GENERAL PLOT SUBROUTINE PACKAGE FOR THE
HEWLETT PACKARD 5451C FOURIER ANALYZER®

Dorothy A. Francis

December, 1984

NAVAL BIODYNAMICS LABORATORY
New Orleans, Louisiana

Approved for public release. Distribution unlimited.
DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.
**Title:** General Plot Subroutine Package for Hewlett Packard 5451C Fourier Analyzer®

**Authors:** Dorothy A. Francis

**Abstract:**

The Hewlett Packard (HP) 5451C Fourier Analyzer® formerly supported two plot packages: one in the Fourier environment and one in the Real-Time Executive (RTE) environment. Thus, programs which produced graphic output in the Fourier environment could not run in the RTE environment and conversely. Therefore, it was necessary to develop a standard package of plot subroutines that could run under any operating system. The package is comprised of several FORTRAN subroutines which interact directly with the plot devices.
and also provide the capability to plot to the operator's terminal. The plot package has been found to be extremely useful for graphic output on the HP-5451C® because it can run under any operating system. The plot package has been tested on the HP-5451C® system and can be recommended to users with a similar need.
GENERAL PLOT SUBROUTINE PACKAGE FOR THE HEWLETT PACKARD 5451C FOURIER ANALYZER®

Dorothy A. Francis

December 1984

Naval Medical Research and Development Command
Work Unit No. MF5852402E0005

Approved by
CDR W. W. McIntosh, MSC USN
Executive Officer

Released by
CAPT Robert J. Biersner, MSC USN
Commanding Officer

Naval Biodynamics Laboratory
P. O. Box 29407-0407
New Orleans, LA 70189

Opinions or conclusions contained in this report are those of the author and do not necessarily reflect the views or the endorsement of the Department of the Navy. Approved for public release; distribution unlimited. Reproduction in whole or in part is permitted for any purpose of the United States Government.
SUMMARY

THE PROBLEM

The Hewlett Packard (HP) 5451C Fourier Analyzer formerly supported two plot packages: One in the Fourier environment and one in the RTE environment. Thus, programs which produced graphic output in the Fourier environment could not run in the RTE environment and conversely. Therefore, it was necessary to develop a standard package of plot subroutines that could run under any operating system. The package is comprised of several FORTRAN subroutines which interact directly with the plot devices, and also provide the capability to plot to the operator's terminal.

FINDINGS

The plot package has been found to be extremely useful for graphic output on the HP-5451C because it can run under any operating system.

RECOMMENDATIONS

The plot package has been tested on the HP-5451C system and can be recommended to users with a similar need.

ACKNOWLEDGEMENTS

Mr. William Anderson designed and implemented many of the subroutines on the EAI Pacer 100® and the PDS Nova 800® systems. He also provided valuable technical and structural suggestions. The author also is indebted to Ms. Judy Holm for typing and assembling the manuscript in an excellent manner.

Trade names of materials or products of commercial or non-government organizations are cited only where essential for precision in describing research procedures or evaluation of results. Their use does not constitute official endorsement or approval of the use of such commercial hardware or software.
GENERAL PLOT SUBROUTINE PACKAGE FOR THE
HEWLETT PACKARD 5451C FOURIER ANALYZER

INTRODUCTION

Data plotting is an integral part of any data acquisition and/or analysis software package. Plots are very helpful in data analysis and problem solving. Furthermore, results in graphical form are usually much easier to analyze than are the results in printed tabular form.

This subroutine package is a programming tool that allows the user to conveniently use various plot devices on the HP-5451C system. The plot devices used were the HP-9872T plotter and the operator's terminal. Libraries of compatible subroutines are provided which permit the user to generate a wide variety of plotted information including the following:

1. Labelled axes.
2. Textual data.
3. Graphs from data arrays (x and y), with optional scaling of either array and centered symbols denoting the location of a data point.
4. Variables from the FORTRAN IV calling program plotted in real or integer format.
5. Individual point and vector plotting.

The user also has control of the following:

2. "Pen" position (up or down).
4. Scaling of plot.
5. Rotation of text and axes.
FUNCTION

The following subroutines are supplied:

PLOTS - Initializes plot software package for user's hardware and software configuration.

PLOT - Moves "pen" to specified (x,y) coordinates with "pen" in up or down position; permits origin control; controls pagination.

FACTO - Scales size of subsequent plotting data.

WHERE - Returns current position and factor to user program.

SYMBO - Plots textual information (such as titles) at desired angle and size with additional capability of plotting special symbols for marking data points.

NUMBE - Plots each digit in a variable, including optional decimal point and truncation.

SCALE - Calculates parameters for axis annotation and size of plot for data array.

AXIS - Plots an axis, at desired angle, including segment markings and optional title.

LINE - Generates the plot of data in two arrays (x and y) with optional connecting lines and special symbols.

Four packages with identical subroutine calling sequences are available. The computer systems, plot peripherals, and unique characteristics of each package are summarized in Table 1.

Four interrelated coordinate systems are used, as shown in Figure 1. Embedded within the subroutines are two machine-dependent coordinate systems; one is defined in plotter machine units, the other in plotter physical inches. These are defined by PLOTS, and normally user will select the default parameters for the particular plotter selected. Users may also define their own imaginary plotting surface in which the relationship of the origin and scale relative to the plotter coordinate system are defined by PLOT and FACTOR. Finally, the relationship between "problem units" and the imaginary "user inches" is defined using SCALE.
SUBROUTINE DESCRIPTIONS

PLOTS

The routine PLOTS must be called at the start of a user program to initialize internal parameters. The call is:

CALL PLOTS (A, I)

A Resolution of plotter: inches/machine unit (0.0: defaults to hardware resolution).

I Logical unit for plot device (0: defaults to standard device address).

PLOTS initializes the scale factor to 1.0 (one user inch equals one physical inch) and sets the "pen" location and origin to (0.0, 0.0).

NOTES: 1) "PLOTS" must be called before any other call to the subroutine package.

2) The origin is initially at the bottom left of the page. (Top left for Printronix 300® printer).

3) The X-axis runs horizontally across the page. The Y-axis runs vertically up the page. (X=down, Y=across for Printronix 300® printer.)

4) Pen # 1 is selected for the HP-9872T® plotter.

PLOT

The routine PLOT causes "pen" motions to be output. PLOT controls the origin and pagination of the plot. The call is:

CALL PLOT (X, Y, I)

X X-coordinate (user inches) relative to most recent origin.

Y Y-coordinate (user inches) relative to most recent origin.

I Integer of the set (-3, -2, 3, 4, 10, -10, 11, 999).

-3 = move to (X,Y) pen up, set new origin to (X,Y).

-2 = move to (X,Y) pen down, set new origin to (X,Y).

2 = move to (X,Y) pen down.

3 = move to (X,Y) pen up.

10 = advance paper one full page.
-10 = advance paper one half page.

11 = End of plots; reset the HP-2648A® graphics terminal (first press). This frees the keyboard and clears I/O operations.

999 = same as 11 (included for completeness and compatibility).

Note: All programs should be ended with the sequence:

CALL PLOT (0.,0.,10); CALL PLOT (0.,0.,-10); or CALL PLOT (0.,0.,11).

FACTOR

FACTOR allows the alteration of the overall scaling factor for plot size. The initial factor is set to 1.0 so that one (1) user inch equals one (1) physical inch. The call is:

CALL FACTOR (Z)

Z Ratio of desired plot size to current size.

1.0 = Reset overall scaling factor to 1.0.

The effects of multiple calls to FACTOR are cumulative.

WHERE

WHERE allows the user program to interrogate the plot package for the current overall scaling factor and the current (X,Y) location with respect to the current origin. The call is:

CALL WHERE (X,Y,Z)

X X-coordinate (user inches) with respect to current origin.

Y Y-coordinate (user inches) with respect to current origin.

Z Current overall scaling factor.

NUMBE

NUMBE plots floating point data in a format similar to FORTRAN IV F-format. One number is plotted for each call to NUMBE. The call is:

CALL NUMBE (X,Y,H,T,A,N)

X X-coordinate (user inches) of number relative to most recent origin.

Y Y-coordinate (user inches) of number relative to most recent origin.

H Height of number (user inches).
T     Number (floating point).
A     Angle of number to edge of paper (degrees CCW from X axis).
N     Format control integer:
      0     T is truncated and plotted as an integer followed by ".".
      -1    T is truncated and plotted as an integer.
      +N    N digits to the right of the decimal point are plotted.

SCALE

SCALE calculates the axis limits and distance between tic marks for an array to be used with AXIS and LINE to produce a properly annotated axis and a graph whose data include all points in a user specified length. SCALE does no plotting; it only provides data for AXIS and LINE. The call is:

CALL SCALE (A,S,N,I)

A Array containing data problem units to be scaled. (Where the dimensions are A(I, N+2)).
S Length of axis that data is to cover (user inches); number of tic marks +1.
N Number of data values in array A. (N must be greater than 0)
I Increment between data values in array A. (Used to allow data from 2-dimensional arrays, A(I, N+2).)

SCALE searches the data array A for minimum and maximum data values and selects convenient values (problem units) for tic marks on the axis. Tic marks are set one inch apart (user inches).

The calculated value (problem units) for the first tic mark is stored at A(N*J+1). The increment (problem units/user inch) between tic marks is stored at A(N*J+J+1), where J is the absolute value of I.

Note: If I is not equal to 1, a format incompatibility with AXIS exists.

AXIS

AXIS draws an axis with labeled tic marks at one inch intervals (user inches), with an optional title centered on the axis. The call is:

CALL AXIS (X,Y,T,N,S,A,F,D)

X     X-coordinate at start of axis (user inches).
Y     Y-coordinate at start of axis (user inches).
T     Title (packed MSB, LSB).
N  Number of characters in title:
   - on clockwise side of axis (normal for X).
   + on counterclockwise side of axis (normal for Y).
S  Length of axis (user inches).
A  Angle in degrees CCW from machine X at which axis is drawn:
   0.0 for X axis.
   90.0 for Y axis.
F  The starting value of first tic on axis (problem units).*
D  Increment between tic marks on axis (problem units).*

SYMBO

The SYMBO routine has two major uses:

1. Print a character string.
2. Print a single character by value.

The call is:

CALL SYMBO (X,Y,H,T,A,N)

X  X-coordinate (user inches) of symbol relative to most recent
    origin.
Y  Y-coordinate (user inches) of symbol relative to most recent
    origin.
H  Height of symbol (user inches).
T  Text string (packed MSB, LSB) or value of single symbol (0.0 -
    127.0).
A  Angle of string or symbol CCW from machine X axis (degrees).
N  + = Number of characters in string.
   0 = No action.
   -1 = Draw symbol corresponding to value T at point X,Y.
   -2 = Draw line from previous X,Y to desired X,Y, then draw symbol.

*F and D are usually supplied by SCALE. F is available in DATA(N+1).
D is available in A (N+2) in SCALE argument list.
LISTING OF A COMMAND FILE FOR AN
OVERLAY GENERATION USING THE PLOT PACKAGE
REFERENCE

TABLE 2: SYMBOLS

<p>| | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>m</td>
<td>16</td>
<td>≠</td>
<td>32</td>
<td>48</td>
<td>0</td>
<td>64</td>
<td>G</td>
<td>80</td>
<td>P</td>
<td>96</td>
</tr>
<tr>
<td>1</td>
<td>o</td>
<td>17</td>
<td>±</td>
<td>33</td>
<td>1</td>
<td>49</td>
<td>1</td>
<td>65</td>
<td>A</td>
<td>81</td>
<td>Q</td>
</tr>
<tr>
<td>2</td>
<td>Δ</td>
<td>18</td>
<td>&quot;</td>
<td>34</td>
<td>2</td>
<td>50</td>
<td>2</td>
<td>66</td>
<td>D</td>
<td>82</td>
<td>R</td>
</tr>
<tr>
<td>3</td>
<td>+</td>
<td>19</td>
<td>#</td>
<td>35</td>
<td>3</td>
<td>51</td>
<td>3</td>
<td>67</td>
<td>C</td>
<td>82</td>
<td>S</td>
</tr>
<tr>
<td>4</td>
<td>×</td>
<td>20</td>
<td>$</td>
<td>36</td>
<td>4</td>
<td>52</td>
<td>4</td>
<td>68</td>
<td>D</td>
<td>84</td>
<td>T</td>
</tr>
<tr>
<td>5</td>
<td>*</td>
<td>21</td>
<td>%</td>
<td>37</td>
<td>5</td>
<td>53</td>
<td>5</td>
<td>69</td>
<td>E</td>
<td>85</td>
<td>U</td>
</tr>
<tr>
<td>6</td>
<td>+</td>
<td>22</td>
<td>&amp;</td>
<td>38</td>
<td>6</td>
<td>54</td>
<td>6</td>
<td>70</td>
<td>F</td>
<td>85</td>
<td>V</td>
</tr>
<tr>
<td>7</td>
<td>x</td>
<td>23</td>
<td>'</td>
<td>39</td>
<td>7</td>
<td>55</td>
<td>7</td>
<td>71</td>
<td>G</td>
<td>87</td>
<td>W</td>
</tr>
<tr>
<td>8</td>
<td>Z</td>
<td>24</td>
<td>(</td>
<td>40</td>
<td>8</td>
<td>56</td>
<td>8</td>
<td>72</td>
<td>H</td>
<td>89</td>
<td>X</td>
</tr>
<tr>
<td>9</td>
<td>Y</td>
<td>25</td>
<td>)</td>
<td>41</td>
<td>9</td>
<td>57</td>
<td>9</td>
<td>73</td>
<td>I</td>
<td>89</td>
<td>Y</td>
</tr>
<tr>
<td>10</td>
<td>X</td>
<td>26</td>
<td>*</td>
<td>42</td>
<td>10</td>
<td>58</td>
<td>10</td>
<td>74</td>
<td>J</td>
<td>93</td>
<td>Z</td>
</tr>
<tr>
<td>11</td>
<td>k</td>
<td>27</td>
<td>+</td>
<td>43</td>
<td>11</td>
<td>59</td>
<td>11</td>
<td>75</td>
<td>K</td>
<td>91</td>
<td>l</td>
</tr>
<tr>
<td>12</td>
<td>X</td>
<td>28</td>
<td>,</td>
<td>44</td>
<td>12</td>
<td>60</td>
<td>&lt;</td>
<td>76</td>
<td>L</td>
<td>92 \</td>
<td>103 \</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>29</td>
<td>–</td>
<td>45</td>
<td>13</td>
<td>61</td>
<td>=</td>
<td>77</td>
<td>M</td>
<td>93</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>–</td>
<td>30</td>
<td>,</td>
<td>46</td>
<td>14</td>
<td>62</td>
<td>&gt;</td>
<td>78</td>
<td>N</td>
<td>94</td>
<td>&gt;</td>
</tr>
<tr>
<td>15</td>
<td>8</td>
<td>31</td>
<td>/</td>
<td>47</td>
<td>15</td>
<td>63</td>
<td>?</td>
<td>79</td>
<td>O</td>
<td>95</td>
<td>+</td>
</tr>
</tbody>
</table>
## TABLE 1: SUMMARY OF SUBROUTINE PACKAGES

<table>
<thead>
<tr>
<th>SYSTEM</th>
<th>PLOT DEVICE</th>
<th>LOGICAL UNIT</th>
<th>X, Y PLOT LIMITS</th>
<th>RESOLUTION</th>
<th>SUBROUTINE FILE</th>
<th>I/O DRIVER FILE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PACER 100</td>
<td>HP7220</td>
<td>132)30</td>
<td>16&quot;, 10.5&quot;</td>
<td>.001&quot;</td>
<td>HPPLOT</td>
<td>HP10</td>
</tr>
<tr>
<td>PACER 100</td>
<td>TEK4010</td>
<td>1)8</td>
<td>10.24&quot;, 7.8&quot;</td>
<td>.01&quot;</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>NOVA 800</td>
<td>TEK4051</td>
<td>10</td>
<td>10.24&quot;, 7.8&quot;</td>
<td>.01&quot;</td>
<td>PLOTIO:PLOTLB.LB</td>
<td>PLOTIO:TEKLIB.LB</td>
</tr>
<tr>
<td>NOVA 800</td>
<td>PDS Display</td>
<td>34)8</td>
<td>10.24&quot;, 10.24&quot;</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>NOVA 800</td>
<td>Printronix</td>
<td>-</td>
<td>Unlimited, 15&quot;</td>
<td>.01&quot;</td>
<td>PLOT300:PLOTLB.LB</td>
<td>NONE REQUIRED</td>
</tr>
<tr>
<td>HP5451C</td>
<td>HP 9872T</td>
<td>5&amp;30</td>
<td>15.75&quot;, 11.2&quot;</td>
<td>.001&quot;</td>
<td>PLTPK</td>
<td>NONE REQUIRED</td>
</tr>
<tr>
<td>HP5451C</td>
<td>HP 2648A</td>
<td>1</td>
<td>6.5&quot;, 4.5&quot;</td>
<td>.0228&quot;</td>
<td>PLTPK</td>
<td>NONE REQUIRED</td>
</tr>
</tbody>
</table>
Procedure to measure time to draw and label axes. Bits zero through five of the S-register will be on. Bits three through five will be turned off at the start of the axes operation. Bits zero through two will be the only bits on. Start timing at this point. Continue timing until all bits except zero through five are on. This signals the completion of the operation.

Plot package efficiency. The efficiency of the general plot subroutine package was measured using the procedures described earlier. A digital timer was used to time the plotting of data and the drawing and labeling of axes. The timing was also measured using the time base generator that is installed on the Fourier Analyzer. The total plot time was thirty-five seconds using the terminal as the plot device and one minute and twenty-eight seconds using the HP-9872T® plotter as the plot device. The time required to draw and label axes was thirty-one seconds using the terminal and one minute and eight seconds using the plotter.

The total plot time includes scaling the data, positioning the pen, plotting the data, and the drawing and labeling of axes. The time required to plot the data only was approximately 4.7 seconds on the terminal and 9.85 seconds on the plotter. The time required to produce a hardcopy of the plotted data is not included in any of the time measurements.

CONCLUSION

The plot package evaluation program, Y9000, was developed in the Fourier Operating System. Although relatively small in size (113 lines of coding), Y9000 required 2237 words of memory, indicating that the 5451C® utility routines for the Fourier system require a considerable amount of memory. Due to the lack of sufficient memory, only 128 words of data could be plotted. The plotting time is slow relative to data acquisition and sampling rates.

The general plot package presented herein is a valuable programming tool. Although somewhat limited in the Fourier Operating System, the package provides significant programming capabilities in the RTE operating system. As mentioned earlier, this package operates in both systems. The general plot package performance was not evaluated in the RTE Operating System because our data acquisition plans require using the Fourier Operating System. Nevertheless, the programming potential of the general plot package in the RTE Operating System are excellent.
The plot package evaluation program provides a means of determining the performance characteristics of the general plot subroutine package, thereby facilitating the estimation of throughput rates and response times for the Fourier system. The program has the following features:

1. A standard procedure to measure time to draw and label axes.
2. A standard procedure to measure time to plot data.
3. Capability to select a plot device interactively.
4. Validation of the major features of the plot package. These features include: A hardcopy from the terminal, pen selection on the HP-9872T® plotter, pagination on the plotter (which may be automatic or manual), and a user-defined scale factor.

FUNCTION:

Y9000 begins by requesting user input. The user must input the logical unit number of the plot device. The user is given the option to run the program with the standard values or with changes. The user may change the pen number and/or page size on the HP-9872T® plotter, request manual paging (if necessary) on the plotter, and/or change the scale factor.

The plot package is initialized and the 5451C utility routines are used to define 128 words of Fourier data and to store the data in X,Y arrays. The general plot subroutines are used to scale and plot the data, and to draw and label the axes. Bits zero through five of the S-register are turned on (set to 1) to signal the start of the plot operation. Some of these bits are turned off (set to 0) during the operation. Bits six through fifteen on (all bits except 0 through 5) signals the completion of an operation. Bits zero through one of the S-register are turned on to signal the start of the drawing and labeling of axes. These bits are turned off to signal the completion of the operation. However, this action is so fast that the end of the program is signaled before the user notices it. Therefore, bits six through fifteen on signals the end of this operation also. A hardcopy of the plotted data is produced.

Procedure to measure time to plot data. Plot time and CPU utilization are the same. Bits zero through five of the S-register will be turned on at the start of the plotting operation. Start timing at this point. Some of the bits will be turned off during the operation. This is not a problem. Continue timing until bits six through fifteen (all bits except zero through five) are on. This signals the completion of plotting.
EVALUATION OF THE PLOT PACKAGE

A plot package evaluation program (Y9000) was developed in order to measure plot performance. The program was written in FORTRAN and was executed under the following machine conditions:

HP-5451C Fourier Analyzer® settings:

ADC CONTROLS

SAMPLE MODE TO KHz/ms
MULTIPLIER to TIME 100 (black 100)
EXT/INT to INT
OVERLOAD VOLTAGE SWITCH A to .8
INPUT SELECTOR to A
DISPLAY SELECTOR to A
TRIGGER SOURCE to FREE RUN
TRIGGER LEVEL to SLOPE

DISPLAY UNIT CONTROLS

CRT Portion:

MAGNIFIER to X1
DISPLAY to INT
INTENSITY as necessary
FOCUS and SCALE as desired
All level switches to center except DISPLAY TYPE to CONT

Control Portion:

GAIN to CAL
MODE to REAL/MAGNITUDE
POLAR ANG/DIV to any position
SCALE to straight up

The HP-3310A Function Generator settings:

FREQUENCY DIAL to 28
RANGE SWITCH to 10
DC OFFSET SWITCH to 0
FUNCTION SWITCH TO TRI (triangle)
OUTPUT LEVEL set with arrow pointing towards "+" sign by DC offset switch
where:

OVSFSA is the base command file for generating overlays.

OVCONA is the name of the command file for building the overlay.

N is the overlay number.

A is the revision (e.g., B would be revision 2).

N is the overlay number.

1 indicates send the test output to the printer (use 0 for output to the terminal).

After the overlay is generated and stored, the program is ready for execution. In order to execute the program, the user will have to boot up the Fourier System and give the following command via the keyboard:

```
USER
PROG  n  n  n  n  n  ENTER
```
USING THE PLOT PACKAGE

All of the plot subroutines and the utility routines have been merged to create two complete files. The file &PLTPK contains the source code and the file %PLTPK contains the binary code. These files are on the RTE and Fourier operating systems. Procedures for using the plot package on both systems are given below:

The program execution procedure using the plot package on the RTE operating system is as follows:

:LOADR,%PROGN
  SE,%SUBR1
  SE,%SUBR1
  ...
  SE,%SUBRN
LIBRARY,%PLTPK
SL
EN

:PROGN

Procedures for using the plot package on the Fourier operating system are as follows:

Write your user programs in FORTRAN IV. Your program must have a name in the format "Ynnnn", where "Y" specifies that it is a user program and "nnnn" is a four-digit positive number (e.g., 1234), excluding numbers of resident user programs. For FORTRAN IV programming, all the interfacing is handled by the compiler. Define the program as a 'SUBROUTINE" and use "CALL" statements to use the utility routines.

Use the Overlay Generation procedure in Section III of the 5451C System Software Manual® to generate an overlay. The Overlay Generation procedure requires a command file containing a list of the user programs to be included in the overlay. The command file OVC05A may be used as a pattern for creating an overlay file that utilizes the plot package. A listing of OVC05A and the output from an Overlay Generation using OVC05A are given at the end of this section.

Once the command file has been created an overlay can be generated by using the command:

TR, OVSFSA, OVCONA, N, 1 (return)
FP2AS

The routine FP2AS converts a floating point number to an ASCII character string. The call is:

CALL FP2AS (FP, NASC, ASC, I)

FP Floating point value.
NASC Number of characters to be returned in ASCII string.
ASC Array in which ASCII character string is returned.
I Number of characters to left of decimal point (output).

PXPCO

The routine PXPCO stores and retrieves the current value of variables that are common to several programs. The call is:

CALL PXPCO (JST, LU, RE, OX, OY, SF, PX, PY)

JST 0: Store values.
    1: Retrieve values.
LU Logical unit for plot device.
RE(2) X,Y conversion factors to go from machine inches to machine units.
OX,OY Plot origin, in machine inches.
SF Conversion factor to go from user inches to machine inches.
PX,PY Current pen position.

The variables LU,, RE, OX, OY, SF, PX, and PY are initialized in the routine PLOTS. The only variable in the CALL statement that requires a number value is JST. If the user's program needs the variables, a call to PXPCO with JST equal to 1 has to be made. Whenever any of these variables are modified, a call to PXPCO with JST equal to 0 must be made in order to store the current values.
CALL HPCSZ (W,H)

W Width of characters in inches.
H Height of characters in inches.

HPOUT

The routine HPOUT causes plotter instructions to be output. An ASCII character string is created using one character per call and the string is output to the plotter when a terminator (;) is received. The call is:

CALL HPOUT (ICHAR)

ICHAR Character to output to plotter

HPPAG

The routine HPPAG controls pagination on the HP-9872T® plotter. Paper may be manually placed in position or advanced a full or half page. The call is:

CALL HPPAG (ISW)

ISW 1: Full page advance.
     2: Half page advance.
     3: Manually place sheet in position.

HPPEN

The routine HPPEN controls pen selection on the HP-9872T plotter. It provides the means to select a pen through program control. The call is:

CALL HPPEN (I)

I The pen position number; it must be an integer in the range 0 through 8.

0 Directs the pen arm to return the pen it is currently holding to its stall.
1-8 Directs the pen arm to fetch the desired pen.
UTILITY SUBROUTINES

Several utility subroutines are used in the general plot subroutine package and are included in the program file. Therefore, a brief description of each is given in this section.

PXPCG

The routine PXPCG extracts a specific character from a given character string (RHS first, LHS second). The call is:

CALL PXPCG (IO, I1, I2)

IO ASCII character string.
I1 Number of the character to be extracted.
I2 Word in which character is returned (RHS).

CGET

The routine CGET extracts a specific character from a given character string (LHS first, RHS second). The call is:

CALL CGET (IO, I1, I2)

IO ASCII character string.
I1 Number of the character to be extracted.
I2 Word in which character is returned (RHS).

HPCDR

The routine HPCDR specifies the absolute direction in which characters are lettered. HPCDR calculates run and rise values using an angle supplied and causes the character direction instruction to be output. The call is:

CALL HPCDR (TH)

TH Angle between character base and machine X-axis in degrees.

HPCSZ

The routine HPCSZ specifies the size of characters and symbols. HPCSZ converts inches to centimeters and causes the character size instruction to be output. The call is:
Table 2 lists the available symbols. Symbols 0 to 31 are centered. Symbols 32 to 127 have their lower left corner at coordinate X,Y.

LINE

LINE combines pairs of data points in two arrays according to user-specified parameters. Points can be plotted with a special symbol (optional) and a line between points (optional). The call is:

CALL LINE (A,B,N,I,L,J)

where:

A Name of array containing values for abscissa (problem units)

where: A(I,N+2): Array dimension.

I: Number of variables interlaced in array.

N: Number of points to be plotted.

A(I,N+1): Minimum value on axis (problem units, normally calculated by subroutine SCALE).

A(I,N+1): Problem units/user inch, normally calculated by subroutine SCALE.

B Name of array containing values for ordinate (problem units).

(Definitions are the same as A.)

N Number of points in each array to be plotted.

I Increment between points in array to be plotted (e.g., A(I,N)).

L Format control:

+L Points connected by line with symbol at point.

0 Points connected by line with no symbols.

-L Symbol at each point with no connecting line.

J Value of special symbol to be plotted.
OVC05A T=00004 IS ON CR00022 USING 00006 BLKS R=0000

0001     00
0002     F,OVRLVN "OVC05A" 54451-14805 REV. A 24 AUG 83
0003     T,SNP51C BASE COMMAND FILE FOR USER OVERLAY GENERATION
0004     F,00006 INSERT Y COMMAND NUMBERS BEFORE 'OE'. AN EXAMPLE:
0005     Y,7001 '0,0077', WHERE '0077'=4-DIGIT Y COMMAND NUMBER
0006     Y,8000
0007     Y,0006
0008     Y,OE
0009     CSORT+ INSERT PROGRAM LOADS BETWEEN 'CSORT+' AND 'CSORT-
0010     F,Y77JOB EXAMPLE: 'F,Y77JOB', WHERE 'Y77JOB'= FILE NAME
0011     Y,Y0077 EXAMPLE: 'Y,Y0077', WHERE 'Y0077'= PROG NAME
0012
0013     F,%Y7001
0014     Y,Y7001
0015     F,%Y8000
0016     Y,Y8000
0017     F,%DEM01
0018     Y,DEM01
0019     F,%DEM03
0020     Y,DEM03
0021     F,%DEM04
0022     Y,DEM04
0023     F,%PLTPK
0024     Y,PLOT
0025     F,%PLTPK
0026     Y,PLT
0027     Y,PLT
0028     Y,FACTO
0029     F,%PLTPK
0030     Y,WHERE
0031     F,%PLTPK
0032     Y,SYMB
0033     F,%PLTPK
0034     Y,NMBE
0035     F,%PLTPK
0036     Y,SCAL
0037     F,%PLTPK
0038     Y,AXIS
0039     F,%PLTPK
0040     Y,LIN
0041     F,%PLTPK
0042     Y,PCG
0043     F,%PLTPK
0044     Y,GCT
0045     F,%PLTPK
0046     Y,PCOF
0047     F,%PLTPK
0048     Y,PCSZ
0049     F,%PLTPK
0050     Y,POUT
0051     F,%PLTPK
0052     Y,PPAC
0053     F,%PLTPK
0054     Y,PPEN
0055     F,%PLTPK
0056     Y,PCG
0057     F,%PLTPK
0058     Y,FP2AS
0059     CSORT-
0060
0061     F,N3500A
0062     L,**
0063     F,N3501A
LISTING OF AN OVERLAY GENERATION OF THE

GENERAL PLOT PACKAGE DEMONSTRATION PROGRAM
*************
*
* F,%PLTPK
* Y,SYMBO
*************

SYMBO  60047 62060  01021 01061

*LOAD*

*************
*
* F,%DEM04
* Y,DEM04
*************

DEM04  62061 62720  01016 01020

*LOAD*

*************
*
* F,%PLTPK
* Y,PLOT
*************

PLOT   62721 63537  01014 01015

*LOAD*

*************
*
* F,%PLTPK
* Y,HPPAG
*************

HPPAG  63540 63740

*LOAD*

*************
*
* F,%PLTPK
* Y,WHERE
*************

WHERE  63741 64017  01013 01013

*LOAD*

*************
*
* F,%PLTPK
* Y,SCALE
*************

SCALE  64020 64514  01010 01012

*LOAD*
SANTA CLARA DIVISION AUTO OVERLAY GENERATION-6K15

*************
* A,OVRLYN 'OVCO5A' 54451-14805 REV. A 24 AUG 83
* T,SNP51C BASE COMMAND FILE FOR USER OVERLAY GENERATION
*************

*ENTER USERCOMMAND NUMBERS, OR TYPE "?"
*************

* INSERT Y COMMAND NUMBERS BEFORE 'OE'. AN EXAMPLE:
* '0,0077', WHERE '0077'=4-DIGIT Y COMMAND NUMBER
* @,7001

Y7001
  Y%COM 01075 01075 00434 00435
  @,8000

Y8000
  Y%COM 01074 01074 00436 00437
  @E
  Y%COM 00440 00441 47440 47440

TYPE "L" TO LOAD USERCOMMAND LIBRARY

*************
* CSORT+
* INSERT PROGRAM LOADS BETWEEN 'CSORT+' AND 'CSORT-'
* EXAMPLE: 'F,Y77JO', WHERE 'Y77JO' = FILE NAME
* 'Y,Y0077', WHERE 'Y0077' = FROG NAME
* F,%PLTPK
* Y,AXIS
*************

AXIS 56442 57726 01066 01073

*LOAD@

*************
* F,%Y7001
* Y,Y7001

*************

Y7001 57727 57770 01063 01065

*LOAD@

*************
* F,%PLTPK
* Y,HPHEN
*************

HPHEN 57771 60046 01062 01062

*LOAD@
************
* 
* F,%PLTPK 
* Y,NUMBE 
************

NUMBE  64515 65113  01006 01007

*LOAD*

************
* 
* F,%PLTPK 
* Y,FP2AS 
************

FP2AS  65114 65503

*LOAD*

************
* 
* F,%PLTPK 
* Y,LINE 
************

LINE  65504 65776

*LOAD*

************
* 
* F,%PLTPK 
* Y,PLOTS 
************

PLOTS  65777 66334

*LOAD*

************
* 
* F,%Y8000 
* Y,Y8000 
************

Y8000  66335 66672  01002 01005

*LOAD*

************
* 
* F,%DEM01 
* Y,DEM01 
************

DEM01  66673 67221  01001 01001

*LOAD*
***************
*              *
*  F,DEM03    *
*  Y,DEM03    *
***************

DEM03  67222 67537

*LOAD@

***************
*              *
*  F,PLTPK     *
*  Y,HPCDR     *
***************

HPCDR  67540 67727 00777 01000

*LOAD@

***************
*              *
*  F,PLTPK     *
*  Y,PXPCG     *
***************

PXPCG  67730 70023 00767 00776

*LOAD@

***************
*              *
*  F,PLTPK     *
*  Y,HPOUT     *
***************

HPOUT  70024 70336

*LOAD@

***************
*              *
*  F,PLTPK     *
*  Y,PXPCO     *
***************

PXPCO  70337 70614

*LOAD@

***************
*              *
*  F,PLTPK     *
*  Y,HPCSZ     *
***************

HPCSZ  70615 70777

*LOAD@
D. 05  76160  76162  00062  00062
D. 06  76163  76165  00060  00060
D. 33  76166  76170  00063  00063
D. 35  76171  76173  00061  00061
D. 76  76174  76176  00055  00055
I. 05  76177  76200  00046  00046
C. 33  76201  76202  00045  00045

*LOAD*

*************
* U , **
*************

*UNDEFINED SYMBOLS
BLANK DTRD GKPLT

*AVAILABLE MEMORY
76203  76224
00442  00545

*LOAD*

*************
* E, 01*
*************

*USER LINKS
76225  76233
00546  01777

*OVERLAY SYMBOL TABLE
.BIO. 74434
.DTA. 74502  01063,1  DFER  75613
.IAY. 74245
.IOR. 74112
.LST. 70101
.RIo. 74032
.XFER 75622  00642,1
.ABS  76007  01007,1
.AMAX1  75154  01012,1
.ARCTA  75660
.BCS  70526
.CGET  71114  01047,1
.D33  76166  00663,1
.DBLE  75262  00574,1
.DEM04  62401  01002,1
.FP2AS  65132
.HPCSZ  79631  01061,1
.MPPEN  57776  01004,1
.LINE  65512  01017,1
.MINO  75057  00777,1
.NUMB  64532  01067,1
.PLOTS  66011

.PXPCG  67733  01040,1

.DIO. 74353  01065,1
.IAR. 74300
.101. 74153
.LDR. 75601
.RAY. 74234
.XCOM  75545  00640,1
.XPAK  75257  00610,1
.AMAX0  75040
.AMIN0  75052
.ATAN  75660  01001,1
.BLANK  ?  ?  ?  01238,1
.D. 05  76160  00662,1
.D. 35  76171  00612,1
.D76  76174  00551
.DEM01  66705  01005,1
.DEM03  67346  01003,1
.DEM04  62401  01002,1
.DTA. 70526
.GKPLT  ?  ?  ?  01234,1
.HPCSR  79631  01061,1
.HPOUT  70175  01027,1
.IABS  76151  01072,1
.MAX0  75045
.MAX1  75161
.MOD  75022  01037,1
.NEUH1  74452
.OLDIO  74457
.PLOT  62742  01070,1
.PXPCG  67733  01040,1
.PXPCF  70365  01073,1
SCALE 64036 01020,1
JHEPE 65750 01066,1

*LINKS
  00546 01777

*AVAILABLE MEMORY
  76203 76224
  00442 00545

*END

*LOAD0

***********
  
*" ENCOUNTERED, OVERLAY GENERATION TERMINATED
LISTINGS OF THE GENERAL PLOT PACKAGE SUBROUTINES
SUBROUTINE PLOTS(RS,IA)

RS   RESOLUTION
0.0->0.01 (DEFAULT)
IA   LOGICAL UNIT FOR OUTPUT
0->10: TEKTRONIX DISPLAY (DEFAULT)
1: HP-2648A GRAPHICS TERMINAL
5: HP-9872T PLOTTER (RTE SYSTEM)
30: HP-9872T PLOTTER (FOURIER SYSTEM)

REV 08 FEB 82: CHANGE FILE NAME TO 'PLOTS'
REV 22 JUN 83: SUPPORT HP-9872T PLOTTER
: INITIALIZE PLOTTER & ENABLE PAPER CUTTER
: SELECT PEN #1
REV 01 SEP 83: SUPPORT HP-2648A GRAPHICS TERMINAL
: MODIFIED TO RUN ON FOURIER SYSTEM
: COMMON STATEMENT REMOVED
: A CALL TO SUBROUTINE 'PRPCO' WAS ADDED

LU   LOGICAL UNIT FOR PLOT OUTPUT
MP   INDEX TO LAST USED WORD IN MB BUFFER
MB   BUFFER FOR OUTPUT TO PLOT DEVICE. NOT READ FOR ALL DEVICES.
PE(2): X,Y CONVERSION FACTORS TO GO FROM MACHINE INCHES TO
MACHINE UNITS
OX, OY: PLOT ORIGIN, IN MACHINE INCHES
SF: CONVERSION FACTOR TO GO FROM USER INCHES TO MACHINE INCHES

REAL AR(2),RE(2)
DATA AR/1.0,1.0/
DATA IESTP015452B,LDLA062141B,ICH041513B
DATA IESLP015550B,IESCL015512B

INITIALIZE PLOT ORIGIN IN MACHINE INCHES
OX=0.0
OY=0.0

INITIALIZE PEN POSITION IN USER UNITS
PX=0.0
PY=0.0

INITIALIZE SCALE FACTOR TO 1.0
SF=1.0

DEFINE OUTPUT LOGICAL UNIT
LU=1A
IF(LU.EQ.0) LU=10

DEFINE FACTOR FOR CONVERSION FROM MACHINE INCHES TO MACHINE UNITS
RT=RS
IF(RT.LE.0.0) AND (LU.EQ.0) RT=.0226
IF(LU.NE.5) AND (LU.NE.30) GO TO 100
IF(RT.LE.0.0) RT=1./400./2.54
100 IF(RT.LE.0.0) RT=.01
RE(1)=RT*AR(1)

33
RE(2)=RT*AR(2)

STORE VALUES

CALL PXPC00,LU,RE,0X,0Y,SP,PX,Py)

IF(LU.EQ.1) GO TO 230

ADVANCE PAPER AND INITIALIZE PLOTTER

WRITE(LU,200)

200 FORMAT("AF;IN")

SELECT PEN #1

WRITE(LU,220)

220 FORMAT("SP1")

GO TO 250

CLEAR SCREEN (ALPHANUMERIC DISPLAY)

CONTINUE

WRITE(LU,240) IESLH,IESCJ

CLEAR GRAPHIC MEMORY AND TURN ON GRAPHIC DISPLAY & CURSOR

WRITE(LU,240) IESTR,LILA,ICK

240 FORMAT("FR")

RETURN

END

!4: COMPILER: HP92834 REV. 2226 (820503)

- NO WARNINGS ++ NO ERRORS ++. PROGRAM: 20€ COMMON: (NONE)
CALL HPUT(1HE)
3 210 CALL HPUT(1ANDICH,377E+2**8)
3 IF(LU.NE.30) GO TO 215
3 CALL HPUT(177E)
1 215 CALL HPUT(3+25E)
2 CALL HPUT(1M;)
3 X=X+W1+C0
4 Y=Y+W1+SI
5 CALL PLOT(X,Y,3)
6 GO TO 40
7 C
8 END

X: COMPILER: HP92834 REV.2226 (820503)

NO WARNINGS ++ NO ERRORS ++ PROGRAM: 1000 COMMON: (NONE)
20 RETURN

30 C  INITIALIZE TO START OF CHARACTER STRING
31 NC0=NC
32 I0=0
33 C  GET NEXT CHARACTER IN STRING
34 40 NC0=NC0-1
35 IF(NCO.LT.0) GO TO 20
36 I0=I0+10
37 CALL CGET(ASC, 10, JCH)
38 50 IF(LU.NE.1) GO TO 55
39 ICH=IAND(JCH,177B)
40 I3=SYM(JCH+1)
41 IF(JCH.GE.32) GO TO 60
42 GO TO 56
43 55 JCH=IAND(JCH,177B)
44 IF(JCH.GE.32) GO TO 200
45 56 CONTINUE
46 C  SPECIAL PLOT CHARACTER MUST BE CENTERED, MOVE PEN
47 I3=SYM(JCH+1)
48 IP=3
49 I1=0
50 X=X+H*(-2.*CO+3.*S1)
51 Y=Y+H*(-2.*CO-2.*S1)
52 WX=X
53 YW=Y
54 GO TO 110
55 60 IF=2
56 I3=I3+13
57 C  GET 8 BIT DEFINITION OF NEXT CHARACTER SEGMENT
58 CALL PXPCG(STB, I3, 11)
59 C  DECODE SEGMENT DEFINITION
60 61 I2=MOD(I1, 64)
61 62 IF(I2.GE.64) IF=3
62 IY=I2-8
63 IX=I2-8*IY
64 X1=IX
65 Y1=IY
66 C  DRAW SEGMENT
67 XW=X+H*IX+CO-Y1*S1
68 YW=Y+H*(Y1+CO+X1+S1)
69 110 CALL PLOT(XW, YW, IF)
70 IF(I1.LT.192) GO TO 60
71 X=XW
72 Y=YW
73 GO TO 40
74 C  WRITE A NORMAL CHARACTER
75 C  CALL HPOUT(1HL)
C MID POINT X, Y COORD & ROTATION CONSTANTS

C X=X0
C Y=Y0

H=TH*0.0174532925
SI=SIN(H)
CO=COS(H)
H=HI/7.0

CALL PXPCO(1,LU,RE,ON,OY,SF,PX,PY)

IF(LU.EQ.1) GO TO 4

C SET CHARACTER SIZE

HO=HI*SF
WO=HO*2.3
W1=W0+1.3333/SF
CALL HPCSZ(W0,HO)

C SET DIRECTION IN WHICH CHARACTERS ARE TO BE LETTERED

CALL HPCDR(TH)

CONTINUE

C MOVE TO NEW POINT

IP=3
IF(NC.EQ.-2) IP=2
CALL PLOT(XA,YA,IP)
IF(NC)10,20,30

C SINGLE CHAR IS TO BE PLOTTED

NC0=0
JCH=ASC(I)+0.1
GO TO 50
<table>
<thead>
<tr>
<th>Page 4</th>
<th>SYMEO OPTS: LXI</th>
<th>3:27 PM WED., 26 DEC., 1984</th>
</tr>
</thead>
<tbody>
<tr>
<td>167</td>
<td>DATA SYN 25/</td>
<td>125, 125, 125, 125</td>
</tr>
<tr>
<td>168</td>
<td>1, 125, 125, 125, 125/</td>
<td></td>
</tr>
<tr>
<td>169</td>
<td>DATA SYN 33/</td>
<td>125, 126, 126, 136, 141</td>
</tr>
<tr>
<td>170</td>
<td>1, 150, 161, 174, 186/</td>
<td></td>
</tr>
<tr>
<td>171</td>
<td>DATA SYN 41/</td>
<td>190, 195, 200, 207</td>
</tr>
<tr>
<td>172</td>
<td>1, 212, 215, 218, 221/</td>
<td></td>
</tr>
<tr>
<td>173</td>
<td>DATA SYN 49/</td>
<td>223, 234, 240, 249</td>
</tr>
<tr>
<td>174</td>
<td>1, 260, 266, 276, 287/</td>
<td></td>
</tr>
<tr>
<td>175</td>
<td>DATA SYN 57/</td>
<td>293, 311, 321, 326</td>
</tr>
<tr>
<td>176</td>
<td>1, 332, 336, 341, 345/</td>
<td></td>
</tr>
<tr>
<td>177</td>
<td>DATA SYN 65/</td>
<td>354, 366, 373, 385</td>
</tr>
<tr>
<td>178</td>
<td>1, 394, 402, 409, 414/</td>
<td></td>
</tr>
<tr>
<td>179</td>
<td>DATA SYN 73/</td>
<td>424, 430, 437, 443</td>
</tr>
<tr>
<td>180</td>
<td>1, 448, 452, 457, 463/</td>
<td></td>
</tr>
<tr>
<td>181</td>
<td>DATA SYN 81/</td>
<td>473, 480, 492, 501</td>
</tr>
<tr>
<td>182</td>
<td>1, 514, 519, 526, 530/</td>
<td></td>
</tr>
<tr>
<td>183</td>
<td>DATA SYN 89/</td>
<td>536, 544, 552, 559</td>
</tr>
<tr>
<td>184</td>
<td>1, 564, 567, 571, 577/</td>
<td></td>
</tr>
<tr>
<td>185</td>
<td>DATA SYN 97/</td>
<td>583, 588, 597, 607</td>
</tr>
<tr>
<td>186</td>
<td>1, 614, 625, 635, 641/</td>
<td></td>
</tr>
<tr>
<td>187</td>
<td>DATA SYN105/</td>
<td>654, 660, 668, 675</td>
</tr>
<tr>
<td>188</td>
<td>1, 681, 687, 697, 704/</td>
<td></td>
</tr>
<tr>
<td>189</td>
<td>DATA SYN113/</td>
<td>714, 724, 735, 740</td>
</tr>
<tr>
<td>190</td>
<td>1, 748, 754, 762, 766/</td>
<td></td>
</tr>
<tr>
<td>191</td>
<td>DATA SYN121/</td>
<td>772, 776, 785, 790</td>
</tr>
<tr>
<td>192</td>
<td>1, 796, 803, 811, 816/</td>
<td></td>
</tr>
<tr>
<td>193</td>
<td>DATA STC 1/</td>
<td>10042, 2008, 11276, -9686</td>
</tr>
<tr>
<td>194</td>
<td>1, 10842, 2233, 2320, 5131</td>
<td></td>
</tr>
<tr>
<td>195</td>
<td>DATA STC 9/</td>
<td>11044, -9694, 10842, 3080</td>
</tr>
<tr>
<td>196</td>
<td>1, -9686, 2666, 7256, 10650</td>
<td></td>
</tr>
<tr>
<td>197</td>
<td>DATA STC 17/</td>
<td>26668, -9716, 10842, 2584</td>
</tr>
<tr>
<td>198</td>
<td>1, 10780, 23256, 10776, 6684</td>
<td></td>
</tr>
<tr>
<td>199</td>
<td>DATA STC 25/</td>
<td>-9718, 11336, 3112, 26842</td>
</tr>
<tr>
<td>200</td>
<td>1, 2092, 22796, -9701, 10330</td>
<td></td>
</tr>
<tr>
<td>201</td>
<td>DATA STC 33/</td>
<td>-9664, 11354, 8547</td>
</tr>
<tr>
<td>202</td>
<td>1, 24872, 2065, 4945, 21260</td>
<td></td>
</tr>
<tr>
<td>203</td>
<td>DATA STC 41/</td>
<td>-9693, 11336, 3176, 10826</td>
</tr>
<tr>
<td>204</td>
<td>1, 7256, 23256, 11304, 3080</td>
<td></td>
</tr>
<tr>
<td>205</td>
<td>DATA STC 49/</td>
<td>-9702, 10826, 22746, -9700</td>
</tr>
<tr>
<td>206</td>
<td>1, 7256, 11366, 3144, 24794</td>
<td></td>
</tr>
<tr>
<td>207</td>
<td>DATA STC 57/</td>
<td>20516, 18708, -9685, 10842</td>
</tr>
<tr>
<td>208</td>
<td>1, 9057, 4945, -14630, 577</td>
<td></td>
</tr>
<tr>
<td>209</td>
<td>DATA STC 65/</td>
<td>2314, 20993, 12849, -14830</td>
</tr>
<tr>
<td>210</td>
<td>1, 12641, 9075, 16886, 29489</td>
<td></td>
</tr>
<tr>
<td>211</td>
<td>DATA STC 73/</td>
<td>24579, 21548, -14832, 2888</td>
</tr>
<tr>
<td>212</td>
<td>1, 6932, 6217, 11305, 626</td>
<td></td>
</tr>
<tr>
<td>213</td>
<td>DATA STC 81/</td>
<td>18630, 26924, 12326, 10545</td>
</tr>
<tr>
<td>214</td>
<td>1, 836, 3083, -14844, -596</td>
<td></td>
</tr>
<tr>
<td>215</td>
<td>DATA STC 89/</td>
<td>2049, 8976, 12843, 6441</td>
</tr>
<tr>
<td>216</td>
<td>1, -14844, 10849, -14796, 6563</td>
<td></td>
</tr>
<tr>
<td>217</td>
<td>DATA STC 97/</td>
<td>785, 29126, -4899, -14847</td>
</tr>
<tr>
<td>218</td>
<td>1, 10826, 4196, 5216, 19142</td>
<td></td>
</tr>
<tr>
<td>219</td>
<td>DATA STC105/</td>
<td>22576, -14820, 4631, 22726</td>
</tr>
<tr>
<td>220</td>
<td>1, -14828, 321, 13510, 1630</td>
<td></td>
</tr>
<tr>
<td>221</td>
<td>DATA STC113/</td>
<td>12584, 11315, 780, 2049</td>
</tr>
</tbody>
</table>
112 EQUIVALENCE (STB(25), STC(25(1))
113 EQUIVALENCE (STB(33), STC(33(1))
114 EQUIVALENCE (STB(41), STC(41(1))
115 EQUIVALENCE (STB(49), STC(49(1))
116 EQUIVALENCE (STB(57), STC(57(1))
117 EQUIVALENCE (STB(65), STC(65(1))
118 EQUIVALENCE (STB(73), STC(73(1))
119 EQUIVALENCE (STB(81), STC(81(1))
120 EQUIVALENCE (STB(89), STC(89(1))
121 EQUIVALENCE (STB(97), STC(97(1))
122 EQUIVALENCE (STB(105), STC(105(1))
123 EQUIVALENCE (STB(113), STC(113(1))
124 EQUIVALENCE (STB(121), STC(121(1))
125 EQUIVALENCE (STB(129), STC(129(1))
126 EQUIVALENCE (STB(137), STC(137(1))
127 EQUIVALENCE (STB(145), STC(145(1))
128 EQUIVALENCE (STB(153), STC(153(1))
129 EQUIVALENCE (STB(161), STC(161(1))
130 EQUIVALENCE (STB(169), STC(169(1))
131 EQUIVALENCE (STB(177), STC(177(1))
132 EQUIVALENCE (STB(185), STC(185(1))
133 EQUIVALENCE (STB(193), STC(193(1))
134 EQUIVALENCE (STB(201), STC(201(1))
135 EQUIVALENCE (STB(209), STC(209(1))
136 EQUIVALENCE (STB(217), STC(217(1))
137 EQUIVALENCE (STB(225), STC(225(1))
138 EQUIVALENCE (STB(233), STC(233(1))
139 EQUIVALENCE (STB(241), STC(241(1))
140 EQUIVALENCE (STB(249), STC(249(1))
141 EQUIVALENCE (STB(257), STC(257(1))
142 EQUIVALENCE (STB(265), STC(265(1))
143 EQUIVALENCE (STB(273), STC(273(1))
144 EQUIVALENCE (STB(281), STC(281(1))
145 EQUIVALENCE (STB(289), STC(289(1))
146 EQUIVALENCE (STB(297), STC(297(1))
147 EQUIVALENCE (STB(305), STC(305(1))
148 EQUIVALENCE (STB(313), STC(313(1))
149 EQUIVALENCE (STB(321), STC(321(1))
150 EQUIVALENCE (STB(329), STC(329(1))
151 EQUIVALENCE (STB(337), STC(337(1))
152 EQUIVALENCE (STB(345), STC(345(1))
153 EQUIVALENCE (STB(353), STC(353(1))
154 EQUIVALENCE (STB(361), STC(361(1))
155 EQUIVALENCE (STB(369), STC(369(1))
156 EQUIVALENCE (STB(377), STC(377(1))
157 EQUIVALENCE (STB(385), STC(385(1))
158 EQUIVALENCE (STB(393), STC(393(1))
159 EQUIVALENCE (STB(401), STC(401(1))
160 EQUIVALENCE (STB(409), STC(409(1))

161 DATA SYN 1/ 0, 8, 20, 26
162 1, 31, 36, 43, 50
163 DATA SYN 9/ 55, 62, 68, 82
164 1, 91, 98, 101, 104
165 DATA SYN 17/ 111, 118, 125, 125
166 1, 125, 125, 125, 125
57 INTEGER STC129(8)
58 INTEGER STC137(8)
59 INTEGER STC145(8)
60 INTEGER STC153(8)
61 INTEGER STC161(8)
62 INTEGER STC169(8)
63 INTEGER STC177(8)
64 INTEGER STC185(8)
65 INTEGER STC193(8)
66 INTEGER STC201(8)
67 INTEGER STC209(8)
68 INTEGER STC217(8)
69 INTEGER STC225(8)
70 INTEGER STC233(8)
71 INTEGER STC241(8)
72 INTEGER STC249(8)
73 INTEGER STC257(8)
74 INTEGER STC265(8)
75 INTEGER STC273(8)
76 INTEGER STC281(8)
77 INTEGER STC289(8)
78 INTEGER STC297(8)
79 INTEGER STC305(8)
80 INTEGER STC313(8)
81 INTEGER STC321(8)
82 INTEGER STC329(8)
83 INTEGER STC337(8)
84 INTEGER STC345(8)
85 INTEGER STC353(8)
86 INTEGER STC361(8)
87 INTEGER STC369(8)
88 INTEGER STC377(8)
89 INTEGER STC385(8)
90 INTEGER STC393(8)
91 INTEGER STC401(8)
92 INTEGER STC409(8)
93 EQUIVALENCE (SYM(1), SYN 1(1))
94 EQUIVALENCE (SYM(9), SYN 9(1))
95 EQUIVALENCE (SYM(17), SYN 17(1))
96 EQUIVALENCE (SYM(25), SYN 25(1))
97 EQUIVALENCE (SYM(33), SYN 33(1))
98 EQUIVALENCE (SYM(41), SYN 41(1))
99 EQUIVALENCE (SYM(49), SYN 49(1))
100 EQUIVALENCE (SYM(57), SYN 57(1))
101 EQUIVALENCE (SYM(65), SYN 65(1))
102 EQUIVALENCE (SYM(73), SYN 73(1))
103 EQUIVALENCE (SYM(81), SYN 81(1))
104 EQUIVALENCE (SYM(89), SYN 89(1))
105 EQUIVALENCE (SYM(97), SYN 97(1))
106 EQUIVALENCE (SYM(105), SYN 105(1))
107 EQUIVALENCE (SYM(113), SYN 113(1))
108 EQUIVALENCE (SYM(121), SYN 121(1))
109 EQUIVALENCE (STB(1), STC 1(1))
110 EQUIVALENCE (STB(9), STC 9(1))
111 EQUIVALENCE (STB(17), STC 17(1))
SUBROUTINE SYMBO(XA,YA,H,ASC,TH,NC)

XA STARTING X-COOD OF LOWER LEFT OF STRING (USER INCHES)
YA STARTING Y-COOD OF LOWER LEFT OF STRING (USER INCHES)
H HEIGHT OF CHARACTERS IN STRING (USER INCHES)
ASC CHARACTER STRING
TH ANGLE OF ORIENTATION (DEGREES)
NC +N: NUMBER OF CHARACTERS
-1: PLOT ONLY SINGLE CHARACTER
-2: DRAW LINE TO POINT AND PLOT SYMBOL

REV 08 FEB 82: CHANGE FILE NAME TO 'SYMBOL'
REV 12 MAY 83: SUBROUTINE NAME CHANGED TO 'SYMBO'
REV 11 JUN 83: CHANGE CALL TO SUBROUTINE 'PXPCG'
REV 01 SEP 83: DRAW CHARACTERS FOR THE HP-2648A GRAPHICS TERMINAL

MODIFIED TO RUN ON THE FOURIER SYSTEM
COMMON STATEMENT REMOVED
A CALL TO SUBROUTINE 'PXPCG' ADDED

REAL ASC(1),RE(2)
INTEGER SYM(128), STC(416)
INTEGER SYN 1(8)
INTEGER SYN 9(8)
INTEGER SYN 17(8)
INTEGER SYN 25(8)
INTEGER SYN 33(8)
INTEGER SYN 41(8)
INTEGER SYN 49(8)
INTEGER SYN 57(8)
INTEGER SYN 65(8)
INTEGER SYN 73(8)
INTEGER SYN 81(8)
INTEGER SYN 89(8)
INTEGER SYN 97(8)
INTEGER SYN 105(8)
INTEGER SYN 113(8)
INTEGER SYN 121(8)
INTEGER STC 1(8)
INTEGER STC 9(8)
INTEGER STC 17(8)
INTEGER STC 25(8)
INTEGER STC 33(8)
INTEGER STC 41(8)
INTEGER STC 49(8)
INTEGER STC 57(8)
INTEGER STC 65(8)
INTEGER STC 73(8)
INTEGER STC 81(8)
INTEGER STC 89(8)
INTEGER STC 97(8)
INTEGER STC 105(8)
INTEGER STC 113(8)
INTEGER STC 121(8)
SUBROUTINE WHERE(XP, YP, SC)

C XP PRESENT X COORDINATE (USER UNITS)
C YP PRESENT Y COORDINATE (USER UNITS)
C SC PRESENT SCALE FACTOR

REV 08 FEB 82: CHANGE FILE NAME TO 'WHERE'

REAL RE(2)

GET PRESENT VALUES OF PX, PY, & SF
CALL PXPCO(1,LU,RE,OX,OY,SC,PX,YP)

GET X,Y VALUE IN USER INCHES & SCALE FACTOR
XP=PX
YP=PY
SC=SF
RETURN
END
SUBROUTINE FACTO(SC)

SC SCALE FACTOR

REV 08 FEB 82: CHANGE FILE NAME TO 'FACTO'

REV 13 MAY 83: CHANGE SUBROUTINE NAME TO 'FACTO'

REAL RE(2)

GET VALUE OF SF

CALL PXPCO(1,LU,RE,OX,OY,SY,PX,PY)

UPDATE SCALE FACTOR. (CONVERT USER UNITS TO MACHINE INCHES)

IF(SC. GT. 0.)SF=SC*SF
IF(SC. EQ. 1.)SF=1.

SAVE NEW VALUE OF SF

CALL PXPCO(0,LU,RE,OX,OY,SY,PX,PY)

RETURN

END
SUBROUTINE PXPC0(JST, LU, RE, OX, OY, SF, PX, PY)
REAL RE(2), VARS(7)

C JST = 0: STORE VALUES IN ARRAY VARS
C 1: RETRIVE VALUES FROM ARRAY VARS

CSTORE VALUES
IF(JST.NE.0) GO TO 10
IUNIT=LU
VARS(1)=RE(1)
VARS(2)=RE(2)
VARS(3)=OX
VARS(4)=OY
VARS(5)=SF
VARS(6)=PX
VARS(7)=PY
RETURN

CUPDATE VALUES
LU=IUNIT
RE(1)=VARS(1)
RE(2)=VARS(2)
OX=VARS(3)
OY=VARS(4)
SF=VARS(5)
PX=VARS(6)
PY=VARS(7)
RETURN
END

FTN40: COMPILER: HP92834 REV.2226 (820503)

** NO WARNINGS ** NO ERRORS ** PROGRAM: 104 COMMON: 'NONE'
112 WRITE(LU,520) IEAND,LP5,IDSP,IEAND,LP5,LU0,ICSF
113 WRITE(LU,520) IESTR,LDA
114 520 FORMAT(7F2)
115 C
116 C STORE VALUES
117 700 CALL PXPCO(0,LU,RE,OY,OF,SF,PX,PY)
118 C
119 RETURN
120 END

FTN4X COMPILER: HP92834 REV.2226 (820503)

++ NO WARNINGS ++ NO ERRORS ++ PROGRAM: 390 COMMON: (NONE)
57 C PEN DOWN, DRAW LINE
58 100 CONTINUE
59 IF(LU.EQ.1) GO TO 115
60 WRITE(LU,110) IX,IY
61 110 FORMAT("PD;PA",I6,"",I6)
62 GO TO 120
63 115 WRITE(LU,116) IESTR,ILPLB,IX,IY,IZSP
64 116 FORMAT(2R2,I6","",I6,R2)
65 120 CONTINUE
66 PX=XA
67 PY=YA
68 C DEFINE NEW ORIGIN IN MACHINE INCHES IF REQUESTED
69 IF (IA. GT. 0) GO TO 700
70 PX=0.0
71 PY=0.0
72 OX=AV1
73 OY=AV2
74 GO TO 700
75 C PEN UP, MOVE TO NEXT X, Y COORDINATE
76 200 CONTINUE
77 IF(LU.EQ.1) GO TO 220
78 WRITE(LU,210) IX,IY
79 210 FORMAT("PU;PA",",",I6)
80 GO TO 120
81 220 WRITE(LU,230) IESTR,ILPLA,IX,IY,IZSP
82 230 FORMAT(2R2,I6","",I6,R2)
83 GO TO 120
84 C END OF PLOTS
85 600 CONTINUE
86 IF(LU.NE.1) GO TO 700
87 C RESET TERMINAL
88 WRITE(LU,520) IESC:
89 GO TO 700
90 C ADVANCE THE PAPER ON THE HP9872T PLOTTER
91 IF(IPAGE.EQ.10) CALL HPPAG(1)
92 IF(IPAGE.LT.0) CALL HPPAG(2)
93 GO TO 700
94 C FULL PAGE ADVANCE
95 500 CONTINUE
96 IF(LU.EQ.1) GO TO 510
97 C HALF PAGE ADVANCE
98 C ADVANCE THE PAPER ON THE HP9872T PLOTTER
99 C FULL PAGE ADVANCE
100 C ADVANCE THE PAPER ON THE HP9872T PLOTTER
101 C FULL PAGE ADVANCE
102 IF(IPAGE.EQ.10) CALL HPPAG(1)
103 C HALF PAGE ADVANCE
104 C HALF PAGE ADVANCE
105 IF(IPAGE.LT.0) CALL HPPAG(2)
106 GO TO 700
107 C ADVANCE THE PAPER ON THE HP9872T PLOTTER
108 510 CONTINUE
109 C HARD COPY, CLEAR SCREEN (GRAPHIC DISPLAY)
SUBROUTINE PLOT(XA,YA,IA)

XA X COORDINATE (USER UNITS)
YA Y COORDINATE (USER UNITS)
IA PEN CONTROL

+999 END OF PLOTS
+11 END OF PLOTS (RESET TERMINAL - FIRST PRESS)
+10 HARD COPY, CLEAR SCREEN (TERMINAL)
+10 ADVANCE PAPER ONE FULL PAGE
-10 ADVANCE PAPER ONE HALF PAGE
+3 MOVE TO <XA, YA> PEN UP
+2 MOVE TO <XA, YA> PEN DOWN
-2 MOVE TO <XA, YA> PEN DOWN SET ORIGIN TO <XA, YA>
-3 MOVE TO <XA, YA> PEN UP SET ORIGIN TO <XA, YA>

ANY OTHER VALUE OF IA IS TREATED AS A NOP

REV 02 FEB 82: CHANGE FILE NAME TO 'PLOT'
REV 22 JUN 83: SUPPORT HP-9872T PLOTTER

REV 01 SEP 83: SUPPORT HP-2648A GRAPHICS TERMINAL
MODIFIED TO RUN ON THE FOURIER SYSTEM
COMMON STATEMENT REMOVED
CALLS TO SUBROUTINE 'FXPCO' ADDED

INTEGER IA, ICO(6)
REAL AV(2), PE(2)

EQUIVALENCE (AV(1), AV1), (AV(2), AV2)
DATA NCO=-6, ICO=2, 3, -2, -3, 10, 11/
DATA IESTR=.015452B/, ILPLA/.070141B/, ILPLB/.070142B/
DATA IXSP/.055040B/, LDA/.062101B/, ICSP/.041440B/
DATA LU0/.072465B/, IEAND/.015448B/, LP5/.070658B/
DATA IDSP/.042040B/, IESCG/.155476B/, IXSF/.055040B/

RETREIVE VALUES
CALL FXPCO(I1, LU, FE, OX, OY, SF, PX, PY)

CONVERT FROM USER UNITS TO MACHINE INCHES AND UNITS
IE=0
IF(XA.EQ.-10) IA=10
IF (10. EQ. 999) IA=11
AV1=SF*XA+OX
AV2=SF*YA+OY
X=AV1/RE(1)
IX=X+.5
Y=AV2/RE(2)
IY=Y+.5

IDENTIFY PEN CONTROL OPTION
DO 30 11=1, NCO
IF (10. EQ. ICO<11>) GO TO <100, 200, 100, 200, 500, 600>, 11
CONTINUE
GO TO 700
SUBROUTINE NUMBE (X, Y, HGH T, Z, T, N)

X, Y  COORDINATES OF THE LOWER LEFT CORNER OF THE FIRST
DIGIT OF OUTPUT.  X, Y IS IN FLT. PT. <USER INCHES>

HGH T  HEIGHT OF THE FLOTTED NUMBER IN FLT. PT. <USER INCHES>

Z  FLT. PT. NUMBER TO BE PLOTTED

T  ORIENTATION ANGLE FOR THE NUMBER

N  NUMBER OF DECIMAL DIGITS FOR OUTPUT

N=-1 WILL SUPPRESS THE DECIMAL POINT

REV 08 FEB 82: CHANGE FILE NAME TO 'NUMBER'

REV 08 FEB 82: FIX CALF OF FPN WHEN N. GE. 0

REV 12 MAY 83: CHANGE SUBROUTINE NAME TO 'NUMBE'

REV 12 MAY 83: CHANGE CALL TO SUBROUTINE 'SYMBO'

REV 26 JUL 83: CHANGED 'ALOG10' TO 'ALOGT'

FPN=ABS(Z)

IF (N. GE.0) FPN=FPN+.5*10.**(-N)

XT=X

YT=Y

IF (Z) 10, 90, 20

DRAW MINUS SIGN FOR NEGATIVE NUMBER

CALL SYMBO (XT, YT, HGH T, 1H-, T, 1)

CALL WHERE (XT, YT, ZT)

GET NO. OF DIGITS TO LEFT OF DECIMAL POINT

I=ALOGT(FPN)+1.000**1

IF (I) 50, 50, 30

DO 40 J = 1, 1

K = FPN+10.0**(-J-1)

CALL SYMBO (XT, YT, HGH T, FLOAT (K+48), T, -1)

FPN = FPN - FLOAT (K+10**(-I-J))

CALL WHERE (XT, YT, ZT)

DRAW DIGITS TO LEFT OF DEC. PT.

DO 40 J = 1, 1

K = FPN+10.0**(-J-1)

CALL SYMBO (XT, YT, HGH T, FLOAT (K+48), T, -1)

CALL WHERE (XT, YT, ZT)

CALL DECIMAL POINT IF REQUESTED


CALL SYMBO (XT, YT, HGH T, 1H., T, 1)

CALL WHERE (XT, YT, ZT)

DRAW DIGITS TO RIGHT OF DECIMAL POINT IF REQUESTED

DO 40 J = 1, N

K=FPN+10.0

CALL SYMBO (XT, YT, HGH T, FLOAT (K+48), T, -1)

CALL WHERE (XT, YT, ZT)

FPN=FPN+10.0-FLOAT(K)

RETURN

SPECIAL CASE IF NUMBER = 0.0

CALL SYMBO (XT, YT, HGH T, 12H0.0000000000, T, 2+N)

RETURN

END
**FTN4X** COMPILER: HP92834 REV.2226 (820503)

++ NO WARNINGS ++ NO ERRORS ++ PROGRAM: 241 COMMON: 'NONE'
SUBROUTINE SCALE (X, S, N, K)

C ARRAY OF DATA TO BE SCANNED FOR MAXIMUM AND MINIMUM
C VALUES. ADJUSTED MINIMUM VALUE STORED IN X(N+K+1).
C ADJUSTED X(MAX-MIN) STORED IN X(N+K+1).
C LENGTH OVER WHICH THIS DATA IS TO BE PLOTTED. <MACHINE INCHES>
C NUMBER OF DATA POINTS IN THE ARRAY X.
C REPEAT CYCLE OF MIXED ARRAY (NORMALLY 1).

REV 26 JUL 83: CHANGED 'ALOG10' TO 'ALOGT'

REAL X(1), XM(5)
DATA XM/2.0, 4.0, 5.0, 8.0, 10.0/

C GET SIZE OF INPUT MATRIX
NF=N+K
L=NF+1
J=L+1

C GET MIN, MAX VALUES IN X ARRAY
XMAX=X(1)
XMIN=XMAX
DO 100 I=1, NF, N
XI=X(I)
XMAX=MAX1(XMAX, XI)
XMIN=MIN1(XMIN, XI)
IF (S.LE.0.0) GO TO 210

C GET RATIO OF USER RANGE: AXIS LENGTH (USER UNITS)
DX=(XMAX-XMIN)/5
IF(DX.LE.0.0) GO TO 210

C ROUND MINIMUM AXIS VALUE TO A CONVENIENT NUMBER
IDX=ALOGT(DX)
XI=10.0**IDX
IF(XMIN)110, 140, 120

C ROUND TIC DISTANCE TO A CONVENIENT NUMBER
T=(XMAX-XMIN)/8
IF(T.LE.0.0) GO TO 210
DX=ALOGT(T)
IDX=DX
XMAX=1.0
DX=10.0**<DX-IDX>

IF(DX-1.0) 160, 200, 170
DX=10.0*DX
IDX=IDX+1
DO 150 160 1=1, 5

51
XI=XMIN(I)

IF(DX.LE.XMIN(I)) GO TO 190

CONTINUE

XMAX=XI

X(J)=XMAX*10.0**IDX

RETURN

C

UNSCALABLE - SET STARTING POINT IN X(L), TIC DIST=1 IN X(J)

X(J)=1.0

X(L)=XMIN-0.5

RETURN

END

FTN4X COMPILER: HP92834 REV.2226 (620503)

** NO WARNINGS ** NO ERRORS ** PROGRAM: 285 COMMON: (NONE)
SUBROUTINE AXIS (X0, Y0, A0, NO, SO, T0, CO, DO)

X0  X COORDINATE OF START OF AXIS (USER INCHES)
Y0  Y COORDINATE OF START OF AXIS (USER INCHES)
A0  CHARACTER STRING TO DESCRIBE AXIS (MSB, LSB)
NO  NUMBER OF CHARACTERS IN STRING
- ON CLOCKWISE SIDE OF AXIS (NORMAL FOR X)
+ ON COUNTER CLOCKWISE SIDE OF AXIS (NORMAL FOR Y)
SO  LENGTH OF AXIS (USER INCHES)
T0  ANGLE OF AXIS TO X AXIS OF PAPER (DEGREES)
0.0 FOR X-AXIS
90.0 FOR Y-AXIS
C0  COORDINATE OF MINIMUM TICK ON AXIS (PROBLEM UNITS)
NORMALLY PROVIDED BY 'SCALE', FOLLOWING THE LAST POINT
IN THE DATA ARRAY TO BE PLOTTED
D0  DISTANCE BETWEEN TICKS (PROBLEM UNITS)
NORMALLY PROVIDED BY 'SCALE', FOLLOWING NO
REAL A0(1), RE(2)

CALL PXPCO(1, LU, RE, X0, Y0, SF, PX, PY)

N1=IABS(NO)
N2=60.0/50
B1=0.5*SO-.05*FLOAT(N1+7)
B2=0.12*FLOAT(N1+1)

CONVERT DEGREES TO RADIANS & SET UP COORD ROTATION

T1=T0*0.017453294
T3=COS(T1)
T4=SIN(T1)

IF(NO.LT.0) GO TO 10

B3=.3675
B4=.18
T2=T0
T5=.1
GO TO 20

10 B3=-.4375
B4=-.25
T2=T0
T5=-.1

GET X,Y COMPONENTS OF TIC MARK & START POINT

T6=T5*T3
T5=T5*T4
X1=X0
Y1=Y0
57 C DRAW TIC MARK AND NEXT SEGMENT OF AXIS
58 C
59 DO 100 I=1,N2
60 X2=XI-T5
61 Y2=YI+T6
62 CALL PLOT (X2, Y2, 3)
63 CALL PLOT (XI, YI, 2)
64 XI=XI+T3
65 YI=YI+T4
66 100 CALL PLOT (XI, YI, 2)
67 C
68 C DRAW LAST TIC MARK
69 X2=XI-T5
70 Y2=YI+T6
71 CALL PLOT (X2, Y2, 2)
72 C
73 C GET VALUES FOR TIC LABELS: DIFF, END, EXPONENT
74 DI=DO
75 CI=C0*N2*D1
76 EI=0.0
77 IF (DI.EQ.0.0) GO TO 140
78 C
79 C CHANGE EXPONENT IF VALUE IS GREATER THAN 100
80 110 IF(DI.LT.100.0) GO TO 130
81 D1=D1*0.1
82 CI=CI*0.1
83 EI=EI+1.0
84 GO TO 110
85 120 D1=D1*10.0
86 CI=CI*10.0
87 EI=EI-1.0
88 C
89 C CHANGE EXPONENT IF VALUE IS LESS THAN .01
90 130 IF(D1.LT.0.01) GO TO 120
91 C
92 C LAST TIC + DISTANCE AWAY FROM AXIS + 1 CHAR TO LEFT
93 140 X2=XI-B4*T4-.0857*T3
94 Y2=Y1+E4*T3-.0857*T4
95 N2=N2+1
96 C
97 C
98 C DEFINE HEIGHT OF CHARACTERS
99 HIGHT=0.10
100 C
101 C CHANGE CHARACTER SIZE FOR THE TERMINAL
102 IF(LU.EQ.1) HIGHT=0.15
103 C
104 C LABEL EACH TIC MARK
105 C
106 DO 150 I=1,N2
107 CALL NUMBE (X2, Y2, HIGHT, CI, T2, 2)
108 CI=CI-D1
109 X2=X2-T3
110 150 Y2=Y2-T4
111 C
112 C WRITE LABEL FOR AXIS
113 X2=X0+B1*T3-B3*T4
114 Y2=Y0+B1*T4+B3*T3
115 CALL SYMBO (X2, Y2, HIGHT, A0, T2, N1)
116 C
117 C WRITE EXPONENTIAL SCALE FACTOR IF REQUIRED
118 IF (E1.EQ.0.0) RETURN
119 X2=X2+B2*T3
120 Y2=Y2+B2*T4
121 IF(T2.EQ.0.0 .AND. LU.EQ.1) X2=X2+.10
122 IF(T2.EQ.90.0 .AND. LU.EQ.1) Y2=Y2+.10
123 CALL SYMBO (X2, Y2, HIGHT, 6H(X10), T2, 6)
124 X2=X2+.48*T3-.07*T4
125 Y2=Y2+.48*T4+.07*T3
126 IF(T2.EQ.0.0 .AND. LU.EQ.1) X2=X2+.20
127 IF(T2.EQ.90.0 .AND. LU.EQ.1) Y2=Y2+.20
128 CALL NUMBE (X2, Y2, HIGHT, E1, T2, -1)
129 RETURN
130 END

FTN4X COMPILER: HP92834 REV. 2226 (820503)

** NO WARNINGS ** NO ERRORS ** PROGRAM: 647 COMMON: (NONE)
SUBROUTINE LINE(X, Y, N, K, J, L)

C X ARRAY OF ABCISSA VALUES (USER UNITS)
C Y ARRAY OF ORDINATE VALUES (USER UNITS)
C N NUMBER OF POINTS IN THE ARRAY
C K REPEAT CYCLE OF A MIXED ARRAY (NORMALLY = 1)
C J >0, SYMBOL AND LINE
C =0, LINE ONLY
C <0, SYMBOL ONLY
C L NUMBER OF SYMBOL, SEE SYMBOL ROUTINE FOR LIST
C 0-17 ARE CENTERED SPECIAL SYMBOLS
C
REV 08 FEB 82: FILE NAME CHANGED TO 'LINE'
REV 13 MAY 83: CHANGE CALL TO SUBROUTINE 'SYMBO'

C THIS ROUTINE EXPECTS XMIN, DX, YMIN AND DY TO BE STORED IN
C X(N*K+1), X(N*K+1+K), Y(N*K+1), AND Y(N*K+1+K) RESPECTIVELY
C
REAL X(1), Y(1)
IF (N.LT. 1) GO TO 110

C CHECK FOR VALID SYMBOL
AL=L
IF(AL.LE.0.0 .OR. AL.GT.127.) AL=0.0

C GET START POINT, SCALING, END OF DATA
I0=N*K+1
II=I0+K
I2=I0-K
XMIN=X(I0)
DX=X(II)
YMIN=Y(I0)
DY=Y(II)

C SET UP INDICES AND FLAGS
I2=1
I3=3
IF(J.GE.0) I3=2
I4=3
I5=N

C CONVERT NEXT X,Y LOCATION TO USER INCHES & MOVE PEN
X1=(X(I2)-XMIN)/DX
Y1=(Y(I2)-YMIN)/DY
CALL PLOT (X1, Y1, I4)
I4=I3

C PLOT SYMBOL IF REQUESTED
IF (J.NE.0) CALL SYMBO (XI, Y1, 0.14, AL, 0.0, -1)
I2=I2+K
I5=I5-1
IF (I5.GT.0) GO TO 100

RETURN
END
PAGE 2  LINE OPTS: LHI  3:30 PM WED., 26 DEC., 1984

FTN4X COMPILER: HP92834 REV.2226 (820503)

++ NO WARNINGS ++ NO ERRORS ++ PROGRAM: 178 COMMON: (NONE)
SUBROUTINE PXPCG (I0, I1, I2)

THIS ROUTINE EXTRACTS CHARACTER #I1 FROM STRING I0
(RHS FIRST, LHS SECOND)

INTEGER I0 (1), I1, I2

GET INDEX -1 OF WORD CONTAINING CHAR
JW=(I1-1)/2

WITHIN WORD INDEX OF DESIRED CHAR (1 OR 2)
JB=I1-JW-JW

PULL TWO CHARACTERS FROM STRING
JW=I0(JW+1)

SHIFT FROM LHS IF SECOND CHARACTER IS DESIRED
IF (JB.NE.1) JW=JW/2**8

RETURN DESIRED CHAR IN RHS OF WORD
I2=IAND(JW, 255)
RETURN

END

**NO WARNINGS** **NO ERRORS** **PROGRAM: 52** COMMON: <NONE>
SUBROUTINE CGET(10, 11, 12)

C THIS ROUTINE EXTRACTS CHARACTER #11 FROM STRING 10
C (LHS FIRST, RHS SECOND)

C RE: 06 FEB 82: CHANGE FILE NAME TO 'CGET'

INTEGER 10(11), 11, 12
JW = 11 - 1
10
JB = 11 - JW
11
JW = 10 + 1
12
IF JB .NE. 2 .LT. JW .GT. 8
14
12 = 14 .AND. JW .LT. 8
16
RETURN
14
END

____ Compiler: HPF3314 REV.2226 (820503)

-- NO WARNING -- NO ERROR -- PROGRAM: 52 COMMON: (NONE)
SUBROUTINE HPCDIR(TH)

THIS ROUTINE SPECIFIES THE ABSOLUTE DIRECTION IN WHICH CHARACTERS ARE LETTERED

TH - ANGLE BETWEEN CHARACTER BASE AND MACHINE X AXIS (DEGREES)

INTEGER ICHAR(10)

DEFINE ABSOLUTE CHARACTER DIRECTION COMMAND

CALL HPOLIT(1HD)

CALL HPOUT(1HI)

SEND RUN VALUE

RUN=COS(TH*.0174532925)

CALL FP2AS(RUN,10,1CHAR,N)

N=MIN(N+4,10)

DO 100 I=1,N

CALL HPOUT(ICHAR(I))

CALL HPOUT(1H,)

SEND RISE VALUE

RISE=SIN(TH*.0174532925)

CALL FP2AS(RISE,10,1CHAR,N)

N=MIN(N+4,10)

DO 200 I=1,N

CALL HPOUT(ICHAR(I))

CALL HPOUT(1H,)

RETURN

END

TN4: COMPILER: HP92834 REV. 2226 (820503)

++ NO WARNINGS ++ NO ERRORS ++ PROGRAM: 118 COMMON: (NONE)
SUBROUTINE HFCSZ(N,H)

C THIS ROUTINE SPECIFIES THE SIZE OF CHARACTERS & SYMBOLS
C
C W - WIDTH OF CHARACTER IN INCHES
C H - HEIGHT OF CHARACTER IN INCHES
C
INTEGER ICHAR(10)

C DEFINE CHARACTER SIZE COMMAND
CALL HPOUT(1HS)
CALL HPOUT(1HI)

C CONVERT TO CM & SEND WIDTH VALUE TO PLOTTER
CALL FP2AS(W+2.54,10,ICCHAR,N)
N=MIN0(N+4,10)
DO 100 1=1,N
100 CALL HPOUT(ICCHAR+I)

C CALL HPOUT(1H.)

C SEND HEIGHT VALUE
C CONVERT TO CM & SEND HEIGHT VALUE TO PLOTTER
CALL FP2AS(H+2.54,10,ICCHAR,N)
N=MIN0(N+4,10)
DO 200 1=1,N
200 CALL HPOUT(1CHAR+1)

C
CALL HFOK(IH.)
RETURN.
END

* COMPILED: HP2050 FRE JUNE 320501
* NO WARNING: -- NO ERRORS: -- PROGRAM: 111 COMMON: (NONE)
SUBROUTINE HPOUT(ICHAR)
C
THIS ROUTINE SENDS OUTPUT TO THE HF-9872T PLOTTER
C
ICHAR - CHARACTER TO OUTPUT TO PLOTTER
C
REV 01 SEP 83: MODIFIED TO RUN ON THE FOURIER SYSTEM
C
: COMMON STATEMENT REMOVED
C
: A CALL TO SUBROUTINE 'PXFCO' ADDED
C
REAL RE(2)
INTEGER IC(100)
DATA ITERM'/035440E'
C
GET UNIT NUMBER
CALL PXFCO('1,LU,RE,0:,0Y,6F,PI,PI)
C
SAVE CHARACTERS FOR OUTPUT TO PLOTTER
I=I+1
N=1
IC(I)=ICHAR
C
CHECK FOR TERMINATOR
IF(ICHAR.EQ. ITERM) I=0
IF(I.NE.0) RETURN
C
OUTPUT CHARACTERS TO PLOTTER
IEND=N-1
WRITE(LU,130) (IC(J),J=1,IEND)
130 FORMAT(100A1)
C
SEND PEN UP COMMAND TO PLOTTER
WRITE(LU,150)
150 FORMAT("PU")
C
RETURN
END

**X COMPILER: HP92834 REV.2226 (820503)**

NO WARNINGS ++ NO ERRORS ++ PROGRAM: 204 COMMON: (NONE)
### FUNCTION UPDATA-7K02

1. ISSR X4SYLB Y 0
2. Y7001 Y7001 Y 0
3. HPPEN XFLTPK Y 0
4. WHERE XFLTPK Y 0
5. PXPCG XFLTPK Y 0
6. CGET XFLTPK Y 0
7. FACTO XFLTPK Y 0
8. HPDSZ XFLTPK Y 0
9. HPCDR XFLTPK Y 0
10. HPPAG XFLTPK Y 0
11. PXPCO XFLTPK Y 0
12. LINE XFLTPK Y 0
13. HPOUT XFLTPK Y 0
14. PLOTS XFLTPK Y 0
15. FP2AS XFLTPK Y 0
16. NUMBE XFLTPK Y 0
17. SCALE XFLTPK Y 0
18. PLOT XFLTPK Y 0
19. AXIS XFLTPK Y 0
20. SYMBO XFLTPK Y 0
21. Y9000 Y9000 Y 0

<table>
<thead>
<tr>
<th>RESS</th>
<th>PGMNAM</th>
<th>LIBNAM</th>
<th>CONTROL</th>
<th>LENGTH</th>
</tr>
</thead>
<tbody>
<tr>
<td>42</td>
<td>AXIS</td>
<td>%FLTPK</td>
<td>Y</td>
<td>01265</td>
</tr>
<tr>
<td>27</td>
<td>Y7001</td>
<td>Y7001</td>
<td>Y</td>
<td>00042</td>
</tr>
<tr>
<td>77</td>
<td>ISSP</td>
<td>X4SYLB</td>
<td>Y</td>
<td>00007</td>
</tr>
<tr>
<td>60</td>
<td>Y9000</td>
<td>Y9000</td>
<td>Y</td>
<td>02237</td>
</tr>
<tr>
<td>57</td>
<td>PLOT</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00017</td>
</tr>
<tr>
<td>56</td>
<td>SCALE</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00075</td>
</tr>
<tr>
<td>55</td>
<td>HPPAG</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00201</td>
</tr>
<tr>
<td>54</td>
<td>HPPEN</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00356</td>
</tr>
<tr>
<td>53</td>
<td>SYMBO</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00012</td>
</tr>
<tr>
<td>35</td>
<td>NUMBE</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00077</td>
</tr>
<tr>
<td>52</td>
<td>FP2AS</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00370</td>
</tr>
<tr>
<td>33</td>
<td>PLOTS</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00356</td>
</tr>
<tr>
<td>71</td>
<td>HPOUT</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00313</td>
</tr>
<tr>
<td>64</td>
<td>CGET</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00074</td>
</tr>
<tr>
<td>60</td>
<td>LINE</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00273</td>
</tr>
<tr>
<td>73</td>
<td>PXPCO</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00256</td>
</tr>
<tr>
<td>51</td>
<td>HPCDR</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00170</td>
</tr>
<tr>
<td>41</td>
<td>HPDSZ</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00163</td>
</tr>
<tr>
<td>24</td>
<td>FACTO</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00111</td>
</tr>
<tr>
<td>35</td>
<td>PXPCG</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00074</td>
</tr>
<tr>
<td>31</td>
<td>WHERE</td>
<td>%FLTPK</td>
<td>Y</td>
<td>00057</td>
</tr>
</tbody>
</table>
COF A T=00004 IS ON CF00022 USING 00006 BLIS P=0000

01 @
02 A, OVRPLIK
03 T, SNF51C
04 INSERT Y COMMAND NUMBERS BEFORE 'OE', AN EXAMPLE:
05 '0,0077', WHERE '0077'=4-DIGIT Y COMMAND NUMBER
06 @.7000
07 @.9000
08 @
09 CSORT+
10 INSERT PROGRAM LOADS BETWEEN 'CSORT+' AND 'CSORT-
11 EXAMPLE: 'F,Y77J0B', WHERE 'Y77J0B'= FILE NAME
12 'Y.Y0077', WHERE 'Y0077'= PROG NAME
13 F, Y7001
14 Y, Y7001
15 F, Y9000
16 Y, Y9000
17 F, XPTPK
18 Y, AXIS
19 F, XPTPK
20 Y, SCALE
21 F, XPTPK
22 Y, LINE
23 F, XPTPK
24 Y, PLOTS
25 F, XPTPK
26 Y, PLOT
27 F, XPTPK
28 Y, HPFAG
29 F, XPTPK
30 Y, NUMBE
31 F, XPTPK
32 Y, SYMBO
33 F, XPTPK
34 Y, PCIPO
35 F, XPTPK
36 Y, FACTO
37 F, XPTPK
38 Y, HCORF
39 F, XPTPK
40 Y, HPFLT
41 F, XPTPK
42 Y, HPFEN
43 F, XPTPK
44 Y, PCIPO
45 F, XPTPK
46 Y, WHERE
47 F, XPTPK
48 Y, CGET
49 F, XPTLFA
50 Y, HPCSZ
51 F, XPTPK
52 Y, FP2AS
53 F, Y4SYLB
54 Y, ISSP
55 CSORT-
56 F, N3500A
57 L,**
58 F, N3501A
59 L,**
60 F, A3060B
61 L,**
62 U,**
63 E, 01
64 **
LISTINGS OF THE COMMAND AND THE OVERLAY GENERATION

OF THE GENERAL PLOT PACKAGE EVALUATION PROGRAM Y9000
END

FTN4 COMPILER: HP92060-16092 FEV. 2026 (800423)

++ NO WARNINGS ++ NO ERRORS ++ PROGRAM = 01183   COMMON = 00000
DEFINE FOURIER DATA IN BLOCK ZERO

CALL KYBD(2HIS,IBS)
CALL KYBD(2HCL,IBLOK)
CALL KYBD(2HPA)

GET SYSTEM BLOCK SIZE
CALL GETO(IBLOK,NQUAL)

IF(NQUAL(5).NE.0) WRITE(6,30) (NQUAL(J),J=1,5)

FORMAT("ERROR",5(16,4X))
BS=NQUAL(1)

PUT FOURIER DATA INTO X,Y ARRAYS
DO 5 I=1,BS
II=I-1
CALL GET(IBLOK,II,DATA1,DATA2)
Y(I)=DATA1
CALL KYBD(2HY,1800,1,II)
CALL KYBD(2HY,1828,2000,1,IBLOK,1)
CALL VGETF(2000,IBUF)
X(I)=XX
CONTINUE

MANUALLY LOAD PAPER ON THE PLOTTER
IF(IPAG.EQ.0 .AND. LU.EQ.0.30) CALL HPPAG(3)

PLOT THE DATA

TURN ON BITS 0-5 OF S-REG TO SIGNAL START OF OPERATION
SOME OF THESE BITS WILL BE TURNED OFF DURING THE OPERATION.
ALL BITS ON EXCEPT 0-5 - SIGNALS COMPLETION OF PLOT OPERATION.

CALL ISSR(77B)
CALL PLOT(1.0,1.0,-3)
CALL SCALE(X,10.0,BS,1)
CALL SCALE(Y,2.0,BS,1)
CALL LINE(X,Y,BS,1,0,0)

DRAW AND LABEL AXES

TURN ON BITS 0-1 OF S-REG TO SIGNAL START OF OPERATION

CALL ISSR(7B)
CALL AXIS(0.0,0.0,8HABSCISSA,-8,10.0,0.0,X(8S+1),X(8S+2))
CALL AXIS(0.0,0.0,8ORDINATE,+8.6,90.0,Y(8S+1),Y(8S+2))

TURN OFF BITS 0-1 OF S-REG TO SIGNAL END OF AXES OPERATION

CALL ISSR(8)

SIGNAL END OF PLOT OPERATION

CALL ISSR(177700B)

GET A HARDCOPY

CALL PLOT(0.0,0.0,IP)

RESET TERMINAL

CALL PLOT(0.0,0.0,11)
SUBROUTINE Y9000

THIS ROUTINE PROVIDES A MEANS OF EVALUATING THE GENERAL PLOT SUBROUTINE PACKAGE ON THE FOURIER SYSTEM.

REAL X(130), Y(130)
INTEGER Nitual(5), IBUF(2), BS
EQUIVALENCE(XX, IBUF)
DATA IBELL/3400B/

IBLOK=0
IBS=128
CLEAR SCREEN AND CLEAR S-REGISTER
CALL Y7001
CALL ISSR(0)
REQUEST INFORMATION FROM USER
WRITE(1,10) IBELL
WRITE(1,15) LU
WRITE(1,18) IBELL
WRITE(1,20) IBELL
WRITE(1,25) LU
WRITE(1,29) IBELL
WRITE(1,29) IBELL
WRITE(1,26) ICHG
WRITE(1,26) ICHG
WRITE(1,26) ICHG
WRITE(1,26) ICHG
WRITE(1,26) ICHG
IF(ICHG.EQ.0) Go To 1
WRITE(1,23) ICHG
READ(1,22) DSF
WRITE(1,23) ICHG
WRITE(1,23) ICHG
WRITE(1,23) ICHG
WRITE(1,23) ICHG
WRITE(1,23) ICHG
READ(1,24) NPEN
WRITE(1,23) ICHG
WRITE(1,23) ICHG
READ(1,24) IPAG
CONTINUE

MODIFY PLOT PACKAGE
CALL PLTS(0.0,LU)
IF(ICHG.NE.0) CALL FACTO(DSF)
IF(ICHG.NE.0) CALL HPFEN(NPEN)
IP=10
IF(IPAG.EQ.1 .AND. LU.EQ.30) IP=-10
LISTING OF THE GENERAL PLOT PACKAGE EVALUATION PROGRAM

AND A PLOT OF THE DATA PRODUCED
57    GO TO 400
58    END

FTN4F COMPILER: HP92834 REV.2226 (820503)

** NO WARNINGS ** NO ERRORS ** PROGRAM: 235 COMMON: (NONE)
SUBROUTINE FP2AS(FP,NASC,ASC,I)

THIS ROUTINE CONVERTS A FLOATING POINT NUMBER TO
AN ASCII CHARACTER STRING

FP - FLOATING POINT VALUE
NASC - NO. OF CHARACTERS TO BE RETURNED IN ASCII STRING
ASC - ARRAY IN WHICH ASCII CHARACTER STRING IS RETURNED
I - NO. OF CHARACTERS TO LEFT OF DECIMAL POINT (OUTPUT)

REV 26 JUL 83: CHANGED 'ALOG10' TO 'ALOGT'

INTEGER ASC(1),IDIG(10)
DATA IDIG/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/
DATA IMI/1H-/,IDF/1H. /

CHECK FOR MINUS SIGN
FPX=ABS(FP)
I=0
IF(NASC.LT.1) RETURN
I=1
IF(FP.GE.0.) GO TO 100
ASC(I)=IMI
IF(I.GE.NASC) RETURN
I=2

CHECK FOR ABSOLUTE VALUE .GT. 1
IF(FPX.GE.1.) GO TO 200
ASC(I)=IDIG(I)
IF(I.GE.NASC) RETURN
I=I+1
GO TO 300

GET NEXT INTEGER DIGIT
IEXP=ALOGT(FPX)+1.001
DO 210 J=1,IEXP
II=FPX/10.**(IEXP-J)
ASC(I)=IDIG(II+1)
IF(I.GE.NASC) RETURN
I=I+1
FPX=FPX-FLOAT(II)*10.**(IEXP-J)

PUT IN DECIMAL POINT
IX=I
I=I-1
ASC(IX)=IDP
IF(IX.GE.NASC) RETURN
IX=IX+1

GET NEXT FRACTION DIGIT
II=FPX*10.
ASC(IX)=IDIG(II+1)
IF(IX.GE.NASC) RETURN
IX=IX+1
FPX=FPX*10.-FLOAT(II)
SUBROUTINE HPPEN(I)

C THIS ROUTINE SELCTS A PEN FOR THE HP-9872T PLOTTER

C I - THE PEN POSITION NUMBER. IT MUST BE AN INTEGER IN THE
C RANGE 0 THROUGH 8.

C 0: DIRECTS THE PEN ARM TO RETURN THE PEN IT IS CURRENTLY
C HOLDING TO ITS STALL

C 1-8: DIRECTS THE PEN ARM TO FETCH THE DESIRED PEN

C REAL RE(2)

C GET UNIT NUMBER
C CALL PXPCO(LU,RE,OX,OY,SF,PX,Py)

C SELECT THE DESIRED PEN
C WRITE(LU,100) 1
C 100 FORMAT("SP",1)
C RETURN
C END

** NO WARNINGS ** NO ERRORS ** PROGRAM: 51 COMMON: (NONE)
SUBROUTINE HPPAG(ISW)
THIS ROUTINE CONTROLS PAGINATION ON THE HP-9872T PLOTTER
ISW 1: FULL PAGE ADVANCE
2: HALF PAGE ADVANCE
3: MANUALLY PLACE SHEET IN POSITION
REAL RE(2)
DATA IBEIL/003440B/
GET UNIT NUMBER
CALL PXPCO(1,LU,RE,OX,OV,SF,PX,PY)
CHECK FOR AUTOMATIC PAGE ADVANCE
IF(ISW.EQ.3) GO TO 250
ADVANCE PAPER
FULL PAGE ADVANCE
IF(ISW.EQ.2) GO TO 150
WRITE(LU,100)
FORMAT("AF")
RETURN
HALF PAGE ADVANCE
WRITE(LU,200)
FORMAT("AH")
RETURN
MANUALLY LOAD SHEET
WRITE(1,300) IBEIL
FORMAT(A2,"PLACE NEW SHEET OF PAPER ON PLOTTER."
"PRESS (SPACE) (RETURN) TO CONTINUE.")
READ(1,350) I
FORMAT(A1)
RETURN
END
**SANTA CLARA DIVISION AUTO OVERLAY GENERATION-8X15**

************
* A.OVRFLAY "OVRFLAY" 54451-14805 REV. A 06 OCT 83*
* T,SNP5IC BASE COMMAND FILE FOR USER OVERLAY GENERATION*

************

*ENTER USERCOMMAND NUMBERS, OR TYPE "?"*

************

* INSERT Y COMMAND NUMBERS BEFORE "&E", AN EXAMPLE:
  "&E,0077", WHERE "0077"=4-DIGIT Y COMMAND NUMBER

* 0,7001

Y7001
Y;COM 01075 01075 00434 00435
+ 0,9000

Y9000
Y;COM 01074 01074 00436 00437
+ 0,E

Y;COM 47440 47440

TYPE "L" TO LOAD USERCOMMAND LIBRARY.

************

* CSOFT+
  INSERT PROGRAM LOADS BETWEEN "CSOFT+" AND "CSOFT-

* EXAMPLE: "F,777JOB", WHERE "777JOB" = FILE NAME
  "Y,Y0077", WHERE "Y0077" = PROG NAME

* F,"PLTFK"
* Y,AXIS

************

AXIS 56442 57726 01066 01073

*LOAD0

************

* 

* F,"Y7001"
* Y,Y7001

************

Y7001 57727 57770 01063 01065

*LOAD0

************

* 

* F,"4SYLB"
* Y,ISSP

************

ISSP 57771 57777

*LOAD0

************

92068-1X019 REV. 2013 750701

*LOAD0
*************
*          
*  F,\%Y9000  
*  Y,\%Y9000  
*************

Y9000  60000  62236  01052  01062

*LOAD*

*************
*          
*  F,\%PLTPK  
*  Y,PLOT  
*************

PLOT  62237  63055  01051  01051

*LOAD*

*************
*          
*  F,\%PLTPK  
*  Y,SCALE  
*************

SCALE  63056  63552  01045  01050

*LOAD*

*************
*          
*  F,\%PLTPK  
*  Y,HPPAG  
*************

HPPAG  63553  63753

*LOAD*

*************
*          
*  F,\%PLTPK  
*  Y,HPPEN  
*************

HPPEN  63754  64031  01043  01044

*LOAD*

*************
*          
*  F,\%PLTPK  
*  Y,SYMBO  
*************

SYMBO  64032  66043  01012  01042
*LOAD*

************
*
* F,%PLTPK
* Y,NUMBE
************

NUMBE  66044 66442 01010 01011

*LOAD*

************
*
* F,%PLTPK
* Y,FP2AS
************

FP2AS  66443 67032

*LOAD*

************
*
* F,%PLTPK
* Y,PLOTS
************

PLOTS  67033 67370

*LOAD*

************
*
* F,%PLTPK
* Y,HPOUT
************

HPOUT  67371 67703

*LOAD*

************
*
* F,%PLTPK
* Y,CGET
************

CGET  67704 67777

*LOAD*

************
*
* F,%PLTPK
* Y,LINELINE
************

LINE  70000 70272
*LOAD@
*************
*
+ F,%PLTPK
+ Y,PXPCO
*************
PXPCO 70273 70550

*LOAD@
*************
*
+ F,%PLTPK
+ Y,HPCDF
*************
HPCDF 70551 70740 01006 01007

*LOAD@
*************
*
+ F,%PLTPK
+ Y,HPCSZ
*************
HPCSZ 70741 71123

*LOAD@
*************
*
+ F,%PLTPK
+ Y,FACTO
*************
FACTO 71124 71234

*LOAD@
*************
*
+ F,%PLTPK
+ Y,PXPCG
*************
PXPCG 71235 71330

*LOAD@
*************
*
+ F,%PLTPK
+ Y,WHERE
*************
WHERE 71331 71407

*LOAD*

************
*          COPT-
* F,N3500A
* L,**     
************
FRMTR 71410 75222 00565 01005
MOD  75223 75242
MXMNI 75243 75356
MXMR

*LX7*
MINI
75357 75464
DELE 75465 75516
SNGL 75517 75561
.XFH 75562 75747
.XCOM 75750 76015 00556 00564
.XFER 76016 76053
.ALOGT 76054 76062

*LOAD*

************
* F,N3501A
* L,**     
************
ABS 76063 76067
.IT0I 76070 76202
.GOTO 76203 76224
.IABS 00442 00450

*LOAD*

************
* F,A30608
* L,**     
************
D.05 " " 00451 00453 00062 00062
D.06 " " 00454 00456 00060 00060
D.33 " " 00457 00461 00063 00063
D.35 " " 00462 00464 00061 00061
D.76 " " 00465 00467 00055 00055
I.05 " " 00470 00471 00046 00046
C.33 " " 00472 00473 00045 00045

*LOAD*

81
*undefined symbols

blank dtad gkplt

*available memory

76225 76224
00474 00555

*load*

************

**e,01**

************

*user links

76225 76233
00556 01777

*overlay symbol table

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>.E10</td>
<td>74637</td>
</tr>
<tr>
<td>.OTA.</td>
<td>74705</td>
</tr>
<tr>
<td>.TAY.</td>
<td>74450</td>
</tr>
<tr>
<td>.I0R.</td>
<td>74315</td>
</tr>
<tr>
<td>.LST.</td>
<td>70101</td>
</tr>
<tr>
<td>.R10.</td>
<td>74235</td>
</tr>
<tr>
<td>.XFER</td>
<td>76025</td>
</tr>
<tr>
<td>.ABS.</td>
<td>76063</td>
</tr>
<tr>
<td>.AMAX1</td>
<td>75357</td>
</tr>
<tr>
<td>.AMIS</td>
<td>75456</td>
</tr>
<tr>
<td>.C.33</td>
<td>00472</td>
</tr>
<tr>
<td>.D.66</td>
<td>00454</td>
</tr>
<tr>
<td>.D.76</td>
<td>00465</td>
</tr>
<tr>
<td>.FACTO</td>
<td>71131</td>
</tr>
<tr>
<td>.HPCDF</td>
<td>70564</td>
</tr>
<tr>
<td>.HPFAG</td>
<td>63560</td>
</tr>
<tr>
<td>.IABS</td>
<td>00445</td>
</tr>
<tr>
<td>.MAX0</td>
<td>75250</td>
</tr>
<tr>
<td>.MOD</td>
<td>75225</td>
</tr>
<tr>
<td>.OLDIO</td>
<td>74662</td>
</tr>
<tr>
<td>.PXFCG</td>
<td>71246</td>
</tr>
<tr>
<td>.SNGL</td>
<td>75517</td>
</tr>
<tr>
<td>.Y0001</td>
<td>57731</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>.DFER</td>
<td>76016</td>
</tr>
<tr>
<td>.GOTO</td>
<td>76203</td>
</tr>
<tr>
<td>.II0.</td>
<td>74223</td>
</tr>
<tr>
<td>.IT01</td>
<td>76070</td>
</tr>
<tr>
<td>.FAR.</td>
<td>74461</td>
</tr>
<tr>
<td>.MAY.</td>
<td>74406</td>
</tr>
<tr>
<td>.XIO.</td>
<td>74252</td>
</tr>
<tr>
<td>.ALOGT</td>
<td>76054</td>
</tr>
<tr>
<td>.AMINO</td>
<td>75255</td>
</tr>
<tr>
<td>.EC5.</td>
<td>70526</td>
</tr>
<tr>
<td>.CGET</td>
<td>67707</td>
</tr>
<tr>
<td>.D.65</td>
<td>00451</td>
</tr>
<tr>
<td>.D.63</td>
<td>00457</td>
</tr>
<tr>
<td>.D.65</td>
<td>00465</td>
</tr>
<tr>
<td>.FF2AS</td>
<td>66461</td>
</tr>
<tr>
<td>.HFCZ</td>
<td>70755</td>
</tr>
<tr>
<td>.HPFEN</td>
<td>63761</td>
</tr>
<tr>
<td>.ISSR</td>
<td>57771</td>
</tr>
<tr>
<td>.BLANK</td>
<td></td>
</tr>
<tr>
<td>.D.05</td>
<td>00451</td>
</tr>
<tr>
<td>.D.35</td>
<td>00462</td>
</tr>
<tr>
<td>.D.05</td>
<td>00462</td>
</tr>
<tr>
<td>.GKPLT</td>
<td></td>
</tr>
<tr>
<td>.HOUT</td>
<td>67542</td>
</tr>
<tr>
<td>.I.05</td>
<td>00470</td>
</tr>
<tr>
<td>.LINE</td>
<td>70066</td>
</tr>
<tr>
<td>.MNH.</td>
<td>75364</td>
</tr>
<tr>
<td>.NEW0</td>
<td>74655</td>
</tr>
<tr>
<td>.NUMEE</td>
<td>66052</td>
</tr>
<tr>
<td>.PLOT</td>
<td>62260</td>
</tr>
<tr>
<td>.FXPCO</td>
<td>70321</td>
</tr>
<tr>
<td>.SYMB</td>
<td>65104</td>
</tr>
<tr>
<td>.Y9000</td>
<td>61017</td>
</tr>
</tbody>
</table>

*links

00556 01777

*available memory

76225 76224
00474 00555

*end

*load*

************
"*$ encountered, overlay generation terminated"