGTM No. 208  
September 1982

**CalComp Graphics**

**An Interface to the  
Unified Graphics System  
with CalComp Calling Sequences**

**Roger B. Chaffee  
Computation Research Group  
Stanford Linear Accelerator Center**

**This is a Working Paper  
Do not quote, cite, abstract,  
or reproduce without prior  
permission of the author.**

## CONTENTS

<u>Chapter</u>	<u>page</u>
1. INTRODUCTION . . . . .	1
2. CALLING SEQUENCES . . . . .	2
PLOT . . . . .	2
PLOTS . . . . .	3
FACTOR . . . . .	3
WHERE . . . . .	4
NEWPAG . . . . .	4
NEWPEN . . . . .	4
SYMBOL . . . . .	5
NUMBER . . . . .	5
SCALE, AXIS, LINE . . . . .	5
3. USE IT . . . . .	6

## Chapter 1

### INTRODUCTION

The Unified Graphics System is very flexible and well-supported at SLAC, but its flexibility and completeness result in a complexity that makes it difficult to use. The graphics package distributed by California Computer Products in the late 1960's, to drive the CalComp pen plotters, is limited but adequate for simple, non-interactive plots. The calling sequences have been copied by other companies, e.g. Versatec in their VersaPlot software, and provide what is in effect a de facto standard for graphics packages.

I have written a short package which has similar calling sequences, and which calls the Unified Graphics System to perform the actual graphic operations. There are two uses I envision for the package:

First, it is suitable for use in applications which need a general package, but don't need all the bells and whistles of U.G.

Second, it is ideal for existing programs which call routines in the CalComp form, and which would otherwise need to be converted in order to run at SLAC.

## Chapter 2

### CALLING SEQUENCES

These subroutines are described in many places. The specifications for this package were taken from 'Programming CalComp Pen Plotters', dated September 1969, which I found in the consulting office. I would be happy to examine a more recent publication, and perhaps change this package.

The fact that the routines are still useful after more than a decade speaks well for the CalComp company, and indicates how little progress we have made in the area of graphics software which works on more than one manufacturer's mainframe and display devices.

All dimensions for x- and y-values are inches. See SUBROUTINE FACTOR if you don't like inches.

#### 2.1 PLOT

The calling sequence is:

```
CALL PLOT(X,Y,LPEN)
```

X,Y            coordinates in inches from the current origin.

LPEN           controls pen up/down status and origin definition.

LPEN=2        DRAW (pen down)

LPEN=3        MOVE (pen up)

LPEN=-2       Draw or move,    and redefine the origin    to be the  
LPEN=-3       new locaton.

LPEN=998      End of plot.    (This doesn't clear the screen,  
              which is nice for Tektronix 4013 plots.)

LPEN=999      End of all plotting.

## 2.2 PLOTS

The calling sequence is:

```
CALL PLOTS(I,J,X)
```

PLOTS is used for initialization and termination. It is different for every CalComp style package I have seen, and this one is no exception. It must be called only once, before any call to PLOT, SYMBOL, NUMBER, AXIS, or LINE, and once again, when all plotting is finished.

I=998 End of plot, as for subroutine PLOT.

I=999 End of plotting.

I=1 Tektronix 4013 output. (U.G. device DWT4013)

I=2 Versatec output to a disk file. To get hard copy,  
CP SPOOL PRINTER CLASS P  
PRINT CALCOMP VERSATEC A (CC)

I=3 Versatec output to a disk file, suitable for ROLL  
form. To get hard copy,  
CP SPOOL PRINTER CLASS P FORM ROLL  
PRINT CALCOMP VERSATEC A (CC)

Don't forget to reset your printer:  
CP SPOOL PRINTER CLASS A FORM STANDARD

J,X Not used. Not even examined.

## 2.3 FACTOR

Calling sequence:

```
CALL FACTOR(FACT)
```

This routine resets the units. E.g. to specify everything in centimeters instead of inches, CALL FACTOR(0.3937).

## 2.4 WHERE

Calling sequence:

```
CALL WHERE(X,Y,FACT)
```

This routine returns the current pen position and scaling factor.

Due to limitations in communication between the Unified Graphics System and a calling program, this routine does not return correct values after a call to SYMBOL, which is, of course, just when you want it most. If the need is there, this package could be modified to handle the interaction between SYMBOL and WHERE.

## 2.5 NEWPAG

The calling sequence is:

```
CALL NEWPAG
```

It clears the screen or moves to a new page on the plotter.

## 2.6 NEWPEN

The calling sequence is:

```
CALL NEWPEN(NP)
```

NP=1     Solid lines.

NP=2     Dots.

NP=3     Dashes

NP=4     Dot-dash

## 2.7 SYMBOL

The calling sequence is:

```
CALL SYMBOL(X,Y,SIZE,TEXT,ANGLE,NCHAR)
```

X,Y position of the first character

SIZE character size, in inches or current units.

TEXT Hollerith text to be plotted.

ANGLE in degrees, measured ccw from the x-axis

NCHAR number of characters in TEXT

Variations using NCHAR=0 and NCHAR<0 are defined by Cal-Comp, and could be supported if the need were there.

## 2.8 NUMBER

The calling sequence is:

```
CALL NUMBER(X,Y,SIZE,VALUE,ANGLE,NDEC)
```

X,Y,SIZE,ANGLE are as in SYMBOL.

VALUE is the value to be written.

NDEC controls the format. It is the number of digits after the decimal point. Zero and negative values do the appropriate thing.

## 2.9 SCALE, AXIS, LINE

As defined by CalComp.

These are "higher-level" subroutines, which in turn call PLOT and SYMBOL to do any plotting. Although they are included in this package, you should probably not use them. If your application uses them, then it is doing things which are easier to do with Top Drawer.

## Chapter 3

### USE IT

This package currently lives on the W-disk, and it invokes U.G. routines, and it invokes the local routine KMAND. To make the necessary routines available to the loader, type the following script just once, before you first try to run your program:

```
GIME WDISK
GIME RCB 192
GLOBAL TXTLIB CALCOMP UGFTNLIB FORTSLAC FORTMOD2
```

Isn't VM fun?