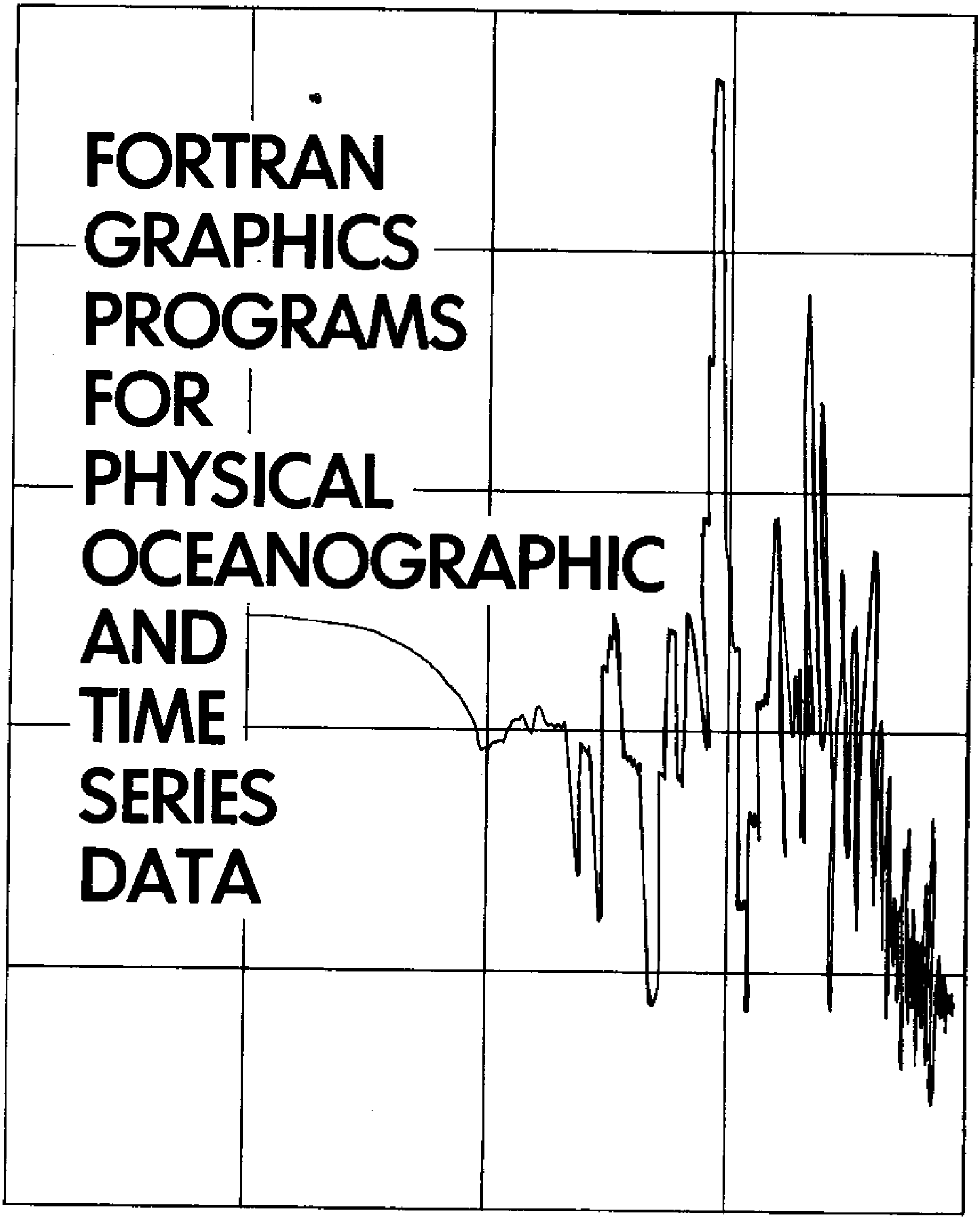
FORTRAN GRAPHICS PROGRAMS FOR PHYSICAL OCEANOGRAPHIC AND TIME SERIES DATA



William P. Kramer
Robert H. Weisberg

Graduate School of Oceanography
NOAA Sea Grant
University of Rhode Island

Marine Technical Report 46

ABSTRACT

A group of FORTRAN programs written to display time series data, X-Y data, and the results of time series analysis are documented. Separate programs are used to plot variables on any combination of linear and logarithmic (base 10) axes. Originally written to plot physical oceanographic data, these programs may be used in wider applications due to their flexibility. Parameters which determine plot size, numbering and labelling of axes are user supplied variables. The data storage device and format are also supplied at execution time.

All of these programs are stored in compiled form for use at URI. Listings are supplied in the appendices for other users.

Additional copies of Marine Technical Report 46 are available from the URI Marine Advisory Service, Narragansett Bay Campus, Narragansett, Rhode Island 02882.

Table of Contents

List of Figures

INTRODUCTION

A group of computer graphics programs have been written to satisfy two major needs: 1) plot variables typically sampled by physical oceanographers and 2) prepare plots of publishable quality. A great deal of flexibility is required to satisfy these basic needs and remain convenient and easy to understand. This document represents detailed documentation of this package of programs. The programs presented are of two groups: 1) programs that plot time series data sampled at constant intervals and 2) programs that plot variables resulting from time series analysis on linear or logarithmic (base 10) axes. Since these programs are independent of data type, they easily lend themselves to wider applications.

The programs in group one allow time series to be stacked one above the other (MULTYPLT) or plotted one over the other (OVERPLOT). Also, a series of vectors may be plotted versus time (STICK). The time axis for the programs in group one is prepared by subroutine TIMEPT allowing the arrangement of times and dates as needed by a user (see Appendix A).

The programs of group two prepare linear or logarithmic axes in various combinations. LINLIN will plot any number of records in X, Y form on axes that are linear in the X-direction and linear in the Y-direction. LINLOG plots a single variable on a linear axis versus frequency and period on a logarithmic axis. LOGLOG plots a single variable on a logarithmic axis versus frequency and period on a logarithmic axis. LOGLIN overplots two variables on a logarithmic axis versus frequency and period on a linear axis. Specifically, LINLIN applies to X, Y data such as temperature versus salinity or oxygen versus depth. The remaining programs of group two can plot any of the results of time series analysis (ie. phase, coherence, spectra, transfer function etc.).

Common Features

A number of special features make these programs convenient and flexible. Segmented plotting subroutines have been combined into FORTRAN main programs with many input options. These main programs are presently stored in compiled form for use from the Narragansett Bay Campus. Each of these modules is a complete ready-to-run program operating independently of the method of data computation. Users need not rewrite their computation programs to include plotting statements. Data that have already been computed and stored may now be conveniently accessed and plotted. Control cards for URI users are included with the documentation of each program.

Program flexibility allows users to create plots specific to their needs. User control is supplied in the form of various input parameters. By specifying a FORTRAN input unit reference number, data may be read from cards, disk or magnetic tape. Users also describe the input format for reading data points. Variables which control the numbering of the axes and the number of decimal characters are available. Labelling of the axes results from alphameric information supplied by the user. Choosing the height of characters printed on these plots allows for readability after any necessary photographic reduction. All of these parameters are required to satisfy a wide variety of demands.

Many of these programs were written to plot a variable number of records. When overplotting, multiplotting or simply plotting X,Y pairs, there is a need to distinquish one set of data from another. Fortunately, three plotting techniques are available to these programs: 1) data points may be spotted using a specified character, 2) data points may be connected with line segments or 3) data points may be spotted and connected. Typically, the value of the variable IOPTN determines the plotting technique.

Software

The URI Graphics Software Package is in the form of FORTRAN accessible subroutines which prepare plotter pen commands encoded as card images. The URI package is a combination of other plotting software packages resulting from work done by Mr. Roger Greenall of the URI computer center staff. Many of the mnemonics used by other major plotter companies are available.

Hardware

The following programs are run on an IBM 370/155 computer housed in Tyler Hall of the University of Rhode Island in Kingston, RI. Remote access to this system is achieved via a Cope 1200 batch station at the Narragansett Bay Campus. The Cope station, a product of Harris Communications, permits access of programs and data stored in Kingston. A Broomall Industries drum plotter is connected on-line to the Cope system and ultimately to the IBM 370/155 computer. Plotting can be done at rates of 800 to 2000 increments per second with an increment of .005 inch. Total plot size may vary from 12 to 30 inches depending on the adapter used.

MULTYPLT

MULTYPLT displays any number of time series as shown in Figure 2. Each variable must be sampled at some constant time interval, therefore only the dependent variable will be read. Time is reconstructed from a start position and a plotting point density. In this way, time series of different samples rates or different start times may be aligned. The general format of MULTYPLT is shown in Figure 1.

A vertical axis is drawn and numbered for each variable. The scale is chosen by the user allowing variables of different type to be combined in one plot. Two twenty character labels are used to annotate each variable with record number, data units or other information. The plotting technique and plot symbol are also chosen for each variable. For variables that span zero, the zero line is drawn.

The time axis is constructed from information on the time card which appears once at the end of each run. Enough flexibility is available to plot nearly any time increment.

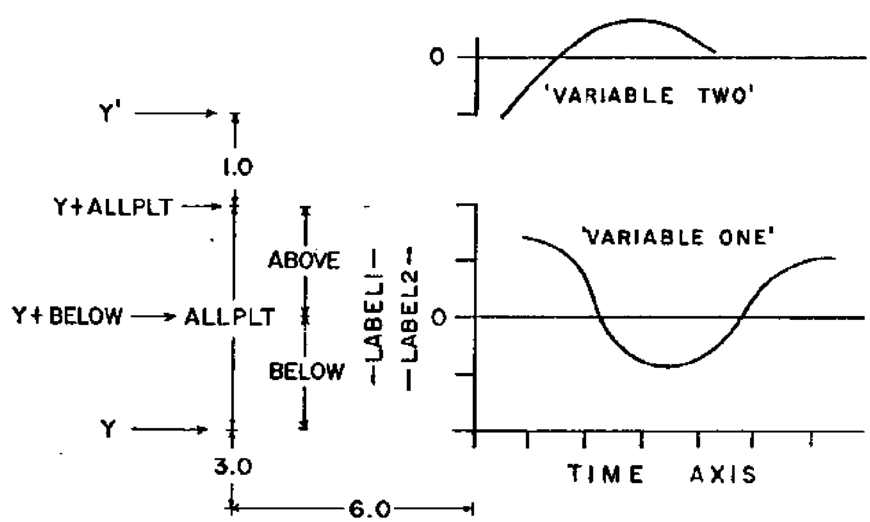


Figure 1. General arrangement of MULTYPLT. ABOVE and BELOW are computed distances.

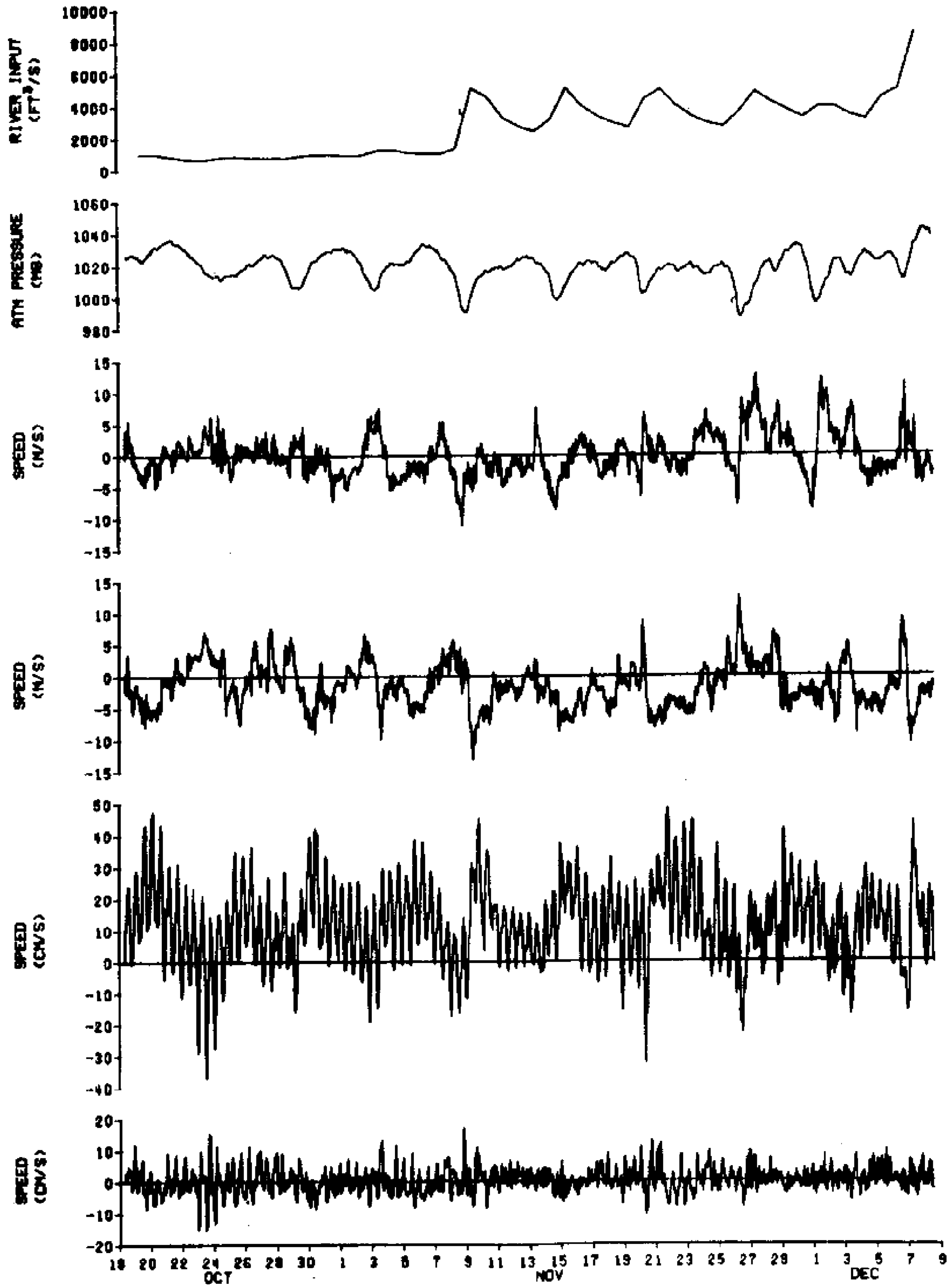


Figure 2. Sample MULTYPLT (from Weisberg, 1975).

MULTYPLT CONTROL CARDS

```
//MULTYPLT JOB (XXX100,128,1,1,1000),'USER'
/*ROUTE  PUNCH REMOTE2
//  EXEC PGMRUN,LIB=WATER,NAME=MULTYPLT,LIB1=TPLOT,LIB2=TPLOT,
//  REGION.GO=128K
//GO.FT01F001 . . .          }  OPTIONAL
//  LABEL=(1,SL) . . .       }
//GO.SYSIN DD *
   DATA CARD 1
   DATA CARD 2               }  Repeated for
   DATA CARD 3               }  each variable
   DATA ACQUISITION          }
   TIME CARD
//
```

MULTYPLT DATA CARDS

<u>CARD 1</u>

Cols 1 - 10	NVARBL	I10	This parameter defines the total number of records to be read and plotted. NVARBL is limited only by the amount of space on your plotter paper.
Cols 11 - 20	JAVA	I10	This overall plotter indicator determines if the basic plotter unit is an inch or a centimeter. If JAVA=0, all plot dimensions are in inches. If JAVA=10, all plot dimensions are in centimeters.

Cols 21 - 30	HEIGHT	F10.5	Specify the height of numbers and letters to be printed on the plot. Choose HEIGHT for readability after any necessary reduction.
Cols 31 - 40	LEAPYR	I10	This indicator controls the number of days in the month of February. If LEAPYR=0, February has 28 days. If LEAPYR≠0, February has 29 days.

CARD 2

Cols 1 - 10	XPOSTN	F10.5	XPOSTN is the horizontal offset used to align the data with the time information. The time information is typically started on a full day boundary at X=6.0 in plotter units (see Figure 1). The offset is computed by the user as explained in Appendix B.
Cols 11 - 20	SCALE	F10.5	SCALE is a value in data units equivalent to one vertical plotter unit. SCALE is strongly dependent on the type of data being plotted. A temperature record could be plotted from minimum to maximum at .5^{o}C increment per centimeter vertically while atmospheric pressure might be plotted at 200 millibars per vertical centimeter (SCALE=0.5 and 200, respectively).

Cols 21 - 30	PNTDEN	I10	PNTDEN is the number of points per inch (or centimeters) in the X direction. All time series plotted by MULTYPLT should be consistent in time, therefore PNTDEN should be adjusted with respect to individual sample rates.
Cols 31 - 40	LENGTH	I10	The number of points to be read (LENGTH $\leq$ 6000)
Cols 41 - 50	KINPUT	I10	Specify the input unit reference number. For card input, KINPUT=5. When tape or disk files are to be read, use KINPUT=1 to 4 or 8 to 99 and appropriate control cards to describe the data set.
Cols 51 - 60	IOPTN	I10	Pick the method of plotting from the following choices. If IOPTN=-1, spot data points. If IOPTN=0, connect data points. If IOPTN=1, connect and spot data points.
Cols 67	ISYMBL	A4	Punch the character to be used for spotting data points (when IOPTN$\neq$0). Any symbol is allowed but ones with a definite center are usually chosen.
Cols 71 - 80	NDECY	I10	Specify the number of digits to the right of the decimal point when numbering the Y-axis.

CARD 3

Cols 1 - 20	LABEL1	5A4	Two twenty character labels are printed
Cols 21 - 40	LABEL2	5A4	vertically for each series. Center these labels within the card columns.
Cols 41 - 80	FMT	10A4	Specify the format to be used in reading the variable. Begin with a left parenthesis in column 41 and use real FORTRAN format codes to describe the arrangement of the data points. Finish FMT with a right parenthesis.

Data Acquisition

The following READ statement is used to access the data.

READ(KINPUT,FMT)(DATA(I)I=1,LENGTH)

TIME CARD

Cols 1 - 10	NOWDAY	I10	The time card appears only once at the end
Cols 11 - 20	ITIS	I10	of each run. The time card contains the
Cols 21 - 30	NOWHR	I10	time and date to be printed at X=6.0.
Cols 31 - 40	NOWMIN	I10	Specify the day, number of the month, hour and minute.
Cols 41 - 50	IDADD	I10	Specify the time increment. Express that
Cols 51 - 60	IHRADD	I10	increment in days, hours and minutes.
Cols 61 - 70	MINADD	I10	
Cols 71 - 80	XINCRM	F10.5	Specify the distance between times to be printed on the plot.

Printed Output

The printed output contains a listing of all input parameters. This aids the user in diagnosing a faulty run. A sample MULTYPLT printed output appears as Figure 3.

INPUT PARAMETERS

NVARBL	JAVA	HEIGHT	LEAPYR
3	10	0,400	0

SERIES #	XPOSTN	SCALE	PNTDEN	LENGTH	KINPUT	IOPTN	ISYMBL	NDECY	FORMAT !
1	7,0000	250,00	4	8	5	0		0	(8F10,3)
2	6,5000	1,00	4	16	5	-1	X	0	(8F10,2)
3	6,2500	0,50	2	10	5	1	+	1	(16F5,2)

MULTYPLT COMPLETED

Figure 3. Sample MULTYPLT printed output.

OVERPLOT

OVERPLOT displays two time series on the same set of axes as shown in Figure 5. The vertical scale used in numbering the common Y-axis is specified by the user. The general arrangement of OVERPLOT is seen in Figure 4. Two twenty character labels are used to annotate the plot.

By reading parameters specific to each series, two time series of different start times, different sample rates or different lengths may be read and overplotted. By choosing a different plotting technique or plot symbol, the records can be easily distinquished. The time card, specified once at the end of the run, prepares a time axis of almost any time increment.

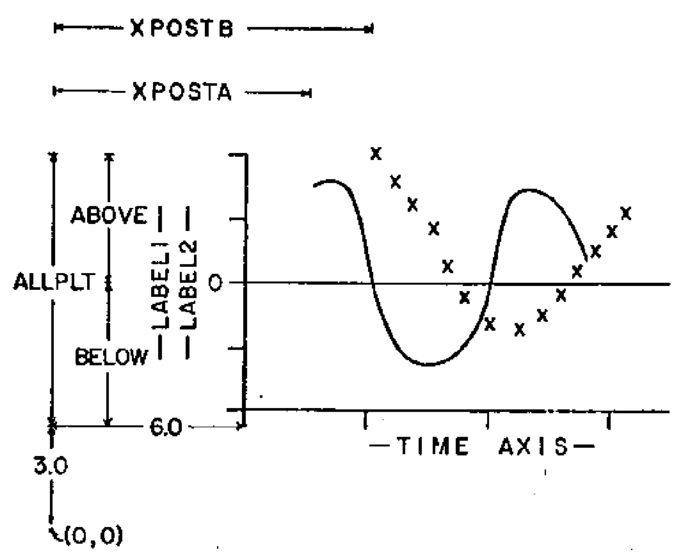


Figure 4. General arrangement of OVERPLOT. ABOVE and BELOW are computed distances.

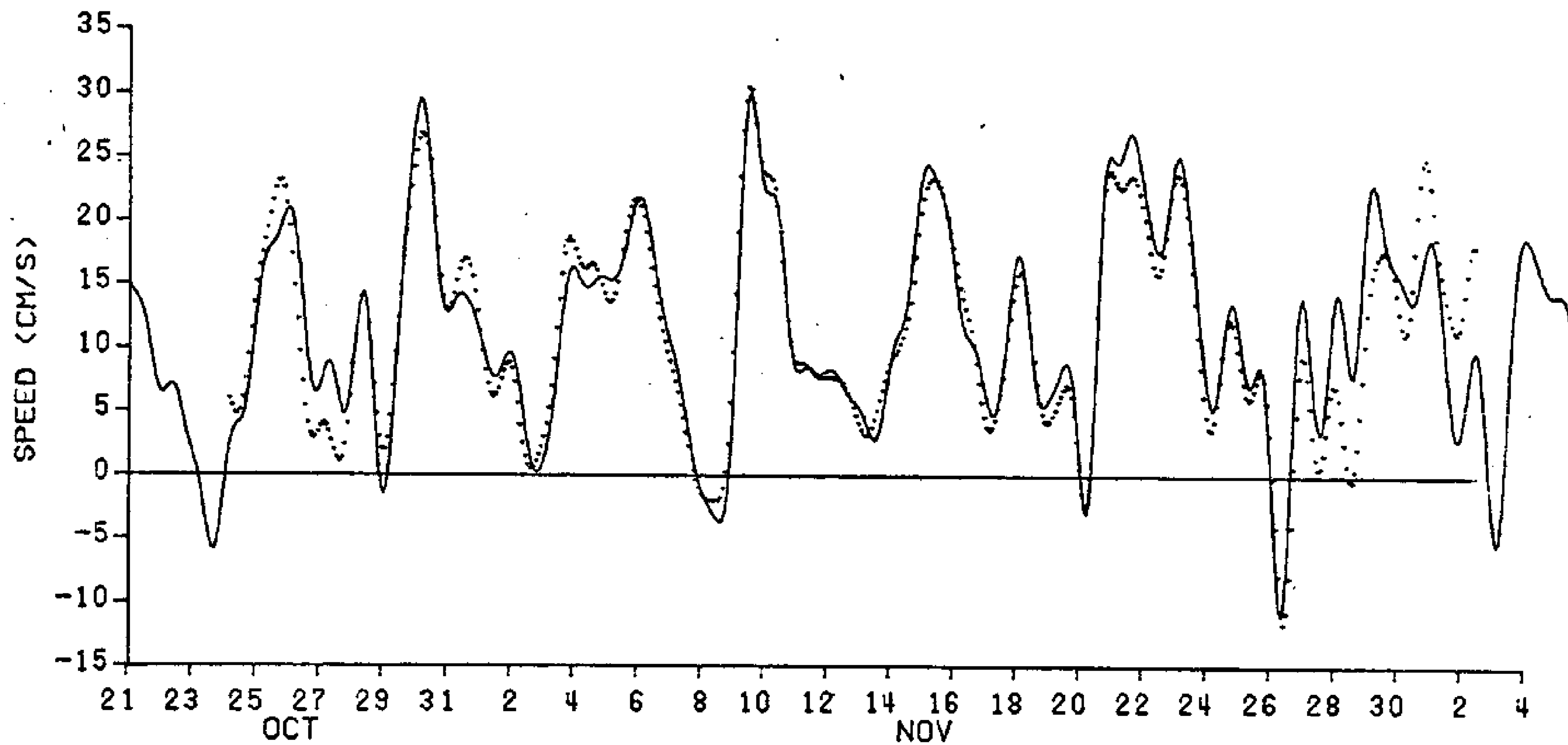


Figure 5. Sample OVERPLOT (from Weisberg, 1975).

OVERPLOT Control Cards

```
//OVERPLOT JOB (XXX100,128,1,1,500),'USER'
/*ROUTE  PUNCH REMOTE2
//  EXEC PGMRUN,LIB=WATER,NAME=OVERPLOT,LIB1=TPLOT,LIB2=TPLOT,
//  REGION.GO=128K
//GO.FT01F001 DD ...   }
//  LABEL=(1,SL) ...   } OPTIONAL
//GO.SYSIN DD *
   DATA CARD 1
   DATA CARD 2
   DATA CARD 3
   DATA ACQUISITION SERIES A
   DATA CARD 4
   DATA CARD 5
   DATA ACQUISITION SERIES B
   TIME CARD
//
```

OVERPLOT DATA CARDS

<u>CARD 1</u>

Cols 1 - 20	LABEL1	5A4	Two separate labels totalling forty columns are used for identifying the plot. The location and orientation of these labels are seen in Figure 4.
Cols 21 - 40	LABEL2	5A4	

Cols 41 - 50	HEIGHT	F10.5	The height of the numbers and letters printed on the plot must be specified.

CARD 2

Cols 1 - 10	XPOSTA	F10.5	The starting position for series A is specified. This is dependent on the start time of series A and the arrangement of the time axis.
Cols 11 - 20	ISCALE	I10	A whole number scale is chosen for the vertical axis. Data points are examined to find the maximum and minimum values, and corresponding distances are then computed.
Cols 21 - 30	DENSTA	I10	The horizontal plotting point density is in units of points per inch (or centimeters). The distance between points on the plot is therefore 1.0/DENSTA. Time series of different sample rates will have different plotting point densities.
Cols 31 - 40	LREAD	I10	The number of points to be read for series A. (LREAD $\leq$ 4000)
Cols 41 - 50	LPLOTA	I10	The number of points to be plotted for series A. (LPLOT $\leq$ LREAD)
Cols 51 - 60	IOPTNA	I10	Choose the method of plotting series A. If IOPTNA=-1, spot data points. If IOPTNA=0, connect data points with a line. If IOPTNA=1, connect and spot data points.

Col 67	ISYMBL(1)	A1	This field contains the character for spotting data points of series A when IOPTN≠0.

CARD 3

Cols 1 - 10	INPUTA	I10	Specify the input unit reference number. If data is to be read from cards, INPUTA=5. For other input units use appropriate control cards to describe the data set.
Cols 11 - 30	FMTA	5A4	Begin this field with a left parenthesis and use real FORTRAN format codes to describe the arrangement of data for series A. End the FMT with a right parenthesis.

Data Acquisition of series A

The following read statement is used to access series A.

```
READ(INPUTA,FMTA)(A(I),I=1,LREAD)
```

CARD 4

Cols 1 - 10	XPOSTB	F10.5	The horizontal starting position of series B. This depends on the times given on the time card and the plotting density chosen.

Cols 11 - 20	JAVA	I10	This field contains the overall plot scale indicator. If JAVA=0, plotter units are inches. If JAVA=10, plotter units are centimeters.
Cols 21 - 30	DENSTB	I10	Specify the horizontal plotting point density for series B.
Cols 31 - 40	LREAD	I10	The number of points to be read for series B. (LREAD $\leq$ 4000)
Cols 41 - 50	LPLOTB	I10	The number of points to be plotted for series B. (LPLOTB $\leq$ LREAD)
Cols 51 - 60	IOPTNB	I10	Choose the method of plotting series B. If IOPTNB=-1,spot data points. If IOPTNB=0, connect data points. If IOPTNB=1, connect and spot data points.
Col 67	ISYMBL(2)	A1	This field contains the character used to spot data points of series B when IOPTNB $\neq$0.

CARD 5

Cols 1 - 10	INPUTB	I10	Specify the input unit reference number for series B.
Cols 11 - 30	FMTB	5A4	Describe the format of series B. Begin with a left parenthesis in column 11 and

use real FORTRAN format codes. End FMTB with a right parenthesis.

Data Acquisition of series B.

The following read statement is used when accessing data for series B.

```
READ(INPUTB,FMTB)(B(I)=1,LREAD)
```

TIME CARD

Cols 1 - 10	NOWDAY	I10	These fields contain the first time to
Cols 11 - 20	ITIS	I10	be printed on the plot at X=6.0. Punch
Cols 21 - 30	NOWHR	I10	the day, number of month, hour, and minute.
Cols 31 - 40	NOWMIN	I10	This time is usually on a full day boundary, and XPOSTA and XPOSTB are shifted to an appropriate start location.
Cols 41 - 50	IDADD	I10	These variables specify the increment in
Cols 51 - 60	IHRADD	I10	time between ticks on the horizontal axis.
Cols 61 - 70	MINADD	I10	Express that time in days, hours, and minutes.
Cols 71 - 80	XINCRM	F10.5	Specify the distance between times printed.

Printed Output

The printed output contains a listing of the input parameters for series A and B(see Figure 6).

```
          INPUT PARAMETERS

   LABEL1              LABEL2            HEIGHT    LEAPYR

 RECORD GSC 120       FILTERED           0.300         0

XPOSTA    ISCALE    DENSTA    LREAD    LPLOTA    IOPTNA   ISYMBL(1)

 6.500       10         8       32        32         0

INPUTA    FMTA

    5     (16F5.2)

XPOSTB    JAVA      DENSTB    LREAD    LPLOTB    IOPTNB   ISYMBL(2)

 7.000    1000         4       16        16         1    +

INPUTB    FMTB

    5     (8F10.2)

      OVERPLOT COMPLETED
```

Figure 6. Sample OVERPLOT printed output.

STICK

Program STICK reads Cartesian coordinate components and displays them as a time series of the resultant vectors as shown in Figure 8. The first data card contains parameters of a general nature. It determines the number of records to be plotted, the plotting units and the height of printed characters. Remaining data cards contains parameters specific to each series indicating the input unit, the input format, the plotting scale and the number of points. Finally the time card is prepared indicating the exact arrangement of the time axis.

Any number of individual records may be plotted by STICK. One application is therefore to compare current meter records at similar depths or on similar moorings. The horizontal offset, XPOSTN, aligns records of different start times and space is provided for labels which appear on the plot as seen in Figure 7.

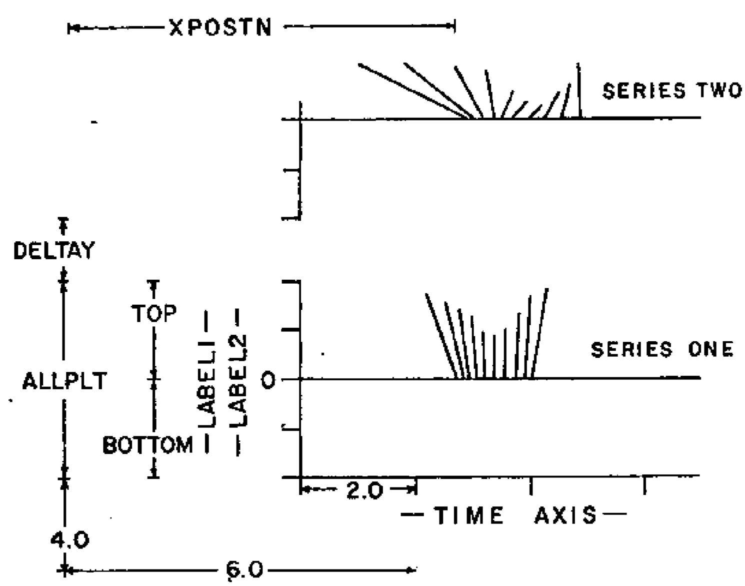


Figure 7. General arrangement of STICK. TOP and BOTTOM are computed distances.

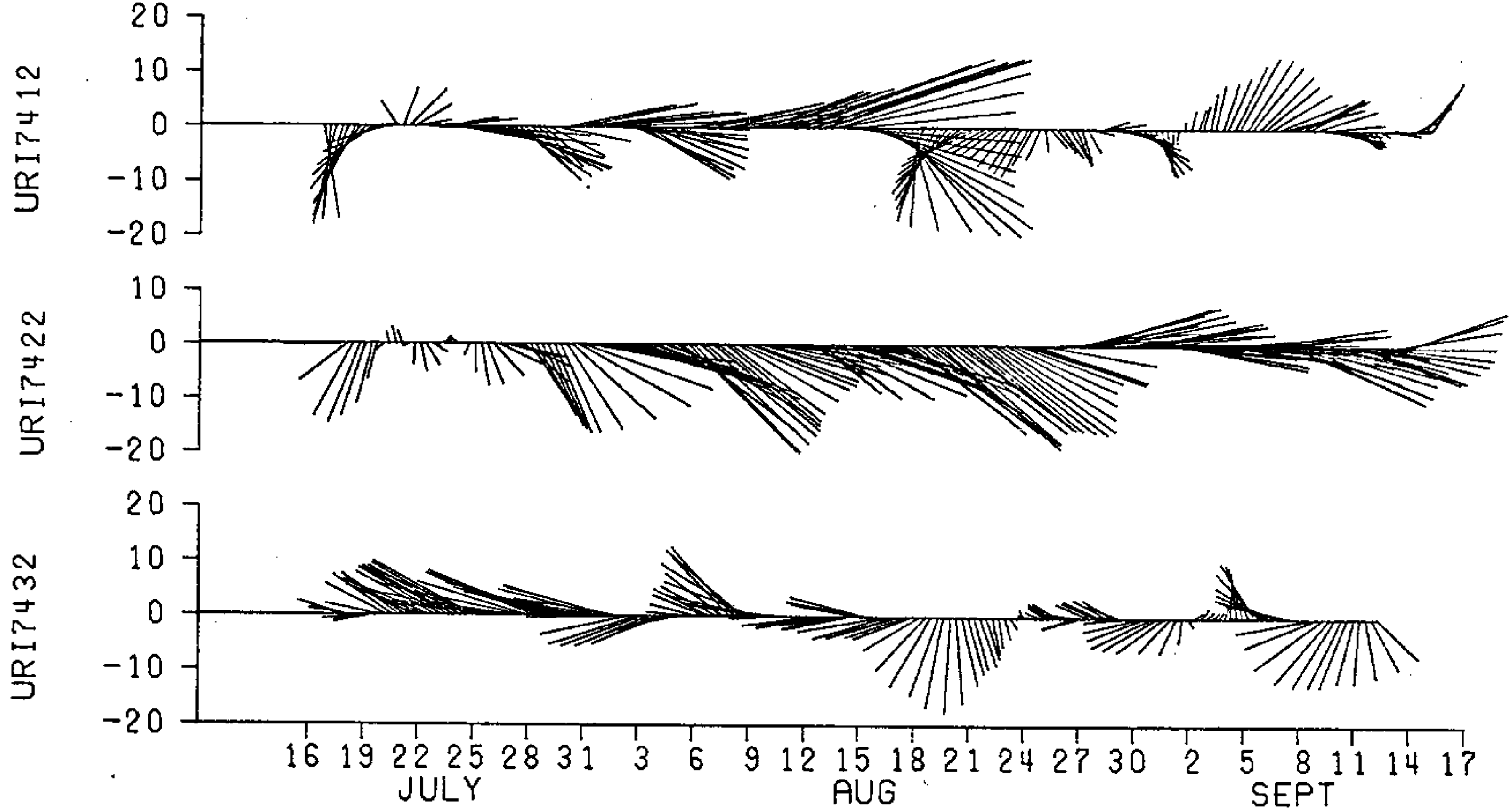


Figure 8. Sample STICK plot (Weisberg personal communication).

STICK CONTROL CARDS

```
//STICK JOB (XXX100,128,1,1,500),'USER'
/*ROUTE  PUNCH REMOTE2
//  EXEC PGMRUN,LIB=WATER,NAME=STICK,LIB1=TPLOT,LIB2=TPLOT,
//  REGION.GO=128K
//GO.FT01F001 DD ...   }  OPTIONAL
//  LABEL=(1,SL) ...   }
//GO.SYSIN DD *
   DATA CARD 1
   DATA CARD 2         }
   DATA CARD 3         }  REPEATED FOR EACH RECORD
   DATA ACQUISITION    }
   TIME CARD
//
```

STICK PLOT DATA CARDS

CARD 1

Cols 1 - 10	NPLOTS	I10	The number of records to be plotted Each record consists of a time series of U and V velocity components.
Cols 11 - 20	NUNITS	I10	All overall plot scale indicator. If NUNITS=0, the basic plotter units are inches. If NUNITS=10, the basic plotter units are centimeters.

Cols 21 - 30	DELTAY	F10.3	The vertical separation of records is specifeid (see Figure 7).
Cols 31 - 40	HEIGHT	F10.3	The height of letters and numbers printed on the plot must be specified. This allows characters to be readable even after reduction.
Cols 41 - 50	LEAPYR	I10	If LEAPYR=0, February has 28 days. If LEAPYR≠0, February has 29 days. This is necessary only when data span February.

CARD 2

Cols 1 - 10	XPOSTN	F10.5	Specify the start position of the record. This depends on the record start time and the arrangement of the time axis.
Cols 11 - 20	LREAD	I10	The number of pairs of points to be read (LREAD $\leq$ 4000)
Cols 21 - 30	LPLOT	I10	The number of pairs of points to be plotted (LPLOT $\leq$ LREAD)
Cols 31 - 40	IVERTS	I10	Choose the speed scale equivalent to one vertical plotter unit. This scale determines the size of the plot.
Cols 41 - 50	PNTDEN	I10	Specify the horizontal plotting point density. Be sure this number corresponds to the time axis as defined later.

Cols 51 - 60	IORDUV	I10	This indicator is used to specify the order of the input data. If IORDUV=0, cartesian components are read North-South then East-West. If IORDUV≠0, the order is East-West then North-South. The FMT field of card 4 is used to specify the details of this component order.
Cols 61 - 70	KINPUT	I10	Specify the input unit reference number.

CARD 3

Cols 1 - 20	LABEL1	5A4	These 40 columns contain the labels for each record. The labels are vertically centered on the axis, therefore labels shorter than 20 columns should be centered in this card field (see Figure 7).
Cols 21 - 30	LABEL2	5A4	
Cols 41 - 80	FMT	10A4	The specific format of the input components is specified here. Begin with a left parenthesis in Column 41 and use real FORTRAN format specifications to describe the arrangement of data points. End the FMT description with a right parenthesis.

DATA ACQUISITION

One of the following READ statements is used to access the data points.

If IORDUV=0,

READ(KINPUT,FMT)(V(I),U(I),I=1,LREAD)

If IORDUV≠0,

READ(KINPUT,FMT)(U(I),V(I)I=1,LREAD)

TIME CARD

Cols 1 - 10	NOWDAY	I10	These fields contain the first time to be printed at X=6.0. Punch the day, number of the month, hour and minute.
Cols 11 - 20	ITIS	I10	
Cols 21 - 30	NOWHR	I10	
Cols 31 - 40	NOWMIN	I10	
Cols 41 - 50	IDADD	I10	Specify the time increment. Punch the day, hour and minute.
Cols 51 - 60	IHRADD	I10	
Cols 61 - 70	MINADD	I10	
Cols 71 - 80	XINCRM	F10.5	Specify the distance between times printed on the plot.

Printed Output

The printed output contains a listing of the input parameters as seen in Figure 9.

INPUT PARAMETERS

NPLOTS	NUNITS	DELTAY	HEIGHT	LEAPYR
3	10	1.000	0.300	0

SERIES #	XPOSTN	LREAD	LPLOT	IVERTS	PNTDEN	IORDUV	KINPUT
1	6.5000	39	39	5	4	0	5
2	6.7500	83	83	5	8	0	5
3	6.9900	153	153	5	8	0	5

STICK PLOT COMPLETED

Figure 9. Sample STICK printed output.

LINLIN

LINLIN plots series of X, Y points on linear axes as shown in Figure 11. Any number of records may be plotted on the same set of axes. The minimum, maximum and numbering increment of both the X and Y-axes are user supplied variables as seen in Figure 10. In this way, the user controls the size and numbering of axes including the number of decimal digits. User supplied alphameric information appears as labels for the X and Y-axes. When a number of different records are plotted, they may be distinquished from each other by using a different plot technique or plot symbol.

Data in X, Y form may be read from any unit using any real FORTRAN format.

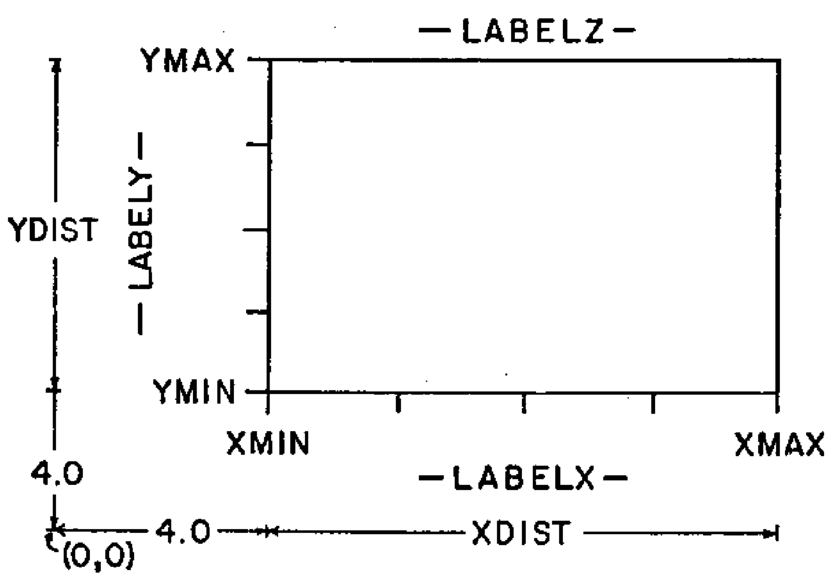


Figure 10. General arrangement of LINLIN. XDIST and YDIST are computed distances.

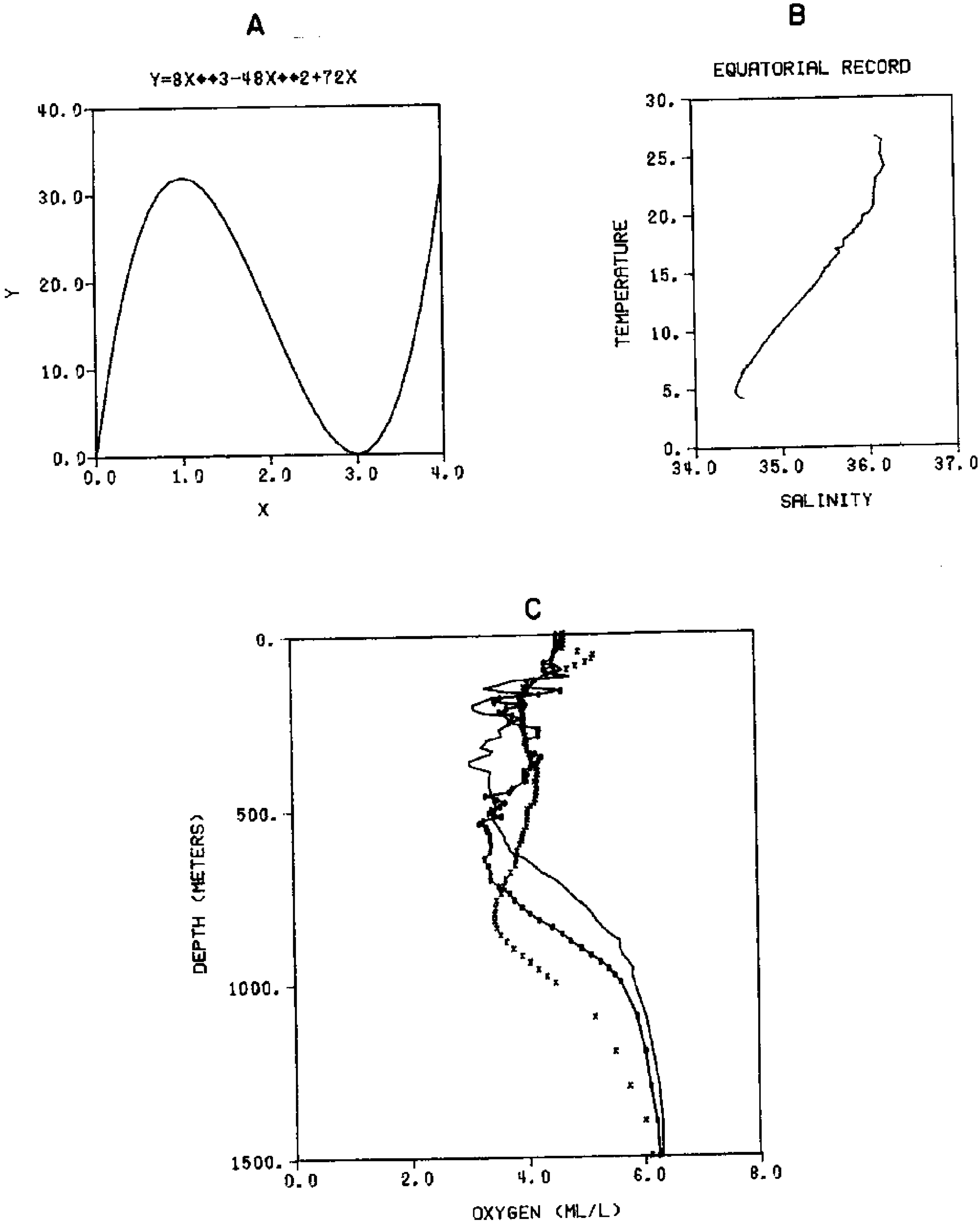


Figure 11. Sample LINLIN plots (B; Lambert, 1974 and C; Miller, personal communication).

LINLIN CONTROL CARDS

```
//LINLIN JOB (XXX100,128,1,1,500),'USER'
/*ROUTE  PUNCH REMOTE2
//  EXEC PGMRUN,LIB=WATER,NAME=LINLIN,LIB1=TPLOT,LIB2=TPLOT,
//  REGION.GO=128K
//GO.FT01F001 DD ...          }  OPTIONAL
//  LABEL=(1,SL,,IN) ...      }
//GO.SYSIN DD *
   DATA CARD 1
   DATA CARD 2
   DATA CARD 3
   DATA CARD 4
   DATA CARD 5                }  REPEATED FOR NVARBL'S
   DATA ACQUISITION           }
//
```

LINLIN DATA CARDS

CARD 1

Cols 1 - 10	XMIN	F10.5	The minimum for the X-axis in data units
Cols 11 - 20	XMAX	F10.5	The maximum for the X-axis in data units
Cols 21 - 30	DELTAX	F10.5	Specify the scale difference between numbers printed on the X-axis.

Cols 31 - 40	XINCRM	F10.5	Specify the distance between numbers printed on the X-axis.
Cols 41 - 50	YMIN	F10.5	The minimum for the Y-axis in data units
Cols 51 - 60	YMAX	F10.5	The maximum for the Y-axis in data units
Cols 61 - 70	DELTAY	F10.5	Specify the scale difference between numbers to be printed on the Y-axis.
Cols 71 - 80	YINCRM	F10.5	Specify the distance between numbers on the Y-axis.

CARD 2

Cols 1 - 10	JAVA	I10	This field contains the overall plot scale indicator. If JAVA=0, plotter units are inches. If JAVA=10, plotter units are centimeters.
Cols 11 - 20	NVARBL	I10	Specify the number of records to be plotted. A record consists of a series of X, Y pairs. There is no limit to the number of records that can be plotted.
Cols 21 - 30	HEIGHT	F10.4	Choose the height of letters and numbers to be printed on the plot.

CARD 3

Cols 1 - 40	LABELX	10A4	Forty columns are allocated for a label to appear below the X-axis.

Cols 41 - 80	LABELY	10A4	Forty columns are also allocated for a label to appear along the Y-axis (see Figure 10).

CARD 4

Cols 1 - 40	LABELZ	10A4	Forty columns are also allocated for a general label that appears above the plot area (see Figure 10).
Cols 41 - 50	NDECX	I10	This integer variable specifies the number of decimal digits to be printed when numbering the X-axis.
Cols 51 - 60	NDECY	I10	This variable specifies the number of decimal digits to be printed when numbering the Y-axis.

CARD 5

Cols 1 - 10	KINPUT	I10	Specify the input unit reference number.
Cols 11 - 20	LREAD	I10	Specify the number of pairs of points to be read. A pair consists of and X and Y point. (LREAD $\leq$ 6000)
Cols 21 - 30	LPLOT	I10	The number of points to be plotted (LPLOT $\leq$ LREAD)
Cols 31 - 40	IOPTN	I10	Choose the method of plotting this series. If IOPTN=-1, spot data points. If IOPTN=0, connect data points. If IOPTN=1, connect and spot data points.

Col 47	ISYMBL	A1	A single character is specified for spotting data (when IOPTN≠0). Choose a character with a distinct center.
Col 51 - 78	FMT	7A4	Specify the format of the input series. Begin with a left parenthesis and use real FORTRAN format codes to describe the reading of X, Y data.

DATA ACQUISITION

Data points are read using the following statement.

```
READ(KINPUT,FMT)(X(I),Y(I),I=LREAD)
```

Printed Output

The printed output lists all input parameters as seen in Figure 12.

INPUT PARAMETERS

XMIN	XMAX	DELTAX	XINCRM	YMIN	YMAX	DELTAY	YINCRM
10.000	20.000	1.000	2.000	32.000	36.000	0.500	2.000

LABELX	LABELY
TEMPERATURE	SALINITY (PPT)

LABELZ	NDECX	NDECY	JAVA	HEIGHT
CRUISE TR 155	1	1	10	0.300

SERIES #	KINPUT	LREAD	LPLOT	IOPTN	ISYMBL	FORMAT :
1	5	16	16	0		(8F10.2)
2	5	8	8	-1	X	(8F10.3)
3	5	10	10	1	O	(8F10.3)

Figure 12. Sample LINLIN printed output.

LINLOG

The LINLOG program plots one variable on a linear axis versus frequency and period on a logarithmic axis. Sample LINLOG plots are shown in Figure 14. Variables such as spectra, phase and coherence may be read from cards, tape or disk and displayed. User supplied variables control the numbering and labelling of the axes as seen in Figure 13. Plot size and character height are also determined by the user.

Only series of data computed at constant frequency increments can be plotted. The first non-zero frequency and the frequency increment are used to reconstruct the frequency associated with each data point. The first data point is assumed to be a residual, or zero estimate and is, therefore always spotted at X=4.0. The range of the X-axis is determined by the range of the frequencies.

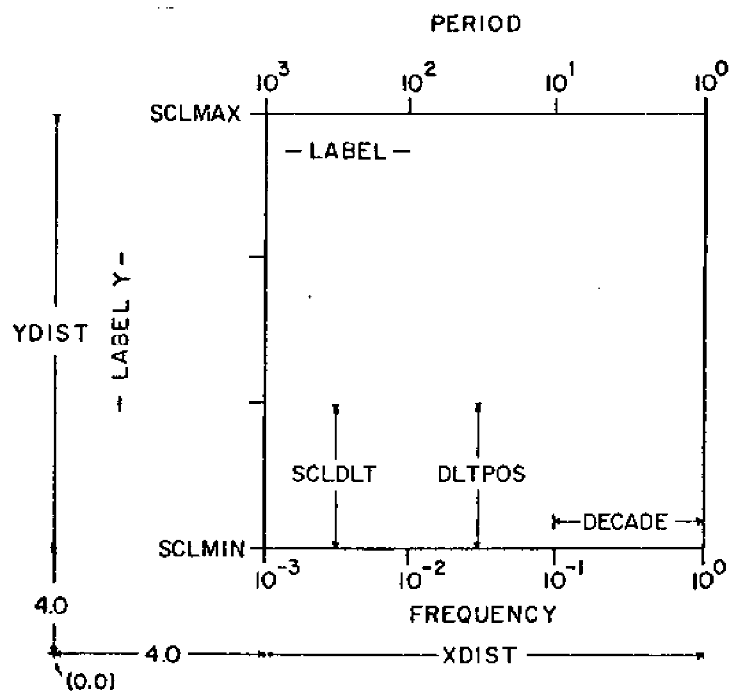
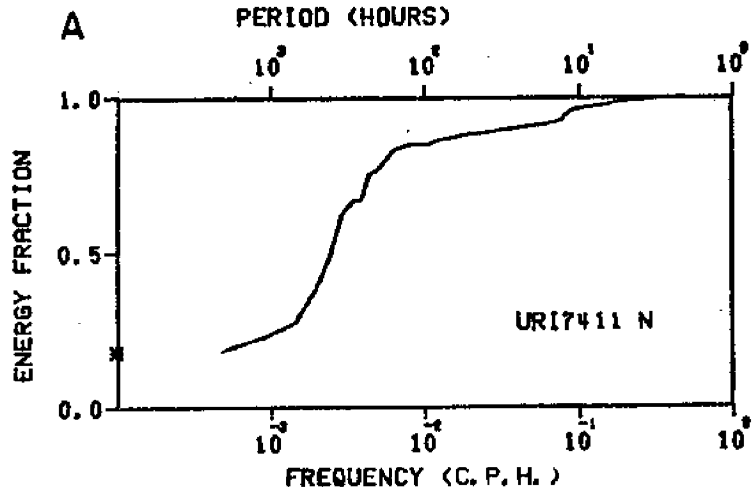
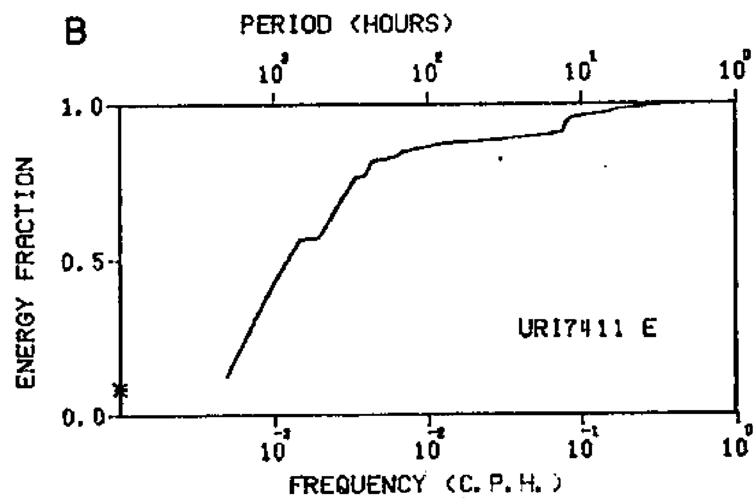


Figure 13. General arrangement of LINLOG. XDIST and YDIST are computed distances.

A



B



C

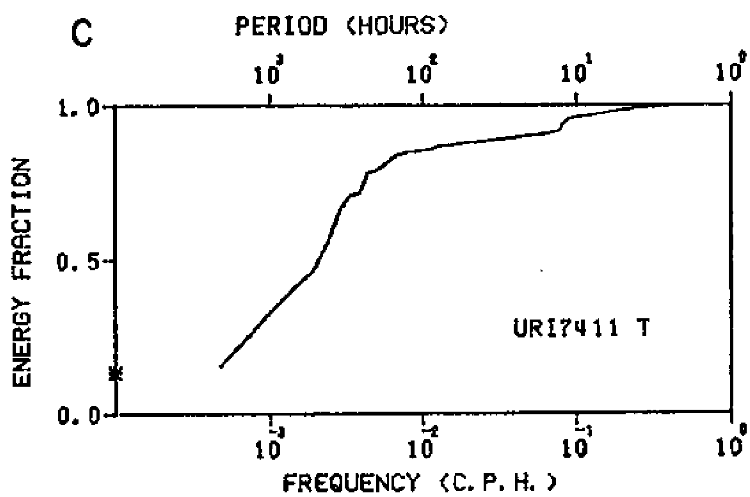


Figure 14. Sample LINLOG plots (Weisberg, personal communication).

LINLOG CONTROL CARDS

```
//LINLOG JOB (XXX100,128,1,1,500),'USER'
/*ROUTE  PUNCH REMOTE2
//  EXEC PGMRUN,LIB=WATER,NAME=LINLOG,LIB1=TPLOT,LIB2=TPLOT,
//  REGION.GO=128K
//GO.FT01F001 ...       }
                        }   OPTIONAL (See KINPUT)
//  LABEL(1,SL) ...     }
//GO.SYSIN DD *
   DATA CARD 1
   DATA CARD 2
   DATA CARD 3
   DATA CARD 4
   DATA ACQUISITION
//
```

LINLOG DATA CARDS

<u>CARD 1</u>

Cols 1 - 10	SCLMIN	F10.3	Specify the minimum value for the linear Y-axis.
Cols 11 - 20	SCLMAX	F10.3	Specify the maximum value for the Y-axis.
Cols 21 - 30	SCLDLT	F10.3	Specify the scale difference between numbers to be printed on the Y-axis.
Cols 31 - 40	DLTPOS	F10.3	Specify the distance between numbers on the Y-axis.

Cols 41 - 50	NDECY	I10	Choose the number of decimal digits to be printed when numbering the Y-axis.
Cols 51 - 60	JAVA	I10	The overall plot scale indicator If JAVA=0, plotter units are inches. If JAVA=10, plotter units are centimeters.
Cols 61 - 70	DECADE	F10.3	Specify the distance between powers of ten on the X-axis.

CARD 2

Cols 1 - 10	FONE	F10.5	Punch the first non-zero frequency to be associated with the first data point.
Cols 11 - 20	DELTAF	F10.5	Punch the increment in frequency between adjacent data points. FONE and DELTAF are used to reconstruct the frequency associated with each data point.
Cols 21 - 30	LREAD	I10	The number of data points to be read (LREAD $\leq$ 4000)
Cols 31 - 40	LPLOT	I10	The number of data points to be plotted (LPLOT $\leq$ LREAD)
Cols 41 - 50	KINPUT	I10	Punch the input unit reference number. Use KINPUT=5 for card input.
Cols 51 - 60	IOPTN	I10	Choose the method of plotting. If IOPTN=-1, spot data points. If IOPTN=0, connect data points. If IOPTN=1, connect and spot data points.
Col 67	ISYMBL	A1	Punch the character to be used for spotting data points. (IOPTN$\neq$0)

Cols 71 - 80	HEIGHT	F10.3	Specify the height of numbers and characters printed on this plot. Adjust HEIGHT for readability after any necessary reductions.

CARD 3

Cols 1 - 40	LABEL	10A4	Forty columns are used for labelling the plot. The location of this label is within the plot border as determined by IQUAD. Pack LABEL right when IQUAD=1 or 2. Pack LABEL left when IQUAD=3 or 4.
Cols 41 - 50	IQUAD	I10	Indicate in which quadrant the label should appear. Move the label to avoid the data points. The possible quadrants are numbered as seen here:

IV	I
III	II

CARD 4

Cols 1 - 40	LABELY	10A4	Forty columns are allocated for identifying the Y-axis. Since different types of variables may be plotted, the units should be included in the label.

Cols 41 - 80	FMT	10A4	Forty columns are allocated for the input format specification. Begin with a left parenthesis in column 41 and use real FORTRAN format codes to describe the arrangement of data. Note that only the dependent variable is read and that frequency is reconstructed within the program.

DATA ACQUISITION

The following FORTRAN READ statement is used to access data:

```
READ(KINPUT,FMT)(DATA(I)I=1,LREAD)
```

LINLOG PRINTED OUTPUT

The LINLOG printed output contains a listing of all input parameters. This list aids the user in diagnosing a faulty execution (see Figure 15).

INPUT PARAMETERS

SCLMIN	SCLMAX	SCLDLT	DLTFOS	NDECY	JAVA	DECADE
0.0	1.000	0.100	1.000	1	10	5.000

FONE	DELTAF	LREAD	LPLOT	KINPUT	IOPTN	ISYMBL	HEIGHT
C.01000	0.01000	5	5	5	0		0.300

LABEL	IQUAD
RECORD GS0122	1

LABELY	FORMAT ;
COHERENCE	(8F10.3)

Figure 15. Sample LINLOG printed output.

LOGLOG

LOGLOG plots a single variable on a logarithmic (base 10) axis versus frequency and period on a logarithmic (base 10) axis. A sample LOGLOG plot is shown in Figure 16.

The general arrangement of LOGLOG is shown in Figure 17. The frequency axis, x, is drawn to accommodate points from the lowest frequency up to a frequency of one cycle per sampling interval in full decades. The frequency along the bottom and period along the top of the X-axis are printed in powers-of-ten notation. Similarly the Y-axis is drawn to accommodate the maximum and minimum data points. The distance between decades is supplied by the user. This distance is consistent for the X and Y axis. By controlling this decade distance, different variables may be compared without overplotting.

Each variable is assumed to have a zero estimate at the first point. This point will be starred at X=3.0. Data estimated at constant increments in frequency are read in any real format as given by the user. Data points are connected with a solid line. A pair of markers are positioned on the plot to indicate the 12.41-hour tidal period and the inertial period as computed from 12.0/SIN (XLATDM) where XLATDM is the latitude of the mooring expressed in degrees. Similarly, LOGLOG may be applied to any of the results of time series analysis.

The dependent variable may be read from cards, tape or disk by specifying an input unit reference number.

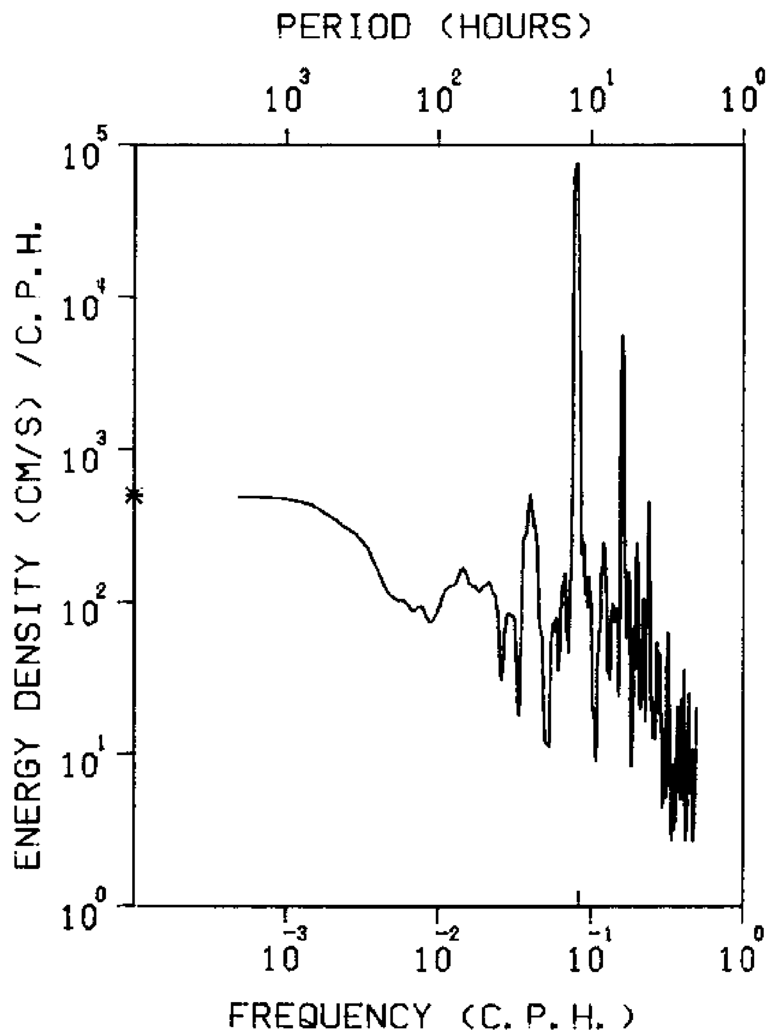


Figure 16. Sample LOGLOG plot (Weisberg, personal communication).

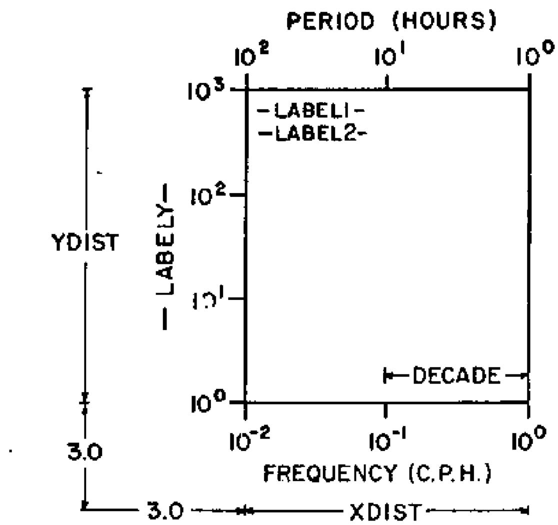


Figure 17. General arrangement of LOGLOG. XDIST and YDIST are computed distances.

LOGLOG Control Cards

```
//LOGLOG JOB (XXX100,128,1,1,500),'USER'
/*ROUTE  PUNCH REMOTE2
//  EXEC PGMRUN,LIB=WATER,NAME=LOGLOG,LIB1=TPLOT,LIB2=TPLOT,
//  REGION.GO=128K
//GO.FT01F001 ...     \
                       } OPTIONAL
//  LABEL=(1,SL) ...  /
//GO.SYSIN DD *
   DATA CARD 1
   DATA CARD 2
   DATA CARD 3
   DATA ACQUISITION
//
```

LOGLOG DATA CARDS

CARD 1

Cols 1 - 10	FONE	F10.7	This field contains the first non-zero frequency. FONE is used to determine the range of the frequency axis in cycles per hour.
Cols 11 - 20	FINCRM	F10.7	Punch the constant increment in frequency between adjacent points of the data. FONE and FINCRM are used to recreate the frequency associated with each spectral point prior to scaling and plotting.
Cols 21 - 30	JAVA	I10	If JAVA=0, plotter units are in inches. JAVA=10, plotter units are in centimeters.
Cols 31 - 40	LPLOT	I10	The number of points to be read and plotted. (LPLOT $\leq$ 4000)
Cols 41 - 50	KINPUT	I10	Specify the input unit reference number. For card input, KINPUT=5.
Cols 51 - 60	HEIGHT	F10.5	Choose the height of the letters and numbers printed on the plot. Adjust for readability after any reduction.
Cols 61 - 70	XLATDM	F10.5	Specify the latitude of the station where data were collected. Express latitude in degrees. For $30^{\circ}30'$, XLATDM=30.5. XLATDM is used to mark the inertial frequency on the plot. That frequency is computed from SIN (XLATDM)/12.0.

Cols 71 - 80	DECADE	F10.5	Specify the distance between each decade on the X and Y-axis.

CARD 2

Cols 1 - 40	LABEL1	10A4	Eighty columns of labels are available for identifying the plot. These labels appear within the border of the plot as seen in Figure 17.
Cols 41 - 80	LABEL2	10A4	

CARD 3

Cols 1 - 40	LABELY	10A4	This label is printed vertically against the Y-axis to identify the units of the data.
Cols 41 - 80	FMTSPT	10A4	Forty columns are allocated for the FORMAT description of the data to be read. Begin with a left parenthesis in column 41, and use real FORTRAN format codes to describe the arrangement of data points. End the FMTSPT description with a right parenthesis.

DATA ACQUISITION

The following read statement accesses data points:

```
READ(KINPUT),FMTSPT)(SP(I),I=1,LPLOT)
```

Printed Output

The printed output contains a listing of the input parameters as shown in Figure 18.

```
              INPUT PATAMETERS

   FONE      FINCRM       JAVA      LPLOT     KINPUT      HEIGHT     XLATOM     DECADE

 0.00049    0.00049         10       1025          2       0.200     45.500      3.000

                LABEL1                                      LABEL2

                GATE RECORD 3                               NORTH COMPONENT

                LABELY                   FORMAT :

              ENERGY DENSITY             (A4,4X)


COMPUTED DISTANCES :               XDIST      YDIST

                                   12.00      18.00
```

Figure 18. Sample LOGLOG printed output.

LOGLIN

LOGLIN plots two variables on a logarithmic (base 10) axis versus frequency and period on a linear axis as shown in Figure 20.

The size of each logarithmic decade and frequency increment are user supplied variables as seen in Figure 19. In this way, a number of plots of the same size can be drawn. A constant factor may be applied to either series to insure compatability with other plots. Three 40 column labels are used to annotate the plot.

The general organization of the data requires a zero estimate and subseqeunt points at constant frequecny increments. The zero estimate is spotted at X=4.0. Following points are plotted at the appropriate frequency. The data are scanned for negative values which are set equal to the zero estimate for plotting purposes. The original data remain untouched. The Y-axis is drawn to accommodate the maximum and minimum data points.

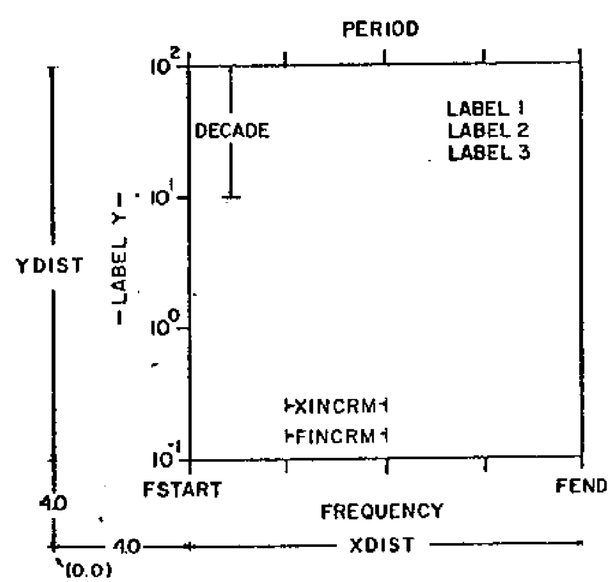


Figure 19. General arrangement of LOGLIN XDIST and YDIST are computed distances.

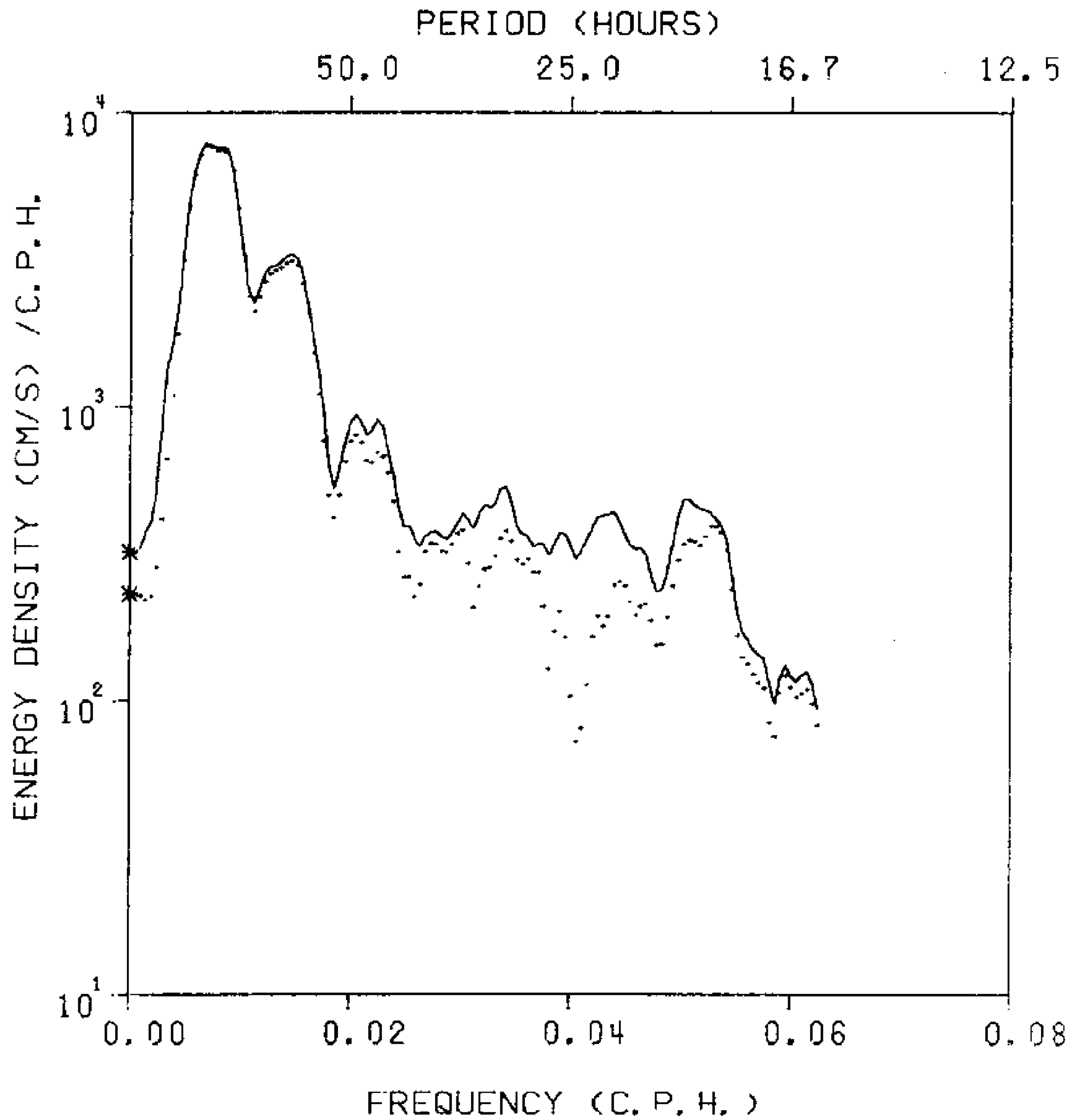


Figure 20. Sample LOGLIN plot (from Weisberg, 1975).

LOGLIN CONTROL CARDS

```
//LOGLIN JOB (XXX100,128,1,1,500),'USER'
/*ROUTE  PUNCH REMOTE2
//  EXEC PGMRUN,LIB=WATER,NAME=LOGLIN,LIB1=TPLOT,LIB2=TPLOT,
//  REGION.GO=128K
//GO.FT01F001 ...        }  OPTIONAL
//  LABEL=(1,SL) ...     }
//GO.SYSIN DD *
   DATA CARD 1
   DATA CARD 2
   DATA CARD 3
   DATA CARD 4
   DATA CARD 5
   DATA ACQUISITION series 1
   DATA CARD 6
   DATA CARD 7
   DATA ACQUISITION series 2
//
```

LOGLIN CONTROL CARDS

<u>CARD 1</u>

Cols 1 - 10	JAVA	I10	The overall plot scale indicator:
			If JAVA=0, plotter units are inches.
			If JAVA=10, plotter units are centimeters.

Cols 11 - 20	FSTART	F10.7	Specify the first frequency for the linear X-axis.
Cols 21 - 30	FEND	F10.7	Specify the last frequency for the linear X-axis.
Cols 31 - 40	FINCRM	F10.7	Punch the increment in frequency between numbers on the X-axis.
Cols 41 - 50	XINCRM	F10.7	Choose the distance between adjacent numbered frequencies on the X-axis.
Cols 51 - 60	H	F10.5	Choose the height of the numbers and letters printed on the plot for readability after any necessary reduction.
Cols 61 - 70	DECADE	F10.5	Specify the distance between adjacent powers of ten on the Y-axis.
Cols 71 - 80	NDECF	K10	Choose the number of decimal digits to be printed in each frequency. (NDECF $\geq$ 0)

CARD 2

Cols 1 - 40	LABEL1	10A4	These two labels are used to identify the
Cols 41 - 80	LABEL2	10A4	records being plotted. Their location is shown in Figure 19. They may contain alphabetic, numeric or special characters which should be packed right. These columns may also be left blank.

CARD 3

Cols 1 - 40	LABEL3	10A4	This label appears with LABEL1 and LABEL2.
Cols 41 - 80	LABELY	10A4	Forty columns are also allocated for identifying the Y-axis. Alphabetic, numeric or special characters may be used. The relative location of LABELY is shown in Figure 19.

CARD 4

Cols 1 - 10	KINPUT	I10	Indicate the input unit reference number for the first series.
Cols 11 - 20	LREAD	I10	The number of points to be read for series one (LREAD $\leq$ 4000)
Cols 21 - 30	LPLOT1	I10	The number of points to be plotted for series one (LPLOT1 $\leq$ LREAD)
Cols 31 - 40	FACTOR	F10.5	Specify a factor to be applied to series one. If no modification is needed, specify FACTOR as 1.0. FACTOR is applied as follows: SP(I) = SP(I) * FACTOR
Cols 41 - 50	IOPTN1	I10	Pick the method of plotting series one. If IOPTN1 = -1, spot data points. If IOPTN1 = 0, connect data points. If IOPTN1 = 1, connect and spot data points.
Col 57	ISYMBL	A1	Specify the symbol to be used in spotting data points of series one. (IOPTN1 $\neq$ 0)

CARD 5

Cols 1 - 40	FMTSP1	10A4	Specify the arrangement of data for series one. Begin with a left parenthesis in column 1 and use real FORTRAN format codes. End FMTSP1 with a right parenthesis.
Cols 41 - 50	FONE	F10.7	Specify the first non-zero frequency for the series being plotted.
Cols 51 - 60	DELTAF	F10.7	Punch the constant frequency increment between adjacent data points.

DATA ACQUISITION #1

Data points of series one are accessed from tape, disk or cards using the following READ statement.

READ(KINPUT,FMTSP1)(SP(I),I=1,LREAD)

CARD 6

Cols 1 - 10	KINPUT	I10	Specify the input unit reference number for series two.
Cols 11 - 20	LREAD	I10	Indicate the number of points to be read for series two (LREAD $\leq$ 4000).
Cols 21 - 30	LPLOT2	I10	The number of points to be plotted for series two (LPLOT2 $\leq$ LREAD).

Cols 31 - 40	FACTOR	F10.5	Punch the factor to be applied to series two. Specify FACTOR = 1.0 if no modification is necessary. SP2(I) = SP2(I) * FACTOR
Cols 41 - 50	IOPTN2	I10	Pick the method of plotting series two. If IOPTN2 = -1, spot data points If IOPTN2 = 0, connect data points If IOPTN2 = 1, connect and spot data points
Col 57	ISYMBL(2)	A1	Choose the symbol to be used when spotting data points of series two (IOPTN2 ≠ 0).

CARD 7

Cols 1 - 40	FMTSP2	10A4	Specify the arrangement of data for series two. Begin with a left parenthesis in column 1 and use real FORTRAN format specifications. End FMTSP2 with a right parenthesis.

DATA ACQUISITION #2

Data points of series two are accessed from tape, disk or card using the following READ statement.

READ(KINPUT,FMTSP2)(SP2(I),I=1,LREAD)

Printed Output

The printed output contains a list of the input parameters as seen in Figure 21. Also given are the computed distances XDIST and YDIST.

```
                INPUT PARAMETERS

FSTART      FEND     FINCRM     XINCRM        H     DECADE     NDECF

0.0       0.5000     0.1000     2.0000     0.300     3.000         1

KINPUT     LREAD     LPLOT1     FACTOR    IOPTN1  ISYMBL(1)   FORMAT:

     5        91         91      1.000         0              (20X,F15.7)

KINPUT     LREAD     LPLOT2     FACTOR    IOPTN2  ISYMBL(2)   FORMAT:

     5        91         91      1.000         1     +        (20X,F15.7)

       FONE =     0.00556    DELTAF =    0.00556

      XDIST =       10.00     YDIST =      12.00

        LOGLIN PLCT COMPLETE
```

Figure 21. Sample LOGLIN printed output.

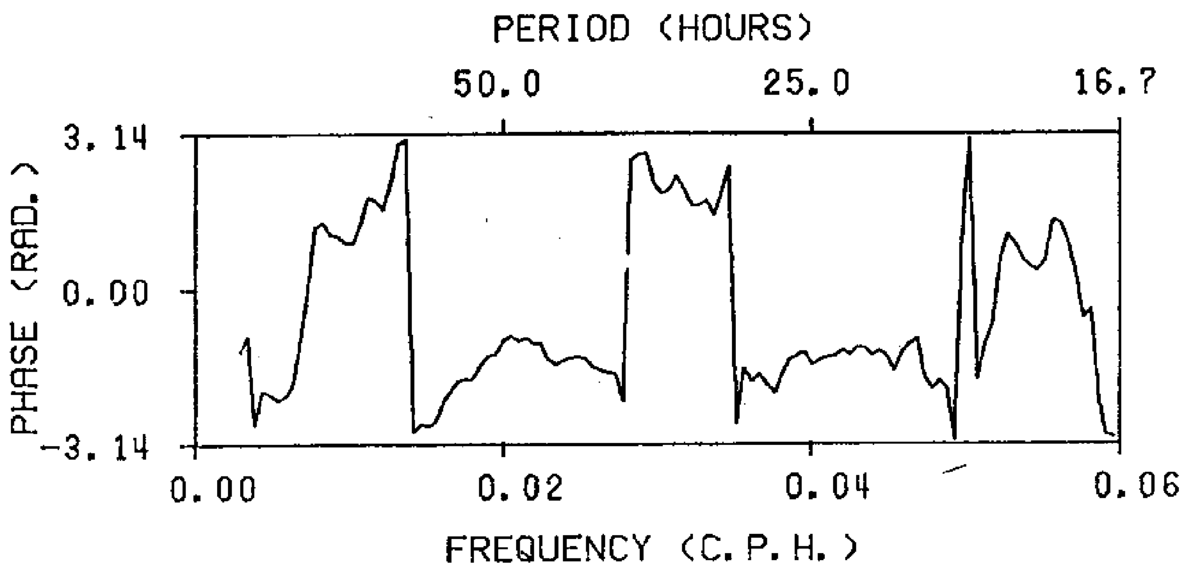
SPECIALL

SPECIALL plots a single variable on linear Cartesian coordinates where the X-axis is frequency and period. Sample SPECIALL plots are shown in Figure 22. Numbering and labelling of the X-and Y-axis results from user supplied information. Plot size and character height are also determined by the user.

Only series of data computed at constant frequency increments can be plotted by SPECIALL. An initial frequency and frequency increment are used to reconstruct the appropriate frequencies for plotting purposes. If all points are plotted the first point is starred at X=4.0. If points are to be skipped, plotting begins at the designated frequency.

SPECIALL CONTROL CARDS

```
//SPECIALL JOB (XXX100,128,1,1,500),'USER'
/*ROUTE  PUNCH REMOTE2
//  EXEC PGMRUN,LIB=WATER,NAME=SPECIALL,LIB1=TPLOT,LIB2=TPLOT,
//  REGION.GO=128K
//GO.FT01F001 ...          }
                           }   OPTIONAL
//  LABEL=(1,SL,,IN) ...   }
//GO.SYSIN DD *
   DATA CARD 1
   DATA CARD 2
   DATA CARD 3
   DATA CARD 4
   DATA CARD 5
   DATA ACQUISITION
//
```



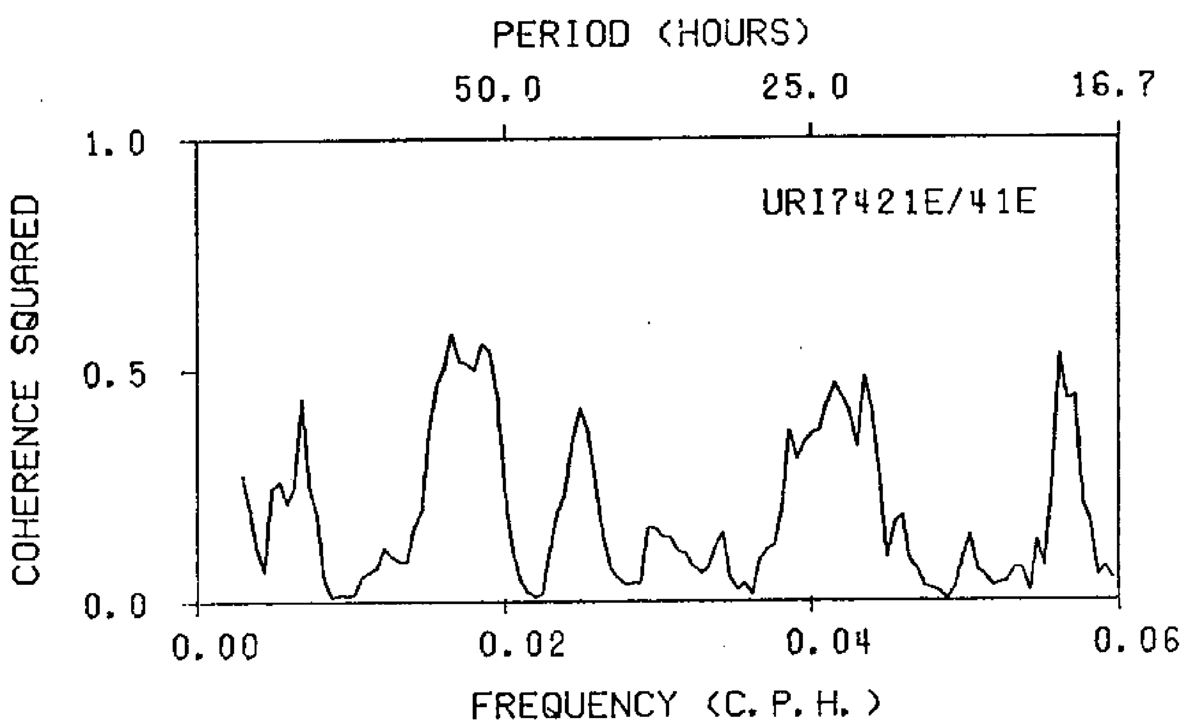


Figure 22. SPECIALL plot (Weisberg, personal communication).

SPECIALL DATA CARDS

CARD 1

Cols 1 - 10	JAVA	I10	Choose the plotting units. If JAVA=0, units are inches. If JAVA=10, units are centimeters.
Cols 11 - 20	KINPUT	I10	Specify the input unit reference number.
Cols 21 - 30	LREAD	I10	Punch the number of points to be read. (LREAD $\leq$ 4000)
Cols 31 - 40	LPLOT	I10	Punch the number of points to be plotted. (LPLOT $\leq$ LREAD)
Cols 41 - 50	IOPTN	I10	Choose the plotting option. If IOPTN=-1, spot data points. If IOPTN=0, connect data points. If IOPTN=1, connect and spot data points. Spot symbol is a plus sign, '+'.
Cols 51 - 78	FMT	7A4	Specify the format of the input series. Begin with a left parenthesis in column 51 and use real FORTRAN format codes. End FMT with a right parenthesis.

CARD 2

Cols 1 - 40	LABELX	10A4	Punch the label which identifies the X-axis.
Cols 41 - 80	LABELY	10A4	Punch the label which appears along the Y-axis.

CARD 3

Cols 1 - 40	LABELZ	10A4	Punch the label which appears within the plot area.

Cols 41 - 50 IQUAD I10 Specify the quadrant in which the label should appear. Quadrant order is shown below.

IV	I
III	II

For IQUAD=1 or 2, pack the label right. For IQUAD=3 or 4, pack the label left.

Cols 51 - 60 NDECX I10 Specify the number of decimal digits when numbering the X-axis.

Cols 61 - 70 NDECY I10 Specify the number of decimal digits when numbering the Y-axis.

<u>CARD 4</u>

Cols 1 - 10 FMIN F10.7 Specify the minimum frequency for the X-axis.

Cols 11 - 20 FMAX F10.7 Specify the maximum frequency for the X-axis.

Cols 21 - 30 SCALEF F10.7 Specify the numbering increment for the X-axis.

Cols 31 - 40 DELTAX F10.7 Specify the distance between numbers on the X-axis.

Cols 41 - 50 YMIN F10.7 The minimum of the Y-axis.

Cols 51 - 60 YMAX F10.7 The maximum of the Y-axis.

Cols 61 - 70 DELTAS F10.7 The numbering increment for the Y-axis.

Cols 71 - 80 DELTAY F10.7 The distance between numbers on the Y-axis.

CARD 5

Cols 1 - 10	FONE	F10.7	Punch the frequency of the first point to be plotted. FONE is non-zero.
Cols 11 - 20	DELTAF	F10.7	Punch the increment in frequency between adjacent data points.
Cols 21 - 30	HEIGHT	F10.3	Choose the of letters and numbers on the plot. Adjust for readability after any reduction.
Cols 31 - 40	KSTART	I10	Specify the number of the point at which plotting should start. If KSTART=1, all points will be plotted.

DATA ACQUISITION

The following READ statement is used to access data points.

```
READ (KINPUT,FMT)(SP(I),I=1,LREAD)
```

Printed Output

The printed output contains a listing of the input parameters and the computed length of the X and Y axes.(Figure 23).

INPUT PARAMETERS

JAVA	KINPUT	LREAD	LPLOT	IOPTN	FMT
10	1	123	123	0	(A4,12X)

LABELX	LABELY
FREQUENCY (C.P.H.)	COHERENCE SQUARED

LABELZ	IQUAD	NDECX	NDECY
URI7421E/41E	1	2	1

FMIN	FMAX	SCALEF	DELTAX	YMIN	YMAX	DELTAS	DELTAY
0.0	0.06000	0.02000	4.00000	0.0	1.00000	0.50000	3.00000

FONE	DELTAF	HEIGHT	KSTART
0.00293	0.00049	0.300	7

SPECIALL COMPLETE

Figure 23. Sample SPECIALL printed output.

ACKNOWLEDGEMENTS

Most of enclosed programs were originally written for usage on a University of Rhode Island Sea Grant: Project Number R/ES-5, "Net Circulation in the Providence River of Narragansett Bay".

The authors received partial support from the National Science Foundation: Grant Number OCD74 - 01739 401.

References

Greenall, R.K., Jr., 1974. Delta Incremental Plotting System FORTRAN Subroutines, Advanced Graphics Manual. University of Rhode Island Computer Laboratory publication. Second Edition, June, 1974, 94 pp.

Kramer, W.P. 1974. FORTRAN Programs for Averaging, Filtering, Plotting and Spectral Analysis of Current Meter Observations on the IBM 370/155. URI Technical Report #74-1, 95 pp.

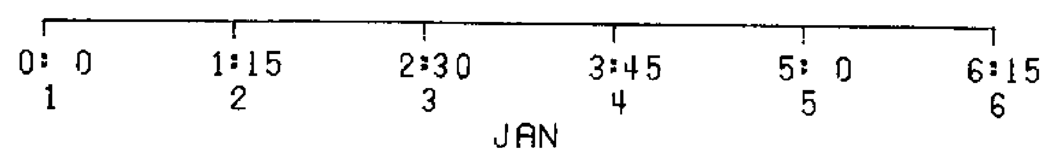
Lambert, R.B., Jr., 1974. Small-scale Dissolved Oxygen variations and the Dynamics of Gulf Stream Eddies. Deep-Sea Research, 21, 529-546.

Weisberg, R.H., 1975. The Non-Tidal Flow in the Providence River of Narragansett Bay: A Stochastic Approach to Estuarine Circulation. PH.D. Thesis, University of Rhode Island, 127 pp.

APPENDIX A

Subroutine TIMEPT

The FORTRAN subroutine TIMEPT prepares a time axis in the style of the Gregorian calendar. By supplying the proper argument list to the subroutine, dates and times will appear on a plot at specified positions and at the chosen size. TIMEPT makes a decision as to the format of the date and time information to be printed. The time increment is specified in units of days, hours and minutes. If the hour or minute time increments are non-zero, the full display of date and time appears as seen below:



When the hour and minute increment are zero, daily increments or greater are to be printed. The following form is then chosen:

Proper use of subroutine TIMEPT requires the following statement within a FORTRAN program:

```
CALL TIMEPT (XX,NOWDAY,ITIS,NOWHR,NOWMIN,IDADD,IHRADD,MINADD,XINCRM,
          IDO,MONTHS,MONEND,Y,HEIGHT)
```

Each variable is described below:

XX	=	The horizontal position of the first time to be printed
NOWDAY ITIS NOWHR NOWMIN	=	The specification of the first date and time to be printed. The order is the day of the month, the number of the month, the hour and minute.
IDADD IHRADD MINADD	=	The time increment between printed times. Specify the time increment in units of days, hours and minutes.
XINCRM	=	The horizontal distance between printed times.
IDO	=	The number of times to be printed.
MONTHS	=	An alphameric array specifying the four characters representing each month. (MONTHS(1)=JAN.)
MONEND	=	An integer array containing the number of days in each month (MONEND(1)=31).
Y	=	The vertical position of the top of the time axis.
HEIGHT	=	The height of individual characters and letters used in the time axis.

The units of plotter distances and character height are determined by previous calls to subroutine PLOT earlier in the main program.

A listing of subroutine TIMEPT appears in Appendix K.

APPENDIX B

COMPUTATION OF XPOSTN

In programs which plot time series, the horizontal offset, XPOSTN, is used to align data points with the time axis. Time is reconstructed within a program from a start position and a plotting point density. This start position is dependent on the start time of the data and the arrangement of the time axis. That value, XPOSTN, may be calculated as follows:

XPOSTN=X+((TDATA-TZERO)*XINCRM)/DELTAT

where:

X = the distance from the origin of the plot to the first printed time.

TDATA = the start time of a time series.

TZERO = the first time printed on the time axis.

XINCRM = the distance between printed times.

DELTAT = the difference in time between printed times.

A sample calculation XPOSTN may make its use clear.

Let:

X = 6.0"

TZERO = 1 March 0:00

XINCRM = 2.0

DELTAT = 24 hours

If the time series begins on 1 March at 12:00, where do we place the first point?

XPOSTN=6.0inches+((12 hours*2.0 inches)/24 hours)

XPOSTN is therefore 7.0 inches. The first data point is positioned at 7.0. Following points are positioned according to the plotting point density chosen.

APPENDIX C

PROGRAM MULTYPLT

```
C
C      PROGRAM MULTYPLT
C
       DIMENSION DATA(6000),MONTHS(12),MONEND(12),FMT(10),LABEL1(5)
       DIMENSION LABEL2(5),X(6000),ISYMBL(1)
       INTEGER PNTDEN
       DATA MONTHS/' JAN',' FEB',' MAR',' APR',' MAY','JUNE','JULY',
      1' AUG','SEPT',' OCT',' NOV',' DEC'/
       DATA MONEND/31,28,31,30,31,30,31,31,30,31,30,31/
C
C      READ THE NUMBER OF VARIABLES TO BE PLOTTED
C      ALSO THE OVERALL PLOT SCALE INDICATOR  'JAVA'
C
       READ(5,10)NVARBL,JAVA,HEIGHT,LEAPYR
    10 FORMAT(2I10,F10.5,I10)
       WRITE(6,11)NVARBL,JAVA,HEIGHT,LEAPYR
    11 FORMAT(//T15,'INPUT PARAMETERS'//T5,'NVARBL',T17,'JAVA',T25,
      1'HEIGHT',T35,'LEAPYR'//2I10,F10.3,I10)
    20 FORMAT(2F10.5,4I10,T67,A4,I10)
    30 FORMAT(20A4)
       CALL PLOT(0.0,0.0,JAVA)
       LONG=00
       Y=3.0
       WRITE(6,12)
    12 FORMAT(//T3,'SERIES #',T13,'XPOSTN',T21,'SCALE',T28,'PNTDEN',T36,
      1'LENGTH',T44,'KINPUT',T53,'IOPTN',T59,'ISYMBL',T67,'NDECY',T74,
      2'FORMAT :')
C
C      START THESE PLOTS AT X=6.0
C
       DO 5000 M=1,NVARBL
C
C      READ INFORMATION SPECIFIC TO EACH VARIABLE
C
       READ(5,20)XPOSTN,SCALE,PNTDEN,LENGTH,KINPUT,IOPTN,ISYMBL(1),NDECY
       READ(5,30)LABEL1,LABEL2,FMT
       WRITE(6,21)M,XPOSTN,SCALE,PNTDEN,LENGTH,KINPUT,IOPTN,ISYMBL(1),
      1            NDECY,FMT
    21 FORMAT(//I10,F8.4,F7.2,4I8,T61,A4,1X,I4,3X,10A4)
       X(1)=XPOSTN
C
C      READ THE DATA FROM THE PROPER INPUT DEVICE AND IN THE RIGHT FORMAT
C
       READ(KINPUT,FMT)(DATA(I),I=1,LENGTH)
C
C      KEEP A CHECK ON THE LARGEST NUMBER OF POINTS INVOLVED
C
       IF(LENGTH-LONG)43,43,35
```

```
   35 LONG=LENGTH
      JUMP=PNTDEN
      SAVE=XPOSTN
   43 FACT=1.0/PNTDEN
      DO 47 L=2,6000
   47 X(L)=X(L-1)+FACT
C
C     EVALUATE THE MAX AND MIN OF EACH SERIES
C     DETERMINE PLOT SIZE FROM MAX,MIN AND ISCALE PARAMETER
C
   49 XMAX=-99999.
      XMIN=9999999.
      DO 50 I=1,LENGTH
      XMIN=AMIN1(XMIN,DATA(I))
   50 XMAX=AMAX1(XMAX,DATA(I))
      IF(XMIN)59,51,51
   51 MINPLT=XMIN/SCALE
      MAXPLT=XMAX/SCALE+1
      ALLPLT=MAXPLT-MINPLT
      LIMIT=MAXPLT-MINPLT+1
      KKK=ALLPLT/2
      BELOW=KKK
      ABOVE=ALLPLT-BELOW
      DO 58 KRAMER=1,LENGTH
   58 DATA(KRAMER)=DATA(KRAMER)/SCALE-MINPLT+Y
      GO TO 65
   59 MAXPLT=XMAX/SCALE+1
      MINPLT=ABS(XMIN)/SCALE+1
      ABOVE=MAXPLT
      BELOW=MINPLT
      ALLPLT=ABOVE+BELOW
      LIMIT=ALLPLT+1
C
C
C     CONVERT DATA TO PLOTTER UNITS
C
      DO 200 I=1,LENGTH
  200 DATA(I)=(DATA(I)/SCALE)+Y+BELOW
C
C     START DRAWING THE AXIS
   65 CALL PLOT(6.0,Y,3)
      CAL' PLOT(6,0,Y+ALLPLT,2)
C
C     NUMBER THE 'Y' AXIS
C
      XV=MAXPLT*SCALE
      YDIGIT=ALOG10(XV)
      XX=6,0-((2,0*HEIGHT)/3,0)
```

```
      XNUMB=XX-((1.0+NDFCY)*HEIGHT)
      XLABEL=XNUMB-((3.0+YDIGIT)*HEIGHT)
      YPOSTN=Y+ALLPLT
      DO 100 I=1,LIMIT
      CALL NUMPLT(XNUMB,YPOSTN-(HEIGHT/2.0),0.0,HEIGHT,XV,NDECY)
      CALL CHAR(XX,YPOSTN,0.0,HEIGHT,'_',1)
      YPOSTN=YPOSTN-1.0
  100 XV=XV-SCALE
C
C     LABEL THE AXIS
C
      CENTRY=Y+(ALLPLT/2.0)-(10.0*HEIGHT)
      CALL CHAR(XLABEL,CENTRY,90.0,HEIGHT,LABEL2(1),20)
      CALL CHAR(XLABEL-2.0*HEIGHT,CENTRY,90.0,HEIGHT,LABEL1(1),20)
  221 CALL VECTOR(X,DATA,LENGTH,1,IOPTN,ISYMBL(1))
C
C     ADJUST THE Y VALUE WHEN DONE PLOTTING THIS VARIABLE
C
      IF(XMIN)250,251,251
  250 CALL PLOT(X(LENGTH),Y+BELOW,3)
      CALL PLOT(6.00,Y+BELOW,2)
  251 Y=Y+ALLPLT+1.0
      IF(KINPUT-5)4999,5000,4999
 4999 REWIND KINPUT
 5000 CONTINUE
      IF(Y-25.0)300,298,298
  298 WRITE(6,299)
  299 FORMAT(///10X,' BETTER USE LARGE PAPER SINCE OVERALL PLOT EXCEEDS
     125 UNITS'//)
C
C     READ IN THE TIME INFORMATION AND DISPLAY IT ON THE PLOT
C
  300 READ(5,301)NOWDAY,ITIS,NOWHR,NOWMIN,IDADD,IHRADD,MINADD,XINCRM
  301 FORMAT(7I10,F10.5)
C     ADJUST FOR LEAPYR IF NECESSARY
      IF(LEAPYR)302,305,302
  302 MONEND(2)=29
  305 II=(SAVE-6.0)/XINCRM
      IDO=LONG/(JUMP*XINCRM)+II+2
      TPOSTN=6.0
      YPOSTN=3.0
      CALL TIMEPT(TPOSTN,NOWDAY,ITIS,NOWHR,NOWMIN,IDADD,IHRADD,MINADD,
     1XINCRM,IDO,MONTHS,MONEND,YPOSTN,HEIGHT)
      WRITE(6,306)
  306 FORMAT(///20X,'MULTYPLT COMPLETED')
      CALL PLOT(6.0,3.0,2)
      CALL PLOT(0.0,0.0,-3)
      STOP
      END
```

APPENDIX D

PROGRAM OVERPLOT

```
      DIMENSION LABEL1(5),LABEL2(5),FMTA(5),FMTB(5),ISYMBL(2),
     1A(4000),B(4000),X(4000),MONTHS(12),MONEND(12)
      DATA MONTHS/' JAN',' FEB',' MAR',' APR',' MAY','JUNE','JULY',
     1' AUG','SEPT',' OCT',' NOV',' DEC'/
      DATA MONEND/31,28,31,30,31,30,31,31,30,31,30,31/
      INTEGER DENSTA,DENSTB
C
C     PROGRAM OVERPLOT
C      REQUIRES SUBROUTINE  TIMEPT
C
   10 FORMAT(10A4,F10.5,I10)
   11 FORMAT(F10.5,5I10,T67,A4)
   12 FORMAT(I10,5A4)
   18 FORMAT(7I10,F10.5)
      READ(5,10)LABEL1,LABEL2,HEIGHT,LEAPYR
      WRITE(6,201)LABEL1,LABEL2,HEIGHT,LEAPYR
  201 FORMAT(///T15,'INPUT PARAMETERS'//T8,'LABEL1',T28,'LABEL2',T45,
     1'HEIGHT',T55,'LEAPYR'//1X,10A4,F9.3,I10)
      IF(LEAPYR)20,21,20
   20 MONEND(2)=29
   21 READ(5,11)XPOSTA,ISCALE,DENSTA,LREAD,LPLOTA,IOPTNA,ISYMBL(1)
      WRITE(6,202)XPOSTA,ISCALE,DENSTA,LREAD,LPLOTA,IOPTNA,ISYMBL(1)
  202 FORMAT(//T5,'XPOSTA',T15,'ISCALE',T25,'DENSTA',T36,'LREAD',T45,
     1'LPLOTA',T55,'IOPTNA',T63,'ISYMBL(1)'//F10.3,5I10,T63,A4)
      READ(5,12)INPUTA,FMTA
      WRITE(6,203)INPUTA,FMTA
  203 FORMAT(//T5,'INPUTA',T17,'FMTA'//I10,6X,10A4)
C
C   READ FIRST SET OF DATA
C
   22 READ(INPUTA,FMTA)(A(I),I=1,LREAD)
   23 READ(5,11)XPOSTB,JAVA,DENSTB,LREAD,LPLOTB,IOPTNB,ISYMBL(2)
      WRITE(6,204)XPOSTB,JAVA,DENSTB,LREAD,LPLOTB,IOPTNB,ISYMBL(2)
  204 FORMAT(//T5,'XPOSTB',T17,'JAVA',T25,'DENSTB',T36,'LREAD',T45,
     1'LPLOTB',T55,'IOPTNB',T63,'ISYMBL(2)'//F10.3,5I10,T63,A4)
      READ(5,12)INPUTB,FMTB
      WRITE(6,205)INPUTB,FMTB
  205 FORMAT(//T5,'INPUTB',T17,'FMTB'//I10,6X,10A4)
C
C   READ SECOND SET OF DATA
C
   24 READ(INPUTB,FMTB)(B(I),I=1,LREAD)
   25 READ(5,18)NOWDAY,NO,NOWHR,NOWMIN,IDADD,IHRADD,MINADD,XINCRM
C
C   FIND MAXIMUM LENGTH OF DATA AND DENSITY OF DATA
C
      IF(LPLOTA-LPLOTB)15,15,16
   15 LPLOT=LPLOTB
```

```
      J=DENSTB
      GO TO 17
   16 LPLOT=LPLOTA
      J=DENSTA
C
C     FIND MAX AND MIN VALUES OF ALL DATA
C
   17 YMAX=-999999.0
      YMIN=9999999.0
      DO 50 I=1,LPLOTA
      YMAX=AMAX1(YMAX,A(I))
   50 YMIN=AMIN1(YMIN,A(I))
      DO 60 I=1,LPLOTB
      YMAX=AMAX1(YMAX,B(I))
   60 YMIN=AMIN1(YMIN,B(I))
      IF(YMIN)61,62,62
C
C     COMPUTE PLOT DISTANCES
C
   61 MINPLT=YMIN/ISCALE
      MAXPLT=YMAX/ISCALE+1
      ALLPLT=MAXPLT-MINPLT
      LIMIT=ALLPLT+1
      KKK=ALLPLT/2
      BELOW=KKK
      ABOVE=ALLPLT-BELOW
      DO 58 KOUNT=1,LPLOTA
   58 A(KOUNT)=A(KOUNT)/ISCALE-MINPLT+3.0
      DO 59 I=1,LPLOTB
   59 B(I)=B(I)/ISCALE-MINPLT+3.0
      GO TO 65
   62 MAXPLT=YMAX/ISCALE+1
      MINPLT=ABS(YMIN)/ISCALE
      ABOVE=MAXPLT
      BELOW=MINPLT
      ALLPLT=ABOVE+BELOW
      LIMIT=ALLPLT+1
      DO 206 I=1,LPLOTA
  206 A(I)=A(I)/ISCALE+BELOW+3.0
      DO 208 I=1,LPLOTB
  208 B(I)=B(I)/ISCALE+BELOW+3.0
C
C     NUMBER THE AXES
C
   65 CALL PLOT(0.0,0.0,JAVA)
      CALL PLOT(6.0,3.0,3)
      CALL PLOT(6.0,ALLPLT+3.0,2)
      Y=ALLPLT+3.0
```

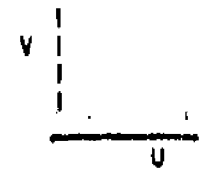
```
      NVALUE=ABOVE*ISCALE
      XX=6.0-((2.0*HEIGHT)/3.0)
      DO 207 I=1,LIMIT
      CALL NUMPLT(XX,Y-.09,0.0,HEIGHT,FLOAT(NVALUE),-1)
      CALL CHAR(XX,Y,0.0,HEIGHT,'_',1)
      Y=Y-1.0
  207 NVALUE=NVALUE-ISCALE
      CENTRY=3.0+ALLPLT/2.0-(10.0*HEIGHT)
C
C     INSERT LABELS
C
      CALL CHAR(6.0-6.5*HEIGHT,CENTRY,90.0,HEIGHT,LABEL1(1),20)
      CALL CHAR(6.0-5.0*HEIGHT,CENTRY,90.0,HEIGHT,LABEL2(1),20)
C
C   CONVERT THE DATA
C
      X(1)=XPOSTA
      FACT=1.0/DENSTA
      DO 101 I=2,LPLOTA
  101 X(I)=X(I-1)+FACT
      CALL VECTOR(X,A,LPLOTA,1,IOPTNA,ISYMBL(1))
      X(1)=XPOSTB
      FACT=1.0/DENSTB
      DO 102 I=2,LPLOTB
  102 X(I)=X(I-1)+FACT
      CALL VECTOR(X,B,LPLOTB,1,IOPTNB,ISYMBL(2))
      OFFSET=3.0+BELOW
      IF(YMIN)110,120,120
  110 IF(X(LPLOTA)-X(LPLOTB))111,111,112
  111 CALL PLOT(X(LPLOTB),OFFSET,3)
      GO TO 113
  112 CALL PLOT(X(LPLOTA),OFFSET,3)
  113 CALL PLOT(6.0,OFFSET,2)
C
C     PRINT THE TIME AXIS
C
  120 IDO=LPLOT/(XINCRM*J)+2
      XXX=6.0
      YYY=3.0
      CALL TIMEPT(XXX,NOWDAY,NC,NOWHR,NOWMIN,IDADD,IHRADD,MINADD,XINCRM,
     1IDO,MONTHS,MONEND,YYY,HEIGHT)
      WRITE(6,209)
  209 FORMAT(//10X,'OVERPLOT COMPLETED')
      CALL PLOT(6.0,3.0,2)
      CALL PLOT(0.0,0.0,-3)
  130 STOP
      END
```

APPENDIX E

PROGRAM STICK

```
      DIMENSION U(4000),V(4000),X(4000)
      DIMENSION MONTHS(12),MONEND(12)
      DIMENSION FMT(10),LABEL1(5),LABEL2(5)
      DATA MONTHS/' JAN',' FEB',' MAR',' APR',' MAY','JUNE','JULY',
     1' AUG','SEPT',' OCT',' NOV',' DEC'/
      DATA MONEND/31,28,31,30,31,30,31,31,30,31,30,31/
      INTEGER PNTDEN
C
C     PROGRAM  STICK
C      REQUIRES SUBROUTINE  TIMEPT
C
      READ(5,10)NPLOTS,NUNITS,DELTAY,HEIGHT,LEAPYR
   10 FORMAT(2I10,2F10.4,I10)
      WRITE(6,9)NPLOTS,NUNITS,DELTAY,HEIGHT,LEAPYR
    9 FORMAT(//15X,'INPUT PARAMETERS'//T5,'NPLOTS',T15,'NUNITS',T26,
     1'DELTAY',T35,'HEIGHT',T45,'LEAPYR'//2I10,2F10.3,I10)
      LONGST=0
      IF(LEAPYR)11,15,11
   11 MONEND(2)=29
   15 Y=4.0
      CALL PLOT(0.0,0.0,NUNITS)
      WRITE(6,16)
   16 FORMAT(//T3,'SERIES #',T15,'XPOSTN',T26,'LREAD',T36,'LPLOT',T45,
     1'IVERTS',T55,'PNTDEN',T65,'IORDUV',T75,'KINPUT')
C
C     BEGIN PLOTTING LOOP
C
      DO 5000 KOUNT=1,NPLOTS
C
C     READ THE PARAMETERS OF EACH SERIES
C
      READ(5,12)XPOSTN,LREAD,LPLOT,IVERTS,PNTDEN,IORDUV,KINPUT
   12 FORMAT(F10.5,6I10)
      IF(LPLOT-LONGST)3,3,2
    2 LONGST=LPLOT
      JUMP=PNTDEN
      SAVEXP=XPOSTN
    3 READ(5,14)LABEL1,LABEL2,FMT
   14 FORMAT(20A4)
      WRITE(6,112)KOUNT,XPOSTN,LREAD,LPLOT,IVERTS,PNTDEN,IORDUV,KINPUT
  112 FORMAT(//I10,F10.4,6I10)
C
C     DETERMINE THE ORDER OF VARIABLES
C     READ THE VARIABLES  U  AND  V
C
      IF(IORDUV)18,17,18
   17 READ(KINPUT,FMT)(V(I),U(I),I=1,LREAD)
      GO TO 20
```

```
   18 READ(KINPUT,FMT)(U(I),V(I),I=1,LREAD)
C
C     SET UP THE X ARRAY OF POINTS DEPENDANT ON   PNTDEN
C
   20 X(1)=XPOSTN
      FACT=1.0/PNTDEN
      DO 19 I=2,LPLOT
   19 X(I)=X(I-1)+FACT
C
C      PLOT U,V AS EAST AND NORTH
C
C
C
C
   25 VMIN=9999.
      VMAX=-99999.
      DO 26 I=1,LPLOT
      VMIN=AMIN1(VMIN,V(I))
   26 VMAX=AMAX1(VMAX,V(I))
      N=VMAX/IVERTS
      M=ABS(VMIN)/IVERTS
      TOP=N+1.0
      BOTTOM=M+1.0
      ALLPLT=TOP+BOTTOM
      Y=Y+BOTTOM
      DO 27 I=1,LPLOT
      U(I)=U(I)/IVERTS+X(I)
   27 V(I)=V(I)/IVERTS+Y
C
C     PLOT THE STICKS
C
   50 DO 60 I=1,LPLOT
   60 CALL PLTLN(X(I),Y,U(I),V(I))
C
C     DRAW THE ZERO LINE
C
      CALL PLOT(X(LPLOT),Y,3)
      CALL PLOT(4.0,Y,2)
C
C     NUMBER THE AXIS OF EACH TIME SERIES
C
      Z=Y+TOP
      CALL PLOT(4.0,Y-BOTTOM,3)
      CALL PLOT(4.0,Z,2)
      NV=TOP*IVERTS
      CENTRY=Y-BOTTOM+(ALLPLT/2.0)-(10.0*HEIGHT)
      XTICK=4.0-(2.0*HEIGHT/3.0)
      XNUMB=XTICK-HEIGHT/2.0
```



```
      REALNV=NV
      IF(REALNV)61,62,63
   61 REALNV=-REALNV
      GO TO 63
   62 REALNV=100.0
   63 YDIGIT=ALOG10(REALNV)
      XL=XNUMB-((3.0+YDIGIT)*HEIGHT)
      NUMB=ALLPLT+1
      DO 86 I=1,NUMB
      CALL NUMPLT(XNUMB,Z-HEIGHT/2.0,0.0,HEIGHT,FLOAT(NV),-1)
      CALL CHAR(XTICK,Z,0.0,HEIGHT,'_',1)
      Z=Z-1.0
   86 NV=NV-IVERTS
C
C     LABEL EACH VARIABLE
C
      CALL CHAR(XL,CENTRY,90.0,HEIGHT,LABEL2(1),20)
      CALL CHAR(XL-2.0*HEIGHT,CENTRY,90.0,HEIGHT,LABEL1(1),20)
      IF(KINPUT-5)4999,5000,4999
 4999 REWIND KINPUT
 5000 Y=Y+TOP+DELTAY
C
C     READ AND PRINT THE TIME INFORMATION
C
      READ(5,54)NOWDAY,ITIS,NOWHR,NOWMIN,IDADD,IHRADD,MINADD,XINCRM
   54 FORMAT(7I10,F10.3)
      II=(SAVEXP-6.0)/XINCRM
      IDO=LONGST/(JUMP*XINCRM)+II+2
      TIMX=6.0
      CALL TIMEPT(TIMX,NOWDAY,ITIS,NOWHR,NOWMIN,IDADD,IHRADD,MINADD,
     1XINCRM,IDO,MONTHS,MONEND,4.0,HEIGHT)
      CALL PLOT(4.0,4.0,2)
      CALL PLOT(0.0,0.0,-3)
      WRITE(6,555)
  555 FORMAT(//10X,'STICK PLOT COMPLETED')
      STOP
      END
```

APPENDIX F

PROGRAM LINLIN

```
C
C     LINEAR-LINEAR PLOT
C
      DIMENSION X(6000),Y(6000),FMT(07),LABELX(10),LABELY(10),LABELZ(10)
C
C     READ VARIOUS PARAMETERS
C
      READ(5,10)XMIN,XMAX,DELTAX,XINCRM,YMIN,YMAX,DELTAY,YINCRM
   10 FORMAT(8F10.5)
      READ(5,12)JAVA,NVARBL,HEIGHT
   12 FORMAT(2I10,F10.4)
      READ(5,17)LABELX,LABELY
   17 FORMAT(20A4)
      READ(5,11)LABELZ,NDECX,NDECY
   11 FORMAT(10A4,2I10)
C
C     PRINT AND CHECK THE INPUT PARAMETERS
C
      WRITE(6,90)XMIN,XMAX,DELTAX,XINCRM,YMIN,YMAX,DELTAY,YINCRM
   90 FORMAT(///15X,'INPUT PARAMETERS'//T7,'XMIN',T17,'XMAX',T25,
     1'DELTAX',T35,'XINCRM',T47,'YMIN',T57,'YMAX',T65,'DELTAY',T75,
     2'YINCRM'//8F10.3)
      WRITE(6,91)LABELX,LABELY
   91 FORMAT(//T18,'LABELX',T58,'LABELY'//1X,20A4)
      WRITE(6,92)LABELZ,NDECX,NDECY,JAVA,HEIGHT
   92 FORMAT(//T18,'LABELZ',T47,'NDECX',T57,'NDECY',T68,'JAVA',T76,
     1'HEIGHT'//1X,10A4,3I10,F10.3)
      WRITE(6,93)
   93 FORMAT(//T3,'SERIES #',T15,'KINPUT',T26,'LREAD',T36,'LPLOT',T46,
     1'IOPTN',T55,'ISYMBL',T68,'FORMAT :')
      KOUNT=0
  101 READ(5,16)KINPUT,LREAD,LPLOT,IOPTN,ISYMBL,FMT
   16 FORMAT(4I10,T47,8A4)
      KOUNT=KOUNT+1
      WRITE(6,94)KOUNT,KINPUT,LREAD,LPLOT,IOPTN,ISYMBL,FMT
   94 FORMAT(/5I10,T57,A4,T66,10A4)
      READ(KINPUT,FMT)(X(I),Y(I),I=1,LREAD)
      IF(KOUNT-1)14,14,201
   14 XDIST=((XMAX-XMIN)/DELTAX)*XINCRM
      YDIST=((YMAX-YMIN)/DELTAY)*YINCRM
      SIZEX=XDIST/XINCRM
      NOX=SIZEX+1.1
      SIZEY=YDIST/YINCRM
      NOY=SIZEY+1.1
C
C     PLOT BORDER
C
      CALL PLOT(0.0,0.0,JAVA)
```

```
      CALL PLOT(4.0,4.0,3)
      CALL PLOT(XDIST+4.0,4.0,2)
      CALL PLOT(XDIST+4.0,YDIST+4.0,2)
      CALL PLOT(4.0,YDIST+4.0,2)
      CALL PLOT(4.0,4.0,2)
C
C     NUMBER THE Y-AXIS
C
      XTICK=4.0-(2.0*HEIGHT/3.0)
      YBIG=AMAX1(YMIN,YMAX)
      IDIGIT=ALOG10(YBIG)
      YDIGIT=IDIGIT
      XNUMBR=XTICK-((1.0+NDECY)*HEIGHT)
      YPOS=4.0
      YVALUE=YMIN
      DO 100 I=1,NOY
      CALL NUMPLT(XNUMBR,YPOS-HEIGHT/2.0,0.0,HEIGHT,YVALUE,NDECY)
      CALL CHAR(XTICK,YPOS,0.0,HEIGHT,'_',1)
      YPOS=YPOS+YINCRM
  100 YVALUE=YVALUE+DELTAY
      CENTRX=XDIST/2.0+4.0-20.0*HEIGHT
      CENTRY=YDIST/2.0+4.0-20.0*HEIGHT
      CALL CHAR(XNUMBR-((3.0+YDIGIT)*HEIGHT),CENTRY,90.0,HEIGHT,
     1LABELY(1),40)
      CALL CHAR(CENTRX,YDIST+2.0*HEIGHT+4.0,0.0,HEIGHT,LABELZ(1),40)
C
C     NUMBER THE X-AXIS
C
      YTICK=4.0-(2.0*HEIGHT/3.0)
      YNUMBR=YTICK-1.5*HEIGHT
      XPOS=4.0
      XVALUE=XMIN
      DO 200 I=1,NOX
      CALL NUMPLT(XPOS,YNUMBR,0.0,HEIGHT,XVALUE,NDECX)
      CALL CHAR(XPOS,YTICK,90.0,HEIGHT,'_',1)
      XPOS=XPOS+XINCRM
  200 XVALUE=XVALUE+DELTAX
      CALL CHAR(CENTRX,YNUMBR-3.0*HEIGHT,0.0,HEIGHT,LABELX(1),40)
C
C     CONVERT DATA POINTS
C
  201 ZX=XINCRM/DELTAX
      ZY=YINCRM/DELTAY
      DO 300 I=1,LPLOT
      X(I)=(X(I)-XMIN)*ZX+4.0
  300 Y(I)=(Y(I)-YMIN)*ZY+4.0
      CALL VECTOR(X,Y,LPLOT,1,ICPTN,ISYMBL)
      IF(KOUNT-NVARBL)298,301,301
```

```
298 IF(KINPUT-5)299,101,299
299 REWIND KINPUT
    GO TO 101
301 CALL PLOT(0.0,0.0,-3)
    STOP
    END
```

APPENDIX G

PROGRAM LINLOG

```
C
C     LIN-LOG
C
      DIMENSION DATA(4000),X(4000),FMT(10),PERIOD(12),LABEL(10)
      DIMENSION LABELY(10)
C
C     READ INPUT VARIABLES
C      PRINT AND CHECK THOSE VARIABLES
C
      READ(5,10)SCLMIN,SCLMAX,SCLDLT,DLTPOS,NDECY,JAVA,DECADE
   10 FORMAT(4F10.3,2I10,F10.3)
      WRITE(6,20)SCLMIN,SCLMAX,SCLDLT,DLTPOS,NDECY,JAVA,DECADE
   20 FORMAT(//T15,'INPUT PARAMETERS'//T5,'SCLMIN',T15,'SCLMAX',T25,
     1'SCLDLT',T35,'DLTPOS',T46,'NDECY',T57,'JAVA',T65,'DECADE'//4F10.3,
     12I10,F10.3)
      READ(5,11)FONE,DELTAF,LREAD,LPLOT,KINPUT,IOPTN,ISYMBL,HEIGHT
   11 FORMAT(2F10.7,4I10,T67,A4,F10.3)
      WRITE(6,21)FONE,DELTAF,LREAD,LPLOT,KINPUT,IOPTN,ISYMBL,HEIGHT
   21 FORMAT(//T7,'FONE',T15,'DELTAF',T26,'LREAD',T36,'LPLOT',T45,
     1'KINPUT',T56,'IOPTN',T65,'ISYMBL',T75,'HEIGHT'//2F10.5,4I10,T67,
     2A4,F10.3)
      READ(5,12)LABEL,IQUAD
   12 FORMAT(10A4,I10)
      WRITE(6,22)LABEL,IQUAD
   22 FORMAT(//T18,'LABEL',T47,'IQUAD'//1X,10A4,I10)
      READ(5,14)LABELY,FMT
   14 FORMAT(20A4)
      WRITE(6,24)LABELY,FMT
   24 FORMAT(//T18,'LABELY',T44,'FORMAT ;'//1X,20A4)
      READ(KINPUT,FMT)(DATA(I),I=1,LREAD)
      YDIST=((SCLMAX-SCLMIN)/SCLDLT)*DLTPOS
      NO=YDIST/DLTPOS+1
      M=ALOG10(FONE)
      XMIN=M-1
      XDIST=ABS(XMIN)*DECADE
C
C     PLOT THE BORDER
C
      CALL PLOT(0.0,0.0,JAVA)
      CALL PLOT(4.0,4.0,3)
      CALL PLOT(XDIST+4.0,4.0,2)
      CALL PLOT(XDIST+4.0,YDIST+4.0,2)
      CALL PLOT(4.0,YDIST+4.0,2)
      CALL PLOT(4.0,4.0,2)
C
C     NUMBER AND LABEL THE Y-AXIS
C
      YPOS=4.0
```

```
      YVALUE=SCLMIN
      IDIGIT=ALOG10(SCLMAX)
      YDIGIT=IDIGIT
      XTICK=4.0-(2.0*HEIGHT)/3.0
      XNUMBR=XTICK-((1.0+NDECY)*HEIGHT)
      XLABEL=XNUMBR-((3.0+YDIGIT)*HEIGHT)
      DO 100 I=1,NO
      CALL NUMPLT(XNUMBR,YPOS-HEIGHT/2.0,0.0,HEIGHT,YVALUE,NDECY)
      CALL CHAR(XTICK,YPOS,0.0,HEIGHT,'_',1)
      YPOS=YPOS+DLTPOS
  100 YVALUE=YVALUE+SCLDLT
      CALL CHAR(XLABEL,YDIST/2.0-(20.0*HEIGHT)+4.0,90.0,HEIGHT,
     1LABELY(1),40)
C
C     SET THE X AND Y POSITICN FOR PRINTING
C      THE LABEL IN THE PROPER QUADRANT
C
      GO TO(101,102,103,104),IQUAD
  101 XL=XDIST+3.0-(40.0*HEIGHT)
      YL=YDIST+3.0
      GO TO 105
  102 XL=XDIST+3.0-(40.0*HEIGHT)
      YL=5.0
      GO TO 105
  103 XL=5.0
      YL=5.0
      GO TO 105
  104 XL=5.0
      YL=YDIST+3.0
  105 CALL CHAR(XL,YL,0.0,HEIGHT,LABEL(1),40)
C
C     NUMBER THE BOTTOM OF THE X-AXIS IN POWERS OF TEN
C     LABEL IT FREQUENCY
C
      JK=ABS(XMIN)
      MX=XMIN+1
      XPOS=4.0+DECADE
      H2=(3.0*HEIGHT)/5.0
      YTICK=4.0-((2.0*HEIGHT)/3.0)
      YPOWER=YTICK-H2-H2/3.0
      YTEN=YPOWER-HEIGHT-HEIGHT/3.0
      DO 110 I=1,JK
      CALL CHAR(XPOS,YTICK,90.0,HEIGHT,'_',1)
      CALL NUMPLT(XPOS+2.*H2,YPOWER,0.0,H2,FLOAT(MX),-1)
      CALL NUMPLT(XPOS+HEIGHT,YTEN,0.0,HEIGHT,FLOAT(10),-1)
      XPOS=XPOS+DECADE
  110 MX=MX+1
      CALL CHAR(XDIST/2.0+4.0-9.0*HEIGHT,4.0-5.0*HEIGHT,0.0,HEIGHT,
```

```
     1'FREQUENCY (C.P.H.)',18)
C
C     NUMBER THE TOP OF THE X-AXIS IN POWERS OF TEN
C     LABEL THIS PERIOD
C
      XP=4.0+DECADE
      YTICK=4.0+YDIST
      YTEN=YTICK+1.5*HEIGHT
      YPOWER=YTEN+HEIGHT+H2/3.
      IJ=ABS(XMIN)
      MX=IJ-1
      DO 400 I=1,IJ
      CALL CHAR(XP,YTICK,90.0,HEIGHT,'_',1)
      CALL NUMPLT(XP+HEIGHT,YTEN,0.0,HEIGHT,FLOAT(10),-1)
      CALL NUMPLT(XP+2.0*H2,YPOWER,0.0,H2,FLOAT(MX),-1)
      MX=MX-1
  400 XP=XP+DECADE
      CALL CHAR(XDIST/2.0+3.0-(7.0*HEIGHT),YPOWER+2.0*HEIGHT,0.0,HEIGHT,
     1'PERIOD (HOURS)',14)
C
C        SPOT FIRST POINT    CONVERT  AND PLOT OTHERS
C
      FIRSTX=4.0
      FIRSTY=((DATA(1)-SCLMIN)/SCLDLT)*DLTPOS+4.0
      CALL GRAF(FIRSTX,FIRSTY,.15,3)
      LPLOT=LPLOT-1
      DO 125 I=1,LPLOT
  125 DATA(I)=((DATA(I+1)-SCLMIN)/SCLDLT)*DLTPOS+4.0
      F=FONE
      DO 126 I=1,LPLOT
      X(I)=(ALOG10(F)-XMIN)*DECADE+4.0
  126 F=F+DELTAF
      CALL VECTOR(X,DATA,LPLOT,1,IOPTN,ISYMBL)
      CALL PLOT(0.0,0.0,-3)
      STOP
      END
```

APPENDIX H

PROGRAM LOGLOG

```
      DIMENSION SP(4000),PERIOD(8),FMTSPT(10),XX(4000)
      DIMENSION LABEL1(10),LABEL2(10),LABEL3(10)
C
C     LOG-LOG PLOT
C
      READ(5,10)FONE,FINCRM,JAVA,LPLOT,KINPUT,HEIGHT,XLATOM,DECADE
   10 FORMAT(2F10.7,3I10,3F10.5)
      WRITE(6,11)FONE,FINCRM,JAVA,LPLOT,KINPUT,HEIGHT,XLATOM,DECADE
   11 FORMAT(//T15,'INPUT PATAMETERS'//T7,'FONE',T15,'FINCRM',T27,
     1'JAVA',T36,'LPLOT',T45,'KINPUT',T55,'HEIGHT',T65,'XLATOM',T75,
     2'DECADE'//2F10.5,3I10,3F10.3)
      READ(5,20)LABEL1,LABEL2
   20 FORMAT(20A4)
      WRITE(6,22)LABEL1,LABEL2
   22 FORMAT(//T18,'LABEL1',T58,'LABEL2'//1X,20A4)
      READ(5,21)LABEL3,FMTSPT
   21 FORMAT(20A4)
      WRITE(6,23)LABEL3,FMTSPT
   23 FORMAT(//T18,'LABELY',T43,'FORMAT :'//1X,20A4)
      READ(KINPUT,FMTSPT,END=601)(SP(I),I=1,LPLOT)
C
C     EVALUATE THE MAX AND MIN
C
      YMAX=-999999.
      YMIN=9999999.
      DO 200 I=1,LPLOT
      IF(SP(I))190,195,195
  190 SP(I)=SP(1)
  195 SP(I)=ALOG10(SP(I))
      YMIN=AMIN1(SP(I),YMIN)
  200 YMAX=AMAX1(SP(I),YMAX)
      M=YMAX
      IF(YMIN)40,50,50
   40 N=YMIN
      YMIN=N-1.0
      GO TO 51
   50 N=YMIN
      YMIN=N
   51 YMAX=M+1.0
      A=ALOG10(FONE)
      N=A
      XMIN=N-1
      XDIST=ABS(XMIN)*DECADE
      YDIST=(YMAX-YMIN)*DECADE
      THETA=(XLATOM*3.14159)/180.
      FINERT=SIN(THETA)/12.0
      FTWELV=1.0/12.41
      FINERT=(ALOG10(FINERT)+ABS(XMIN))*DECADE+3.0
```

```
      FTWELV=(ALOG10(FTWELV)+ABS(XMIN))*DECADE+3.0
      WRITE(6,52)XDIST,YDIST
   52 FORMAT(///T3,'COMPUTED DISTANCES :',T36,'XDIST',T46,'YDIST'//
     130X,2F10.2)
      CALL PLOT(0.0,0.0,JAVA)
      CALL PLOT(3.0,3.0,3)
      CALL PLOT(XDIST+3.0,3.0,2)
      CALL PLOT(XDIST+3.0,YDIST+3.0,2)
      CALL PLOT(3.0,YDIST+3.0,2)
      CALL PLOT(3.0,3.0,2)
C
C   NUMBER THE X AXIS
C
      H2=(3.0*HEIGHT)/5.0
      X=3.0+DECADE
      IJ=ABS(XMIN)
      MX=XMIN+1
      YTICK=3.0-((2.0*HEIGHT)/3.0)
      YPOWER=YTICK-H2-H2/3.0
      YTEN=YPOWER-HEIGHT-HEIGHT/3.0
      DO 300 I=1,IJ
      CALL CHAR(X,YTICK,90.0,HEIGHT,'_',1)
      CALL NUMPLT(X+2.0*H2,YPOWER,0.0,H2,FLOAT(MX),-1)
      CALL NUMPLT(X+HEIGHT,YTEN,0.0,HEIGHT,FLOAT(10),-1)
      X=X+DECADE
  300 MX=MX+1
C
C     MARK THE INERTIAL AND TIDAL FREQUENCIES
C
      CALL CHAR(FINERT,3.0,90.0,HEIGHT,'_',1)
      CALL CHAR(FTWELV,3.0,90.0,HEIGHT,'_',1)
      CALL CHAR(XDIST/2.0+3.0-9.0*HEIGHT,YTEN-2.5*HEIGHT,0.0,HEIGHT,
     1'FREQUENCY (C.P.H.)',18)
C
C     LABEL THE Y AXIS
C
      YEND=3.0
      NP=YMIN
      XTICK=3.0-((2.0*HEIGHT)/3.0)
      XPOWER=XTICK-H2/2.0
      XTEN=XPOWER-HEIGHT/5.0
      K=YMAX-YMIN+1
      DO 310 I=1,K
      CALL CHAR(XTICK,YEND,00.0,HEIGHT,'_',1)
      CALL NUMPLT(XPOWER,YEND+H2/3.,0.0,H2,FLOAT(NP),-1)
      CALL NUMPLT(XTEN,YEND-HEIGHT,0.0,HEIGHT,FLOAT(10),-1)
      NP=NP+1
  310 YEND=YEND+DECADE
```

```
C
C     LABEL THE TOP OF THE X AXIS    PERIOD
C
      X=3.0+DECADE
      YTICK=3.0+YDIST
      YTEN=YTICK+1.5*HEIGHT
      YPOWER=YTEN+HEIGHT+H2/3.
      IJ=ABS(XMIN)
      MX=IJ-1
      DO 400 I=1,IJ
      CALL CHAR(X,YTICK,90.0,HEIGHT,'-',1)
      CALL NUMPLT(X+HEIGHT,YTEN,0.0,HEIGHT,FLOAT(10),-1)
      CALL NUMPLT(X+2.0000*H2,YPOWER,0.0,H2,FLOAT(MX),-1)
      MX=MX-1
  400 X=X+DECADE
      CALL CHAR(XDIST/2.0+3.0-(7.0*HEIGHT),YPOWER+2.0*HEIGHT,0.0,HEIGHT,
     1'PERIOD (HOURS)',14)
      XLB=XDIST+2.5-(40.0*HEIGHT)
      CALL CHAR(XLB,YDIST+2.0,0.0,HEIGHT,LABEL1(1),40)
      CALL CHAR(XLB,YDIST+1.0,0.0,HEIGHT,LABEL2(1),40)
      CALL CHAR(XTEN-3.0*HEIGHT,(YDIST/2.0)-(20.0*HEIGHT)+3.0,90.0,
     1HEIGHT,LABEL3(1),40)
C
C
C     CONVERT THE DATA POINTS
C
      X1=3.0
      SP(1)=(SP(1)-YMIN)*DECADE+3.0
      CALL GRAF(X1,SP(1),.20,3)
      LPLOT=LPLOT-1
      DO 500 I=1,LPLOT
  500 SP(I)=(SP(I+1)-YMIN)*DECADE+3.0
      XX(1)=(ALOG10(FONE)-XMIN)*DECADE+3.0
      F=FONE
      DO 600 I=2,LPLOT
      F=F+FINCRM
  600 XX(I)=(ALOG10(F)-XMIN)*DECADE+3.0
      CALL VECTOR(XX,SP,LPLOT,1,0,'?')
      CALL PLOT(0.0,0.0,-3)
      GO TO 603
  601 WRITE(6,602)KINPUT
  602 FORMAT(///10X,'***** EOF ON UNIT ',I4,'  *******'/)
  603 STOP
      END
```

APPENDIX I

PROGRAM LOGLIN

```
C
C     LOG-LIN
C
      DIMENSION SP(4000),SP2(4000),X(4000),FMTSP1(10),FMTSP2(10),
     1LABEL1(10),LABEL2(10),LABEL3(10),PERIOD(12),LABELY(10)
      DIMENSION ISYMBL(2)
      INTEGER PNTDEN
C
C     INPUT THE GENERAL FEATURES
C
      READ(5,10)JAVA,FSTART,FEND,FINCRM,XINCRM,H,DECADE,NDECF
   10 FORMAT(I10,4F10.7,2F10.5,I10)
      READ(5,11)LABEL1,LABEL2
      READ(5,11)LABEL3,LABELY
   11 FORMAT(20A4)
C
C     READ PARAMETERS SPECIFIC TO SERIES ONE
C
      READ(5,12)KINPUT,LREAD,LPLOT1,FACTOR,IOPTN1,ISYMBL(1)
   12 FORMAT(3I10,F10.5,I10,T57,A4)
      READ(5,16)FMTSP1,FONE,DELTAF
   16 FORMAT(10A4,2F10.7)
      READ(KINPUT,FMTSP1)(SP(I),I=1,LREAD)
C
C     PRINT THE INPUT VARIABLES
C     CHECK THEIR VALUE
C
      WRITE(6,14)FSTART,FEND,FINCRM,XINCRM,H,DECADE,NDECF,KINPUT,LREAD,
     1           LPLOT1,FACTOR,IOPTN1,ISYMBL(1),FMTSP1
   14 FORMAT(//20X,'INPUT PARAMETERS'//T5,'FSTART',T17,'FEND',T25,
     1'FINCRM',T35,'XINCRM',T48,'H',T55,'DECADE',T66,'NDECF'//4F10.4,
     22F10.3,I10///T5,'KINPUT',T16,'LREAD',T25,'LPLOT1',T35,'FACTOR',
     3T45,'IOPTN1',
     4T53,'ISYMBL(1)',T66,'FORMAT:'//3I10,F10.3,I10,T56,A4,T66,10A4//)
      YMAX=-99999.
      YMIN=9999999.
      DO 55 I=1,LPLOT1
      SP(I)=SP(I)*FACTOR
      IF(SP(I))51,52,52
   51 SP(I)=SP(1)
   52 SP(I)=ALOG10(SP(I))
      YMAX=AMAX1(YMAX,SP(I))
   55 YMIN=AMIN1(YMIN,SP(I))
C
C     READ PARAMETERS SPECIFIC TO SERIES TWO
C
      READ(5,12)KINPUT,LREAD,LPLOT2,FACTOR,IOPTN2,ISYMBL(2)
      READ(5,11)FMTSP2
```

```
      READ(KINPUT,FMTSP2)(SP2(I),I=1,LREAD)
      WRITE(6,15)KINPUT,LREAD,LPLOT2,FACTOR,IOPTN2,ISYMBL(2),FMTSP2
   15 FORMAT(T5,'KINPUT',T16,'LREAD',T25,'LPLOT2',T35,'FACTOR',T45,
     1'IOPTN2',T53,'ISYMBL(2)',T66,'FORMAT:'//3I10,F10.3,I10,T56,A4,T66,
     210A4)
      WRITE(6,17)FONE,DELTAF
   17 FORMAT(//11X,'FONE = ',F10.5,4X,'DELTAF = ',F10.5)
      DO 65 I=1,LPLOT2
      SP2(I)=SP2(I)*FACTOR
      IF(SP2(I))61,62,62
   61 SP2(I)=SP(1)
   62 SP2(I)=ALOG10(SP2(I))
      YMAX=AMAX1(YMAX,SP2(I))
   65 YMIN=AMIN1(YMIN,SP2(I))
      IF(YMIN)200,201,201
  200 N=YMIN
      YMIN=N-1.0
      GO TO 202
  201 N=YMIN
      YMIN=N
  202 M=YMAX
      YMAX=M+1.0
      IF(YMAX-YMIN-1.0)104,104,105
  104 YMAX=YMAX+1.0
  105 CALL PLOT(0.0,0.0,JAVA)
      YDIST=(YMAX-YMIN)*DECADE
      XDIST=((FEND-FSTART)/FINCRM)*XINCRM
      WRITE(6,401)XDIST,YDIST
  401 FORMAT(//10X,'XDIST = ',F10.2,5X,'YDIST = ',F10.2)
C
C     PLOT THE BORDER
C
      CALL PLOT(4.0,4.0,3)
      CALL PLOT(4.0+XDIST,4.0,2)
      CALL PLOT(4.0+XDIST,4.0+YDIST,2)
      CALL PLOT(4.0,4.0+YDIST,2)
      CALL PLOT(4.0,4.0,2)
C
C     NUMBER AND TICK THE X-AXIS (FREQUENCY)
C
      FREQ=FSTART
      NOX=XDIST/XINCRM+1.5
      XX=4.0
      YLEVEL=4.0-(2.0*H)/3.
      K=0
      DO 110 I=1,NOX
      CALL NUMPLT(XX,4.0-2.25*H,0.0,H,FREQ,NDECF)
      CALL CHAR(XX,YLEVEL,90.0,H,'_',1)
```

```
      K=K+1
      IF(FREQ)109,108,109
  108 PERIOD(K)=0.0
      GO TO 112
  109 PERIOD(K)=1.0/FREQ
  112 FREQ=FREQ+FINCRM
  110 XX=XX+XINCRM
      CALL CHAR((XDIST/2.0)+4.0-(9.0*H),4.0-5.5*H,0.0,H,
     1'FREQUENCY (C.P.H.)',18)
C
C     NUMBER THE Y-AXIS IN POWER OF TEN NOTATION
C
      IDIFF=YMAX-YMIN
      IDD=IDIFF+1
      Z=YDIST
      H2=(3.0*H)/5.0
      YEND=4.0
      NP=YMIN
      XTICK=4.0-(2.0*H)/3.0
      XPOWER=XTICK-H2/2.
      XTEN=XPOWER-H/5.0
      DO 120 K=1,IDD
      CALL CHAR(XTICK,YEND,0.0,H,'_',1)
      CALL NUMPLT(XPOWER,YEND+H2/3.0,0.0,H2,FLOAT(NP),-1)
      CALL NUMPLT(XTEN,YEND-H,0.0,H,FLOAT(10),-1)
      NP=NP+1
  120 YEND=YEND+DECADE
      XLABL=XDIST+3.5-(40.0*H)
      CALL CHAR(XLABL,Z+3.0,0.0,H,LABEL1(1),40)
      CALL CHAR(XLABL,Z+2.0,0.0,H,LABEL2(1),40)
      CALL CHAR(XLABL,Z+1.0,0.0,H,LABEL3(1),40)
      XEND=4.0+XINCRM
      YLEVEL=Z+4.0
      DO 125 I=2,NOX
      CALL NUMPLT(XEND+.15,Z+4.0+1.5*H,0.0,H,PERIOD(I),1)
      CALL CHAR(XEND,YLEVEL,90.0,H,'_',1)
  125 XEND=XEND+XINCRM
      CALL CHAR((XDIST/2.0)+4.0-(7.0*H),Z+4.0+3.5*H,0.0,H,
     1'PERIOD (HOURS)',14)
      CALL CHAR(XTEN-3.0*H,YDIST/2.0-(20.0*H)+4.0,90.0,H,LABELY(1),40)
C
C     CONVERT THE X AND Y POINTS TO PLOTTER UNITS
C     SPOT BOTH ZERO ESTIMATES AT X=4.0
C
      FIRSTX=4.0
      FIRSTY=(SP(1)-YMIN)*DECADE+4.0
      CALL GRAF(FIRSTX,FIRSTY,.20,3)
      FIRSTY=(SP2(1)-YMIN)*DECADE+4.0
```

```
      CALL GRAF(FIRSTX,FIRSTY,.20,3)
C
C      PLOT ALL OTHERS AT THE APPROPRIATE FREQUENCY LOCATION
C
      LPLOT1=LPLOT1-1
      DO 149 I=1,LPLOT1
  149 SP(I)=(SP(I+1)-YMIN)*DECADE+4.0
      LPLOT2=LPLOT2-1
      DO 150 I=1,LPLOT2
  150 SP2(I)=(SP2(I+1)-YMIN)*DECADE+4.0
      IF(LPLOT1-LPLOT2)160,155,155
  155 LENGTH=LPLOT1
      GO TO 161
  160 LENGTH=LPLOT2
C
C     RECONSTRUCT THE X POINTS.   FREQUENCY
C
  161 ZX=XINCRM/FINCRM
      F=FONE
      DO 175 I=1,LENGTH
      X(I)=(F-FSTART)*ZX+4.0
  175 F=F+DELTAF
C
C     DISPLAY THE SPECTRA IN OPTION CHOSEN
C
      CALL VECTOR(X,SP,LPLOT1,1,IOPTN1,ISYMBL(1))
      CALL PLOT(0.0,0.0,-3)
      CALL VECTOR(X,SP2,LPLOT2,1,IOPTN2,ISYMBL(2))
      CALL PLOT(0.0,0.0,-3)
      WRITE(6,93)
   93 FORMAT(//10X,'   LOGLIN PLOT COMPLETE')
      STOP
      END
```

APPENDIX J

PROGRAM SPECIALL

```
C
C     SPECIALL
C     THE FOLLOWING IS A MAIN PROGRAM FOR PLOTTING VARIABLES
C     IN A LINEAR-LINEAR MODE WHEN THE X VARIABLE IS ACTUALLY FREQUENCY
C
      DIMENSION SP(4000),X(4000),LABELX(10),LABELY(10),LABELZ(10),FMT(7)
      DIMENSION PERIOD(12)
      READ(5,10)JAVA,KINPUT,LREAD,LPLOT,IOPTN,FMT
   10 FORMAT(5I10,7A4)
      WRITE(6,30)JAVA,KINPUT,LREAD,LPLOT,IOPTN,FMT
   30 FORMAT(//15X,'INPUT PARAMETERS'//T7,'JAVA',T15,'KINPUT',T26,
     1'LREAD',T36,'LPLOT',T46,'IOPTN',T58,'FMT'//5I10,T57,7A4)
      READ(5,14)LABELX,LABELY
      WRITE(6,31)LABELX,LABELY
   31 FORMAT(//T18,'LABELX',T58,'LABELY'//1X,20A4)
      READ(5,15)LABELZ,IQUAD,NDECX,NDECY
   14 FORMAT(20A4)
   15 FORMAT(10A4,3I10)
      READ(5,18)FMIN,FMAX,SCALEF,DELTAX,YMIN,YMAX,DELTAS,DELTAY
   18 FORMAT(8F10.4)
      WRITE(6,32)LABELZ,IQUAD,NDECX,NDECY
   32 FORMAT(//T18,'LABELZ',T46,'IQUAD',T56,'NDECX',T66,'NDECY'//
     11X,10A4,I9,2I10)
      WRITE(6,33)FMIN,FMAX,SCALEF,DELTAX,YMIN,YMAX,DELTAS,DELTAY
   33 FORMAT(//T7,'FMIN',T17,'FMAX',T25,'SCALEF',T35,'DELTAX',T47,
     1'YMIN',T57,'YMAX',T65,'DELTAS',T75,'DELTAY'//8F10.5)
      READ(5,19)FONE,DELTAF,HEIGHT,KSTART
   19 FORMAT(2F10.7,F10.3,I10)
      WRITE(6,35)FONE,DELTAF,HEIGHT,KSTART
   35 FORMAT(//T7,'FONE',T15,'DELTAF',T25,'HEIGHT',T35,'KSTART'//2F10.5,
     1F10.3,I10)
      READ(KINPUT,FMT)(SP(I),I=1,LREAD)
      XDIST=((FMAX-FMIN)/SCALEF)*DELTAX
      YDIST=((YMAX-YMIN)/DELTAS)*DELTAY
      NDX=XDIST/DELTAX+1.1
      NDY=YDIST/DELTAY+1.1
C
C     PLOT THE BORDER
C
      CALL PLOT(0.0,0.0,JAVA)
      CALL PLOT(4.0,4.0,3)
      CALL PLOT(XDIST+4.0,4.0,2)
      CALL PLOT(XDIST+4.0,YDIST+4.0,2)
      CALL PLOT(4.0,YDIST+4.0,2)
      CALL PLOT(4.0,4.0,2)
C
C     NUMBER THE Y AXIS
C
```

```
      YPOS=4.0
      YVALUE=YMIN
      DO 100 I=1,NOY
      CALL NUMPLT(3.1-(NDECY*.25),YPOS-.15,0.0,HEIGHT,YVALUE,NDECY)
      CALL CHAR(3.82,YPOS,0.0,.24,'-',1)
      YPOS=YPOS+DELTAY
  100 YVALUE=YVALUE+DELTAS
      CALL CHAR(4.0-7.0*HEIGHT,YDIST/2.0-2.0,90.0,HEIGHT,LABELY(1),40)
      PERIOD(1)=0.0
C
C     NUMBER THE X AXIS  BOTTOM AND TOP
C
      XPOS=4.0
      XVALUE=FMIN
      DO 200 I=1,NOX
      CALL NUMPLT(XPOS,3.3,0.0,HEIGHT,XVALUE,NDECX)
      CALL CHAR(XPOS,3.86,90.0,.24,'-',1)
      IF(XVALUE)199,199,198
  198 PERIOD(I)=1.0/XVALUE
  199 XPOS=XPOS+DELTAX
  200 XVALUE=XVALUE+SCALEF
      CALL CHAR(XDIST/2.0-2.0,4.0-5.*HEIGHT,0.0,HEIGHT,LABELX(1),40)
      XPOS=4.0+DELTAX
      DO 250 I=2,NOX
      CALL NUMPLT(XPOS,YDIST+4.5,0.0,HEIGHT,PERIOD(I),1)
      CALL CHAR(XPOS,YDIST+4.00,90.,.24,'-',1)
  250 XPOS=XPOS+DELTAX
      CALL CHAR(XDIST/2.0+1.9,YDIST+4.0+4.0*HEIGHT,0.0,HEIGHT,
     1'PERIOD (HOURS)',14)
      GO TO (251,252,253,254),IQUAD
  251 XL=XDIST+3.0-(40.0*HEIGHT)
      YL=YDIST+3.0
      GO TO 255
  252 XL=XDIST+3.0-(40.0*HEIGHT)
      YL=5.0
      GO TO 255
  253 XL=5.0
      YL=5.0
      GO TO 255
  254 XL=5.0
      YL=YDIST+3.0
  255 CALL CHAR(XL,YL,0.0,HEIGHT,LABELZ(1),40)
C
C     CONVERT POINTS
C
      ZX=DELTAX/SCALEF
      ZY=DELTAY/DELTAS
      IF(KSTART-1)222,295,298
```

```
  295 FIRSTX=4.0
      FIRSTY=(SP(1)-YMIN)*ZY+4.0
      CALL GRAF(FIRSTX,FIRSTY,HEIGHT,3)
      N=2
      GO TO 299
  298 N=KSTART
C
C     CONVERT THE REMAINING POINTS
C
  299 F=FONE
      KOUNT=0
      DO 300 I=N,LPLOT
      KOUNT=KOUNT+1
      X(KOUNT)=(F-FMIN)*ZX+4.0
      F=F+DELTAF
  300 SP(KOUNT)=(SP(I)-YMIN)*ZY+4.0
C
C     DO THE PLOTTING
C
      CALL VECTOR(X,SP,KOUNT,1,IOPTN,'+')
      CALL PLOT(0.0,0.0,-3)
      WRITE(6,37)
   37 FORMAT(//10X,'SPECIALL COMPLETE')
  222 STOP
      END
```

APPENDIX K

SUBROUTINE TIMEPT

```
      SUBROUTINE TIMEPT(XX,NOWDAY,ITIS,NOWHR,NOWMIN,IDADD,IHRADD,MINADD,
     1XINCRM,IDO,MONTHS,MONEND,Y,HEIGHT)
      DIMENSION MONTHS(1),MONEND(1)
      IF(IHRADD)222,10,60
   10 IF(MINADD)222,12,60
C
C     BOTH DAYADD AND MIN ADD ARE ZERO IE.  DAILY INCREMENTS
C
   12 XXSAVE=XX
      Z=HEIGHT/2.0
      YDAY=Y-HEIGHT-.4
      YMONTH=YDAY-HEIGHT-Z
      DO 50 I=1,IDO
      CALL CHAR(XX,Y-(2.0*HEIGHT/3.0),90.0,HEIGHT,'_',1)
      CALL NUMPLT(XX+HEIGHT,YDAY,0.0,HEIGHT,FLOAT(NOWDAY),-1)
      NOWDAY=NOWDAY+IDADD
      IF(NOWDAY-MONEND(ITIS))30,30,20
   20 NOWDAY=NOWDAY-MONEND(ITIS)
      XDIST=XX-XXSAVE
      XBAR=(XDIST/2.0)+XXSAVE
      IF(XBAR-XXSAVE-2.0)28,28,26
   26 CALL CHAR(XBAR-HEIGHT*2.,YMONTH,0.0,HEIGHT,MONTHS(ITIS),4)
   28 XXSAVE=XX
      ITIS=ITIS+1
      IF(ITIS-12)30,30,29
   29 ITIS=1
   30 XX=XX+XINCRM
   50 CONTINUE
      XF=XX-XINCRM
      XDIST=XF-XXSAVE
      XBAR=XDIST/2.0+XXSAVE
      IF(XBAR-XXSAVE-2.0)250,51,51
   51 CALL CHAR(XBAR-HEIGHT*2.0,YMONTH,0.0,HEIGHT,MONTHS(ITIS),4)
      GO TO 250
C
C     SMALL TIME INCREMENT LOOP
C
   60 YHOUR=Y-HEIGHT-.4
      Z=HEIGHT/2.0
      YDAY=YHOUR-HEIGHT-Z
      YMONTH=YDAY-HEIGHT-Z
  166 XXSAVE=XX
      DO 106 I=1,IDO
      CALL CHAR(XX,Y-(2.0*HEIGHT/3.0),90.0,HEIGHT,'_',1)
      CALL NUMPLT(XX,YHOUR,0.0,HEIGHT,FLOAT(NOWHR),-1)
      CALL CHAR(XX-.40*HEIGHT,YHOUR,0.0,HEIGHT,':',1)
      CALL NUMPLT(XX+2.5*HEIGHT,YHOUR,0.0,HEIGHT,FLOAT(NOWMIN),-1)
      CALL NUMPLT(XX+HEIGHT,YDAY,0.0,HEIGHT,FLOAT(NOWDAY),-1)
```

```
      NOWMIN=NOWMIN+MINADD
      IF(NOWMIN-60)80,75,75
   75 NOWMIN=NOWMIN-60
      NOWHR=NOWHR+IHRADD
   80 NOWHR=NOWHR+IHRADD
      IF(NOWHR-24)90,85,85
   85 NOWHR=NOWHR-24
      NOWDAY=NOWDAY+1
   90 NOWDAY=NOWDAY+IDADD
      IF(NOWDAY-MONEND(ITIS))150,150,100
  100 NOWDAY=NOWDAY-MONEND(ITIS)
      XDIST=XX-XXSAVE
      XBAR=(XDIST/2.0)+XXSAVE
      IF(XBAR-XXSAVE-2.0)104,102,102
  102 CALL CHAR(XBAR-HEIGHT*2.0,YMONTH,0.0,HEIGHT,MONTHS(ITIS),4)
  104 XXSAVE=XX
      ITIS=ITIS+1
      IF(ITIS-12)150,150,105
  105 ITIS=1
  150 XX=XX+XINCRM
  106 CONTINUE
      XF=XX-XINCRM
      XDIST=XF-XXSAVE
      XBAR=XDIST/2.0+XXSAVE
      IF(XBAR-XXSAVE-2.0)250,151,151
  151 CALL CHAR(XBAR-HEIGHT*2.0,YMONTH,0.0,HEIGHT,MONTHS(ITIS),4)
      GO TO 250
  222 WRITE(6,800)
  800 FORMAT(//10X,' ERROR IN TIME SUBROUTINE'//)
  250 CALL PLOT(XF,Y,3)
      RETURN
      END
```