DOE/NASA TECHNICAL
MEMORANDUM

DOE/NASA TM-78293

THREE COMPUTER CODES TO READ, PLOT, AND TABULATE OPERATIONAL TEST-SITE RECORDED SOLAR DATA

By Stephen D. Stewart, Robert J. Sampson, Jr., Richard E. Stonemetz, and Sandra L. Rouse, Systems Analysis and Integration Laboratory

National Aeronautics and Space Administration
George C. Marshall Space Flight Center, Alabama 35812

July 1980

For the U. S. Department of Energy

U.S. Department of Energy

Solar Energy

TABLE OF CONTENTS

Page

INTRODUCTION 1

I. DATA SOURCE 1

A. Byte Locations 2

B. System Subtypes 3

C. Source Access Arrangement 3

II. PROGRAM USAGE 5

A. TAPFIL 5

B. CHPLOT 8

C. WRTCNL 13

III. PROGRAM SOFTWARE IMPLEMENTATION 14

A. Data Conversion Routines 14

B. TEKPLOT Library Routines 15

C. UTILITIES Library Routines 17

Appendix A- Sample Problem 19

Appendix B- Flowcharts and Reference Tables 27

Appendix C- TAPFIL Source Listing 31

Appendix D- CHPLOT Source Listing 53

Appendix E- WRTCNL Source Listing 81

LIST OF ILLUSTRATIONS

Figure	Title	Page
1.	IBM 360 Tape Data Arrangement	3
2.	Sample IPCL	4
3.	TAPFIL Screen Display	6
4.	TAPFIL Time Profile	7
5.	CHPLOT Screen Display	9
6.	Data Plot	10
7.	Plotting Options	11
8.	Scaling Option Display	11
9.	Scaled Plot	12
10.	WRTCNL Screen Display	13

INTRODUCTION

Under contract with the Department of Energy, the National Aeronautics and Space Administration is managing 48 operational solar heating and cooling test sites (OTS) and numerous large commercial systems. The sites are expected to provide information which will help engineers and designers evaluate the solar system's effectiveness in (1)reducing fossil fuel consumption and (2)maintaining design performance goals.

The information from these operational test sites (OTS) is provided as a large mass of data (insolation, flow rates, temperatures, etc.) stored on tape. In order to attain adequate engineering assessments of these systems, procedures must be developed to analyze the data on a real time basis. This report presents a detailed discussion of three computer programs (TAPFIL, CHPLOT, WRTCNL) developed by MSFC to read the OTS site data tapes and put the data in a form usable with other solar heating and cooling system analysis computer programs. These programs utilize the MSFC Systems Analysis and Integration Laboratory's PDP 11/70 computer facility. TAPFIL verifies the analytical methodology, CHPLOT plots the site data for examination of system/subsystem trends, and WRTCNL lists the data in tabular form.

Implementation of the program code on another computer system would be subject to an understanding of the program execution and the software capabilities of that system. The computer code development procedures are presented and discussed in such a manner as to prepare the reader to utilize this program code on any computer which has adequate capabilities.

I. DATA SOURCE

Each OTS solar system built under MSFC management utilizes a site data acquisition system (SDAS) developed by IBM. The SDAS's are designed to retrieve raw data from 48 site measurements every 5 minutes and 20 seconds continuously. Several large OTS systems require more than 48 measurements of data to determine total system performance and require two SDAS's. This allows a maximum of 96 channels of site data to be retrieved. When two SDAS's are used at one site, two methods are used to join them together. One method is to connect the two of them to one clock so that the measurements may be sent to the central data processing facility in time sequence. The other method is to connect each SDAS to a separate clock, thus creating two time profiles which must be sorted and merged at the central data processing facility to obtain a time sequence.

The distinction between which type of time profile is created at a site is made by the system subtype. There are five subtypes. The subtype is determined from reading the IBM 360 data tapes. These tapes contain EBCDIC characters in a certain order according to subtype. These characters are sequentially stored in 62 records (210 bytes long) per data block of 13020 bytes. Each record contains time information, site identification information and 48 channels of data. The order of the data arrangement and time profile is defined below according to subtype. The arrangement of the data on the tape is shown in Figure 1.

SUBTYPE	DATA ACQUISITION	CHARACTER REPRESENTATION
1	One 48 channel SDAS	● Byte 1: No significant use ● Bytes 2-4: 3 digit Site ID number ● Bytes 5-16: Year, Month, Day, Hour, Minute, Seconds For example - YYMMDDHHNNSS is represented by Bytes 5, 6, ...15,16. If date and time were 9/1/79 and 10:32:58, the tape would yield the following Byte representation - 790901103258 (Bytes 5-16) ● Byte 17: System Subtype ● Byte 18: No significant use ● Bytes 19-210: 48 4-byte hexadecimal EBCDIC words representing channels 1-48
(There is no subtype 2)		
3	1st 48 channel SDAS of sorted and merged time profile	Same as above except always used in conjunction with sybtype 4. Bytes 19-210 represent channels 1-48.
4	2nd 48 channel SDAS of sorted and merged time profile	Same as above except always used with subtype 3. Bytes 19-210 represent channels 49-96.
5	1st 48 channel SDAS of connected time profile.	Same as above except always used with subtype 6. Bytes 19-210 represent channels 1-48.
6	2nd 48 channel SDAS of connected time profile.	Same as above except used with subtype 5. Bytes 19-210 represent channels 49-96.

These records are stored in a data block. Subtype 1 records, utilizing one SDAS to retrieve data, are stored one record at a time with each time point being 5 minutes and 20 seconds apart. Subtype 3 records should be read immediately preceeding subtype 4 records. These two subtypes should have the exact same time point. The subtype 4 record should then be followed by another subtype 3 record with a new time point 5 minutes and 20 seconds later and continue in this subtype 3 - subtype 4 sequential manner until completion of 270 output records. In some instances, however, contiguous records of subtype 3 or subtype 4 may be encountered. In these instances, the contiguous subtype records are not stored and the number of output records will be less than 270. Subtypes 5 and 6 are read and stored exactly the same as subtypes 3 and 4 except no contiguous records will be encountered.

All the above mentioned subtypes are written on 9-track data tapes by an IBM 360 computer at a density of 1600 bits per inch (BPI) with one end-of-file (EOF) written after each month of site data.

Each OTS has an Instrumentation Program and Components List (IPCL) which is used to correlate the channel number to the actual measurement at the site. Listed in the IPCL are the channel numbers, the measurement numbers, the operating ranges, and the names of each measurement. Once the IBM data tape has been successfully read and the data stored, the IPCL is used to determine which measurements are needed for calculations and plots. A sample IPCL can be found in Figure 2. The data read from the IBM tape is stored by channel number in a data file. This will allow computer programs to be developed that do certain required calculations dependent upon a set of input channel numbers. Since every operational test site has a different channel orientation, these programs can be used to analyze and/or evaluate OTS systems according to the input channel numbers. This type of data storage (direct access unformatted) was chosen because it requires very little disk storage space compared to other types of storage, and is easily accessible, in that no data formatting is required.

We have discussed the formulation of the data source (IBM 360 OTS Tapes), the tape format, the storage of the source data on the PDP 11/70 system disk, and the use of the IPCL to determine channel information for analysis and plotting. With this background, the specific programs used to read, convert and analyze the source data will be discussed next.

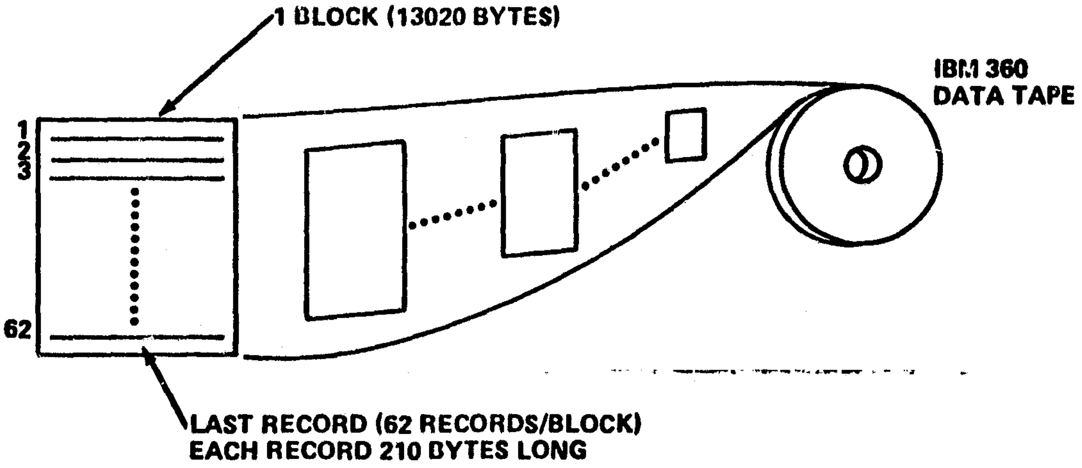
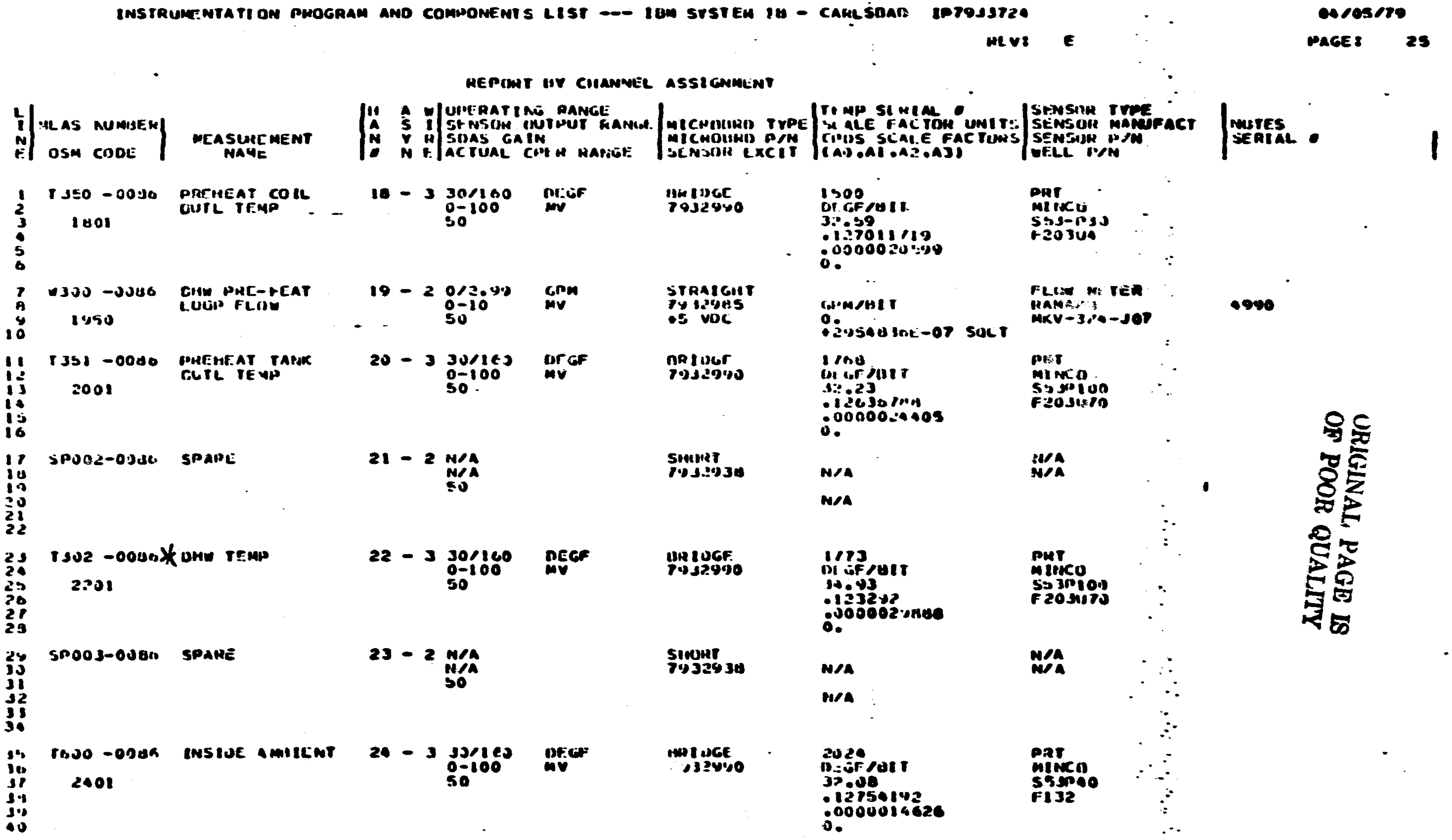


FIGURE 1. IBM 360 TAPE DATA ARRANGEMENT

FIGURE 2. SAMPLE IPCL



```
        INSTRUMENTATION PROGRAM AND COMPONENTS LIST --- IBM SYSTEM 18 - CARLSBAD  IP7933724                      04/05/79
                                                                                      REV:   E                  PAGE:    25

                                             REPORT BY CHANNEL ASSIGNMENT

L|MLAS NUMBER|                  |H  A W|OPERATING RANGE      |MICROBRD TYPE|TEMP SERIAL #      |SENSOR TYPE               |NOTES
I|           |   MEASUREMENT    |A  S I|SENSOR OUTPUT RANGE  |MICROBRD P/N |SCALE FACTOR UNITS |SENSOR MANUFACT           |SERIAL #
N|           |      NAME        |N  Y R|SDAS GAIN            |SENSOR EXCIT |CDOS SCALE FACTORS |SENSOR P/N                |
E| OSM CODE  |                  |#  N E|ACTUAL OPER RANGE    |             |(A0,A1,A2,A3)      |WELL P/N                  |

 1 T350 -0086  PREHEAT COIL     18 - 3 30/160     DEGF        BRIDGE        1500                PRT
 2             OUTL TEMP               0-100      MV          7932990       DEGF/BIT            MINCO
 3   1801                              50                                   32.59               S53-P30
 4                                                                          .12701179           F20304
 5                                                                          .0000020599
 6                                                                          0.

 7 W300 -0086  DHW PRE-HEAT     19 - 2 0/2.99     GPM         STRAIGHT                          FLOW METER
 8             LOOP FLOW               0-10       MV          7932985       GPM/BIT             RAMAPO                     4990
 9   1950                              50                     +5 VDC        0.                  MKV-3/4-J07
10                                                                          +2954836E-07 SQRT

11 T351 -0086  PREHEAT TANK     20 - 3 30/160     DEGF        BRIDGE        1768                PRT
12             OUTL TEMP               0-100      MV          7932990       DEGF/BIT            MINCO
13   2001                              50                                   32.23               S53P100
14                                                                          .12636788           F203070
15                                                                          .0000024405
16                                                                          0.

17 SP002-0086  SPARE            21 - 2 N/A                    SHORT                             N/A
18                                     N/A                    7932938       N/A                 N/A
19                                     50
20                                                                          N/A
21
22

23 T302 -0086* DHW TEMP         22 - 3 30/160     DEGF        BRIDGE        1773                PRT
24                                     0-100      MV          7932990       DEGF/BIT            MINCO
25   2201                              50                                   34.93               S53P100
26                                                                          .123292             F203070
27                                                                          .0000029888
28                                                                          0.

29 SP003-0086  SPARE            23 - 2 N/A                    SHORT                             N/A
30                                     N/A                    7932938       N/A                 N/A
31                                     50
32                                                                          N/A
33
34

35 T600 -0086  INSIDE AMBIENT   24 - 3 30/160     DEGF        BRIDGE        2024                PRT
36                                     0-100      MV          7932990       DEGF/BIT            MINCO
37   2401                              50                                   32.08               S53P40
38                                                                          .12754192           F132
39                                                                          .0000014626
40                                                                          0.
```

II. PROGRAM USAGE

The following three programs are designed to be run on the MSFC PDP 11/70, utilizing FORTRAN IV PLUS. CHPLOT and WRTCNL can be run only after a direct access unformatted file has been created by TAPFIL.

Appendix A contains a sample problem utilizing all three programs. An OTS site contracted by WORMSER and located in Columbia, S.C. is analyzed in this problem. Appendix B contains flowcharts of the three programs, utilizing only the main subroutines. The functions performed by these subroutines are also listed in this appendix. A FORTRAN program listing for TAPFIL can be found in Appendix C, for CHPLOT in Appendix D, and for WRTCNL in Appendix E.

A brief description concerning the operation of these programs will now be presented.

A. TAPFIL OPERATIONAL PROCEDURES

The program TAPFIL is designed to create a direct access unformatted file from an IBM 360 tape. This file can then be used with the channel plotting program CHPLOT and the tabulation program WRTCNL, using the PDP 11/70. Due to the software used in the program, TAPFIL must be run from a TEKTRONIX 4014 Demand Terminal.

After program execution has begun, the first statement of Figure 3 will appear on the screen. The first user input will be the number of channels at the site. To ensure program execution, 48 is the miminum number that can be entered. The user should enter 48 even if the site actually has fewer than 48 channels. If the site has more than 48 channels, the input should be in increments of 16, ranging from 48 to 96. The maximum number that can be input is 96, as this is the number of channels on two SDAS's.

The second user input is the desired date of data to be retrieved. This should be entered as MO DA YR. The program searches through the time profile on the tape, finds the date input by the user, and sets flags to begin output to the system disk.

The next user input is the data file name. This name should have a maximum of 29 characters. The convention used at MSFC is to enter the two digit site number with an "s" to indicate site, then the date as MODAYR, followed by .DAT to indicate a data file. For example, if the user wants to use the data for June 13, 1980 from OTS site #12, the input will be 12S061380.DAT.

The fourth user input will be the file number of the data date. The months contained on a tape are listed on the front of that tape. To obtain the file number, count the months from the start of the list until the desired month is reached. This is the file number.

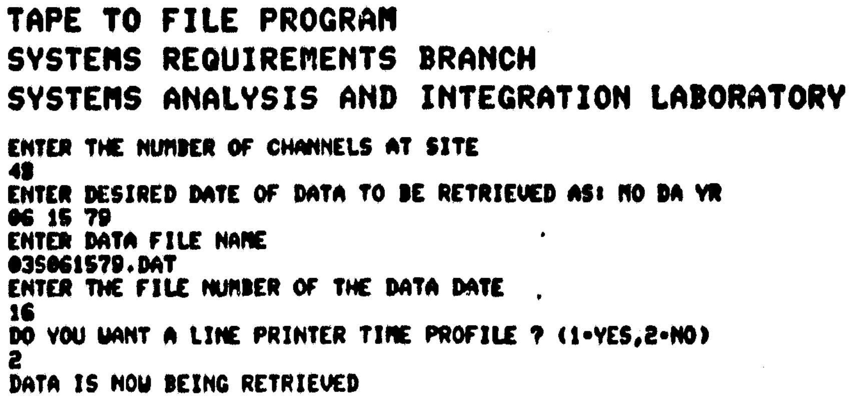
The last user input will be a decision as to whether or not a line printer time profile is wanted. After all inputs are complete, an output statement will appear stating that the data is being retrieved.

The next output on the screen will be a time profile, as shown in Figure 4. If the user has so requested, this time profile will also be output to the line printer. Also written will be a statement telling how many records the data is stored in. If an error is encountered during execution, various error messages will be printed to inform the user.

Once the time profile has been written, the program is in pause mode. The user must depress any key to continue program execution.

A statement will then appear requiring user input as to whether more data is to be retrieved. If so, the statements of Figure 3 will be repeated, with the exception of the headings and the question concerning the number of channels at the site. If no more data is to be retrieved, a statement will appear stating that the tape is being rewound.

The TAPFIL program has now created direct access unformatted files. These files are ready for use with CHPLOT and WRTCNL to analyze the site data.



```
TAPE TO FILE PROGRAM
SYSTEMS REQUIREMENTS BRANCH
SYSTEMS ANALYSIS AND INTEGRATION LABORATORY

ENTER THE NUMBER OF CHANNELS AT SITE
48
ENTER DESIRED DATE OF DATA TO BE RETRIEVED AS: MO DA YR
06 15 79
ENTER DATA FILE NAME
035061579.DAT
ENTER THE FILE NUMBER OF THE DATA DATE
16
DO YOU WANT A LINE PRINTER TIME PROFILE ? (1=YES,2=NO)
2
DATA IS NOW BEING RETRIEVED
```

FIGURE 3. TAPFIL SCREEN DISPLAY

FIGURE 4.

TPFIL TIME PROFILE

```
06/15  00:03:40   06/15  04:08:30   06/15  08:08:30   06/15  12:08:30   06/15  16:08:30   06/15  20:08:30
06/15  00:09:00   06/15  04:13:50   06/15  08:13:50   06/15  12:13:50   06/15  16:13:50   06/15  20:13:50
06/15  00:14:20   06/15  04:19:10   06/15  08:19:10   06/15  12:19:10   06/15  16:19:10   06/15  20:19:10
06/15  00:19:40   06/15  04:24:30   06/15  08:24:30   06/15  12:24:30   06/15  16:24:30   06/15  20:24:30
06/15  00:25:00   06/15  04:29:50   06/15  08:29:50   06/15  12:29:50   06/15  16:29:50   06/15  20:29:50
06/15  00:30:20   06/15  04:35:10   06/15  08:35:10   06/15  12:35:10   06/15  16:35:10   06/15  20:35:10
06/15  00:35:40   06/15  04:40:30   06/15  08:40:30   06/15  12:40:30   06/15  16:40:30   06/15  20:40:30
06/15  00:41:00   06/15  04:45:50   06/15  08:45:50   06/15  12:45:50   06/15  16:45:50   06/15  20:45:50
06/15  00:46:20   06/15  04:51:10   06/15  08:51:10   06/15  12:51:10   06/15  16:51:10   06/15  20:51:10
06/15  00:51:40   06/15  04:56:30   06/15  08:56:30   06/15  12:56:30   06/15  16:56:30   06/15  20:56:30
06/15  00:57:00   06/15  05:01:50   06/15  09:01:50   06/15  13:01:50   06/15  17:01:50   06/15  21:01:50
06/15  01:02:20   06/15  05:07:10   06/15  09:07:10   06/15  13:07:10   06/15  17:07:10   06/15  21:07:10
06/15  01:07:40   06/15  05:12:30   06/15  09:12:30   06/15  13:12:30   06/15  17:12:30   06/15  21:12:30
06/15  01:13:00   06/15  05:17:50   06/15  09:17:50   06/15  13:17:50   06/15  17:17:50   06/15  21:17:50
06/15  01:18:20   06/15  05:23:10   06/15  09:23:10   06/15  13:23:10   06/15  17:23:10   06/15  21:23:10
06/15  01:23:40   06/15  05:28:30   06/15  09:28:30   06/15  13:28:30   06/15  17:28:30   06/15  21:28:30
06/15  01:29:00   06/15  05:33:50   06/15  09:33:50   06/15  13:33:50   06/15  17:33:50   06/15  21:33:50
06/15  01:34:20   06/15  05:39:10   06/15  09:39:10   06/15  13:39:10   06/15  17:39:10   06/15  21:39:10
06/15  01:39:40   06/15  05:44:30   06/15  09:44:30   06/15  13:44:30   06/15  17:44:30   06/15  21:44:30
06/15  01:45:00   06/15  05:49:50   06/15  09:49:50   06/15  13:49:50   06/15  17:49:50   06/15  21:49:50
06/15  01:50:20   06/15  05:55:10   06/15  09:55:10   06/15  13:55:10   06/15  17:55:10   06/15  21:55:10
06/15  01:55:40   06/15  06:00:30   06/15  10:00:30   06/15  14:00:30   06/15  18:00:30   06/15  22:00:30
06/15  01:59:58   06/15  06:05:50   06/15  10:05:50   06/15  14:05:50   06/15  18:05:50   06/15  22:05:50
06/15  02:05:18   06/15  06:11:10   06/15  10:11:10   06/15  14:11:10   06/15  18:11:10   06/15  22:11:10
06/15  02:10:38   06/15  06:16:30   06/15  10:16:30   06/15  14:16:30   06/15  18:16:30   06/15  22:16:30
06/15  02:15:58   06/15  06:21:50   06/15  10:21:50   06/15  14:21:50   06/15  18:21:50   06/15  22:21:50
06/15  02:21:18   06/15  06:27:10   06/15  10:27:10   06/15  14:27:10   06/15  18:27:10   06/15  22:27:10
06/15  02:26:38   06/15  06:32:30   06/15  10:32:30   06/15  14:32:30   06/15  18:32:30   06/15  22:32:30
06/15  02:37:44   06/15  06:37:50   06/15  10:37:50   06/15  14:37:50   06/15  18:37:50   06/15  22:37:50
06/15  02:43:10   06/15  06:43:10   06/15  10:43:10   06/15  14:43:10   06/15  18:43:10   06/15  22:43:10
06/15  02:48:30   06/15  06:48:30   06/15  10:48:30   06/15  14:48:30   06/15  18:48:30   06/15  22:48:30
06/15  02:53:50   06/15  06:53:50   06/15  10:53:50   06/15  14:53:50   06/15  18:53:50   06/15  22:53:50
06/15  02:59:10   06/15  06:59:10   06/15  10:59:10   06/15  14:59:10   06/15  18:59:10   06/15  22:59:10
06/15  03:04:30   06/15  07:04:30   06/15  11:04:30   06/15  15:04:30   06/15  19:04:30   06/15  23:04:30
06/15  03:09:50   06/15  07:09:50   06/15  11:09:50   06/15  15:09:50   06/15  19:09:50   06/15  23:09:50
06/15  03:15:10   06/15  07:15:10   06/15  11:15:10   06/15  15:15:10   06/15  19:15:10   06/15  23:15:10
06/15  03:20:30   06/15  07:20:30   06/15  11:20:30   06/15  15:20:30   06/15  19:20:30   06/15  23:20:30
06/15  03:25:50   06/15  07:25:50   06/15  11:25:50   06/15  15:25:50   06/15  19:25:50   06/15  23:25:50
06/15  03:31:10   06/15  07:31:10   06/15  11:31:10   06/15  15:31:10   06/15  19:31:10   06/15  23:31:10
06/15  03:36:30   06/15  07:36:30   06/15  11:36:30   06/15  15:36:30   06/15  19:36:30   06/15  23:36:30
06/15  03:41:50   06/15  07:41:50   06/15  11:41:50   06/15  15:41:50   06/15  19:41:50   06/15  23:41:50
06/15  03:47:10   06/15  07:47:10   06/15  11:47:10   06/15  15:47:10   06/15  19:47:10   06/15  23:47:10
06/15  03:52:30   06/15  07:52:30   06/15  11:52:30   06/15  15:52:30   06/15  19:52:30   06/15  23:52:30
06/15  03:57:50   06/15  07:57:50   06/15  11:57:50   06/15  15:57:50   06/15  19:57:50   06/15  23:57:50
06/15  04:03:10   06/15  08:03:10   06/15  12:03:10   06/15  16:03:10   06/15  20:03:10
THE DATA HAS BEEN STORED WITH  269  RECORDS
```

ORIGINAL PAGE IS
OF POOR QUALITY

B. CHPLOT OPERATIONAL PROCEDURES

The purpose of CHPLOT is to plot the data stored in a file created by TAPFIL. The program has the capability to plot as many as five plots on one graph. CHPLOT must be run from a TEKTRONIX 4014 Demand Terminal due to the software used in the program.

When program execution has begun, the first lines of Figure 5 will appear on the screen. The first user input is the file name in which the data to be plotted is stored. This should be the name of a file created by TAPFIL. The second user input is the number of plots desired on the graph. The maximum number allowable is five. The next two user inputs are the captions to be written at the bottom and on the left margin of the plot. The first caption allows 49 characters to be output; the second caption allows 16 characters.

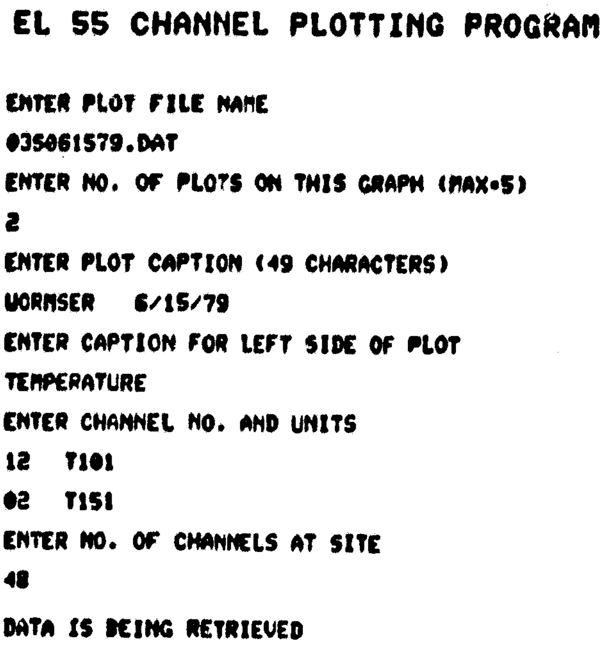
The fifth user input is the channel numbers to be plotted, along with appropriate units. The IPCL should be referenced to obtain the correct channel number of the measurement desired. After entering a channel number, the next input should be the mnemonic name and the units of that measurement. The carriage return should be depressed after both the name and the units. This is repeated until the desired number of channel numbers and units has been input.

The last input shown on Figure 5 is the number of channels of data at that site. This is subject to the same restrictions specified in the description of TAPFIL. After all user inputs are completed, an output statement will appear stating that the data is being retrieved.

After the data has been retrieved, a plot similar to the one in Figure 6 will appear on the screen. The grid is designed to fit onto a regular 8-1/2 by 11 inch sheet after xeroxing for use in reports to which these data are pertinent.

When the plot is completed, the program is in pause mode. The user must depress any key to continue execution. The next output to the screen is shown in Figure 7. This display gives the user a choice of five options for plotting. With the exception of option 2, the scaling option, the options are self-explanatory. The user must make his decision, enter the number of that option, and depress the carriage return.

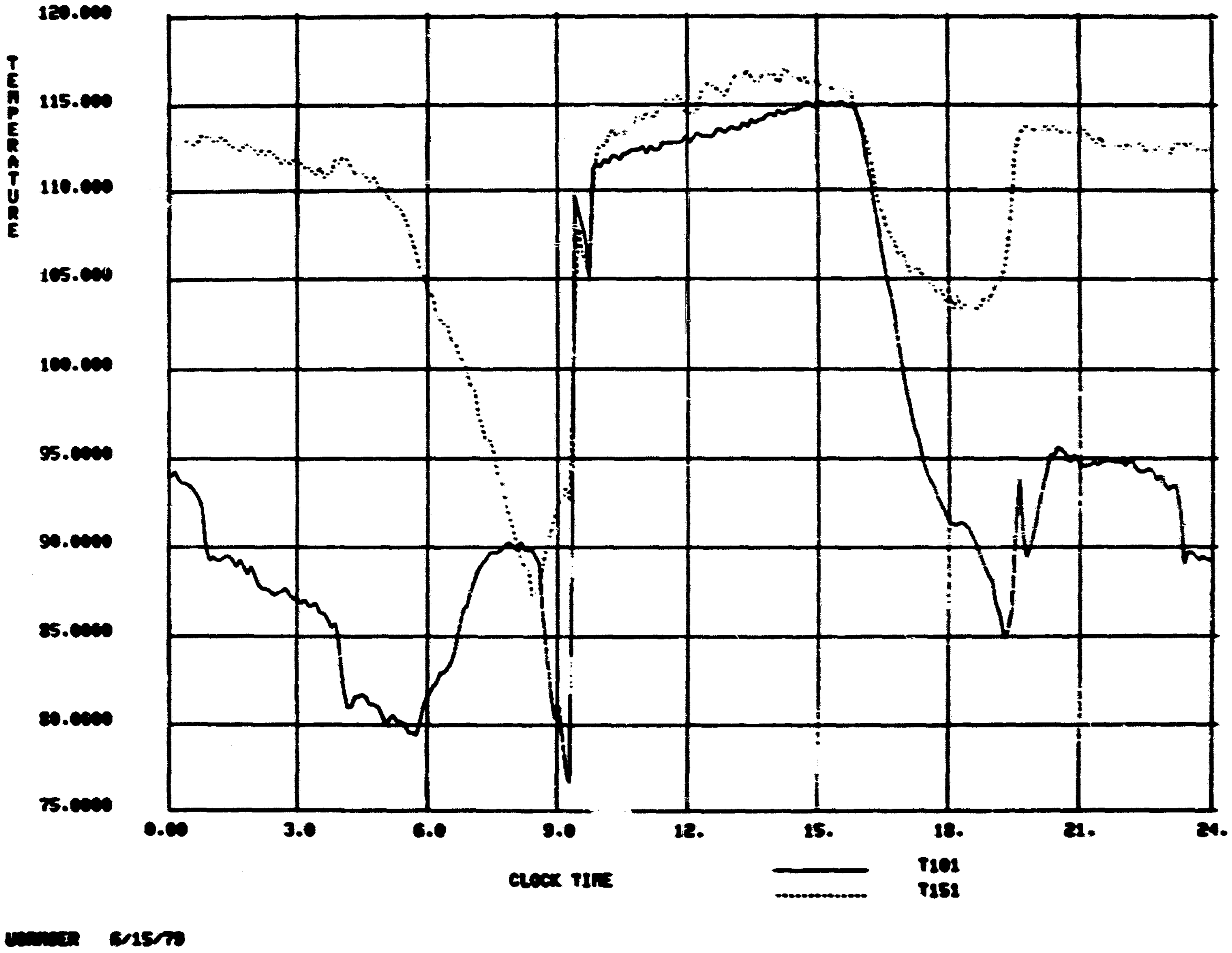
An example of the scaling options display is shown in Figure 8. This option gives the user the capability of windowing in on a certain section of the plot that was just made. The first input to the scaling option (XMAX) is the maximum abscissa value to be plotted; the second is the smallest abscissa value to be plotted. The third and fourth inputs are the maximum and minimum values for the ordinate to be plotted. After entering the last value in the scaling option, the plot in Figure 9 will appear on the screen. The plot will have the user input maximum and minimum values at the appropriate locations on the grid. After this plot has been completed, the program is in pause mode. Depressing a key will return control to the plotting options in Figure 7.

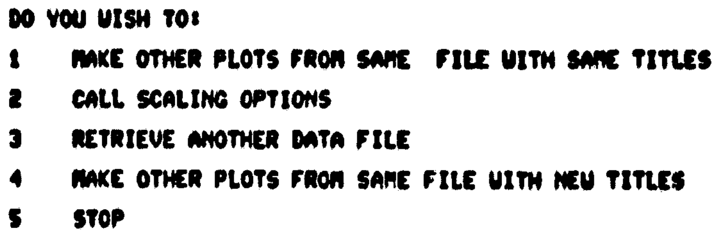


```
EL 55 CHANNEL PLOTTING PROGRAM

ENTER PLOT FILE NAME
035061579.DAT
ENTER NO. OF PLOTS ON THIS GRAPH (MAX=5)
2
ENTER PLOT CAPTION (49 CHARACTERS)
WORMSER   6/15/79
ENTER CAPTION FOR LEFT SIDE OF PLOT
TEMPERATURE
ENTER CHANNEL NO. AND UNITS
12   T101
02   T151
ENTER NO. OF CHANNELS AT SITE
48

DATA IS BEING RETRIEVED
```

FIGURE 5.   CHPLOT SCREEN DISPLAY

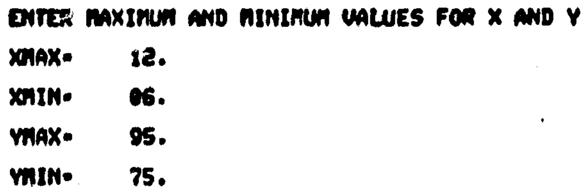
FIGURE 6. DATA PLOT





```
DO YOU WISH TO:
1     MAKE OTHER PLOTS FROM SAME  FILE WITH SAME TITLES
2     CALL SCALING OPTIONS
3     RETRIEVE ANOTHER DATA FILE
4     MAKE OTHER PLOTS FROM SAME FILE WITH NEW TITLES
5     STOP
```

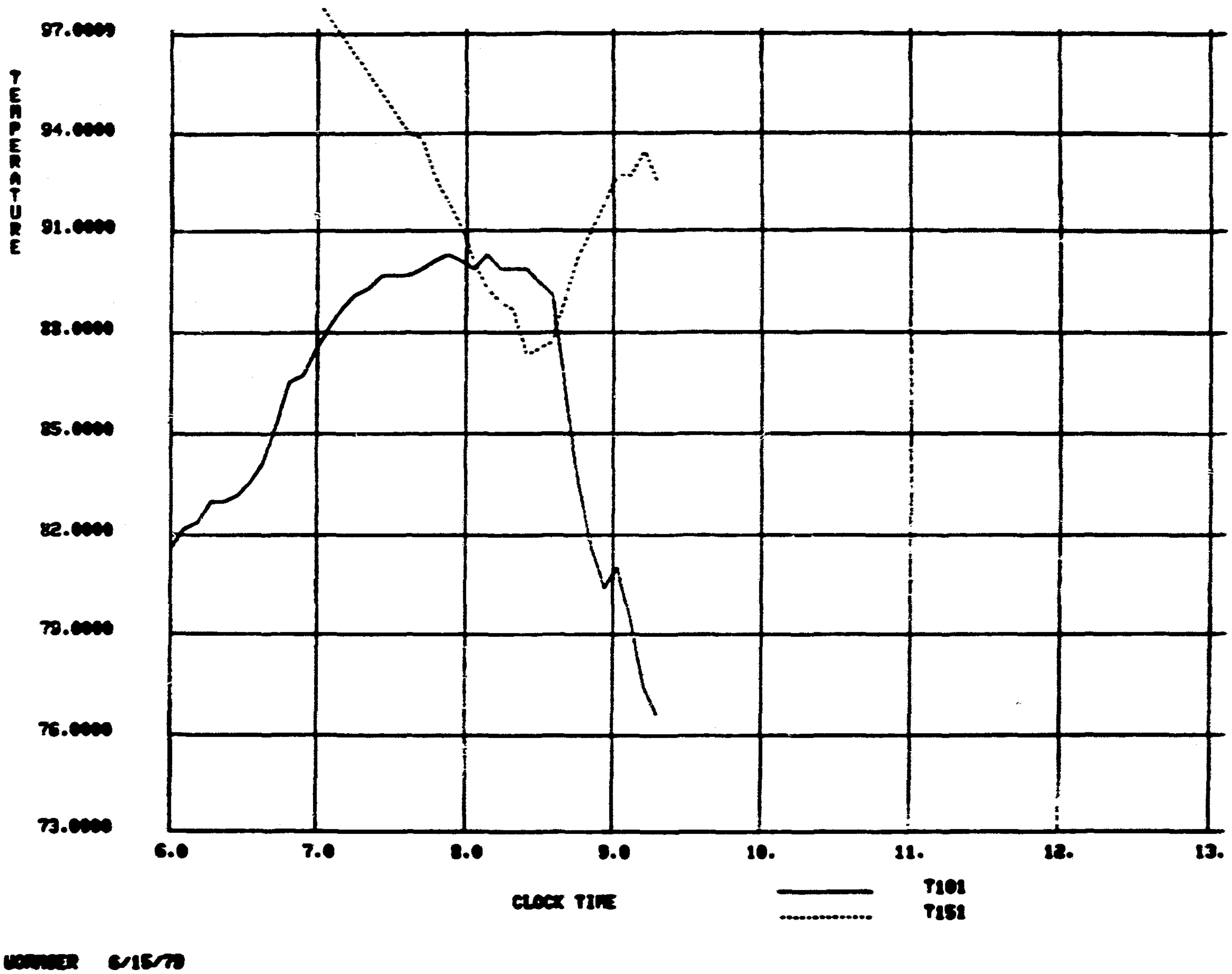
FIGURE 7. PLOTTING OPTIONS



```
ENTER MAXIMUM AND MINIMUM VALUES FOR X AND Y
XMAX=     12.
XMIN=     06.
YMAX=     95.
YMIN=     75.
```

FIGURE 8. SCALING OPTION DISPLAY

FIGURE 9. SCALED PLOT

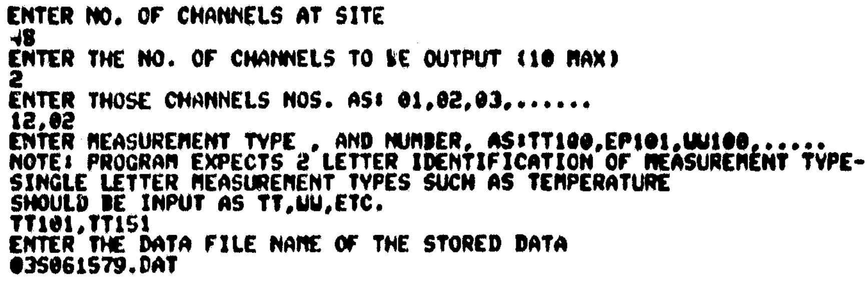


C. WRTCNL OPERATIONAL PROCEDURES

The purpose of WRTCNL is to list in a tabular form the data stored in a direct access unformatted file created by TAPFIL. The program has the capability to list the data for up to ten channels. Since WRTCNL does not contain any special routines, this program may be run from any terminal with the necessary capabilities.

The user inputs to this program are shown in Figure 10. The first input is the number of channels at the site. This is subject to the same restrictions specified in the description of TAPFIL. The next input is the number of channels to be output, with a maximum of ten. The third input is the channel numbers to be output. The IPCL should be referenced to obtain the correct channel numbers of the measurements desired. This is also the case for the next input, which is the measurement type and number. The last user input is the data file name of the stored data. This should be the name of a file created by TAPFIL.

After all inputs are completed, a listing of the requested data will be output to the line printer.



```
ENTER NO. OF CHANNELS AT SITE
48
ENTER THE NO. OF CHANNELS TO BE OUTPUT (10 MAX)
2
ENTER THOSE CHANNELS NOS. AS: 01,02,03,......
12,02
ENTER MEASUREMENT TYPE , AND NUMBER, AS:TT100,EP101,UU100,.....
NOTE: PROGRAM EXPECTS 2 LETTER IDENTIFICATION OF MEASUREMENT TYPE-
SINGLE LETTER MEASUREMENT TYPES SUCH AS TEMPERATURE
SHOULD BE INPUT AS TT,UU,ETC.
TT101,TT151
ENTER THE DATA FILE NAME OF THE STORED DATA
035061579.DAT
```

FIGURE 10. WRTCNL SCREEN DISPLAY

III. PROGRAM SOFTWARE IMPLEMENTATION

The previously described programs were written for use on a PDP 11/70 system with the 11-M operating version. Software capabilities of other computer systems could differ greatly; therefore, the software routines used in these programs will be explained in detail.

Three types of software routines are used by these programs - TEKPLOT Library routines, UTILITIES Library routines and Data Conversion routines. The first and largest set of routines used are the TEKPLOT routines. These routines are standard library software routines for the PDP 11/70. Libraries on other computer systems contain routines with different names that perform the same functions. Before these programs can be used on another computer, the TEKPLOT routines must be replaced by similar routines compatible to the computer used. The TEKPLOT routines are listed and described in Table 1.

The second type of software routines are in the UTILITIES Library. There are four of these routines; three of these are tape handling routines and the fourth is a data conversion routine. Table 2 contains a listing and description of these routines.

Bytes 19-210 of a record are converted from EBCDIC floating point words to PDP 11/70 words by a subroutine not included in the UTILITIES Library. This routine is known as CFP360 and is used as CFP360(DAT(1),48,0). The first argument represents a real data array dimensioned large enough to contain 48 words. DAT(1) represents the starting position to perform the required conversion and the second argument represents the number of words to be converted. One byte of data on the IBM 360 tape is equivalent to one byte on the PDP 11/70 and is represented by a LOGICAL*1 statement. On the PDP 11/70, one floating point word contains four bytes; therefore, 192 bytes of data are stored in the LOGICAL*1 data array from bytes 19-210. This indicates that 48 words are to be converted by the CFP360 routine. The last argument represents a byte-swap operation; a byte-swap is required for conversion from EBCDIC to PDP words, and is accomplished by setting the fourth argument to zero. The correlation between the LOGICAL*1 data array and the real data array used in CFP360 is done by an EQUIVALENCE statement. These statements are used:

```
DIMENSION      DAT(48)
LOGICAL*1      DATA(210)
EQUIVALENCE    (DATA(19),DAT(1))
```

This will equivalence the real dimensioned array DAT to the last 192 bytes of the LOGICAL*1 array DATA so that CFP360 can convert real data words.

The tape handling routines are standard software routines and can easily be compared to routines on another computer with a tape drive, but the two data conversion routines, EBCASC and CFP360, were developed for the MSFC PDP 11/70. Therefore, the comparable conversion routine to be used on another computer would need to convert EBCDIC characters to compatible characters for that computer system. Once the use of the above routines is understood, any experienced programmer should be able to implement these programs on any computer using FORTRAN IV PLUS.

TABLE 1

TEKPLOT ROUTINES

ALPHA-

Places terminal in alphanumeric mode, cursor at current (x,y) position.

AXIS(XLOW,YLOW,XLNG,YLNG,XTIC,YTIC,MARKX,MARKY)-

Draws X and Y axes. Each line must pass through the origin. If not, the routine returns with no action. After the axes are drawn, the user is left in vector mode positioned at the origin.

XLOW- starting position on X axis in scaled, rotated units. Must be the lowest point through which the X axis will pass.

YLOW- equivalent of XLOW pertaining to the Y axis.

XLNG- length of X axis in scaled units

YLNG- length of Y axis in scaled units

XTIC- distance between tic marks on X axis. If XTIC is negative, then is the distance between decades for a log axis.

YTIC- equivalent of XTIC pertaining to Y axis

MARKX- =0, no axis annotation for X axis
.GT. 0 , tic mark annotation for X axis
.LT. 0 , grid annotation for X axis

MARKY- equivalent of MARKX pertaining to the Y axis.

BEGIN(JBAUD)-

Initializes the terminal status area and the software character generator. Should always be the first subroutine called in any program.

JBAUD- the baud rate of the current installation.

CHOUT(J)-

Outputs a single character.

J- ASCII value of a character to be output to the T4014 scope

ERASE-

Erases the screen, places the terminal in alphanumeric mode, positions cursor to home position.

HOME-

Positions cursor to the upper left corner of the screen

POINT-

Places terminal in point plot mode and positions cursor to the current graphics position.

SCALE(XFACT,YFACT,XORG,YORG)-

Establishes X and Y scaling factors and X and Y origin

XFACT- X scaling factor (=tekpoints/scaled unit)

YFACT- Y scaling factor (=tekpoints/scaled unit)

XORG- X origin in tekpoints

YORG- Y origin in tekpoints

TPAUSE-

Causes a user to become I/O bound for an indefinite period of time. This allows the user to sit and wait while inspecting a plot without incurring CPU charges when working in a time sharing environment.

TPLOT(X,Y,IPEN,MARK)-

Plots an (x,y) coordinate in the current mode.

X- X value in scaled, rotated points

Y- Y value in scaled rotated points

IPEN- =0, dark plot (pen up)
.NE. 0, bright plot (pen down)

MARK- the number of a data mark from the following list:

0- no data mark
1- small x
2- small down arrow
3- small up arrow
4- small square
5- small triangle
6- small asterick
11- large x
12- large down arrow
13- large up arrow
14- large square
15- large triangle
16- large asterick

VECTOR-

Draws a smooth line between the current graphics position and the (x,y) point specified as a parameter to any of the plotting routines.

TABLE 2

UTILITIES ROUTINES

EBCASC(DATA,NC)-

Converts EBCDIC characters to ASCII characters for use on the PDP 11/70.

DATA- name of array that contains EBCDIC characters

NC- number of characters to be converted

SFUN(LUN,ISTAT,NF,IRC,IDIR)-

Performs special functions such as rewinding, forward spacing and backspacing records and files.

LUN- logical unit number of the tape drive

ISTAT- status code for error interpretation; use ISTAT(1) for starting location

NF- number of files on the IBM 360 tape to space

IRC- =0, normal operation
=1, end of file
=2, I/O error
=-1, end of tape

IDIR- represents which tape function to perform

=3, rewind
=10, forward space files
=13, backspace files

TGET(LUN,ISTAT,DATA,NW)-

Retrieves a block of data off the tape. One block contains 13020 bytes, or 6510 PDP 11/70 words since there are 2 bytes of LOGICAL*1 data per word on the PDP 11/70.

LUN- same as above

ISTAT- same as above

DATA- an array data is stored in; use DATA(1) for starting location

NW- number of words of data to be retrieved

TWAIT(LUN,ISTAT,NWORDS,IRC)-

Suspends task execution until the last I/O request for the specified LUN has been completed.

LUN- same as above

ISTAT- same as above

NWORDS- number of words of data remaining in the block after TGET retrieves the required number of words

IRC- same as above

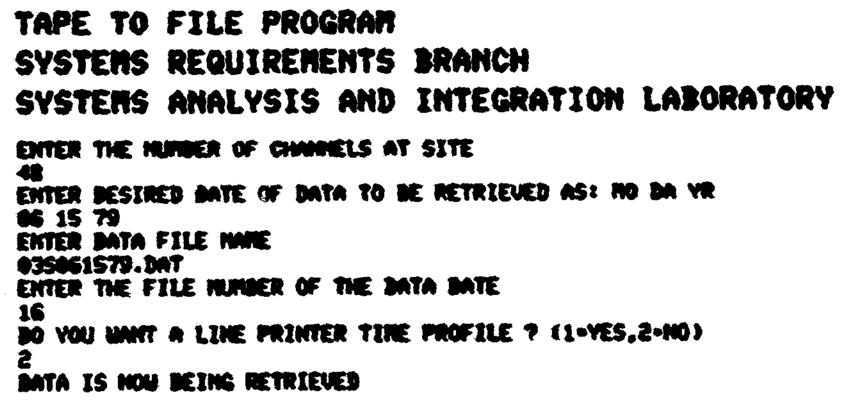
Of the three programs previously described, the understanding of TAPFIL is the most essential. CHPLOT and WRTCNL cannot be run until TAPFIL has created a direct access unformatted file from the data tape. Several other programs which require the use of TAPFIL to retrieve the data are currently in use at MSFC. These programs perform various analyses pertaining to collector efficiency.

CHPLOT can be used to plot daily data, thus enabling the user to find a day which appears to have usable data. WRTCNL can be used if a listing of this data is desired. After a good day has been found using CHPLOT, the data can then be used in the analysis of collector efficiency.

Another program currently in the development stage at MSFC will allow monthly data to be plotted without the prior use of TAPFIL. This program will read the data straight from the IBM tape and plot it one week at a time. The advantage of this program will be that the user can find good days without using TAPFIL to create numerous individual files. After the good days have been located, TAPFIL can then be used to create only the necessary files.

APPENDIX A

SAMPLE PROBLEM

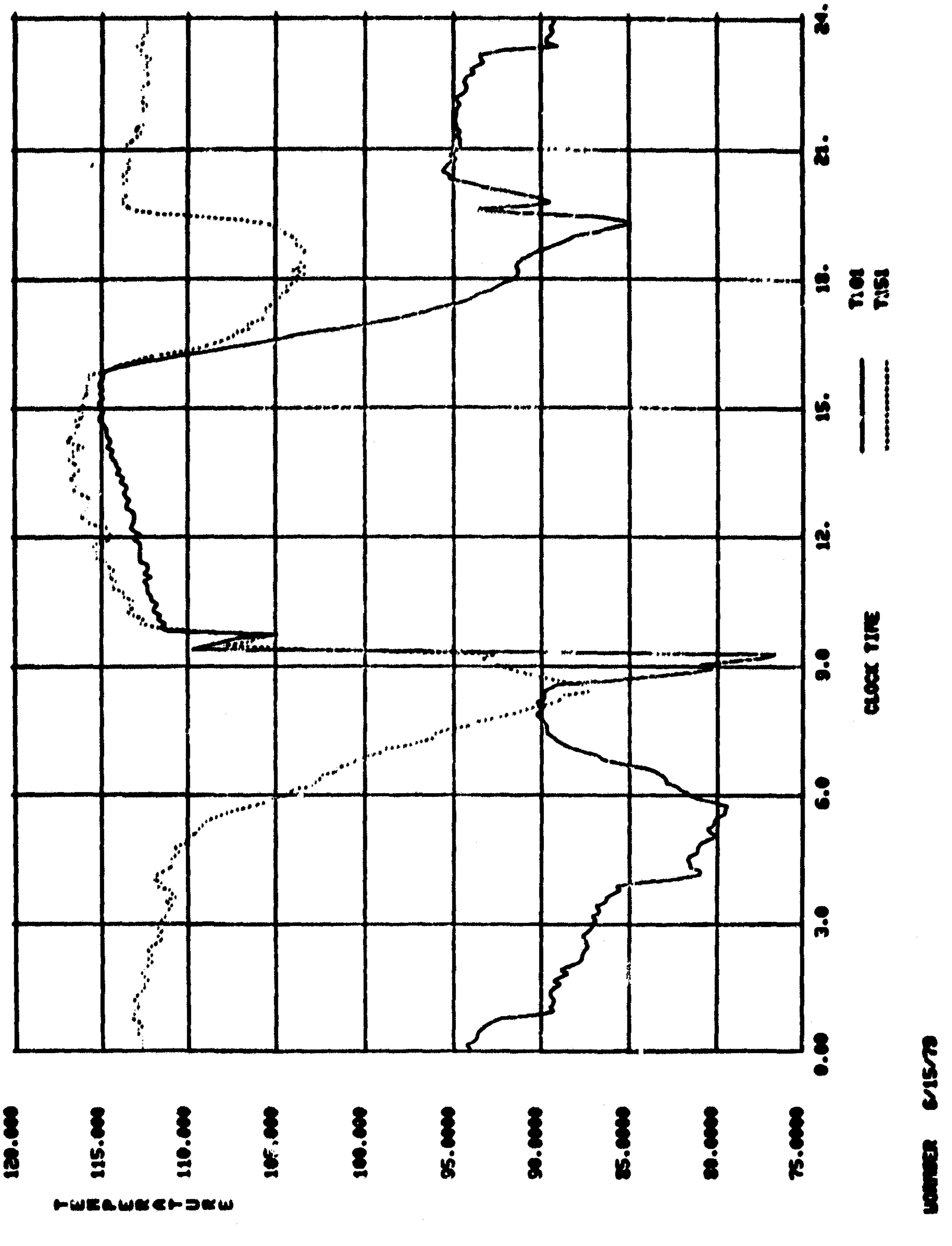


```
TAPE TO FILE PROGRAM
SYSTEMS REQUIREMENTS BRANCH
SYSTEMS ANALYSIS AND INTEGRATION LABORATORY

ENTER THE NUMBER OF CHANNELS AT SITE
48
ENTER DESIRED DATE OF DATA TO BE RETRIEVED AS: MO DA YR
06 15 79
ENTER DATA FILE NAME
03S061579.DAT
ENTER THE FILE NUMBER OF THE DATA DATE
16
DO YOU WANT A LINE PRINTER TIME PROFILE ? (1=YES,2=NO)
2
DATA IS NOW BEING RETRIEVED
```

TPFIL TIME PROFILE

```
06/15  00:03:40   06/15  04:08:30   06/15  08:08:30   06/15  12:08:30   06/15  16:08:30   06/15  20:08:30
06/15  00:09:00   06/15  04:13:50   06/15  08:13:50   06/15  12:13:50   06/15  16:13:50   06/15  20:13:50
06/15  00:14:20   06/15  04:19:10   06/15  08:19:10   06/15  12:19:10   06/15  16:19:10   06/15  20:19:10
06/15  00:19:40   06/15  04:24:30   06/15  08:24:30   06/15  12:24:30   06/15  16:24:30   06/15  20:24:30
06/15  00:25:00   06/15  04:29:50   06/15  08:29:50   06/15  12:29:50   06/15  16:29:50   06/15  20:29:50
06/15  00:30:20   06/15  04:35:10   06/15  08:35:10   06/15  12:35:10   06/15  16:35:10   06/15  20:35:10
06/15  00:35:40   06/15  04:40:30   06/15  08:40:30   06/15  12:40:30   06/15  16:40:30   06/15  20:40:30
06/15  00:41:00   06/15  04:45:50   06/15  08:45:50   06/15  12:45:50   06/15  16:45:50   06/15  20:45:50
06/15  00:46:20   06/15  04:51:10   06/15  08:51:10   06/15  12:51:10   06/15  16:51:10   06/15  20:51:10
06/15  00:51:40   06/15  04:56:30   06/15  08:56:30   06/15  12:56:30   06/15  16:56:30   06/15  20:56:30
06/15  00:57:00   06/15  05:01:50   06/15  09:01:50   06/15  13:01:50   06/15  17:01:50   06/15  21:01:50
06/15  01:02:20   06/15  05:07:10   06/15  09:07:10   06/15  13:07:10   06/15  17:07:10   06/15  21:07:10
06/15  01:07:40   06/15  05:12:30   06/15  09:12:30   06/15  13:12:30   06/15  17:12:30   06/15  21:12:30
06/15  01:13:00   06/15  05:17:50   06/15  09:17:50   06/15  13:17:50   06/15  17:17:50   06/15  21:17:50
06/15  01:18:20   06/15  05:23:10   06/15  09:23:10   06/15  13:23:10   06/15  17:23:10   06/15  21:23:10
06/15  01:23:40   06/15  05:28:30   06/15  09:28:30   06/15  13:28:30   06/15  17:28:30   06/15  21:28:30
06/15  01:29:00   06/15  05:33:50   06/15  09:33:50   06/15  13:33:50   06/15  17:33:50   06/15  21:33:50
06/15  01:34:20   06/15  05:39:10   06/15  09:39:10   06/15  13:39:10   06/15  17:39:10   06/15  21:39:10
06/15  01:39:40   06/15  05:44:30   06/15  09:44:30   06/15  13:44:30   06/15  17:44:30   06/15  21:44:30
06/15  01:45:00   06/15  05:49:50   06/15  09:49:50   06/15  13:49:50   06/15  17:49:50   06/15  21:49:50
06/15  01:50:20   06/15  05:55:10   06/15  09:55:10   06/15  13:55:10   06/15  17:55:10   06/15  21:55:10
06/15  01:55:40   06/15  06:00:30   06/15  10:00:30   06/15  14:00:30   06/15  18:00:30   06/15  22:00:30
06/15  01:59:58   06/15  06:05:50   06/15  10:05:50   06/15  14:05:50   06/15  18:05:50   06/15  22:05:50
06/15  02:05:18   06/15  06:11:10   06/15  10:11:10   06/15  14:11:10   06/15  18:11:10   06/15  22:11:10
06/15  02:10:38   06/15  06:16:30   06/15  10:16:30   06/15  14:16:30   06/15  18:16:30   06/15  22:16:30
06/15  02:15:58   06/15  06:21:50   06/15  10:21:50   06/15  14:21:50   06/15  18:21:50   06/15  22:21:50
06/15  02:21:18   06/15  06:27:10   06/15  10:27:10   06/15  14:27:10   06/15  18:27:10   06/15  22:27:10
06/15  02:26:38   06/15  06:32:30   06/15  10:32:30   06/15  14:32:30   06/15  18:32:30   06/15  22:32:30
06/15  02:37:44   06/15  06:37:50   06/15  10:37:50   06/15  14:37:50   06/15  18:37:50   06/15  22:37:50
06/15  02:43:10   06/15  06:43:10   06/15  10:43:10   06/15  14:43:10   06/15  18:43:10   06/15  22:43:10
06/15  02:48:30   06/15  06:48:30   06/15  10:48:30   06/15  14:48:30   06/15  18:48:30   06/15  22:48:30
06/15  02:53:50   06/15  06:53:50   06/15  10:53:50   06/15  14:53:50   06/15  18:53:50   06/15  22:53:50
06/15  02:59:19   06/15  06:59:10   06/15  10:59:10   06/15  14:59:10   06/15  18:59:10   06/15  22:59:10
06/15  03:04:30   06/15  07:04:30   06/15  11:04:30   06/15  15:04:30   06/15  19:04:30   06/15  23:04:30
06/15  03:09:50   06/15  07:09:50   06/15  11:09:50   06/15  15:09:50   06/15  19:09:50   06/15  23:09:50
06/15  03:15:10   06/15  07:15:10   06/15  11:15:10   06/15  15:15:10   06/15  19:15:10   06/15  23:15:10
06/15  03:20:30   06/15  07:20:30   06/15  11:20:30   06/15  15:20:30   06/15  19:20:30   06/15  23:20:30
06/15  03:25:50   06/15  07:25:50   06/15  11:25:50   06/15  15:25:50   06/15  19:25:50   06/15  23:25:50
06/15  03:31:10   06/15  07:31:10   06/15  11:31:10   06/15  15:31:10   06/15  19:31:10   06/15  23:31:10
06/15  03:36:30   06/15  07:36:30   06/15  11:36:30   06/15  15:36:30   06/15  19:36:30   06/15  23:36:30
06/15  03:41:50   06/15  07:41:50   06/15  11:41:50   06/15  15:41:50   06/15  19:41:50   06/15  23:41:50
06/15  03:47:10   06/15  07:47:10   06/15  11:47:10   06/15  15:47:10   06/15  19:47:10   06/15  23:47:10
06/15  03:52:30   06/15  07:52:30   06/15  11:52:30   06/15  15:52:30   06/15  19:52:30   06/15  23:52:30
06/15  03:57:50   06/15  07:57:50   06/15  11:57:50   06/15  15:57:50   06/15  19:57:50   06/15  23:57:50
06/15  04:03:10   06/15  08:03:10   06/15  12:03:10   06/15  16:03:10   06/15  20:03:10
THE DATA HAS BEEN STORED WITH  260  RECORDS
```

```
EL 55 CHANNEL PLOTTING PROGRAM

ENTER PLOT FILE NAME

035061579.DAT

ENTER NO. OF PLOTS ON THIS GRAPH (MAX=5)

2

ENTER PLOT CAPTION (49 CHARACTERS)

UORMSER   6/15/79

ENTER CAPTION FOR LEFT SIDE OF PLOT

TEMPERATURE

ENTER CHANNEL NO. AND UNITS

12   T101

02   T151

ENTER NO. OF CHANNELS AT SITE

48

DATA IS BEING RETRIEUED
```

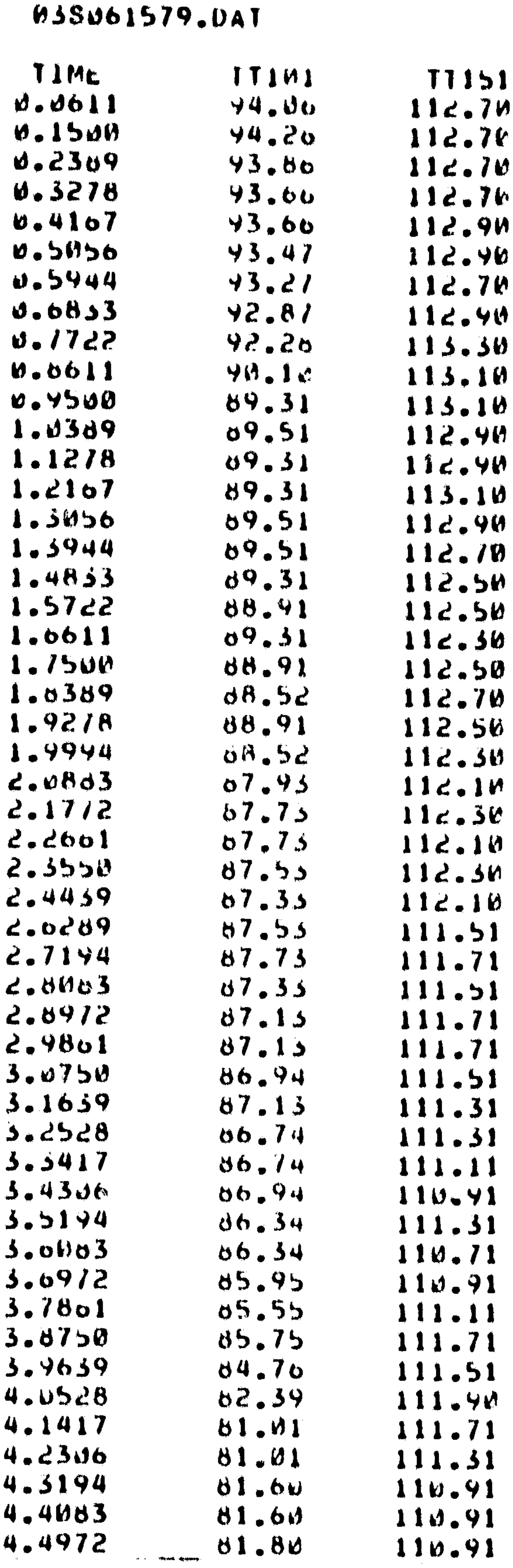


120.000
115.000
110.000
105.000
100.000
95.000
90.000
85.000
80.000
75.000
TEMPERATURE
0.00
3.0
6.0
9.0
12.
15.
18.
21.
24.
CLOCK TIME
T101
T151
WARMER 6/15/79

```
DO YOU UISH TO:
1    MAKE OTHER PLOTS FROM SAME  FILE UITH SAME TITLES
2    CALL SCALING OPTIONS
3    RETRIEVE ANOTHER DATA FILE
4    MAKE OTHER PLOTS FROM SAME FILE UITH NEU TITLES
5    STOP
```

WRTCNL PROGRAM

```
ENTER NO. OF CHANNELS AT SITE
48
ENTER THE NO. OF CHANNELS TO BE OUTPUT (10 MAX)
2
ENTER THOSE CHANNELS NOS. AS: 01,02,03,......
12,02
ENTER MEASUREMENT TYPE , AND NUMBER, AS:TT100,EP101,UU100,.....
NOTE: PROGRAM EXPECTS 2 LETTER IDENTIFICATION OF MEASUREMENT TYPE-
SINGLE LETTER MEASUREMENT TYPES SUCH AS TEMPERATURE
SHOULD BE INPUT AS TT,UU,ETC.
TT101,TT151
ENTER THE DATA FILE NAME OF THE STORED DATA
03S061579.DAT
```

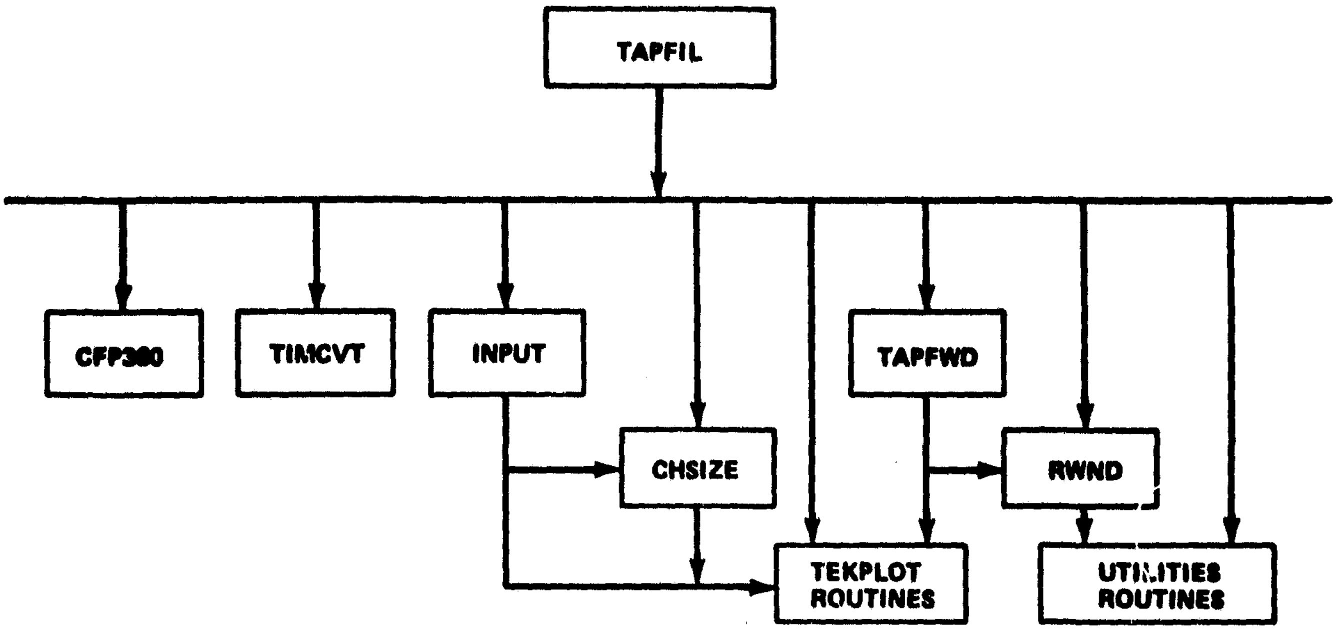


```
03S061579.DAT

  TIME          TT101          TT151
0.0611          94.06         112.70
0.1500          94.26         112.70
0.2389          93.86         112.70
0.3278          93.66         112.70
0.4167          93.66         112.90
0.5056          93.47         112.90
0.5944          93.27         112.70
0.6833          92.87         112.90
0.7722          92.26         113.30
0.8611          90.12         113.10
0.9500          89.31         113.10
1.0389          89.51         112.90
1.1278          89.31         112.90
1.2167          89.31         113.10
1.3056          89.51         112.90
1.3944          89.51         112.70
1.4833          89.31         112.50
1.5722          88.91         112.50
1.6611          89.31         112.30
1.7500          88.91         112.50
1.8389          88.52         112.70
1.9278          88.91         112.50
1.9944          88.52         112.30
2.0883          87.93         112.10
2.1772          87.73         112.30
2.2661          87.73         112.10
2.3550          87.53         112.30
2.4439          87.33         112.10
2.6289          87.53         111.51
2.7194          87.73         111.71
2.8083          87.33         111.51
2.8972          87.13         111.71
2.9861          87.13         111.71
3.0750          86.94         111.51
3.1639          87.13         111.31
3.2528          86.74         111.31
3.3417          86.74         111.11
3.4306          86.94         110.91
3.5194          86.34         111.31
3.6083          86.34         110.71
3.6972          85.95         110.91
3.7861          85.55         111.11
3.8750          85.75         111.71
3.9639          84.76         111.51
4.0528          82.39         111.90
4.1417          81.01         111.71
4.2306          81.01         111.31
4.3194          81.60         110.91
4.4083          81.60         110.91
4.4972          81.80         110.91
```

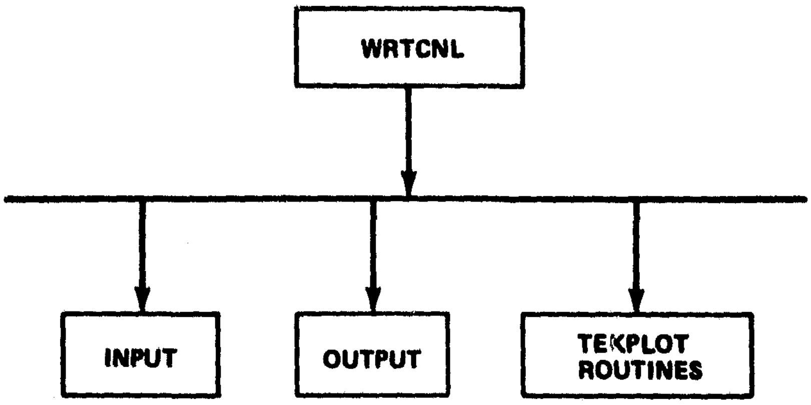
This is a sample of the printout produced by WRTCNL.

APPENDIX B

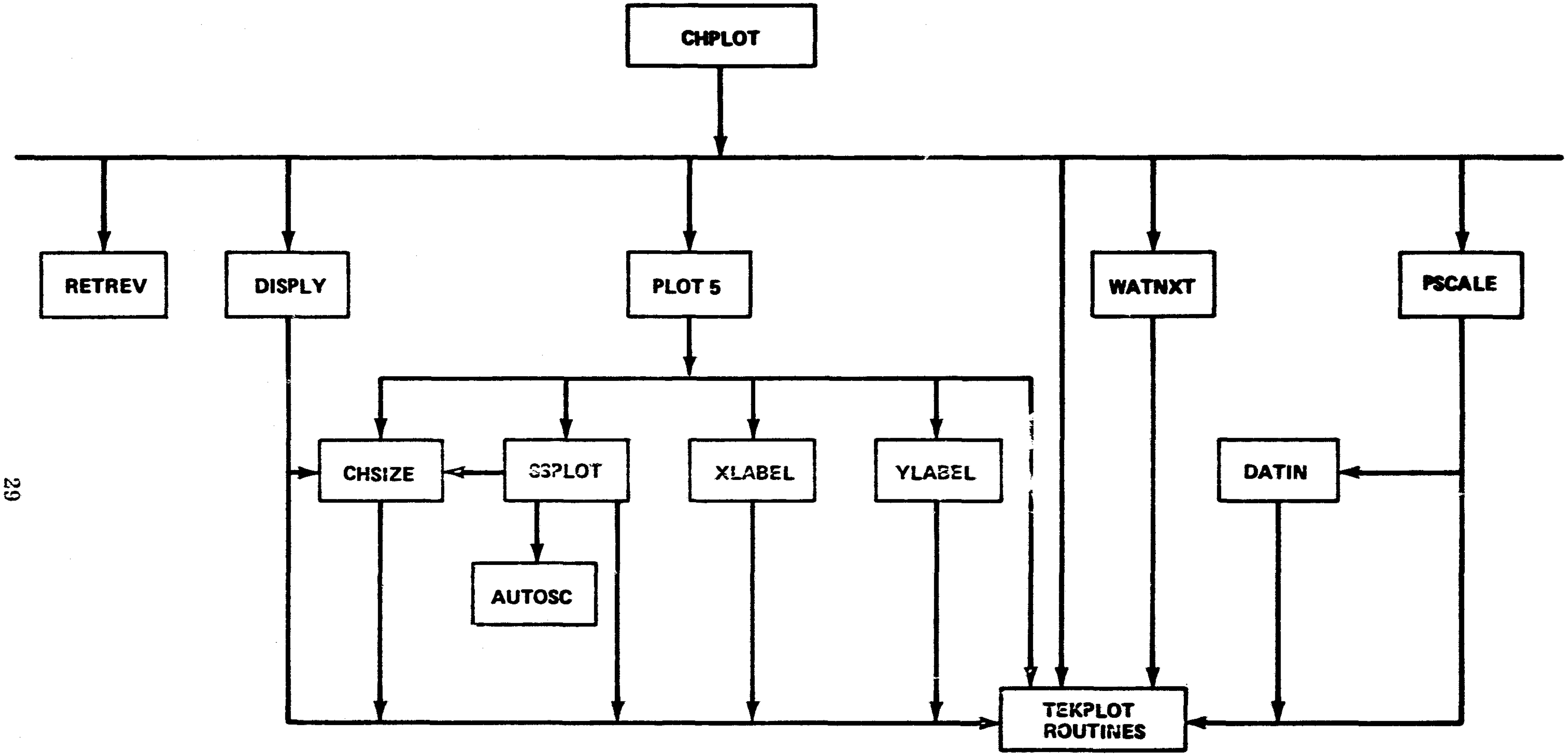
FLOWCHARTS and REFERENCE TABLES



TAPFIL FLOWCHART



WRTCNL FLOWCHART



CHPLOT FLOWCHART

TAPFIL SUBROUTINE LIST

CFP360-	converts EBCDIC floating point words to PDP 11/70 words
CHSIZE-	changes character size on a TEKTRONIX 4014 display scope
INPUT-	reads user inputs regarding site data imformation
RWND-	rewinds tape
TAPFWD-	moves tape to correct file
TEKPLOT-	routines associated with use of the TEKTRONIX 4014 scope
TIMCVT-	converts ASCII characters to decimal equivalents
UTILITIES-	tape handling and data conversion routines

CHPLOT SUBROUTINE LIST

AUTOSC-	sets grid for the plot to the proper scale
CHSIZE-	changes character size on TEKTRONIX 4014 scope
DATIN-	retrieves floating point data from the scope
DISPLY-	reads user input regarding site data information for plotting
PLOT5-	builds plots of data
PSCALE-	allows the user to plot portions of data for a particular day
RETREV-	retrieves the file to be plotted from the tape
SSPLOT-	draws axis and grid, plots and labels data
TEKPLOT-	routines associated with use of the TEKTRONIX 4014 scope

WRTCNL SUBROUTINE LIST

INPUT-	reads user input regarding site data information for printing
OUTPUT-	prints site data information
TEKPLOT-	routines associated with use of the TEKTRONIX 4014 scope

APPENDIX C

TAPFIL SOURCE LISTING

```
;THIS ROUTINE CONVERTS IBM 360 32-BIT FLOATING POINT NUMBERS
;TO PDP-11 FORMAT.  THE FORTRAN(F4P) CALLING SEQUENCE IS,
;
;
;               CALL CFP360(ARRAY,N,ISWAP)
;
;
;WHERE,
;
;               ARRAY - STARTING ADDRESS OF AN ARRAY CONTAINING
;                       THE 360 FORMAT DATA.  IT DOESN'T MATTER IF
;                       IT IS INTEGER OR REAL.  IT ONLY REQUIRES THAT THE
;                       360 DATA IS IN 32-BIT PAIRS.
;              N - THE INTEGER NUMBER OF 32-BIT 360 WORDS TO CONVERT.
;              ISWAP - 0 FOR BYTE SWAP FOR DEC
;                      1 FOR NO BYTE SWAP
;
;
;
      .TITLE  CFP360
      .GLOBL  CFP360
CFP360:
      MOV    R0,-(SP)
      MOV     R1,-(SP)
      MOV     R2,-(SP)
      MOV     R3,-(SP)
      MOV     R4,-(SP)
      MOV     R5,-(SP)
      MOV      @4(R5),R4      ;FETCH NO. OF 32-BIT WORDS TO CONVERT
      MOV     2(R5),R0       ;FETCH ARG. ADDRESS TO R0
      MOV     @6(R5),SWAP    ;FETCH SWAP ARGUMENT
5$:
      CLR     FLAG           ;CLEAR NEG. FLAG
      MOV     (R0)+,R1       ;FIRST 16 BITS TO R1
      MOV     (R0),R2        ;LAST 16 BITS TO R2
      TST     R1             ;TEST FOR ZERO
      BNE     7$             ;BRANCH IF NOT ZERO
```

```
        TST     R2                 ;TEST FOR ZERO
        BEQ     8$                 ;BRANCH IF ZERO
  7$:
        TST     SWAP               ;TEST FOR SWAP
        BNE     6$                 ;BRANCH IF NO BYTE SWAP
        SWAB    R1                  ;BYTE SWAP FOR DEC
        SWAB    R2                   ;SAME
 6$:
        TST     R1                 ;TEST FOR NEGATIVE
        BGE     1$                 ;NO
        BIS     #100000,FLAG        ;YES, SET FLAG
 1$:
        MOV     R1,R3
        CLRB    R3                 ;CLEAR MANTISSA PORTION
        ASL     R3                 ;SHIFT LEFT 1
        BIC     #140000,R3         ;CLEAR BIAS & SIGN
        BIT     #40000,R1          ;TEST BIAS BIT
        BEQ     2$                 ;OFF
        BIS     #40000,R3          ;ON , SET IT
  2$:
        BIC     #177400,R1         ;CLEAR EXPONENT PORTION
 3$:   TSTB     R1                 ;SEE IF BINARY NORMALIZED
        BLT     4$                 ;YES
        ASL     R1                 ;NO, SHIFT LEFT
        ASL     R2                 ;R2,SHIFT,R2
        ADC     R1                 ;ADD CARRY TO R1
        SUB     #200,R3            ;ADD BIT TO EXPONENT
        BR      3$
 4$:
        BIC     #200,R1            ;HIDDEN BIT
        ADD     R3,R1              ;COMBINE EXP. AND MANTISSA
        BIS     FLAG,R1             ;SET NEGATIVE BIT
  8$:
        MOV     R2,(R0)          ;STORE BACK IN ARGUMENT
        MOV     R1,-(R0)
        ADD      #4,R0           ;POSITION PONTER TO NEXT 32 BIT WORD
        DEC      R4                ;DECREMENT, ZERO FLAG SET IF THAT IS ALL
        BGT      5$                ;NO, THAT IS NOT ALL
```

```
          MOV     (SP)+,R5
          MOV     (SP)+,R4
          MOV     (SP)+,R3
          MOV     (SP)+,R2
          MOV     (SP)+,R1
          MOV      (SP)+,R0
          RTS     PC
FLAG:     .WORD    0
SWAP:     .WORD    0
          .END
PIP>
```

```
      SUBROUTINE CHSIZE(N)
C***   THIS ROUTINE CHANGES CHARACTER SIZE ON A TEKTRONIX 4014-1 DISPLAY
C***   SCOPE ACCORDING TO N AS FOLLOUS:  N=1 - LARGEST
C***                                     N=2 - NEXT SMALLER
C***                                     N=3 - NEXT SMALLER
C***                                     N=4 - SMALLEST
      NP=N+55
      CALL CHOUT(27)
      CALL CHOUT(NP)
      RETURN
      END
PIP>
```

```
      SUBROUTINE INPUT(NCH,NFILE,ILP,IREP)
      LOGICAL*1 IFILE(30),IDATE(6)
      COMMON/FILE/IFILE
      COMMON/DATE/IDATE
      COMMON/WHERE/XS,YS
      YS=730.
      XS=50.
      CALL ALPHA
      LUN=7
      IF(IREP.EQ.1) GO TO 22
      CALL CHSIZE(2)
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,5)
5     FORMAT(1H+,'TAPE TO FILE PROGRAM')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,6)
6     FORMAT(1H+,'SYSTEMS REGUIREMENTS BRANCH')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,8)
8     FORMAT(1H+,'SYSTEMS ANALYSIS AND INTEGRATION LABORATORY')
      YS=YS-30.
      CALL CHSIZE(3)
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,15)
15    FORMAT(1H+,'ENTER THE NUMBER OF CHANNELS AT SITE')
      YS=YS-15.
      CALL TPLOT(XS,YS,0,0)
      READ(1,20) NCH
20    FORMAT(I4)
      YS=YS-15.
 22   CALL TPLOT(XS,YS,0,0)
      WRITE(1,25)
25    FORMAT(1H+,'ENTER DESIRED DATE OF DATA TO BE RETREIVED
     * AS: MO DA YR ')
      YS=YS-15.
      CALL TPLOT(XS,YS,0,0)
      READ(1,30) (IDATE(I),I=3,6),(IDATE(I),I=1,2)
 30   FORMAT(3(2A1,1X))
      YS=YS-15.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,35)
 35   FORMAT(1H+,'ENTER DATA FILE NAME')
      YS=YS-15.
      CALL TPLOT(XS,YS,0,0)
      READ(1,40) (IFILE(I),I=1,29)
 40   FORMAT(29A1)
      YS=YS-15.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,45)
 45   FORMAT(1H+,'ENTER THE FILE NUMBER OF THE DATA DATE')
      YS=YS-15.
      CALL TPLOT(XS,YS,0,0)
      READ(1,20) NFILE
      YS=YS-15.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,50)
 50   FORMAT(1H+,'DO YOU WANT A LINE PRINTER TIME PROFILE
     * ? (1=YES,2=NO)')
      YS=YS-15.
      CALL TPLOT(XS,YS,0,0)
      READ(1,20) ILP
      YS=YS-15.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,55)
 55   FORMAT(1H+,'DATA IS NOW BEING RETREIVED')
      RETURN
      END
PIP>
```

```
      SUBROUTINE RWND(ISTAT)
      DIMENSION ISTAT(1)
      LUN=7
      CALL SFUN(LUN,ISTAT,IDAT,IRC,3)
      CALL TWAIT(LUN,ISTAT,NWORDS,IRC)
      RETURN
      END
PIP>
```

TAPFIL:

```
C**********    SECTION A    ****************************
C**********   SET UP STORAGE AND INPUTS
      DIMENSION DAT1(48),DAT2(48),ISTAT(4)
      LOGICAL*1 DATA(13020),DATAR1(210),DATAR2(210),DATAR(210)
      LOGICAL*1 IFILE(30),IDATE(6),ICHK(2)
      COMMON/TIM/DATAR
      COMMON/FILE/IFILE
      COMMON/DATE/IDATE
      COMMON/WHERE/XS,YS
      EQUIVALENCE(DAT1(1),DATAR1(19))
      EQUIVALENCE(DAT2(1),DATAR2(19))
      DATA IFILE(30)/0/
      CALL BEGIN(9600)
      LUN=7
      CALL ASSIGN(LUN,'MT0:')
      MFILE=1
      IREP=2
    2 NRDAT=1
      JRK=0
      YK=40.
      XM=50.
      NORECS=0
      CALL ALPHA
      CALL ERASE
      CALL SCALE(1.0,1.0,0.0,0.0)
      CALL INPUT(NCH,NFILE,ILP,IREP)
C**********   ROLL TAPE FORWARD TO APPROPRIATE FILE
      CALL TAPFWD(NFILE,MFILE,ISTAT(1),IREP)
      MFILE=NFILE
      CALL CHSIZE(2)
      CALL ERASE
      YS=730.
      CALL TPLOT(305.,YS,0,0)
      WRITE(1,5)
 5    FORMAT(1H+,'TPFIL TIME PROFILE')
      CALL CHSIZE(3)
      YS=YS-15.
      YSAV=YS
      NT=NCH+1
C***********   OPEN PRINT FILE AND STORAGE FILE
C***********   ALSO PRINT LINE PRINTER HEADINGS
      IF(ILP.EQ.1)OPEN(UNIT=3,NAME='PRINT.LST',TYPE='NEW',ERR=600)
      OPEN(UNIT=2,NAME=IFILE,TYPE='NEW',ACCESS='DIRECT',
     *RECORDSIZE=NT,ASSOCIATEVARIABLE=NRDAT,MAXREC=270,ERR=600)
      IF(ILP.EQ.1) WRITE(3,7)
 7    FORMAT(1H1,T2,'TPFIL TIME PROFILE')
      IF(ILP.EQ.1) WRITE(3,8) (IFILE(I),I=1,29)
 8    FORMAT(T2,29A1)
      IF(ILP.EQ.1) WRITE(3,9)
 9    FORMAT(T7,'MO',T11,'DAY',T15,'HR',T19,'MIN',T23,'SEC')
C***********    SECTION B    ****************************
C***********  CHECK TAPE DATE AND SITE SUBTYPE
C***********  THE NEXT 8 STATEMENTS FINDS INPUT DATE ON TAPE
 10   CALL TGET(LUN,ISTAT,DATA(1),6510)
      CALL TWAIT(LUN,ISTAT,NWORDS,IRC)
      IF(IRC.EQ.1) GO TO 900
      ICHK(1)=DATA(12819)
      ICHK(2)=DATA(12820)
      CALL EBCASC(ICHK(1),2)
      IF(ICHK(1).GE.IDATE(5).AND.ICHK(2).GE.IDATE(6)) GO TO 20
      GO TO 10
C***********   NOW CHECK DATA BLOCK FOR INPUT DATE AND
C***********   CALCULATE THE STARTING RECORD POSITION
 20   NRC=1
      J=1
 23   DO 25 I=1,210
      DATAR1(I)=DATA(J)
 25   J=J+1
      CALL EBCASC(DATAR1(1),18)
      CALL CFP360(DAT1(1),48,0)
      IF(DATAR1(9).EQ.IDATE(5).AND.DATAR1(10).EQ.IDATE(6)) GO TO 40
      IF(J.GE.13020) NRC=1
      IF(J.GE.13020) GO TO 10
      NRC=NRC+1
      GO TO 23
C**********    NOW DIRECT CONTROL TO SITE SUBTYPE DATA RETRIEVAL
 40   NSUB=DATAR1(17)-48
      NSUB1=DATAR1(17)
      GO TO (50,50,60,60,60,60) NSUB
C**********    SECTION C    ************************************
C**********    DATA RETREIVAL FOR SDAS SUBTYPE 1  ***************************
C**********    CALCULATE STARTING POSITION FOR RECORD(NRC)
 50   NSTRT=1
      NSTRT=NSTRT+((NRC*210)-210)
      NSTOP=NSTRT+209
C**********    INSERT ACTUAL RECORD IN OPERATING RECORD(DATAR1)
      J=1
 53   DO 55 I=NSTRT,NSTOP
      DATAR1(J)=DATA(I)
 55   J=J+1
C**********    INSERT DATAR1 IN DATAR FOR USE WITH TIME CONVERSION(TIMCVT)
      DO 56 I=1,18
 56   DATAR(I)=DATAR1(I)
C**********    CONVERT TIME RECORD AND DATA RECORD
      CALL EBCASC(DATAR(1),18)
      CALL EBCASC(DATAR1(1),18)
C**********    CONVERT EBCDIC WORDS TO PDP WORDS
      CALL CFP360(DAT1(1),48,0)
C**********    CHECK DATE ON TAPE(DATAR1(9 AND 10)) AGAINST INPUT DATE (IDATE(5 A
ND 6))
      IF(DATAR1(9).NE.IDATE(5).OR.DATAR1(10).NE.IDATE(6)) GO TO 500
C**********    CONVERT TIME FROM ASCII CHARACTERS TO DECIMAL EQUIVALENT
      CALL TIMCVT(TIME)
C**********    NEXT 3 STATEMENTS CHECK FOR ERRORS IN TIME CONVERSION
      IF(NORECS.EQ.0) GO TO 57
      CHK=TIME-TSAV
      IF(CHK) 800,800,57
C**********    OUTPUT CONVERTED RECORD TO SYSTEMS DISK FOR STORAGE
 57   WRITE(2'NRDAT,ERR=700) TIME,(DAT1(I),I=1,48)
C**********    LINE PRINTER OUTPUT CHECK
      D=FLOAT(NSTRT)
      IF(ILP.EQ.1) WRITE(3,59) (DATAR1(I),I=7,16),NRC,D
```

```
C**********    SITE DATA TIME PROFILE OUTPUT TO SCREEN
      YS=YS-15.
      CALL TPLOT(XM,YS,0,0)
      WRITE(1,69) (DATAR1(I),I=7,16)
C**********    NEXT 5 STATEMENTS CHECK SCREEN LOCATION FOR PROPER OUTPUT
      IF(YS.LT.45.) XM=XM+150.
      IF(YS.LT.45.) JRK=JRK+1
      IF(XM.GT.900.) XM=50.
      IF(YS.LT.45.) YS=YSAV
      IF(YS.LT.45..AND.JRK.GT.6) YS=700.
C**********    INCREMENT THE NUMBER OF STORED RECORDS(NORECS)
      NORECS=NORECS+1
 59   FORMAT(T2,'TIME='5(2A1,2X),3X,'RECORD NO.=',3X,I4,3X,'START POSITI
     *ON =',3X,F8.0)
C**********    CHECK FOR BLOCK DATA USED UP,IF(NRC.GT.62) GET ANOTHER BLOCK
      IF(NRC.GE.62) GO TO 99
C**********    INCREMENT THE RECORD NUMBER(NRC)
      NRC=NRC+1
      TSAV=TIME
C**********   270 RECORDS IN 1 DAY, GREATER THAN THAT MEANS STOP
      IF(NORECS.GE.270) GO TO 500
C**********    RETURN AND CALCULATE DATA RECORD STARTING POSITION (NRC)
      GO TO 50
C**********    NEXT 5 STATEMENTS RETRIEVE ANOTHER DATA BLOCK
C**********    AND RESETS RECORD NUMBER TO 1
 99   CALL TGET(LUN,ISTAT,DATA(1),6510)
      CALL TWAIT(LUN,ISTAT,NWORDS,IRC)
      IF(IRC.EQ.1) GO TO 900
      NRC=1
      GO TO 50
C**********    SECTION D    **************************************
C*********  DATA RETREIVAL FOR SDAS SUBTYPES 3,4,5,6  ******************
C**********    CALCULATE STARTING POSITION FOR RECORD(NRC)
 60   NSTRT=1
 61   NSTRT=NSTRT+((NRC*210)-210)
      NSTOP=NSTRT+209
C**********    INSERT ACTUAL RECORD IN OPERATING RECORD
      J=1
 63   DO 65 I=NSTRT,NSTOP
      DATAR1(J)=DATA(I)
      IF(I.LT.12811) DATAR2(J)=DATA(I+210)
 65   J=J+1
C**********    CONVERT OPERATING RECORD FROM EBCDIC CHARACTERS TO ASCII CHARACTER
S
      CALL EBCASC(DATAR1(1),18)
      IF(NSTRT.LT.12811) CALL EBCASC(DATAR2(1),18)
C**********    CONVERT EBCDIC DATA WORDS TO PDP WORDS
      CALL CFP360(DAT1(1),48,0)
      IF(NSTRT.LT.12811) CALL CFP360(DAT2(1),48,0)
      IF(NSTRT.GE.12811) GO TO 90
C**********    CHECK FOR CONTIGUOUS RECORDS
      IF(DATAR1(17).EQ.DATAR2(17)) GO TO 75
C**********    CHECK DATE ON TAPE AGAINST INPUT DATE
      IF(DATAR2(9).NE.IDATE(5).OR.DATAR2(10).NE.IDATE(6)) GO TO 500
      IF(DATAR1(17).EQ.53) GO TO 80
C**********    INSERT DATAR1 IN DATAR FOR USE WITH TIME CONVERSION(TIMCVT)
      DO 66 I=1,18
 66   DATAR(I)=DATAR1(I)
C**********    CONVERT TIME FROM ASCII CHARACTERS TO DECIMAL EQUIVALENT
      CALL TIMCVT(TIME1)
C**********    INSERT DATAR2 IN DATAR FOR USE WITH TIME CONVERSION(TIMCVT)
      DO 67 I=1,18
 67   DATAR(I)=DATAR2(I)
C**********   CONVERT TIME FROM ASCII CHARACTERS TO DECIMAL EQUIVALENT
      CALL TIMCVT(TIME)
      NK=NCH-48
C**********    CHECK FOR TIME SEQUENCE
      IF(TIME1.NE.TIME) GO TO 400
C**********    NEXT 3 STATEMENTS CHECK FOR ERRORS IN TIME ONVERSION
      IF(NORECS.EQ.0) GO TO 63
      CHK=TIME-TSAV
      IF(CHK) 800,800,63
C**********    OUTPUT CONVERTED RECORD TO SYSTEMS DISK FOR STORAGE
 68   WRITE(2'NRDAT,ERR=700) TIME,(DAT1(I),I=1,48),(DAT2(I),I=1,NK)
C**********    LINE PRINTER OUTPUT CHECK
      D=FLOAT(NSTRT)
      IF(ILP.EQ.1) WRITE(3,59) (DATAR1(I),I=7,16),NRC,D
C**********    SITE DATA TIME PROFILE OUTPUT TO SCREEN
      YS=YS-15.
      CALL TPLOT(XM,YS,0,0)
      WRITE(1,69) (DATAR1(I),I=7,16)
C**********    NEXT 5 STATEMENTS CHECK SCREEN LOCATION FOR PROPER OUTPUT
 69   FORMAT(1H+,1((2A1,'/'),2A1,2X,2(2A1,':'),2A1,2X))
      IF(YS.LT.45.) XM=XM+150.
      IF(XM.GT.900.) XM=50.
      IF(YS.LT.45.) JRK=JRK+1
      IF(YS.LT.45.) YS=YSAV
      IF(YS.LT.45..AND.JRK.GT.6) YS=730.
C**********    INCREMENT THE NUMBER OF OUTPUT RECORDS(NORECS)
      NORECS=NORECS+1
      TSAV=TIME
C**********    CHECK FOR BLOCK DATA USED UP,IF(NRC.GE.61) GET ANOTHER BLOCK
      IF(NRC.GE.61) GO TO 73
C**********    INCREMENT THE RECORD NUMBER(NRC)
      NRC=NRC+2
C**********    270 RECORDS IN 1 DAY,GREATER THAN THAT MEANS STOP
      IF(NORECS.GE.270) GO TO 500
C**********    RETURN AND CALCULATE DATA RECORD STARTING POSITION (NRC)
      GO TO 60
C**********    NEXT 5 STATEMENTS RETRIEVE ANOTHER DATA BLOCK
C**********    AND RESETS RECORD NUMBER TO 1
 73   CALL TGET(LUN,ISTAT,DATA(1),6510)
      CALL TWAIT(LUN,ISTAT,NWORDS,IRC)
      IF(IRC.EQ.1) GO TO 900
      NRC=1
      GO TO 60
C**********    CONTROL TRANSFER TO HANDLE CONTIGUOUS RECORDS
 75   N1=NSTRT+16
 76   N1=N1+210
      CALL EBCASC(DATA(N1),18)
      IF(DATA(N1).EQ.NSUB1) GO TO 76
      NS=((N1-16)-NSTRT)/210
      NRC=NRC+(NS-1)
      YS=YS-15.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,77) NS
 77   FORMAT(1H+,I4,2X,'CONTIGUOUS RECORDS WERE ENCOUNTERED')
      IF(NRC.LE.62) GO TO 60
      CALL TGET(LUN,ISTAT,DATA(1),6510)
      CALL TWAIT(LUN,ISTAT,NWORDS,IRC)
      IF(IRC.EQ.1) GO TO 900
      NRC=1
      GO TO 60
C**********    CONTROL TRANSFER TO HANDLE A SUBTYPE SHIFT IN BLOCK
 80   NSTRT=211
      GO TO 61
C**********    CONTROL TRANSFER TO HANDLE SUBTYPE 3 AND 5 AS LAST RECORD IN BLOCK

 90   CALL EBCASC(DATAR1(1),18)
      CALL CFP360(DAT1(1),48,0)
      DO 95 I=1,18
 95   DATAR(I)=DATAR1(I)
      CALL TIMCVT(TIME1)
      CALL TGET(LUN,ISTAT,DATA(1),6510)
      CALL TWAIT(LUN,ISTAT,NWORDS,IRC)
      IF(IRC.EQ.1) GO TO 900
      DO 100 I=1,210
 100  DATAR2(I)=DATA(I)
      CALL EBCASC(DATAR2(1),18)
      CALL CFP360(DAT2(1),48,0)
      DO 105 I=1,18
 105  DATAR(I)=DATAR2(I)
      CALL TIMCVT(TIME)
      IF(TIME1.NE.TIME) GO TO 400
      WRITE(2'NRDAT,ERR=700) TIME,(DAT1(I),I=1,48),(DAT2(I),I=1,NK)
      YS=YS-15.
      CALL TPLOT(XM,YS,0,0)
      WRITE(1,69) (DATAR1(I),I=7,16)
      NORECS=NORECS+1
      NRC=1
      NSTRT=211
      IF(NORECS.GE.270) GO TO 500
      GO TO 61
C**********    ERROR MESSAGES AND OUTPUT INFO BELOW***********************
 400  YS=YK-15.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,405) (DATAR1(I),I=7,16)
 405  FORMAT(1H+,'THIS DATA IS OUT OF TIME SEQUENCE AT',2X,
     *3(2A1,'/'),3X,3(2A1,':'))
      GO TO 1000
 500  YS=YK-15.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,505) NORECS
 505  FORMAT(1H+,'THE DATA HAS BEEN STORED WITH ',I4,2X,'RECORDS')
      GO TO 1000
 600  YS=YK-15.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,605)
 605  FORMAT(1H+,'THERE IS AN ERROR IN FILE MANIPULATION')
      GO TO 990
 700  YS=YK-15.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,705)
 705  FORMAT(1H+,'THERE IS AN ERROR IN WRITING THE FILE')
      GO TO 1000
 800  YS=YK-15.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,805)
 805  FORMAT(1H+,'THERE IS AN ERROR IN THE TIME CONVERSION')
      GO TO 1000
 900  YS=YK-15.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,905)
      MFILE=MFILE+1
 905  FORMAT(1H+,'AN END OF FILE WAS ENCOUNTERED DURING EXECUTION')
1000  YS=YS-15.
      CALL TPLOT(XS,YS,0,0)
      CALL TPAUSE
      CALL ERASE
      WRITE(1,2000)
      READ(1,2050) IREP
      IF(ILP.EQ.1) CLOSE(UNIT=3,DISPOSE='PRINT',ERR=9999)
      CLOSE (UNIT=2,ERR=9999)
      IF(IREP.EQ.1) GO TO 2
 990  CALL RWND(ISTAT(1))
      WRITE(1,1001)
1001  FORMAT(//1X,'THE TAPE IS BEING REWOUND')
 2000 FORMAT(//1X,'DO YOU WISH TO RETRIEVE MORE DATA ?  (1=YES,2=NO)')
 2050 FORMAT(I1)
9999  STOP
PIP>  END
```

```
      SUBROUTINE TAPFWD(NFILE,MFILE,ISTAT,IREP)
      DIMENSION ISTAT(1)
      LUN=7
      IF(IREP.EQ.2.AND.NFILE.EQ.1) RETURN
      IF(NFILE.EQ.1)  IDIR=3
      IF(NFILE.EQ.1) GO TO 60
      IF(NFILE-MFILE) 20,20,50
   20 NF=MFILE-NFILE+1
      IDIR=13
      GO TO 60
   50 NF=NFILE-MFILE
      IDIR=10
   60 CALL SFUN(LUN,ISTAT,NF,IRC,IDIR)
      CALL TWAIT(LUN,ISTAT,NWORDS,IRC)
      RETURN
      END
      SUBROUTINE TIMCUT(TIME)
      INTEGER*2 ITIME(3),ITIME1(6)
      LOGICAL*1 DATAR(210)
      COMMON/TIM/DATAR
      K=11
 5    M=K-10
      DO 10 J=48,57
      IF(DATAR(K).EQ.J)ICHK=J
 10   CONTINUE
      ITIME1(M)=ICHK-48
      IF(M.GT.6) GO TO 15
      K=K+1
      GO TO 5
 15   J=1
      DO 20 I=1,5,2
      ITIME1(I)=ITIME1(I)*10
      ITIME(J)=ITIME1(I)+ITIME1(I+1)
      J=J+1
 20   CONTINUE
      T1=(FLOAT(ITIME(3))/60.)
      T2=(FLOAT(ITIME(2))+T1)/60.
      TIME=(FLOAT(ITIME(1))+T2)
      RETURN
PIP>  END
```

TAPFIL VARIABLES

CHK- used for the difference between two consecutive time points

D- real value of starting position in data block for conversion and storage

IDIR- tape function to be performed (3= rewind, 10= forward space files, 13= backspace files)

ILP- integer variable for line printer output decision (1= yes, 2= no)

IRC- denotes error status (0= correct read, 1= end of file, 2= tape error)

IREP- determines whether want more data (1=yes, 2= no)

IRK- denotes the screen time output column number (1-6)

LUN- logical unit number of tape drive

MFILE- current file number

N1- subscripted variable designating subtype value; used to check for contiguous records

NCH- the number of channels at a site

NF- number of files to space on the IBM 360 tape

NFILE- denotes the tape file number in which the desired date is located

NK- the number of channels of data contained in subtype records 4 and 6

NORECS- the number of output records on the system disk

NRC- the operating record number in the data block from the tape

NRDAT- associate variable for opening direct access file (IFILE) and for outputting to the system disk

NS- the number of contiguous records

NSTOP- stopping position in data block, used with NSTRT (NSTOP=NSTRT + 209)

NSTRT- starting position in data block for conversion and storage; calculated from NRC

NSUB- denotes the number of the subtype

NSUB1- the ASCII code equivalent of the subtype

NT- the number of channels + 1 for time

NWORDS- the number of words not read from the block on the tape

T1- converts excess seconds to minutes in TIMCVT

T2- converts excess minutes to hours in TIMCVT

TIME- decimal equivalent time value converted from ASCII by TIMCVT

TIME1- decimal equivalent time value converted from ASCII characters stored in subtypes 4 and 6

TSAV- used to save previous time point for use with CHK to check time conversions

XM- T4014 scope column abscissa value for time output (50-800)

XS- T4014 scope abscissa location used for output

YK- T4014 scope ordinate value used at end of execution for output (YK=40)

YS- T4014 scope ordinate location used for output

YSAV- T4014 scope ordinate value used to save last value of YS

TAPFIL ARRAYS

DAT1(48)- real storage of subtype 1,3, and 5 records equivalenced to DATAR1(19-210) (channels 1-48)

DAT2(48)- real storage of subtype 4 and 6 records equivalenced to DATAR2(19-210) (channels 49-96)

DATA(13020)- LOGICAL *1 array to store IBM 360 words from tape

DATAR(210)- LOGICAL *1 array for time conversion in subroutine TIMCVT

DATAR1(210)- LOGICAL *1 array to store individual subtype 1, 3, and 5 records (channels 1-48)

DATAR2(210)- LOGICAL *1 array to store individual subtype 4 and 6 records (channels 49-96)

ICHK(2)- LOGICAL *1 array used to locate user date on tape

IDATE(6)- LOGICAL *1 array of desired date to be retrieved (input from scope)

IFILE(30)- LOGICAL *1 array used to store data file name for storage

ISTAT(4)- status code for tape handling

ITIME(3)- stores decimal equivalent of hours, minutes, and seconds

ITIME1(6)- stores decimal equivalent of each character of desired time

APPENDIX D

CHPLOT SOURCE LISTING

```
      SUBROUTINE AUTOSC(XMAX,XMIN,STX,RLX,TICX)
      INTEGER*4 IA,ISTX,IB,IEX
C    XMAX.....(INPUT ARGUMENT) MAX. VALUE
C    XMIN.....(INPUT ARGUMENT) MIN. VALUE
C    STX......(RETURN ARGUMENT) STARTING X VALUE ON AXIS
C    RLX......(RETURN ARGUMENT) LENGTH OF AXIS
C    TICX.....(RETURN ARGUMENT) LENGTH BETWEEN TIC MARKS
      RLX=XMAX-XMIN
      Q=RLX
      IF(RLX.EQ.0.) GO TO 4
      IA=ALOG10(RLX)
      IF(IA)1,1,3
 3    CONTINUE
      ISTX=XMIN/(10.**IA)
      IB=ISTX
      IF(XMIN.LT.0.)IB=ISTX-1
      ISTX=IB*(10**IA)
      IEX=XMAX/(10.**IA)
      IB=IEX+1
      IF(XMAX.LT.0.)IB=IEX
      IEX=IB*(10**IA)
      RLX=IEX-ISTX
      STX=ISTX
      TICX=(IEX-ISTX)/10
      IF(Q.LE.10.)TICX=1.
      GO TO 4
 1    CONTINUE
      IF(RLX.GE.1.)GO TO 3
      IA=IA-1
      ISTX=XMIN/(10.**IA)
      IB=ISTX
      IF(XMIN.LT.0.)IB=ISTX-1
      STX=IB*(10.**IA)
      IEX=XMAX/(10.**IA)
      IB=IEX+1
      IF(XMAX.LT.0.)IB=IEX
      EX=IB*(10.**IA)
```

```
        RLX=EX-STX
        TICX=(10.**IA)
 8      IF(TICX.GE.(RLX/10.))GO TO7
        TICX=TICX+(10.**IA)
        GO TO 8
 7      CONTINUE
        IF((RLX/TICX).GT.4.)GO TO 4
        TICX=TICX/2.
        GO TO 7
 4      RETURN
        END
PIP>
```

CHPLOT:

```
      DIMENSION SX1(6),SX2(6),SX3(6),SX4(6),SX5(6)
      DIMENSION SAV1(300),SAV2(300),SAV3(300),SAV4(300),SAV5(300)
      DIMENSION PID(4),XTITL(4),TIME(300)
      COMMON/SX/SX1,SX2,SX3,SX4,SX5
      COMMON/PID1/PID
      COMMON/STOPT/ISTOPT,ISTOPF
      COMMON/POINTS/NPT
      COMMON/HOWM/NPLT,IFILE,NSCH
      COMMON/LUN/IS,IP,IC,ISD
      COMMON/UTL1/IFLAGX,XMAX,XMIN
      COMMON/UTL2/IFLAGY,YMAX,YMIN
      COMMON/NUMBR/SAV1,SAV2,SAV3,SAV4,SAV5,TIME
      DATA XTITL/'CLOC','K TI','ME  ','    '/
      CALL BEGIN(9600)
      IS=1
      ISTOPT=0
      ISTOPF=0
  1   CALL DISPLY
      IFLAGX=0
      IFLAGY=0
      CALL RETREV
  5   CALL PLOT5(PID(1),XTITL(1),SAV1(1),SAV2(1),SAV3(1),SAV4(1),SAV5(1)
     *,TIME(1),NPT,NPT,NPT,NPT,NPT,-1)
      CALL TPAUSE
      CALL WATNXT(IR)
      ISTOPT=1
      ISTOPF=1
      IF(IR.EQ.1) GO TO 1
      ISTOPT=0
      ISTOPF=0
      IF(IR.EQ.3) GO TO 1
      ISTOPF=1
      ISTOPT=0
      IF(IR.EQ.4) GO TO 1
      IF(IR.EQ.5) GO TO 10
      CALL PSCALE
      GO TO 5
 10   STOP
      END
      SUBROUTINE CHSIZE(N)
C***   THIS ROUTINE CHANGES CHARACTER SIZE ON A TEKTRONIX 4014-1 DISPLAY
C***   SCOPE ACCORDING TO N AS FOLLOUS:  N=1 - LARGEST
C***                                     N=2 - NEXT SMALLER
C***                                     N=3 - NEXT SMALLER
C***                                     N=4 - SMALLEST
      NP=N+55
      CALL CHOUT(27)
      CALL CHOUT(NP)
      RETURN
      END
PIP>
```

```
      SUBROUTINE DATIN(XS,YS,DATW)
C***   THIS ROUTINE ALLOWS THE USER TO RETRIVE FLOATING POINT DATA FROM
C***   THE SCOPE IN G13.6 FORMAT, ONE WORD PER CALL.  XS IS THE ABSOLUTE
C***   SCREEN COORDINATE IN TEKPOINTS, LIKEWISE YS.  DATA FROM THE SCOPE
C***   IS RETURNED IN DATW.
      COMMON/LUN/IS,IP,IC,ISD
      CALL SCALE(1.0,1.0,0.0,0.0)
      CALL ALPHA
      CALL TPLOT(XS,YS,0,0)
      READ(IS,100) DATW
100   FORMAT(G13.6)
      RETURN
      END
PIP>
```

```
      SUBROUTINE DISPLY
      DIMENSION SX1(6),SX2(6),SX3(6),SX4(6),SX5(6)
      DIMENSION PID(4)
      LOGICAL*1 IFILE(30),IFILET(50)
      COMMON/CHNO/ NCH1,NCH2,NCH3,NCH4,NCH5
      COMMON/PID1/PID
      COMMON/STOPT/ISTOPT,ISTOPF
      COMMON/SX/SX1,SX2,SX3,SX4,SX5
      COMMON/HOWM/NPLT,IFILE,NSCH
      COMMON/CHECK/ICHECK,ANPLT
      COMMON/FILEB/IFILET
      DATA IFILE(30)/0/
      DATA IFILET(50)/0/
      CALL ERASE
      CALL SCALE(1.0,1.0,0.0,0.0)
      CALL ALPHA
      CALL CHSIZE(2)
      CALL TPLOT(25.,650.,0,0)
      WRITE(1,5)
    5 FORMAT(1H+,'EL 55 CHANNEL PLOTTING PROGRAM')
      CALL CHSIZE(3)
      XS=20.
      YS=650.
      IF(ISTOPF.EQ.1) GO TO 16
      YS=YS-50.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,10)
   10 FORMAT(1H+,'ENTER PLOT FILE NAME')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      READ(1,15) (IFILE(I),I=1,29)
   15 FORMAT(29A1)
   16 YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,20)
   20 FORMAT(1H+,'ENTER NO. OF PLOTS ON THIS GRAPH (MAX=5)')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      READ(1,25) NPLT
 25   FORMAT(I2)
      ANPLT=FLOAT(NPLT)
      IF(ISTOPT.EQ.1) GO TO 29
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,26)
26    FORMAT(1H+,'ENTER PLOT CAPTION (49 CHARACTERS)')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      READ(1,27) (IFILET(I),I=1,49)
27    FORMAT(49A1)
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,43)
43    FORMAT(1H+,'ENTER CAPTION FOR LEFT SIDE OF PLOT')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      READ(1,44) (PID(I),I=1,4)
44    FORMAT(4A4)
29    YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,30)
 30   FORMAT(1H+,'ENTER CHANNEL NO. AND UNITS')
      K=1
      RS=XS+40.
 35   YS=YS-25.
      GO TO (40,45,50,55,60) K
 40   CALL TPLOT(XS,YS,0,0)
      READ(1,41) NCH1
 41   FORMAT(I2)
      CALL TPLOT(RS,YS,0,0)
 42   FORMAT(6A4)
      READ(1,42) (SX1(I),I=1,6)
      IF(K.EQ.NPLT) GO TO 70
      K=K+1
      GO TO 35
```

```
45  CALL TPLOT(XS,YS,0,0)
    READ(1,41) NCH2
    CALL TPLOT(RS,YS,0,0)
    READ(1,42) (SX2(I),I=1,6)
    IF(K.EQ.NPLT)GO TO 70
    K=K+1
    GO TO 35
50  CALL TPLOT(XS,YS,0,0)
    READ(1,41) NCH3
    CALL TPLOT(RS,YS,0,0)
    READ(1,42) (SX3(I),I=1,6)
    IF(K.EQ.NPLT) GO TO 70
    K=K+1
    GO TO 35
55  CALL TPLOT(XS,YS,0,0)
    READ(1,41) NCH4
    CALL TPLOT(RS,YS,0,0)
    READ(1,42) (SX4(I),I=1,6)
    IF(K.EQ.NPLT) GO TO 70
    K=K+1
    GO TO 35
60  CALL TPLOT(XS,YS,0,0)
    READ(1,41) NCH5
    CALL TPLOT(RS,YS,0,0)
    READ(1,42) (SX5(I),I=1,6)
70  YS=YS-25.
    CALL TPLOT(XS,YS,0,0)
    IF(ISTOPF.EQ.1) GO TO 75
    WRITE(1,71)
71  FORMAT(1H+,'ENTER NO. OF CHANNELS AT SITE')
    YS=YS-25.
    CALL TPLOT(XS,YS,0,0)
    READ(1,41) NSCH
    YS=YS-30.
    CALL TPLOT(XS,YS,0,0)
75  WRITE(1,72)
72  FORMAT(1H+,'DATA IS BEING RETRIEVED')
    RETURN
     END
```

```
      SUBROUTINE PLOT5(PID,UNITS,Y1,Y2,Y3,Y4,Y5,X,NPT1,NPT2,NPT3,NPT4,
     *NPT5,IG)
C***   PLOT6 BUILDS 5 PLOTS OF DATA ON A STANDARA GRAPH FOR USE IN THE
C***   QUICK-LOOK GRAPHICS PROGRAM. PID IDENTIFIES THE NAME OF THE DATA BASE.
C***   UNITS IS THE ARRAY CONTAINING THE UNITS OF MEASUREMENT OF THE
C***   DEPENDENT PARAMETER. XTITL IS THE ARRAY CONTAINING THE INDEPEDENT
C***   PARAMETER UNITS. X AND Y ARE THE ARRAY ELEMENTS FROM THE FIRST ARRAY
C***   OF DATA READ IN,CORRESPONDING WITH SYMBOL XKENO. X2 AND Y2 ARE THE ARRAY
C***   ELEMENTS FROM THE SECOND ARRAY OF DATA,CORRESPONDING WITH SYMBOL XKYNO.
C***   NPID IS THE NUMBER OF CHARACTERS TO BE PRINTED FOR PID. NUN IS THE NUMBER
C***   OF CHARACTERS TO BE PRINTED FOR UNITS. NXT IS THE NUMBER OF CHARACTERS
C***   TO BE PRINTED FOR XTITL. NPXY IS THE NUMBER OF POINTS IN THE FIRST
C***   DATA ARRAY, NPXY2 IS THE NUMBER OF POINTS IN THE SECOND DATA ARRAY.
C***   IG IS THE GRID LINE VARIABLE,(IG.EQ.-1)FOR GRID LINES. (IG.EQ.1)
C***   FOR NO GRID LINES.
      DIMENSION PID(4),UNITS(4),Y1(1),X(1),Y2(1),Y3(1)
      DIMENSION SX1(6),SX2(6),SX3(6),SX4(6),SX5(6)
      DIMENSION FILET(12)
      DIMENSION Y4(1),Y5(1)
      COMMON/FILEB/IFILET
      COMMON/SX/SX1,SX2,SX3,SX4,SX5
      COMMON/CHECK/ICHECK,ANPLT
      COMMON/LUN/IS,IP,IC,ISD
      LOGICAL*1 IFILET(50)
      COMMON/VECT/IVECT
      EQUIVALENCE(FILET(1),IFILET(1))
      DATA IFILET(50)/0/
 10   IVECT=1
      CALL SSPLOT(Y1(1),Y2(1),Y3(1),Y4(1),Y5(1),X(1),850.,
     *623.,140.,118.,NPT1,NPT2,NPT3,NPT4,NPT5,IG,IG)
      CALL CHSIZE(3)
      CALL YLABEL(10.,700.,PID(1),16)
      CALL XLABEL(420.,58.,UNITS(1),16)
      CALL SCALE(1.0,1.0,0.0,0.0)
      CALL TPLOT(10.,10.,0,0)
      CALL ALPHA
      WRITE(1,59) (IFILET(I),I=1,49)
 59   FORMAT(1H+,49A1)
      NPLT=INT(ANPLT)
      DO 100 J=1,NPLT
      GO TO (101,102,103,104,105) J
101   CALL TPLOT(634.,70.,0,0)
      CALL VECTOR
      CALL TPLOT(710.,70.,1,0)
      CALL POINT
      CALL ALPHA
      CALL XLABEL(753.,70.,SX1(1),24)
      GO TO 100
 102  CALL TPLOT(634.,50.,0,0)
      CALL VECTOR
      CALL CHOUT(27)
      CALL CHOUT(97)
      CALL TPLOT(710.,50.,1,0)
      CALL POINT
      CALL ALPHA
      CALL XLABEL(753.,50.,SX2(1),24)
      GO TO 100
103   CALL TPLOT(634.,30.,0,0)
      CALL VECTOR
      CALL CHOUT(27)
      CALL CHOUT(99)
      CALL TPLOT(710.,30.,1,0)
      CALL POINT
      CALL ALPHA
      CALL XLABEL(753.,30.,SX3(1),24)
      GO TO 100
 104  CALL TPLOT(118.,70.,0,0)
      CALL VECTOR
      CALL CHOUT(27)
      CALL CHOUT(100)
      CALL TPLOT(194.,70.,1,0)
      CALL POINT
      CALL ALPHA
      CALL XLABEL(237.,70.,SX4(1),24)
      GO TO 100
 105  CALL TPLOT(118.,50.,0,0)
      CALL VECTOR
      CALL CHOUT(27)
      CALL CHOUT(98)
      CALL TPLOT(194.,50.,1,0)
      CALL TPLOT(194.,50.,0,0)
      CALL POINT
      CALL ALPHA
      CALL XLABEL(237.,50.,SX5(1),24)
 100  CONTINUE
      CALL TPLOT(237.,50.,0,0)
      CALL ALPHA
      CALL HOME
      RETURN
PIP>  END
```

```
      SUBROUTINE PSCALE
      COMMON/UTL1/IFLAGX,XMAX,XMIN
      COMMON/UTL2/IFLAGY,YMAX,YMIN
      CALL ERASE
      IFLAGX=1
      IFLAGY=1
      XS=25.
      YS=600.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,20)
20    FORMAT(1H+,'ENTER MAXIMUM AND MINIMUM VALUES FOR X AND Y')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,25)
25    FORMAT(1H+,'XMAX=')
      CALL DATIN(105.,YS,XMAX)
      CALL ALPHA
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,26)
26    FORMAT(1H+,'XMIN=')
      CALL DATIN(105.,YS,XMIN)
      CALL ALPHA
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,27)
27    FORMAT(1H+,'YMAX=')
      CALL DATIN(105.,YS,YMAX)
      CALL ALPHA
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,28)
28    FORMAT(1H+,'YMIN=')
      CALL DATIN(105.,YS,YMIN)
      CALL ALPHA
      RETURN
      END
      SUBROUTINE RETREV
      DIMENSION SAV1(300),SAV2(300),SAV3(300),SAV4(300),SAV5(300)
      DIMENSION DATA(100),TIME(300)
      LOGICAL*1 IFILE(30)
      COMMON/NUMBR/SAV1,SAV2,SAV3,SAV4,SAV5,TIME
      COMMON/HOWM/NPLT,IFILE,NSCH
      COMMON/POINTS/NPT
      COMMON/CHNO/ NCH1,NCH2,NCH3,NCH4,NCH5
      DATA IFILE(30)/0/
      NT=NSCH+1
      OPEN(UNIT=2,NAME=IFILE,TYPE='OLD',ACCESS='DIRECT',
     *RECORDSIZE=NT,ASSOCIATEVARIABLE=NRDAT,ERR=25)
      NRDAT=1
    5 K=1
   10 READ(2'NRDAT,ERR=21) (DATA(I),I=1,NT)
      TIME(K)=DATA(1)
      SAV1(K)=DATA(NCH1+1)
      IF(NPLT.LT.2) GO TO 20
      SAV2(K)=DATA(NCH2+1)
      IF(NPLT.LT.3) GO TO 20
      SAV3(K)=DATA(NCH3+1)
      IF(NPLT.LT.4) GO TO 20
      SAV4(K)=DATA(NCH4+1)
      IF(NPLT.LT.5) GO TO 20
      SAV5(K)=DATA(NCH5+1)
   20 IF(K.LT.270) K=K+1
      IF(K.LT.270) GO TO 10
   21 NPT=K-1
      CLOSE(UNIT=2,ERR=25)
   25 RETURN
      END
PIP>
```

```
      SUBROUTINE SSPLOT(Y1,Y2,Y3,Y4,Y5,X,TPX,TPY,XO,YO,NPT1,NPT2
     *,NPT3,NPT4,NPT5,IGX,IGY)
C***   SSPLOT AUTOMATICALLY SCALES THE X AND Y DATA STORED IN ARRAYX AND
C***   ARRAYY TO FIT INTO THE TEKPOINTS AVAILABLE AS SHOWN IN TPX AND TPY.
C***   SSPLOT THEN DRAWS THE AXIS AND GRID, PLOTS THE DATA IN ARRAYX
C***   AND ARRAYY AND LABLES THE NUMERICAL VALUES OF THE TIC MARKS ON THE AXES.
C***   XO AND YO ARE ORIGIN LOCATIONS FOR THE GRID IN ABSOLUTE SCREEN
C***   COORDINATES REFERENCED FROM SCREEN (0,0). IGX AND IGY ARE
C***   INDICATORS FOR GRID LINES ON THE PLOT (.LT.0) OR TIC MARKS
C***   (.GT.0) ON THE OUTPUT PLOT.
      DIMENSION Y1(300),Y2(300),Y3(300),Y4(300),Y5(300)
      DIMENSION X(300)
      COMMON/LUN/IS,IP,IC,ISD
      COMMON/CHECK/ICHECK,ANPLT
      COMMON/VECT/ IVECT
      COMMON/UTL1/ IFLAGX,XMAX,XMIN
      COMMON/UTL2/ IFLAGY,YMAX,YMIN
      INTEGER*4 N
      CALL ERASE
      L=1
      IF(IFLAGX.GT.0) GO TO 11
      XMAX=X(1)
      XMIN=X(1)
      DO 10 I=1,NPT1
      IF(X(I).GT.XMAX) XMAX=X(I)
      IF(X(I).LT.XMIN) XMIN=X(I)
10    CONTINUE
11    CALL AUTOSC(XMAX,XMIN,STX,RLX,TICX)
C***   CHECK PLOTTING EFFICIENCY
      N=RLX/TICX
      DO 12 I=1,N
      IF((STX+TICX).GE.XMIN) GO TO 13
      STX=STX+TICX
      RLX=RLX-TICX
12    CONTINUE
13    CONTINUE
      DO 14 I=1,N
      IF((STX+RLX-TICX).LE.XMAX) GO TO 16
      RLX=RLX-TICX
14    CONTINUE
16    CONTINUE
      INRLX=RLX/TICX
      CRLX=INRLX*TICX
      IF(CRLX.LT.RLX) INRLX=INRLX+1
      RLX=INRLX*TICX+0.1
      IF(IFLAGY.GT.0) GO TO 21
C*****   FIND LIMITING Y VALUES FOR LEFT SIDE *******
      YMAX=Y1(1)
      YMIN=Y1(1)
      NPLT=INT(ANPLT)
      DO 17 J=1,NPLT
      GO TO (99,101,102,103,104) J
  99  DO 18 I=1,NPT1
      IF(Y1(I).GT.YMAX) YMAX=Y1(I)
      IF(Y1(I).LT.YMIN) YMIN=Y1(I)
 18   CONTINUE
      GO TO 17
 101  DO 19 I=1,NPT2
      IF(Y2(I).GT.YMAX) YMAX=Y2(I)
      IF(Y2(I).LT.YMIN) YMIN=Y2(I)
 19   CONTINUE
      GO TO 17
 102  DO 20 I=1,NPT3
      IF(Y3(I).GT.YMAX) YMAX=Y3(I)
      IF(Y3(I).LT.YMIN) YMIN=Y3(I)
 20   CONTINUE
      GO TO 17
 103  DO 200 I=1,NPT4
      IF(Y4(I).GT.YMAX) YMAX=Y4(I)
      IF(Y4(I).LT.YMIN) YMIN=Y4(I)
 200  CONTINUE
      GO TO 17
 104  DO 201 I=1,NPT5
      IF(Y5(I).GT.YMAX) YMAX=Y5(I)
      IF(Y5(I).LT.YMIN) YMIN=Y5(I)
 201  CONTINUE
17    CONTINUE
21    CALL AUTOSC(YMAX,YMIN,STY,RLY,TICY)
C***   CHECK PLOTTING EFFICIENCY
      N=RLY/TICY
      DO 22 I=1,N
      IF((STY+TICY).GE.YMIN) GO TO 23
      STY=STY+TICY
      RLY=RLY-TICY
22    CONTINUE
23    CONTINUE
      DO 25 I=1,N
      IF((STY+RLY-TICY).LE.YMAX) GO TO 26
      RLY=RLY-TICY
25    CONTINUE
26    CONTINUE
      INRLY=RLY/TICY
      CRLY=INRLY*TICY
      IF(CRLY.LT.RLX) INRLY=INRLY+1
      RLY=INRLY*TICY+0.1
C*****  CHECK PLOTTING EFFICIENCY *********
      IF(CRLYR.LT.RLYR) INRLYR=INRLYR+1
      RLYR=INRLYR*TICYR+.1
      XFACT=TPX/RLX
      YFACT=TPY/RLY
      YOS=YO
      YOM=YO
      XOS=XO
      XO=XO-STX*XFACT
      YO=YO-STY*YFACT
      YOR=YOM-STYR*YFACTR
      IF(YO.GT.(YOS+TPY)) YO=YOS+TPY
      IF(YOR.GT.(YOS+TPY)) YOR=YOS+TPY
30    CALL SCALE(XFACT,YFACT,XOS,YOS)
      CALL AXIS(0.0,0.0,RLX,RLY,TICX,TICY,-1,-1)
      CALL SCALE(XFACT,YFACT,XO,YO)
      CALL POINT
      IF(IVECT.GT.0) CALL VECTOR
      I=1
      NPLT=INT(ANPLT)
      DO 800 J=1,NPLT
      INET=0
      GO TO (300,301,302,303,304) J
300   IF(IFLAGX.LT.1) GO TO 299
      INET=1
      DO 38 I=1,NPT1
      IF(X(I).GT.XMIN.AND.Y1(I).GT.YMIN) CALL TPLOT(X(I),Y1(I),0,0)
      IF(X(I).GT.XMIN.AND.Y1(I).GT.YMIN) L=I
      IF(X(I).GT.XMIN.AND.Y1(I).GT.YMIN) GO TO 299
38    CONTINUE
299   IF(INET.LT.1) CALL TPLOT(X(1),Y1(1),0,0)
      DO 39 I=L,NPT1
      IF(X(I) .GT.XMAX.OR.Y1(I).GT.YMAX) GO TO 39
      IF(X(I).LT.XMIN.OR.Y1(I).LT.YMIN)GO TO 39
      CALL TPLOT(X(I),Y1(I),I,0)
      IF(I.EQ.NPT1)CALL TPLOT(X(NPT1),Y1(NPT1),0,0)
39    CONTINUE
      CALL VECTOR
      CALL POINT
      CALL TPLOT(0.0,0.0,0,0)
      GO TO 800
301   IF(IFLAGX.LT.1) GO TO 400
      INET=1
      DO 402 I=1,NPT2
      IF(X(I).GT.XMIN.AND.Y2(I).GT.YMIN) CALL TPLOT(X(I),Y2(I),0,0)
      IF(X(I).GT.XMIN.AND.Y2(I).GT.YMIN) L=I
      IF(X(I).GT.XMIN.AND.Y2(I).GT.YMIN) GO TO 400
402   CONTINUE
400   IF(INET.LT.1) CALL TPLOT(X(1),Y2(1),0,0)
      DO 40 I=L,NPT2
      CALL VECTOR
      CALL CHOUT(27)
      CALL CHOUT(97)
      IF(X(I).GT.XMAX.OR.Y2(I).GT.YMAX) GO TO 40
      IF(X(I).LT.XMIN.OR.Y2(I).LT.YMIN) GO TO 40
      CALL TPLOT(X(I),Y2(I),I,0)
      IF(I.EQ.NPT2) CALL TPLOT(X(NPT2),Y2(NPT2),0,0)
40    CONTINUE
      CALL POINT
      CALL TPLOT(0.0,0.0,0,0)
      GO TO 800
302   IF(IFLAGX.LT.1) GO TO 404
      INET=1
      DO 403 I=1,NPT3
      IF(X(I).GT.XMIN.AND.Y3(I).GT.YMIN) CALL TPLOT(X(I),Y3(I),0,0)
      IF(X(I).GT.XMIN.AND.Y3(I).GT.YMIN) L=I
      IF(X(I).GT.XMIN.AND.Y3(I).GT.YMIN) GO TO 404
403   CONTINUE
404   IF(INET.LT.1) CALL TPLOT(X(1),Y3(1),0,0)
      DO 41 I=L,NPT3
      CALL VECTOR
      CALL CHOUT(27)
      CALL CHOUT(99)
      IF(X(I).GT.XMAX.OR.Y3(I).GT.YMAX) GO TO 41
      IF(X(I).LT.XMIN.OR.Y3(I).LT.YMIN) GO TO 41
      CALL TPLOT(X(I),Y3(I),I,0)
      IF(I.EQ.NPT3) CALL TPLOT(X(NPT3),Y3(NPT3),0,0)
41    CONTINUE
      CALL POINT
      CALL TPLOT(0.0,0.0,0,0)
      GO TO 800
303   IF(IFLAGX.LT.1) GO TO 406
      INET=1
      DO 405 I=1,NPT4
      IF(X(I).GT.XMIN.AND.Y4(I).GT.YMIN) CALL TPLOT(X(I),Y4(I),0,0)
      IF(X(I).GT.XMIN.AND.Y4(I).GT.YMIN) L=I
      IF(X(I).GT.XMIN.AND.Y4(I).GT.YMIN) GO TO 406
405   CONTINUE
406   IF(INET.LT.1) CALL TPLOT(X(1),Y4(1),0,0)
      DO 42 I=L,NPT4
      CALL VECTOR
      CALL CHOUT(27)
      CALL CHOUT(100)
      IF(X(I).GT.XMAX.OR.Y4(I).GT.YMAX) GO TO 42
      IF(X(I).LT.XMIN.OR.Y4(I).LT.YMIN) GO TO 42
      CALL TPLOT(X(I),Y4(I),I,0)
      IF(I.EQ.NPT4) CALL TPLOT(X(NPT4),Y4(NPT4),0,0)
 42   CONTINUE
      CALL POINT
      CALL TPLOT(0.0,0.0,0,0)
      GO TO 800
 304  IF(IFLAGX.LT.1) GO TO 408
      INET=1
      DO 407 I=1,NPT5
      IF(X(I).GT.XMIN.AND.Y5(I).GT.YMIN) CALL TPLOT(X(I),Y5(I),0,0)
      IF(X(I).GT.XMIN.AND.Y5(I).GT.YMIN) L=I
      IF(X(I).GT.XMIN.AND.Y5(I).GT.YMIN) GO TO 408
 407  CONTINUE
 408  IF(INET.LT.1) CALL TPLOT(X(1),Y5(1),0,0)
      DO 43 I=L,NPT5
      CALL VECTOR
      CALL CHOUT(27)
      CALL CHOUT(98)
      IF(X(I).GT.XMAX.OR.Y5(I).GT.YMAX) GO TO 43
      IF(X(I).LT.XMIN.OR.Y5(I).LT.YMIN) GO TO 43
      CALL TPLOT(X(I),Y5(I),I,0)
      IF(I.EQ.NPT5) CALL TPLOT(X(NPT5),Y5(NPT5),0,0)
 43   CONTINUE
      CALL POINT
      CALL TPLOT(0.0,0.0,0,0)
 800  L=1
C***   DATA PLOTTING IS COMPLETE, LABLE SCALES ON AXES
      CALL ALPHA
      CALL SCALE(1.0,1.0,0.0,0.0)
      CALL CHSIZE(3)
      XO=XOS
      YO=YOS
      TY=STY+RLY
      YL=YO
      XL=XO-121.0
50    CALL TPLOT(XL,YL,0,0)
      WRITE(IS,100) STY
```

```
100   FORMAT(1H+,G13.6)
      YL=YL+TICY*YFACT
      STY=STY+TICY
      IF(STY.LE.TY) GO TO 50
      XT=STX+RLX
      YL=YOS-20.0
      XL=XOS-60.0
      XLZ=XL-120.
60    CALL TPLOT(XL,YL,0,0)
      IF((XL-XLZ).GT.115.0) WRITE(IS,901) STX
 901  FORMAT(1H+,G13.2)
      XL=XL+TICX*XFACT
      STX=STX+TICX
      IF(STX.LE.XT) GO TO 60
      CALL VECTOR
      RETURN
      END
PIP>
```

```
      SUBROUTINE WATNXT(IR)
      CALL ERASE
      XS=20.
      YS=500.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,20)
 20   FORMAT(1H+,'DO YOU WISH TO:')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,23)
 23   FORMAT(1H+,'1    MAKE OTHER PLOTS FROM SAME
     * FILE WITH SAME TITLES')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,24)
 24   FORMAT(1H+,'2    CALL SCALING OPTIONS')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,25)
 25   FORMAT(1H+,'3    RETRIEVE ANOTHER DATA FILE')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,30)
 30   FORMAT(1H+,'4    MAKE OTHER PLOTS FROM SAME
     * FILE WITH NEW TITLES')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      WRITE(1,35)
 35   FORMAT(1H+,'5    STOP')
      YS=YS-25.
      CALL TPLOT(XS,YS,0,0)
      READ(1,22) IR
  22  FORMAT(I2)
      RETURN
      END
PIP>
```

```
      SUBROUTINE XLABEL(XS,YS,ARRAY,NPA)
C***   THIS ROUTINE OUTPUTS TEXT IN THE +X DIRECTION ON THE 4014 SCOPE.
C***   XS&YS ARE ABSOLUTE SCREEN COORDINATE LOCATIONS IN TEKPOINTS,
C***   ARRAY IS THE FIRST LOCATION IN WHICH THE TEXT ARRAY IS STORED
C***   (4 CHARACTERS/LOCATION), NPA IS THE NUMBER OF CHARACTERS TO BE
C***   OUTPUT TO SCOPE LUN=IS.
      COMMON/LUN/IS,IP,IC,ISD
      DIMENSION ARRAY(1)
      CALL SCALE(1.0,1.0,0.0,0.0)
      CALL ALPHA
      CALL TPLOT(XS,YS,0,0)
C***   NOTE:  PDP ISSUES LINE FEED COMMAND BEFORE WRITING, USER MUST
C***   ADJUST YS ACCORDINGLY, THUS FORMAT CONTAINS 1H+ AND YS IS START
      NW=NPA/4
      IF((NPA-NW*4).NE.0) NW=NW+1
      WRITE(IS,100) (ARRAY(I),I=1,NW)
100   FORMAT(1H+,30A4)
      RETURN
      END
PIP>
```

```
      SUBROUTINE YLABEL(XS,YS,ARRAY,NPA)
C***   THIS ROUTINE WILL VERTICALLY PRINT A CHARACTER STRING ON THE
C***   T4014 STARTING AT ABSOLUTE LOCATION (XS,YS) IN TEKPOINTS.  DATA
C***   TO BE PRINTED ARE STORED IN ARRAY WITH 4 CHARACTERS TO THE WORD.
C***   NPA IS THE NUMBER OF CHARACTERS TO BE PRINTED.
      COMMON/LUN/IS,IP,IC,ISD
      DIMENSION ARRAY(1),X(1)
      LOGICAL*1 IX(4),IBLK
      EQUIVALENCE (X(1),IX(1))
      DATA IBLK/1H /
      NW=NPA/4
      IF((NW*4-NPA).EQ.0)GO TO 20
      X(1)=ARRAY(NW+1)
      DO 10 I=1,4
      IF((NW*4+I).LE.NPA) GO TO 10
      IX(I)=IBLK
10    CONTINUE
      ARRAY(NW+1)=X(1)
      NW=NW+1
20    CALL SCALE(1.0,1.0,0.0,0.0)
      CALL ALPHA
      CALL TPLOT(XS,YS,0,0)
      IC=1
30    X(1)=ARRAY(IC)
      DO 40 I=1,4
      CALL CHOUT(IX(I))
      CALL CHOUT(10)
      CALL CHOUT(8)
40    CONTINUE
      IC=IC+1
      IF(IC.LE.NW)GO TO 30
      RETURN
      END
PIP>
```

CHPLOT VARIABLES

ANPLT-	number of plots on the graph (a real number)
CRLX-	integer length of X axis
CRLY-	integer length of Y axis
CRLYR-	variable used in checking plotting efficiency
IC-	loop counter for the number of words to be printed
IFILE-	plot file name (29 characters)
IFLAGX-	determines whether to reset limiting X values (0-yes, 1-no)
IFLAGY-	determines whether to reset limiting Y values (0-yes, 1-no)
IG-	grid line variable (-1 for grid lines, 1 for no grid lines)
IGX-	indicator for grid annotation (.LT.0=grid lines, .GT.0=tic marks)
INET-	determines if limiting values have been reset (1-yes, 0-no)
INRLX-	integer number of divisions on the X axis
INRLY-	integer number of divisions on the Y axis
INRLYR-	variable used in checking plotting efficiency
IR-	represents the next plotting option
IS-	used as the unit number that data is read from or written to
ISTOPF-	new file decision (0-yes, 1-no)
ISTOPT-	new title decision (0-yes, 1-no)
IVECT-	determines whether to call VECTOR (0-no, .GT.0-yes)
K-	counter for which channel is being plotted
L-	counter for point being plotted
NCH(1..5)-	channel numbers to go on the plot
NPLT-	number of plots on the graph (max=5)
NPT-	number of points on the plot
NPT(1..5)-	number of points on the plot for five channels

NRDAT- associate variable for opening file

NSCH- number of channels at the site

NT- number of channels at the site + 1 for time

NW- the number of words to be printed (4 characters to a word)

RLX- length of the X axis

RLY- length of the Y axis

RLYR- variable used in checking plotting efficiency

RS- variable for screen location

STX- starting X value on axis

STY- starting Y value on axis

STYR- variable used in checking plotting efficiency

TICX- length between tic marks on the X axis

TICY- length between tic marks on the Y axis

TICYR- variable used in checking plotting efficiency

TPX- number of tekpoints available in the X direction

TPY- number of tekpoints available in the Y direction

TY- final value on the Y axis

XFACT- X scaling factor (tekpoints/scaled unit)

XL- variable used for plotting (used in calling TPLOT)

XLZ- used in labeling axes after data plotting is complete

XMAX- maximum abscissa value

XMIN- minimum abscissa value

XO- origin abscissa location for the grid in absolute screen coordinates referenced from screen (0,0)

XOS- used in checking plotting efficiency (used in calling SCALE)

XS- T4014 abscissa location

XT- final X value on axis

YFACT-	Y scaling factor (tekpoints/scaled units)
YFACTR-	used in checking plotting efficiency
YL-	used for plotting (used in calling TPLOT)
YMAX-	maximum ordinate value
YMIN-	minimum ordinate value
YO-	origin ordinate location for the grid in absolute screen coordinates from screen (0,0)
YOM-	used in checking plotting efficiency
YOR-	used in checking plotting efficiency
YOS-	used to check plotting efficiency (used in calling SCALE)
YS-	T4014 ordinate location

CHPLOT ARRAYS

DATA(100)-	contains data for one record
IFILE(30)-	plot file name
IFILET(50)-	plot caption
PID(4)-	plot caption for left side
SAV(1..5)(300)-	contains channel data for one day
SX(1..5)(300)-	units for the channels on the plot
TIME(300)-	stores time data for one day
X(300)-	contains X values of points for one day
XTITL(4)-	contains the units for the independent parameter
Y(1..5)(300)-	contains Y values of the points for 5 channels for one day
UNITS(4)-	contains the units for the dependent parameter

APPENDIX E

WRTCNL SOURCE LISTING

```
      SUBROUTINE INPUT(ICHNLS,NCH,NOUT,XNAMES,XNUM)
      DIMENSION ICHNLS(1),XNAMES(1),XNUM(1)
      LOGICAL*1 IFILE(30)
      COMMON/FILE/IFILE
      WRITE(1,10)
10    FORMAT(T2,'ENTER NO. OF CHANNELS AT SITE ')
      READ(1,15) NCH
15    FORMAT(I2)
      WRITE(1,20)
20    FORMAT(T2,'ENTER THE NO. OF CHANNELS TO BE OUTPUT (10 MAX)')
      READ(1,15) NOUT
      WRITE(1,25)
25    FORMAT(T2,'ENTER THOSE CHANNELS NOS. AS: 01,02,03,......')
      READ(1,30) (ICHNLS(I),I=1,NOUT)
30    FORMAT(10(I2,1X))
      WRITE(1,35)
 35   FORMAT(T2,'ENTER MEASUREMENT TYPE , AND NUMBER, AS:
     *TT100,EP101,WW100,.....'/' NOTE: PROGRAM EXPECTS 2 LETTER
     * IDENTIFICATION OF MEASUREMENT TYPE-'/' SINGLE LETTER MEASUREMENT
     * TYPES SUCH AS TEMPERATURE'/' SHOULD BE INPUT AS TT,WW,ETC.')
      READ(1,40) (XNUM(I),XNAMES(I),I=1,NOUT)
40    FORMAT(10(A2,A3,1X))
      WRITE(1,45)
45    FORMAT(T2,'ENTER THE DATA FILE NAME OF THE STORED DATA')
      READ(1,50) (IFILE(I),I=1,29)
50    FORMAT(29A1)
      RETURN
      END
PIP>
```

```
      SUBROUTINE OUTPUT(ICHNLS,DATA,XNAMES,XNUM,NOUT,NCH)
      DIMENSION ICHNLS(1),XNAMES(1),XNUM(1),DATA(1),DAT(96)
      LOGICAL*1 IFILE(30)
      COMMON/FILE/IFILE
      J=1
      NRDAT=1
      ICNT=1
      WRITE(3,3) IFILE
3     FORMAT(5X,29A1)
      WRITE(3,1) (XNUM(I),XNAMES(I),I=1,NOUT)
 1    FORMAT(5X,'TIME',4X,10(3X,A2,A3,3X)//)
 2    READ(2'NRDAT,END=30,ERR=30) (DAT(I),I=1,NCH+1)
      DO 10 I=1,NOUT
      JJ=ICHNLS(I)+1
 10   DATA(I)=DAT(JJ)
      IF(ICNT .LE. 50)GO TO 20
      WRITE(3,4)
    4 FORMAT(1H1)
      WRITE(3,3) IFILE
      WRITE(3,1) (XNUM(I),XNAMES(I),I=1,NOUT)
      ICNT=1
   20 NRDAT=NRDAT+1
      WRITE(3,15) DAT(1),(DATA(I),I=1,NOUT)
 15   FORMAT(F10.4,10F11.2)
      ICNT=ICNT+1
      J=J+1
      IF(J.GT.270) GO TO 30
      GO TO 2
 30   CLOSE(UNIT=3,DISPOSE='PRINT',ERR=900)
 900  RETURN
      END
PIP>
```

WRTCNL:

```
      DIMENSION DATA(10),ICHNLS(10),XNAMES(10),XNUM(10)
      LOGICAL*1 IFILE(30)
      COMMON/FILE/IFILE
      OPEN(UNIT=3,NAME='PRT.LST',TYPE='NEW',ERR=900)
      CALL ERASE
      CALL INPUT(ICHNLS(1),NCH,NOUT,XNAMES(1),XNUM(1))
      NT=NCH+1
      OPEN(UNIT=2,NAME=IFILE,TYPE='OLD',ACCESS='DIRECT',
     *RECORDSIZE=NT,ASSOCIATEVARIABLE=NRDAT,ERR=900)
      CALL OUTPUT(ICHNLS(1),DATA(1),XNAMES(1),XNUM(1),NOUT,NCH)
      CLOSE(UNIT=2,ERR=900)
  900 STOP
 PIP> END
```

WRTCNL VARIABLES

ICNT-	line counter (50 lines/page)
NCH-	number of channels at the site
NOUT-	number of channels to be output
NRDAT-	associate variable for opening IFILE
NT-	number of channels at the site + 1 for time

WRTCNL ARRAYS

DAT(96)-	array to store data for time and channels
DATA(1)-	stores data for each channel to be output
ICHNLS(10)-	array of channel numbers to be output
IFILE(30)-	data file name of stored data
XNAMES(10)-	array of measurement number
XNUM(10)-	array of measurement type (TT,WW,etc.)

APPROVAL

THREE COMPUTER CODES TO READ, PLOT, AND TABULATE OPERATIONAL TEST-SITE RECORDED SOLAR DATA

By

Stephen D. Stewart, Robert J. Sampson, Jr., Richard E. Stonemetz, and
Sandra L. Rouse

The information in this report has been reviewed for technical content. Review of any information concerning Department of Defense or nuclear energy activities or programs has been made by the MSFC Security Classification Officer. This report, in its entirety, has been determined to be unclassified.

Thomas P. Isbell

THOMAS P. ISBELL
Director, Systems Analysis and Integration Lab.

WILLIAM A. BROOKSBANK, JR.
Manager, Solar Energy Applications Projects

☆ U.S. GOVERNMENT PRINTING OFFICE 1980--640-247/85 REGION NO.4