1 Overview Of Graphics Capabilities in SAS® Software

Laying the Foundation

1.1 Introduction

Several SAS software products contain procedures capable of generating plots. This chapter gives you some basic information about base SAS and SAS/GRAPH graphics capabilities.

While SAS/GRAPH software contains the tools and flexibility to produce high-resolution, presentation quality graphs, the capabilities of the procedures in the base product are not nearly as sophisticated. Nevertheless, graphs produced with the base product have several advantages, and the graphics procedures from both products deserve consideration.

1.2 Base Product Graphics

With the exception of SAS/GRAPH, SAS/INSIGHT, SAS/LAB, SAS/OR, and SAS/QC software, the plots and charts produced by SAS are designed to be printed on a line or impact printer and, therefore, have low resolution. This can be an advantage when high-resolution plots are not required, because plots generated through the base product are

generally easy and quick to produce, do not require a graphic-capable printer, can be printed faster, and can be stored in smaller, more easily accessed files.
The plots and charts generated from procedures in products such as base SAS typically have from 80 to 132 horizontal plot positions and from 24 to 60 vertical positions. These low-resolution plots are often unable to draw straight or curved lines or round circles, and you have very little control over font and color selection.

You can use these procedures in the base product to generate plots and graphs of data:

CHART	produces histograms and pie, block, and star charts.
PLOT	is used to generate scatter plots of two or more variables.
TIMEPLOT	creates observation-specific plots of sequence data.
UNIVARIATE	produces probability, stem-and-leaf, box, and schematic plots.
CALENDAR	displays data from a SAS data set in the form of a monthly calendar.

Although base product graphs are not intended to serve as report-quality graphs, they do have distinct advantages and are discussed in more detail in Chapter 2, "Plotting without SAS/GRAPH Software."

1.3 SAS/GRAPH Graphics

SAS/GRAPH provides you the flexibility to generate custom-designed, presentation-quality graphs. This flexibility, of course, adds complexity both in programming and in hardware/software requirements. Once mastered, the capabilities of the system are extensive.

The generation of graphs using SAS/GRAPH may increase the time to create the SAS code and will almost certainly increase the production time of individual graphs and charts. There are two major advantages of graphs created by

SAS/GRAPH that justify this investment of time. First you can customize graphs created by SAS/GRAPH to convey specific information using almost any format and color scheme allowed by your hardware. Second, your program can be written to allow the attributes of the graph to depend on the data. Graphs generated with SAS/GRAPH can include all of the necessary elements required for presentation-quality graphs. It is possible to manipulate all aspects of the graph to create graphs that will suit any presentation.

The term presentation quality implies that the developer has control over most, if not all, aspects of the setup of the graph. SAS/GRAPH allows each of the following:

- FONT creation, control, and selection
- TITLE placement and attributes
- LINE, PATTERN, and SYMBOL selection, placement, and attributes
- AXIS and LEGEND design, placement, and attributes
- graphic page layout design and control.

Unlike many other graphics packages, SAS/GRAPH can be run in batch mode with the data defining many of the graphic's attributes. This is a major strength of SAS/GRAPH. Several graphics packages enable the user to create very nice graphs, one at a time. It is not unusual to need a series of graphs where all are similar, but each is based on slightly different data. Creating such a series of graphs is easy in SAS/GRAPH and is impossible in many other packages.

The data can define the titles and axes as well as the plot symbols. The ANNOTATE facility, an integral portion of SAS/GRAPH that is not discussed in this book, is especially useful when the programmer wishes to base the symbols, labels, and titles on parameter values contained in the data.

1.4 Graphics Options for SAS/GRAPH

The graphics environment is set up and controlled by graphics options in much the same way as options are used in

the base product. SAS/GRAPH options, GOPTIONS, are described in *SAS/GRAPH Software: Reference, Version 6, First Edition, Volume 1* and *Volume 2* and to some extent in *SAS/GRAPH Software: Usage, Version 6, First Edition*.

Most of the figures in this book were generated using DEVICE=HP7475A. Unless you use the same device, you need to change this option when you execute the programs in Appendix B. Remember, we used Release 6.08 of the SAS System under the Windows environment to generate and test the examples in this book. Therefore the form of the file name designation may be somewhat different for your system. Also, if the graph is sent directly to the plotter the FILENAME statement and the GSFNAME= and GSFMODE= options may not be needed.

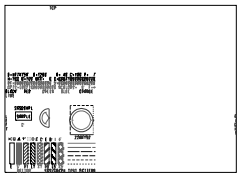
The options presented in the following sections are often either under utilized or used improperly.

MORE INFORMATION:

SAS/GRAPH Software: Reference, pp 291-302 - The GOPTIONS statement is covered in Chapter 12.

SAS/GRAPH Software: Usage - Various GOPTIONS are used in examples throughout the book.

1.4.1 Checking the GOPTIONS using the GTESTIT procedure



The current settings for the graphics options, especially those that depend on the current graphics device, are often not immediately obvious. You may need to determine the current settings and their effect on the graphs produced on the specified device. Because you need to determine these values for several of the examples in this book, it is fortunate that you can use the GTESTIT procedure to quickly check the current GOPTION settings and their effects.

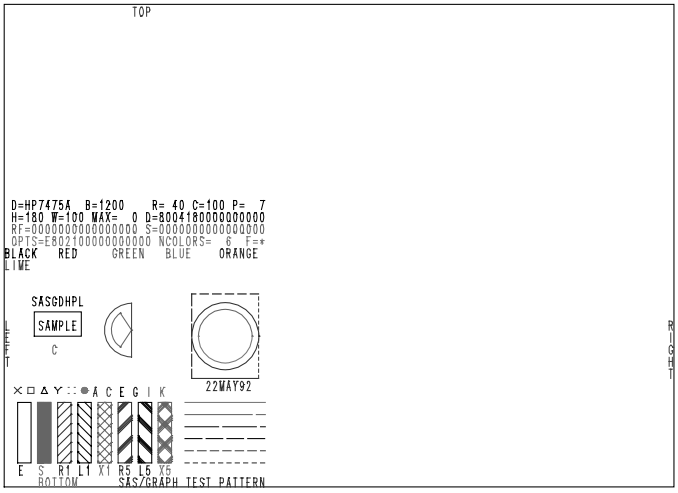
The following code, which produces Figure 1.4.1a, uses PROC GTESTIT to demonstrate the effects of the current set of graphics options:

```
GOPTIONS DEVICE=hp7475a;

proc gtestit pic=1;
run;
```

The specific results of PROC GTESTIT depend on the graphics device as well as the system options currently in effect. Most of the figures in this book were generated using DEVICE=HP7475A; your PROC GTESTIT results will differ from these, unless you use the same driver.

Figure 1.4.1a is the first of three possible graphs that can be generated using PROC GTESTIT. All of the figures produced for this book are reproduced in black and white even though the original code may have produced a graph containing color. Some of these colors do not show up when they are converted to black and white graphs. Consequently, your graphs may vary
from some of the figures shown.



Much of the same information that is shown in the graph produced by GTESTIT is also written to the log (Figure 1.4.1b).

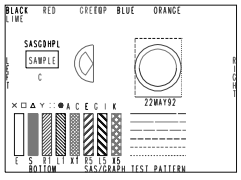
```
D=HP7475A  B=1200 R= 40 C=100 P=7
H=180 W=100 MAX=  0 D=8004180000000000
RF=0000000000000000 S=0000000000000000
OPTS=E802100000000000 NCOLORS=  6
Background color = WHITE
Color 1 = BLACK
Color 2 = RED
Color 3 = GREEN
Color 4 = BLUE
Color 5 = ORANGE
Color 6 = LIME
Ratio = 0.71984
Hsize =  9.8424
Vsize =   7.085
F=10
```

It is important to understand the relationship between the GTESTIT values shown in Figures 1.4.1a and 1.4.1b, and selected GOPTIONS. These relationships are detailed in *SAS/GRAPH Software: Reference* in the chapter on GTESTIT, and some of these values are also discussed in the following sections in this book.

MORE INFORMATION:

SAS/GRAPH Software: Reference, Chapter 38 - Discussion and reference for PROC GTESTIT.

1.4.2 Choosing between HSIZE & VSIZE and HPOS & VPOS



Several factors determine the size of a graph and the size of the letters and numbers on it. Any given device driver has a maximum vertical and horizontal size that is stored as part of that driver's definition (which is controlled through PROC GDEVICE).

Given the hardware constraints, you can also define a plot size. Usually, the plot size is the same as the hardware maximum. However, the size can be changed by using the HSIZE (horizontal size) and VSIZE (vertical size) graphics options. The units for these options are usually inches but can also be centimeters.
The default size for letters and numbers (h=1) is based on cell size, which is a factor of the number of vertical and horizontal cell positions available for the given HSIZE and VSIZE options. The default varies for different devices but can be changed by using the graphics options HPOS (number of cells horizontally) and VPOS (number of cells vertically).

The default sizes are usually sufficient, but an understanding of how and when to change these options can be useful. Resizing a graph becomes important when

- a smaller physical size is desired
- graphs are truncated when imported into a word processor
- the plot device has insufficient memory to plot the entire graph.

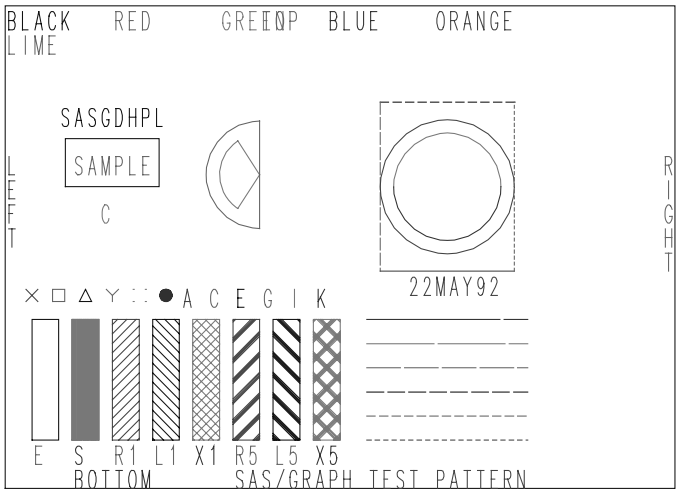
Sizing a graph to fit within specified constraints is common. When HSIZE and VSIZE are reduced and the number of character positions (VPOS and HPOS) stays constant, the individual size of a character cell is reduced (h=1 creates a smaller character). For a constant HSIZE and VSIZE, the cell size can then be increased by decreasing the number of vertical and horizontal positions (VPOS and HPOS). Although it is usually more efficient to increase the size of letters and symbols by using the HTEXT= option or the HEIGHT= option

in the TITLE, FOOTNOTE, AXIS, and LABEL statements, there are times when it is necessary to make changes in these options.

The PROC GTESTIT results in the previous section (Figures 1.4.1a and 1.4.1b) show the default values for the HP7475A device. VPOS is the number of rows and is signified by R (in this case 40). The number of columns, HPOS, is designated by C (in this case 100).

You can double the cell size by halving the number of positions (cells) in both directions, as shown in the following GOPTIONS statement:

```
goptions hpos=50 vpos=20;
proc gtestit pic=1;
run;
```



The resulting graph (Figure 1.4.2) contains most of the same information as the previous GTESTIT. However, the letters and figures are larger, and some information is lost. When changing the cell size by using either HSIZE and VSIZE or HPOS and VPOS, it is usually

advisable to change the vertical and horizontal options proportionately. If the change is not proportional, the cell becomes distorted and characters formed by hardware or simulated hardware fonts can be shaped incorrectly.

MORE INFORMATION:

SAS/GRAPH Software: Reference, pp 25-26 - Cell size is discussed further in the section ‖Defining the Graphics Output Area.=

SAS/GRAPH Software: Usage, pp 678-679 - Example and discussion of the use of HPOS= and VPOS=.

SEE ALSO:

Section 1.4.3 - Uses increased HPOS and VPOS values to fit a large block chart.

1.4.3 Using HPOS and VPOS

Increasing the HPOS and VPOS values decreases the cell size and effectively creates more space to create a graph by decreasing the size of each individual character and symbol. When fitting a block chart with a large number of cells or a scatter plot with one or both axes controlled by the ORDER= option with several levels, it is not unusual to receive an error message such as ‖vertical

axis cannot be fit as specified.= This occurs when more vertical (or horizontal) character positions are needed than are available.

The following code attempts to create a block chart. The variable MONTH has 12 values, which creates a wide chart.

```
proc gchart data=vol1.ca88air;
title1 h=4 'Block Chart of Average Carbon Monoxide';
footnote j=l h=4 f=simplex 'Figure 1.4.3a';
block month / sumvar=co type=mean
              group=station discrete;
run;
```

The block chart is not created, and the following error message appears in the log.

```
ERROR: A BLOCK chart for MONTH is not possible. At least 43 vertical
       positions and 100 horizontal positions are needed. The device is
       currently defined with 40 vertical positions and 100 horizontal
       positions.
       More positions may be needed if any titles, footnotes, or parts of
       the legend are specified in inches, centimeters, or percent.
```

Increasing the HPOS and VPOS values corrects the problem. The following GOPTIONS statement doubles both of these values.

```
goptions vpos=80 hpos=200 htext=2;
```

This produces the following figure:

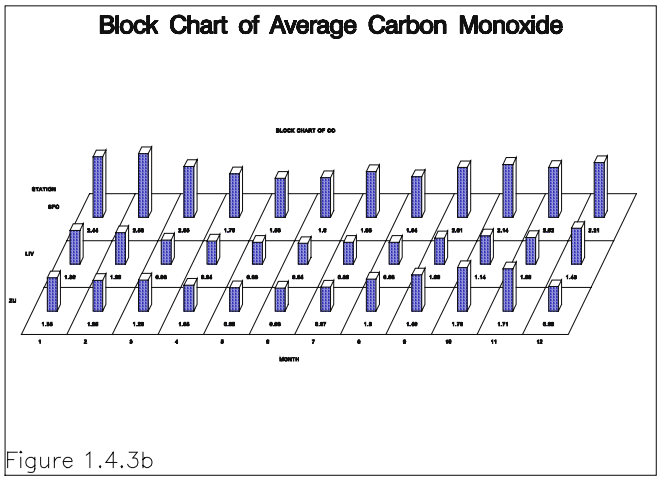


Figure 1.4.3b

The HTEXT=2 option increases the default text size to compensate for the decreased cell size. Because HPOS and VPOS were both doubled, so was the HTEXT= option.

1.4.4 Viewing and plotting using the TARGETDEVICE= option

Often, you may find that you develop a graph on one hardware device and display it on another. Unfortunately, the appearance of a graph often changes from device to device, causing carefully positioned legends and labels to fall in the wrong place. Setting GUNIT=PCT on the GOPTIONS statement helps to make the appearance of graphs more consistent among devices.

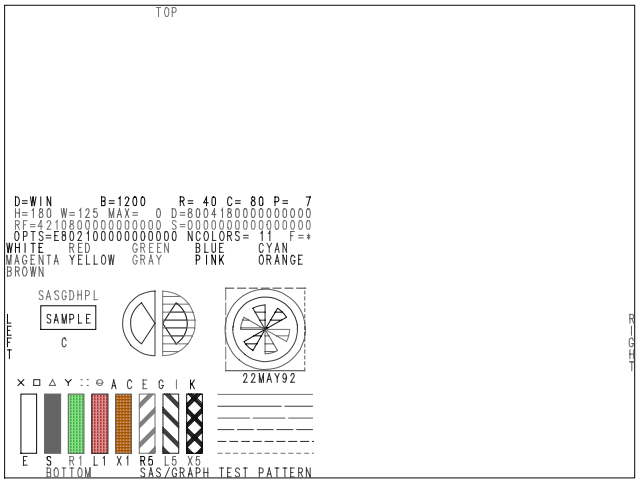
The problem of the changing appearance of the graph can also be mitigated somewhat by utilizing the TARGETDEVICE= option. This GOPTION allows the developer to view a graph on one device approximately as it will be displayed on a second device (the target device). The graph produced when you use the TARGETDEVICE= option is the best representation of what

the graph will look like. A device without color capabilities cannot produce a color plot even if the target device is a color plotter.

The following code runs GTESTIT with TARGETDEVICE=WIN:

```
goptions device=hp7475a targetdevice=win;
proc gtestit pic=1;
run;
```

Compare the following graph (Figure 1.4.4 produced using TARGETDEVICE=WIN) with the GTESTIT graph produced in Section 1.4.1 (Figure 1.4.1a). Both were produced with a DEVICE=HP7475A. However, the target device causes SAS/GRAPH to use the device options associated with that device (in this case, a monitor operating under Release 6.08 of the SAS System for the Windows environment) rather than the device that is actually producing the graph.



MORE INFORMATION:

SAS/GRAPH Software: Usage, pp. 603-604 - An example using the TARGETDEVICE= option.