TIGS - AN INTERACTIVE GRAPHICAL SYSTEM FOR THE CREATION AND CORRECTION OF TABULAR DATA SETS

M. J. Caddy
Aircraft and Crew Systems Technology Directorate
NAVAL AIR DEVELOPMENT CENTER
Warminster, Pennsylvania 18974

1 MAY 1983

FINAL REPORT
AIRTASK NO. A5365360/001F/80598W0598002
Work Unit No. RF908
AIRTASK NO. A2695100/001C/8W04770003-01
Work Unit No. XN401/X5701

DTIC QUALITY INSPECTED 2

Prepared For
NAVAL AIR SYSTEMS COMMAND
Department of the Navy
Washington, D.C. 20361
**REPORT DOCUMENTATION PAGE**

1. **REPORT NUMBER**
   NADC-83030-60

2. **GOVT ACCESSION NO.**

3. **RECIPIENT'S CATALOG NUMBER**

4. **TITLE (and Subtitle)**
   TIGS - An Interactive Graphical System for the Creation and Correction of Tabular Data Sets

5. **TYPE OF REPORT & PERIOD COVERED**
   Final Report

6. **PERFORMING ORG. REPORT NUMBER**

7. **AUTHOR(s)**
   M. J. Caddy

8. **CONTRACT OR GRANT NUMBER(s)**

9. **PERFORMING ORGANIZATION NAME AND ADDRESS**
   Aircraft & Crew Systems Technology Directorate
   (Code 6052)
   Naval Air Development Center, Warminster, Pa. 18901

10. **PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS**
    A53653607001F/80599W0598002
    RF906 & A2695100/001C/
    8W04770003-01 XN401/X5701

11. **CONTRACTING OFFICE NAME AND ADDRESS**
    Naval Air Systems Command
    Department of the Navy
    Washington, DC 20361

12. **REPORT DATE**
    1 May 1983

13. **NUMBER OF PAGES**
    53

14. **MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)**

15. **SECURITY CLASS. (of this report)**
    UNCLASSIFIED

15a. **DECLASSIFICATION/DOWNGRADING SCHEDULE**

16. **DISTRIBUTION STATEMENT (of this Report)**
    Approved for public release; distribution unlimited.

17. **DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)**

18. **SUPPLEMENTARY NOTES**

19. **KEY WORDS (Continue on reverse side if necessary and identify by block number)**
   Interactive Computer Graphics

20. **ABSTRACT (Continue on reverse side if necessary and identify by block number)**
    A general purpose interactive graphical computer code is described which permits interactive graphical creation and correction of tabular data sets. This code was developed for Tektronics 4015 hardware utilizing the NADC 6600/Cyber 175 computer facilities.
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>3</td>
</tr>
<tr>
<td>Discussion</td>
<td>3</td>
</tr>
<tr>
<td>Code Development</td>
<td>3</td>
</tr>
<tr>
<td>Hardware Requirements</td>
<td>3</td>
</tr>
<tr>
<td>Software Overview</td>
<td>5</td>
</tr>
<tr>
<td>User Experience</td>
<td>5</td>
</tr>
<tr>
<td>Conclusions</td>
<td>8</td>
</tr>
<tr>
<td>References</td>
<td>8</td>
</tr>
<tr>
<td>Appendix A – User’s Guide</td>
<td>A-1</td>
</tr>
<tr>
<td>Input Considerations</td>
<td>A-2</td>
</tr>
<tr>
<td>Table Data Format</td>
<td>A-2</td>
</tr>
<tr>
<td>Examples</td>
<td>A-2</td>
</tr>
<tr>
<td>Limitations</td>
<td>A-2</td>
</tr>
<tr>
<td>Interactive Prompting</td>
<td>A-2</td>
</tr>
<tr>
<td>Tablet Initiation Procedure</td>
<td>A-10</td>
</tr>
<tr>
<td>Tigs Interactive Commands</td>
<td>A-12</td>
</tr>
<tr>
<td>Example Interactive Session</td>
<td>A-13</td>
</tr>
<tr>
<td>Appendix B – Tigs Code Fortran Listing</td>
<td>B-1</td>
</tr>
</tbody>
</table>
### LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure No.</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Input and Output Axis Representation</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>TIGS Example Plot</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>TIGS Segmentation Modules</td>
<td>7</td>
</tr>
<tr>
<td>A-1</td>
<td>One Parameter Table Look-up</td>
<td>A-4</td>
</tr>
<tr>
<td>A-2</td>
<td>Two Parameter Table Look-up</td>
<td>A-6</td>
</tr>
<tr>
<td>A-3</td>
<td>Three Parameter Table Look-up</td>
<td>A-8</td>
</tr>
<tr>
<td>A-4</td>
<td>Tablet Command Menu</td>
<td>A-11</td>
</tr>
<tr>
<td>A-5</td>
<td>Example Plot</td>
<td>A-14</td>
</tr>
<tr>
<td>A-6</td>
<td>Example Plot</td>
<td>A-16</td>
</tr>
<tr>
<td>A-7</td>
<td>Example Plot</td>
<td>A-17</td>
</tr>
<tr>
<td>A-8</td>
<td>Example Plot</td>
<td>A-18</td>
</tr>
<tr>
<td>A-9</td>
<td>Example Plot</td>
<td>A-19</td>
</tr>
<tr>
<td>A-10</td>
<td>Example Plot</td>
<td>A-20</td>
</tr>
<tr>
<td>A-11</td>
<td>Example Plot</td>
<td>A-21</td>
</tr>
<tr>
<td>A-12</td>
<td>Example Plot</td>
<td>A-22</td>
</tr>
<tr>
<td>A-13</td>
<td>Example Plot</td>
<td>A-23</td>
</tr>
<tr>
<td>A-14</td>
<td>Example Plot</td>
<td>A-24</td>
</tr>
<tr>
<td>A-15</td>
<td>Example Plot</td>
<td>A-26</td>
</tr>
<tr>
<td>A-16</td>
<td>Example Plot</td>
<td>A-27</td>
</tr>
<tr>
<td>A-17</td>
<td>Example Plot</td>
<td>A-28</td>
</tr>
</tbody>
</table>

### LIST OF TABLES

<table>
<thead>
<tr>
<th>Table No.</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-I</td>
<td>Data Input Instructions</td>
<td>A-3</td>
</tr>
<tr>
<td>A-II</td>
<td>One Parameter Card Inputs</td>
<td>A-5</td>
</tr>
<tr>
<td>A-III</td>
<td>Two Parameter Card Inputs</td>
<td>A-7</td>
</tr>
<tr>
<td>A-IV</td>
<td>Three Parameter Card Inputs</td>
<td>A-9</td>
</tr>
</tbody>
</table>
INTRODUCTION

The NAVAIRDEVVCN (Naval Air Development Center) is a large user of vehicle and propulsion design and performance computer codes and is constantly seeking to improve their efficiency and flexibility. A significant number of these codes are dependent on the use of input tabular data sets. Quite frequently these data sets are initially received in a format incompatible with direct use in these codes, resulting in a time consuming, error prone transformation task. To circumvent this problem, development of a rapid data transformation code was undertaken. The impetus for this effort was the need to prepare for a planned substantial increase in analyses of various aircraft and propulsion systems.

This present report describes a code based on the use of an interactive graphics system that permits direct creation of digital tabular data sets from material in graph form, utilizing a Tektronics 4015 graphics terminal, digitizer tablet and hardcopy unit. In addition the user may edit and correct these data directly from the digitizer tablet or from the graphics display screen using cursor cross hairs and tablet commands. This code, entitled TIGS (Table Plot Interactive Graphics System) was developed using the NAVAIRDEVVCN CDC 6600/Cyber 175 computer facilities. A user’s guide for this code is shown in Appendix A. A Fortran listing of the TIGS code is shown in Appendix B.

DISCUSSION

CODE DEVELOPMENT

The TIGS code was developed as a general purpose computer tool to permit the user to prepare and edit tabular data sets, using interactive graphics, prior to use in other computer codes. The tabular data sets may represent a functional relationship between a dependent variable and several independent variables, an example of which is shown in Figure 1. In this figure FXYZ is the dependent variable and is a function of the independent variables X, Y, and Z. The basic output of the code is graphical plots on a Tektronics 4015 type of storage tube graphics terminal along with a computer file consisting of the digital tabular data representation of that plot. These digital tabular data are suitable for use in nearly all of the vehicle and propulsion design computer codes used within the Aircraft and Crew Systems Technology Directorate at the NAVAIRDEVVCN. Further details of the tabular data output format are discussed in the user’s guide Appendix A and in reference (a). While the TIGS code is a stand-alone interactive system, the graphical executive portion of the code may be used in conjunction with any other user written code. In effect this flexibility permits the user to interactively prepare and edit data which in turn is passed to the user’s code. Experience in using TIGS has shown that the time required to prepare data for use in the vehicle and propulsion design codes has been reduced by a factor of 10.

HARDWARE REQUIREMENTS

The TIGS code is specialized in that it was written for a CDC 6600/Cyber 175 computer system using a 1200 baud line under the CDC telex time sharing system. Graphical implementations are provided by a Tektronics model 4015 terminal with the enhanced graphics option. A large Tektronics tablet may be employed in the digitization process along with a model 4631 hardcopy unit. The TIGS system could be modified for use with other graphics systems. Figure 2 shows a typical TIGS hardcopy plot.
FIGURE 1. INPUT AND OUTPUT AXIS REPRESENTATION
SOFTWARE OVERVIEW

The TIGS code is comprised of seven basic modules using the standard utility Tektronics release 3.2 software compiled under Fortran IV. TIGS uses the Cyber segmentation loader requiring about 40000 octal memory locations to execute. The segmentation setup consists of seven modules described below. The information flow between these modules is represented by Figure 3.

TIGS is the main executive module that controls the input and output and interplays with the graphics executive.

TABR contains the code to input and output the digital data in the required format.

TIGPPR is the graphics executive module. This module controls the graphical input and permits the user to interact with the graphical screen and digital tablet controlling data point values, plot sizes, curve options, titles and scaling.

The TIGPPR module performs these functions through connections to other segmentation modules GETVAL, LOPTIM, LABEL, and DRAWIT.

GETVAL is used to input data points either from the graphics screen or the digitizer tablet.

LOPTIM implements the axes scaling and grid options selected by the user.

LABEL uses the data values to compute the axes tic marks and other data related to fitting the plot on the graphical screen.

DRAWIT processes the scaling, axes, along with other plot data and generates the commands that draw the vectors on the graphic screen.

There are two basic operating modes in the graphics executive: creation and correction. In the creation mode a digital data file is created using the cross hair cursor either directly from the Tektronics screen or from the digitizer tablet. Commands from the screen are implemented by first positioning the cross hairs and then keying a single letter indicating the command. Commands from the digitizer tablet are implemented in two steps: first the command code letter is keyed using a tablet command menu; second, the coordinate position going with the command is keyed at the desired position. From either the screen or the tablet, the graphics executive receives the command and coordinate position. The commands received by the graphics executive are generally used in three different ways:

1) add, delete or change a coordinate point

2) change a graphics executive switch from off to on or on to off

3) control the size and view of the graphical plot.

Some commands available on the screen can not be used on the digitizer tablet. A more detailed discussion of these commands is found in the user's guide, Appendix A.

USER EXPERIENCE

The TIGS system has proved to be a very powerful, flexible tool. Task cost reductions of 10 to 1 have been shown to date using TIGS to prepare tabular inputs for other codes.
FIGURE 2. TIGS EXAMPLE PLOT
FIGURE 3. TIGS SEGMENTATION MODULES
CONCLUSIONS

An interactive graphical code system has been developed which is capable of rapid transformation of graphical information into tabular data formats which are compatible with the input requirements for a large variety of in-house programs.

REFERENCES

(a) Caddy Michael J., “TREAD/TLOOK - Multipurpose Computer Routine for Interpolation and Extrapolation of Tabular Data” NADC Report 76366-30, 1977
APPENDIX A
USER'S GUIDE
A.1 INPUT CONSIDERATIONS

The TIGS code will permit a direct creation of a data file from screen and or tablet commands. In addition, existing table data, input as file TAPE1, may be edited and corrected. In either case a new table data source with corrections is produced as an output on the TAPE7 file. The format of files TAPE1 and TAPE7 is the same. In the next section this format is illustrated.

A.2 TABLE DATA FORMAT

The table data may represent a dependent (output) parameter as a function of 1, 2, or 3 independent (input) parameters. The basic method for inputting these tables is described in reference (a). However, for the sake of completeness, the user's guide portion of reference (a) has been extracted and duplicated herein and includes those modifications introduced since its initial publication. This information is shown in Table A-1.

A.3 EXAMPLES

Card input data set-ups for three different examples are illustrated as follows:

Example 1 (Drag coefficient as a function of Mach number)
The dependent variable is drag coefficient and the independent variable is Mach number. Figure A-1 illustrates the graphical relationship. This is a one parameter table look-up so the other two parameters are dummies. Table A-1 shows the card set-up for this example. The EOT (end of table) parameter label terminates the data for this table.

Example 2 (Drag coefficient as a function of Mach number and lift coefficient)
The dependent variable is drag coefficient and the independent variables are Mach number and lift coefficient, illustrated in Figure A-2. This is a two parameter table look-up so that the third parameter is a dummy. Table A-2 shows the card set-up. In Table A-2 the last Mach parameter data repeats the previous Mach parameter data. In this situation, the last Mach parameter data card can be omitted. As a general rule, whenever the data on the X parameter axis as shown in Figure A-1, is repeated, then the X parameter data card need not be repeated.

Example 3 (Drag coefficient as a function of Mach number, lift coefficient and CG location)
The dependent variable is drag coefficient and independent variables are Mach number, lift coefficient, and CG location, illustrated in Figure A-3. Table A-3 shows the card set-up for this three parameter example. Note that the input card set-up is symmetrical in that each CL parameter data card begins data for each CG parameter.

A.4 LIMITATIONS

The TIGS system as presently written is limited to a maximum of 30 curves per plot, 150 points per curve, or a total of 300 points per plot. For example, a plot with 10 curves could be described with 5 curves using 40 points per curve, and the remaining 5 curves using 20 points per curve.

A.5 INTERACTIVE PROMPTING

The TIGS code has been designed to prompt the user in supplying information in the correct format. Selection of the baud rates compatible with available transmission lines is possible. After
### TABLE A-I
DATA INPUT INSTRUCTIONS

<table>
<thead>
<tr>
<th>Card No.</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1X,14,7A10</td>
</tr>
<tr>
<td>2</td>
<td>A4,13,3X,7F10.0</td>
</tr>
<tr>
<td>2a,b,etc.</td>
<td>10X,7F10.0</td>
</tr>
<tr>
<td>3 and following</td>
<td></td>
</tr>
</tbody>
</table>

All remaining cards have the same format as card 2, 2a, b, etc. The item which distinguishes the card types is the value of the independent variable. The 4 character identifiers of each independent variable must not be identical. The 4 characters of each independent variable card (after the title card) are user selected. The card order of each independent variable is significant. The first four cards with respective independent variables are as follows:

<table>
<thead>
<tr>
<th>Card</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>2,2a,b,etc.</td>
<td>third independent variable, identifier and values</td>
</tr>
<tr>
<td>3,3a,b,etc.</td>
<td>second independent variable, identifier and values</td>
</tr>
<tr>
<td>4,4a,b,etc.</td>
<td>first independent variable, identifier and values</td>
</tr>
<tr>
<td>5,5a,b,etc.</td>
<td>dependent variable, identifier and values</td>
</tr>
</tbody>
</table>

The remaining input cards use these same identifier values as input above. On cards 4, 4a, b, etc. and 5, 5a, b, etc. are the dependent and first independent variable values along the line given by the first value of the second independent variable and in the plane of the first value of the third independent variable.

Cards with the same respective identifier value cards 4, 4a, b, etc. and 5, 5a, b, etc. are repeated for different values of second independent variable until all second independent variables have been exhausted. The next card has an identifier corresponding to the second independent variable and new values of that variable for the plane of the second value of third independent variables. The values of the first independent variable need not be repeated if they are the same along each line of constant second independent variable. In each instance where the values are changed a new card is required.

Last Table input termination indicator, EOT

All remaining tables for this input section follow the same pattern as above. To end the table read-in mode, a blank table reference number is input behind the last table of the entire table set.
FIGURE A-1. ONE PARAMETER TABLE LOOK-UP
TABLE A-II. ONE PARAMETER CARD INPUTS

<table>
<thead>
<tr>
<th>COLUMN LOCATION</th>
<th>123456789012345678901234567890123456789012345678901234567890</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>DRAG</td>
</tr>
<tr>
<td>Z</td>
<td>1</td>
</tr>
<tr>
<td>Y</td>
<td>1</td>
</tr>
<tr>
<td>MACH</td>
<td>4</td>
</tr>
<tr>
<td>CD</td>
<td>0.010</td>
</tr>
<tr>
<td>EOT</td>
<td>0.011</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
FIGURE A-2. TWO PARAMETER TABLE LOOK-UP
### TABLE A-III. TWO PARAMETER CARD INPUTS

<table>
<thead>
<tr>
<th>COLUMN LOCATION</th>
<th>123456789012345678901234567890123456789012345678901234567890</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z 1</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CL 3</td>
<td>0.0</td>
<td>0.2</td>
<td>0.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MACH 4</td>
<td>0.0</td>
<td>0.1</td>
<td>0.2</td>
<td>0.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CD 4</td>
<td>0.01</td>
<td>0.02</td>
<td>0.03</td>
<td></td>
<td>0.04</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MACH 3</td>
<td>0.0</td>
<td>0.15</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CD 3</td>
<td>0.1</td>
<td>0.02</td>
<td>0.03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MACH 3</td>
<td>0.0</td>
<td>0.15</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CD 3</td>
<td>0.02</td>
<td>0.03</td>
<td>0.04</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EOT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
FIGURE A-3. THREE PARAMETER TABLE LOOK-UP
### TABLE A-IV. THREE PARAMETER CARD INPUTS

**COLUMN LOCATION**

12345678901234567890123456789012345678901234567890

<table>
<thead>
<tr>
<th>1226</th>
<th>CD VS M, CL, AND CG</th>
</tr>
</thead>
<tbody>
<tr>
<td>CG 3</td>
<td>0.0</td>
</tr>
<tr>
<td>CL 3</td>
<td>0.0</td>
</tr>
<tr>
<td>MACH 4</td>
<td>0.0</td>
</tr>
<tr>
<td>CD 4</td>
<td>0.01</td>
</tr>
<tr>
<td>CD 4</td>
<td>0.02</td>
</tr>
<tr>
<td>MACH 3</td>
<td>0.0</td>
</tr>
<tr>
<td>CD 3</td>
<td>0.03</td>
</tr>
<tr>
<td>CL 4</td>
<td>0.0</td>
</tr>
<tr>
<td>MACH 3</td>
<td>0.0</td>
</tr>
<tr>
<td>CD 3</td>
<td>0.011</td>
</tr>
<tr>
<td>CD 3</td>
<td>0.015</td>
</tr>
<tr>
<td>CD 3</td>
<td>0.020</td>
</tr>
<tr>
<td>CD 3</td>
<td>0.025</td>
</tr>
<tr>
<td>CL 3</td>
<td>0.2</td>
</tr>
<tr>
<td>MACH 4</td>
<td>0.0</td>
</tr>
<tr>
<td>CD 4</td>
<td>0.01</td>
</tr>
<tr>
<td>MACH 3</td>
<td>0.0</td>
</tr>
<tr>
<td>CD 3</td>
<td>0.011</td>
</tr>
<tr>
<td>CD 3</td>
<td>0.021</td>
</tr>
</tbody>
</table>

EOT
logging into the host system and the baud rate has been selected, different prompts will appear depending on the user response to the initial interactive query. The response will depend on whether the user intends to correct an existing file or create a new file via the screen or tablet. After the baud rate selection the next query to appear will be:

(a) "IS THIS A CREATION RUN?"

A "Y" response indicates a TAPE1 file is to be newly created and the following prompts will appear.

(b) "ENTER TABLE TITLE CARD
(COLUMNS 1-5 SHOULD BE TABLE REFERENCE NUMBER)"

The user should refer to the instructions in section A-2 Table A-1, card 1.

(c) "ENTER 4 CHARACTERS FOR EACH LABEL FOR Z, Y, X, FXYZ (separated by commas)"

The user should refer to instructions in section A.2 Table A-1, card 2.

(d) "ENTER NUMBER OF Z VALUES"

The user should refer to instructions in section A.2 Table A-1, card 2.

(e) "ENTER Z VALUES IN ASCENDING ORDER"

The user should now enter the values of the Z parameter with blanks or commas between the data pieces.

(f) "WANT TO SPECIFY DECIMAL PLACES ON TAPE2?"

An "N" response will by default, set the number of places at the maximum allowable. If the user enters a "Y", this will be prompt query (g).

(g) "ENTER NUMBER OF DECIMAL PLACES FOR Z,Y,X,FXYZ"

The user should specify the number of decimal places (up to 9) separated by blanks or commas for Z,Y,X,FXYZ parameters.

At this point, the following message will appear:

"NO DATA TO BE FOUND...ENTER COMMAND"

The user may now begin creating the tabular data set with either a "N" (new line) command or a "T" (tablet operation) command. The reader is referred to sections A.5 and A.6 for additional information.

If the response to the initial query (query (a) above) is "N" then this means that data on the TAPE1 is to be used and queries (b) thru (e) are skipped.

A.6 TABLET INITIATION PROCEDURE

Tablet commands are issued using a command menu. The command menu is a section of the tablet, 20 one inch squares (10 columns by 2 rows) in which keying the coordinates within a
square is interpreted as the indicated command. Figure A-4 shows the positions of the menu commands within the 10 by 2 inch squares.

Tablet operation begins initially by attaching the command menu at any convenient location on the tablet. The menu should be approximately parallel with the lower edge of the tablet. Upon first entering the “T” command, the user enters the position of the upper left corner of the menu.

The coordinates of this position are then used to determine the commands. Next, the user attaches the graph that is to be used at a convenient tablet location. Squaring the graph is not necessary since any angular correction required is performed in TIGS. The user then may select any convenient orthogonal axes and enters the following requested information:

a) the coordinate position of the crossing point of the orthogonal axes, and the coordinate values X and FXYZ respectively at the crossing point.

b) the coordinate position of any X axis point and its value. (usually this coordinate position is the maximum axis length)

After these entries have been made the “NO DATA FOUND TO PLOT” message will appear. At this time the user may issue commands from the tablet menu.

A.7 TIGS INTERACTIVE COMMANDS

Commands from the screen involve only positioning the cross hairs and keying the appropriate command. Commands from the tablet involve first selecting the command from the menu and then indicating the coordinate position. Once a tablet command has been set it remains set until changed. The user is free to change to and from tablet and screen command modes. The following commands are available:

“A”- add point after. The user positions the cross hairs and keys the “A” command (or indicates the tablet command and position). The system will respond by drawing the symbol at the new point. (Note. See the “C” command for further discussion.)

“B”- add point before. This command is exactly like the “A” command except that the point is added before the pointer position.

“C”- position the pointer to the array location that the user wishes to add a new point. The next command following the “C” command to add a point may be an “A” to add after or a “B” to add before the pointer position. In addition an “M” command may be used to move to a new location the point indicated by the position pointer. A “V” command may also be used. It should be noted that the pointer position after each added point becomes the position of the added point. Possible valid commands would be “CAABBAVVVA” permitting the user to continuously add new points very rapidly. Any other command drops the pointer position, which must be restored by another “C” command to add new points.

“D”- delete the point closest to the cross hairs or pen position.

“E”- end or terminate this plot and return to TIGS for next plot if any.

“F”- format or change type of curve drawn for each line as follows:
FIGURE A-4. TABLET COMMAND MENU
ITIP- switch determining type of curve drawn (ITIP=2, default) ITIP can have the following values: (Note: a negative value will have the same meaning except no symbols are drawn.)

0 indicates symbols only, no curve drawn

1 indicates linear fit

2 indicates smooth spline like fit with respect to x axis

3 same as ITIP=2 except with respect to y axis

4 indicates data is multivalued and the fit is with respect to arc length along curve

5 indicates data is multivalued and forms a closed figure; the fit is with respect to arc length and joined at the ends.

NOTE: if data is not in ascending order when ITIP = 2 or ITIP = 3, then curve fit will default to ITIP = 4; DEFAULT format is ITIP = 2.

"G"- IGRID switch- turns grid from on to off or off to on.

"H"- halt tablet and returns control to screen. This only applies to tablet modes.

"I"- initialize tablet starting with graph coordinate locations. This only applies to tablet mode.

"M"- move the point indicated by the pointer position to the new coordinates indicated by the cross hairs.

"N"- begin a new curve at point indicated. Whenever this command is issued, the user will also enter the new curve value. The pointer position becomes the new point permitting commands such as "NAAMAAAAABBVAB".

"P"- re-plot data with scale as shown. see Note

"R"- rescale data to the largest size and re-plot. see Note

"S"- show the current coordinate values at the position indicated.

"V"- values input; same as "A" or "B" command except the actual X and F coordinate values are input.

"W"- window the plot to fit within the rectangle indicated by the diagonal between two coordinate positions (two points are sent).

Note: The commands "F", "P" and "R" have a dual meaning. If the vertical cross hair is to the left of the vertical plot axis then:

"F" indicates to change the ITIP of the curve indicated by the position of the horizontal cross hairs. "P" indicates activate TIGS to plot the curve indicated by the position of the horizontal cross hairs. This action is cumulative in that one or more of many curves may be indicated in this manner. "R" is used to deactivate this special mode and thus restore all curves to plotting status.
A.8 EXAMPLE INTERACTIVE SESSION

In this example the TIGS system is used to create a plot. The figures in this example are actual copies of what the user would see on the Tektronix screen. In these figures a "?" followed by data indicates that these data were the user's response to the indicated query. In the following discussion numbers enclosed by circles refer to corresponding numbers on a figure pointing to a feature under discussion. In figure A-5 ① is the command used to begin execution of TIGS. ② is the user response to the query as to transmission line rate. ③ indicates that a creation is requested and results in queries ④ thru ⑦. ④ is the main title on the plot preceded by the table reference number. The table reference number should be a 5 digit integer number. The title can be up to 4 lines. The user may indicate a new line by leaving three consecutive blanks between words. ⑤ is the response to the query requesting four variable names for the respective data. Note that each variable name must be 4 characters in length; blanks count as characters. ⑥ is the response to the number of Z variables requested. Each Z value represents a single plane. ⑦ is the response to input each Z value. ⑧ is the response to the decimal place query related to the TAPE7 file. This file is an output file containing all of the data generated during this session. Each prompt, as shown, indicates that the data on TAPE7 will contain the maximum decimal places that will fit with each space. Some caution is necessary if the user specifies the number of decimal places for each parameter; precision could be lost if a low number of decimal places is initially selected. A good technique is to examine the TAPE7 file with the maximum decimal places specified first and then re-enter TIGS, if necessary, and specify decimal places as required.

When the user responds to ⑧, figure A-6 will be displayed. The meaning of figure A-6 is that a plot command as implied and that data was not found to plot. This is a proper response since the user, through ③, on figure A-5, elected a creation run and there is no data as yet to plot. The user will notice for the first time that cross hairs also have appeared on the screen. (Note these are not shown in figure A-6). The cross hair is a prompt signal that an input is requested. The input is a single upper case letter. A "RETURN" is not required after typing the single letter command. The single letter command issued in this example was a "N" indicating a new line. The response to this command shown in figure A-7. The first prompt, ⑨, requests one set of data coordinates, X and FXYZ, for one point. The purpose of this is to scale the final plot. The response at ⑩ is a value assigned to this line (this one set of coordinates is the beginning of a potential curve).

After entering the number one for this query as noted by ⑩, the screen will appear as shown in figure A-8. The "Y" shown on the left top of this figure is the 4 character label entered in ⑤ figure A-5. The "A" and number under the "Y" is the symbol for the first line and the line value assigned to it as, ⑩ in figure 7. The "A" in the center of the plot at (0,0) is the first point (and only point) of line A. The pointer positioned message indicates that the reference point from which to add points has been identified. This occurred automatically since only one point at this time is in the plot, the first point. All of the other 4 character labels, including the main plot label, are also shown. The value of the plot plane (Z value) is zero and is shown at the top right.

In the next steps the user has moved the cross hairs and "keyed" the "A" characters indicating "add point after". The "add point after" in this context means that the data point storage of the new point is after the point indicated by pointer position. The curves are always drawn in the order towards the "after" point. After each point is added the pointer position becomes the position of the added point. Figure A-9 shows the addition of added four points as they would appear on the screen. Figure A-10 is a replot of the data resulting from the user keying a "P". This command simply plots a curve through the data points shown.

Figure A-11 is a resize and replot resulting from an "R" command. The plot axes have been rescaled to permit the largest plot of the data points that will fit within the screen.
FIGURE A-5. EXAMPLE PLOT
NO DATA FOUND TO PLOT ... ENTER COMMAND

FIGURE A-6. EXAMPLE PLOT

A-16
NO DATA FOUND TO PLOT ..ENTER COMMAND

Input X,Y

? 8 8

Input Line Value

? 1

FIGURE A-7. EXAMPLE PLOT
TEST EXAMPLE CREATION

FIGURE A-8. EXAMPLE PLOT
FIGURE A-10. EXAMPLE PLOT
FIGURE A-11. EXAMPLE PLOT
In figure A-12 the cursor was first positioned at approximate co-ordinate locations of .15 and .16 and a “C” command issued. The “C” indicates to identify the closest point to the intersection of the cross hairs as pointer position. This command also resulted in the message stating “pointer positioned” at the top left. The four points shown on figure 12 were then added by the user moving the cursor and “Keying” the “A” command. In figure A-13 a “P” command was issued first and then the cross hairs were at the position indicated by the “B” symbol and the “N” command was keyed. This resulted in the “input line value” query shown in the top left of figure A-13. In figure A-14 the query response is shown and the user has inputted more points by just moving the cursor and using the “A” command.

In figure A-15 the user has replotted the data with a “P” command and then the cursor was positioned near the end “B” point at X=.14 and a “C” command was keyed. The next command sent by the user was a “V”, to input an exact value. This prompted the query “Input X,Y” to appear. The last query “A or B mode?” simply request that the user identify where in the data storage is the new data point stored, before the pointer or after the pointer.

Figure A-16 is a final plot of the data showing the new point. At this point the user keyed an “E” command and “ended” the execution. In figure A-17, the output file created during this example is listed using the CED text editor, showing all the data points.
FIGURE A-12. EXAMPLE PLOT
TEST EXAMPLE CREATION

FIGURE A-13. EXAMPLE PLOT
FIGURE A-14. EXAMPLE PLOT

A-25
TEST EXAMPLE CREATION

FIGURE A-16. EXAMPLE PLOT
END TIGS
.737 CP SECONDS EXECUTION TIME
/CED_TAPE
CED 1.2
\? Pa
00081 TEST EXAMPLE CREATION
Z 1 8.
X 0 1.0000000 2.0000000
  8.0000000 8.0000000
  8.0000000 8.0000000
FXYZ 0 8.0000000 8.0000000
  8.0000000 8.0000000
  8.0000000 8.0000000
  8.0000000 8.0000000
X 0 1200000.0 1600000.0
  1600000.0 1600000.0
  1600000.0 1600000.0
  1600000.0 1600000.0
FXYZ 0 1200000.0 1600000.0
  1600000.0 1600000.0
  1600000.0 1600000.0
  1600000.0 1600000.0
LOT
--EDR--
--EDF--
END OF INFORMATION
\? S
ABORTED

FIGURE A-17. EXAMPLE PLOT
APPENDIX B
FORTRAN LISTING
CTIGS

PROGRAM TIGS(INPUT=101, OUTPUT, TAPE1=101, TAPE7=101, TAPE5=INPUT) 
C ********************************************
C***** TIGS TPLOY INTERACTIVE GRAPHICS SYSTEM
C*****
C***** M CADDY JAN 30 78
DIMENSION LT(7), XV(30), NPTS(30), X(300), Y(300), Z(30), A(99)
DATA NPLT/0/
DATA NT/8/
DATA ITIP, IGRID/2, 1 /
DATA NPTS, XV/30*0, 30*0./
DATA IEND/10HEOT /
10 FORMAT(A5,7A10)
REWIND 1
REWIND 7
PRINT 20
20 FORMAT(* TIGS VER 2.0 8/2/78 *,
1 /* IF THIS IS A CREATION RUN ENTER Y*)
READ 30, IC
30 FORMAT(1R1)
IC=IC-30B
IF(IC.EQ.1) GO TO 40
C**** FILE IS NOT BEING CREATED READ IT FROM TAPE1
C****
31 READ (1,10) LNO,LT
IF(LNO.EQ.10H ) GO TO 251
C****
C***** CALL IN Z VALUES
C*****
CALL TABR(LZ,NZ,Z,1)
C*****
CALL TABR(LY,NY,A,1)
CALL TABR(LX,N,X,1)
CALL TABR(LF,N,Y,1)
GO TO 100
40 PRINT 50
50 FORMAT(* ENTER TABLE TITLE CARD*/,
1 *(COLUMNS 1-5 SHOULD BE THE TABLE REFERENCE NUMBER]*)
READ 10, LNO,LT
C**** READ TABLE NUMBER AND TITLE
C****
PRINT 60
60 FORMAT(* ENTER 4 CHARACTERS FOR EACH LABEL FOR Z,Y,X,FXYZ*/
1  * (SEPARATED BY COMMAS)*)
C**** READ TITLES FOR EACH VARIABLE 4 CHARACTERS LONG
C**** READ 70,LZ,LY,LX,LF
   70 FORMAT(4(A4,1X))
C**** GET NUMBER OF Z VARIABLES AND VALUES
C**** PRINT 80,LZ
   80 FORMAT(* ENTER NUMBER OF *,A4,* VARIABLES---FREE FORM*)
   CALL GETIN(1,Z)
   NZ=Z(1)
   PRINT 90,LZ
   90 FORMAT(* ENTER *,A4,* VALUES, ASCENDING ORDER---FREE FORM*)
   CALL GETIN(NZ,Z)
C**** WRITE TO TAPE7 TITLE CARD AND TABLE NUMBER
C**** 100 WRITE(7,10) LNO,LT
   PRINT 110,LZ,LY,LX,LF
   110 FORMAT(* ENTER NUMBER OF DECIMAL PLACES FOR *,4(A4,1X)
   1,* FREE FORM*)
C**** GET NUMBER OF DECIMAL PLACES FOR EACH VARIABLE
C**** CALL GETIN(4,XV)
   LZDP=XV(1)
   LYDP=XV(2)
   LXDP=XV(3)
   LFDP=XV(4)
C**** WRITE TO TAPE7 THE Z VALUES ETC...
C**** CALL TFORM(1,LZ,NZ,Z,LZDP,7)
C**** INITIALIZE TEK SOFTWARE
C**** CALL INITT(120)
   CALL TERM(3,4096)
   CALL CHR5IZ(4)
   DO 250 IX=1,NZ
C**** IF CREATION MODE THEN SET DEFAULTS TO 0
C**** IF(IC.NE.1) GO TO 120
   NPTS(1)=0
   X(1)=0.
Y(1)=0.
GO TO 210

C*****
C***** NON CREATION MODE
C*****
120 CONTINUE
K=1

C***** TRANSFER SECOND INDEPENDENT VARIABLE TO XV ARRAY
C*****
DO 130 J=1, NY
130 XV(J)=A(J)
IF(IZ.EQ.1) GO TO 140
CALL TABR(LX,N,X,1)
CALL TABR(LF,N,Y,1)

140 LNX=N
LNY=N
NPTS(1)=N
NPTS(2)=0

C***** READ NEXT SET
C*****
150 CALL TABR(LW,N,A,1)

C***** CHECK FOR NEXT Z GROUP
C*****
IF(LW.EQ.LY) GO TO 210

C***** CHECK FOR END OF TABLE
C*****
IF(LW.EQ.4HEOT) GO TO 210

C***** CHECK FOR NEXT X DATA
C*****
IF(LW.NE.LX) GO TO 170

C***** DATA IS X DATA STORE IT
C*****
LOX=LNX
DO 160 J=1,N
LNX=LNX+1
160 X(LNX)=A(J)
GO TO 150

C***** DATA HAD BETTER BE LY
C*****
170 IF(LW.NE.LF) STOP
C*****
C**** IF DATA HAS NOT BE INPUT FOR X DATA USE LAST VALUES
C****
IF(LNX.GT.LNY) GO TO 190
LL=LOX
DO 180 J=1,N
LNX=LNX+1
LL=LL+1
180 X(LNX)=X(LL)
C****
C**** UPDATE COUNTERS
C****
190 K=K+1
NPTS(K)=N
NPTS(K+1)=0
C****
C**** LOAD Y DATA
C****
DO 200 J=1,N
LNY=LNY+1
200 Y(LNY)=A(J)
C****
C**** GO BACK TO GET NEXT GROUP
C****
GO TO 150
C****
C**** PLOT DATA
C****
210 CALL TIGPPR(NPLOT,LF,1,LX,1,LT,8,X,Y,NPTS,LY,1,XV,LYDP,ITIP,
1 IGRID,LZ,Z(IZ))
CALL ANMODE
C****
C**** COUNT NUMBER OF Y VALUES
C****
NY=0
DO 220 I=1,30
IF(NPTS(I).EQ.0) GO TO 230
NY=NY+1
220 CONTINUE
GO TO 250
C****
C**** WRITE TO TAPE7 Y DATA ETC....
C****
230 CALL TFORM(1,LY,NY,XV,LYDP,7)
LOC=1
J=0
240 J=J+1
NP=NPTS(J)
IF(NP.EQ.0) GO TO 250
WRITE TO TAPE7 X DATA ETC...
CALL TFORM(LOC,LX,NP,X,LXDP,7)
WRITE TO TAPE7 Y DATA ETC...
CALL TFORM(LOC,LF,NP,Y,LFDP,7)
LOC=LOC+NP
GO TO 240
250 CONTINUE
WRITE (7,10) IEND
IF (IC.NE.1) GO TO 31
251 WRITE (7,10)
REWIND 7
END

SUBROUTINE TFORM(LOC,LAB,N,X,IP,K)
DIMENSION X(1), IFORM(3)
FORMATTING SUBROUTINE FOR TPLT FORMAT
LOC IS THE LOCAL ARRAY POSITION TO PRINT FROM
LAB IS THE 4 CHARACTER LABEL
N IS THE NUMBER TO PRINT
X IS THE ARRAY CONTAINING THE VALUES
IP IS THE NUMBER OF DECIMAL PLACES TO USE IN FORMAT
IF(IP.LT.0) IP=0
IF(IP.GT.9) IP=9
JO=LOC-1
NP=N
IF(NP.GT.7) NP=7
IFORM(1)=10H(A4,I3,3X,
IFORM(2)=5555555420634335733B+IP
IFORM(3)=10H)
WRITE(K,IFORM) LAB,N,(X(I+JO),I=1,NP)
IF(N.GT.7) WRITE(K,IFORM)(X(I+JO),I=1,N)
RETURN
END

SUBROUTINE TABR(LAB,N,A,K)
DIMENSION A(1)
READ(K,10) LAB,N,(A(I),I=1,7)
10 FORMAT(A4,I3,3X,7F10.0)
IF(N.GT.7) READ(K,20) (A(I),I=8,N)
20 FORMAT(10X,7F10.0)
30 ISUB(I)=I+1
C
C    MERGE HERE TO REPLOT
C
40 CALL BINITT
   LCNT=3120
   IGRID1=(3*IGRID+7)*.5
C    SUM UP NUMBER OF POINTS
   NL=0
   NPTOT=0
   DO 60 I=1,30
   N=NPTA(I)
   IF(N.EQ.0) GO TO 70
   NL=NL+1
   60 NPTOT=NPTOT+N
C    SET STORAGE LIMIT TO NPTOT FIRST PASS
70 IF(NSTOR.EQ.0) NSTOR=NPTOT
   IF(NPTOT.GT.0)GO TO 90
   NSTOR=0
   CALL MOVABS(0,LCNT)
   CALL ANMODE
   PRINT 80
   80 FORMAT(* NO DATA FOUND TO PLOT ..ENTER COMMAND*)
   LCNT=LCNT-LDEL
   IPLOT=0
   GO TO 200
   C
C
C    SECOND INDEPENDENT VARIABLE TITLE
C
90 IF(NCC.LE.0)GO TO 140
   CALL MOVABS(0,LCNT)
   CALL ANMODE
   PRINT 110,(LABVAL(J1),J1=1,NCC)
   CALL MOVABS(2800,2800)
   CALL ANMODE
   PRINT 100,LZ,ZVAL
100 FORMAT(A4,=*,G13.5)
110 FORMAT(8A10)
   LCNT=LCNT-LDEL
   KL=0
   KH=55B
   DO 130 J1=1,NL
   LCNT=LCNT-LDEL
   CALL MOVABS(0,LCNT)
   KL=KL+1
   CALL ANMODE
   PRINT 120,KH,KL,VLABL(J1)
RETURN
END

C

TEK INTERACTIVE GPPR  M CADDY FEB 78

C

SUBROUTINE TIGPR(NPLOT,LABY,N1,LABX,N2,LABTL,NT,X,Y,
1 NPTA,LABVAL,NCC,VLABEL,NDECVIN,ITIP,IGRID,LZ,ZVAL)
COMMON/TKTRNX/ITEKC(60)
DIMENSION X(200),Y(200),LABTL(9),NPTA(30),VLABEL(30),
1 LABX(5),LABY(5),VTEM(8),LABVAL(8),IQUICK(30),ISUB(300)
DIMENSION MSG1(20),MSG2(20),MSG4(10),MSG5(10),MSG6(15),IALTM(6,2)
EQUIVALENCE (BEG(1),XBEG),(BEG(2),YBEG)
EQUIVALENCE (DEL(1),DELX),(DEL(2),DELY),(ITAB,LTV(2))
EQUIVALENCE (EN(1),XEND),(EN(2),YEND)
EQUIVALENCE (IOFF,ITEKC(30)),(TXMIN,ITEKC(1))

C SET LINE SPACING
COMMON/TEKPPR/LDEL,LCNT,MAXSR,LTV(17),EN(2),DEL(2),BEG(2),RDX2,
1RDY2,NLINE,NDRAW(30),MODE(30)
DATA MSG1/46,q46,80,111,105,110,116,101,114,32,
1 80,111,115,105,110,116,101,110,101/100
DATA MSG2/73,110,112,117,116,32,76,105,110,101,
1 32,86,97,108,117,101,32,32,32/2

C**** ILLEGAL MESSAGE
DATA MSG5/73,110,112,117,116,32,88,44,89,32/
DATA MSG6/65,32,111,114,32,66,32,109,111,100,
1 101,63,32,32,32/
DATA ((IALTM(I,J),I=1,6),J=1,2)/65,102,116,101,114,32,
1 66,101,102,111,114,101/
DATA IQUICK/0,0,1,1,2,3,10,4,0,0,
1 0,0,0,5,0,6,7,8,0,
2 0,0,9,0,0,0,0,0,0/1
A=1. //KIN(1.)
LDEL=50
IF(NPLOT.GT.0) GO TO 20
DO 10 I=1,8
10 LTV(I)=0
20 IWIN=0
NPLT=NPLT+1
1GRID=1
DO 21 I=1,30
21 MODE(I)=ITIP
NTL=NT
NLINE=0
NSTOR=0

C SET STORAGE POINTER TO INITIAL SEQUENCE
DO 30 I=1,299
120 FORMAT(1X,2R1,G13.5)          TIGP0100
130 CONTINUE                     TIGP0101

C PREPARE TEKTRONIX AGII COMMON  TIGP0102
C
140 CONTINUE                     TIGP0103
   I PLOT=I PLOT+1
   CALL CHRSIZ(4)                 TIGP0104
   CALL SLIMX(640,4000)           TIGP0105
   CALL SLIMY(300,2700)           TIGP0106
   CALL T ICK SIZES               TIGP0107
   CALL XTICS(14)                 TIGP0108
   CALL YTICS(10)                 TIGP0109
   IF(IWIN.NE.0) GO TO 170        TIGP0110
   AXMAX=-1.E99                   TIGP0111
   AYMAX=-1.E99                   TIGP0112
   AXMIN=+1.E99                   TIGP0113
   AYMIN=+1.E99                   TIGP0114

C SET MIN AND MAX DATA VALUES    TIGP0115
   K=1                           TIGP0116
   DO 150 I=1,NPTOT                TIGP0117
      AXMIN=AMIN1(AXMIN,X(K))      TIGP0118
      AYMIN=AMIN1(AYMIN,Y(K))      TIGP0119
      AXMAX=AMAX1(AXMAX,X(K))      TIGP0120
      AYMAX=AMAX1(AYMAX,Y(K))      TIGP0121
   KLAST=K                        TIGP0122

C SET KLAST TO END STORAGE VALUE TIGP0123
   150 K=ISUB(K)                  TIGP0124
   IF(AXMIN.NE.AXMAX) GO TO 160   TIGP0125
   AXMIN=AXMIN-.5                 TIGP0126
   AXMAX=AXMAX+.5                 TIGP0127
   IF(AYMIN.NE.AYMAX) GO TO 170   TIGP0128
   AYMIN=AYMIN-.5                 TIGP0129
   AYMAX=AYMAX+.5                 TIGP0130

C SET VIRTUAL WINDOW             TIGP0131
   170 CALL DLIMX(AXMIN,AXMAX)    TIGP0132
   CALL DLIMY(AYMIN,AYMAX)        TIGP0133
   CALL XLEN(28)                  TIGP0134
   CALL YLEN(28)                  TIGP0135
   CALL XFRM(IGRID1)              TIGP0136
   CALL YFRM(IGRID1)              TIGP0137
   NBASE=IBASEX(0)                TIGP0138
   DO 180 I=1,2                    TIGP0139
   CALL LOPTIM(NBASE)             TIGP0140
   CALL WIDTH(NBASE)              TIGP0141
   CALL SPREAD(NBASE)             TIGP0142
B-9
CALL TSET(NBASE)
180 NBASE=IBASEY(0)
   EN(1)=COMGET(IBASEX(27))
   EN(2)=COMGET(IBASEY(27))
   BEG(1)=COMGET(IBASEX(26))
   BEG(2)=COMGET(IBASEY(26))
   DELX=(XEND-XBEG)/3360.
   DELY=(YEND-YBEG)/2400.

C FIND VIRTUAL SPACE TO SCREEN SPACE SCALING PARAMETERS
C
   RDX2=1./(DELX*DELY)
   RDX2=1./(DELY*DELY)
   CALL SETWIN
   CALL GRID
   CALL LABEL(IBASEY(0))
   CALL LABEL(IBASEX(0))
   CALL DRAWIT(NL,NPTA,X,Y,ISUB)

C AXIS LABELS
C
   CALL CHR5IZ(3)
   CALL TITLE(2320,3000,NTL,LABTL,80,0)
   CALL TITLE(2320,100,N2,LABX,80,0)
   CALL TITLE(450,1500,N1,LABY,80,1)

C MERGE HERE FOR INTERACTIVE FUNCTIONS (BELL)
C
200 IF(LCNT.LT.220) GO TO 530
   CALL CHR5IZ(4)
   IF(NPTOT.EQ.1) GO TO 240
   CALL GETVAL(ICHAR,XO,YO)
210 IF(ICHAR.LE.64.OR.ICHAR.GE.95)GO TO 220
   ICHAR=ICHAR-64
   ICHECK=IQIQUICK(ICHAR)
   IF(ICHECK.EQ.0) GO TO 220
   GO TO (300,400,440,200,460,500,540,560,590,455),ICHECK
220 LCNT=LCNT-LDEL
   CALL NOTATE(0,LCNT,10,MSG4)
   GO TO 200

C ADD POINT AFTER OR BEFORE SPECIFIED POINT (A OR B)
C
C CHECK IF C COMMAND AND FIRST POINT.
C
230 IF(NPTOT.EQ.0)GO TO 460
240 LCNT=LCNT-LDEL
   CALL NOTATE(0,LCNT,20,MSG1)
250 CALL GETVAL(ICHAR,XO,YO)
C CHECK FOR NEW LINE COMMAND
   IF(ICHAR.EQ.86) GO TO 580
C CHECK FOR ADD AFTER
260 IF(ICHAR.EQ.65) GO TO 270
C CHECK FOR MOVE
   IF(ICHAR.EQ.77) GO TO 270
   IF(NOT A B OR M GO TO NEW COMMAND
   IF(ICHAR.NE.66) GO TO 210
270 CALL POINTA(XO,YO)
   CALL MOVEA(XO,YO)
   IF(IOFF.EQ.0)CALL ANCHO(IS)
   IF(ICHAR.EQ.77) GO TO 290
   NPTOT=NPTOT+1
C INCREMENT STORAGE COUNTER
   NSTOR=NSTOR+1
   NPTA(ISAVE)=NPTA(ISAVE)+1
C MOVE POINT OF CLOSEST POINT TO END
   ISUB(NSTOR)=ISUB(JSAVE)
C CHANGE CLOSEST POINTER TO ACCESS LAST POINT
   ISUB(JSAVE)=NSTOR
   IF(NPTOT.EQ.65) GO TO 280
C MOVE OLD POINT TO LAST POINT ( INSERT BEFORE)
   X(NSTOR)=X(JSAVE)
   Y(NSTOR)=Y(JSAVE)
   GO TO 290
C NEW POINT ADD AFTER
280 IF(KLAST.EQ.JSAVE) KLAST=NSTOR
   JSAVE=NSTOR
290 X(JSAVE)=X0
   Y(JSAVE)=Y0
   GO TO 250
C DELETE POINT (D)

300 DSAVE=1.40
   IF(NPTOT.EQ.0) GO TO 200
   IS=64
   NSUM=1
   K=1
   DO 340 I=1,NSUM
   NEND=NSUM+NPTA(I)-1
   DO 330 J=NSUM,NEND
      IF(NLINE.EQ.0) GO TO 310
      IF(NDRAW(I).EQ.0) GO TO 320
310 XDX=X(K)-X0
   YDY=Y(K)-Y0
   DIST=XDX*XD*XRD+YDY*YDY*RDY2
   CONTINUE
330 CONTINUE
340 CONTINUE
IF(DIST.GE.DSAVE)GO TO 320
DSAVE=DIST
JSAVE=K
ISAVE=I
320 K=ISUB(K)
330 NSUM=NEND+1
IS=ISAVE+64
340 IF(IS.LE.90)GO TO 360
IS=IS-90
GO TO 350
350 IF(IOFF.EQ.0)CALL ANCHO(IS)
IF(ICHAR.NE.4) GO TO 230
NPTOT=NPTOT-1
K=JSAVE
C  IF DELETED POINT IS LAST ONE SKIP SHIFT
IF(KLAST.EQ.JSAVE)GO TO 370
C  GET POINTER OF NEXT POINT
K=ISUB(JSAVE)
C  TRANSFER POINTER OF NEXT POINT TO DELETED POINT
ISUB(JSAVE)=ISUB(K)
C  MOVE VALUE OF NEXT POINT TO DELETED POINT
X(JSAVE)=X(K)
Y(JSAVE)=Y(K)
370 IF(NPTOT.EQ.1)NSTOR=1
C  ZERO DELETED POINTER
ISUB(K)=0
NPTA(ISAVE)=NPTA(ISAVE)-1
IF(NPTA(ISAVE).GT.0)GO TO 390
NPTA(ISAVE)=0
J=0
DO 380 I=1,NL
IF(I.EQ.ISAVE)GO TO 380
J=J+1
NPTA(J)=NPTA(I)
VLABL(J)=VLABL(I)
380 CONTINUE
NPTA(NzL)=0
NL=NL-1
390 GO TO 200
C  END (E)
C
400 CALL NEWPAG
L=1
DO 431 I=2,NPTOT
K=ISUB(L)
IF(I.EQ.K) GO TO 431
J=K
JLEFT=NPTOT+1-I
DO 420 KK=1,JLEFT
IF(J.EQ.I) GO TO 430
JO=J

420 J=ISUB(JO)
430 ISUB(JO)=K
ISUB(L)=I
IS=ISUB(I)
ISUB(I)=ISUB(K)
ISUB(K)=IS
XS=X(I)
X(I)=X(K)
X(K)=XS
XS=Y(I)
Y(I)=Y(K)
Y(K)=XS

431 L=I
RETURN

C
FORMAT (F)
C
440 LCNT=LCNT-LDEL
CALL MOVABS(0,LCNT)
CALL ANMODE
IY=(YO-YBEG)/DELY+300
II=(3045-IY)/50+1
IF(II.LT.1) II=1
IF(II.GT.NL) II=NL
PRINT 450,MODE(II)
450 FORMAT(* ITIP = *,I2)
CALL GETIN(1,VTEM)
MODE(II)=VTEM(1)
LCNT=LCNT-LDEL
C
IF F OUTSIDE OF AXIS THE SET ALL CURVE MODES
C
IF(X0.LE.TXMIN) GO TO 200
DO 451 I=1,30
451 MODE(I)=VTEM(1)
GO TO 200
C
CHANGE GRID OPTION
C
455 IGRID=-IGRID
GO TO 200
C
C NEW LINE (N) TIGP0319
C 460 NPTOT=NPTOT+1 TIGP0320
460 NSTOR=NSTOR+1 TIGP0321
460 IF(NPTOT.NE.1) ISUB(KLAST)=NSTOR TIGP0322
460 KLAST=NSTOR TIGP0323
460 IF(IPLLOT+ITAB.EQ.0) GO TO 580 TIGP0324
461 X(NSTOR)=XO TIGP0325
461 Y(NSTOR)=YO TIGP0326
461 NL=NL+1 TIGP0327
461 IS=NL+64 TIGP0328
C NO DATA THEN DO[T SYMBOL IT TIGP0329
C 470 IF(IPLLOT.EQ.0) GO TO 490 TIGP0330
470 IF(IS.LE.90) GO TO 480 TIGP0331
470 IS=IS-90 TIGP0332
470 GO TO 470 TIGP0333
480 CALL POINTA(XO,YO) TIGP0334
480 IF(IOFF.EQ.0) CALL ANCHO(IS) TIGP0335
490 NPTA(NL)=1 TIGP0336
490 NL1=NL+1 TIGP0337
490 NPTA(NL1)=0 TIGP0338
490 LCNT=LCNT-LDEL TIGP0339
490 CALL NOTEAT(0,LCNT,20,MSG2) TIGP0340
490 LCNT=LCNT-LDEL TIGP0341
490 CALL MOVABS(0,LCNT) TIGP0342
490 CALL ANMCDE TIGP0343
490 CALL GETIN(1,VLABL(NL)) TIGP0344
490 ISAVE=NL TIGP0345
490 JSAVE=NSTOR TIGP0346
490 IF(IPLLOT.EQ.0) GO TO 530 TIGP0347
490 GO TO 240 TIGP0348
C PLOT (P) TIGP0349
C C CHECK FOR TABLET MODE, SKIP SPECIAL P SECTION IF TABLET C
C 500 IF(ITAB.EQ.1) GO TO 530 TIGP0350
500 IF(XO.GT.TXMIN) GO TO 530 TIGP0351
500 IF(NLINE.GT.0) GO TO 520 TIGP0352
DO 510 I=1,NL TIGP0353
510 NDRAW(I)=0 TIGP0354
520 IY=(YO-YBEG)/DELY+300 TIGP0355
520 II=(3045-IY)/50+1 TIGP0356
520 IF(II.LT.1) II=1 TIGP0357
520 IF(II.GT.NL) II=NL TIGP0358

B-14
NDRAW(I)=1  
NLIN\_E=1  
GO TO 200  
530 CALL NEWPAG  
GO TO 40  

RESTORE WINDOW (R)  

540 IF(XO.GT.TXMIN.OR.NLINE.EQ.0) GO TO 550  
NLIN\_E=0  
GO TO 200  
550 IWIN=0  
GO TO 530  

SHOW VALUE (S)  

560 LCNT=LCNT-LDEL  
CALL MOVABS(0,LCNT)  
CALL ANMODE  
PRINT 570,X0,Y0  
570 FORMAT('X='G13.5,/,*Y='G13.5)  
LCNT=LCNT-LDEL  
GO TO 200  

VALUE IN (V)  

580 LCNT=LCNT-LDEL  
CALL NOTATE(0,LCNT,10,MSG5)  
LCNT=LCNT-LDEL  
CALL MOVABS(0,LCNT)  
CALL ANMODE  
CALL GETIN(2,VTEM)  
X0=VTEM(1)  
Y0=VTEM(2)  

CHECK FOR N COMMAND VALUE INPUT SECTION.  

IF(I\_PLOT.EQ.0) GO TO 461  
LCNT=LCNT-LDEL  
CALL NOTATE(0,LCNT,13,MSG6)  
CALL TINPUT(ICH\_AR)  
GO TO 260  

WINDOW (W)  

590 CALL GETVAL(ICH\_AT,X1,Y1)  
AXMIN=AMIN1(XO,X1)  
AXMAX=AMAX1(XO,X1)
AYMIN=AMIN1(YO,Y1)
AYMAX=AMAX1(YO,Y1)
IWIN=1
GO TO 530
END

CGETVAL
SUBROUTINE GETVAL(ICHAR,XV,YV)
COMMON/TEKGPR,DUM(3),ICL,ITAB,ITABS,XS,YS,DUM2(20),NLINE
C ICL=0 INITIALIZE TABLET
C IC=0 NOT IN CONTINUOUS MODE
C ITAB=0 SCREEN CURSER
C****
C**** GET VALUE AND CHARACTER FROM CROSS HAIRS OR TABLET
C****
C**** CHECK FOR TABLET INPUTS
C****
10 IF(ITAB.EQ.1)GO TO 20
 CALL VCURSR(ICHAR,XV,YV)
C****
C**** CHECK TO SEE IF SCREEN COMMAND WAS TO ACTIVATE TABLET
C****
 IF(ICHAR.NE.84)GO TO 30
 ITAB=1
 ICL=ITABS
C****
C**** SET TABLET LAST CHARACTER (ALSO USED AS A FLAG FOR INITIALIZATION
C****
20 ICHAR=ICL
 CALL TABVU(ICHAR,XV,YV)
C****
C**** SAVE LAST TABLET CHARACTER COMMAND
C****
 ICL=ICCHAR
 NLINE=0
C C SET FLAG TO PLOT ALL LINES IN TABLET MODE
C
C****
C**** CHECK FOR TABLET HALT COMMAND
C****
 IF(ICL.NE.72)GO TO 30
C****
C**** TURN OFF TABLET AND SAVE LAST COMMAND
C****
 ITAB=0
 ITABS=ICL
 RETURN
30 IF(ICHAR.EQ.69) ITABS=0
RETURN
END

CTABVU
SUBROUTINE CTABVU(ICHAR,XV,YV)
COMMON/TEKGPPR/LDEL,LCNT,MAXSR,LTV(5)
1,LS,MX1,MY1,MX2,MY2,XB,YB,FACX,FACY,ANG,MYB
DIMENSION MSG1(54),MSG2(43),MSG3(43),MSG4(18),ICONV(2,10)
1 IRETN(2,10),XTEM(2)
DATA ((ICONV(I,J),J=1,10),I=1,2)/65,66,67,68,69,71,72,
1 105,78,80,82,83,86,87,
1 32,32,32,32,32,32/0
DATA ((IRETN(I,J),J=1,10),I=1,2)/0,0,0,0,1,1,1,
1 1,0,1,1,0,1,0,
1 1,1,1,1,1,1/0
DATA MSG1/83,113,117,97,114,101,32,109,101,110,
1 117,32,119,105,116,104,32,116,97,98,
1 108,101,116,32,97,110,100,32,116,111,11,
1 117,99,104,32,117,112,102,114,32,
1 108,101,102,116,32,109,101,110,117,32,
1 100,111,116,46/
DATA MSG2/84,111,117,99,104,32,97,120,105,115,
1 32,111,114,105,103,105,110,32,97,110,11,
1 100,32,101,110,116,101,114,32,118,97,
1 108,117,101,115,32,88,32,97,110,100,
1 32,89,46/
DATA MSG3/84,111,117,99,104,32,32,32,97,120,
1 105,115,32,97,116,32,109,97,120,32,
1 108,101,110,116,104,32,97,110,100,
1 32,101,110,116,101,114,32,118,97,108,
1 117,101,46/
DATA MSG4/76,97,115,116,32,99,111,109,109,97,
1 110,100,32,119,97,115,32,32/0
IF(ICHAR.NE.0)GO TO 30
C TABLET HAS NOT BEEN SET CHECK IT
LS=100
CALL TABINT(1,0,0)
CALL NEWPAG
LCNT=3120-LDEL
C GET MENU POSITION
CALL NOTATE(0,LCNT,54,MSG1)
CALL BELL
CALL ONEPNT(MX1,MY1)
MX2=MX1+1000
MY2=MY1-200
GO TO 20
10 LCNT=3120
CALL NEWPAG

B-17
20 LCNT=LCNT-LDEL
C
GET COORDINATE INTERSECTION
CALL NOTATE(0,LCNT,43,MSG2)
CALL BELL
CALL ONEPNT(MXB,MYB)
LCNT=LCNT-LDEL
CALL MOVABS(0,LCNT)
CALL ANMODE
CALL GETIN(2,XTEM)
XB=XTEM(1)
YB=XTEM(2)
LCNT=LCNT-LDEL
MSG3(7)=88
C
GET X AXIS POSITION MAX
CALL NOTATE(0,LCNT,43,MSG3)
CALL BELL
CALL ONEPNT(MXM,NXM)
LCNT=LCNT-LDEL
CALL MOVABS(0,LCNT)
CALL ANMODE
C
GET VALUE AT POSITION
CALL GETIN(1,XM)
DX=MXM-MXB
DY=NXM-MYB
C
COMPUTE ANGLE CORRECTION
ANG=ATAN2(DY,DX)
LCNT=LCNT-LDEL
MSG3(7)=89
C
GET Y AXIS POSITION MAX
CALL NOTATE(0,LCNT,43,MSG3)
CALL BELL
CALL ONEPNT(MYM,NYM)
LCNT=LCNT-LDEL
CALL MOVABS(0,LCNT)
CALL ANMODE
C
GET VALUE AT POSITION
CALL GETIN(1,YM)
DY=NYM-MYB
CCSA=COS(ANG)
C
SET UP COMMON FACTORS FOR ANGLE CORRECTIONS
FACX=(XM-XB)*CCSA/DX
FACY=(YM-YB)*CCSA/DY
INIT=1
XV=XM
YV=YM
ICCHAR=87
C
RETURN PLOT COMMAND
RETURN
C   CHECK FOR TABLET INITIALIZED
30 IF(INIT.NE.1)GO TO 40
   ICHAR=87
   XV=XB
   YV=YB
   INIT=0
   RETURN
40 CALL BELL
   CALL ONEPNT(IX,IY)
   C   CHECK TO SEE IF POINT SENT IS A MENU COMMAND
      IF(IX.GT.MX2.OR.IX.LT.MX1)GO TO 50
      IF(IY.GT.MY1.OR.IY.LT.MY2)GO TO 50
      IC=(IX-MX1)/LS+1
      IR=(MY1-IY)/LS+1
   C   CONVERT ROW AND COLUMN POSITION TO COMMAND CHARACTER
      ICHAR=ICNV(IR,IC)
      IF(ICHAR.EQ.32) RETURN
      MSG4(18)=ICHAR
      LCNT=LCNT-LDEL
   C   LAST MESSAGE COMMAND
      CALL NOTATE(0,LCNT,18,MSG4)
      IF(ICHAR.EQ.105) GO TO 10
      IF(I'RET(IR,IC).EQ.1)RETURN
      GO TO 40
   C   CONVERT TABLET UNITS TO VIRTUAL UNITS WITH ANGLE CORRECTION
50 DX=IX-MXB
   DY=IY-MYB
   IF(DX.EQ.0.) DX=1.E-20
   R=SQRT(DX*DX+DY*DY)
   ANGR=ATAN2(DY,DX)-ANG
   XV=R*FACX*COS(ANGR)+XB
   YV=R*FACY*SIN(ANGR)+YB
   RETURN
END

CDRAWIT
   SUBROUTINE DRAWT(NL,NPTA,X,Y,ISUB)
   COMMON/TKTRNX/ITEKC(60)
   COMMON/TEKPPR/DUM(20),EN(2),DEL(2),BEG(2),RDY2,NS,NDRAW(30),MODE(30)
   DIMENSION QSY(306),QSX(306),NPTA(1),X(1),Y(1),ISUB(1)
   EQUIVALENCE (IOFF,ITEKC(30))
   C
   C     0 SYMBOLS 1 LINE 2 SPLINE WRT X 3 SPLINE WRT Y 4 ARC FIT 5 CLOSED
   C
      IT=64
      K=1
      NSUM=1

B-19
NC=0
DO 290 I=1,NL
ISYM=MCDE(I)
ITYP=IABS(ISYM)
IF(ITYP.GT.1) GO TO 40
END=NSUM+NPTA(I)-1
IT=IT+1
IF(IT.GT.90)IT=65
DO 20 J=NSUM,END
IF(NLINE.EQ.0) GO TO 10
IF(NDRAW(I).EQ.0) GO TO 20
10 XP=X(K)
YP=Y(K)
IF(J.EQ.NSUM) CALL MOVEA(XP,YP)
IF(ITYP.EQ.1) CALL DRAWA(XP,YP)
IF(ISYM.LT.0) GO TO 20
CALL MOVEA(XP,YP)
IF(IOFF.EQ.0) CALL ANCHO(IT)
CALL MOVEA(XP,YP)
20 K=ISUB(K)
30 NSUM=END+1
GO TO 290
C
PLOT WITH SPLINE
C
40 NS=NC
NPT=NPTA(I)
NC=NC+NPT
IT=IT+1
IF(IT.GT.90)IT=65
IF(NLINE.EQ.0) GO TO 60
IF(NDRAW(I).NE.0) GO TO 60
C
LOCATE POINTER OT NEXT LINE
DO 50 L=1,NPT
50 K=ISUB(K)
GO TO 290
60 JFIT=2
YO=Y(K)
K1=ISUB(K)
IF(ITYP.GT.2) GO TO 80
XO=X(K)
C
CHECK X DATA FOR ASCENDING ORDER
DO 70 L=2,NPT
X1=X(K1)
IF(X1.LE.XO) GO TO 110
K1=ISUB(K1)
70 XO=X1
GO TO 210

B-20
80 IF(ITYP.GT.3) GO TO 100
   CHECK Y DATA FOR ASCENDING ORDER
   DO 90 L=2,NPT
      Y1=Y(K1)
   IF(Y1.LE.YO) GO TO 110
      K1=ISUB(K1)
90   YO=Y1
   GO TO 210
100 JFIT=ITYP-2
110 NCIR=0
   IF(JFIT.EQ.3) NCIR=-NPT/2-1
   MPT=NPT-2*NCIR
   QSY(1)=MPT
   QSX(1)=MPT
   S=0.
   KA=NS
   KO=KA
   KE=KO+NPT
   KSAVE=K
   KA=KA+NCIR
   DO 160 M=1,MPT
      M1=M+1
      KA=KA+1
   IF(KA.GT.KO) GO TO 130
   NDO=NPT+NCIR
   DO 120 II=1,NDO
120   K=ISUB(K)
      KA=KA+NPT
      GO TO 140
130 IF(KA.NE.(KE+1)) GO TO 140
      JSAVE=K
      K=KSAVE
      KA=KA-NPT
140 CONTINUE
   L=M1+MPT
   YYYY=Y(K)
   XXXP=X(K)
   K=ISUB(K)
   IF(M.EQ.1) GO TO 150
   DS=SQRT(RDX2*(XXXP-XO)**2+RDY2*(YYYY-YO)**2)
   S=S+DS
150   XO=XXXP
      YO=YYYY
   QSY(M1)=S
   QSY(M1)=S
   QSX(K)=XXXP
160 QSY(K)=YYYY
   KA=KO+NPT
   DRAWC065
   DRAWC066
   DRAWC067
   DRAWC068
   DRAWC069
   DRAWC070
   DRAWC071
   DRAWC072
   DRAWC073
   DRAWC074
   DRAWC075
   DRAWC076
   DRAWC077
   DRAWC078
   DRAWC079
   DRAWC080
   DRAWC081
   DRAWC082
   DRAWC083
   DRAWC084
   DRAWC085
   DRAWC086
   DRAWC087
   DRAWC088
   DRAWC089
   DRAWC090
   DRAWC091
   DRAWC092
   DRAWC093
   DRAWC094
   DRAWC095
   DRAWC096
   DRAWC097
   DRAWC098
   DRAWC099
   DRAWC100
   DRAWC101
   DRAWC102
   DRAWC103
   DRAWC104
   DRAWC105
   DRAWC106
   DRAWC107
   DRAWC108
   DRAWC109
   DRAWC110
   DRAWC111
   DRAWC112
QSX(L+1)=0.
QSY(L+1)=0.
QSX(L+2)=1.
QSY(L+2)=1.
XO=QSX(MPT+2-NCIR)
YO=QSY(MPT+2-NCIR)
CALL MOVEA(XO,YO)
IF(IOFF.EQ.0) CALL ANCHO(IT)
CALL MOVEA(XO,YO)
SCK=QSX(3-NCIR)
S=QSX(2-NCIR)
IF(NPT.LE.1) GO TO 290
DC=40.
DS=40.
NCK=2
S=S+DS
XP=SPLNQ1(1,QSX,S)
YP=SPLNQ1(1,QSY,S)
DCK=SQRT(RDX2*(XO-XP)**2+RDY2*(YO-YP)**2)
DS= DC*DS/DCK
170 IF(S.LT.SCK) GO TO 200
NSYM=MPT+1+NCK-NCIR
XS=QSX(NSYM)
YS=QSY(NSYM)
CALL DRAWA(XS,YS)
IF(ISYM.LE.0.AND.NCK.NE.NPT) GO TO 190
CALL MOVEA(XS,YS)
IF(IOFF.EQ.0) CALL ANCHO(IT)
180 CALL MOVEA(XS,YS)
NCK=NCK+1
SCK=QSX(NCK+1-NCIR)
IF(NCK.LE.NPT+JFIT-2) GO TO 180
IF(JFIT.EQ.3) K=JSAVE
GO TO 290
200 CALL DRAWA(XP,YP)
XO=XP
YO=YP
GO TO 170
210 QSX(1)=NPT
DO 240 M=1,NPT
M=M+1
KA=NS+M
L=N+NPT
XP=X(K)
YP=Y(K)
IF(M.NE.1.AND.M.NE.NPT.AND.ISYM.LE.0) GO TO 220
CALL MOVEA(XP,YP)
IF(IOFF.EQ.0) CALL ANCHO(IT)
220 IF(ITYP.NE.3) GO TO 230
    QSX(N)=YP
    QSX(L)=XP
    GO TO 240
230 QSX(N)=XP
    QSX(L)=YP
240 K=ISUB(K)
    QSX(L+1)=0.
    QSX(L+2)=1.
    XEN=QSX(NPT+1)
    XIN=QSX(2)
    IFITP=ITYP-1
    BCK=BEG(IFITP)
    ECK=EN(IFITP)
    DELT=DEL(IFITP)*30.
    IF(XIN.LT.BCK) XIN=BCK
    IF(XEN.GT.ECK) XEN=ECK
    KILL=0
    DO 280 M=1,200
    XI=XIN+DELT*(M-1)
    IF(XI.LT.XEN) GO TO 250
    KILL=1
    XI=XEN
250 YI=SPLNQ1(1, qsx, xi)
    IF(ITYP.EQ.3) GO TO 260
    XP=XI
    YP=YI
    GO TO 270
260 XP=XI
    YP=YI
270 IF(M.EQ.1) CALL MOVEA(XP,YP)
    CALL DRAWA(XP,YP)
    IF(NPT.EQ.1) GO TO 290
    IF(KILL.EQ.1) GO TO 290
280 CONTINUE
290 CONTINUE
300 RETURN
END

CSPLNQ1

FUNCTION SPLNQ1 (NLOC,X,XINDEP)
C*** LOCAL CUBIC FIT 8/9/77 M.J. CADDY
DIMENSION X(1),QM(3)
EQUIVALENCE (QM(1),T3),(QM(2),Q2),(QM(3),Q3)
XIN=XINDEP
NS=NLOC
NOPTS=X(NS)
ID=NS+NOPTS
NSP1=NS+1
NSP2=NS+2
IF(NOPTS.LE.1) GO TO 130
IF(NOPTS.GT.2) GO TO 10
N=ID+NOPTS
T3=(X(N)-X(N-1))/(X(ID)-X(ID-1))
M=ID
NTRAP=1
GO TO 280
10 NS2=NOPTS*2+NSP1
L=X(NS2)
LSC=NS2+1
IQMODE=X(LSC)
K=L+NS
NL=NSP1
NH=ID
NTRAP=-1
C*** BINARY SEARCH FOR INTERVAL
   IF(XIN-X(ID))30,140,20
20 NTRAP=0
   GO TO 150
30 IF(XIN-X(NSP1))40,40,60
40 NTRAP=1
50 K=NSP2
   GO TO 160
60 IF(L)120,120,70
70 IF(XIN-X(K))80,100,100
80 NH=K
   K=K-1
90 IF(XIN-X(K))110,100,100
100 NL=K
   GO TO 120
110 NH=K
120 K=(NH-NL)/2+NL
   IF(K-NL)90,140,90
130 YOUT=X(NSP2)
   GO TO 320
140 LFAST=L-NH+NS
   X(NS2)=NH-NS
150 K=NH
160 M=K
   N=M+NOPTS
   Y3=X(N-1)
   X3=X(M-1)
C*** CHECK FOR FAST MODE AND EXTRAPOLATION
   IF(NTRAP.GE.0) GO TO 180
   IF(IQMODE*L.EQ.0.OR.LFAST.NE.0) GO TO 180
   DO 170 I=1,3
170 QM(I)=X(LSC+I)
GO TO 310

180 Y4=X(N)
    X4=X(M)
    A3=Y4-X3
    S3=(Y4-Y3)/A3
    IF(M.EQ.NSP2) GO TO 190
    X2=X(M-2)
    Y2=X(N-2)
    S2=(Y3-Y2)/(X3-X2)
    IF(M.EQ.ID) GO TO 200

190 X5=X(M+1)
    Y5=X(N+1)
    S4=(Y5-Y4)/(X5-X4)
    IF(M.EQ.NSP2) S2=S3+S3-S4
    GO TO 210

200 S4=S3+S3-S2

210 IF(M.LE.(NSP2+1)) GO TO 220
    S1=(Y2-X(N-3))/(X2-X(M-3))
    GO TO 230

220 S1=S2+S2-S3

230 IF(M.GE.(ID-1)) GO TO 240
    S5=(X(N+2)-Y5)/(X(M+2)-X5)
    GO TO 250

240 S5=S4+S4-S3

250 W2=ABS(S4-S3)
    W3=ABS(S2-S1)
    SW=W2+W3
    IF(SW.NE.0.0) GO TO 260
    W2=0.5
    W3=0.5
    SW=1.0

260 T3=(W2*S2+W3*S3)/SW
    W3=ABS(S3-S4)
    W4=ABS(S3-S2)
    SW=W3+W4
    IF(SW.NE.0.0) GO TO 270
    W2=0.5
    W4=0.5
    SW=1.0

270 T4=(W3*S3+W4*S4)/SW
    IF(NTRAP.LT.0) GO TO 290
    IF(NTRAP.EQ.0) T3=T4

280 IX=M-NTRAP

C*** FAST EXIT FOR 2 POINTS AND LINEAR EXTRAPOLATION
    YOUT=X(IN+NOPTS)+(XN-X(IN))*T3
    GO TO 320

290 Q2=(2.0*(S3-T3)+S3-T4)/A3
    Q3=(-S3-S3+T3+T4)/(A3*A3)

B-25
IF(IQMODE*LFAST.EQ.0) GO TO 310
DO 300 I=1,3
300 X(LSC+I)=QM(I)
310 DX=XIN-X3
YOUT=Y3+DX*(T3+DX*(Q2+DX*Q3))
320 SPLNQ1=YOUT
RETURN
END

CTTITE
SUBROUTINE TTTIE(IX,IY,NTL,LABTL,NM,IA)
DIMENSION LABTL(1),IP(136)
C NTL =NUMBER OF 10 CHARACTER WORDS
C NM MAX CHARACTERS PER LINE
C IA SWITCH, IA=0 HORIZ, IA=1 VERTICAL
C IX SCREEN CENTER
C IY SCREEN CENTER
IF(NTL.LE.0) RETURN
NC =10*NTL
C GET CHARACTER SIZE
CALL CSIZE(IHORZ,IVERT)
C CONVERT LABEL TO ADE
CALL KAM2AS(NC,LABTL,IP)
IX1=IX
IY1=IY
ITAL1=0
NBLK=0
DO 70 K=1,NC
C CHECK FOR LEADING BLANKS
IF(IP(K).NE.32) GO TO 10
IF(ITAL1.EQ.0) GO TO 70
NBLK=NBLK+1
C CHECK FOR 3 BLANKS TO TERMINATE LINE
IF(NBLK.NE.3) GO TO 20
ITAL1=ITAL1-2
GO TO 50
10 NBLK=0
C CHECK FOR MAX LINE LENGTH EXCEEDED
20 IF(ITAL1.LT.NM) GO TO 30
IF(IP(K).EQ.32) GO TO 50
30 ITAL1=ITAL1+1
IP(ITAL1)=IP(K)
IF(K.LT.NC) GO TO 70
40 ITAL1=ITAL1-NBLK
C CHECK FOR VERTICAL OR HORIZ LABEL
50 IF(I.A.NE.0) GO TO 60
IX1=IX-IHORZ*ITAL1*.5
CALL NOTATE(IX1,IY1,ITAL1,IP)
IY1=IY1-IVERT*1.1

TTIT0001
TTIT0002
TTIT0003
TTIT0004
TTIT0005
TTIT0006
TTIT0007
TTIT0008
TTIT0009
TTIT0010
TTIT0011
TTIT0012
TTIT0013
TTIT0014
TTIT0015
TTIT0016
TTIT0017
TTIT0018
TTIT0019
TTIT0020
TTIT0021
TTIT0022
TTIT0023
TTIT0024
TTIT0025
TTIT0026
TTIT0027
TTIT0028
TTIT0029
TTIT0030
TTIT0031
TTIT0032
TTIT0033
TTIT0034
TTIT0035
TTIT0036
TTIT0037
TTIT0038
TTIT0039
TTIT0040

B-26
ITL1=0
GO TO 70
60  IX1=IX1+IVERT*ITL1*.5
    CALL MOVABS(IX1,IX1)
    CALL VLABEL(ITL1,IP)
    IX1=IX1+IHORZ*.1
    ITL1=0
    70 CONTINUE
END

CGETIN

SUBROUTINE GETIN(NIN,Y)

C
C    MICHAEL CADDY  3/19/78
C    DIMENSION Y(1),IC(80)
C    FREE FORM INPUT CODE

NW=0
10  J=0
    READ 20,IC
    IF(EOF(5).EQ.0) GO TO 30
    NIN=NW
    RETURN
20  FORMAT(80R1)*
30  JC=0
    JD=0
    JS=1
    NC=0
    X=0.
40  J=J+1
    ONLY ONE CARD PER INPUT READ
    MODIFIED TO READ MORE THAN ONE CARD 4/26/78 MJC
    IF(J.GT.80) GO TO 10
    I=IC(J)
    CHECK FOR VALID NUMERIC FIELD
    IF(I.GT.32B.AND.I.LT.45B) GO TO 110
    IGNORE LEAD + SIGN
    IF(I.EQ.45B) GO TO 40
    SET FLAG FOR NEGATIVE VALUE
    IF(I.NE.46B) GO TO 50
    JS=-1
    GO TO 40
    CHECK FOR DECIMAL
    50  IF(I.NE.57B) GO TO 60
    IF(JC.EQ.-1) GO TO 120
    IF THIS IS SECOND DECIMAL BLOW OFF TO ERROR CODE
    JC=-1
    GO TO 40

GET10001
GET10002
GET10003
GET10004
GET10005
GET10006
GET10007
GET10008
GET10009
GET10010
GET10011
GET10012
GET10013
GET10014
GET10015
GET10016
GET10017
GET10018
GET10019
GET10020
GET10021
GET10022
GET10023
GET10024
GET10025
GET10026
GET10027
GET10028
GET10029
GET10030
GET10031
GET10032
GET10033
GET10034
GET10035
GET10036
GET10037
GET10038
C CHARACTER IS BLANK TREAT AS COMMA IF NOT LEADING
60 IF(I.EQ.55B)GO TO 80
   IF(I.EQ.56B)GO TO 70
   GO TO 120
70 IF(NC.GT.0)GO TO 90
   NW=NW+1
   IF(NW.GT.NIN) RETURN
   GO TO 40
C TWO COMMAS ..IGNORE THIS DATA FIELD AND GO ON TO NEXT
80 IF(NC.EQ.0) GO TO 40
C SHIFT DECIMAL TO NUMBER
90 X=JS*X*10.**JD
   NW=NW+1
   Y(NW)=X
   IF(NW.GE.NIN) RETURN
   GO TO 30
110 JD=JD+JC
   NC=NC+1
C ADD DIGIT TO NUMBER ,,CAREFULLY
   X=X*10+(I-33B)
   GO TO 40
C ERROR CODE
120 DO 130 K=1,80
130 IC(K)=55B
   IC(J)=47B
   PRINT 140 ,IC
140 FORMAT(2X,80R1)
   PRINT 150
150 FORMAT(* BAD FIELD, RE-ENTER DATA*)
   GO TO 10
END