PLOTSPC
A FORTRAN 77 Program for Plotting Spectral Data from
The Varian CARY 2390 UV-VIS-NIR Spectrophotometer

R. A. Binstead

GEO-CENTERS, Inc.,
10903 Indian Head Highway
Fort Washington, MD 20744

J. C. Cooper

Polymeric Materials Branch
Chemistry Division

October 20, 1988
**Title:** PLOTSPEC - A FORTRAN 77 Program for Plotting Spectral Data from the Varian CARY 2390 UV-VIS-NIR Spectrophotometer

**Personal Authors:** Binstead, R.A. and Cooper, J.C.

**Type of Report:** Interim

**Time Covered:** From 10/87 to present

**Date of Report:** October 20, 1988

**Source of Funding Numbers:**

**Program Element No:** 61153N
**Project No:** 01-11-00
**Task No:** 00
**Accession No:** 63224G

**Abstract:**

PLOTSPEC is a FORTRAN 77 program designed to plot UV-VIS-NIR spectra using data obtained from the CARY 2300 - 2400 series spectrophotometers. The program is written to operate on a Hewlett-Packard 1000 computer system but with very few system dependent features to enable easy conversion for operation on other host systems. PLOTSPEC supports a wide range of Hewlett-Packard digital plotters using the HP-GL command language. The program uses a menu driven environment with automatic scaling and axes setup in the...
19. ABSTRACTS (Continued)

Coordinate and abscissa modes of choice. PLOTSPEC offers abscissa choices of Wavelength (nm) or Wavenumbers $x \times 10^{-3}$ and ordinate choices of Absorbance, Extinction Coefficient, % Reflectance or % Transmission. Annotation and digitizing facilities are also included for easy labelling and peak determinations.
## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTRODUCTION</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>IMPLEMENTATION</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>Hardware Interface</td>
<td>2</td>
</tr>
<tr>
<td>1.1</td>
<td>System Handshaking</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>HP-GL Software Control Of HP Plotters</td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>HP-GL Commands Summary</td>
<td>4</td>
</tr>
<tr>
<td>1.4</td>
<td>Default Plotting Parameters</td>
<td>6</td>
</tr>
<tr>
<td>SOFTWARE DESCRIPTION</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.0</td>
<td>Purpose Of PLOTSPEC</td>
<td>7</td>
</tr>
<tr>
<td>2.1</td>
<td>Program Structure</td>
<td>8</td>
</tr>
<tr>
<td>2.2</td>
<td>Plotting Menu Commands</td>
<td>13</td>
</tr>
<tr>
<td>2.3</td>
<td>COMMON DATA</td>
<td>16</td>
</tr>
<tr>
<td>2.4</td>
<td>SUBROUTINES</td>
<td>18</td>
</tr>
<tr>
<td>2.5</td>
<td>FUNCTIONS</td>
<td>23</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>PROGRAM CODE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.0</td>
<td>Source Code Availability</td>
<td>23</td>
</tr>
<tr>
<td>3.1</td>
<td>Variable Names And Usage</td>
<td>23</td>
</tr>
<tr>
<td>3.2</td>
<td>Program Listing</td>
<td>29</td>
</tr>
</tbody>
</table>

| APPENDIX         | Sample Plots                         | 68   |

LIST OF TABLES

<table>
<thead>
<tr>
<th>TABLE</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Data File Format</td>
<td>10</td>
</tr>
<tr>
<td>II</td>
<td>Glossary Of Integer Variables</td>
<td>24</td>
</tr>
<tr>
<td>III</td>
<td>Glossary Of Real Variables</td>
<td>25</td>
</tr>
<tr>
<td>IV</td>
<td>Glossary Of Real Array Variables</td>
<td>26</td>
</tr>
<tr>
<td>V</td>
<td>Glossary Of Character Variables</td>
<td>27</td>
</tr>
</tbody>
</table>
PLOTSPEC
A FORTRAN 77 Program For Plotting Spectral Data From
The Varian CARY 2390 UV-VIS-NIR Spectrophotometer

INTRODUCTION

Spectral data acquisition from the CARY 2300 and 2400 series spectrophotometers has been implemented successfully using a Hewlett-Packard 1000 minicomputer system using the FORTRAN 77 program CARYSPEC, which is described in a separate report. The ability to acquire high quality UV-VIS-NIR spectra must be complemented with flexible data analysis and plotting software to make full use of the spectroscopic data. This report describes a fully tested FORTRAN 77 program PLOTSPEC which fulfills the plotting requirements of most users in a simple to use menu driven environment. The program supports Absorbance, Extinction Coefficient, % Transmission and % Reflectance ordinates vs linear Wavelength (nm) or Energy (cm⁻¹) abscissa. Automatic data rescaling is provided between Absorbance units and % Transmission so that either type of data file from CARYSPEC may be plotted in the ordinate of choice. PLOTSPEC supports two concurrent spectra, of up to 1000 data points each, allowing difference spectra to be obtained with automatic scaling for differences in concentration and pathlength. This mode is only valid for Absorbance of Extinction Coefficient ordinate choices and % Transmission will be automatically rescaled before plotting. Data treatment in the % Reflectance mode is identical to % Transmission with only the plotting label altered to match the use of this measurement technique. The abscissa axis scaling is bidirectional to suit individual preferences for the direction of increasing wavelength or energy units.

PLOTSPEC has been developed for use with an HP 1000 minicomputer system running the RTE-6/VM operating system and CI shell. In this environment the program communicates with Hewlett-Packard model 9872A and 7550A digital plotters via the IEEE-488 and RS-232C I/O subsystems, respectively. These communication functions are transparent to high level languages on the HP 1000 enabling simple READ/WRITE statements from FORTRAN to control the external plotting operations. The program resides in a single 32K word memory segment and utilizes 79 pages of Extended Memory Addressing (EMA) memory for the large data arrays. Since PLOTSPEC makes use of very few special features of the HP 1000 computer system the program could be modified easily to run on other host systems supporting the FORTRAN 77 language and either the IEEE-488 or RS-232C interface standards.

Manuscript approved August 9, 1988.
IMPLEMENTATION

1.0 Hardware Interface:

The IEEE-488 interface for the HP 1000 system is implemented with an HP 59310B interface card which utilizes 4 Logical Unit (LU) addresses in the system. The LU addresses used within the program are system dependent and are defined during the system generation. The interface card accessed by PLOTSPEC occupies LU addresses 31-34, corresponding to card addresses 0-3. Address 0 enables a special addressing mode for sending low level bus command sequences. Addresses 1-3 are predefined automatic READ/WRITE channels which select device numbers 1-3 on the bus. PLOTSPEC uses LU 33 to automatically address device #2 on the bus (HP 9872A plotter). To simplify alterations to the plotter address the program uses the INTEGER variable PU for the unit number in all READ/WRITE operations with the plotter.

The RS-232C interface to the HP 7550A plotter is implemented via a standard HP 1000 8 channel multiplexer card (MUX) operating at 9600 baud with 8 data bits and XON/XOFF software handshaking protocol. PLOTSPEC does not send device control commands to set these communications parameters within the HP 7550A since these are easy to select from the front panel control of the plotter. In this application the HP 7550A plotter is operated in the STANDALONE mode with DIRECT connection to the MUX output of the HP 1000.

1.1 System Handshaking:

The IEEE-488 subsystem of the HP 1000 is operated by the RTE driver DVA37 configured for ASCII Data Record mode, enabling automatic ASCII \textarrow{\rightleftharpoons} NUMERIC conversion on I/O. This mode sends and expects to receive an End Of Record (EOR) with data transmission in the form of a Carriage Return/Line Feed (CR/LF) sequence which matches the requirements of Hewlett-Packard digital plotters.

The RS-232C subsystem is operated by the RTE driver DDV00 which also performs automatic ASCII \textarrow{\rightleftharpoons} NUMERIC conversions for FORTRAN I/O statements. This allows PLOTSPEC to control either plotter without regard for differences in the handshaking protocols. Character labelling mode on these plotters requires a special terminator character and PLOTSPEC issues the default Etx character, ASCII 3.
1.2 HP-GL Software Control Of HP Plotters:

PLOTSPEC uses a subset of the Hewlett-Packard Graphics Language (HP-GL) to control the operations of the HP 9872A and HP 7550A digital plotters. The early generation of HP plotters often only allow scaling and plotting commands to use integer units which restricts the utility of the automatic scaling command 'SC' in these plotters. This lack of utility is so severe that the operating manual for the HP 9872A plotter does not even document the existence of the 'SC' command in the HP-GL syntax. However, PLOTSPEC utilizes a general scaling procedure with the 'SC' command which functions on all models of HP plotters, maintaining a 0-10000 unit scale on each axis at all times. The 'SC' command has the advantage of automatically mapping these user units onto the scaling points P1 (lower left) and P2 (upper right). In order to map the real user units onto this 10000 x 10000 unit grid the program only needs to use a multiplier (XMULT,YMULT) for the data on each axis. For example, a % Transmission range of 0-100% would require YMULT = 100 for correct scaling of the user's data. This technique maintains software resolution at 1 in 10000 units regardless of the range of the user's MIN,MAX values.

PLOTSPEC also supports bidirectional plotting on the X-axis even though the 'SC' scaling command of early HP plotters does not permit setting Xmin > Xmax, i.e. numbers increasing towards the left. In order to accomplish this task in a general manner the program uses two Xmin,Xmax ranges, 0 - +10000 (increasing to the right) and -10000 - 0 (increasing to the left). The change in sign has the effect of reversing the direction of plotting. This scheme is known to work on HP 9872A, 7225A, 7470A, 7475A and 7550A model plotters.

The HP-GL commands used within PLOTSPEC are summarized below together with their syntax and parameter types. The type labels INTEGER and DECIMAL should be interpreted as their ASCII representations. All data sent to or received from the plotters is in ASCII code. Therefore, where the command syntax below indicates a program variable, a parameter value enclosed within literal string delimiters is equally applicable. The HP 1000 computer system performs the conversion between numeric and ASCII representations automatically on I/O operations allowing liberal mixtures of literal string and numeric variables to appear in HP-GL instructions, with a few exceptions. For example, the character plot instruction 'CP' fails on receipt of the value 0.0 from a program variable but happily accepts the literal '0.0'. Where HP-GL commands are known fail to function correctly for valid ranges of parameters PLOTSPEC first converts the values to their string representation before sending them to the plotters.
1.3 HP-GL Commands Summary:

'AP'  Automatic Pen Pickup
Syntax = 'AP;'  

'CP'  Character Plot...moves the pen in units of character width, height  
Syntax = 'CP', X, ',', Y, ';
DECIMAL X, Y  

'DF'  Default Settings...sets the plotter to default parameters  
Syntax = 'DF;'  

'DI'  Direction...sets the labeling direction  
Syntax = 'DI', X, ',', Y, ';
where X, Y = cos(θ), sin(θ)  
DECIMAL X, Y  

'IP'  Input P1,P2 Scaling Points...sets the size of the plot  
Syntax = 'IP', Xl, ',', Y1, ',', X2, ',', Y2, ';
where (Xl,Y1) = lower left scaling coordinate (plotter units)  
where (X2,Y2) = upper right scaling coordinate (plotter units)  
INTEGER Xl, Y1, X2, Y2  

'IW'  Input Window...sets the size of the plotting window  
Syntax = 'IW', Xl, ',', Y1, ',', X2, ',', Y2, ';
where (Xl,Y1) = lower left soft clip coordinate (plotter units)  
where (X2,Y2) = upper right soft clip coordinate (plotter units)  
INTEGER Xl, Y1, X2, Y2  

'LB'  Label...draws the following literal characters  
Syntax = 'LB', string, Etx  
where Etx = CHAR(3) terminates LB  
ASCII string  

'LT'  Line Type...solid or broken lines styles  
Syntax = 'LT', PN, ',', PL, ';
where PN = Pattern number (1-6)  
INTEGER PN, DECIMAL PL  
where PL = Pattern length (0-100%)
'OA' Output Actual Pen Position...sends the current coordinate position
Syntax = 'OA;' followed by READ (...) X,Y,Z
where (X,Y) = current position (plotter units)
where Z = 0 or 1 for pen status (up/down)
INTEGER X,Y,Z

'OP' Output Scaling Points P1,P2...sends the coordinates of plot limits
Syntax = 'OP;' followed by READ (...) X1,Y1,X2,Y2
where (X1,Y1) = lower left scaling point
where (X2,Y2) = upper right scaling point
INTEGER X1,Y1,X2,Y2

'PA' Plot Absolute...moves the pen to an absolute coordinate
Syntax = 'PA',X,,Y,; where X,Y = coordinates (plotter units)
HP 9872A: INTEGER X,Y
HP 7550A: INTEGER X,Y or DECIMAL X,Y if user scaling is on

'PD' Pen Down...sets the pen on the plotting surface
Syntax = 'PD;'

'PU' Pen Up...picks up the pen from the plotting surface
Syntax = 'PU;'

'SC' Scale Plot...maps user units for X,Y axes onto P1,P2 scaling points
Syntax = 'SC',X1,,X2,,Y1,,Y2,; where (X1,Y1) = coordinate of scaling point P1 (user units)
where (X2,Y2) = coordinate of scaling point P2 (user units)
HP 9872A: INTEGER X1,Y1,X2,Y2
HP 7550A: DECIMAL X1,Y1,X2,Y2

'SR' Size Relative...sets the character size in proportion to axes scaling
Syntax = 'SR',WIDTH,,HEIGHT,; DECIMAL WIDTH,HEIGHT
Select Pen...picks up the pen from selected stall or carousel position
Syntax = 'SP',X,;'
where X = pen number (0-8)

Tick Length...sets the length of positive tick marks for axes
Syntax = 'TL',X,;'
where X = % of full scale (0-100)

Velocity Select...sets the pen down speed
Syntax = 'VS',X,;'
where X = velocity (cm/sec)

X-tick...draws a vertical tick mark at the current pen position
Syntax = 'XT;'

Y-tick...draws a horizontal tick mark at the current pen position
Syntax = 'YT;'

1.4 Default Plotting Parameters:

PLOTSPEC initializes the HP plotters using the 'DF' default parameters command and then sets up the following plotting style:

LABELS The initial label direction is set to horizontal with nominal character fields of 1 x 2 cm and the size relative option is selected.

PEN Pen #1 is selected with automatic pen pickup and set to the UP state. The pen down velocity is set to 5 cm/sec for best line quality with felt tip pens on graphics paper.

SCALE Initial scaling coordinates for the HP 9872A plotter are (1350,1000) and (8500,7000) corresponding to the NOTEBOOK size plot on A-size paper while the plotting window is set to the hard clip limits.

TICKS Tick marks are set to +1% of full scale.
2.0 Purpose Of PLOTSPEC:

This program is intended to provide a flexible plotting facility for spectral data acquired from the Cary 2390 spectrophotometer using the program CARYSPEC, which is described in a separate report. The HP 1000 computer system provides a large EMA area to hold data arrays and PLOTSPEC makes use of this feature to access two concurrent spectra of up to 10001 points each. This allows for difference spectra to be plotted with relative ease compared with the experimental difficulties incurred with real time subtraction. While the raw data from the spectrophotometer consists of Absorbance or % Transmission measurements vs Wavelength (nm), PLOTSPEC allows for presentation of the data in more meaningful units as well. The abscissa may be scaled in Wavelength (nm) or Wavenumbers x 10^{-3} with increasing values towards the left or right. The ordinate may be presented in Absorbance, Extinction Coefficient (Molar Absorptivity), % Transmission or % Reflectance. PLOTSPEC automatically rescales the original ordinate data to match the choice of plotting ordinate. Furthermore, automatic range, tick and label position selection is provided for fast and efficient setup of a neat and tidy plot. However, the user can easily alter the default selections in the menu driven plotting environment.

PLOTSPEC also provides some useful annotation facilities for documenting the plot and for producing presentation material. A digitizing routine is interwoven with the annotation commands to enable semi-automatic labelling of peak maxima with the ease of a point and shoot technique. More accurate determinations of peak maxima can also be made using a page scrolling data listing routine, though the digitizing facility seems to be perfectly adequate in most cases. The default plotting mode produces a small NOTEBOOK size plot on A-size paper, together with a list of spectral parameters for documentation purposes. A FULL size plotting mode can also be selected to produce a larger plot on A-size paper without a parameter list. Both modes include sufficient space for a centered title over the plot. Additionally, the program provides a USER SET plotting size with default boundaries suited for a full size plot on B-size paper. This mode is ideal for producing poster presentation material. PLOTSPEC supports output to two generic types of Hewlett-Packard digital plotters, the older HP 9872A and the more recent HP 7550A model. However, PLOTSPEC is designed to function with all generations of HP plotters with only minor differences in the position of the plotting origin.
2.1 Program Structure:

PLOTSPEC comprises a large main program unit containing the console menu displays, string data for the instrument settings variables and a number of subroutines that perform string processing, input validation and communication with the HP 9872A and HP 7550A digital plotters. The main program unit is responsible for all the logic flow and the subroutines execute specific support tasks, which are summarized below:

The main program unit of PLOTSPEC comprises 6 distinct segments of code to carry out the the functions of disk file data retrieval, instrument settings display, plotting mode setup and actual plotting operations. The code fragments appear under the following assigned labels: MENU, READ, SETTINGS, DIFF, PLOT and EXIT.

**MAIN PROGRAM**

**MENU:**

This is the first and main control menu of the program, selecting entry to the data retrieval, instrument settings, plotting mode and exit routines. The choices are as follows:

'R1' .....Read Spectrum #1

This command branches to label READ and the program performs logical tests for the presence of a valid spectrum in memory before allowing previous data to be overwritten by a new file. The data file format is listed below in Table I. Files are read into the first dimension of the data arrays and may be plotted using the 'P1' command.

'R2' .....Read Spectrum #2

This command performs the same operation as 'R1' but the file is read into the second dimension of the data arrays. The second spectrum can be plotted independently of the first or used for difference spectroscopy.

'IS' .....Instrument Settings

This option branches to label SETTINGS and prompts for the number of the spectrum. This routine allows for display of the most important operating conditions of the Cary 2390 spectrophotometer during acquisition of the selected spectrum. This provides an on-line reference when comparing a number of spectra.
'PS' .....Plotter Selection
This option allows the user to select either the HP 9872A (default) or HP 7550A plotter as the output device. Digitizing is more convenient with the older HP 9872A plotter but output quality and plotting speed are much improved with the HP 7550A.

'P1' .....Plot Spectrum #1
Selection of this command branches to label PLOT which performs a check for the presence of a valid file in memory. The routine then starts the plotting mode setup for the first spectrum with selection of the abscissa and ordinate scaling modes. The user has a choice of Nanometres and Wavenumbers for the abscissa and a choice of ABSORBANCE, Extinction Coefficient, % Transmission and % Reflectance for the ordinate. The program then proceeds to scale the data and set default plotter operating modes before presenting a menu of plotting options. The original data file may have been recorded in either Absorbance or % Transmission units so the data will be rescaled if the choice of plotting ordinate differs from the file ordinate.

'P2' .....Plot Spectrum #2
This command performs the same operations as 'P1' but uses data residing in the second dimension of the data arrays.

'DS' .....Difference Spectrum (1-2)
Selection of this option causes a branch to label DIFF which first checks for the presence of two valid spectra in memory and then tests for matching of their wavelength range and step size (nm). If the spectra are matched the program continues to label PLOT in the normal fashion. In this case, however, all plotting operations will use the scaled difference between the two data sets. The data from the second spectrum are scaled for differences in concentration or pathlength before subtraction during plotting.

'EX' .....Exit
The final option causes an unconditional branch to the label EXIT which terminates the program.
TABLE I

Data File Format

<table>
<thead>
<tr>
<th>Line</th>
<th>File Variables</th>
<th>Format Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TITLE(2)</td>
<td>CHARACTER (A72)</td>
</tr>
<tr>
<td>2</td>
<td>DATE(2)</td>
<td>CHARACTER (A8)</td>
</tr>
<tr>
<td>3</td>
<td>FMIN(2), FMAX(2), FSTEP(2), CONC(2), PATH(2)</td>
<td>REAL (*)</td>
</tr>
<tr>
<td>4</td>
<td>ORD(2), ABSC(2), CELL(2), CYCLE(2), SAMPLE(2), WAVE(2), TIMER(2), TEMP(2), DIST(2)</td>
<td>REAL (*)</td>
</tr>
<tr>
<td>5</td>
<td>J,K, NDATA(2,1)</td>
<td>INTEGER (I3,I3,I6)</td>
</tr>
<tr>
<td>6-54</td>
<td>PARAM(2,1)</td>
<td>INTEGER (I2)</td>
</tr>
<tr>
<td>55</td>
<td>VARIABLE(2,1)</td>
<td>REAL (*)</td>
</tr>
<tr>
<td>56-?</td>
<td>A(2,1)</td>
<td>REAL (*)</td>
</tr>
<tr>
<td>/-eof</td>
<td>W(2,1)</td>
<td>REAL (*)</td>
</tr>
</tbody>
</table>

*a*: Disk data files are read into either half of the data arrays

*b*: (*) indicates free field format
READ:
This section of PLOTSPEC opens a disk file previously stored by the data acquisition program CARYSPEC. The data are stored in ASCII code and contain a complete description of the instrument parameter settings as well as the spectral data. The format of the data file is listed above in Table I. The program prompts the user for both the filename and subdirectory. The latter defaults to the user's directory if only a RETURN character is entered. If a directory name is given the program builds a complete CI pathname for use in the OPEN statement. If the file cannot be found or the file is already open the program displays an error message and then returns to the main MENU. Otherwise, the file is read and the file descriptors are displayed on the console while the remainder of the data are being transferred. After completion of the data transfer the program prompts for changes to the concentration or pathlength variables. This feature allows for on-line corrections to the file variables before plotting, though permanent corrections should be made using the file editing program EDITSPEC. The program then returns to the main MENU.

SETTINGS:
This section of code displays a list of the most important instrument settings of the Cary 2390 spectrophotometer during acquisition of the specified spectrum. The program includes a large amount of string data for the various settings in the CHARACTER array Pstr. The INTEGER array PARAM is used as an index to this string data while numeric data for the table are obtained from the REAL array VARIABLES.

DIFF:
Entry into this section of code is made prior to plotting in order to verify that the data in memory are suitable for difference spectral plots. The program first checks that there are two spectra present and proceeds to test for exact matching of the wavelength limits and step size. The editing program EDITSPEC includes routines for preparing subset data files to support the difference plotting mode of PLOTSPEC.
PLOT:

This section of PLOTSPEC allows the user to set up the axes scaling to suit the type of data to be plotted. If the selected spectrum is not present in memory an error message is displayed and the program returns to the main MENU. Otherwise, a prompt is displayed for the choice of abscissa. PLOTSPEC supports two abscissa modes, Wavelength (nm) and Wavenumbers x 10^-3. After this selection the program sets a number of X-axis plotting variables and calls Subroutine Xaxis to setup the default tick and label positioning. The abscissa is bidirectional allowing the user to select the direction of increasing values. However, the default direction is for values to increase towards the right.

The following prompt is for the choice of ordinate mode. Data may be plotted in a choice of ABSORBANCE, Extinction Coefficient, % Transmission and % Reflectance. The latter pair are identical except for the label on the Y-axis. Data files from the CARYSPEC program may contain either Absorbance or % Transmission values so that rescaling is provided automatically to suit the user's choice of plotting ordinate. If the Difference Spectrum mode has been invoked only the Absorbance and Extinction Coefficient ordinates are valid. If the plotting ordinate uses Absorbance data the program will search the data array for the maximum value to provide automatic ranging on the Y-axis. If Extinction Coefficient mode is specified the program also calculates a suitable Y-axis exponent. Otherwise, the Y-axis defaults to 0-100% limits. The default tick and label positions are then set by calling Subroutine Yaxis.

The user is then prompted to check that the selected plotter is connected and turned on before proceeding to send plotter commands. Finally, the user is presented with the plotting menu with default settings for axis and label sizes, line type, pen velocity, the location of tick marks and axis labels and the command interface for plotting, annotating and digitizing.

EXIT:

The final portion of PLOTSPEC issue an erase command line instruction and then terminates the program unconditionally.
2.2 Plotting Menu Commands:

The interactive plotting control menu is designed both for flexibility and ease of use in setting quite a large number of plotting parameters. On most occasions, the default limits and spacing between tick marks and labels will be perfectly satisfactory. However, the user has complete control over these parameters to suit individual preferences. The various command selections are listed below:

'PL' ....Plot Limits
This command sets the actual range of the data file to be plotted. The default value is the full scan range. If the plotting scale is made smaller than the scan range the plot limits are reduced to match in order to prevent attempts to plot off scale data. If the plotting scale is made larger than the scan range the plot limits are set to the actual scan limits.

'XS' .....X-scale
The range of the X-axis plotting scale defaults to the scan range of the spectrum but may be made larger or smaller than the scan limits. The XMIN,XMAX values may be entered in reverse order to produce a plot with values increasing towards the left. These limits are passed to Subroutine Xaxis to reset the plot limits, tick marks and label positions automatically.

'XT' .....X-tick Spacing
This option allows for customizing the X-axis tick mark spacing with prompts for First, Last and Space. The limits are validated to lie within the current X-scale range and the label positions are automatically reset to match the tick positions.

'XL' .....X-label Spacing
This option allows the user to reset the automatic label positioning. The entries for First, Last and Space are treated in a similar manner to the tick marks.

'YS' .....Y-scale
The range of the Y-axis plotting scale defaults to the MIN,MAX scale appropriate for the selected ordinate. For %R and %T scales the default scale is 0-100%. For Absorbance and Extinction Coefficient ordinates autoranging is used to fit the entire spectrum on scale. The scale range is passed to Subroutine Yaxis to reset the plot limits, tick marks and label positions automatically.
'YT' .....Y-tick Spacing
This option allows for customizing the Y-axis tick mark spacing with prompts for First, Last and Space. The label positions are automatically reset to match the tick marks. No input validation is provided for the limits as it is assumed the user can read the updated console display before plotting.

'YL' .....Y-label Spacing
This option allows the user to reset the automatic label positioning. The entries for First, Last and Space are treated in a similar manner to the tick marks.

'CS' .....Char.Size
The size of character labels can be altered to suit individual preferences. However, the size should not be made much larger than the default fields (1x2 cm) when using the NOTEBOOK or FULL size axes definitions since the labels may not fit within the hard clip limits of the plotters.

'PV' .....Pen Velocity
Plotting quality is very dependent on the pen speed, the type of pen and plotting medium chosen. The program uses a default pen speed of 5 cm/sec for both the axes and spectrum plots. The user can alter these within the range 1-36 cm/sec. While the manual for the HP 7550A plotter recommends a speed of 50 cm/sec for the combination of felt tip pen and graphics paper there is no doubt that this is determined more by the quest for throughput than quality. Roller ball pens also work well at the 5 cm/sec pen speed. Thus, there is little need to alter the default values in most cases though transparency plots can benefit from a higher speed (10 cm/sec).

'LT' .....Line Type
The plotters have 6 inbuilt line type functions. The default value specifies a SOLID line. Since PLOTSPEC can plot multiple spectra on the same graph the BROKEN line type function is quite useful. The various patterns can be found by reference to the plotter manuals. The length of the patterns is entered as a % of full scale and a useful guide is to select the same length as the pattern number.
'AX' .....Axes Size
The program includes three, predefined plot types NOTEBOOK, FULL and USER SET. The plotting boundaries vary to suit different functions and paper sizes. The default NOTEBOOK size is a small, A-size plot which allows room for labelling a list of spectral file parameters on the right hand margin. The FULL size plot nicely fills the plotting area of A-size paper with sufficient space for a centered Title over the plot. The USER SET size allows for custom size plots up to the limits of B-size paper. This mode has predefined limits which produce a FULL size plot on B-size paper but the scaling coordinates can be easily altered by the user to suit individual needs.

'AN' .....Annotate Plot
This command allows the use to enter, move and plot Labels, a list of spectral Parameters or a centered Title. The Title defaults to the data file descriptor but the user can enter a new Title if required. The Parameters option is only valid for the NOTEBOOK size plot which allocates room for these descriptors in the right hand margin. The Labelling options allows for entry of descriptive text (up to 40 chars.) which may be positioned and drawn anywhere on the plotting surface. The character size and labelling direction are selectable while the plotting mode supports centered, left and right justification. The character size for labels defaults to 0.6*Normal plot size.

'PA' .....Plot Axes
This command draws the axes grid according to the selected tick and label positions listed in the plotting menu.

'PS' .....Plot Spectrum
Before proceeding with the plot this command pauses for the possible entry of a step multiplier. This allows the user to speed plotting for very large data files.

'DD' .....Digitize/Display Data
The program supports manual use of the plotters for digitizing peak positions from the plot. This routine will allow the user to annotate the plot at these positions with either the X coordinate, the Y coordinate or Both. Labels are drawn with small characters in the vertical direction and may be plotted in centered, left or right justified format. In this case, the annotation routine provides offsets to avoid writing over the spectrum.
2.3 COMMON DATA:

All COMMON variables used by PLOTSPEC are held in named COMMON blocks. The variables contained within the COMMON blocks are listed below:

**/DATA/**

Contains EMA REAL Arrays of spectroscopic data

A
Array containing the Absorbance or %T data
Values read from disk data file
Values used in main program unit

W
Array containing the wavelength data (nm)
Values read from disk data file
Values used in main program unit

**/DIGIT/**

Contains REAL and INTEGER variables for scaling digitized data

FACTOR
REAL variable for scaling absorbance to extinction coefficient units
Values set in main program unit (from data file variables)
Values used in main program unit and Subroutine Digitize

XMIN, XMAX
REAL variables containing the current X-axis limits
Values set in main program unit
Values used in main program unit, Subroutine Digitize, Subroutine Order and Subroutine Xaxis

YOFF, YSCALE
REAL variables containing the current Y-axis offset & scale length
Values set in main program unit
Values used in main program unit, Subroutine Digitize, Subroutine Ord'v and Subroutine Yaxis

IX, IY
INTEGER variables containing the coordinates of the digitized point

X1, Y1
INTEGER variables containing the coordinates of the scaling points P1, P2

X2, Y2
Values set in main program unit and Subroutine Digitize
Values used in main program unit and Subroutine Digitize
/XPARAM/ Contains REAL variables for plotting control

LMIN, Current X-axis plotting limits for validation of Tick or Label entries
LMAX
WN,WX Current plotting limits for Wavelength or Wavenumber scale
Xtick, X-axis tick mark spacing
FXT,LXT First and Last X-axis tick marks
Xlabel, X-axis label spacing
FXL,LXL First and Last X-axis labels
Values set in Subroutine Xaxis
Values used in main program unit

/YPARAM/ Contains REAL variables for plotting control

Ytick, Y-axis tick mark spacing
FYT,LYT First and Last Y-axis tick marks
Ylabel Y-axis label spacing
FYL,LYL First and Last Y-axis labels
Values set in Subroutine Yaxis
Values used in main program unit
2.4 SUBROUTINES:

PLOTSPEC uses subroutines to perform specific tasks which are required more than once, including string manipulations, input validation, digitizing and data rescaling. The purpose and calling sequences are listed below:

Center(TITLE)

Prints a string on the user console centred within a 72 column line.

TITLE CHARACTER*72 string, contents set by calling unit

CALLED BY: Main program unit
CALLS: None

Coeff(E,String,K)

Converts a numeric extinction coefficient into a rounded string with exponent, if required. Special handling is executed for 0.0000 and 1.0000 values.

E REAL variable input from calling unit

String CHARACTER*(*) output containing floating point string
DIMENSION is set by the calling unit

K INTEGER variable input from calling unit specifying the number of digits of precision required in string. Output value returns the number of characters in string to the calling unit.

CALLED BY: SUBROUTINE Digitize
CALLS: SUBROUTINE Str
Digitize(X,Xcode,Y,Ext,K)

Reads the current pen position from the plotter and converts the coordinates to the user's units. The routine also converts the Y-coordinate to extinction coefficient format. The main program determines whether the latter value is valid for the ordinate mode in use.

X REAL variable output containing the abscissa value in user units

Xcode CHARACTER variable input used to select scaling of the abscissa units

Y REAL variable output containing the ordinate value in user units

Ext CHARACTER*(*) output string containing the extinction coefficient
DIMENSION is set by the calling unit

K INTEGER variable input to specify the number of digits precision in Ext
Output value contains the number of characters in the returned string

CALLED BY: Main program unit
CALLS: SUBROUTINE Coeff

Exponent(N,Mult)

Converts the scale factor (power of 10) for extinction coefficient mode into string exponents for labelling the Y-axis in the screen menu and on the plotted graph (e.g. '1E-3' or '1 ...').

N REAL variable input, power of 10

Mult CHARACTER*(*) output string containing multiplier
DIMENSION is set by the calling unit

CALLED BY: Main program unit
CALLS: SUBROUTINE Str
Line(N)

Prints a line of '-' characters to the user console N columns wide and centred within a 72 column line.

N INTEGER variable input from calling unit

CALLED BY: Main program unit
CALLS: None

Order(MIN,MAX)

Validates the input arguments so that MIN < MAX

MIN,MAX REAL variables input and output with order swapped if required

CALLED BY: Main program unit
CALLS: None

Rescale(J,K,ND)

Converts ordinate data array between Absorbance units and % Transmission. The data array A(J,I) is accessed from EMA COMMON.

J INTEGER variable input (1 or 2) to specify the spectrum

K INTEGER variable input (0 or 1) to specify the scaling procedure. The original value for K is obtained from PAR:*M(J,1) in the data file. On return to the main program unit, the latter parameter is altered to reflect the current data format.

ND INTEGER variable input containing the number of data points in A(J,I)

CALLED BY: Main program unit
CALLS: None
Str(VALUE, String, PREC)

Performs a conversion from a numeric value to a string number for floating point numbers only with up to 12 digits precision.

VALUE REAL variable input to be processed by the routine

String CHARACTER*14 string output corresponding to VALUE

PREC INTEGER variable input to set the rounding precision for string

CALLED BY: Main program unit, SUBROUTINE Coeff, SUBROUTINE Exponent

CALLS: None

Upper(Code)

Performs a check for lower case characters in a string of arbitrary length and converts to upper case if necessary.

Code CHARACTER(*) variable passed into routine and UPPER case on exit

DIMENSION is set by the calling unit

CALLED BY: Main program unit

CALLS: None

EXTENSION: LEN(string) function, an HP extension to FORTRAN 77

Val(String, VALUE)

Performs a conversion from string to numeric value for a string number containing up to 10 digits. This is more than required by PLOTSPEC.

String CHARACTER*(*) string input to be processed by routine

DIMENSION is set by calling unit

VALUE REAL variable output

CALLED BY: Main program unit

CALLS: None

21
Wait(DELAY)

Performs a loop which tests the system clock until DELAY seconds have elapsed. The routine does not make provision for the special case at the transition to 2400 hours.

DELAY
REAL variable holding the value of the delay period in seconds

 CALLED BY:  Main program unit
 CALLS: FUNCTION Time(I)

Xaxis(XL,XH)

Performs both limits settings for the X-axis range variables and selection of the default Tick and Label positions. The routine starts with Xtick set to 1/10 of the full scale range and then compares this with a table of preferred values. If the range is anomalously small or large the Tick and Label positions are based on the 1/10 scale spacing. Otherwise, a neat and tidy spacing is selected from the table values with a label at every other tick position.

Yaxis(YSCALE)

Performs Tick and Label position selection using preferred table values, if possible. The routine start with Ytick equal to 1/10 of YSCALE. If the ordinate uses Extinction Coefficient scaling or the range is anomalously small or large the Tick and Label positions are based on the 1/10 scale spacing. Otherwise, a neat and tidy spacing is selected from the table values with a label at every other tick position.
2.5 FUNCTIONS:

PLOTSPEC uses only one function subprogram that makes an EXEC call to read the system time.

Time(I)

Performs an EXEC call to read the system clock and converts the reading to seconds and centiseconds.

I Dummy argument

CALLED BY: SUBROUTINE Wait only
CALLS: EXEC(ICODE,ITIME) system level command

PROGRAM CODE

3.0 Source Code Availability:

The source code for program PLOTSPEC is a 73K ASCII text file available on either a Hewlett-Packard cartridge, 9 track tape or an IBM 360K format floppy disk. All requests should be accompanied by the blank medium desired. A printed copy of the source code is listed below.

3.1 Variable Names And Usage:

A complete listing of the INTEGER, REAL, REAL Array and CHARACTER variables for the MAIN segment of PLOTSPEC is given below in Tables II, III, IV & V, respectively. The subroutines use the same names as the main program for the same variables. Additional variables in the subroutines and simple integers, I–N, are not documented since their usage is rather obvious. The logical variable MATCH is used within the program when comparing two spectra for the difference spectrum mode.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFF</td>
<td>Assigned Label - Difference Spectrum</td>
<td>370</td>
</tr>
<tr>
<td>EXIT</td>
<td>Assigned Label - Terminate Program</td>
<td>9000</td>
</tr>
<tr>
<td>MENU</td>
<td>Assigned Label - Main Control Menu</td>
<td>10</td>
</tr>
<tr>
<td>PLOT</td>
<td>Assigned Label - Plot Spectrum, 1 or 2</td>
<td>400</td>
</tr>
<tr>
<td>READ</td>
<td>Assigned Label - Read Spectrum, 1 or 2</td>
<td>570</td>
</tr>
<tr>
<td>SETTINGS</td>
<td>Assigned Label - Scan Conditions, 1 or 2</td>
<td>300</td>
</tr>
<tr>
<td>ASCII</td>
<td>ASCII equivalent of digits in Str</td>
<td>48-57</td>
</tr>
<tr>
<td>FINISH</td>
<td>Index of final abscissa value to plot</td>
<td>1-10001</td>
</tr>
<tr>
<td>IX,IY,IZ</td>
<td>Plotter position variables READ/WRITE</td>
<td>0-16000</td>
</tr>
<tr>
<td>NCOL</td>
<td>Number of screen columns in menu display</td>
<td>50-70</td>
</tr>
<tr>
<td>ND</td>
<td>Number of data points in plotted spectrum</td>
<td>1-10001</td>
</tr>
<tr>
<td>NP</td>
<td>Number of parameters to read from file</td>
<td>49</td>
</tr>
<tr>
<td>NV</td>
<td>Number of variables to read from file</td>
<td>14</td>
</tr>
<tr>
<td>PREC</td>
<td>Precision for rounding function in Str</td>
<td>3.4</td>
</tr>
<tr>
<td>PN</td>
<td>Plotter model selection number</td>
<td>1,2</td>
</tr>
<tr>
<td>PU</td>
<td>Plotter Logical Unit #, READ/WRITE</td>
<td>17,33</td>
</tr>
<tr>
<td>PXOFF</td>
<td>Plotter X-zero position offset</td>
<td>0,200</td>
</tr>
<tr>
<td>PYOFF</td>
<td>Plotter Y-zero position offset</td>
<td>0,100</td>
</tr>
<tr>
<td>START</td>
<td>Index of first abscissa value to plot</td>
<td>1-10001</td>
</tr>
<tr>
<td>UX1,UX2</td>
<td>Absolute Plotter scaling coordinates,</td>
<td>0-16000</td>
</tr>
<tr>
<td>UY1,UY2</td>
<td>USER size plot dimensions</td>
<td>0-10100</td>
</tr>
<tr>
<td>X1,X2</td>
<td>Plotter X &amp; Y coordinates for reading</td>
<td>0-16000</td>
</tr>
<tr>
<td>Y1,Y2</td>
<td>soft clip windowing points</td>
<td>0-10100</td>
</tr>
<tr>
<td>XP1,XP2</td>
<td>Absolute Plotter scaling coordinates,</td>
<td>0-16000</td>
</tr>
<tr>
<td>YP1,YP2</td>
<td>for WRITE to plot</td>
<td>0-10100</td>
</tr>
<tr>
<td>NDATA(2)</td>
<td>Number of data points in spectrum</td>
<td>1-10001</td>
</tr>
<tr>
<td>PARAM(2,49)</td>
<td>Instrument operating modes table</td>
<td>1-16</td>
</tr>
</tbody>
</table>
### Table III

**Glossary of REAL Variables**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BAND</td>
<td>Spectral Bandwidth (nm) - AUTO GAIN mode</td>
</tr>
<tr>
<td>BHEIGHT</td>
<td>Large label character height</td>
</tr>
<tr>
<td>BWIDTH</td>
<td>Large label character width</td>
</tr>
<tr>
<td>COFF</td>
<td>Character offset variable, labelling mode</td>
</tr>
<tr>
<td>ESCALE</td>
<td>Y-axis scale length, Ext. Coeff. mode</td>
</tr>
<tr>
<td>EMULT</td>
<td>Power of 10 for scaling, Ext. Coeff. mode</td>
</tr>
<tr>
<td>EOFF</td>
<td>Y-axis offset, Ext. Coeff. mode</td>
</tr>
<tr>
<td>FXL,LXL</td>
<td>First &amp; last X-axis labels</td>
</tr>
<tr>
<td>FXT,LXT</td>
<td>First &amp; last X-axis tick marks</td>
</tr>
<tr>
<td>FYL,LYL</td>
<td>First &amp; last Y-axis labels</td>
</tr>
<tr>
<td>FYT,LYT</td>
<td>First &amp; last Y-axis tick marks</td>
</tr>
<tr>
<td>GAIN</td>
<td>Instrument gain - AUTO SLIT mode</td>
</tr>
<tr>
<td>HEIGHT</td>
<td>Normal label character height</td>
</tr>
<tr>
<td>LHEIGHT</td>
<td>Small label character height</td>
</tr>
<tr>
<td>LWIDTH</td>
<td>Small label character width</td>
</tr>
<tr>
<td>LMIN</td>
<td>Current X-axis lower limit, data entry check</td>
</tr>
<tr>
<td>LMAX</td>
<td>Current X-axis upper limit, data entry check</td>
</tr>
<tr>
<td>NUMBER</td>
<td>General purpose data entry variable</td>
</tr>
<tr>
<td>PMIN</td>
<td>Pen scale minimum limit</td>
</tr>
<tr>
<td>PMAX</td>
<td>Pen scale maximum limit</td>
</tr>
<tr>
<td>RATIO</td>
<td>General purpose scaling variable</td>
</tr>
<tr>
<td>STEP</td>
<td>Step size (nm) interval, plotting mode</td>
</tr>
<tr>
<td>T1,...,T4</td>
<td>Powers of 10 (10,...,1E4)</td>
</tr>
<tr>
<td>WIDTH</td>
<td>Normal label character width</td>
</tr>
<tr>
<td>WN,WX</td>
<td>Current plot limits, default = WMIN,WMAX</td>
</tr>
<tr>
<td>WMIN,WMAX</td>
<td>Absolute scan limits (nm) or (kK)</td>
</tr>
<tr>
<td>X,Y</td>
<td>General purpose plotting variables</td>
</tr>
<tr>
<td>Xlabel</td>
<td>X-axis label spacing</td>
</tr>
<tr>
<td>XMIN,XMAX</td>
<td>Left &amp; Right edges of X-scale on plot</td>
</tr>
<tr>
<td>XMULT</td>
<td>Multiplier to scale user X-scale to 10000 units</td>
</tr>
<tr>
<td>XOFF</td>
<td>Offset to user X-scale units to set min'm at 0</td>
</tr>
<tr>
<td>Xtick</td>
<td>X-axis tick spacing</td>
</tr>
<tr>
<td>XN,XX</td>
<td>Data entry variables for X-scale, ticks &amp; labels</td>
</tr>
<tr>
<td>YN,YY</td>
<td>Data entry variables for Y-scale, ticks &amp; labels</td>
</tr>
<tr>
<td>Ylabel</td>
<td>Y-axis label spacing</td>
</tr>
<tr>
<td>YMULT</td>
<td>Scales Y-axis data &amp; labels for Ext. Coeff. mode</td>
</tr>
<tr>
<td>YOFF</td>
<td>Offset variable on current Y-axis scale</td>
</tr>
<tr>
<td>YSCALE</td>
<td>Current Y-axis scale length</td>
</tr>
<tr>
<td>Ytick</td>
<td>Y-axis tick spacing</td>
</tr>
</tbody>
</table>


Table IV

Glossary of REAL Array Variables

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORD(2)</td>
<td>Final ordinate value in file</td>
</tr>
<tr>
<td>ABSC(2)</td>
<td>Final abscissa value in file</td>
</tr>
<tr>
<td>CELL(2)</td>
<td>Final cell # value in file</td>
</tr>
<tr>
<td>CYCLE(2)</td>
<td>Final cycle # value in file</td>
</tr>
<tr>
<td>SAMPLE(2)</td>
<td>Final sample # value in file</td>
</tr>
<tr>
<td>WAVE(2)</td>
<td>Final wavelength value in file</td>
</tr>
<tr>
<td>TIMER(2)</td>
<td>Final time value in file</td>
</tr>
<tr>
<td>DIST(2)</td>
<td>Final distance value in file</td>
</tr>
<tr>
<td>A(2,10001)</td>
<td>Absorbance or %T array</td>
</tr>
<tr>
<td>CONC(2)</td>
<td>Concentration of sample (M), from data file</td>
</tr>
<tr>
<td>FACTOR(2)</td>
<td>Concentration * Pathlength for scaling spectra</td>
</tr>
<tr>
<td>FMAX(2)</td>
<td>Starting wavelength of scan (nm)</td>
</tr>
<tr>
<td>FMIN(2)</td>
<td>Ending wavelength of scan (nm)</td>
</tr>
<tr>
<td>FSTEP(2)</td>
<td>Step size (nm), from data file</td>
</tr>
<tr>
<td>PATH(2)</td>
<td>Pathlength of sample cell (cm), from data file</td>
</tr>
<tr>
<td>VARIABLE(2,14)</td>
<td>Instrument operating conditions table</td>
</tr>
<tr>
<td>W(2,10001)</td>
<td>Wavelength array</td>
</tr>
</tbody>
</table>
### Table V

Glossary Of CHARACTER Variables

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Screen Control:</strong></td>
<td></td>
</tr>
<tr>
<td>BELL</td>
<td>CHAR(7) bell character</td>
</tr>
<tr>
<td>CLR*2</td>
<td>Clear screen</td>
</tr>
<tr>
<td>DOWN*2</td>
<td>Move cursor down 1 line</td>
</tr>
<tr>
<td>ESC</td>
<td>CHAR(27) escape character</td>
</tr>
<tr>
<td>HOME*2</td>
<td>Move cursor to upper left corner</td>
</tr>
<tr>
<td>UP*2</td>
<td>Move cursor up 1 line</td>
</tr>
<tr>
<td><strong>File Status:</strong></td>
<td></td>
</tr>
<tr>
<td>DATE(2)*8</td>
<td>Date (mm/dd/yy)</td>
</tr>
<tr>
<td>Fname(2)*20</td>
<td>Filename, CI convention</td>
</tr>
<tr>
<td>LABEL(2)*72</td>
<td>Descriptor of spectrum</td>
</tr>
<tr>
<td>Pstr(23,16)*14</td>
<td>Table of parameter setting names</td>
</tr>
<tr>
<td>Smin(2)*4</td>
<td>Spectrum ending wavelength (nm)</td>
</tr>
<tr>
<td>Smax(2)*4</td>
<td>Spectrum starting wavelength (nm)</td>
</tr>
<tr>
<td>Sinc(2)*4</td>
<td>Step size interval (nm)</td>
</tr>
<tr>
<td>Sstat(2)*10</td>
<td>Spectrum status (VALID, INVALID)</td>
</tr>
<tr>
<td><strong>Program Control:</strong></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Literal comma ','</td>
</tr>
<tr>
<td>Code*2</td>
<td>Main menu selection, valid until reset</td>
</tr>
<tr>
<td>Directory*40</td>
<td>User directory name, CI convention</td>
</tr>
<tr>
<td>Ext*14</td>
<td>Extinction Coefficient, digitize &amp; label modes</td>
</tr>
<tr>
<td>Filespec*63</td>
<td>Full pathname, CI convention</td>
</tr>
<tr>
<td>Icode</td>
<td>General purpose selection key entry</td>
</tr>
<tr>
<td>Mult*5</td>
<td>Y-axis multiplier label, Ext. Coeff. mode</td>
</tr>
<tr>
<td>Pcode*2</td>
<td>Plotting menu option, valid until reset</td>
</tr>
<tr>
<td>Psize*8</td>
<td>Screen label for axes type selected</td>
</tr>
<tr>
<td>Size</td>
<td>Select type of plotted axes, 'N', 'F' or 'U'</td>
</tr>
<tr>
<td>String*14</td>
<td>String to pass data to or from subroutines</td>
</tr>
<tr>
<td>Text*40</td>
<td>General purpose string for 'abelling plot</td>
</tr>
<tr>
<td>TITLE*72</td>
<td>String to be printed to screen</td>
</tr>
<tr>
<td>Xcode</td>
<td>Selects X-axis scale, (nm) or (kK)</td>
</tr>
<tr>
<td>Xord*7</td>
<td>X-axis label (nm) or (cm⁻¹), digitize mode</td>
</tr>
<tr>
<td>Ycode</td>
<td>Selects Y-axis scale, 'A', 'E', 'R' or 'T'</td>
</tr>
</tbody>
</table>
Plotter Control:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR</td>
<td>CHAR(13), carriage return</td>
</tr>
<tr>
<td>Etx</td>
<td>CHAR(3), Terminator for label mode on plotter</td>
</tr>
<tr>
<td>LF</td>
<td>CHAR(10), line feed</td>
</tr>
<tr>
<td>Length*6</td>
<td>Broken line style pattern length (%)</td>
</tr>
<tr>
<td>Lmode*8</td>
<td>Line type mode (SOLID or BROKEN)</td>
</tr>
<tr>
<td>Pattern*6</td>
<td>Broken line type number</td>
</tr>
<tr>
<td>Plotter*10</td>
<td>Literal 'HP 9872A' or 'HP 7550A' identifier</td>
</tr>
<tr>
<td>Vaxes*2</td>
<td>Pen velocity (cm/sec), axes vectors</td>
</tr>
<tr>
<td>Vspec*2</td>
<td>Pen velocity (cm/sec), spectrum</td>
</tr>
<tr>
<td>Xleft*6</td>
<td>Literal '0' or '-10000', for scale command SC</td>
</tr>
<tr>
<td>Xright*6</td>
<td>Literal '10000' or '0', for scale command SC</td>
</tr>
<tr>
<td>Xstring*15</td>
<td>X-axis label, plotter output</td>
</tr>
<tr>
<td>Ylow*6</td>
<td>Literal '0'</td>
</tr>
<tr>
<td>Ystring*15</td>
<td>Y-axis label, screen and plotter output</td>
</tr>
<tr>
<td>Ytop*6</td>
<td>Literal '10000'</td>
</tr>
</tbody>
</table>
PROGRAM PLOTSPEC

This Program Is Designed To Plot Spectral Data Acquired From
The CARY 2390 UV-VIS-NIR Spectrophotometer With CARYSPEC.RUN

The Program Supports Output To HP 9872A And HP 7550A Plotters
The HP 9872A Is Plotter #1 At Address LU 33 (IEEE-488 Device #2)
The HP 7550A Is Plotter #2 At Address LU 17 (RS-232C Device)

AUTHOR: Dr. Robert A. Binstead,
Chemistry Division, Code 6125,
Naval Research Laboratory,
Washington, D.C. 20375-5000

WRITTEN: February, 1987

VERSION: 1.72

REVISED: March, 1987:
- Debugged plotting DO loops
- Added Ordinate/Abcissa mode checks
- Added ending wavelength check

April, 1987:
- Altered to use expt'l X-values
- Added screen display routine
- Removed rounding from SUB Xlimits
- Altered rounding to use NINT function

May, 1987:
- Altered annotation mode to support
  Title, Parameters & Labels on plot
- Altered AXES selection to provide
  default USER values & easy updates
- Changed FFRCL to 79 characters to
  prevent unwanted line wrapping

June, 1987:
- Altered Filename convention to match
  the directory structure of the new CI
  operating system
- The program had to be stripped down
  to run under the CI operating system
  by eliminating most I/O error checks

July, 1987:
- SEGMENTED versions of the full size
program failed with Memory Protect
errors at the INQUIRE statement if
the file existed already

August, 1987:
- Eliminated INQUIRE statement for file
exists or open check to avoid system
 crash in the segmented version of the
 program. The error checks are now made
 using error numbers returned in the
 OPEN statement.

April, 1988:
- Removed needless DATA for Pstr, unused
arrays and most I/O error checks on
WRITE to allow room for the complete
program in a single 32K word segment.
- Added Uppercase string conversion
routine for all string entries.
- Subroutines Xlimits & Ylimits were
replaced by Subroutine Order(MIN, MAX)
- Subroutine Xaxis was added to avoid
duplication in setting X-axis Limits,
Xtick and Xlabel spacing.
- Subroutine Yaxis was added to avoid
duplication in setting Y-axis Limits,
Ytick and Ylabel spacing.
- Altered Annotation Mode For Labels Up
To 40 Characters In Centered, Left &
Right Justified Formats. Positioning
& Character Size Commands Were Added.
- The Pen Position Digitizing Routine
Was Rewritten In Subroutine Form And
A Labelling Command Added To The Menu.
- The Annotation Routine Was Augmented
To Support Peak Position Labels In
Centered, Left & Right Justified Modes
With Special Offset Features To Avoid
Writing Over The Spectrum.
- % Transmission & % Reflectance Modes
Were Added To The Y-scale Selections.
- Re-scaling Between %T & Absorbance
Was Added To Support Both Plotting
Modes From Either Type Of Data File.
- Rounding Functions Were Altered To
Use ANINT (Whole Number) Command To
Avoid Integer Overflow With %T & %R
Labelling.
- Added Automatic Tick & Label Routines
To Select Preferred Values Based On
X & Y Scale Ranges. Defaults To :/10
Of Scale For Entry Of An Anomalously
Small Or Large Range.
- Added Code For Selection Of Plotter Type (HP 9872A / HP 7550A) And Made Adjustments To The P1,P2 Scaling Points To Fit Output Within The Hard Clip Limits Of Both Plotters.

Y-AXIS: Absorbance or Extinction Coefficient

X-AXIS: Wavelength or Wavenumbers x 1E-3

MEMORY: 30,000 Words (PROGRAM) + 80,000 Words EMA (DATA)

Dimension Screen Control String Variables

Dimension Program Parameter Variables

Define Rounding Functions Using Nearest Integer Commands

RO(X) = NINT(X) ! Type - INTEGER
<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>160</td>
<td>R1(X)=ANINT(X*10.0)/10.0</td>
<td>Type = REAL</td>
</tr>
<tr>
<td>161</td>
<td>R2(X)=ANINT(X*100.0)/100.0</td>
<td>Type = REAL</td>
</tr>
<tr>
<td>162</td>
<td>R3(X)=ANINT(X*1.0E3)/1.0E3</td>
<td>Type = REAL</td>
</tr>
<tr>
<td>163</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>164</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>165</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>166</td>
<td>Initialize String Variables</td>
<td></td>
</tr>
<tr>
<td>167</td>
<td>DATA (Plotter(I),I=1,2)'/HP 9872A ','HP 7550A '/</td>
<td></td>
</tr>
<tr>
<td>168</td>
<td>DATA (Pstr(1,I),I=1,6)'/ABSORBANCE','% TRANSMISSION',</td>
<td></td>
</tr>
<tr>
<td>169</td>
<td>&amp;'TEMPERATURE','% REFLECTANCE','CONCENTRATION','EMISSION'</td>
<td></td>
</tr>
<tr>
<td>170</td>
<td>DATA (Pstr(2,I),I=1,4)'/WAVELENGTH','TIME','TEMPERATURE',</td>
<td></td>
</tr>
<tr>
<td>171</td>
<td>&amp;'DISTANCE'</td>
<td></td>
</tr>
<tr>
<td>172</td>
<td>DATA (Pstr(3,I),I=1,11)'/OFF','0.01','0.02','0.05','0.1','0.2',</td>
<td></td>
</tr>
<tr>
<td>173</td>
<td>&amp;'0.5','1.0','2.0','5.0','10.0/'</td>
<td></td>
</tr>
<tr>
<td>174</td>
<td>DATA Pstr(4,1)'/OFF'</td>
<td></td>
</tr>
<tr>
<td>175</td>
<td>DATA (Pstr(4,I),I=6,15)'/0.2','0.5','1.0','2.0','5.0,'</td>
<td></td>
</tr>
<tr>
<td>176</td>
<td>&amp;'10','20','50','100','200/'</td>
<td></td>
</tr>
<tr>
<td>177</td>
<td>DATA (Pstr(5,I),I=1,4)'/AUTO SELECT','AUTO GAIN','AUTO SLIT',</td>
<td></td>
</tr>
<tr>
<td>178</td>
<td>&amp;'SINGLE BEAM'</td>
<td></td>
</tr>
<tr>
<td>179</td>
<td>DATA (Pstr(6,I),I=1,5)'/OFF','NORMAL','1ST DERIV','2ND DERIV',</td>
<td></td>
</tr>
<tr>
<td>180</td>
<td>&amp;'LOG'</td>
<td></td>
</tr>
<tr>
<td>181</td>
<td>DATA (Pstr(7,I),I=1,9)'/0.01','0.02','0.05','0.1','0.2','0.5',</td>
<td></td>
</tr>
<tr>
<td>182</td>
<td>&amp;'1.0','2.0','4.0/'</td>
<td></td>
</tr>
<tr>
<td>183</td>
<td>DATA (Pstr(8,I),I=10,16)'/2','5','10','20','50','100','200/'</td>
<td></td>
</tr>
<tr>
<td>184</td>
<td>DATA (Pstr(9,I),I=12,15)'/10','20','50','100/'</td>
<td></td>
</tr>
<tr>
<td>185</td>
<td>DATA (Pstr(10,I),I=1,5)'/-1.9 TO 0.6','-2.0 TO 0.5',</td>
<td></td>
</tr>
<tr>
<td>186</td>
<td>&amp;'-2.1 TO 0.4','-2.2 TO 0.3','-2.3 TO 0.2'/</td>
<td></td>
</tr>
<tr>
<td>187</td>
<td>DATA (Pstr(11,I),I=1,16)'/+/-0.01', '+/-0.02', '+/-0.05', '+/-0.1',</td>
<td></td>
</tr>
<tr>
<td>188</td>
<td>&amp;'+/-0.2','+/-0.5','+/-1.0','+/-2.0','+/-5.0','+/-10.0','+/-20.0',</td>
<td></td>
</tr>
<tr>
<td>189</td>
<td>&amp;'+/-50','+/-100','+/-200','+/-500','+/-1000'</td>
<td></td>
</tr>
<tr>
<td>190</td>
<td>DATA (Pstr(12,I),I=1,16)'/+/-0.01', '+/-0.02', '+/-0.05', '+/-0.1',</td>
<td></td>
</tr>
<tr>
<td>191</td>
<td>&amp;'+/-0.2','+/-0.5','+/-1.0','+/-2.0','+/-5.0','+/-10.0','+/-20.0',</td>
<td></td>
</tr>
<tr>
<td>192</td>
<td>&amp;'+/-50','+/-100','+/-200','+/-500','+/-1000'</td>
<td></td>
</tr>
<tr>
<td>193</td>
<td>DATA (Pstr(14,I),I=1,10)'/0','10','20','30','40','50','0.0','70',</td>
<td></td>
</tr>
<tr>
<td>194</td>
<td>&amp;'80','90'</td>
<td></td>
</tr>
<tr>
<td>195</td>
<td>DATA (Pstr(15,I),I=1,4)'/0.5','1.0','3.0','10'/</td>
<td></td>
</tr>
<tr>
<td>196</td>
<td>DATA (Pstr(16,I),I=1,2)'/NORMAL','REVERSE'</td>
<td></td>
</tr>
<tr>
<td>197</td>
<td>DATA (Pstr(17,I),I=1,2)'/OFF','ON'</td>
<td></td>
</tr>
<tr>
<td>198</td>
<td>DATA (Pstr(18,I),I=1,2)'/REPEAT SCAN','SGL/MULTI'</td>
<td></td>
</tr>
<tr>
<td>199</td>
<td>DATA (Pstr(19,I),I=1,2)'/SERIAL','OVERLAY'</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>DATA (Pstr(20,I),I=1,4)'/BOTH ON','UV ONLY','VIS/NIR ONLY',</td>
<td></td>
</tr>
<tr>
<td>201</td>
<td>&amp;'BOTH OFF'</td>
<td></td>
</tr>
<tr>
<td>202</td>
<td>DATA (Pstr(21,I),I=1,3)'/AUTO','UV','VIS/NIR'</td>
<td></td>
</tr>
<tr>
<td>203</td>
<td>DATA (Pstr(22,I),I=1,3)'/AUTO','UV/VIS','NIR'</td>
<td></td>
</tr>
<tr>
<td>204</td>
<td>DATA (Pstr(23,I),I=1,2)'/FULL','1/3'</td>
<td></td>
</tr>
<tr>
<td>205</td>
<td></td>
<td></td>
</tr>
<tr>
<td>206</td>
<td></td>
<td></td>
</tr>
<tr>
<td>207</td>
<td>BELL=CHAR(7)</td>
<td></td>
</tr>
<tr>
<td>208</td>
<td>ESC=CHAR(27)</td>
<td></td>
</tr>
<tr>
<td>209</td>
<td>CLR=ESC/&quot;J&quot;</td>
<td></td>
</tr>
<tr>
<td>210</td>
<td>HOME=ESC/&quot;h&quot;</td>
<td></td>
</tr>
<tr>
<td>211</td>
<td>UP=ESC/&quot;A&quot;</td>
<td></td>
</tr>
<tr>
<td>212</td>
<td>DOWN=ESC/&quot;B&quot;</td>
<td></td>
</tr>
<tr>
<td>213</td>
<td>ERASE=ESC/&quot;K&quot;</td>
<td></td>
</tr>
</tbody>
</table>

32
214 C
215 PN-1
216 ! Select Plotter = 1 (HP 9872A)
217 T1 = 10.0
218 T2 = 100.0
219 T3 = 1000.0
220 T4 = 10000.0
221 C
222 C
223 C
224 C
225 C
226 C
227 ASSIGN 10 TO MENU
228 ASSIGN 100 TO READ
229 ASSIGN 300 TO SETTINGS
230 ASSIGN 370 TO PLOT
231 ASSIGN 390 TO DIFF
232 ASSIGN 9000 TO EXIT
233 C
234 C
235 C
236 C
237 C
238 C
239 C
240 CALL FFRCL(79)
241 10 WRITE (1,*) HOME, CLR, '
242 NCOL = 70
243 TITLE = 'Cary 2390 Spectral Data Plotting'
244 CALL Center(TITLE)
245 WRITE (1, '(T61,A2,A9)') UP, 'Rev: 1.72'
246 CALL Line(NCOL)
247 WRITE (1, 20) 'CODE', 'FUNCTION', 'STATUS', 'MIN', 'MAX', 'INC'
248 % FORMAT (T4, A4, T18, A9, T38, A7, T50, A3, T58, A3, T66, A3)
249 CALL Line(NCOL)
250 WRITE (1, 30) 'R1', '.....Read Spectrum = 1............', Sstat(1)
251 &Smin(1), Smax(1), Sen(1)
252 WRITE (1, 30) 'R2', '.....Read Spectrum = 2............', Sstat(2)
253 &Smin(2), Smax(2), Sen(2)
254 WRITE (1, 50) 'IS', '.....Instrument Settings.....'
255 WRITE (1, 30) 'PS', '.....Plotter Selection.....', Plott(PN)
256 WRITE (1, 40) 'P1', '.....Plot Spectrum = 1.....', Fname(1)
257 WRITE (1, 40) 'P2', '.....Plot Spectrum = 2.....', Fname(2)
258 WRITE (1, 50) 'DS', '.....Difference Spectrum.....'
259 WRITE (1, 50) 'EX', '.....EXIT Menu.............'
260 30 FORMAT (T5, A7, A30, A10, T50, A4, T58, A4, T66, A4)
261 40 FORMAT (T5, A2, T7, A30, A16)
262 50 FORMAT (T5, A2, T7, A30)
263 WRITE (1,*)
264 CALL Line(NCOL)
265 WRITE (1,*)
266 70 WRITE (1,*) UP, ERASE, '_'
WRITE (1,'(T3,A15,A,A2)') 'Enter the CODE: ', BELL, '_'
READ (1,80) Code
80 FORMAT (A2)
CALL Upper(Code)

C

IF (((Code.EQ.'R1').OR.(Code.EQ.'R2'))) THEN
  J=ICHAR(Code(2:2))-48
  GO TO READ
END IF
IF (Code.EQ.'IS') GO TO SETTINGS
IF (Code.EQ.'PS') THEN
  WRITE (1,*) UP, ERASE,' Select: 1...HP 9872A,. '
  & ' 2...HP 7550A ? ', BELL, '_'
  READ (1,99) Icode
  IF ((Icode.NE.'1').AND.(Icode.NE.'2')) THEN
    GO TO 90
  END IF
  PN=ICHAR(Icode)-48
  GO TO MENU
END IF
IF (((Code.EQ.'P1').OR.(Code.EQ.'P2'))) THEN
  J=ICHAR(Code(2:2))-48
  GO TO PLOT
END IF
IF (Code.EQ.'DS') THEN
  J=1
  GO TO DIFF
END IF
IF (Code.EQ.'EX') GO TO EXIT
GO TO 70
C
C
300 C
C
301 C
C
302 C
C
99 FORMAT (A1)
100 IF (Sstat(J).EQ.'VALID') THEN
  WRITE (1,*) UP, ERASE,' Spectrum is PRESENT: ',
  & 'Proceed (Y or N) ? ', BELL, '_'
  READ (1,99) Icode
  CALL Upper(Icode)
  IF (Icode.EQ.'N') GO TO 70
  IF (Icode.EQ.'Y') GO TO 100
END IF
WRITE (1,*) HOME, CLR, '_'
TITLE='Read Spectrum = '//Code(2:2)
CALL Center(TITLE)
CALL Line(NCOL)
WRITE (1,*) DOWN, ' Enter Filename: ', BELL, '_'
READ (1,'(A20)') Fname(J)
CALL Upper(Fname(J))
WRITE (1,*) DOWN,' Directory, (Return = /DEFAULT/) '.BELL.'_
READ (1,'(A40)') Directory
IF (Directory.EQ.' ') THEN
  Filespec=Filename(J)
  GO TO 110
END IF
K=40
DO WHILE (Directory(K:K).EQ.' ')
  K=K-1
END DO
IF (Directory(K:K).EQ.'/') K=K-1
Filespec=Directory(1:K)//'///Filename(J)
110 K=63
DO WHILE (Filespec(K:K).EQ.' ')
  K=K-1
END DO
WRITE (1,*) UP,ERASE,' Validating: ',Filespec(1:K),'_
CALL Wait(1.0)
OPEN (UNIT=66,FILE=Filespec(1:K),IOSTAT=N,STATUS='OLD')
WRITE (1,*)
IF (N.NE.0) THEN
  WRITE (1,*) UP,ERASE,'_'
  N=N-500
  IF (N.EQ.6) WRITE (1,*) ' File does NOT EXIST: ',BELL.'_
  IF (N.EQ.8) WRITE (1,*) ' File is already OPEN: ',BELL.'_
  IF ((N.NE.6).AND.(N.NE.8)) WRITE (1,*) ' Disk Error = '.N, & BELL,'_
  CLOSE (UNIT=66,STATUS='DELETE')
  CALL Wait(2.0)
  WRITE (1,*)
  Sstat(J)=''INVALID''
  GO TO MENU
END IF
WRITE (1,*) UP,ERASE,' Reading File: ',Filespec(1:K)
"54 READ (66,FMT=120,IOSTAT=N,ERR=210) LABEL(J)
120 FORMAT (A72)
WRITE (1,*) DOWN,' Title:'
WRITE (1,*) ',',LABEL(J)
130 FORMAT (A8)
WRITE (1,*) DOWN,' Date: ',DATE(J)
140 FORMAT (A72)
WRITE (1,*) DOWN,' FMIN(J),FMAX(J),FSTEP(J), & CONC(J),PATH(J)
READ (66,*) ORD(J),ABSC(J),CELL(J),CYCLE(J), & SAMPLE(J),WAVE(J),TIMER(J),TEMIP(J),DIST(J)
WRITE (1,140) DOWN,'Scan Range: ',FMAX(J),' to ',FMIN(J), &'
CONC(J)','run at ',FSTEP(J),' nm s.e.p.s'
WRITE (1,'(T4,A2,A12,F6.2,F6.2,A7,F4.2,A9)') DOWN,'Temp. (C) : ',TEMP(J), &
150 FORMAT (T4,A2,A12,F6.2)
CALL Str(FMIN(J),String,4)
Smin(J)=String(2:5)
CALL Str(FMAX(J),String,4)
Smax(J)=String(2:5)
CALL Str(FSTEP(J),String,4)
Sinc(J)=String(2:5)
READ (66,FMT=160,IOSTAT=N,ERR=210) NP,NV,NDATA(J)
160 FORMAT (13,13,16)
IF ((NP.NE.49).OR.(NV.NE.14)) GO TO 200
WRITE (1,*) DOWN,' Warning! Scan ended at ',WAVE(J),', nm'.BEL
CALL Wait(2.0)
END IF
WRITE (1,*)
TITLE='#* READING SPECTRUM ~
WRITE (1,*)
HOME,'_'
CALL Center(TITLE)
READ (66,FMT=170) (PARAM(J,K),I(-1,NP)
170 FORMAT (12)
READ (66,*)(VARIABLE(J,K),K=1,NV)
READ (66,*)(W(J,K),K=1,NDATA(J))
Sstat(J)='VALID'
180 CLOSE (UNIT=66,IOSTAT=N,ERR=210)
IF (Sstat(J).NE.'VALID') GO TO MENU
WRITE (1,*) UP,ERASE,' Data Format Error: ',BELL,'_'
Sstat(J)='ABORTED'
GO TO 220
WRITE (1,*) UP,ERASE,' Disk Error =',N,BELL,'_'
Sstat(J)='INVALID'
210 CALL Wait(2.0)
WRITE (1,*) UP,ERASE,' Concentration, KM): ',BELL,'_'
READ (1,99) CONC(J)
GO TO MENU
230 CALL Wait(2.0)
WRITE (1,*) UP,ERASE,' (C...Conc. , P...Path) ? ',BELL,'_'
READ (1,99) Icode
CALL Upper(Icode)
IF (Icode.EQ..'P') GO TO 250
IF (Icode.NE..'C') GO TO 230
WRITE ('*,UP,ERASE,' Concentration, (M): ',BELL,'_'
READ (1,*ERR=240) CONC(J)
240 GO TO 190
WRITE (1,*ERR=250) PATH(J)
250 GO TO 190
Error Messages & Returns To Menu

260 TITLE=' Conc. or Path = 0.0 '
270 TITLE=' Spectra are MISMATCHED: '
280 TITLE=' Spectrum is ABSENT: '
290 WRITE (1,*') UP,ERASE,TITLE(1:26),BELL,'_'
   CALL Wait(2.0)
   WRITE (1,*')
   GO TO 70

Display Instrument Settings

300 IF ((Sstat(l).NE.'VALID').AND.(Sstat(2).NE.'VALID')) GO TO 280
310 WRITE (1,*') UP,ERASE,' Spectrum #*, (1 or 2): ',BELL,'_'
   READ (1,'(12)',ERR-310) J
   IF ((J.NE.1).AND.(J.NE.2)) GO TO 310
   IF (Sstat(J).NE.'VALID') GO TO 280
   BAND-VARIABLE(J,10)
   GAIN-VARIABLE(J,6)
   Strtng-Pstr(7,PARAM(J,7)+1)
   K=1
   IF (PARAM(J,1).NE.0) THEN
      String-Pstr(8,PARAM(J,8)+1)
      K=11
   END IF
   CALL Val(String,PMAX)
   PMIN=VARIABLE(J,K)
   PMAX=PMIN+PMAX
   K=11
   IF (PARAM(J,6).EQ.4) K=10
   String=Pstr(K,PARAM(J,K)+1)
   WRITE (1,*') HOME,CLR,'_'
   TITLE='Instrument Settings'
   CALL Center(TITLE)
   CALL Line(50)
   WRITE (1,'(T20,A8,T40,A")') 'FUNCTION','SETTING'
   CALL LINE(50)
   WRITE (1,*')
   WRITE (1,320) 'WAVELENGTH LIMITS.......',FMIN(J),',',FMAX(J)
   WRITE (1,330) 'ORDINATE..................',Pstr(1,PARAM(J,1)+1)
   WRITE (1,330) 'ABSCISSA...................',Pstr(2,PARAM(J,2)+1)
   WRITE (1,330) 'SCAN RATE (nm/sec).......',Pstr(3,PARAM(J,3)+1)
WRITE (1,330) 'CHART DISPLAY (nm/cm)..' ,Pstr(4,PARAM(J,4)+1)
WRITE (1,330) 'REFERENCE MODE........',Pstr(5,PARAM(J,5)+1)
WRITE (1,340) 'SBW (nm), GAIN........',BAND,'.',C.IN
WRITE (1,330) 'PEN FUNCTION..............',Pstr(6,PARAM(J,6)+1)
WRITE (1,340) 'PEN LIMITS (Min,Max).....',PMIN,',',PMAX
IF (PARAM(J,6).GT.1) WRITE (1,350) UP,ERASE,String
WRITE (1,330) 'RESPONSE TIME (sec).....',Pstr(15,PARAM(J,15)+1)
WRITE (1,330) 'BEAM INTERCHANGE........',Pstr(16,PARAM(J,16)+1)
WRITE (1,330) 'SLIT HEIGHT.............',Pstr(23,PARAM(J,23)+1)
WRITE (1,330) 'LAMP SELECT.............',Pstr(21,PARAM(J,21)+1)
WRITE (1,330) 'DETECTOR SELECT........',Pstr(22,PARAM(J,22)+1)

320 FORMAT (T15,A23,T40,F5.2,A,F5.2)
330 FORMAT (T15,A23,T40,A14)
340 FORMAT (T15,A23,T40,F4.2,A,F5.2)
350 FORMAT (T15,A23,T40,F4.2,A,F5.2)
360 WRITE (1,'(T14, ,A2, ,A2, ,A23, ,A, ,A)') UP,ERASE,
&'Press RETURN for MENU: ',BELL,'_'
READ (1,99) Icode
IF (Icode.NE.' ') GO TO 360
GO TO MENU

----------------------------------------------

Plot Mode Setup

----------------------------------------------

370 IF (Sstat(J).NE.'VALID') GO TO 280
GO TO 400
390 IF (((Sstat(1).NE.'VALID').OR.(Sstat(2).NE.'VALID'))) GO TO 280
MATCH=.TRUE.
IF (FSTEP(1).NE.FSTEP(2)) MATCH=.FALSE.
IF (FMIN(1).NE.FMIN(2)) MATCH=.FALSE.
IF (FMAX(1).NE.FMAX(2)) MATCH=.FALSE.
IF (MATCH) GO TO 400
GO TO 270

----------------------------------------------

400 TITLE='Plot Spectrum #'//CHAR(J+48)
FACTOR(1)=CONC(1)*PATH(1)
FACTOR(2)=CONC(2)*PATH(2)
IF (Code.EQ.'DS') THEN
IF ((FACTOR(1).EQ.0.0).OR.(FACTOR(2).EQ.0.0)) GO TO 260
RATIO=FACTOR(1)/FACTOR(2) Temporary Scaling Factor
'TITLE='Difference Spectrum Plot'
END IF
IF (FACTOR(J).EQ.0.0) FACTOR(J)=1.0 ! Avoid Division By Zero
WRITE (1,*) HOME,CLR,'_'
CALL Center(TITLE)
CALL Line(NCOL)
WRITE (1,*) DOWN,DOWN
WRITE (1,'(T10,A32)') 'X-scale units: N.....Nanometres'
WRITE (1,*)
WRITE (1,'(T26,A17)') 'W.....Wavenumbers'
WRITE (1,*)
DOWN
WRITE (1,420) UP,ERASE,'Code: ',BELL,'_'
FORMAT (T19,A2,A2,A7,A,A)
READ (1,99) Xcode
CALL Upper(Xcode)
IF (Xcode.EQ.'W') THEN
  WRITE (1,*)
  WRITE (1,'(T26,A2,A2,All)') UP,ERASE,'Wavenumbers'
  Xord=' (cm-1)'
  WMAX=R2(T4/FMIN(J))
  WMIN=R2(T4/FMAX(J))
  GO TO 430
END IF
IF (Xcode.NE.'N') GO TO 410
WRITE (1,*)
WRITE (1,'(T26,A2,A2,A10)') UP,ERASE,'Nanometres'
Xord-' (run)'
WMAX=RI(FMAX(J))
WMIN=RI(FMIN(J))
END IF
430 ND=NDATA(J) ! Number Of Data Points
STEP=FSSTEP(J) ! Step Size (nm)
XMIN=WMIN ! X-scale MIN/MAX = Scan Limits
XMAX=WMAX
CALL Xaxis(XMIN,XMAX) ! Set Limits, Tick & Label Spacing
WRITE (1,*)
WRITE (1,'(T10,A32)') 'Y-scale units: A.....Absorbance'
WRITE (1,*)
WRITE (1,'(T26,A28)') 'E.....Extinction Coefficient'
WRITE (1,*)
WRITE (1,'(T26,A19)') 'R.....% Reflectance'
WRITE (1,*)
WRITE (1,'(T26,A20)') 'T.....% Transmission'
WRITE (1,*)
DOWN
WRITE (1,420) UP,ERASE,'Code: ',BELL,'_'
READ (1,99) Ycode
CALL Upper(Ycode)
IF (Code.EQ.'DS') THEN
  IF ((Ycode.EQ.'R').OR.(Ycode.EQ.'T')) GO TO 440
END IF
Ystring='ABSORBANCE' ! Select Y-axis Label For Plot
K=0 ! Ordinate Mode = Absorbance
ELSE IF (Ycode.EQ.'E') THEN
  K=0 ! Ordinate Mode = Absorbance
ELSE IF (Ycode.EQ.'R') THEN
  Ystring='% Reflectance'
  K=1 ! Ordinate Mode = % Transmission
ELSE IF (Ycode.EQ.'A') THEN
  Ystring='ABSORBANCE' ! Select Y-axis Label For Plot
  K=0 ! Ordinate Mode = Absorbance
ELSE IF (Ycode.EQ.'E') THEN
  K=0 ! Ordinate Mode = Absorbance
ELSE IF (Ycode.EQ.'R') THEN
  Ystring='% Reflectance'
  K=1 ! Ordinate Mode = % Transmission
ELSE IF (Ycode.EQ.'T') THEN
  Ystring='% Transmission'
  K=1
END IF
ELSE
  GO TO 440
END IF

IF (PARAM(J,1).NE.K) THEN
  CALL Rescale(J,K,ND) ! Rescale Between Absorbance & %T
  PARAM(J,1)=K ! Reset Ordinate Variable To Match
END IF

IF (Code.EQ.'DS') THEN
  K=0 ! Only Absorbance Mode Allowed
  IF (PARAM(2,1).NE.K) THEN
    CALL Rescale(2,K,ND) ! Rescale To Absorbance If Required
    PARAM(2,1)=K ! Reset Ordinate Variable To Match
  END IF
END IF

IF (Ycode.NE.'E') THEN
  WRITE (1,*)
  Text(1:24),UP ! Erase Unwanted Labels
  YMULT=1.0
  IF (Ycode.EQ.'A') GO TO 450
  YOFF=0.0
  YSCALE=1.00
  CALL Yaxis(YSCALE,Ycode)
END IF

IF (FACTOR(J).EQ.0.0) THEN
  WRITE (1,'(T19,A2,A2,T30,A19,A,A)') UP,ERASE,'Conc. or Path - 0.0',BELL,'_'
  CALL Wait(2.0)
  WRITE (1,*)
  GO TO 440
END IF

IF (FACTOR(J).EQ.0.0) THEN
  WRITE (1,'(T19,A2,A2,T30,A19,A,A)') UP,ERASE,'Conc. or Path - 0.0',BELL,'_'
  CALL Wait(2.0)
  WRITE (1,*)
  GO TO 440
END IF

DO WHILE (K.LE.ND) ! Find Maximum Absorbance
  NUMBER=A(J,K)
  IF (Code.EQ.'DS') THEN
    NUMBER=ABS(A(1,K)-A(2,K)*RATIO)
  END IF
  IF (PMAX.LT.NUMBER) PMAX=NUMBER
  K=K+1
END DO

SCALE=R3(PMAX+0.001) ! Round Up Absorbance Max'

IF (PMAX.GT.0.01) SCALE=R2(PMAX+0.01)
IF (PMAX.GT.0.10) SCALE=R1(PMAX+0.10)
YOFF=0.0 !Initial Y-scale Min'm
YSCALE=SCALE !Initial Y-scale Max'm
ESCALE=SCALE !Initial E-scale Max'm

IF (Code.EQ.'DS') THEN
    YSCALE=2.0*SCALE !2 x Y-scale for +/- Max'm Range
    YOFF=SCALE !Set Zero Position At Half Y-scale
END IF
CALL Yaxis(YSCALE,Ycode) !Set Tick & Label Spacing
IF (Ycode.EQ.'A') GO TO 460

Scale YMULT for Extinction Coefficient Mode

DO WHILE (ESCALE/FACTOR(J).LT.1.0)
    ESCALE=ESCALE*10.0
END DO
DO WHILE (ESCALE/FACTOR(J).GT.10.0)
    ESCALE=ESCALE/10.0
END DO

RATIO=ESCALE/SCALE
YMULT=RATIO/FACTOR(J) Scaling Factor For Ext. Coeff.

CALL Exponent(RATIO,Mult) !Convert Exponent To String

Plotting Routine: HP 9872A & HP 7550A

PU-33 !Plotter Logical Unit # (HP 9872A)
PXOFF=0 !X-zero Position Offset (HP 9872A)
PYOFF=0 !Y-zero Position Offset (HP 9872A)

IF (PN.EQ.2) THEN
    PU-17 !Plotter Logical Unit # (HP 7550A)
    PXOFF=200 !X-zero Position Offset (HP 7550A)
    PYOFF=100 !Y-zero Position Offset (HP 7550A)
END IF

WIDTH=1.0 !Character Field Width (cm)
HEIGHT=2.0 !Character Field Height (cm)
Lmode='SOLID' !Solid Line Type ON
Pattern=' ' !Broken Line Type OFF
Length=' ' !Not Used For Solid Pattern
LF=CHAR(10) !Linefeed Character
CR=CHAR(13) !Carriage Return Character
C='.' !Data Separator In HP-GL Commands
Et x=CHAR(3) !Label Mode Terminator (HP Default)
Size='N' !Plot Size Default Is NOTEBOOK
Vaxes='5' !Pen Speed (cm/sec) For Axes Vectors
Vspec='5' !Pen Speed (cm/sec) For Spectrum
WRITE (1,*) DOWN,DOWN,DOWN
470 WRITE (1,480) UP,ERASE,'Press RETURN if plotter is ON;',BELL,'_'
480 FORMAT (T10,A2,A2,A31,A,A)
READ (1,99) icode
IF (icode.NE.' ') GO TO 470
WRITE (1,'(T10,A2,A2,A21)') UP,ERASE,'Initializing Plotter:'
WRITE (PU,*,IOSTAT-N,ERR-9999) 'DF'
UX1=1800
UX2=15500 ! Default User Set Plotting Boundaries
UY1=1200 ! P1 = (1800,1200) & P2 = (15500,9500)
UY2=9500 ! For B-size Paper On HP 7550A Plotter
490 IF (Size.EQ.'F') THEN
Psize='FULL'
XP1=1350-PXOFF
XP2=10000-PXOFF ! FULL A-size Paper Plotting Boundaries
YP1=1000-PYOFF ! P1 = (1350,1000) & P2 = (10000,7500)
YP2=7500-PYOFF ! Offset P1,P2 For HP 7550A Plotter
GO TO 500
END IF
Psize='NOTEBOOK'
XP1=1350-PXOFF
XP2=8500-PXOFF ! NOTEBOOK Size Plotting Boundaries
YP1=1000-PYOFF ! P1 = (1350,1000) & P2 = (8500,7000)
YP2=7000-PYOFF ! Offset P1,P2 For HP 7550A Plotter
500 WRITE (PU,FMT-510)
510 FORMAT ('SPl;PU;AP;TL1;DII,0;IW;')
WRITE (PU,*) 'IP',XP1,C,YP1,C,XP2,C,YP2,';'
Xleft='0'
Xright='10000'
XOFF-XMIN ! Scales User Units To 0-10000 Range
IF (XMIN.GT.XMAX) THEN
Xleft='-10000'
Xright='0'
END IF
XMULT=T4/(XMAX-XMIN)
Ylow='0'
Ytop='100000'
YMULT Is Set In Y-scale Routines
For A Scaled Range Of 0-10000 Units
WRITE (PU,*) 'SC',WIDTH,',',HEIGHT,';VS'.Vaxes.':'
RATIO-0.6
LWIDTH=WIDTH*RATIO ! LITTLE Size Characters Default To 0.6 Of The User Set Character Size
LHEIGHT=HEIGHT*RATIO
RATIO=4.0/3.0
BWIDTH=WIDTH*RATIO ! BIG Size Characters Are Kept In Proportion To The User Set Size
BHEIGHT=HEIGHT*RATIO
WRITE (PU,*) 'SR',WIDTH,',',HEIGHT,';VS'.Vaxes.':'
520 WRITE (1,*) HOME,CLR,'_'
CALL Line(NCOL)
WRITE (1,530) 'CODE','FUNCTION','MIN','MAX','INC'
530 FORMAT (T4,A4,T18,A8,T40,A3,T50,A3,T60,A3)

CALL Line(NCOL)
WRITE (1,540) 'PL',...'Plot Limits...............',WN,WX
FORMAT (T4,A4,T18,A8,T40,A3,T50,A3,T60,A3)
CALL Line(NCOL)
WRITE (1,540) 'PL',...'Plot Limits...............',WN,WX
WRITE (1,550) 'XT',...'X-tick Spacing ............',FXT,LXT,Xtick
WRITE (1,550) 'XT',...'X-tick Spacing ............',FXT,LXT,Xtick
WRITE (1,570) 'YL',...'Y-label Spacing ..........',FYL,YLY,Ylabel
WRITE (1,570) 'YL',...'Y-label Spacing ..........',FYL,YLY,Ylabel
WRITE (1,590) 'AX',...'Axes Size .................',Psize,'',
WRITE (1,590) 'AX',...'Axes Size .................',Psize,'',
WRITE (1,610) 'AN',...'Annotate Plot .............',Lmode,Pattern.
WRITE (1,610) 'AN',...'Annotate Plot .............',Lmode,Pattern.
WRITE (1,610) 'DD' .......DIGITIZE/DISPLAY DATA...
WRITE (1,610) 'DD' .......DIGITIZE/DISPLAY DATA...
WRITE (1,*)
WRITE (1,*)
WRITE (1,530) 'YS',...'Y-scale....................',
WRITE (1,530) 'YS',...'Y-scale....................',
WRITE (1,540) 'EX',...'EXIT PLOTTING MODE......
WRITE (1,540) 'EX',...'EXIT PLOTTING MODE......
WRITE (1,550) 'YT',...'Y-tick Spacing ..........',
WRITE (1,550) 'YT',...'Y-tick Spacing ..........',
WRITE (1,560) 'YS',...'Y-scale....................',
WRITE (1,560) 'YS',...'Y-scale....................',
WRITE (1,570) 'YL',...'Y-label Spacing ..........',
WRITE (1,570) 'YL',...'Y-label Spacing ..........',
WRITE (1,580) 'CS',...'Char.Size (Width,Height)',WlDTH,HEICHT
WRITE (1,580) 'CS',...'Char.Size (Width,Height)',WlDTH,HEICHT
WRITE (1,590) 'LT','....Line Type................',Lmode,Pattern.
WRITE (1,590) 'LT','....Line Type................',Lmode,Pattern.
WRITE (1,*)
WRITE (1,*)
WRITE (1,540) 'AX',...'Axes Size .................',Psize,'',
WRITE (1,540) 'AX',...'Axes Size .................',Psize,'',
WRITE (1,610) 'AN',...'Annotate Plot .............',Lmode,Pattern.
WRITE (1,610) 'AN',...'Annotate Plot .............',Lmode,Pattern.
WRITE (1,610) 'DD' .......DIGITIZE/DISPLAY DATA...
WRITE (1,610) 'DD' .......DIGITIZE/DISPLAY DATA...
WRITE (1,*)
WRITE (1,*)
WRITE (1,530) 'YS',...'Y-scale....................',
WRITE (1,530) 'YS',...'Y-scale....................',
WRITE (1,540) 'EX',...'EXIT PLOTTING MODE......
WRITE (1,540) 'EX',...'EXIT PLOTTING MODE......
WRITE (1,550) 'YT',...'Y-tick Spacing ..........',
WRITE (1,550) 'YT',...'Y-tick Spacing ..........',
WRITE (1,560) 'YS',...'Y-scale....................',
WRITE (1,560) 'YS',...'Y-scale....................',
WRITE (1,570) 'YL',...'Y-label Spacing ..........',
WRITE (1,570) 'YL',...'Y-label Spacing ..........',
WRITE (1,580) 'CS',...'Char.Size (Width,Height)',WlDTH,HEICHT
WRITE (1,580) 'CS',...'Char.Size (Width,Height)',WlDTH,HEICHT
WRITE (1,590) 'LT','....Line Type................',Lmode,Pattern.
WRITE (1,590) 'LT','....Line Type ................',Lmode,Pattern.
WRITE (1,*)
WRITE (1,*)
WRITE (1,540) 'AX',...'Axes Size .................',Psize,'',
WRITE (1,540) 'AX',...'Axes Size .................',Psize,'',
WRITE (1,610) 'AN',...'Annotate Plot .............',Lmode,Pattern.
WRITE (1,610) 'AN',...'Annotate Plot .............',Lmode,Pattern.
WRITE (1,610) 'DD' .......DIGITIZE/DISPLAY DATA...
WRITE (1,610) 'DD' .......DIGITIZE/DISPLAY DATA...
WRITE (1,*)
WRITE (1,*)
WRITE (1,530) 'YS',...'Y-scale....................',
WRITE (1,530) 'YS',...'Y-scale....................',
IF (Pcode.EQ.'AX') GO TO 890
IF (Pcode.EQ.'EX') GO TO 980
IF (Pcode.EQ.'PA') GO TO 1000
IF (Pcode.EQ.'AN') GO TO 1100
IF (Pcode.EQ.'PS') GO TO 2000
IF (Pcode.EQ.'DD') GO TO 3000
GO TO 500

700 WRITE (1,*) UP,ERASE, 'Plot Limits: (Min,Max)', BELL, '_'
READ (1,*,ERR=700) WN,WX
CALL Order(WN,WX)
710 IF ( (WN.LT.WMIN).OR.(WN.GT.WMAX) ) WN=WMIN
IF ( (WX.LT.WMIN).OR.(WX.GT.WMAX) ) WX=WMAX
GO TO 500

720 WRITE (1,*) UP,ERASE, 'X-scale: (Min,Max)', BELL, '_'
READ (1,*,ERR=720) XMIN,XMAX
XN=XMIN ! X-axis Is Bidirectional So Only
XX=XMAX ! XN,XX Are Ordered For Tick & Label
CALL Order(XN,XX)
CALL Xaxis(XN,XX) ! Set Limits, Tick & Label Spacing
GO TO 710

730 WRITE (1,*) UP,ERASE, 'X-tick: (First,Last,Space)', BELL, '_'
READ (1,*,ERR=730) XN,XX,XTick
CALL Order(XN,XX)
IF ( (XN.LT.XMIN).OR.(XX.GT.XMAX) ) GO TO 730
XTick=ABS(XTick)
FXT=XN
LXT=XX
Xlabel=XTick
GO TO 750

740 WRITE (1,*) UP,ERASE, 'X-label: (First,Last,Space)', BELL, '_'
READ (1,*,ERR=740) XN,XX,Xlabel
CALL Order(XN,XX)
IF ( (XN.LT.XMIN).OR.(XX.GT.XMAX) ) GO TO 740
Xlabel=ABS(Xlabel)
FXL=XN
LXL=XX
GO TO 500

750 EOFF--YN
IF (Ycode.NE.'E') C TO 800

FORMAT (A2,A2,A20,A6,A37,A.A)
READ (1,99) Icode
CALL Upper(Icode)
IF (Icode.EQ.'A') GO TO 800

44
IF (Icode.EQ.'U') THEN
  EMULT=10.0
  GO TO 790
END IF

IF (Icode.EQ.'D') THEN
  EMULT=0.1
  GO TO 790
END IF

GO TO 770

790 EScaled=EScale*EMULT / Update Ext. Coeff. Scale
RATIO=EScale/Scale / Power Of 10 For Exponent
YMULT=RATIO/FACTOR(J) / Scaling Factor For Ext. Coeff.
CALL Exponent(RATIO,Mult) / Convert Exponent To String
GO TO 770

800 YSCALE=(YX-YN)/YMULT / YMULT Alters Scaling For The
YOFF=-OFF/YMULT / Extinction Coefficient Moda
CALL Yaxis(YSCALE,Ycode) / Set Tick & Label Spacing
GO TO 500

810 WRITE (1,*) UP,ERASE,' Y-tick: (First,Last,Space) ',BELL,'__'
READ (1,*,ERR=810) YN,YX,Ytick
CALL Order(YN,YX)
Ytick=ABS(Ytick)/YMULT
FYt=(YN+EOFF)/YMULT
LYt=(YX+EOFF)/YMULT
Ylabel=Ytick
GO TO 830

820 WRITE (1,*) UP,ERASE,' Y-label: (First,Last,Space) ',BELL,'__'
READ (1,*,ERR=820) YN,YX,Ylabel
CALL Order(YN,YX)
Ylabel=ABS(Ylabel)/YMULT
Fyi=(YN+EOFF)/YMULT
LyI=(YX+EOFF)/YMULT
GO TO 500

840 WRITE (1,*) UP,ERASE,' Char.size: (Width,Height) ',BELL,'__'
READ (1,*,ERR=840) WIDTH,HEIGHT
IF (Pcode.EQ.'AN') GO TO 1140 / Return To Annotation Mode
GO TO 500

850 WRITE (1,*) UP,ERASE,' Pen Velocity, (Axes,Spectrum): ',BELL,'__'
READ (1,*,ERR=850) X,Y
IF ((X.LT.1.0).OR.(X.GT.36.0)) GO TO 850
IF ((Y.LT.1.0).OR.(Y.GT.36.0)) GO TO 850
CALL Str(X,String,2)
Vaxes=String(2:3)
CALL Str(Y,String,2)
Vspec=String(2:3)
GO TO 500

45
900 860 WRITE (1,*) UP, ERASE, ' Line Type: (S.....Solid, B.....Broken)'
901 &BELL,'_'
902 READ (1,99) Icode
903 CALL Upper(Icode)
904 IF (Icode.EQ.'S') THEN
905 Lmode='SOLID'
906 Pattern=' '
907 Length=' '
908 WRITE (PU,*) 'LT,'
909 GO TO 500
910 END IF
911 IF (Icode.NE.'B') GO TO 860
912 Lmode='PATTERN'
913 870 WRITE (1,*) UP, ERASE, ' Pattern #: (1-6) ',BELL,'_'
914 READ (1,'(12)',ERR=870) N
915 IF ((N.LT.1).OR.(N.GT.6)) GO TO 870
916 NUMBER=FLOAT(N)
917 CALL Str(NUMBER,String,1)
918 Pattern(1:1)=#''
919 Pattern(3:3)=String(2:2)
920 880 WRITE (1,*) UP, ERASE, ' Pattern Length: (0-10)% ',BELL,'_'
921 READ (1,'(1,7.1)',ERR=880) NUMBER
922 IF ((NUMBER.LT.0.0).OR.(NUMBER.GT.10.0)) GO TO 880
923 CALL Str(NUMBER,String,2)
924 K=2
925 DO WHILE (String(K:K).NE.'')
926 K=K+1
927 END DO
928 Length=String(2:K)
929 WRITE (PU,*) 'LT',Pattern(3:3),',',Length,';'
930 Length(K:K)='%'
931 GO TO 500
932 C----------------------------------------------------------------------------------
933 890 WRITE (1,*) UP, ERASE, ' F....FULL, N....NOTEBOOK, U....USER SET ?',BELL,'_'
934 READ (1,99) Size
935 CALL Upper(Size)
936 IF (Size.EQ.'F') GO TO 490
937 IF (Size.EQ.'N') GO TO 490
938 IF (Size.NE.'U') GO TO 890
939 Psize='USER SET'
940 900 WRITE (1,*) UP, ERASE, ' Lower Left: (X,Y) = '
941 WRITE (1,'(1,910)',UX1,',',UY1,BELL,'_'
942 910 FORMAT (15,A15,': A...Alter or RETURN ? .A.A)
943 READ (1,99) Icode
944 CALL Upper(Icode)
945 IF (Icode.EQ.' ') GO TO 940
946 IF (Icode.NE.'A') GO TO 900
947 920 WRITE (1,'(1,930)',UP,ERASE,BELL,' ? ',
948 930 FORMAT (T23,A2,A2,A3)
949 READ (1,'(1,9)',UX1,UY1)
950 UX1=ABS(UX1)
951 UY1=ABS(UY1)
953  XPI=UX1
954  YP1=UY1
955  IF (XPI.GT.16000) GO TO 920
956  IF (YP1.GT.11400) GO TO 920
957  WRITE (1,*) UP,ERASE,' Upper Right: (X,Y) = '
958  WRITE (1,910) UX2,',',UY2,BELL,'_
959  READ (1,99) Icode
960  CALL Upper(Icode)
961  IF (Icode.EQ.' ') GO TO 970
962  IF (Icode.NE.'A') GO TO 950
963  WRITE (1,930) UP,ERASE,BELL,'?
964  READ (1,*,ERR-960) UX2,UY2
965  UX2=ABS(UX2)
966  UY2=ABS(UY2)
967  XP2=UX2
968  YP2=UY2
969  IF (XP2.GT.16000) GO TO 960
970  IF (YP2.GT.10100) GO TO 960
971  IF ((XPI.GE.XP2).OR.(YP1.GE.YP2)) GO TO 900
972  IF ((XP2-XP1).LT.1000) GO TO 900
973  IF ((YP2-YP1).LT.1000) GO TO 900
974  GO TO 500
975  C
976  WRITE (PU,*) 'SPO,'
977  CLOSE (PU, IOSTAT=N,ERR-9999)
978  GO TO MENU
979  C
980  C
981  AXES PLOTTING ROUTINE
982  C
983  - Scaling is ON -
984  C
985  Y-axis: 0-10000 user units
986  X-axis: 0-10000 user units (increasing -> right)
987  -10000-0 user units (decreasing -> right)
988  C
989  Only integral position values are sent in PA commands
990  for compatibility with early model HP plotters.
991  C
992  XMULT & YMULT variables scale data to 10000 digits
993  for full scale, ensuring that no round-off errors
994  affect plotting resolution on early model HP plotters
995  which do not accept decimal fractions in SC or PA.
996  C
997  Real & Integer values are sent within the HP-GL commands
998  as appropriate - some computers send extra nulls in these
999  modes causing errors on early HP plotters. In such
1000  cases the values must be converted to string literals
1001  first. The present form works correctly on the HP1000.
1002  C
1003  C
1004  C
1005  C
WRITE (1,*) UP, ERASE, ’Plotting Axes: ’, BELL, ’_’
WRITE (PU,*) ’VS’, Vaxes, ’; LT; PA’, Xleft, C, Ylow,’; PD;’
WRITE (PU,*) ’PA’, Xright, C, Ylow, C, Xright, C, Ytop, C, Xleft, C, Ytop, C,
& Xleft, C, Ylow,’; PU;’
DX=1E-5*ABS(XMAX) ! Delta X & Y ensure completion of
DY=1E-5*ABS(YSCALE) ! DO loops with fractional steps
------------------------------------------
DO 1010 V=FXT, LXT+DX, Xtick
 I=RO((V-XOFF)*X.MULT)
 WRITE (PU,*) ’PA’, I,’,’ , Ylow,’; XT;’
CONTINUE
------------------------------------------
DO 1020 V=FXL, LXDT+DX, Xlabel
 I=RO((V-XOFF)*X.MULT)
 WRITE (PU,*) ’PA’, I,’,’ , Ylow,’; ’
 X=R2(V)
 CALL Str(X, String, 4)
------------------------------------------
Find # of digits in string, ignoring sign & trailing blanks
 K=2
 DO WHILE (String(K:K).NE. ’ ‘) 
 K=K+1
 END DO
 X=(FLOAT(K-2)-0.33)/2.0
 WRITE (PU,*), X, ’; ’
 & ’SR’, WIDTH,’; ’
 END IF
IF (Xcode.EQ.’W’) THEN
 Xstring=’Wavenumber x 10’
 COFF=7.5+0.75-0.33
 END IF
String=’5000’
 IF (XMULT.LT.0.0) String=’-5000’
 WRITE (PU,*) ’CP’, ’- ’, -1; LB’, String(2:K-1), Etx
CONTINUE
------------------------------------------
DO 1040 V=FYT, LYT+DY, Ytick
 I=RO(V*T4/yscale)
 WRITE (PU,*) ’PA’, Xleft,’,’ , I,’,’ , YT;’
CONTINUE
------------------------------------------
DO 1050 V=FYL, LYL+DY, Ylabel
 Y=R3(V-YOFF)
 IF (Ycode.EQ. ’E’) THEN
 Y=R2((V-YOFF)*ESCALE/FACTOR(J)/SCALE)
 END IF

I=RO(V*T4/YSRACE)
WRITE (PU,*), 'PA', Xleft, ',', I, '.'
CALL Str(Y, String, 4)
IF (String.EQ. '0.0') String= '0'
C Find # of digits in string, ignoring sign & trailing blanks
K=2
DO WHILE (String(K:K).NE. ' ')
K=K+1
END DO
Y=FLOAT(K-1)+.33
WRITE (PU,*), 'CP', -Y, ',', .25; LB', String(1:K-1), Etx
CONTINUE
WRITE (PU,*), 'PA', Xleft, ',5000; SR', BWIDTH, ', ', BHEIGHT,
& '; CP-.5,0; DIO,1; '
IF (Ycode.EQ. 'E') THEN
K=5
IF (Mult(5:5).EQ. ' ') K=4
IF (Mult(5:5).EQ. '.') K=1
Ystring= '/Mult(1:K)
COFF= (FLOAT(K)+13.5)/2.0-1.5
IF (K.EQ.1) COFF=COFF-0.5
WRITE (PU,*), 'CP', -COFF, ', 0; CP-.33, .5; DR0, -1; LB3', Etx
WRITE (PU,*), 'CP0, .25; SR', BWIDTH, ': ', .HEIGHT, ':; LB-1', Etx
WRITE (PU,*), 'SR', BWIDTH, ': ', .BHEIGHT, ':; CP-.5, -.25; LBcm', Etx
WRITE (PU,*), 'CP0, .25; SR', BWIDTH, ': ', .HEIGHT, ':; LB-1', Etx
WRITE (PU,*), 'SR', BWIDTH, ': ', .BHEIGHT, ':
WRITE (PU,*), 'CP0, -.25; LB', Etx, ':; D11,0; '
GO TO 1060
END IF
IF (Ycode.EQ. 'A') K=10
'994 IF (Ycode.EQ. 'R') K=13
IF (Ycode.EQ. 'T') K=14
COFF=FLOAT(K)/2.0-.167
WRITE (PU,*), 'CP', -COFF, ', 0; CP-.33, .5; Ystring(1:K), Etx ':; D11,0; '
WRITE (1,*) UP, ERASE, ' Plot a dashed baseline, (Y or N) ? '
& BELLS,'_
READ (1,99) Icode
CALL Upper(Icode)
IF (Icode.EQ. 'N') GO TO 1090
IF (Icode.EQ. 'Y') GO TO 1070
WRITE (1,*) UP, ERASE, ' Dash Length: (1-5)% ', BELL, '_'
READ (1,*) X
IF ((X.LT.1.0).OR.(X.GT.5.0)) GO TO 1080
IX=R0(X)
IZ=R0(YOFF*T4/YSRACE)
1112 WRITE (PU,*,'LT2',';IX',';PA',Xright,';IZ',';PD;PA','Xleft, &',';IZ',';PU;LT;'
1113 1090 IF (Psize.EQ.'NOTEBOOK') GO TO 1200
1115 GO TO 2080
1116 C
1117 C
1118 C
1119 C
1120 C
1121 C
1122 C
1123 1100 WRITE (1,*) UP,ERASE,' L...Labels, P...Parameters, T...Title', &', X...Exit ? ',BELL,'_'
1125 READ (1,99) Icode
1126 CALL Upper(Icode)
1127 IF (Icode.EQ.'T') GO TO 1110
1128 IF (Icode.EQ.'L') GO TO 1120
1129 IF (Icode.EQ.'P') GO TO 1200
1130 IF (Icode.EQ.'X') GO TO 2080
1131 GO TO 1100
1132 C
1133 1110 TITLE=LABEL(J)
1134 WRITE (1,'(A2,A2,A4,A72,T76,A)') UP,ERASE,' ? <','TITLE,'>'
1135 WRITE (1,'(T5,A2,A,A)') UP,BELL,'_'
1136 READ (1,120) TITLE
1137 IF (TITLE.EQ.' ') TITLE=LABEL(J)
1138 WRITE (1,*) UP,ERASE,' Plotting Title: ',BELL
1139 K=72
1140 DO WHILE (TITLE(K:K).EQ.' ')
1141 K=K-1
1142 END DO
1143 COFF=FLOAT(K)/2.0
1144 IX=5000
1145 IF (XMULT.LT.0.0) IX=-5000
1146 IX=!
1147 IF (Psize.EQ.'NOTEBOOK') IY=2
1148 WRITE (PU,*)'PA',IX,C,Ytop,';SR1,2;CP',-COFF,'...IY,'':'
1149 WRITE (PU,*)'LB',TITLE(1:K),Etx
1150 CO TO 1100
1151 C
1152 1120 WRITE (1,'(A2,A2,A13,T58,A)') UP,ERASE,' Label: ? <','>'
1153 WRITE (';','(T14,A2,A,A)') UP,BELL,'_'
1154 READ (1,'(A40)') Text
1155 IF (Text.EQ.' ') GO TO 1100
1156 K=40
1157 DO WHILE (Text(K:K).EQ. ' ')
1158 Y=K-1
1159 END DO
1160 1130 WRITE (1,*) UP,ERASE,' Label: ',Text(1:K)',OK (Y or N) ?', &BELL,'_'
1161 &BELL,'_'
1162 READ (1,99) Icode
1163 CALL Upper(Icode)
1164 IF (Icode.EQ.'N') GO TO 1120
1165 IF (Icode.NE.'Y') GO TO 1130
WRITE (1,*) UP,ERASE, ' C...Char.size, M...Move, P...Plot, ',
α'V...Vplot, X...Exit ?, 'BELL,'_'
READ (1,99) Icode
CALL Upper(Icode)
IF (Icode.EQ.'C') GO TO 1150
IF (Icode.EQ.'M') GO TO 1160
IF (Icode.EQ.'V') GO TO 1170
IF (Icode.EQ.'P') GO TO 1180
IF (Icode.NE.'X') GO TO 1140
GO TO 1100
C
WRITE (1,*) UP,ERASE, ' Char.Size (Width,Height) ? ',BELL,'-
READ (1,*,ERR=1150) LWIDTH,LHEIGHT
GO TO 1140
C
WRITE (1,*) UP,ERASE, ' Coordinate, 0-100%: (X,Y) ? ',BELL,'-
READ (1,*,ERR=1160) X,Y
IX=R0(X*100.0+(XMIN-XOFF)*MULT)
IY=R0(Y*100.0)
WRITE (PT,*)'PA',IX,',',IY,';
GO TO 1140
C
Special Entry Point From Digitize Routine (Pcode.EQ.'DD')
Special Offset Controls Are Used For Digitize Mode Labels
C
WRITE (PU,*) 'D[0,1;' !Rotate Labelling Axis 90 Degrees
WRITE (1,*) UP,ERASE, ' C...Centered, L...Left Justified, ',
&'R...Right Justified ? ',BELL,'-
READ (1,99) Icode
CALL Upper(Icode)
IF (Icode.EQ.'C') THEN
  X=-FLOAT(K)/2.0
IF (Pcode.EQ.'DD') THEN
  X=X-0.5 !Alter Centering For Leading Blank
IF (K.GT.8" THEN !Alter Centering For Labelling Both
  I=2 !X & Y Coordinates In Digitize Mode
  DO WHILE (Text(I:I).NE.' ') !Find Space Between X & Y Labels
    I=I+1
  END DO
  X=-FLOAT(I+1)
END IF
END IF
IF (Icode.EQ.'R') THEN
  X=-FLOAT(K)
IF (Pcode.EQ.'D" THEN
  X=X-2.0 !Alter Right Justification To Stop
END IF
END IF
IF (Icode.NE.'L') GO TO 1180
X=0.0 !Convert X=0.0 To '0.0' With Str
1219 IF (Pcode.EQ.'DD') THEN
1220 X=1.0  ! Alter Left Justification To Stop
1221 END IF  ! Peak Label Writing Over Spectrum
1222 CALL Str(X,String,4)
1223 WRITE (1,*) UP,ERASE,', Plotting Label:',BELL
1224 WRITE (PU,*) 'SR',LWIDTH,',',LHEIGHT,',',CP',String(1:5),',-.25:'
1225 WRITE (PU,*) 'LB',Text(1:K),Etx,',',CP0,.25;DII,0,'
1226 IF (Pcode.EQ.'DD') THEN
1227 IX=NINT(T4*FLOAT(IX-X1)/FLOAT(X2-X1)+(XMIN-XOFF)*MULT)
1228 IY=NINT(T4*FLOAT(IY-Y1)/FLOAT(Y2-Y1))
1229 WRITE (PU,*) 'PA',IX,':',IY,';PU,'
1230 GO TO 3000  ! Return To Digitize Routine
1231 END IF
1232 GO TO 1100
1233
c
1234 1200 IF (Psize.EQ.'FULL') GO TO 1100
1235 1210 WRITE (1,*) UP,ERASE,' Plot Parameters, (Y or N) ?',BELL,'
1236 READ (1,99) Icode
1237 CALL Upper(Icode)
1238 IF (Icode.EQ.'N') GO TO 1100
1239 IF (Icode.NE.'Y') GO TO 1210
1240 WRITE (1,*) UP,ERASE,' Plotting Parameters:',BELL
1241 WRITE (PU,*) 'PA',Xright,C,Ytop,';'
1242 &CR,LF,LF,Etx
1243 WRITE (PU,*) 'SR.75,2;CP0,-.5;LB',LF,LF,' File: ',Fname(J),
1244 &CR,LF,LF,Etx
1245 WRITE (PU,*) 'LB Date: ',DATE(J),CR,LF,LF,Etx
1246 WRITE (PU,*) 'LB Conc. (M): ',CONC(J),CR,LF,LF,Etx
1247 WRITE (PU,*) 'LB Path (cm): ',PATH(J),CR,LF,LF,Etx
1248 WRITE (PU,*) 'LB Rate (nm/sec): ',
1249 &Pstr(3,PARAM(J,3)+I),CR,LF,LF,Etx
1250 WRITE (PU,*) 'LB Period (sec): ',
1251 &Pstr(15,PARAM(J,15)+1),CR,LF,LF,Etx
1252 WRITE (PU,*) 'LB Ref. Mode: ',
1253 &Pstr(5,PARAM(J,5)+1),CR,LF,LF,Etx
1254 WRITE (PU,*) 'LB Beam Mode: ',
1255 &Pstr(16,PARAM(J,16)+1),CR,LF,LF,Etx
1256 WRITE (PU,*) 'LB SBW (nm) ':,VARIABLE(J,10),CR,LF,LF,Etx
1257 WRITE (PU,*) 'LB Slit Gain: ',VARIABLE(J,6),CR,LF,LF,Etx
1258 WRITE (PU,*) 'LB Slit Height: ',
1259 &Pstr(23,PARAM(J,23)+1),Etx
1260 GO TO 1100
1261
c
1262
c
1263
c
1264
c
1265
c
1266
c
1267
c
1268
c
1269
c
1270
c
1271
c
1272
c

******************************************************************************** END OF AXES PLOTTING ********************************************************************************

SPECTRUM PLOTTING ROUTINE

- Velocity Select = 5 cm/sec (Paper), 10 cm/sec (Acetate)
- Real variables are used in 'VS' commands
- Scaling in ON with Absorbance * 1E4 to match axes
- Both Integer and String Numbers are used in 'PA' commands
- HP I/O Subsystem Performs Binary <-> ASCII Conversions

52
WRITE (1,*) UP, ERASE, ' Increase step size, (Y or N) ? ', BELL, '_'
READ (1,99) Icode
CALL Upper(Icode)
IF (Icode.EQ.'N') GO TO 2020
IF (Icode.NE.'Y') GO TO 2000
WRITE (1,*) UP, ERASE, ' Step Multiplier, (>1): ', BELL, '_'
READ (1,'(13)',ERR=2010) M
IF (M.LT.1) MI-1
2010 WRITE (1,*) UP, ERASE, 'Step Multiplier, (>1): ', BELL, '_'
READ (1,'(13)',ERR=2010) M
IF (M.LT.1) MI-1
2020 X-STEP*FLOAT(M)
CALL Str(X,String,4)
C
Find # of digits, ignoring sign & trailing blanks
1286 C
1287 C
K=2
DO WHILE (String(K:K).NE.' ')
K=K+1
END DO
WRITE (1,*) UP, ERASE, 'Plotting Spectrum @ ', String(2:K-1), & ' nm steps: ', BELL, '_'
C
WINDOW SETTING ROUTINE
C
- Soft clip limits are set to P1 & P2 scaling coordinates so that off scale data do not plot on graph. This should not happen normally since the plot boundaries are made to match the X-scale limits automatically if the latter are smaller than the scan range.
- Plotter sends output parameters as A,B,C,D,CR,LF where (A-D) are string integers (ASCII), which are read into integer variables (ASCII translation is automatic).
- Some computers handshake on CR leaving L1 in buffer. In these cases use READ A,B,C,D,E (N.B. CHARACTER E*1) to clear plotter buffer before the next READ cycle.
C
2030 WRITE (PU,*) 'VS',Vspec, ';OP;'
READ (PU,*) X1,Y1,X2,Y2
WRITE (PU,*) 'IW',X1,C,Y1,C,X2,C,Y2,';'
IF (Xcode.EQ.'N') THEN
START-RO((WMAX-WX)/STEP)+1
FINISH-RO(FLOAT(ND)-(WN-WMIN)/STEP)
END IF
START-RO((T4/WMIN-T4/WN)/STEP)+1
FINISH-RO(FLOAT(ND)-(T4/WX-T4/WMAX)/STEP)
2040 I=0
RATIO-FACTOR(1)/FACTOR(2)
DO 2070 K=START,FINISH,M
X=W(J,K)
! Acquired Wavelength
IF (Xcode.EQ.'N') THEN
X=WX*FLOAT(I)*STEP ! Calculated Wavelength
IX=R0((X-XOFF)*MULT)
GO TO 2050
END IF
X=T4/WN*FLOAT(I)*STEP ! Calculated Wavelength
IX=R0((T4/X-XOFF)*MULT)
2050 IF (Code.EQ.'DS') THEN
IY=R0(((A(1,K)-A(2,K)*RATIO)+YOFF)*T4/YSCALE)
GO TO 2060
END IF
IY=R0((A(J,K)+YOFF)*T4/YSCALE)
2060 WRITE (PU,*), 'PA',IX,C,IY,';PD;'
I=I+M
2070 CONTINUE

WRITE (PU,*) 'PU;LT;' Lmode='SOLID'
Pattern=''
Length=''
GO TO 500

PLOTTER DIGITIZING ROUTINE
- HP-GL command 'OA' provides pen position without using
the clumsy ENTER key on the plotter.
The 'DP' & 'OD' commands are used with ENTER in cases
where the plotter is remote from the keyboard.

3000 WRITE (1,*), HOME,CLR,'_'
TITLE='Digitize Pen Position'
CALL Center(TITLE)
CALL Line(NCOL)
CALL Digitize(X,Xcode,Y,Ext,K)
WRITE (1,3010) 'X-position = ',X,Xord
String='Absorbance = '
IF (Ycode.EQ.'R') String=' % Reflect. = '
IF (Ycode.EQ.'T') String=' % Transmit. = '
WRITE (1,3020) String,Y
WRITE (1,3030) 'Ext.Coeff. = '
WRITE (1,*) Ext(1:K),'/M/cm'
END IF
3010 FORMAT (/,T5,A14,T20,F5.3,A7)
3020 FORMAT (/,T5,A14,T20,F6.4)
3030 FORMAT (/,T5,A14)
3040 FORMAT (/,T5,A14)
WRITE (1,*) DOWN
30
WRITE (1,3040) 'D.....Digitize Position'
WRITE (1,3040) 'L.....Label At Position'
WRITE (1,3040) 'S.....Segment Display'
WRITE (1,3040) 'X.....EXIT To Plotting'
3040 FORMAT (/ , T16, A23)
WRITE (1,* ) DOWN, DOWN
3050 WRITE (1,'(T10,A2,A2,A6,A,A)') UP, ERASE, 'Code: ', BELL, '_'
READ (1,99) Icode
CALL Upper(Icode)
IF (Icode.EQ.'X') GO TO 500
IF (Icode.EQ.'D') GO TO 3000
IF (Icode.EQ.'S') GO TO 4000
IF (Icode.NE.'L') GO TO 3050
3060 WRITE (1,*) UP, ERASE, 'Place PEN In Stall =1, '
& 'Then Press RETURN ', BELL, '_'
READ (1,99) Icode
CALL Upper(Icode)
IF (Icode.NE.') GO TO 3060
CALL Digitize(X,Xcode,Y,Ext,K)
3070 WRITE (1,'(T10,A2,A2,A6,A,A)') UP, ERASE, 'X.....X-label, Y.....Y-label, B.....Both',
& ' Q.....Quit ? ', BELL, '_'
READ (1,99) Icode
CALL Upper(Icode)
IF (Icode.EQ.'Q') GO TO 3000
IF (Icode.EQ.'Y') GO TO 3080
IF ((Icode.NE.'X').AND.(Icode.NE.'B')) GO TO 3070
CALL Str(X,String,5)
Text=String ! String*14, Text*40 And TITLE*?2
DO WHILE (Text(K:K).EQ.') ! Are Reusable String Variables
  K=K-1 ! Find Start Of Trailing Blanks
END DO
IF (Icode.EQ.'X') GO TO 3110
C---------------------------
3080 IF (Icode.EQ.'E') THEN
  String=Ext
  GO TO 3090
END IF
CALL Str(Y,String,5)
3090 IF (Icode.EQ.'B') THEN
  TITLE=Text(1:K)//' ('//String
  Text=TITLE(1:40)
  GO TO 3100
END IF
Text=String
K=40
3100 K=K-1 ! Find Start Of Trailing Blanks
DO WHILE (Text(K:K).EQ.')
  K=K-1
END DO
IF (Icode.EQ.'B') THEN
  Text(K+1:K+2)-'
END IF
3110 GO TO 1170 ! Transfer To Vertical Label Routine
DISPLAY Segment Of Spectrum

4000 String=" (nm)'
   IF (Xcode.EQ. 'W') String=' (cm-1) x 1E-3'
1412 WRITE (1,*) UP,ERASE,' Scan Range: ',WMIN,' to',WMAX,'_'
1413 WRITE (1,*) String,DOWN,DOWN
1414 4010 WRITE (1,*) UP,ERASE,' DISPLAY Range: (MIN,MAX) ? ',BELL,'_'
1415 READ (1,*,ERR=4010) W1,W2
1416 CALL Order(W1,W2)
1417 IF ((W1.LT.WMIN).OR.(W1.GT.WMAX)) GO TO 4010
1418 IF ((W2.LT.WMIN).OR.(W2.GT.WMAX)) GO TO 4010
1419 IF (Xcode.EQ. 'N') THEN
1420 START-RO((WMAX-W2)/STEP)+1
1421 FINISH-RO(FLOAT(ND)-(W1-WMIN)/STEP)
1422 GO TO 4020
1423 END IF
1424 START-RO((T4/WMIN T4/W1)/STEP)+1
1425 FINISH-RO(FLOAT(ND)-(T4/W2-T4/WMAX)/STEP)
1426 4020 IF ((Ycode.EQ. 'R').OR.(Ycode.EQ. 'T')) THEN
1427 Text=string
1428 IF (Ycode.EQ. 'R') Text=' '//string
1429 TITLE='Energy Wavelength '//Text(l:15)
1430 GO TO 4030
1431 END IF
1432 TITLE='Energy Wavelength Absorbance Ext.Coeff. (/M/cm)'
1433 4030 WRITE (1,*),HOME,CLR,'_
1434 CALL Center(TITLE)
1435 CALL Line(60)
1436 L=1
1437 DO 4050 I=START,FINISH
1438 X=T4/W(J,I)
1439 Y=A(J,I)
1440 IF (Code.EQ. 'DS') Y=A(1,1)-A(2,1)*FACTOR(1)/FACTOR(2)
1441 WRITE (1,*)
1442 IF ((Ycode.EQ. 'A').OR.(Ycode.EQ. 'E')) THEN
1443 WRITE (1,4060) X,W(J,I),Y,Y/FACTOR(J)
1444 ELSE
1445 WRITE (1,4070) X,W(J,I),Y
1446 END IF
1447 L=L+1
1448 IF (I.EQ. 10) THEN
1449 WRITE (1,*) DOWN
1450 4040 WRITE (1,4080) UP,ERASE,'Press RETURN to Continue ',BELL,'_
1451 READ (1,99) Icode
1452 IF (Icode.NE. ' ') GO TO 4040
1453 WRITE (1,*) HOME,CLR,'_
1454 CALL Center(TITLE)
1455 CALL Line(60)
1456 L=I
56
END IF
CONTINUE
FORMAT (T9,F7.3,T21,F7.2,T35,F7.4,T49,G10.4)
FORMAT (T14,F7.3,T30,F7.2,T47,F7.2)
FORMAT (T10,A2,A2,A25,A,A)
WRITE (1,*) DOWN
WRITE (1,4080) UP,ERASE, 'Press RETURN for MENU , BELL, '_
READ (1,99) Icode
IF (Icode.NE.' ') GO TO 4090
GO TO 3000
Write Program
-------------------------------------------------
READ (1,99) Icode
IF (Icode.NE.' ') GO TO 4090
GO TO 3000
Exit Program
-------------------------------------------------
READ (1,99) Icode
IF (Icode.NE.' ') GO TO 3000
GO TO 3000
STOP
C
C
C
9000 WRITE (1,*) UP,ERASE, UP
STOP
C
C
IEEE-488 Error Exit
-------------------------------------------------
WRITE (1,*)' Error #',N
STOP
END
C
**************************** END OF MAIN PROGRAM ****************************
C
Print a TITLE Centered in 72 columns
-------------------------------------------------
SUBROUTINE Center(TITLE)
INTEGER I,J,N
CHARACTER TITLE*72,BLANK*36
BLANK=''
I=72
J=0
DO WHILE (ICHAR(TITLE(I:I)).EQ.32)
J=J+1
I=I-1
END DO
N=J/2
WRITE (1,*) BLANK(1:N),TITLE(1:I)
RETURN
END
Extinction Coefficient Rounding

SUBROUTINE Coeff(E,String,K)

INTEGER F,K,M
REAL E,S
CHARACTER*(*)String
CHARACTER Mult*3

F=0
15:0 S=1.0
1551: IF (E.LT.0.0) S=-1.0
1552 E=ABS(E)
1553 IF ((E.NE.0.0).AND.(E.NE.1.0)) GO TO 10
1554 String=' 0.0000'
1555 IF (E.EQ.1.0) String=' 1.0000'
1556 K=7
1557 GO TO 50
1558 10 IF ((E.GT.1.0).AND.(E.LT.10.0)) THEN
1559 K=K+1
1560 GO TO 30
1561 END IF
1562 IF (E.GE.10.0) GO TO 20
1563 DO WHILE (E.LT.1.0)
1564 E=E*10.0
1565 F=F-1
1566 END DO
1567 GO TO 30
1568 20 DO WHILE (E.GE.10.0)
1569 E=E/10.0
1570 F=F+1
1571 END DO
1572 30 E=E*S
1573 CALL Str(E,String,K)
1574 40 K=2
1575 DO WHILE (String(K:K).NE.' ') K=K+1
1576 END DO
1577 IF (F.EQ.0) GO TO 50
1578 E=FLOAT(F)
1580 CALL Str(E,Mult,2)
1581 IF (Mult(1:1).EQ.' ') Mult(1:1)='+'
1582 M=3
1583 IF (Mult(3:3).EQ.' ') M=2
1584 String(K:K+M)='E'/Mult(1:M)
1585 K=K+M
1586 50 RETURN
1587 END
SUBROUTINE Digitize(X,Xcode,Y,Ext,K)
INTEGER IX,IY,IZ,J,K,PU,X1,X2,Y1,Y2
REAL FACTOR(2),NUMBER,XMIN,XMAX,YOFF,YSCALE,X,Y
CHARACTER*(*) Ext,Xcode
COMMON /DICIT/FACTOR,XMIN,XMAX,YOFF,YSCALE,IX,IY,J,PU,X1,X2,Y1,Y2
WRITE (PU,*)(OP;'
READ (PU,*) XI,Y1,X2,Y2
WRITE (PU,*)(OA;'
READ (PU,*) IX,IY,IZ
X-(XMAX-XMIN)*FLOAT(IX-X1)/FLOAT(X2-X1)+XMIN
IF (Xcode.EQ.'W') THEN
X-ANINT(X*1000.0)
GO TO 10
END IF
X-ANINT(X*10.0)/10.0
10 Y-YSCALE*FLOAT(IY-Y1)/FLOAT(Y2-Y1)-YOFF
K=4
! K = # Of Digits Precision
NUMBER=NUMBER/FACTOR(J)
CALL Coef(NUMBER,Ext,K)
! K = # Of Characters In String
RETURN
END

SUBROUTINE Exponent(N,Mult)
INTEGER I
REAL M,N
CHARACTER String*14
CHARACTER(*) Mult
M=0.0
N=ABS(N)
I=INT(N+.5)
IF (I.EQ.1) THEN
Mult='1 ...'
GO TO 20
END IF
IF (I.GT.1) GO TO 10
DO WHILE (I.LT.1)
N=N*10.0
M=M-1.0
I=INT(N+.5)
END DO
CALL Str(M, String, 2)
Mult='IE-'/String(2:3)
GO TO 20
10 DO WHILE (I.GT.1)
   N=N/10.0
   M=M+1.0
   I=INT(N+.5)
END DO
CALL Str(M, String, 2)
Mult='IE+'//String(2:3)
20 RETURN
END

C--------------------------------------------------------------------------------------------------------
C
C---------------------------------------------------------------
SUBROUTINE Line(N)
   INTEGER I,N
   CHARACTER BLANK*72, DLINE*72, SPACE*36
   SPACE=SPACE//SPACE
   BLANK=SPACE//SPACE
   DLINE=SPACE//SPACE
   IF (N.GT.72) N=72
   I=(72-N)/2
   WRITE (1,* ) BLANK(1:I), DLINE(1:N)
   RETURN
END

C--------------------------------------------------------------------------------------------------------
C
C-------------------------------------------------------------------------------
SUBROUTINE Order(MIN,MAX)
   REAL MIN, MAX, SWAP
   IF (MIN.LT.MAX) GO TO 10
   SWAP=MIN
   MIN=MAX
   MAX=SWAP
10 RETURN
END
Rescale Data Array Between Absorbance And Transmission

\$EMA/DATA/

SUBROUTINE Rescale(J,K,ND)
INTEGER J,K,ND
REAL A(2,10001)
COMMON /DATA/A
IF (K.EQ.1) GO TO 20
DO 10 I=1,ND
A(J,I)=ALOG10(100.0/A(J,I))
10 CONTINUE
RETURN
20 DO 30 I=1,ND
A(J,I)=100.0/10.0**(A(J,I))
30 CONTINUE
RETURN
END

Convert Number To ASCII String

SUBROUTINE Str(VALUE,String,PREC)
INTEGER ASCII,DECPT,I,J,LENSTR,NDIGIT,NUMBER,PREC
REAL VALUE
DOUBLE PRECISION DECIMAL,FRACTIONTEN
CHARACTER Concat*14,Digit(12),Sign,String*14
LOGICAL INTEGER
LOGICAL TEST
TEST=.FALSE.
DECPT=0
J=0
TEN=10.0
Sign=''
Concat=''
IF (TEST) WRITE (1,*) ' Value Entered = ',VALUE
IF (VALUE.LT.0.0) Sign='-'
IF (VALUE.EQ.0.0) GO TO 100
DECIMAL=ABS(VALUE)
DO WHILE (DECIMAL.GE.1.0)
DECIMAL=DECIMAL/TEN
J=J+1
END DO
DECPT=J
IF (TEST) WRITE (1,*) ' # of Whole Digits: ',DECPT
1740 IF (DECPT.EQ.0) GO TO 30
1741 DO 20 J=1,DECPT
1742 DECIMAL=DECIMAL*TEN
1743 NUMBER=INT(DECIMAL)
1744 ASCII=NUMBER+48
1745 Digit(J)=CHAR(ASCII)
1746 FRACTION=DECIMAL-NUMBER
1747 DECIMAL=DINT(FRACTION*TEN**(PREC-J)+.5)/TEN**(PREC-J)
1748 20 CONTINUE
1749 IF (.NOT.TEST) GO TO 30
1750 C WRITE (1,*) ' The Whole Digits = ',(Digit(I), I=1,DECPT)
1751 30 J=DECPT
1752 C IF (TEST) WRITE (1,*) ' Decimal Fraction = ',DECIMAL
1753 IF (DECIMAL.NE.0.0) INTEGER=.FALSE.
1754 IF (DECPT.GE.12) GO TO 40
1755 DO WHILE (DECIMAL.NE.0.0)
1756 J=J+1
1757 DECIMAL=DECIMAL*TEN
1758 NUMBER=INT(DECIMAL)
1759 ASCII=NUMBER+48
1760 Digit(J)=CHAR(ASCII)
1761 FRACTION=DECIMAL-NUMBER
1762 DECIMAL=DINT(FRACTION*TEN**(PREC-J)+.5)/TEN**(PREC-J)
1763 IF (DECIMAL.EQ.1.0) THEN
1764 Digit(J)=CHAR(ASCII+1)
1765 DECIMAL=0.0
1766 END IF
1767 IF (J.GE.12) DECIMAL=0.0
1768 END DO
1769 40 NDIGIT=J
1770 C IF (.NOT.TEST) GO TO 50
1771 C WRITE (1,*) ' The Characters = ',(Digit(I), I=1,NDIGIT)
1772 50 IF (NDIGIT.GT.12) GO TO 200
1773 DO 60 I=1,NDIGIT
1774 Concat(I:1)=Digit(I)
1775 60 CONTINUE
1776 IF (INTEGER) GO TO 80
1777 IF (DECPT.EQ.0) GO TO 70
1778 String=Sign//Concat(1:DECPT)//'.'//Concat(DECPT+1:14)
1779 RETURN
1780 70 String=Sign//'.'//Concat
1781 RETURN
1782 80 String=Sign//'.'//Concat
1783 RETURN
1784 100 String=' 0.0'
1785 RETURN
1786 20C WRITE (1,*) ' Error in data: (too many digits)'
1787 STOP
1788 END
SUBROUTINE Upper(Code)
  INTEGER LENSTR,N
  CHARACTER(*) Code
  LENSTR=LEN(Code)
  DO 10 I=1,LENSTR
    N=ICHAR(Code(I:1))
    IF (N.GT.96) Code(I:1)=CHAR(N-32)
  10 CONTINUE
  RETURN
END
SUBROUTINE Val(String, VALUE)

INTEGER DECPT, EXPON, LENSTR, N, NUM(10)
REAL VALUE
DOUBLE PRECISION MULT, SIGN, TEN, DECIMAL

CHARACTER Ascii

INTEGER .TRUE.

J = 1
K = 0
DECPT = 0
SIGN = 1.0
TEN = 10.0
DECIMAL = 0.0
LENSTR = LEN(String)

IF (TEST) WRITE (1, *) ' String Number = ', String

IF (TEST) WRITE (1, *) ' String Length = ', LENSTR

DO 100 I = 1, LENSTR

Ascii = STRING(1:1)
N = ICHAR(Ascii)

IF ((N GE 48).AND.(N LE 57)) GO TO 20

IF (N EQ 46) INTEGER = .FALSE.

IF (N EQ 45) DECPT = K

IF (N EQ 45) SIGN = -1.0

GO TO 100

20 NUM(J) = N - 48

K = J

J = J + 1

100 CONTINUE

IF ((DECPT .EQ. 0).AND.(INTEGER), DECPT = K

DO 200 J = 1, K

EXPON = DECPT - J

MULT = TEN**EXPON

DECIMAL = DECIMAL + NUM(J)*MULT

200 CONTINUE

VALUE = SIGN*DECIMAL

IF (TEST) WRITE (1, *) ' Value = ', VALUE

RETURN

END
SUBROUTINE Xaxis(XL,XH)

REAL  LMIN,LMAX,WN,WX,Xtick,FXT,LXT,Xlabel,FXL,LXL,XH,XL, XS(17)

COMMON /XPARAM/LMIN,LMAX,WN,WX,Xtick,FXT,LXT,Xlabel,FXL,LXL

DATA (XS(I),I=1,17)/.05,.1,.2,.25,.5,.1,.2,.5,.5,1.,2.,10.,20.,25.,
&50.,100.,200.,250.,500./

LMIN-XL  LMAX-XH

WN-XL  WX-XH

C

Xtick-0.1*(XH-XL)  ! Set Xtick To 1/10 X-scale

IF ((Xtick.LT.XS(2)).OR.(Xtick.GT.XS(17))) THEN

FXT=XL+Xtick  ! Test For Xtick Outside The

LXT=XH-Xtick  ! Preferred Interval Range

Xlabel=Xtick

FXL=FXT

LXL=LXT

GO TO 10  ! Use Default 1/10 Interval

END IF

I=1

DO WHILE (Xtick.GT.XS(I))  ! Compare 1/10 Scale Xtick Value

I=I+1  ! To Find The Nearest Preferred

END DO

Xtick=XS(I-1)

FXT=Xtick*AINT(XL/Xtick)+Xtick  ! Truncate -> FXT <= XL (+Xtick)

LXT=Xtick*AINT(XH/Xtick)  ! Truncate -> LXT <= XH

IF (ABS(XH-LXT).LT.0.01) THEN

LXT=LXT-Xtick  ! Decrement If LXT = XH

END IF

Xlabel=Xtick+2.0  ! Truncate -> FXL <= XL (+Xlabel)

FXL=Xlabel+AINT(XL/Xlabel)+Xlabel

LXL=Xlabel+AINT(XH/Xlabel)  ! Truncate -> LXL <= XH

IF (ABS(XH-LXL).LT.0.01) THEN

LXL=LXL-Xlabel  ! Decrement If LXL = XH

LXL=LXL-Xlabel

END IF

RETURN

10 RETURN
SUBROUTINE Yaxis(YSCALE,Ycode)
REAL Ytick,FYT,LYT,Ylabel,FYL,LYL,YSCALE,YS(17)

COMMON /YPARAM/Ytick,FYT,LYT,Ylabel,FYL,LYL

DATA (YS(I),I=1,17)/.0005,.001,.002,.005,.01,.02,.05,.1,.2,.5,
&1.,2.,5.,10.,20.,50.,100./

Ytick=0.1*YSCALE

IF ((Ytick.LT.YS(2)).OR.(Ytick.GT.YS(17)).OR.(Ycode.EQ..'E')) THEN
  FYT=Ytick
  LYT=YSCALE-Ytick
  Ylabel=Ytick
  FYL=FYT
  LYL=LYT
  GO TO 10
END IF

DO WHILE (Ytick.GT.YS(I)) ! Compare 1/10 Scale Ytick Value
  I=I+1          ! To Find The Nearest Preferred
  END DO ! Interval < 1/10 YSCALE

Ytick=YS(I-1)

FYT=Ytick

LYT=Ytick*AINT(YSCALE/Ytick) ! Truncate -> LYT <= YSCALE

IF (ABS(YSCALE-LYT).LT.1E-4) THEN ! Decrement If LYT = YS
  LYT=LYT-Ytick
END IF

Ylabel=Ytick*2.0

FYL=Ylabel

LYL=Ylabel*AINT(YSCALE/Ylabel) ! Truncate -> LYL <= YSCALE

IF (ABS(YSCALE-LYL).LT.1E-4) THEN ! Decrement If LYL = YSCALE
  LYL=LYL-Ylabel
END IF

RETURN

END
SUBROUTINE Wait(DELAY)
REAL DELAY, PERIOD, Tzero, Time
PERIOD = 0.0
Tzero = Time(I)
DO WHILE (PERIOD.LT.DELAY)
   PERIOD = Time(I) - Tzero
END DO
RETURN
END

REAL FUNCTION Time(I)
INTEGER ICODE, ITIME(S)
ICODE = 11
CALL EXEC(ICODE, ITIME)
Time = FLOAT(ITIME(1))/100.0 + FLOAT(ITIME(2)) + FLOAT(ITIME(3)) * 60.0 + FLOAT(ITIME(4)) * 3600.0
RETURN
END
Appendix

Sample Plots
Example Of Ext. Coeff. vs Energy Plot

Wavenumber x 10^{-3}
Example of Reverse X-scale Plot

\[ E \times 1 \times 10^{-4} \text{ (M}^{-1} \text{ cm}^{-1}) \]

\[ Wavenumber \times 10^{-3} \]