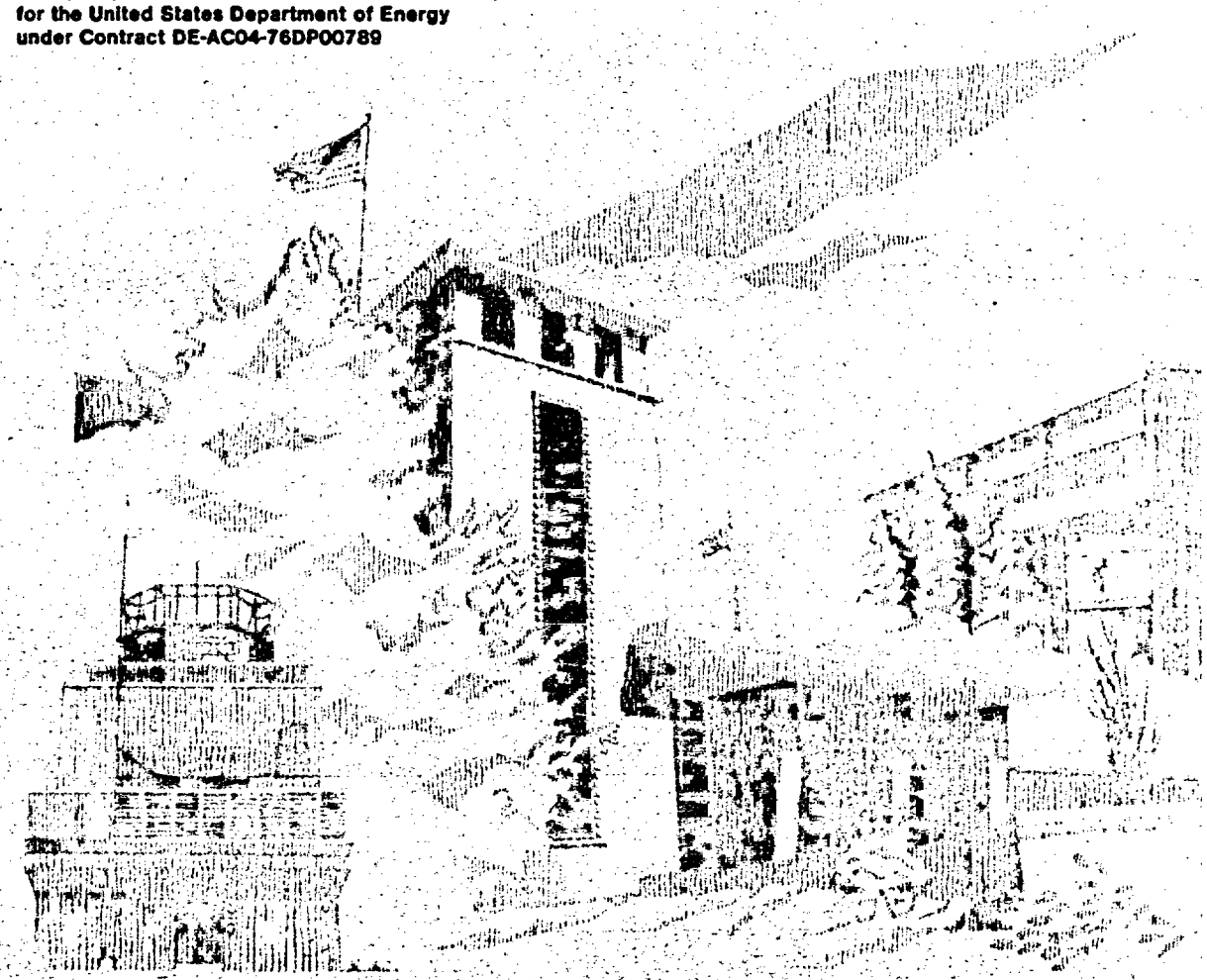
WM-277

SANDIA REPORT SAND81-1059 • Unlimited Release • UC-70
Printed October 1981

In Situ Tuff Water Migration/ Heater Experiment: The Data Acquisition and Playback System

David R. Waymire, Carl O. Duimstra

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550
for the United States Department of Energy
under Contract DE-AC04-76DP00789



SF2900Q(8-81)

HYDROLOGY DOCUMENT NUMBER 184

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.
NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof or any of their contractors or subcontractors.

Printed in the United States of America
Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161

NTIS price codes
Printed copy: $7.00
Microfiche copy: A01

SAND81-1059
Unlimited Release
Printed October 1981

Distribution
Category UC-70

In Situ Tuff Water Migration/ Heater Experiment: The Data Acquisition and Playback System

David R. Waymire
Experiments Planning Division 1112

Carl O. Duimstra
Field Instrumentation Division I, 1124

Sandia National Laboratories
Albuquerque, NM 87185

Abstract

An HP9845T calculator/controller was used as part of an HP3052A Data Acquisition System to collect data during the Tuff Water Migration/ Heater experiment.

The 3052 system was used to log data during the actual thermal experiment, the pretest background, and cool-down periods. The system was also used for various gas pressurization, water level calibration, and rockmass permeability tests. After each test the HP9845 was used to reduce the data and furnish data listings and graphs as required by the experimenters. Various software programs are listed and explained as an aid in the understanding of the test data and the capabilities of the HP3052A system.

Acknowledgments

The authors acknowledge the help of J. Keith Johnstone, Division 4537, who was the Principal Investigator for the In Situ Tuff Water Migration/Heater field experiment, and who provided input for data acquisition and reduction criteria. John T. Lindman, Division 1123,operated, and with Ted J. Stetz, Division 1124, performed the field installation of the entire Data Acquistion System.

Contents

Introduction 7
Data Acquisition System Hardware 8
Heater Test Control/Data Acquisition Software 11
Test Data Printing Software 13
Software for Data Time History Plotting 13
Isotherms and Temperature vs Radius Plotting 16
  Thermal Profiles 16
  Isotherm Plotting 17
Flow Permeability Test Data Analysis 17
Additional Software 18
Conclusions 18
References 19
APPENDIX A — TUFFF Program 21
APPENDIX B — AUTSTE Program 35
APPENDIX C — PRINTC Program 43
APPENDIX D — WATERQ Program 49
APPENDIX E — PLOTE Program 59
APPENDIX F — SEMLGC Program 69

Illustrations

Figure

1 Water Migration Field Experiment 7
2 Block Diagram of the Tuff Water Migration Field Experiment for DAS 8
3 Sample of Graphs With Both Left- and Right-Hand Side Scaling 14
4 Temperature vs Radius for Heater-Midplane Thermocouples During the In Situ Tuff Water Migration/Heater Experiment 16
5 Sample Isotherm Plots for the In Situ Tuff Water Migration/ Heater Experiment 16
6 WM-1 Shut-In Pressures for Posttest In Situ Permeability Tests 18

Tables

1 Tuff Channel ID Listing, Revision D 9
2 Real-Time Scan Data for All Channels Listed in Table 1 for Scan 1315 Hours on April 3 11

In Situ Tuff Water Migration/ Heater Experiment: The Data Acquisition and Playback System

Introduction

During FY 1980 an in situ water migration/heater experiment was carried out in the Grouse Canyon Welded Tuff formation in the Area 12 sector of the Nevada Test Site.[1] The experiment was conducted in the G-Tunnel complex. It was part of the field program of Geological Projects Division 4537 at Sandia National Laboratories, Albuquerque (SNLA) and the Nevada Nuclear Waste Storage Investigations (NNWSI) project of the Nevada Operations Office of the Department of Energy. The details of these experiments and the reduced data are in Reference 2.

The major field activity of this particular project was the water migration/heater experiment, in which thermal, thermomechanical, and water effects were studied. The experiment hole layout is shown in Figure 1.

The heater hole labeled HH-1 had a 4-ft-long heater, 4 in. in diameter. The heater was inserted to a depth of 59.5 to 63.5 ft from an alcove. Packers at about 54.5 and 56 ft sealed the heated section of the hole and facilitated the measurement of pressures, relative humidity, pH, and fluid levels during the test. Type E thermocouples were used to monitor heater and cavity temperatures.

The holes labeled WM-1, WM-2, and WM-3 contained instruments to record temperature, pressure,

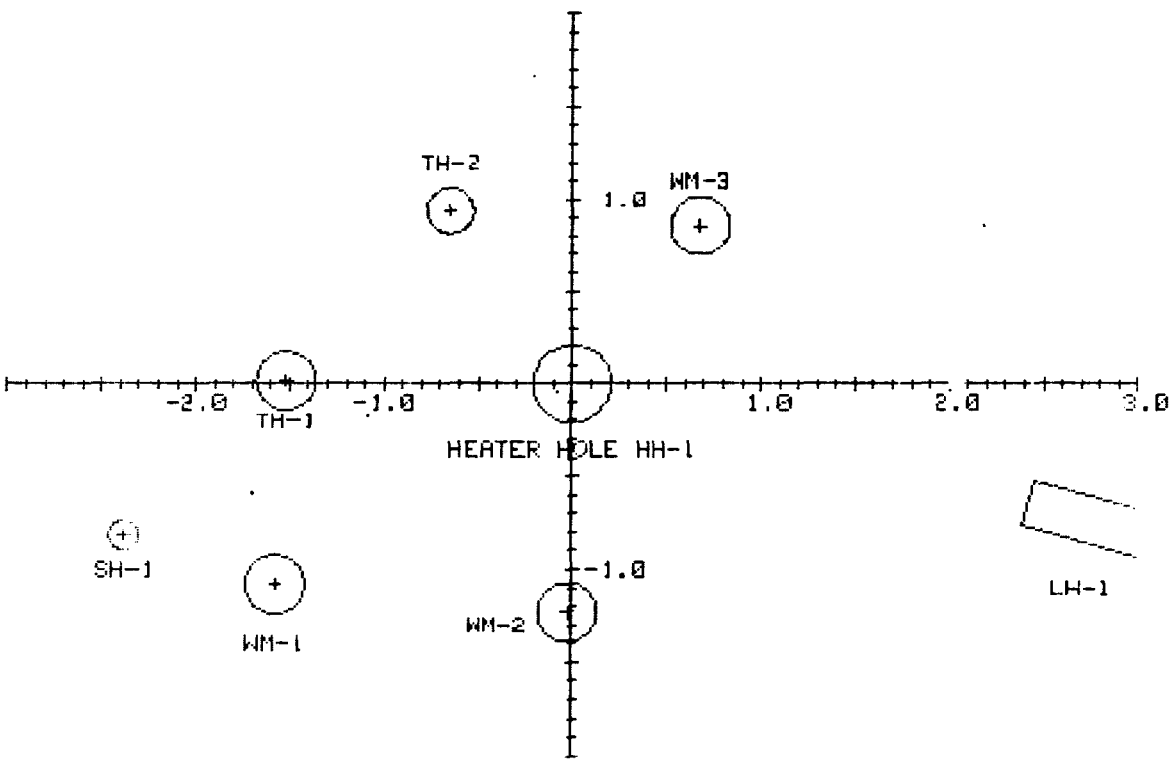


Figure 1. Water Migration Field Experiment. Hole array layout at hole collar (dimensions in feet)

relative humidity, pH, and fluid level at locations nominally 15 and 24 in. from the heater centerline at a depth about equal to the heater midplane at 61.5 ft.

Three strain-gauged platten type "stress" gauges were fielded in SH-1 at about the heater midplane, approximately 32 in. from the heater centerline. A stabilized LASER interferometer was fielded in LH-1 to measure rock displacements.

Type E thermocouples were fielded at various locations in holes TH-1 and TH-2 to obtain thermal profiles. TH-1 was about 25 in. and TH-2 about 15 in. from the centerline of HH-1.

A list of the data channels monitored during the field experiment is given in Table 1. One hundred and twenty separate channels were monitored.

The HP3052A Data Acquisition System (DAS), with an HP9845T calculator/controller, was used to collect baseline data for several weeks before turning on the heater. The DAS was used also to log data during the heating and cool-down phases of the test.

During and after the in situ field test, the DAS was used for instrument calibration. The DAS was also used to log data during in situ fluid permeability tests and used to reduce and print/plot all the data accumulated during the field and calibration tests.

Software for the DAS was written, tested, and used for the data acquisition and control of the water migration field experiment.

Data Acquisition System Hardware

The basic hardware included in the DAS is shown in Figure 2. Details of the system are discussed in a separate report.[3] The HP9845T constitutes the most important unit of the system. The 9845 uses the interrupt capability of a real-time clock (Part No. HP98035A) to pace the data collection function. Scan rates are programmable from the 9845 keyboard. The instrument output data were also accumulated in the 9845 where conversions were made to account for instrument offset and sensitivity, and appropriate engineering units were applied. Current instrument status, real-time, safety warnings, and data storage information were continuously displayed on the internal cathode ray tube (CRT). The internal line printer was used to obtain a hard copy of the displayed data. Data of this type are shown in Table 2.

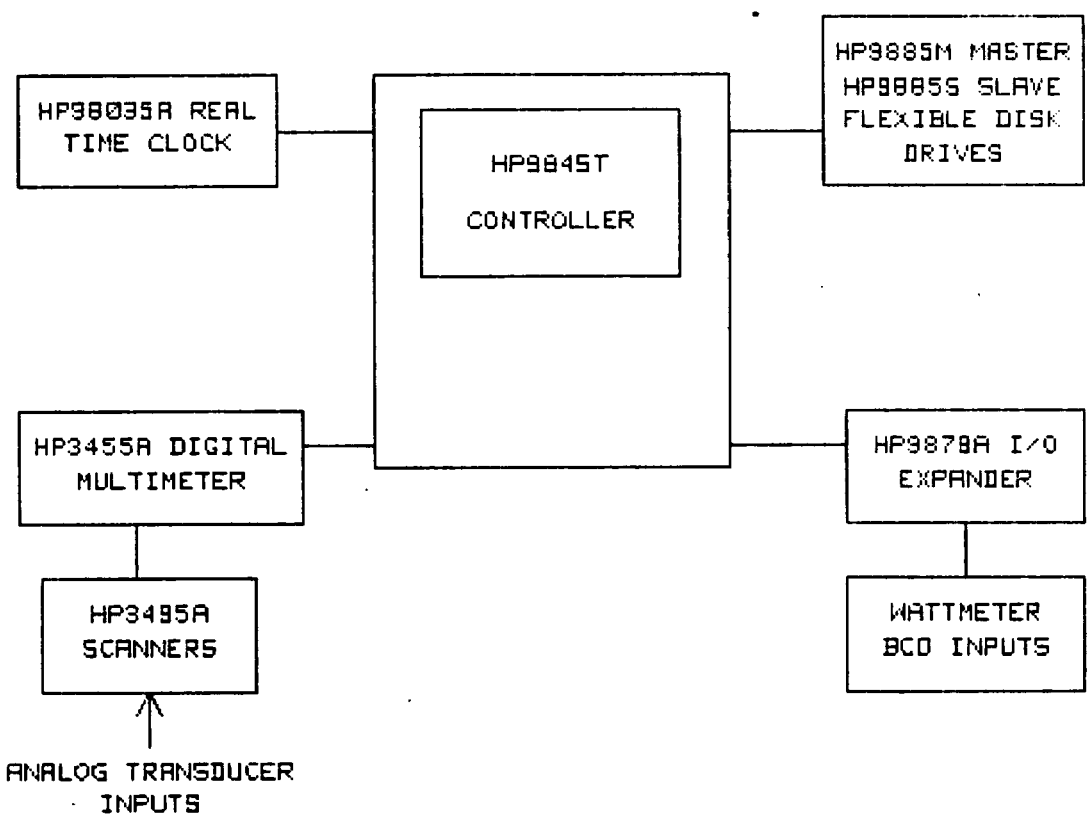


Figure 2. Block Diagram of the Tuff Water Migration Field Experiment for DAS

Table 1. Tuff Channel ID Listing, Revision D (Last Update and Date of Current Printing 3/28/80)

Ch No.	ID	Gage Type	Location	Units	Calibration
0	RF-1	Reference Junc	Card 1	Ohms	SNLA
1	T-1	Type E TC 10 TCM	Heater Element E-1	°C	NBS
2	T-2	Type E TC 11	Heater Element E-2	°C	NBS
3	T-3	Type E TC 12	Insul Section Hi	°C	NBS
4	T-4	Type E TC 13	Insul Section Lo	°C	NBS
5	T-5	Type E TC 14	Junction Section	°C	NBS
6	T-6	Type E TC 15	Heater Instr Air	°C	NBS
7	T-7	Type E TC 16	Heater Instr H20	°C	NBS
8	T-8	Type E TC 6	Near Heater End 2	°C	NBS
9	T-9	Type E TC 3 TCM	Near Heater End 6	°C	NBS
10	T-10	Type E TC 9	Near Heater End 10	°C	NBS
11	T-11	Type E TC 5	Heater Midplane 2	°C	NBS
12	T-12	Type E TC 2 TCM	Heater Midplane 6	°C	NBS
13	T-13	Type E TC 8	Heater Midplane 10	°C	NBS
14	T-14	Type E TC 4	Far Heater End 2	°C	NBS
15	T-15	Type E TC 1 TCM	Far Heater End 6	°C	NBS
16	T-16	Type E TC 7	Far Heater End 10	°C	NBS
17	T-17	Type E TC 22	Near Heater Rock 12	°C	NBS
18	T-18	Type E TC	Bkgd Standard-Ice	°C	NBS
19	Cal V	Analogic	Card 1 Calibration	Volts	SNLA
20	RF-2	Reference Junc	Card 2	Ohms	SNLA
21	T-19	Type E TC 19	Near Heater Rock 4	°C	NBS
22	T-20	Type E TC 25	Near Heater Rock 8	°C	NBS
23	T-21	Type E TC 21	Mid-Heater Rock 12	°C	NBS
24	T-22	Type E TC 18 TCM	Mid-Heater Rock 4	°C	NBS
25	T-23	Type E TC 24	Mid-Heater Rock 8	°C	NBS
26	T-24	Type E TC 20	Far Heater Rock 12	°C	NBS
27	T-25	Type E TC 17	Far Heater Rock 4	°C	NBS
28	T-26	Type E TC 23	Far Heater Rock 8	°C	NBS
29	T-27	Type E TC 1	WM-1 Water 62.0 ft	°C	NBS
30	T-28	Type E TC 2	WM-1 Air 62.17 ft	°C	NBS
31	T-29	Type E TC 3	WM-1 Left 62.3 ft	°C	NBS
32	T-30	Type E TC 4	WM-1 Right 62.3 ft	°C	NBS
33	T-31	Type E TC 1	WM-2 Water 61.5 ft	°C	NBS
34	T-32	Type E TC 2	WM-2 Air 61.67 ft	°C	NBS
35	T-33	Type E TC 3	WM-2 Left 61.8 ft	°C	NBS
36	T-34	Type E TC 4	WM-2 Right 61.8 ft	°C	NBS
37	T-35	Type E TC 1	WM-3 Water 61.08 ft	°C	NBS
38	T-36	Type E TC	Bkgd Standard-Ice	°C	NBS
39	Cal V	Analogic	Card 2 Calibration	Volts	SNLA
40	Ref-2	Reference Junc	Card 3	Ohms	SNLA
41	T-37	Type E TC 2	WM-3 Air 61.25 ft	°C	NBS
42	T-38	Type E TC 3	WM-3 Left 61.38 ft	°C	NBS
43	T-39	Type TC 4	WM-3 Right 61.38 ft	°C	NBS
44	T-40	Type E TC	TH-1 53.5 ft	°C	NBS
45	T-41	Type E TC	TH-1 56.5 ft	°C	NBS
46	T-42	Type E TC	TH-1 59.5 ft	°C	NBS
47	T-43	Type E TC	TH-1 60.5 ft	°C	NBS
48	T-44	Type E TC	TH-1 61.5 ft	°C	NBS
49	T-45	Type E TC	TH-1 62.5 ft	°C	NBS
50	T-46	Type E TC	TH-1 63.5 ft	°C	NBS
51	T-47	Type E TC	TH-1 64.5 ft	°C	NBS
52	T-48	Type E TC	TH-1 69.5 ft	°C	NBS
53	T-49	Type E TC	TH-2 56.5 ft	°C	NBS
54	T-50	Type E TC	TH-2 58.5 ft	°C	NBS
55	T-51	Type E TC	TH-2 60 ft	°C	NBS
56	T-52	Type E TC	TH-2 60.5 ft	°C	NBS
57	T-53	Type E TC	TH-2 61 ft	°C	NBS
58	T-54	Type E TC	Bkgr Standard-Ice	°C	NBS

Table 1 (cont)

59	Cal V	Analogic	Card 3 Calibration	Volts	SNLA
60	RF-4	Reference Junc	Card 4	Ohms	SNLA
61	T-55	Type E TC	TH-2 61.5 ft	°C	NBS
62	T-56	Type E TC	TH-2 62 ft	°C	NBS
63	T-57	Type E TC	TH-2 62.5 ft	°C	NBS
64	T-58	Type E TC	TH-2 63 ft	°C	NBS
65	T-59	Type E TC	TH-2 63.5 ft	°C	NBS
66	T-60	Type E TC	TH-2 64.5 ft	°C	NBS
67	T-61	Type E TC	TH-2 65.5 ft	°C	NBS
68	T-62	Type E TC	TH-2 67.5 ft	°C	NBS
69	T-63	Type E TC	Stress 62.3 ft	°C	NBS
70	T-64	Type E TC	Stress 62.3 ft	°C	NBS
71	T-65	Type E TC	LH-1 29.8 ft	°C	NBS
72	T-66	Type E TC	LH-2 28.7 ft	°C	NBS
73	T-67	Type E TC	Instr Alcove	°C	NBS
74	T-68	Type E TC	Laser 28.7 ft	°C	NBS
75	T-69	Type E TC	Alcove Rock Wall	°C	NBS
76	T-70	Type E TC	Instr Rack	°C	NBS
77	T-71	Type E TC	Laser Alcove	°C	NBS
78	T-72	Type E TC	Bkgd Standard-Ice	°C	SNLA
79	Cal V	Analogic	Card 4 Calibration	Volts	SNLA
80	PR-1	Pressure SN 1002	Heater Hole	psi	0.0427 V/psi
81	PR-2	Pressure SN 1025	Heater Hole	psi	0.0480 V/psi
82	PR-3	Pressure SN 1005	Hole WM-1	psi	0.0430 V/psi
83	PR-4	Pressure SN 1017	Hole WM-1	psi	0.0482 V/psi
84	PR-5	Pressure SN 1004	Hole WM-2	psi	0.0433 V/psi
85	PR-6	Pressure SN 1018	Hole WM-2	psi	0.0481 V/psi
86	PR-7	Pressure SN 1006	Hole WM-3	psi	0.0430 V/psi
87	PR-8	Pressure SN 1023	Hole WM-3	psi	0.0490 V/psi
88	PR-9	Pressure SN 1020	Alcove	psi	0.0476 V/psi
89	S-1	Stress SE-28	SH-1 0° 62.75 ft	mV	SNLA
90	S-2	Stress SE-7	SH-1 0° 62.29 ft	mV	SNLA
91	S-3	Stress SE-13	SH-1 90° 61.5 ft	mV	SNLA
92	LAS-2	Laser Strain	LH-1	Micron	10 V/4096 ct
93	PH-1	PH Sensor R11713	Heater Hole	PH	SNLA
94	PH-2	PH Sensor R11714	Alcove	PH	SNLA
95	PH-3	PH Sensor R11840	WM-2	PH	SNLA
96	PH-4	PH Sensor R11712	WM-3A	PH	SNLA
97	RH-2	Rel Humid SN 501	WM-1	%RH	TH Sci
98	RH-1	ReL Humid SN 502	Heater Hole	%RH	TH Sci
99	Cal V	Analogic	Card 5 Calibration	Volts	SNLA
100	RH-4	Rel Humid SN 503	WM-3A	%RH	TH Sci
101	RH-3	Rel Humid SN 504	WM-2	%RH	TH Sci
102	V-1	Element Voltage	Heater Hole	Volts	SNLA
103	A-1	Element Current	Heater Hole	Amps	SNLA
104	P-1	Heater Power	Heater Hole	Watts	SNLA
105	V-2	Element Voltage	Heater Hole	Volts	SNLA
106	A-2	Element Current	Heater Hole	Amps	SNLA
107	P-2	Heater Power	Heater Hole	Watts	SNLA
108	WAT-1	Water Level	Heater Hole	Volts	SNLA
109	WAT-2	Water Level	WM-1	Volts	SNLA
110	WAT-3	Water Level	WM-2	Volts	SNLA
111	WAT-4	Water Level	WM-3	Volts	SNLA
112	LAS-1	Laser Strain	LH-1	Micron	20 V/4096 ct
113	Cal V	Analogic	Card 6 Calibration	Micron	SNLA
114	TH-1	Thermistor	TC-1 58.5 ft	°C	SNLA
115	TH-2	Thermistor	TC-1 67.5 ft	°C	SNLA
116	TH-3	Thermistor	TC-2 59.5 ft	°C	SNLA
117	TH-4	Thermistor	TC-2 69.5 ft	°C	SNLA
118	P-1D	Heater Power	Heater Hole	Watts (BCD)	SNLA
119	P-2D	Heater Power	Heater Hole	Watts (BCD)	SNLA
120	P-3D	Heater Power	Total Power	Watts (BCD)	SNLA

Table 2. Real-Time Scan Data for all Channels Listed in Table 1 for Scan 1315 Hours on April 3

DATE AND TIME 04:03:13:15:00

CHANNEL	01	02	03	04	05	06	07	08	09	10
000	430.12	441.89	147.23	54.30	48.13	36.72	33.41	264.64	258.00	262.69
010	300.99	300.95	298.95	278.93	272.14	282.10	183.91	-.02	107.27	29.50
020	181.90	175.84	235.70	225.28	222.69	194.75	187.92	195.95	54.73	55.97
030	53.23	56.39	70.22	72.77	72.63	72.68	74.33	.03	106.68	29.18
040	76.13	76.68	72.27	24.68	35.21	56.21	60.31	60.70	52.62	45.75
050	42.64	23.15	36.96	60.82	83.56	87.89	87.32	.38	106.39	28.73
060	86.53	82.58	78.15	73.55	68.91	52.03	44.15	29.43	47.31	47.22
070	43.49	34.69	22.96	30.07	22.59	25.71	34.19	.22	106.00	-.29
080	-.68	-1.62	-2.14	-.12	5.78	-1.96	-2.32	-.18	.01	.02
090	.01	271.58	6.52	7.07	6.53	9.57	-117.48	87.49	10.00	-.01
100	.01	8.07	7.22	383.15	53.23	7.21	379.32	.52	.39	.48
110	.56	258.28	163.62	71.40	24.01	48.35	30.10	383.00	379.00	762.00

After each scan the data was stored on prime and backup flexible disks. After 20 scans the data was stored also on cassette tape.

The major portion of the data was collected with HP3495A scanners. Analog voltage or resistance signals were wired to the scanner plug-in boards. Upon command from the 9845, scanner relays were closed sequentially to allow a reading by the system digital multimeter (Model No. HP3455A). Two scanners were used to provide analog data input and relay closures for pressure safety, power outage, and heater overtemperature diagnostics. Binary coded decimal (BCD) input interfaces were used to record digital heater power data from the Magtrol power meters.

Heater Test Control/Data Acquisition Software

The software program, TUFFF, used to control the scan rates and accumulate data during the heater operation is listed in Appendix A. In the event that power is lost to the HP9845T during unattended operation, an auto-start program, AUTSTE (Appendix B), could be "booted up" automatically upon return to the power-on state. AUTSTE is similar in most respects to the normal operating program. The major differences between TUFFF and AUTSTE are the capability of AUTSTE to set automatically a predetermined scan rate to begin at the next even hour or 10-min mark after power is restored. It also uses different data storage file names, which prevents overwriting previously stored data. Since TUFFF and AUTSTE are by and large the same program, only TUFFF will be discussed in detail.

TUFFF is loaded into the HP9845 read/write memory by inserting the cassette tape or disk containing the program into the appropriate mass storage device and typing GET "TUFFF:T15" ***EXECUTE***, if it is on cassette tape in the right-hand "T15" drive, or GET "TUFFF:F8,0" ***EXECUTE***, if it is on flexible disk in the master disk drive, F8,0 (Model No. HP9885M). In these commands and in all others used in this document, italicized words or commands refer to the depression of the appropriately labeled keyboard key.

After the program is loaded into the 9845 it is run by depressing the ***RUN*** key.

The first operation after setting aside common blocks is to initialize the real-time clock and the instruments on the IEEE interface bus. The initialization and channel assignments for the various types of transducers and measurement instruments are done in subroutine Init.

Instructions to the operator/user are displayed on the CRT. The first instruction is

> PUT INITIALIZED BLANK CASSETTE (RECORD MODE) IN T14 TAPE DRIVE, PRESS ***CONT***

This instruction requests that cassette tape be inserted into the auxiliary tape drive for the recording of data. After the ***CONT*** key is depressed, a data file DATA1 is created on the cassette.

The CRT then displays the instruction:

> PUT INITIALIZED BLANK FLOPPY INTO MASTER DRIVE, PRESS ***CONT***

This instruction requests that a flexible disk be placed in the master disk drive, F8,0. After the ***CONT*** key is depressed, a data file DATA1 is created on the master disk preparatory to storing data.

A question is then asked as to whether or not an auxiliary disk drive is being used as a backup storage device. If the response to this query is YES then the instruction

INSERT INITIALIZED BLANK FLOPPY
INTO SLAVE DRIVE, PRESS ***CONT***

is displayed. After the depression of the ***CONT*** key, a data file DATA1 is created on the slave disk drive ":F8,1" to be used as backup data storage.

The next operation taking place between program steps 330 and 380 is to set up Special Function Key (SFK) interrupts that enable the operator to cause end-of-the-line branching to particular program segments by the simple pressing of selected SFKs. This is done to allow changes to be made to the normal program operation.

SFK#0 causes branching to the KEY#0 PROCESSOR subprogram, allowing the operator to change the scan rate at which the real-time clock causes data to be input to the 9845. Input is accomplished by sequentially scanning through all the data input channels and taking readings of transducer or device outputs.

SFK#1 causes branching to the KEY#1 PROCESSOR subprogram, forcing a data scan independent of any real-time clock interrupts.

SFK#3 causes branching to the KEY#3 PROCESSOR subprogram, calling subroutine Tplot and providing a plot on the CRT and internal line printer. This subroutine plots the thermocouple temperature data as a function of the thermocouple placement in the experimental test array. To conserve memory and allow for more rapid recovery following power outages, this subprogram and associated plotting subroutines were not included in the AUTSTE version of this program.

SFK#5 causes branching to the KEY#5 PROCESSOR subprogram, causing instantaneous display of the current clock time.

SFK#7 causes branching to the KEY#7 PROCESSOR subprogram, calling subroutine Rplot. This subroutine plots the temperature along the heater midplane as a function of radius for the last set of temperature data acquired. This subprogram and associated subroutines, like those used with SFK#3, were not included in the AUTSTE version.

SFK#15 causes branching to the KEY#15 PROCESSOR subprogram, halting the program immediately. The data files on all storage devices are updated and then closed.

After the Special Function Keys have been defined subroutine Setclock allows the operator to make any required corrections to the real-time clock output. The operator is then requested to enter data pertinent to the clock interrupts. The first instruction

ENTER THE DESIRED SCAN TIME IN
MINUTES

requires the operator to enter the appropriate scan interval. When the operator presses ***CONT***, the program proceeds. After displaying the entered value for operator verification, the interval time is converted to milliseconds, which is the value the real-time clock actually uses.

The program then requires input regarding the clock time when the scanning should begin. The match time for the clock must be entered by using the format HR:MN:SC, with each variable using two digits, and must be separated from the other variables in the list by a delimiter other than a comma.

Operation of the program continues automatically with the 9845 staying in wait loops until interrupted by the clock for scanning and subsequent data conversion and storage, or by the operator by means of SFKs, as defined above.

After receiving a real-time clock match to the specified scan commencement time or subsequent interval interrupts, the program branches to subroutine Monitor. This subroutine calls subroutine Scan, which alternately sets up the Digital Multimeter (DMM) to the proper function and range and calls subroutine Measure.

The actual scanning, reading, and unit conversions for the inputted data are all accomplished by subroutine Measure. The converted input data is also stored as an element in the array Table 2 (Scans, Ch), according to the scan and channel numbers.

Subroutine Scan also causes data to be input from the Magtrol power meters by the BCD interfaces.

After the scanning operation is complete, control returns to subroutine Monitor. At this time the entire updated arrays, Table$ (Scans) and Table 2 (Scans,Ch), as well as the current scan number are stored on the flexible disks. The array Table$ (Scans) contains the real-time of each scan.

The input data for the latest scan, including data storage file and time information, is then printed on the internal printer in the format shown in Table 2.

Control then returns to the main calling program MAIN PROGRAM LOOP and loops between program lines 581 and 650 until interrupted again.

At the conclusion of the 20th scan control jumps to line 1020 in the KEY#15 PROCESSOR. All of the data arrays are stored on all storage media and the data files are closed. New data files are created, and if

the cassette tape is not full, control is returned to MAIN PROGRAM LOOP.

This program also monitors hole pressures for safety considerations, prints warning notes, and closes relays, which switch on warning lights in case pressures above a few psi develop. Additional relays are closed if heater hole temperatures exceed predetermined limits, and an automatic telephone message system is activated.

Test Data Printing Software

The program, PRINTC (Appendix C), was used to read the data cassettes or diskettes and print data acquired by the TUFFF and AUTSTE programs. The type of input medium to be used is specified by answering the display prompt

IS DATA GOING TO BE READ IN ON FLOPPY DISK?

The operator types in either YES or NO and presses the ***CONT*** key. If the answer is NO the default right-hand cassette tape drive is used.

Each recorded data file contains data for 20 scans for all 120 channels. Data can be chained together and stored in so-called "Monster" files to permit printing and subsequent plotting of many scans of a few data channels. These Monster files can be read by the PRINTC program by responding to the following queries displayed on the CRT:

IS DATA ALREADY PRE-RECORDED ON MONSTER FILE?

and

WHAT IS MONSTER FILE NAME?

If data are not currently in a large file it is possible to read in data, in groups of 20 scans each, up to a maximum of 260 scans, the limit set by memory size. Data can be read in by responding to the following questions:

WHAT IS FIRST DATA FILE TO BE READ?
DO YOU WANT TO READ IN MORE DATA, Y OR N?
WHAT IS NEXT DATA FILE TO BE READ?

In addition to listing data on the internal line printer, it is also possible to list data on an external line printer by means of an RS-232C interface.

All the data can be printed in columnized form with high resolution, although only five channels can be listed at one time. The data for each channel are printed in a column chronologically with scan times printed in the first column. Data to four decimal places are printed for each data channel. In this mode, individual channels can be selected for printing. Identification headings must be provided by the operator.

If only one decimal place is desired the data will be printed 10 channels at a time. Channels are printed sequentially in groups of 10. Data for each channel are printed chronologically in a column with the first column containing scan-time data. The data can also be printed in tabular form, grouped together according to scans, in a format identical to Table 2.

After printing options have been exhausted, the data in read/write memory can be stored on disk or tape as a newly created monster file. This is done by properly answering questions found between program lines 1980 and 2120 in Appendix C.

Software for Data Time History Plotting

The program, WATERQ (Appendix D), was developed to plot transducer and instrument output vs time for the water migration/heater field experiment. It is capable of reading disks or tapes produced by TUFFF and AUTSTE programs. To permit relatively easy plotting of large amounts of data for each channel, it was necessary to read in numerous data files, which were acquired during the field experiment, and chain them together in chronological order. This chaining operation required that large files of this type handle data for a limited number of channels. Thus the data were chained in groups of 10 channels each and stored in Monster files that were more convenient for plotting. The group size was dictated by the memory capacity of the calculator/controller and the maximum array size it could access.

The choice of cassette tape or flexible disks as data input media is determined by the answer to the question:

IS DATA GOING TO BE READ IN FROM A FLOPPY DISK? Y OR N

If the answer is NO, the program defaults to cassette tape. The operator then indicates which is the first channel of the 10-channel group to be stored and plotted: for example, Channel 21 would be the first channel of the group containing data for Channels 21 through 30.

If data for this group of channels have previously been chained together, this Monster file can now be read into read/write memory from disk or tape. These data can then be plotted directly, or, if space is available in the storage array, additional data can be added from data tapes or disks. The program tells the operator how much space is left in the arrays. If additional data are desired, a catalog of the tape or

disk is displayed for the operator to view. The operator then inputs the index number of the lowest numbered data file that is to be added to the Monster file and the number of consecutively numbered files that are to be loaded. For example, if data files DATA13 to DATA19 are to be inputted, the operator enters the value 13 in response to the request

WHAT IS THE FIRST DATA FILE NUMBER TO BE READ?

The number of consecutive files to be read would be seven. This number would be entered in response to

HOW MANY CONSECUTIVELY NUMBERED FILES ARE TO BE READ IN?

The data files to be entered using this series of commands must be consecutive. If files from another tape/disk or nonconsecutive files need to be added, they can be appended after the first set has been read. At this point the program automatically returns to ask whether more files are to be added.

After all the desired or allowed data have been chained together for this particular 10-channel group, the data can be saved on disk or tape as a new Monster file for future plotting or printing. A series of instructions is ouput to the operator for accomplishing this task.

The actual plotting segment of the program WATERQ begins in line 1030 at the label Plot. The machine first requires the particular channel number to be plotted. The program is capable of putting different scales on the left- and right-hand sides of the plot. Thus different functions, such as temperature and pressure, can be plotted on the same figure for comparison. A sample of this type of data is shown in Figure 3. The operator must also specify if he is using the HP9872A four-pen plotter. If not, the program defaults to the internal CRT with hard copies being made on the internal line printer. If an external device other than the four-pen plotter is desired, the operator must change program lines 1360 through 1410 to reflect the different select code and/or instruction set for setting the plotting limits. The time limits for the plotting surface X-AXIS are set in response to

WHAT IS THE J-DAY WHEN YOU WANT THE PLOT TO BEGIN?

and a similar instruction for the time the plot is to end.

The Julian day of the year for these responses is necessary. If the beginning or ending time is not to be at midnight, the exact time of day can be included as a decimal fraction. Thus the plotting will begin at 6 a.m. on January first if 1.25 is entered as the J-DAY for the plot begin.

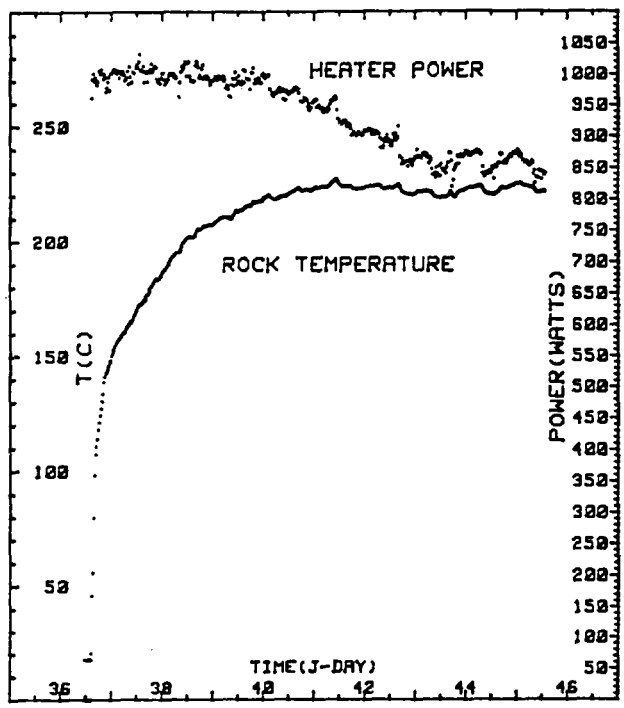


Figure 3. Sample of Graphs With Both Left- and Right-Hand Side Scaling

The software is capable of finding the maximum value of the data for the selected channel. The next higher multiple of 10 is then used for the maximum value of the plotting surface for the left-hand side scaling factor. If the autoscaling option is not desired, the operator may specify minimum and maximum values for the ordinate scale. This might be desirable if several different plots having different maxima are to be compared.

In response to the machine command

ENTER LABEL FOR UNITS ON LEFT-SIDE Y-AXIS

the operator can type in whatever label is appropriate for the left-hand side ordinate axis. The maximum character length of the label is 18.

If a different scale factor has been requested for the right-hand side the commands

ENTER MAXIMUM VALUE DESIRED FOR RIGHT-SIDE Y-AXIS PLOTTING
ENTER MINIMUM VALUE DESIRED FOR RIGHT-SIDE Y-AXIS PLOTTING

and

ENTER LABEL FOR UNITS ON RIGHT-SIDE Y-AXIS

are displayed. As currently written, the software does not do autoscaling for right-hand side scaling. However, steps similar to those between lines 1210 and 1250 could be added to provide this feature.

If the HP9872A four-pen plotter is specified as the plotting device, the operator is asked to provide the decimal fraction of the aspect ratio, Xscale/Yscale, to tell the software what the shape of the paper is. Typically the ratio will be 0.70 (7 in. wide by 10 in. high), 1.50 (15 in. wide by 10 in. high), or similar values for metric ruled paper.

Appropriate axes, tick marks, scales, and labels are then calculated and plotted by using information and commands found between lines 1370 and 2820.

Sometimes the operator does not want to wait for the plotter to plot every data point for a given channel. The option is available to speed up plotting by choosing every 5th, 10th, or 20th point. This is done by answering the question

WHICH OPTION DO YOU WANT?
1-EVERY POINT; 2-EVERY FIFTH;
3-EVERY TENTH;
4-EVERY TWENTIETH?

with an input of 1, 2, 3, or 4 depending on the option desired.

The machine also asks the question

WHICH PEN DO YOU WANT FOR
PLOTTING. 1, 2, 3, or 4?

Any pen color currently mounted on the 9872 four-pen plotter can be accessed depending on the position of its mounting stall. The choice of pen is immaterial if the internal CRT and line printer are used.

Between lines 3010 and 3260, the time values for the data are determined and the data for the specified channel are plotted.

A third-order spline fit curve is available for theoretical (or other operator entered) data. The operator must specify the number of data points which will be entered for fitting and the time and value for each data point. It is possible for the machine to separate the theoretical fitted curve into two segments in case a sharp break occurs in the slope of the data, as was the case in the field experiment at the time the heater power was turned off.

After the data have been plotted, additional operator controlled labeling can be done with the interactive keyboard. Writing can be done anywhere on the plotting surface. Examples of additional labeling are curve identifications, figure captions, dates, and explanations.

The operator lettering mode is entered when the flashing cursor appears in the lower left-hand corner of the CRT. The cursor or the pen on the 9872 plotter can be moved to any desired location using the four arrows found on the upper center section of the HP9845 operator console. When the cursor or pen is located at the start (left side) of the desired writing area, the operator keyboard can then be used as a typewriter. Lettering is slower, especially for the 9872 pen, than normal printing so lettering must be entered rather slowly. Between lines, characters, or labels the cursor (pen) can be moved to other locations by pressing the arrow keys on the keyboard.

When labeling is completed the operator can leave the lettering mode by pressing the ***CONT*** key.

If the CRT is used as the plotting device additions, corrections, or changes can be made. To do this the operator responds with a YES to the question

DO YOU WANT TO ERASE OR CHANGE
ANYTHING, Y OR N?

The erasure mode is then entered and letters can be erased by moving the cursor over the incorrect symbol and retyping the incorrect letter over the original mistake.

After all erasures are completed ***CONT*** is pressed to reenter the lettering mode. The operator then moves the cursor to the correct position and types in the correct lettering or additions desired. The operator again leaves the lettering mode by pressing ***CONT***.

More curves from the same 10-channel group can be plotted on the same figure, and one or more curves can be plotted against the right-hand scale if prescribed.

After all plotting from the original 10-channel group of data has been finished for the figure, other time data for the same 10 channels or other groups of data can be read in from tape or disk and added to the plot. If more data are desired, the program essentially starts over in requesting data files to be read in for inclusion on the plot.

After the new data have been plotted and labeled, a hard copy of data on the CRT is produced on the internal line printer. Additional copies can also be made. Other independent plots with other scale factors can be made for the same or other channels from the data currently in memory. To do this a positive reply must be made to the prompt

DO YOU WANT OTHER PLOTS FROM
THIS DATA?

after which the program returns to the label Plot and new scale factors and channel information are requested for the additional plots. Otherwise, the operator is asked whether or not he wants to read in more data for other plots. If he does, the program returns to request more files to be read from tape or disk. If not, the program terminates.

Isotherms and Temperature vs Radius Plotting

In addition to time history plots obtained through the program WATERQ, discussed in the previous section, isotherms and temperature vs radius plots for a given time can be obtained by using the program PLOTE (Appendix E). An example of the temperature vs radius plot is found in Figure 4, and a sample of the isotherm capability is seen in Figure 5.

Data can be entered from tapes or disks in the same manner as with WATERQ. However, in PLOTE the data for the entire 120 channels are chained together. This is done to allow any thermocouple channels to be plotted. Since each curve is a composite of data taken during one specific scan or time, only selected times or scans and not all consecutive scans and files need to be entered.

The type of curve which is to be plotted is determined by the operator response to the command

THE FOLLOWING OPTIONS ARE AVAILABLE FOR PLOTTING:

OPTION 1: TEMPERATURE AS FUNCTION OF RADIUS FOR SEVERAL TIMES.

OPTION 2: ISOTHERMS FOR A GIVEN TIME.

PICK YOUR OPTION BY PRESSING 1 OR 2 THEN ***CONT***

The operator must also specify whether the HP9872A four-pen plotter is being used. The program then branches depending on the plotting option picked.

Thermal Profiles

The operator requests the number of radii he wants included for temperature vs radius profiles. The radii should correspond to actual radii from the heater hole centerline where thermocouples at the heater midplane are available.

The command

ENTER NUMBER OF TEMPERATURE SCANS (DIFFERENT TIMES) FOR PLOTTING

requests the number of different temperature vs radius curves desired on the same plot. Each such curve is for the same radii but for different times. Choosing appropriate times provides a look at the change of the thermal profile as a function of time.

The radius in meters to each hole or thermocouple of interest must be entered along with the channel number from Table 1 for the thermocouple used at that location.

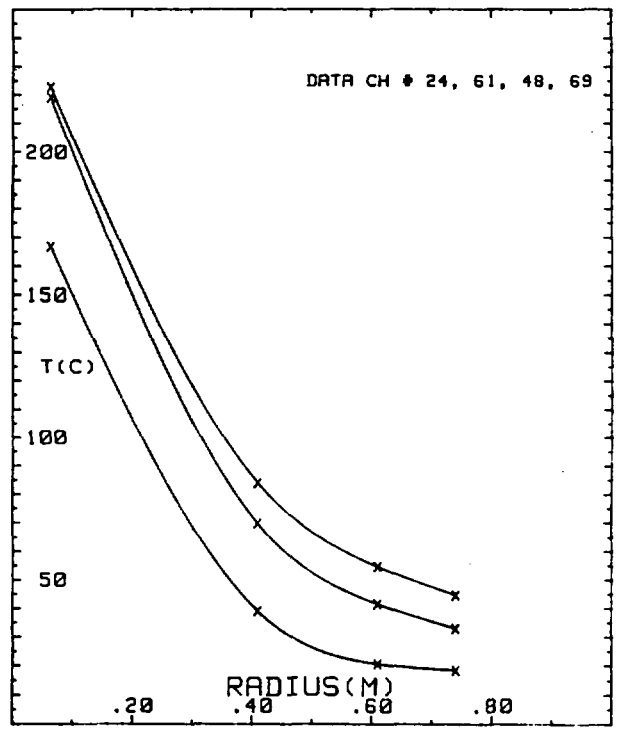


Figure 4. Temperature vs Radius for Heater-Midplane Thermocouples During the In Situ Tuff Water Migration/Heater Experiment

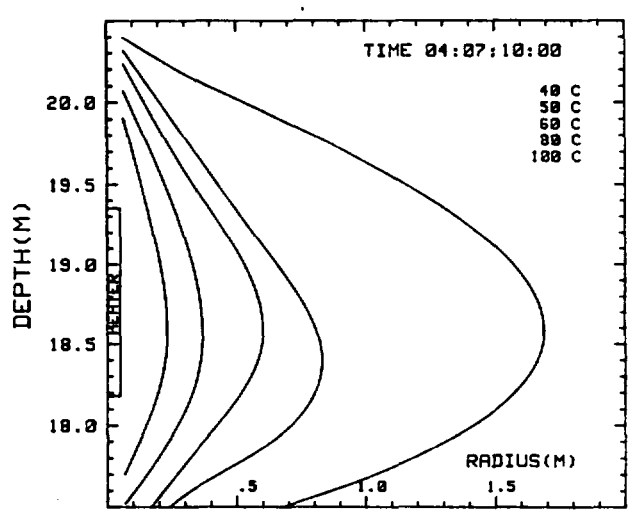


Figure 5. Sample Isotherm Plots for the In Situ Tuff Water Migration/Heater Experiment

The scan number for each curve is entered for the number of curves requested earlier. The scan number is the index for the appropriate scan as found in the chained together file in memory, and not particularly the scan number assigned during data acquisition.

The operator must respond to

DO YOU WANT AUTO SCALING OF Y-AXIS?

Autoscaling is done between lines 690 and 750 in a manner identical to scaling in WATERQ. If autoscaling is not desired, the operator is requested to supply the maximum value for the Y-AXIS (temperature) scale.

Program lines 780 to 1230 calculate and draw axes, labels, and scales prior to data plotting. Before data are plotted, the question of which pen of the four colors (mounting stall positions) in the 9872 plotter is asked. If the 9872 plotter is not in use the reply has no effect. The actual data points are plotted according to commands between lines 1280 and 1410.

Following data plotting the program asks

DO YOU HAVE THEORETICAL PROFILE TO PLOT, Y OR N?

If the operator wishes, he can enter predicted temperatures for the thermocouple locations used in the plot. A cubic spline fit to these points is made and the resulting curve drawn.

At this point a flashing cursor appears in the lower left-hand corner of the CRT. The operator can move the cursor and provide additional lettering in a manner identical to that described for the program WATERQ.

Additional copies can also be made on the internal line printer for plots generated on the CRT. After the required copies have been made, the program returns to line 490 to ask if more plots are to be made.

Isotherm Plotting

If the operator had chosen to plot isotherms instead of thermal profile plots, the program would have jumped to the label Isotherm at location 1760. The operator would then select a particular scan number as indexed in the chained data array in the computer memory. The chained data array selects a particular time for which isotherms are to be plotted. Lines 1800 to 2080 initialize variables pertaining to the thermocouple placement in the Water Migration experiment. The heater hole and the thermocouple holes TH-1 and TH-2 are used in this consideration. Hole radii and thermocouple depths and channel numbers are established. All this is done automatically and without operator input.

The prompt

DO YOU WANT THE TEMPERATURES PRINTED ON THE PLOT, Y OR N?

allows the operator to determine whether or not actual recorded temperatures (°C) are to be printed on the plot at the appropriate thermocouple locations; otherwise only tick marks will indicate actual thermocouple locations.

The operator is then asked whether or not the isotherms are to be drawn. This choice allows only the thermocouple locations and actual thermocouple data to be indicated if desired.

If isotherm drawing is wanted, a least-squares fitting routine is entered that obtains fit parameters for interpolation among data as a function of radius for three depths corresponding to heater midplane, heater shallow end, and heater deep end. This is necessary in case isotherm temperatures higher than any measured temperatures in either TH-1 or TH-2 are requested.

Unless the operator wants to enter his own temperatures for isotherms, the program will default to five curves at 60°, 70°, 80°, 90°, and 100°. Operator-requested isotherms must be greater than 18° to 20° since these are ambient conditons. However, it is suggested that values above 25° to 30° be used to avoid poor interpolations and fitting caused by small fluctuations in ambient readings. If the operator wants to enter his own isotherm temperatures, he is requested to do so, beginning at the smallest value (lowest selected temperature) and continuing by increasing values to the highest temperature. The temperature values are then printed on the plot, and after required fitting routines have been used the isotherms are plotted.

After the isotherms have been drawn, additional copies of CRT plots can be made. The program returns to line 490 and asks whether additional plots from the data currently in memory are desired. If no further plots from the data are wanted, the operator is asked whether or not more data should be entered from tape or disk to be plotted. If so, the program returns to line 120 to start over for new data. If no more data or plots are desired, the program terminates.

Flow Permeability Test Data Analysis

A shortened version of the TUFFF program, called PERMB, was used to acquire data during in situ fluid flow tests. These tests were conducted in

holes in the heater array after all instrumentation had been removed. There posttest data were compared with pretest data taken before the 3052A DAS was available for field experiments. PERMB is the same as TUFFF, except that it is shortened to accomodate only those transducer inputs necessary for the fluid flow tests. It will, therefore, not be discussed in detail.

To shorten TUFFF and to make it run faster so that the scan rate could be increased, subroutines Scan and Measure were shortened by eliminating all unnecessary references to DVMs and Scanners. Program segments dealing with unused transducers were deleted. Subroutines supplying thermocouple conversions and all plot routines were also deleted. Scan intervals of 5 s were used for most of the permeability tests. However, scan intervals of less than 1 s were possible. The scan interval for the field experiment with all 120 channels was 5 min.

The scanning capability was changed from 20 scans of 120 channels each to 1000 scans of 5 channels each. Only fluid pressures and flow rates were monitored and stored as a function of time.

The fluid flow data can be plotted by using a program called PERMP which is very similar to the larger program WATERQ. However, since fluid flow data were stored in files of 1000 scans each, no Monster file chaining capability was provided. In most cases 1000 scans are enough to handle all of the data from a particular injection and shut-in pressure test. Thus all the data for a particular injection pressure/flow and subsequent shut-in pressure test were already contained in one file and could be easily plotted.

PERMP accomodates autoscaling, left- and right-hand scaling and labeling, plot speed variation, operator lettering, theoretical curve, and multiple copy options as found in WATERQ.

The fluid flow data analysis is enhanced by the capability of producing semilog plots of pressure vs log time or inverse flow vs log time. This capability provides useful plots for graphic analysis necessary in some formalisms to determine permeabilities.

Pressure vs log time plots were made by using the program SEMLGC (Appendix F). Inverse flow vs time was plotted by using a companion program 1/FLOW, which because of its similarity with SEMLGC, will not be discussed separately.

SEMLGC reads in data in a manner identical to PERMP. However, no autoscaling option is provided. A choice of the number of cycles for semilog plotting is required:

HOW MANY CYCLES IS LOG PLOT?

The minimum and maximum pressures for the Y-AXIS bounds are also required. Once the beginning and final scan numbers have been entered, the program plots pressures vs log time with the time referenced to the beginning scan specified earlier. A graph of such data using SEMLGC is given in Figure 6.

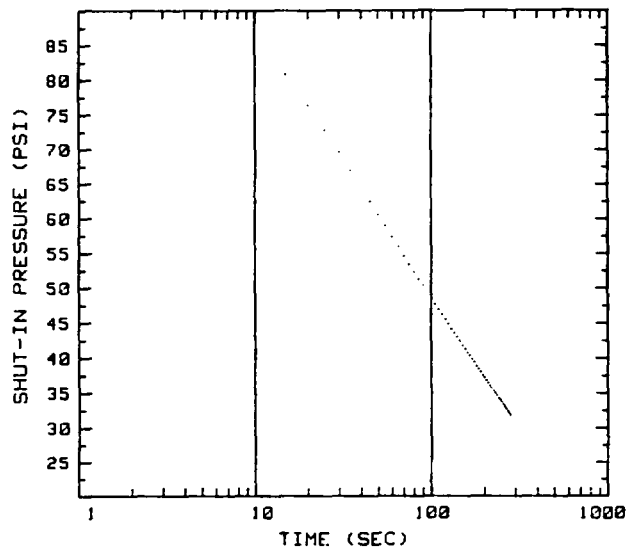


Figure 6. WM-1 Shut-In Pressures for Posttest In Situ Permeability Tests

Additional Software

Other programs almost identical to PERMB and PERMP were used for posstest pressure and water level calibration tests both in situ and in the laboratory. Plotting routines for these tests and other transducer calibration data were also written in support of the Water Migration Test.

Conclusions

The HP3052A Data Acquisition System was successfully programmed and configured to control heater experiment functions, accumulate and store data, print and plot data as a function of time, pace other calibration and in situ experiments, and provide a variety of plot types.

The smart controller capability provided the operators of the in situ experiments with the flexibility to change scan rates; monitor safety related functions; convert analog signals to digitized data in engineering units; and monitor through displays, printed lists, and plots the course of the experiment. Rapid turn-around data plotting and analysis were available because of data format and medium compatabilities and the interactive nature of the calculator operating system.

The HP9845T and associated software and hardware thus fulfilled the design requirements for the Water Migration Experiment. Copies of the programs described in this report and other programs used in support of this activity are available from the authors on request.

References

[1]J. Keith Johnstone, *In-Situ Tuff Water Migration/Heater Experiment: Experimental Plan*, SAND 79-1276 (Albuquerque, NM: Sandia National Laboratories, August 1980).

[2]J. Keith Johnstone, *In-Situ Tuff Water Migration/Heater Experiment:Final Report*, SAND 81-1918 (Albuquerque, NM: Sandia National Laboratories, to be published).

[3]D. R. Waymire and C. O. Duimstra, *In-Situ Tuff Water Migration/Heater Experiment: Instrumentation Design and Fielding*, SAND 81-1058 (Albuquerque, NM: Sandia National Laboratories, to be published).

APPENDIX A

TUFFF Program

```
10      REM   *****   TUFF PROGRAM TO RECIEVE/STORE AND PLOT DATA  2/25/80 ****
20      REM   *****   REVISION F  ********
30      REM  ***********  INITIALIZATION
40      OPTION BASE 1
50      COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
60      COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
70      COM Data(120),Ref,Zero
80      COM Tbegin$[8],Int$,Bumps,D(10)
90      COM Holes,Number(3),Temp(3,13)
100     COM Depth(3,13),M,Mod(3),Domain(20),Func(20),Ch(3)
110     CALL Init(#1,Scan1)
120     ON ERROR CALL Eproc
130     CALL Comments
140     PRINT "PUT INITIALIZED BLANK CASSETTE(RECORD MODE) IN T14 TAPE DRIVE, P
RESS CONTINUE"
150     PAUSE
160     C$="DATA1 :T14"
170     Scans=1
180     CREATE C$,1,10000
190     PRINT C$
200     PRINT "INSERT INITIALIZED BLANK FLOPPY INTO MASTER DRIVE, PRESS CONTINU
E"
210     PAUSE
220     E$="DATA1 :F8,0"
230     CREATE E$,1,10000
240     PRINT E$
250     INPUT "ARE YOU USING SLAVE FLOPPY FOR BACKUP FILES, Y OR N",B1$
260     IF UPC$(B1$[1,1])="N" THEN 320
270     PRINT "INSERT INITIALIZED BLANK FLOPPY INTO SLAVE DRIVE, PRESS CONTINUE
"
280     PAUSE
290     G$="DATA1 :F8,1"
300     CREATE G$,1,10000
310     PRINT G$
320     REM ---KEY DEFINITIONS
330     ON KEY #0 GOSUB 660     ! CHANGE SCAN RATE
340     ON KEY #1 GOSUB 850     ! FORCE A SCAN
350     ON KEY #3 GOSUB 880     ! PLOT LOCATION DATA
360     ON KEY #5 GOSUB 920     ! DISPLAY TIME
370     ON KEY #7 GOSUB 980     ! PLOT RADIUS DATA
380     ON KEY #15 GOTO 1000    ! STOP RUN
390     CALL Setclock     ! THIS IS WHERE I UPDATE OR SET THE CLOCK TIME ********
400     INPUT "ENTER THE DESIRED SCAN TIME IN MINUTES",Time
410     PRINT "SCAN TIME = ";Time;" MINUTES"
420     Int$=VAL$(60000*Time)
430     INPUT "ENTER THE DESIRED START TIME TO BEGIN SCANNING, HR:MIN:SEC",Tbegin
$
440     PRINT "SCAN BEGIN = ";Tbegin$
450     Beep=0
460     Bumps=0
470     CALL Match
```

```
480     CALL Realtime
490     IF Beep>0 THEN 450
500     ON INT #Tmr GOTO Begin
510     IF Bumps>0 THEN Begin
520     CONTROL MASK Tmr;192
530     CARD ENABLE Tmr
540     GOTO 480
550 Begin: CALL Timer
560     U1=01
570     CALL Monitor
580     REM ---------------- MAIN PROGRAM LOOP -------------
581     ON INT #Tmr CALL Monitor
582     CONTROL MASK Tmr;192
583     CARD ENABLE Tmr
590     CALL Realtime
600     IF Beep>0 THEN 450
630     IF Scans>20 THEN 1020
650     GOTO 590
660     REM ----------------KEY#0 PROCESSOR
670     BEEP
680     OFF INT #Tmr
690     OUTPUT Tmr;"A"
700     PRINT "CURRENT SCAN RATE = ";Int$;" MILLISECONDS"
710     INPUT "DO YOU WANT TO CHANGE SCAN RATE, Y OR N?",A3$
720     IF UPC$(A3$[1,1])<>"Y" THEN Return
730     INPUT "ENTER THE NEW SCAN INTERVAL (IN MINUTES)",Time
740     PRINT "NEW SCAN INTERVAL =";Time;" MINUTES"
750     Int$=VAL$(60*Time*1000)
760     Beep=Beep+1
770     INPUT "WHEN SHOULD NEW SCAN RATE BEGIN (HR:MIN:SEC)",Tbegin$
780     PRINT "NEW SCAN WILL BEGIN AT ";Tbegin$
790 Return:   INPUT "DO YOU WISH TO CHANGE YOUR MIND (Y OR N)",A3$
800     IF UPC$(A3$[1,1])="Y" THEN 730
810 Start: U1=01
820     CONTROL MASK Tmr;192
830     CARD ENABLE Tmr
840     RETURN
850     REM ------------ KEY #1 PROCESSOR
860     CALL Monitor
870     RETURN
880     REM ---------- KEY#3 PROCESSOR ******* PLOT LOCATION DATA *****
890     CALL Tplot
900     RETURN
910     REM ---------- KEY #5 PROCESSOR ****** TO DISPLAY TIME *******
920     OUTPUT Tmr;"R"
930     ENTER Tmr USING "T";Tyme$
940     DISP "CURRENT TIME IS ";Tyme$
950     WAIT 800
960     RETURN
970     REM ----------- KEY #7 PROCESSOR ****** PLOT RADIUS DATA ********
980     CALL Rplot
990     RETURN
1000    REM ---------KEY#15 PROCESSOR
1010    Dump=1
1020    IF Scans>20 THEN Scans=20
1030    OFF INT #Tmr
1040    ASSIGN #1 TO C$
1050    BEEP
1060    PRINT #1;Table$(*),Table2(*),Scans
1070    PRINT #1;END
1080    ASSIGN #1 TO *
1090    PRINTER IS 0
1100    PRINT "DUMPED DATA ON :T14"
1110    ASSIGN #2 TO E$
1120    PRINT #2;Table$(*),Table2(*),Scans
1130    PRINT #2;END
```

```
1140   ASSIGN #2 TO *
1150   IF UPC$(B1$[1,1])="N" THEN 1200
1160   ASSIGN #3 TO G$
1170   PRINT #3;Table$(*),Table2(*),Scans
1180   PRINT #3;END
1190   ASSIGN #3 TO *
1200   IF Dump=1 THEN 1420
1210   Scans=1
1220   IF VAL(C$[5,6])>=20 THEN 1390
1230   C$[5,6]=VAL$(VAL(C$[5,6])+1)
1240   CREATE C$,1,10000
1250   PRINTER IS 0
1260   PRINT C$
1270   E$[5,6]=VAL$(VAL(E$[5,6])+1)
1280   CREATE E$,1,10000
1290   PRINT E$
1300   IF UPC$(B1$[1,1])="N" THEN 1340
1310   G$[5,6]=VAL$(VAL(G$[5,6])+1)
1320   CREATE G$,1,10000
1330   PRINT G$
1340   PRINTER IS 16
1350   ON INT #Tmr CALL Monitor
1360   CONTROL MASK Tmr;192
1370   CARD ENABLE Tmr
1380   GOTO 580
1390   PRINT "I AM AT THE END OF THE TAPE AND FLOPPY, PUT IN NEW ONES AND
PRESS CONTINUE"
1400   PAUSE
1410   GOTO 140
1420   PRINT "I AM ENDING THE PROGRAM NOW !!!!!"
1430   END
1440   REM  ************************************************ Monitor  ******
1450   SUB Monitor
1460   OPTION BASE 1
1470   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
1480   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
1490   COM Data(120),Ref,Zero
1500   COM Tbegin$[8],Int$,Bumps,D(10)
1510   Bumps=Bumps+1
1520   CALL Scan(Fstore)
1530   ASSIGN #2 TO E$
1540   PRINT #2;Table$(*),Table2(*),Scans
1550   IF UPC$(B1$[1,1])="N" THEN 1580
1560   ASSIGN #3 TO G$
1570   PRINT #3;Table$(*),Table2(*),Scans
1580   PRINTER IS 16
1590   ON Scans GOTO 1600,1640,1640,1600,1640,1640,1600,1640,1640,1600,1640,1640
,1600,1640,1640,1600,1640,1640,1600,1640
1600   PRINTER IS 0
1610 ! IF Scans=1 THEN PRINT PAGE
1620 ! IF Scans=10 THEN PRINT PAGE
1630 ! IF Scans=19 THEN PRINT PAGE
1640   PRINT LIN(4)
1650   PRINT Time$;"    Scans =";Scans;"   Reference Junction = ";Zero;" Deg C",
"File Name is ";E$
1660   PRINT LIN(1)
1670   PRINT USING "6X,10(5X,ZZ)";1,2,3,4,5,6,7,8,9,10
1680   FIXED 2
1690   FOR I=1 TO 115 STEP 10
1700    D(1)=Table2(Scans,I)
1710    IF (D(1)>999) OR (D(1)<-999) THEN D(1)=999
1720    D(2)=Table2(Scans,I+1)
1730    IF (D(2)>999) OR (D(2)<-999) THEN D(2)=999
1740    D(3)=Table2(Scans,I+2)
```

```
1750    IF (D(3)>999) OR (D(3)<-999) THEN D(3)=999
1760    D(4)=Table2(Scans,I+3)
1770    IF (D(4)>999) OR (D(4)<-999) THEN D(4)=999
1780    D(5)=Table2(Scans,I+4)
1790    IF (D(5)>999) OR (D(5)<-999) THEN D(5)=999
1800    D(6)=Table2(Scans,I+5)
1810    IF (D(6)>999) OR (D(6)<-999) THEN D(6)=999
1820    D(7)=Table2(Scans,I+6)
1830    IF (D(7)>999) OR (D(7)<-999) THEN D(7)=999
1840    D(8)=Table2(Scans,I+7)
1850    IF (D(8)>999) OR (D(8)<-999) THEN D(8)=999
1860    D(9)=Table2(Scans,I+8)
1870    IF (D(9)>999) OR (D(9)<-999) THEN D(9)=999
1880    D(10)=Table2(Scans,I+9)
1890    IF (D(10)>999) OR (D(10)<-999) THEN D(10)=999
1900    PRINT USING "3X,ZZZ,2X,10(MDDD.DD)";I-1,D(1),D(2),D(3),D(4),D(5),D(6),D(
7),D(8),D(9),D(10)
1910   NEXT I
1920   PRINTER IS 0
1921   IF Data(108)>.71 THEN Pwr=1
1930   IF Pwr=1 THEN CALL Chan(121)
1940   IF Data(108)>.71 THEN PRINT LIN(2),"WATER SAMPLE BEGUN AT ";Time$;"   WAT
ER SENSOR AT ";Data(108);" VOLTS"
1941   IF Data(108)<.6 THEN CALL Chan(128)
1942   IF (Data(108)<.6) AND (Pwr=1) THEN PRINT LIN(1),"WATER SAMPLING ENDED AT
";Time$;"   WATER SENSOR AT ";Data(108);" VOLTS"
1943   IF Data(108)<.6 THEN Pwr=0
1950   IF (Data(83)>100) OR (Data(81)>100) THEN CALL Chan(122)
1960   IF (Data(83)>100) OR (Data(81)>100) THEN WAIT 1000
1970   IF (Data(11)>330) AND (Data(13)>330) THEN CALL Chan(123)
1980   IF (Data(11)>330) AND (Data(13)>330) THEN WAIT 5000
1990   IF (Data(11)>400) AND (Data(13)>400) THEN CALL Chan(124)
2000   IF (Data(11)>400) AND (Data(13)>400) THEN WAIT 5000
2001   PRINTER IS 16
2002   CONTROL MASK Tmr;192
2010   Scans=Scans+1
2011   CARD ENABLE Tmr
2020   SUBEND
2030   REM ************************************************************ Eproc **
2040   SUB Eproc
2050   OPTION BASE 1
2060   PRINTER IS 16
2070   PRINT ERRM$
2080   SUBEND
2090   REM ***************************************************  Realtime   ****
2100   SUB Realtime
2110   OPTION BASE 1
2120   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
2130   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
2140   OUTPUT Tmr;"R"
2150   ENTER Tmr USING "T";Tyme$
2160   Time$=Tyme$
2170   DISP Time$
2180   CARD ENABLE Tmr
2190   SUBEND
2200   REM ************************************************************ Scan **
2210   SUB Scan(Fstore)
2220   OPTION BASE 1
2230   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
2240   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
2250   COM Data(120),Ref
2260   CALL Realtime
```

```
2270   Table$(Scans)=Time$
2280   CALL Dvm(1,7,0,0,1)                    ! THERMOCOUPLE VOLTAGE
2290   CALL Measure(Tc1,Tc2,1)
2300   CALL Measure(Tf1,Tf2,2)
2310   CALL Dvm(1,7,0,0,1)                    ! VOLTAGE
2320   CALL Measure(Pr1,Pr2,3)
2330   CALL Measure(Sm1,Sm2,4)
2340   CALL Dvm(5,7,0,0,1)                    ! 4-WIRE OHMS
2350   CALL Measure(Th1,Th2,5)
2360   CALL Dvm(1,7,0,0,1)                    ! VOLTAGE
2370   CALL Measure(Ph1,Ph2,6)
2380   CALL Dvm(1,7,0,0,1)
2390   CALL Measure(Rh1,Rh2,7)
2400   RESET 4
2410   ENTER 4;P$,Q$
2420   Data(118)=VAL(P$[7,12])
2430   RESET 3
2440   ENTER 3;P$,Q$
2450   Data(119)=VAL(P$[7,12])
2460   Table2(Scans,118)=Data(118)
2470   PRINT "DATA CH # 118 IS ";Data(118)
2480   Table2(Scans,119)=Data(119)
2490   PRINT "DATA CH # 119 IS ";Data(119)
2500   Data(120)=Data(118)+Data(119)
2510   Table2(Scans,120)=Data(120)
2520   PRINT "DATA CH # 120 IS ";Data(120)
2530   CALL Dvm(1,7,0,0,1)
2540   CALL Measure(V1,W2,8)
2550   CALL Dvm(1,7,0,0,1)
2560   CALL Measure(Wat1,Wat2,9)
2570   CALL Dvm(1,7,0,0,1)
2580   CALL Measure(Las1,Las2,10)
2590   CALL Measure(Las3,Las4,11)
2600   SUBEND
2610   REM ******************************************************** Measure ***
2620   SUB Measure(INTEGER First,Last,Funct)
2630   OPTION BASE 1
2640   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
2650   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
2660   COM Data(120),Ref,Zero
2670   IF Last=0 THEN SUBEXIT
2680   FOR Channel=First TO Last
2690    Ch=Channel
2700    IF Channel=0 THEN Go_on            !--IF REF.CH. THEN SKIP
2710    IF Channel=20 THEN Go_on
2720    IF Channel=40 THEN Go_on
2730    IF Channel=60 THEN Go_on
2740     Offset=Channel DIV 60            !--FIND IF 2ND SCANNER
2750     CALL Chan(Channel)
2760     Reading=FNRdvm                   !--STORE DATA TEMPORARILY
2770     ON Funct GOSUB Tc,Tc,Pr,Sm,Th,Ph,Rh,Pwr,Wat,Las,Xtra !CONVERSION
2780     RESET Scn+Offset            !--RESET SCANNER, 1ST OR 2ND
2790   PRINT "DATA CH # ";Ch;" IS ";Data(Ch)
2800     Table2(Scans,Ch)=Data(Ch)
2810   NEXT Channel
2820   SUBEXIT
2830 Skip: Data(Channel)=-8E60                    !--UNUSED CHANNEL
2840   Table2(Scans,Channel)=Data(Channel)
2850   GOTO 2810
2860 Go_on: IF Channel=0 THEN Card=1
2870    IF Channel=20 THEN Card=2
2880    IF Channel=40 THEN Card=3
2890    IF Channel=60 THEN Card=4
2900    CALL Reference(Junc,Card)
```

```
2910    Ref=Junc
2920    IF Channel=0 THEN 2950
2930    Data(Ch)=Ref
2940    GOTO 2780
2950    Zero=Ref
2960    RESET Scn+Offset
2970    PRINT "REFERENCE JUNCTION IS ";Zero
2980    GOTO 2810
2990   REM ------------- Conversions
3000 Tc:  Xtime=Ch DIV 20*20
3010   Data(Ch)=FNTemp(Ch,Reading,Ref,Tc2)
3020   RETURN
3030 Ph:  Data(Ch)=(Reading+.00706)*1000
3040   RETURN
3050 Rh:  Data(Ch)=Reading*100
3060   RETURN
3070 Pr:  Press=Ch-79 ! DETERMINE WHICH PRESSURE GAUGE IS USED
3080   ON Press GOTO 3090,3120,3150,3180,3210,3240,3270,3300,3330
3090   Data(Ch)=(Reading-.703)/.0427
3100   IF Data(Ch)>15 THEN 3350
3110   RETURN
3120   Data(Ch)=(Reading-.025)/.048
3130   IF Data(Ch)>15 THEN 3350
3140   RETURN
3150   Data(Ch)=(Reading-.630)/.043
3160   IF Data(Ch)>15 THEN 3350
3170   RETURN
3180   Data(Ch)=(Reading+.048)/.0482
3190   IF Data(Ch)>15 THEN 3350
3200   RETURN
3210   Data(Ch)=(Reading-.678)/.0433
3220   IF Data(Ch)>15 THEN 3350
3230   RETURN
3240   Data(Ch)=(Reading+.066)/.0481
3250   IF Data(Ch)>15 THEN 3350
3260   RETURN
3270   Data(Ch)=(Reading-.636)/.043
3280   IF Data(Ch)>15 THEN 3350
3290   RETURN
3300   Data(Ch)=(Reading+.017)/.049
3310   IF Data(Ch)>15 THEN 3350
3320   RETURN
3330   Data(Ch)=(Reading-.005)/.0476
3340   RETURN
3350   FOR I=1 TO 10
3360    BEEP
3370    WAIT 700
3380   NEXT I
3390   PRINTER IS 16
3400   PRINT "*************************************"
3410   PRINT "*****  PRESSURE AT DANGER LEVEL  *****"
3420   PRINT "*************************************"
3430   PRINTER IS 0
3440   PRINT LIN(3),"******** NOTE: POSSIBLE DANGEROUS PRESSURES ! ! !  *****",L
IN(1)
3450   PRINTER IS 16
3460   RETURN
3470 Sm:  Data(Ch)=Reading
3480   RETURN
3490 Wat: Data(Ch)=Reading
3500   RETURN
3510 Las: ! MUST DEFINE CONVERSION FOR LASER STRAINMETER
3520   Data(Ch)=(Reading+10)*4096*.6328/160          !(VOLTS)*(CTS/20VOLTS)*(MIC
RON/WAVE)/((20)*(8CTS/WAVE))
3530   RETURN
3540 Th:  Num=3976.74*298  ! Num=BETA*T0
```

```
3550   Den=298*LOG(Reading/4978.52)+3976.74  ! Den=T0*LOG(R/R0)+BETA
3560   Data(Ch)=Num/Den-273  ! T[C]=T[K]-273
3570   RETURN
3580 Pwr: ! Factor NEEDS TO BE DEFINED FOR EACH READING TYPE
3590   IF Channel=V1 THEN Data(Ch)=Reading*29.8
3600   IF Channel=V2 THEN Data(Ch)=Reading*30
3610   IF Channel=A1 THEN Data(Ch)=Reading*1.99
3620   IF Channel=A2 THEN Data(Ch)=Reading*2
3630   IF Channel=W1 THEN Data(Ch)=Reading*310
3640   IF Channel=W2 THEN Data(Ch)=Reading*301
3650   RETURN
3660 Xtra:  ! NEW LASER FRINGE COUNTER INSTALLED 2/13/80
3670   Data(Ch)=(Reading+5)*4096*.6328/80  !(VOLTS)*(CTS/10VOLTS)*(MICRON/WAVE)/
((10*(8CTS/WAVE))
3680   RETURN
3690   SUBEND
3700   REM ************************************************************ Init **
3710   SUB Init(#1,Scan1)
3720   OPTION BASE 1
3730   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
3740   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
3750   COM Data(120),Ref,Zero
3760   COM Tbegin$[8],Int$
3770   Tbegin$="12:00:00"
3780   PRINT "INITIALIZATION"
3790   READ Tmr,Scn,Scn1,Dvm,Bus                  !--SET UP ADDRESSES
3800   DATA 5,709,710,722,7
3810   ABORTIO Bus                                !--RESET DEVICES ON BUS
3820   REMOTE Bus                                   !--BUS IN REMOTE MODE
3830   RESET Bus                                  !--RESET BUS
3840   LOCAL LOCKOUT 7
3850   READ IO Tmr,5;Sig                     !--IF CLOCK IS PRESENT, THEN START
3860     IF BINAND(Sig,48)=32 THEN OUTPUT Tmr USING "A";"B"
3870   Mistake=0
3880   OUTPUT Tmr;"U1=01"
3890   READ Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,Pr1,Pr2,Sm1,Sm2,Wat1,Wat2,Las1,Las2,
W1,W2,V1,V2,A1,A2,Th1,Th2,Las3,Las4,Time
3900   DATA 0,79,0,0,93,96,97,101,80,88,89,91,108,111,112,113,104,107,102,105,10
3,106,114,117,92,92,300000
3910   Int$=VAL$(Time)
3920   PRINT "PROGRAM RUNNING"
3930   SUBEND
3940   REM ************************************************************ Dvm **
3950   SUB Dvm(Function,Range,High_res,Data_rqs,Auto_cal)
3960   OPTION BASE 1
3970   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
3980   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
3990   Error=((Function<1) OR (Function>6))+2*((Range<1) OR (Range>7))
4000     IF Error THEN CALL Problem("Dvm",Error)
4010    RESET Dvm
4020    OUTPUT Dvm USING Fmt;Function,Range,High_res,Data_rqs,Auto_cal
4030 Fmt: IMAGE "E",D,"R",D,"H",D,"D",D,"T3","A",D
4040   SUBEND
4050   REM ************************************************************ Chan **
4060   SUB Chan(Channel)
4070   OPTION BASE 1
4080   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
4090   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
4100   Offset=ABS(Channel) DIV 80
4110   Error=(Scn MOD 100+Offset>30)+2*(Channel>239)
```

```
4120   IF Error THEN CALL Problem("Chan",Error)
4130      IF Channel>=0 THEN OUTPUT Scn+Offset USING "ZZ";INT(Channel-80*Offset)
4140     SUBEND
4150   REM ******************************************************** Problem **
4160   SUB Problem(Where$,Error)
4170   PRINTER IS 0
4180   PRINT LIN(1);"ERROR TYPE ";Error;"IN ";Where$;" SUBROUTINE";LIN(1);ERRM$
4190   PRINTER IS 16
4200   SUBEND
4210   REM ********************************************************** Timer **
4220   SUB Timer
4230   OPTION BASE 1
4240   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
4250   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
4260   COM Data(120),Ref,Zero
4270   COM Tbegin$[8],Int$
4280   OUTPUT Tmr;"A"
4290   U1=01
4300   OUTPUT Tmr;"B U1P",Int$
4310   OUTPUT Tmr;"U1G"
4320   SUBEND
4330   REM ********************************************************** Rdvm **
4340   DEF FNRdvm
4350   OPTION BASE 1
4360   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
4370   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
4380   TRIGGER Dvm
4390   ENTER Dvm USING "F";Reading
4400   RETURN Reading
4410   FNEND
4420   REM ******************************************************** Refjct **
4430   DEF FNRefjct(Resistance)
4440   PRINT Resistance
4450   RETURN 5041.6/(LOG(Resistance)+7.15)-314.052    !YSI THERMISTOR
4460   FNEND
4470   REM ********************************************************** Temp **
4480   DEF FNTemp(Channel,Voltage,Junct,INTEGER T1)
4490   DIM Junct(3),Coeff(9)
4500   IF Voltage>.028863 THEN A
4510   Junct(1)=5.8637565E1
4520   Junct(2)=4.670025E-2
4530   Junct(3)=-1.4438022E-5
4540   Coeff(1)=1.7022525E-2          ! NBS MONOGRAM 125 QUADRATIC FIT 0-440 DEG
4550   Coeff(2)=-2.2097240E-7
4560   Coeff(3)=5.4809314E-12
4570   Coeff(4)=-5.7669892E-17
4580   Temp=FNPoly1(Coeff(*),FNPoly2(Junct(*),Junct)+Voltage*1E6)
4590   GOTO Retn
4600 A:  Junct(1)=5.8637565E1
4610     Junct(2)=4.670025E-2
4620     Junct(3)=-1.4438022E-5
4630     Coeff(1)=1.6410783E-2      ! NBS MONOGRAM 125 QUADRATIC FIT 0-1000 DEG
4640     Coeff(2)=-1.3560189E-7
4650     Coeff(3)=1.8600342E-12
4660     Coeff(4)=-8.5537337E-18
4670     Temp=FNPoly1(Coeff(*),FNPoly2(Junct(*),Junct)+Voltage*1E6)
4680 Retn:  RETURN Temp
4690   REM ******************************************************** Poly1 **
4700   DEF FNPoly1(Co(*),Val)=(((Co(4)*Val+Co(3))*Val+Co(2))*Val+Co(1))*Val
4710   REM ******************************************************** Poly2 **
4720   DEF FNPoly2(Co(*),Value)=((Co(3)*Value+Co(2))*Value+Co(1))*Value
4730   REM ******************************************************* Comments **
```

```
4740   SUB Comments
4750   PRINTER IS 0
4760   PRINT LIN(5)
4770   PRINT "     THIS IS A PROGRAM TO RECORD, MANIPULATE, AND STORE DATA TAKEN
 FOR TUFF HEATER EXPERIMENT.";LIN(1)
4780   PRINT " CERTAIN PROCEDURES MUST BE FOLLOWED TO CHANGE ANY OF THESE ASSIGN
MENTS.";LIN(1)
4790   PRINT "     THE FOLLOWING IS A LIST OF THE FIRST/LAST CHANNEL ASSIGNMENTS
 AND THEIR"
4800   PRINT "CORRESPONDING VARIABLE NAMES.";LIN(1)
4810   DIM G$[55]
4820   DATA FIRST CHANNEL OF TYPE E THERMOCOUPLE,0,LAST CHANNEL OF TYPE E THERMO
COUPLE,79
4830   DATA FIRST CHANNEL OF SECOND SET TYPE E THERMOCOUPLE,0,LAST CHANNEL OF TY
PE T THERMOCOUPLE,0
4840   DATA FIRST CHANNEL OF PH GAUGE,93,LAST CHANNEL OF PH GAUGE,96
4850   DATA FIRST CHANNEL OF REL HUMID GAUGE,97,LAST CHANNEL OF REL HUMID GAUGE,
101
4860   DATA FIRST CHANNEL OF PRESSURE GAUGE,80,LAST CHANNEL OF PRESSURE GAUGE,88
4870   DATA FIRST CHANNEL OF STRESS METER GAUGE,89,LAST CHANNEL OF STRESS METER
GAUGE,91
4880   DATA FIRST CHANNEL OF WATER LEVEL GAUGE,108,LAST CHANNEL OF WATER LEVEL G
AUGE,111
4890   DATA CHANNEL FOR FIRST LASER STRAINMETER READOUT,92,CHANNEL FOR SECOND LA
SER STRAINMETER READOUT,112
4900   DATA CHANNEL OF WATTAGE READING-W1,104,CHANNEL OF VOLTAGE READING-V1,102,
CHANNEL OF AMPERAGE READING-A1,103
4910   DATA CHANNEL OF WATTAGE READING-W2,107,CHANNEL OF VOLTAGE READING-V2,105,
CHANNEL OF AMPERAGE READING-A2,106
4920   DATA FIRST CHANNEL OF THERMISTOR GAGE #1,114,LAST CHANNEL OF THERMISTOR G
AGE #2,117
4930   FIXED 0
4940   FOR I=1 TO 22
4950     READ G$
4960     READ Datum
4970     PRINT TAB(10);G$;TAB(65);Datum
4980     NEXT I
4990   PRINT LIN(2);"TO CHANGE ANY OF THE SCAN TIMES, DEPRESS S.F.KEY 0.";LIN(1)
5000   PRINT "TO FORCE A SCAN, DEPRESS S.F.KEY 1";LIN(1)
5010   PRINT "TO USE THE TEMP PLOTTING ROUTINE, DEPRESS S.F.KEY 3.";LIN(1)
5020   PRINT "TO DISPLAY THE TIME, DEPRESS S.F.KEY 5.";LIN(1)
5030   PRINT "TO USE THE TEMP VS RADIUS PLOT ROUTINE, DEPRESS S.F.KEY 7.";LIN(1)
5040   PRINT "TO STOP THE PROGRAM, DEPRESS S.F.KEY 15."
5050   PRINTER IS 16
5060   SUBEND
5070   REM ************************************************ Setclock ********
5080   SUB Setclock
5090   OPTION BASE 1
5100   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
5110   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
5120   OUTPUT Tmr;"R"
5130   ENTER Tmr USING "T";Time$
5140   PRINT "CURRENT TIME IS ";Time$
5150   INPUT "DO YOU WANT TO RESET THE CLOCK, Y OR N?",Rs$
5160   IF UPC$(Rs$[1,1])<>"Y" THEN SUBEXIT
5170   U1=01
5180   OUTPUT Tmr;"A"
5190   INPUT "WHAT IS CURRENT TIME MO:DA:HR:MN:SC",R$
5200   OUTPUT Tmr;"S",R$
5210   OUTPUT Tmr;"U1G"
5220   SUBEND
5230   REM ****************************************  Reference  ********
5240   SUB Reference(Ref,Card)
5250   OPTION BASE 1
```

```
5260   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
5270   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
5280   CALL Dvm(5,4,1,0,1)
5290   IF Card>1 THEN 5320
5300   CALL Chan(0)
5310   GOTO 5390
5320   IF Card>2 THEN 5350
5330   CALL Chan(20)
5340   GOTO 5390
5350   IF Card>3 THEN 5380
5360   CALL Chan(40)
5370   GOTO 5390
5380   CALL Chan(60)
5390   Ref=FNRefjct(FNRdvm)
5400   CALL Dvm(1,1,0,0,1)
5410   SUBEND
5420   REM  ********************************************  Tplot  ********'
5430 SUB Tplot
5440 OPTION BASE 1
5450 COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEGER
 Scans
5460 COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,Pr1
,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
5470 COM Data(120),Ref,Zero
5480 COM Tbegin$[8],Int$,Bumps,D(10)
5490 COM Holes,Number(3),Temp(3,13)
5500 COM Depth(3,13),M
5510 Holes=3
5520 Number(1)=5
5530 Number(2)=13
5540 Number(3)=9
5550 Radius(1)=.0635
5560 Radius(2)=.406 ! WHATEVER RADIUS TO FIRST HOLE IS ********************
5570 Radius(3)=.610 ! WHATEVER RADIUS TO SECOND HOLE IS *****************
5580 DATA 17.07,18.29,18.75,19.2,21.18
5590 DATA 6,17,23,26,52
5600 DATA 17.22,17.83,18.29,18.44,18.59,18.75,18.90,19.05,19.2,19.35,19.66,19.96
,20.57
5610 DATA 53,54,55,56,57,61,62,63,64,65,66,67,68
5620 DATA 16.31,17.22,18.14,18.44,18.75,19.05,19.35,19.66,21.18
5630 DATA 44,45,46,47,48,49,50,51,52
5640 FOR I=1 TO Holes
5650  FOR J=1 TO Number(I)
5660   READ Depth(I,J)
5670  NEXT J
5680  FOR J=1 TO Number(I)
5690   READ K
5700   Temp(I,J)=Table2(Scans-1,K)
5710  NEXT J
5720 NEXT I
5730 FIXED 2
5740 Maxim: PLOTTER IS 13,"GRAPHICS"
5750 GRAPHICS
5760 PRINTER IS 0
5770 PRINT PAGE,LIN(5)
5780 Nocap: LOCATE 25,115,18,99
5790 SCALE 0,1.2,16,22
5800 AXES .2,1,0,16,5,5,3
5810 AXES .2,1,-1,22,5,5,3
5820 AXES .2,1,1.2,12,5,5,3
5830 CSIZE 2.5
5840 LORG 5
5850 MOVE .03,18.75
5860 LDIR PI/2
```

```
5870 LABEL USING "K";"HEATER"
5880 LDIR 0
5890 MOVE 0,18.18
5900 PLOT .051,18.18,-1
5910 PLOT .051,19.35
5920 PLOT 0,19.35
5930 CSIZE 3
5940 LORG 8
5950 FOR V=16 TO 22
5960 MOVE 0,V
5970 LABEL USING "M2DX";V
5980 NEXT V
5990 MOVE -.1,19
6000 CSIZE 4
6010 LORG 5
6020 LDIR PI/2
6030 LABEL USING "K";"DEPTH(M)"
6040 LDIR 0
6050 LORG 6.
6060 CSIZE 3
6070 FOR H=.2 TO 1 STEP .2
6080  MOVE H,15.9
6090  LABEL USING "MD.D";H
6100 NEXT H
6110 CSIZE 4
6120 MOVE .6,15.65
6130 LABEL USING "K";"RADIUS(M)"
6140 LORG 3
6150 CSIZE 3
6160 MOVE .1,15.2
6170 LABEL USING "K";"G-TUNNEL HEATER EXPERIMENT THERMAL DATA "
6180 MOVE .15,14.9
6190 LABEL USING "K";"FOR ";Table$(Scans-1);" AND SCAN # ";Scans-1;" IN ";C$[1,6
]
6200 Cycle:FIXED 0
6210 CSIZE 2.5
6220 FOR K=1 TO Holes
6230  LINE TYPE 1
6240  MOVE Radius(K),16
6250  DRAW Radius(K),22
6260  FOR I=1 TO Number(K)
6270   MOVE Radius(K),Depth(K,I)
6280   LORG 2
6290   LABEL USING "K,3D.D,K";"-";Temp(K,I);" C"
6300  NEXT I
6310 NEXT K
6320 Dump:PRINTER IS 0
6330 DUMP GRAPHICS
6340 EXIT GRAPHICS
6350 PRINTER IS 16
6360 SUBEND
6370 REM   ***********************************************  Match  ********
6380 SUB Match
6390 OPTION BASE 1
6400 COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEGER
 Scans
6410 COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,Pr1
,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
6420 COM Data(120),Ref,Zero
6430 COM Tbegin$[8],Int$
6440 OUTPUT Tmr;"A"
6450 U1=01
6460 OUTPUT Tmr;"B U1M",Tbegin$
6470 OUTPUT Tmr;"U1G"
6480 SUBEND
6490  REM   ********************************************  Rplot  *******
```

```
6500  SUB Rplot
6510  OPTION BASE 1
6520 COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEGER
 Scans
6530 COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,Pr1
,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
6540 COM Data(120),Ref,Zero
6550 COM Tbegin$[8],Int$,Bumps,D(10)
6560 COM Holes,Number(3),Temp(3,13)
6570 COM Depth(3,13),M,Mod(3),Domain(30),Func(30),Ch(3)
6580  Holes=3
6590  Narg=30
6600  Radius(1)=.0635
6610  Radius(2)=.4064
6620  Radius(3)=.6096
6630  Ch(1)=24
6640  Ch(2)=61
6650  Ch(3)=48
6660 Setup:  PLOTTER IS 13,"GRAPHICS"
6670  GRAPHICS
6680  PRINTER IS 0
6690  PRINT PAGE,LIN(5)
6700  LOCATE 25,115,18,99
6710  SCALE 0,1,0,400
6720  AXES .1,10,0,0,5,5
6730  AXES .1,10,1,400,5,5
6740  CSIZE 3
6750  LORG 8
6760  FOR V=50 TO 400 STEP 50
6770   MOVE 0,V
6780   LABEL USING "M3D,X";V
6790  NEXT V
6800  MOVE -.1,200
6810  CSIZE 4
6820  LABEL USING "K";"T(C)"
6830  LORG 6
6840  CSIZE 3
6850  FOR H=0 TO 1 STEP .2
6860   MOVE H,-4
6870   LABEL USING "MD.D";H
6880  NEXT H
6890  CSIZE 4
6900  MOVE .5,-15
6910  LABEL USING "K";"RADIUS(M)"
6920  LORG 3
6930  CSIZE 3
6940  MOVE .05,-30
6950  LABEL USING "K";"THERMOCOUPLES T-22; T-55; AND T-44"
6960  MOVE .1,-45
6970  LABEL USING "K";"FOR ";Table$(Scans-1);" AND SCAN # ";Scans-1;" IN ";C$[1,
6]
6980  LORG 5
6990  N=1
7000  FOR I=1 TO Holes
7010   CSIZE 2
7020   Ch=Ch(I)
7030   Mod(I)=Data(Ch)
7040   MOVE Radius(I),Mod(I)
7050   LABEL USING "K";"X"
7060  NEXT I
7070  WAIT 2000
7080  CALL Spline1(Narg)
7090  FOR I=1 TO Narg
7100   PLOT Domain(I),Func(I)
7110  NEXT I
7120  DUMP GRAPHICS
```

```
7130  EXIT GRAPHICS
7140  PRINTER IS 0
7150  PRINT PAGE,LIN(2)
7160  PRINTER IS 16
7170  SUBEND
7180  REM ************************************  Spline1 *********
7190  SUB Spline1(Narg)
7200  OPTION BASE 1
7210   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
7220   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
7230   COM Data(120),Ref,Zero
7240   COM Tbegin$[8],Int$,Bumps,D(10)
7250   COM Holes,Number(3),Temp(3,13)
7260   COM Depth(3,13),M,Mod(3),Domain(30),Func(30),Ch(3)
7270  DIM Deriv(Narg)
7280  FOR I=1 TO Narg
7290   Domain(I)=I*(Radius(Holes)-Radius(1))/Narg+Radius(1)
7300  NEXT I
7310  PRINTER IS 0
7320  CALL Spline(Holes,Narg,Radius(*),Mod(*),Domain(*),Func(*),Deriv(*),Int,1E-
6)
7330  SUBEND
7340  REM ************************************  Spline  ****************
7350  SUB Spline(N,Narg,X(*),Y(*),Domain(*),Func(*),Deriv(*),Int,Eps)
7360  !
7370 ! ******************************************************************
7380 ! *** SPLINE FIT FOR FUNCTION VALUES, INTEGRATION AND DIFFERENTIATION.
7390 ! ******************************************************************
7400 !
7410 !
7420 ! *** BAD DATA CHECK.
7430 Baddta=(N<=0) OR (Eps<=0)
7440 IF Baddta=0 THEN 7540
7450 ! *** PRINT ERROR MESSAGE AND PAUSE.
7460 ! *** USER MAY CORRECT DATA AND CONTINUE.
7470 PRINT LIN(2),"ERROR IN SUBPROGRAM Spline."
7480 PRINT "N=";N,"Eps=";Eps,LIN(2)
7490 PAUSE
7500 GOTO 7430
7510 !
7520 !
7530 ! *** BEGIN SUBPROGRAM.
7540 OPTION BASE 1
7550 DIM S(N),G(N-1),Work(N-1)
7560 FOR I=2 TO N-1
7570    Xi=X(I)
7580    Xim1=X(I-1)
7590    Xip1=X(I+1)
7600    Yi=Y(I)
7610    Yim1=Y(I-1)
7620    Yip1=Y(I+1)
7630    X=Xi-Xim1
7640    H=Xip1-Xim1
7650    Work(I)=.5*X/H
7660    T=((Yip1-Yi)/(Xip1-Xi)-(Yi-Yim1)/X)/H
7670    S(I)=2*T
7680    G(I)=3*T
7690 NEXT I
7700 S(1)=S(N)=0
7710 ! *** W IS THE RELAXATION FACTOR FOR SUCCESSIVE OVER-RELAXATION.
7720 W=8-4*SQR(3)
7730 U=0
7740 FOR I=2 TO N-1
7750    T=W*(-S(I)-Work(I)*S(I-1)-(.5-Work(I))*S(I+1)+G(I))
```

```
7760    H=ABS(T)
7770    IF H>U THEN U=H
7780    S(I)=S(I)+T
7790 NEXT I
7800 IF U>=Eps THEN 7730
7810 FOR I=1 TO N-1
7820    G(I)=(S(I+1)-S(I))/(X(I+1)-X(I))
7830 NEXT I
7840 IF Narg=0 THEN 8160
7850 !
7860 !
7870 ! *** CALCULATE FUNCTION VALUES AND DERIVATIVES.
7880 FOR J=1 TO Narg
7890 Corrector:    I=1
7900    T=Domain(J)
7910    IF T>=X(1) THEN 7990
7920    PRINT LIN(2),"ERROR IN SUBPROGRAM Spline."
7930    PRINT "ARGUMENT OUT OF BOUNDS."
7940    PRINT "X(1)=";X(1),"X(N)=";X(N),"Domain(";J;")=";Domain(J),LIN(2)
7950    PAUSE
7960    GOTO 7890
7970    !
7980    !
7990    I=I+1
8000    IF I>N THEN 7920
8010    IF T>X(I) THEN 7990
8020    I=I-1
8030    H=Domain(J)-X(I)
8040    T=Domain(J)-X(I+1)
8050    X=H*T
8060    S=S(I)+H*G(I)
8070    Z=1/6
8080    U=Z*(S(I)+S(I+1)+S)
8090    W=(Y(I+1)-Y(I))/(X(I+1)-X(I))
8100    Func(J)=W*H+Y(I)+X*U
8110    Deriv(J)=W+(H+T)*U+Z*X*G(I)
8120 NEXT J
8130 !
8140 !
8150 ! *** CALCULATE INTEGRAL FROM X(1) TO X(N).
8160 Int=0
8170 FOR I=1 TO N-1
8180    H=X(I+1)-X(I)
8190    Int=Int+.5*H*(Y(I)+Y(I+1))-1/24*H^3*(S(I)+S(I+1))
8200 NEXT I
8210 SUBEND
```

APPENDIX B

AUTSTE Program

```
10      REM   ****  TUFF AUTOSTART TO RECIEVE/STORE AND PLOT DATA  3/10/80 ****
20      REM   *****  REVISION E  ********
30      REM  ***********  INITIALIZATION **********
40      OPTION BASE 1
50      COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
60      COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
70      COM Data(120),Ref,Zero
80      COM Tbegin$[8],Int$,Bumps,D(10)
90      COM Holes,Number(3),Temp(3,13)
100     COM SHORT Depth(3,13),M
110     CALL Init(#1,Scan1)
120     ON ERROR CALL Eproc
130     PRINTER IS 0
140     OUTPUT Tmr;"R"
150     ENTER Tmr USING "T";Tyme$
160     PRINT "AUTOSTART PROGRAM OPERATIONAL AT ";Tyme$
170     PRINTER IS 16
180     C$="DATA31:T15"
190     Scans=1
200     CREATE C$,1,10000
210     PRINT C$
220     E$="DATA31:F8,0"
230     CREATE E$,1,10000
240     PRINT E$
250     REM ---KEY DEFINITIONS
260     ON KEY #0 GOSUB 630     ! CHANGE SCAN RATE
270     ON KEY #1 GOSUB 820     ! FORCE A SCAN
280     ON KEY #5 GOSUB 860     ! DISPLAY TIME
290     ON KEY #15 GOTO 910     ! STOP RUN
300     Time=10   ! 10 MINUTE SCAN RATE USED IN AUTOSTART
310     PRINT "SCAN INTERVAL = ";Time;" MINUTES"
320     Int$=VAL$(60*Time*1000)
330     Tbegin$[1,8]="00:00:00"
340     OUTPUT Tmr;"R"
350     ENTER Tmr USING "T";Tyme$
360     Ten=VAL(Tyme$[10,10])
370     IF VAL(Tyme$[11,11])>8 THEN Ten=Ten+1
380     Ten=Ten+1
390     IF Ten>5 THEN Ten=Ten-6
400     Try$[1,1]=VAL$(Ten)
410     Tbegin$[4,8]=Try$[1,1]&"0:00"
420     Beep=0
430     Bumps=0
440     CALL Match
450     CALL Realtime
460     IF Beep>0 THEN 420
470     ON INT #Tmr GOTO Begin
480     IF Bumps>0 THEN Begin
490     CONTROL MASK Tmr;192
500     CARD ENABLE Tmr
510     GOTO 450
520 Begin: CALL Timer
```

```
530     U1=01
540     CALL Monitor
550     REM ---------------- MAIN PROGRAM LOOP -------------
560     CALL Realtime
570     IF Beep>0 THEN 420
580     CONTROL MASK Tmr;192
590     CARD ENABLE Tmr
600     IF Scans>20 THEN 930
610     ON INT #Tmr CALL Monitor
620     GOTO 550
630     REM ----------------KEY#0 PROCESSOR
640     BEEP
650     OFF INT #Tmr
660     OUTPUT Tmr;"A"
670     PRINT "CURRENT SCAN RATE = ";Int$;" MILLISECONDS"
680     INPUT "DO YOU WANT TO CHANGE SCAN RATE, Y OR N?",A3$
690     IF UPC$(A3$[1,1])<>"Y" THEN Return
700     INPUT "ENTER THE NEW SCAN INTERVAL (IN MINUTES)",Time
710     PRINT "NEW SCAN INTERVAL =";Time;" MINUTES"
720     Int$=VAL$(60*Time*1000)
730     Beep=Beep+1
740     INPUT "WHEN SHOULD NEW SCAN RATE BEGIN (HR:MIN:SEC)",Tbegin$
750     PRINT "NEW SCAN WILL BEGIN AT ";Tbegin$
760 Return:   INPUT "DO YOU WISH TO CHANGE YOUR MIND (Y OR N)",A3$
770     IF UPC$(A3$[1,1])="Y" THEN 700
780 Start: U1=01
790     CONTROL MASK Tmr;192
800     CARD ENABLE Tmr
810     RETURN
820     REM ------------ KEY #1 PROCESSOR
830     CALL Monitor
840     RETURN
850     REM ---------- KEY #5 PROCESSOR ****** TO DISPLAY TIME *******
860     OUTPUT Tmr;"R"
870     ENTER Tmr USING "T";Tyme$
880     DISP "CURRENT TIME IS ";Tyme$
890     WAIT 800
900     RETURN
910     REM ---------KEY#15 PROCESSOR
920     Dump=1
930     IF Scans>20 THEN Scans=20
940     OFF INT #Tmr
950     ASSIGN #1 TO C$
960     PRINT #1;Table$(*),Table2(*),Scans
970     PRINT #1;END
980     ASSIGN #1 TO *
990     ASSIGN #2 TO E$
1000    PRINT #2;Table$(*),Table2(*),Scans
1010    PRINT #2;END
1020    ASSIGN #2 TO *
1030    IF Dump=1 THEN 1210
1040    Scans=1
1050    IF VAL(C$[5,6])>=50 THEN 1180
1060    C$[5,6]=VAL$(VAL(C$[5,6])+1)
1070    CREATE C$,1,10000
1080    PRINTER IS 0
1090    PRINT C$
1100    E$[5,6]=VAL$(VAL(E$[5,6])+1)
1110    CREATE E$,1,10000
1120    PRINT E$
1130    PRINTER IS 16
1140    ON INT #Tmr CALL Monitor
1150    CONTROL MASK Tmr;192
1160    CARD ENABLE Tmr
1170    GOTO 550
1180    PRINT "I AM AT THE END OF THE TAPE AND FLOPPY, PUT IN NEW ONES AND
```

```
PRESS CONTINUE"
1190   PAUSE
1200   GOTO 170
1210   PRINT "I AM ENDING THE PROGRAM AT YOUR REQUEST, GOOD-BYE !!!!"
1220   END
1230   REM  ************************************************ Monitor  ******
1240   SUB Monitor
1250   OPTION BASE 1
1260   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
1270   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
1280   COM Data(120),Ref,Zero
1290   COM Tbegin$[8],Int$,Bumps,D(10)
1300   Bumps=Bumps+1
1310   CALL Scan(Fstore)
1320   ASSIGN #2 TO E$
1330   PRINT #2;Table$(*),Table2(*),Scans
1340   PRINTER IS 16
1350   ON Scans GOTO 1360,1380,1380,1360,1380,1380,1360,1380,1380,1360,1380,1380
,1360,1380,1380,1360,1380,1380,1360,1380
1360   PRINTER IS 0
1370   PRINT LIN(4)
1380   PRINT Time$;"  Scans =";Scans;"  Reference Junction = ";Zero;" Deg C","Fi
le Name is ";E$
1390   PRINT LIN(1)
1400   PRINT USING "6X,10(5X,ZZ)";1,2,3,4,5,6,7,8,9,10
1410   FIXED 2
1420   FOR I=1 TO 115 STEP 10
1430    D(1)=Table2(Scans,I)
1440    IF (D(1)>999) OR (D(1)<-999) THEN D(1)=999
1450    D(2)=Table2(Scans,I+1)
1460    IF (D(2)>999) OR (D(2)<-999) THEN D(2)=999
1470    D(3)=Table2(Scans,I+2)
1480    IF (D(3)>999) OR (D(3)<-999) THEN D(3)=999
1490    D(4)=Table2(Scans,I+3)
1500    IF (D(4)>999) OR (D(4)<-999) THEN D(4)=999
1510    D(5)=Table2(Scans,I+4)
1520    IF (D(5)>999) OR (D(5)<-999) THEN D(5)=999
1530    D(6)=Table2(Scans,I+5)
1540    IF (D(6)>999) OR (D(6)<-999) THEN D(6)=999
1550    D(7)=Table2(Scans,I+6)
1560    IF (D(7)>999) OR (D(7)<-999) THEN D(7)=999
1570    D(8)=Table2(Scans,I+7)
1580    IF (D(8)>999) OR (D(8)<-999) THEN D(8)=999
1590    D(9)=Table2(Scans,I+8)
1600    IF (D(9)>999) OR (D(9)<-999) THEN D(9)=999
1610    D(10)=Table2(Scans,I+9)
1620    IF (D(10)>999) OR (D(10)<-999) THEN D(10)=999
1630    PRINT USING "3X,ZZZ,2X,10(MDDD.DD)";I-1,D(1),D(2),D(3),D(4),D(5),D(6),D(
7),D(8),D(9),D(10)
1640   NEXT I
1650   PRINTER IS 0
1651   IF Data(108)>.71 THEN Pwr=1
1660   IF Pwr=1 THEN CALL Chan(121)
1670   IF Data(108)>.71 THEN PRINT LIN(2),"WATER SAMPLE BEGUN AT ";Time$
1671   IF Data(108)<.6 THEN CALL Chan(128)
1672   IF (Data(108)<.6) AND (Pwr=1) THEN PRINT LIN(1),"WATER SAMPLE ENDED AT ";
Time$
1673   IF Data(108)<.6 THEN Pwr=0
1680   IF (Data(83)>100) OR (Data(81)>100) THEN CALL Chan(122)
1690   IF (Data(83)>100) OR (Data(81)>100) THEN WAIT 1000
1700   IF (Data(11) AND Data(13))>330 THEN CALL Chan(123)
1710   IF (Data(11) AND Data(13))>330 THEN WAIT 5000
1720   IF (Data(11) AND Data(13))>400 THEN CALL Chan(124)
1730   IF (Data(11) AND Data(13))>400 THEN WAIT 5000
```

```
1731   PRINTER IS 16
1740   Scans=Scans+1
1750   SUBEND
1760   REM ************************************************************ Eproc **
1770   SUB Eproc
1780   OPTION BASE 1
1790   PRINTER IS 16
1800   PRINT ERRM$
1810   SUBEND
1820   REM ***************************************************  Realtime   ****
1830   SUB Realtime
1840   OPTION BASE 1
1850   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
1860   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
1870   OUTPUT Tmr;"R"
1880   ENTER Tmr USING "T";Tyme$
1890   Time$=Tyme$
1900   DISP Time$
1910   CARD ENABLE Tmr
1920   SUBEND
1930   REM ************************************************************ Scan **
1940   SUB Scan(Fstore)
1950   OPTION BASE 1
1960   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
1970   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
1980   COM Data(120),Ref
1981   PRINTER IS 16
1982   RESET 710
1990   CALL Realtime
2000   Table$(Scans)=Time$
2010   CALL Dvm(1,7,0,0,1)                  ! THERMOCOUPLE VOLTAGE
2020   CALL Measure(Tc1,Tc2,1)
2030   CALL Measure(Tf1,Tf2,2)
2040   CALL Dvm(1,7,0,0,1)                  ! PRESSURE AND STRESS GAUGE VOLTAGE
2050   CALL Measure(Pr1,Pr2,3)
2060   CALL Measure(Sm1,Sm2,4)
2070   CALL Dvm(5,7,0,0,1)                  ! THERMISTOR 4 WIRE OHMS
2080   CALL Measure(Th1,Th2,5)
2090   CALL Dvm(1,7,0,0,1)                  ! Ph GAUGE READING VOLTAGE
2100   CALL Measure(Ph1,Ph2,6)
2110   CALL Dvm(1,7,0,0,1)                  ! R.H. VOLTAGE
2120   CALL Measure(Rh1,Rh2,7)
2130   RESET 4
2140   ENTER 4;P$,Q$
2150   Data(118)=VAL(P$[7,12])
2160   PRINT "DATA CH # 118 IS ";Data(118)
2170   Table2(Scans,118)=Data(118)
2180   RESET 3
2190   ENTER 3;P$,Q$
2200   Data(119)=VAL(P$[7,12])
2210   PRINT "DATA CH # 119 IS ";Data(119)
2220   Table2(Scans,119)=Data(119)
2230   Data(120)=Data(118)+Data(119)
2240   Table2(Scans,120)=Data(120)
2250   PRINT "DATA CH # 120 IS ";Data(120)
2260   CALL Dvm(1,7,0,0,1)                  ! POWER METER ANALOG OUTPUTS
2270   CALL Measure(V1,W2,8)
2280   CALL Dvm(1,7,0,0,1)                  ! WATER LEVEL GAUGE VOLTAGE OUTPUT
2290   CALL Measure(Wat1,Wat2,9)
2300   CALL Dvm(1,7,0,0,1)                  ! LASER ANALOG OUTPUT
2310   CALL Measure(Las1,Las2,10)
2320   CALL Measure(Las3,Las4,11)
```

```
2330   SUBEND
2340   REM ******************************************************** Measure ***
2350   SUB Measure(INTEGER First,Last,Funct)
2360   OPTION BASE 1
2370   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
2380   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
2390   COM Data(120),Ref,Zero
2400   IF Last=0 THEN SUBEXIT
2410   FOR Channel=First TO Last
2420    Ch=Channel
2430    IF Channel=0 THEN Go_on            !--IF REF.CH. THEN SKIP
2440    IF Channel=20 THEN Go_on
2450    IF Channel=40 THEN Go_on
2460    IF Channel=60 THEN Go_on
2470     Offset=Channel DIV 80
2480     CALL Chan(Channel)
2490     Reading=FNRdvm                   !--STORE DATA TEMPORARILY
2500     ON Funct GOSUB Tc,Tc,Pr,Sm,Th,Ph,Rh,Pwr,Wat,Las,Xtra !CONVERSION
2510     RESET Scn+Offset                 !--RESET SCANNER
2520     PRINT "DATA CH # ";Ch;" IS ";Data(Ch)
2530     Table2(Scans,Ch)=Data(Ch)
2540   NEXT Channel
2550   SUBEXIT
2560 Skip: Data(Channel)=-8E60            !--UNUSED CHANNEL
2570   Table2(Scans,Channel)=Data(Channel)
2580   GOTO 2540
2590 Go_on: IF Channel=0 THEN Card=1
2600    IF Channel=20 THEN Card=2
2610    IF Channel=40 THEN Card=3
2620    IF Channel=60 THEN Card=4
2630    CALL Reference(Junc,Card)
2640    Ref=Junc
2650    IF Channel=0 THEN 2680
2660    Data(Ch)=Ref
2670    GOTO 2510
2680    Zero=Ref
2690    RESET Scn+Offset
2700    PRINT "REFERENCE JUNCTION IS ";Zero
2710    GOTO 2540
2720   REM ------------- Conversions
2730 Tc:  Xtime=Ch DIV 20*20
2740   Data(Ch)=FNTemp(Ch,Reading,Ref,Tc2)
2750   RETURN
2760 Ph:  Data(Ch)=(Reading+.00706)*1000
2770   RETURN
2780 Rh:  Data(Ch)=Reading*100
2790   RETURN
2800 Pr:  Press=Ch-79  ! DETERMINE WHICH PRESSURE GAUGE IS USED
2810  ON Press GOTO 2820,2850,2880,2910,2940,2970,3000,3030,3060
2820  Data(Ch)=(Reading-.703)/.0427
2830  IF Data(Ch)>15 THEN 3080
2840  RETURN
2850  Data(Ch)=(Reading-.025)/.048
2860  IF Data(Ch)>15 THEN 3080
2870  RETURN
2880  Data(Ch)=(Reading-.630)/.043
2890  IF Data(Ch)>15 THEN 3080
2900  RETURN
2910  Data(Ch)=(Reading+.048)/.0482
2920  IF Data(Ch)>15 THEN 3080
2930  RETURN
2940  Data(Ch)=(Reading-.678)/.0433
2950  IF Data(Ch)>15 THEN 3080
2960  RETURN
```

```
2970  Data(Ch)=(Reading+.066)/.0481
2980  IF Data(Ch)>15 THEN 3080
2990  RETURN
3000  Data(Ch)=(Reading-.636)/.043
3010  IF Data(Ch)>15 THEN 3080
3020  RETURN
3030  Data(Ch)=(Reading+.017)/.049
3040  IF Data(Ch)>15 THEN 3080
3050   RETURN
3060  Data(Ch)=(Reading-.005)/.0476
3070  RETURN
3080   FOR I=1 TO 10
3090    BEEP
3100    WAIT 700
3110   NEXT I
3120   PRINTER IS 16
3130   PRINT "*****************************************"
3140   PRINT "******  PRESSURE AT DANGER LEVEL  ********"
3150   PRINT "*****************************************"
3160   PRINTER IS 0
3170   PRINT LIN(3),"******** NOTE: POSSIBLE DANGEROUS PRESSURES ! ! !   *****",
LIN(1)
3180   PRINTER IS 16
3190   RETURN
3200 Sm:  Data(Ch)=Reading
3210   RETURN
3220 Wat: Data(Ch)=Reading
3230  RETURN
3240 Las: ! MUST DEFINE CONVERSION FOR LASER STRAINMETER
3250   Data(Ch)=(Reading+10)*4096*.6328/160 !(VOLTS)*(CTS/20VOLTS)*)MICRON/WAVE)
/((20)*(8 CTS/WAVE))
3260   RETURN
3270 Th:  Num=3976.74*298
3280   Den=298*LOG(Reading/4978.52)+3976.74
3290   Data(Ch)=Num/Den-273
3300   RETURN
3310 Pwr:  ! READ POWER METER
3320   IF Channel=V1 THEN Data(Ch)=Reading*29.8
3330   IF Channel=V2 THEN Data(Ch)=Reading*30
3340   IF Channel=A1 THEN Data(Ch)=Reading*1.99
3350   IF Channel=A2 THEN Data(Ch)=Reading*2
3360   IF Channel=W1 THEN Data(Ch)=Reading*310
3370   IF Channel=W2 THEN Data(Ch)=Reading*301
3380   RETURN
3390 Xtra:  ! NEW LASER FRINGE COUNTER INSTALLED 2/13/80
3400   Data(Ch)=(Reading+5)*4096*.6328/80  !(VOLTS)*(CTS/10VOLTS)*(MICRON/WAVE)/
(10*(8CTS/WAVE))
3410   RETURN
3420   SUBEND
3430   REM ********************************************************** Init **
3440   SUB Init(#1,Scan1)
3450   OPTION BASE 1
3460   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
3470   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
3480   COM Data(120),Ref,Zero
3490   COM Tbegin$[8],Int$
3500   Tbegin$="12:00:00"
3510   PRINT "INITIALIZATION"
3520   READ Tmr,Scn,Dvm,Bus                    !--SET UP ADDRESSES
3530   DATA 5,709,722,7
3540   ABORTIO Bus                                  !--RESET DEVICES ON BUS
3550   REMOTE Bus                                      !--BUS IN REMOTE MODE
3560   RESET Bus                                    !--RESET BUS
```

```
3570   READ IO Tmr,5;Sig                    !--IF CLOCK IS PRESENT, THEN START
3580     IF BINAND(Sig,48)=32 THEN OUTPUT Tmr USING "A";"B"
3590   Mistake=0
3600   OUTPUT Tmr;"U1=O1"
3610   READ Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,Pr1,Pr2,Sm1,Sm2,Wat1,Wat2,Las1,Las2,
W1,W2,V1,V2,A1,A2,Th1,Th2,Las3,Las4,Time
3620   DATA 0,79,0,0,93,96,97,101,80,88,89,91,108,111,112,113,104,107,102,105,10
3,106,114,117,92,92,300000
3630   Int$=VAL$(Time)
3640   PRINT "PROGRAM RUNNING"
3650   SUBEND
3660   REM ************************************************************ Dvm **
3670   SUB Dvm(Function,Range,High_res,Data_rqs,Auto_cal)
3680   OPTION BASE 1
3690   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
3700   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
3710   Error=((Function<1) OR (Function>6))+2*((Range<1) OR (Range>7))
3720     IF Error THEN CALL Problem("Dvm",Error)
3730    RESET Dvm
3740    OUTPUT Dvm USING Fmt;Function,Range,High_res,Data_rqs,Auto_cal
3750 Fmt: IMAGE "F",D,"R",D,"H",D,"D",D,"T3","A",D
3760   SUBEND
3770   REM ********************************************************** Chan **
3780   SUB Chan(Channel)
3790   OPTION BASE 1
3800   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
3810   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
3820   Offset=ABS(Channel) DIV 80
3830   Error=(Scn MOD 100+Offset>30)+2*(Channel>239)
3840   IF Error THEN CALL Problem("Chan",Error)
3850      IF Channel>=0 THEN OUTPUT Scn+Offset USING "ZZ";INT(Channel-80*Offset)
3860     SUBEND
3870   REM ******************************************************** Problem **
3880   SUB Problem(Where$,Error)
3890   PRINTER IS 0
3900   PRINT LIN(1);"ERROR TYPE ";Error;"IN ";Where$;" SUBROUTINE";LIN(1);ERRM$
3910   PRINTER IS 16
3920   SUBEND
3930   REM ********************************************************** Timer **
3940   SUB Timer
3950   OPTION BASE 1
3960   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
3970   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
3980   COM Data(120),Ref,Zero
3990   COM Tbegin$[8],Int$
4000   OUTPUT Tmr;"A"
4010   U1=O1
4020   OUTPUT Tmr;"B U1P",Int$
4030   OUTPUT Tmr;"U1G"
4040   SUBEND
4050   REM ********************************************************** Rdvm **
4060   DEF FNRdvm
4070   OPTION BASE 1
4080   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
4090   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
4100   TRIGGER Dvm
4110   ENTER Dvm USING "F";Reading
4120   RETURN Reading
```

```
4130   FNEND
4140   REM ********************************************************** Refjct **
4150   DEF FNRefjct(Resistance)
4160   PRINT Resistance
4170   RETURN 5041.6/(LOG(Resistance)+7.15)-314.052    ! YSI THERMISTOR
4180   FNEND
4190   REM ************************************************************ Temp **
4200   DEF FNTemp(Channel,Voltage,Junct,INTEGER T1)
4210   DIM Junct(3),Coeff(9)
4220   IF Voltage>.028863 THEN A
4230   Junct(1)=5.8637565E1
4240   Junct(2)=4.670025E-2
4250   Junct(3)=-1.4438022E-5
4260   Coeff(1)=1.7022525E-2   ! NBS MONOGRAM 125 QUADRATIC FIT 0-400 DEG
4270   Coeff(2)=-2.2097240E-7
4280   Coeff(3)=5.4809314E-12
4290   Coeff(4)=-5.7669892E-17
4300   Temp=FNPoly1(Coeff(*),FNPoly2(Junct(*),Junct)+Voltage*1E6)
4310   GOTO Retn
4320 A:  Junct(1)=5.8637565E1
4330     Junct(2)=4.670025E-2
4340     Junct(3)=-1.4438022E-5
4350     Coeff(1)=1.6410783E-2  ! NBS MONOGRAM 125 QUADRATIC FIT 0-1000 DEG
4360     Coeff(2)=-1.3560189E-7
4370     Coeff(3)=1.8600342E-12
4380     Coeff(4)=-8.5537337E-18
4390     Temp=FNPoly1(Coeff(*),FNPoly2(Junct(*),Junct)+Voltage*1E6)
4400 Retn:  RETURN Temp
4410   REM ********************************************************** Poly1 **
4420   DEF FNPoly1(Co(*),Val)=(((Co(4)*Val+Co(3))*Val+Co(2))*Val+Co(1))*Val
4430   REM ********************************************************** Poly2 **
4440   DEF FNPoly2(Co(*),Value)=((Co(3)*Value+Co(2))*Value+Co(1))*Value
4450   REM ********************************************** Reference ********
4460   SUB Reference(Ref,Card)
4470   OPTION BASE 1
4480   COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEG
ER Scans
4490   COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,P
r1,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
4500   CALL Dvm(5,4,1,0,1)
4510   IF Card>1 THEN 4540
4520   CALL Chan(0)
4530   GOTO 4610
4540   IF Card>2 THEN 4570
4550   CALL Chan(20)
4560   GOTO 4610
4570   IF Card>3 THEN 4600
4580   CALL Chan(40)
4590   GOTO 4610
4600   CALL Chan(60)
4610   Ref=FNRefjct(FNRdvm)
4620   CALL Dvm(1,1,0,0,1)
4630   SUBEND
4640   REM  ********************************************  Match  ********
4650 SUB Match
4660 OPTION BASE 1
4670 COM Table$(20),Radius(3),SHORT Table2(20,120),A$,C$,E$,G$,B1$,Time$,INTEGER
 Scans
4680 COM INTEGER Tmr,Scn,Dvm,Bus,Pwr,Mistake,Tc1,Tc2,Tf1,Tf2,Ph1,Ph2,Rh1,Rh2,Pr1
,Pr2,Sm1,Sm2,W1,W2,V1,V2,A1,A2,Th1,Th2,Wat1,Wat2,Las1,Las2,Las3,Las4
4690 COM Data(120),Ref,Zero
4700 COM Tbegin$[8],Int$
4710 OUTPUT Tmr;"A"
4720 U1=01
4730 OUTPUT Tmr;"B U1M",Tbegin$[4,8]
4740 OUTPUT Tmr;"U1G"
4750 SUBEND
```

APPENDIX C

PRINTC Program

```
10   REM  PROGRAM TO READ AND PRINT DATA FROM TUFF HEATER  4/25/80  *****
20    REM ***************** REVISION C *********************
30   OPTION BASE 1
40   COM Table$(20),SHORT Table2(20,120),INTEGER Scans
50   COM Time$(260),SHORT Data(260,120),INTEGER Tscan,SHORT Ch(10),Ch$(10)[5]
60   PRINTER IS 16
70   FIXED 0
80   INPUT "IS DATA GOING TO BE READ IN ON FLOPPY DISK?",A$
90   IF UPC$(A$[1,1])="N" THEN 110
100  MASS STORAGE IS ":F8"
110   INPUT "IS DATA ALREADY PRE-RECORDED ON MONSTER FILE?",M$
120   IF UPC$(M$[1,1])<>"Y" THEN 170
130   INPUT "WHAT IS MONSTER FILE NAME?",C$
140   ASSIGN #1 TO C$[1,6]
150   READ #1;Time$(*),Data(*),Tscan
160   GOTO 420
170  INPUT "WHAT IS FIRST DATA FILE TO BE READ?",F$[1,6]
180  ASSIGN #1 TO F$[1,6]
190  READ #1;Table$(*),Table2(*),Scans
200  FOR I=1 TO Scans
210   Time$(I)=Table$(I)
220   FOR J=1 TO 120
230    Data(I,J)=Table2(I,J)
240   NEXT J
250  NEXT I
260  Tscan=Scans
270 Read: Left=260-Tscan
280  PRINT "ROOM FOR ";Left;" MORE SCANS."
290  INPUT "DO YOU WANT TO READ IN MORE DATA, Y OR N?",B$
300  IF UPC$(B$[1,1])="N" THEN Proceed
310  INPUT "WHAT IS NEXT DATA FILE TO BE READ?",F$[1,6]
320  ASSIGN #1 TO F$[1,6]
330  READ #1;Table$(*),Table2(*),Scans
340  FOR I=1 TO Scans
350   Time$(I+Tscan)=Table$(I)
360   FOR J=1 TO 120
370    Data(I+Tscan,J)=Table2(I,J)
380   NEXT J
390  NEXT I
400  Tscan=Tscan+Scans
410  GOTO Read
420 Proceed: INPUT "DO YOU WANT TO PRINT DATA, Y OR N?",C$
430  FIXED 3
440  IF UPC$(C$[1,1])="N" THEN Save
450  INPUT "ARE YOU USING DTC TERMINAL FOR PRINTING, Y OR N?",D$
460   PRINTER IS 0
470   FIXED 0
480  INPUT "DO YOU WANT A COLUMNIZED PRINTOUT OF SELECTED CHANNELS OF HIGH RESOL
UTION,Y OR N",E$
490  IF UPC$(E$[1,1])<>"Y" THEN General
500   PRINTER IS 16
510   PRINT "I WILL ALLOW 5 CHANNELS OF SPECIAL CHANNEL OUTPUT!"
520   FOR I=1 TO 5
530    DISP "WHAT IS DATA CHANNEL # TO BE PRINTED IN COLUMN # ";I;
```

```
540     INPUT Ch(I)
550     DISP "WHAT IS CHANNEL I.D. FOR CHANNEL ";Ch(I);
560     INPUT Ch$(I)
570    NEXT I
580    PRINTER IS 0
590    PRINT PAGE,LIN(6)
600    PRINT USING "2X,K,5(8X,M3D)";"TIME  CH#";Ch(1),Ch(2),Ch(3),Ch(4),Ch(5)
610    PRINT USING "13X,5(7X,5A)";Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5)
620    PRINT LIN(1)
630    FOR I=1 TO Tscan
640     IF I=51 THEN 720
650     IF I=101 THEN 720
660     IF I=151 THEN 720
670     IF I=201 THEN 720
680     IF I=251 THEN 720
690     PRINT USING "X,K,5(2X,M4D.DDDD)";Time$(I)[1,11],Data(I,Ch(1)),Data(I,Ch(2
)),Data(I,Ch(3)),Data(I,Ch(4)),Data(I,Ch(5))
700    NEXT I
710    GOTO General
720    PRINT PAGE,LIN(6)
730    PRINT USING "2X,K,5(8X,M3D)";"TIME  CH#";Ch(1),Ch(2),Ch(3),Ch(4),Ch(5)
740    PRINT USING "13X,5(7X,5A)";Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5)
750    GOTO 690
760 General:  INPUT "DO YOU WANT COLUMNIZED PRINT OUT OF ALL CHANNELS",F$
770    IF UPC$(F$[1,1])<>"Y" THEN Group
780    DATA  1,2,3,4,5,6,7,8,9,10
790    READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
800    DATA T1,T2,T3,T4,T5,T6,T7,T8,T9,T10
810    READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
820    GOSUB Print
830    DATA 11,12,13,14,15,16,17,18,19,20
840    READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
850    DATA T11,T12,T13,T14,T15,T16,T17,T18,Cal,Ref
860    READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
870    GOSUB Print
880    DATA 21,22,23,24,25,26,27,28,29,30
890    READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
900    DATA T19,T20,T21,T22,T23,T24,T25,T26,T27,T28
910    READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
920    GOSUB Print
930    DATA 31,32,33,34,35,36,37,38,39,40
940    READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
950    DATA T29,T30,T31,T32,T33,T34,T35,T36,Cal,Ref
960    READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
970    GOSUB Print
980    DATA 41,42,43,44,45,46,47,48,49,50
990    READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
1000   DATA T37,T38,T39,T40,T41,T42,T43,T44,T45,T46
1010   READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
1020   GOSUB Print
1030   DATA 51,52,53,54,55,56,57,58,59,60
1040   READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
1050   DATA T47,T48,T49,T50,T51,T52,T53,T54,Cal,Ref
1060   READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
1070   GOSUB Print
1080   DATA 61,62,63,64,65,66,67,68,69,70
1090   READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
1100   DATA T55,T56,T57,T58,T59,T60,T61,T62,T63,T64
1110   READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
```

```
1120  GOSUB Print
1130  DATA 71,72,73,74,75,76,77,78,79,80
1140  READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
1150  DATA T65,T66,T67,T68,T69,T70,T71,T72,Cal,PR-1
1160  READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
1170  GOSUB Print
1180  DATA 81,82,83,84,85,86,87,88,89,90
1190  READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
1200  DATA PR-2,PR-3,PR-4,PR-5,PR-6,PR-7,PR-8,PR-9,S-1,S-2
1210  READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
1220  GOSUB Print
1230  DATA 91,92,93,94,95,96,97,98,99,100
1240  READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
1250  DATA S-3,LAS-2,PH-1,PH-2,PH-3,PH-4,RH-2,RH-1,Cal,RH-4
1260  READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
1270  GOSUB Print
1280  DATA 101,102,103,104,105,106,107,108,109,110
1290  READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
1300  DATA RH-3,V-1,A-1,P-1,V-2,A-2,P-2,WAT-1,WAT-2,WAT-3
1310  READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
1320  GOSUB Print
1330  DATA 111,112,113,114,115,116,117,118,119,120
1340  READ Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch(6),Ch(7),Ch(8),Ch(9),Ch(10)
1350  DATA WAT-4,LAS-1,CAL,TH-1,TH-2,TH-3,TH-4,P-1D,P-2D,P-3D
1360  READ Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7),Ch$(8),Ch$(9),Ch$(10
)
1370  GOSUB Print
1380  GOTO Group
1390 Print:  PRINT PAGE,LIN(6)
1400  IF UPC$(D$[1,1])="Y" THEN Dtc
1410  PRINT USING "2X,K,10(2X,M3D)";"TIME  CH#";Ch(1);Ch(2);Ch(3);Ch(4);Ch(5);Ch
(6);Ch(7);Ch(8);Ch(9);Ch(10)
1420  PRINT USING "13X,10(X,5A)";Ch$(1);Ch$(2);Ch$(3);Ch$(4);Ch$(5);Ch$(6);Ch$(7
);Ch$(8);Ch$(9);Ch$(10)
1430  PRINT LIN(1)
1440  GOTO 1490
1450 Dtc:  ! SET UP FOR DTC TERMINAL PRINT
1460  PRINT USING "2X,K,10(6X,K,M3Z)";"SCAN TIME","CH",Ch(1),"CH",Ch(2),"CH",Ch(
3),"CH",Ch(4),"CH",Ch(5),"CH",Ch(6),"CH",Ch(7),"CH",Ch(8),"CH",Ch(9),"CH",Ch(10)
1470  PRINT USING "15X,10(8X,K)";Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7
),Ch$(8),Ch$(9),Ch$(10)
1480  PRINT LIN(1)
1490 FOR I=1 TO Tscan
1500  IF (Data(I,Ch(1))>999) OR (Data(I,Ch(1))<-999) THEN Data(I,Ch(1))=999
1510  IF (Data(I,Ch(2))>999) OR (Data(I,Ch(2))<-999) THEN Data(I,Ch(2))=999
1520  IF (Data(I,Ch(3))>999) OR (Data(I,Ch(3))<-999) THEN Data(I,Ch(3))=999
1530  IF (Data(I,Ch(4))>999) OR (Data(I,Ch(4))<-999) THEN Data(I,Ch(4))=999
1540  IF (Data(I,Ch(5))>999) OR (Data(I,Ch(5))<-999) THEN Data(I,Ch(5))=999
1550  IF (Data(I,Ch(6))>999) OR (Data(I,Ch(6))<-999) THEN Data(I,Ch(6))=999
1560  IF (Data(I,Ch(7))>999) OR (Data(I,Ch(7))<-999) THEN Data(I,Ch(7))=999
1570  IF (Data(I,Ch(8))>999) OR (Data(I,Ch(8))<-999) THEN Data(I,Ch(8))=999
1580  IF (Data(I,Ch(9))>999) OR (Data(I,Ch(9))<-999) THEN Data(I,Ch(9))=999
1590  IF (Data(I,Ch(10))>999) OR (Data(I,Ch(10))<-999) THEN Data(I,Ch(10))=999
1600  IF I=51 THEN 2130
1610  IF I=101 THEN 2130
1620  IF I=151 THEN 2130
1630  IF I=201 THEN 2130
1640  IF I=251 THEN 2130
1650  IF UPC$(D$[1,1])="Y" THEN Wide
1660  PRINT USING "#,X,K,5(M3D.D)";Time$(I)[1,11],Data(I,Ch(1)),Data(I,Ch(2)),Da
ta(I,Ch(3)),Data(I,Ch(4)),Data(I,Ch(5))
1670  PRINT USING "5(M3D.D)";Data(I,Ch(6)),Data(I,Ch(7)),Data(I,Ch(8)),Data(I,Ch
```

```
(9)),Data(I,Ch(10))
1680  GOTO 1720
1690 Wide: ! SET UP FOR DTC TERMINAL
1700  PRINT USING "#,10X,K,5(M8D.DD)";Time$(I)[7,11],Data(I,Ch(1)),Data(I,Ch(2))
,Data(I,Ch(3)),Data(I,Ch(4)),Data(I,Ch(5))
1710  PRINT USING "5(M8D.DD)";Data(I,Ch(6)),Data(I,Ch(7)),Data(I,Ch(8)),Data(I,C
h(9)),Data(I,Ch(10))
1720 NEXT I
1730  RETURN
1740 Group:INPUT "DO YOU WANT TO PRINT SCAN-GROUPED DATA, Y OR N?",Sc$
1750 IF UPC$(Sc$[1,1])="N" THEN Save
1760 FOR I=1 TO Tscan
1770  IF FRACT(I/2)<>0 THEN PRINT PAGE
1780  PRINT LIN(3)
1790  PRINT Time$(I),LIN(2)
1800  IF UPC$(D$[1,1])<>"Y" THEN Size
1810  PRINT USING "15X,10(8X,ZZ)";1,2,3,4,5,6,7,8,9,10
1820  PRINT LIN(1)
1830  FIXED 4
1840  FOR J=1 TO 115 STEP 10
1850   PRINT USING "#,11X,ZZZ,3X,5(MDDDD.DDDD)";J-1,Data(I,J),Data(I,J+1),Data(I
,J+2),Data(I,J+3),Data(I,J+4)
1860   PRINT USING "5(MDDDD.DDDD)";Data(I,J+5),Data(I,J+6),Data(I,J+7),Data(I,J+
8),Data(I,J+9)
1870  NEXT J
1880  GOTO Finish
1890 Size:  PRINT USING "6X,10(5X,ZZ)";1,2,3,4,5,6,7,8,9,10
1900  PRINT LIN(1)
1910  FIXED 2
1920  FOR J=1 TO 115 STEP 10
1930   PRINT USING "#,3X,ZZZ,2X,5(MDDD.DD)";J-1,Data(I,J),Data(I,J+1),Data(I,J+2
),Data(I,J+3),Data(I,J+4)
1940   PRINT USING "5(MDDD.DD)";Data(I,J+5),Data(I,J+6),Data(I,J+7),Data(I,J+8),
Data(I,J+9)
1950  NEXT J
1960 Finish:NEXT I
1970 Save:INPUT "DO YOU WANT TO SAVE COMBINED DATA TABLES ON TAPE/DISK, Y OR N?"
,Sa$
1980 IF UPC$(Sa$[1,1])<>"Y" THEN Exit
1990 INPUT "DO YOU WANT TO SAVE IT ON DISK, Y OR N?",Sa$
2000 IF UPC$(Sa$[1,1])="N" THEN Tape
2010 INPUT "WHAT IS THE NAME OF THE FILE WHERE DATA IS TO BE STORED ON DISK?",Fi
le$[1,6]
2020 File$[7,11]=":F8,0"
2030 CREATE File$,13,10000
2040 ASSIGN #2 TO File$
2050 PRINT #2;Time$(*),Data(*),Tscan
2060 GOTO Exit
2070 Tape:INPUT "WHAT IS FILE NAME WHERE DATA IS TO BE STORED ON TAPE?",File$
2080 File$[7,11]=":T15 "
2090 CREATE File$,13,10000
2100 ASSIGN #3 TO File$
2110 PRINT #3;Time$(*),Data(*),Tscan
2120 GOTO Exit
2130 PRINT PAGE,LIN(6)
2140  IF UPC$(D$[1,1])="Y" THEN Big
2150  PRINT USING "2X,K,10(2X,M3D)";"TIME  CH#",Ch(1),Ch(2),Ch(3),Ch(4),Ch(5),Ch
(6),Ch(7),Ch(8),Ch(9),Ch(10)
2160  PRINT USING "13X,10(X,5A)";Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7
),Ch$(8),Ch$(9),Ch$(10)
2170  PRINT LIN(1)
2180  GOTO 2220
2190 Big: ! SET UP DTC TERMINAL
2200  PRINT USING "5X,K,10(8X,K,M3D)";"SCAN TIME","CH",Ch(1),"CH",Ch(2),"CH",Ch(
3),"CH",Ch(4),"CH",Ch(5),"CH",Ch(6),"CH",Ch(7),"CH",Ch(8),"CH",Ch(9),"CH",Ch(10)
```

```
2210  PRINT USING "15X,10(8X,K)";Ch$(1),Ch$(2),Ch$(3),Ch$(4),Ch$(5),Ch$(6),Ch$(7
),Ch$(8),Ch$(9),Ch$(10)
2220 GOTO 1660
2230 Exit: END
```

APPENDIX D

WATERQ Program

```
10    REM WATERQ PROGRAM TO READ DATA TAPE/FLOPPY AND PLOT  6/10/80 *****
20     REM  ****************  REVISION A  *****************
30    OPTION BASE 1
40    COM Table$(20),SHORT Table2(20,120),INTEGER Scans
50    COM Time$(2500),SHORT Data(2500,10),INTEGER Tscan,Group,REAL Tbegin,Tend
60    COM Tyme(15),Model(15),Domain(20),Func(20),P1$,Repeat$
70    Narg=20
80     Tscan=0
90     Diff$=""
100    Repeat$=""
110    P=1
120    Q=0
130 Begin:  Different=0
131    Tscan=0
140    ON KEY #3 GOTO 600
150    ON KEY #4 GOTO 840
160    ON KEY #5 GOTO 1030
170   PRINTER IS 16
180    INPUT "IS DATA GOING TO BE READ IN FROM A FLOPPY DISK? Y OR N",A$
190    PRINT "THIS PROGRAM ACCEPTS AND PLOTS DATA IN 10-CHANNEL GROUPS",LIN(1),"T
HE GROUPS MUST CORRESPOND TO THOSE ON TAPES/DISKS WITH CURRENT MONSTER"
200    PRINT "FILES. USUALLY THESE GROUPS  WILL BEGIN WITH CHANNELS 1,11,21,31, E
TC.",LIN(1)
210    INPUT "WHAT IS FIRST CHANNEL OF TEN CHANNEL GROUP BEING READ IN?",Group
220    EXIT GRAPHICS
230    PRINT PAGE
240   IF UPC$(A$[1,1])="N" THEN 300
250   MASS STORAGE IS ":F8"
260    INPUT "IS DATA ALREADY PRE-RECORDED ON A MONSTER FILE?",M$
270    IF UPC$(M$[1,1])<>"Y" THEN 550
280    PRINT "PUT MONSTER DATA FILE DISK IN F8 FLOPPY DISK DRIVE",LIN(1)
290    GOTO 340
300    MASS STORAGE IS ":T15"
310    INPUT "IS DATA ALREADY PRE-RECORDED ON A MONSTER FILE?",M$
320    IF UPC$(M$[1,1])<>"Y" THEN 550
330    PRINT "PUT MONSTER DATA FILE TAPE IN T15 CASSETTE TAPE DRIVE",LIN(1)
340    INPUT "WHAT IS MONSTER FILE NAME? (NEEDS SIX CHARACTERS)",D$
350    PRINT PAGE
360    ASSIGN #1 TO D$[1,6]
370    READ #1;Time$(*),Data(*),Tscan
380    PRINT "YOU CAN ADD MORE DATA FILES TO THE END OF THIS MONSTER FILE",LIN(1)
,"PROVIDING IT IS NOT ALREADY FILLED UP. THE COMBINED FILE"
390    PRINT "CAN THEN BE SAVED AS A NEW, MORE COMPLETE MONSTER FILE.",LIN(5)
400    INPUT "IS THERE MORE DATA TO BE ADDED TO THIS MONSTER FILE? Y OR N",A$
410    PRINT PAGE
420    IF UPC$(A$[1,1])="N" THEN Plot
430    INPUT "IS THE ADDITIONAL DATA TO BE READ IN FROM A FLOPPY DISK? Y OR N",Ad
d$
440    IF UPC$(Add$[1,1])="N" THEN 480
450    MASS STORAGE IS ":F8,0"
460    PRINT "PUT FLOPPY DISK FROM WHICH ADDITIONAL DATA IS TO BE READ INTO THE",
LIN(1),"MASTER DISK DRIVE, F8,0.",LIN(1)
470    GOTO 500
480    MASS STORAGE IS ":T15"
490    PRINT "PUT CASSETTE TAPE FROM WHICH ADDITIONAL DATA IS TO BE READ INTO THE
```

```
",LIN(1),"RIGHT HAND TAPE DRIVE, T15.",LIN(1)
500   PRINT "....THEN PRESS CONTINUE. THE CATALOG OF THIS TAPE/FLOPPY WILL THE
N",LIN(1),"BE PRINTED ON CRT DISPLAY FOR YOU TO SEE WHAT FILES ARE THERE",LIN(1)
510   PAUSE
520   PRINT PAGE
530   CAT
540   GOTO Read
550   PRINT "PUT TAPE OR FLOPPY (WHICHEVER WAS SPECIFIED AS BEING THE MEDIUM",LI
N(1),"FROM WHICH DATA WAS TO BE READ) INTO THE PROPER DRIVE."
560   PRINT "....THEN PRESS CONTINUE. THE CATALOG OF THIS TAPE/FLOPPY WILL THE
N",LIN(1),"BE PRINTED ON CRT DISPLAY FOR YOU TO SEE WHAT FILES ARE THERE",LIN(1)
570   PAUSE
580   PRINT PAGE
590   CAT
600 Read: Left=2500-Tscan
610  FIXED 0
620  PRINT "ROOM FOR ";Left;" MORE SCANS."
630  INPUT "DO YOU WANT TO READ IN MORE DATA, Y OR N?",B$
640  IF UPC$(B$[1,1])="N" THEN Save
650   PRINT LIN(5),"NOTE: IT IS ASSUMED THAT ALL THE DATA IS STORED IN FILES U
SING THE FORMAT:",LIN(1),"DATA#, WHERE # IS SOME NUMERAL 1,2,3,...., ETC."
660   PRINT LIN(3),"IN INPUTING INFORMATION TO READ IN ADDITIONAL DATA ONLY TYPE
 IN",LIN(1),"THE NUMERAL OF THE FIRST DATA FILE AND NOT THE ENTIRE DATA"
670   PRINT "FILE NAME.....",LIN(10)
680  INPUT "WHAT IS THE FIRST DATA FILE NUMBER TO BE READ?",F
690   INPUT "HOW MANY CONSECTUTIVELY NUMBERED FILES ARE TO BE READ IN?",N
700   F$[1,4]="DATA"
710  FOR K=F TO F+N-1 STEP 1
720   F$[5,6]=VAL$(K)
730  ASSIGN #1 TO F$[1,6]
740  READ #1;Table$(*),Table2(*),Scans
750  FOR I=1 TO Scans
760   Time$(I+Tscan)=Table$(I)
770   FOR J=1 TO 10
780    Data(I+Tscan,J)=Table2(I,J+Group-1)
790   NEXT J
800  NEXT I
810  Tscan=Tscan+Scans
820   NEXT K
830  GOTO Read
840 Save: PRINT LIN(5),"YOU CAN SAVE THIS DATA ON DISK AS A MONSTER FILE. IF YOU
 WANT",LIN(1),"TO SAVE ON TAPE YOU MUST CHANGE LINE #910 FROM F8,0 TO T15 THEN"
850   PRINT "CONT FROM LINE#840.",LIN(5)
860   INPUT "DO YOU WANT TO SAVE COMBINED DATA TABLES ON TAPE/DISK, Y OR N?",Sa$
870   PRINT PAGE
880   IF UPC$(Sa$[1,1])="N" THEN Plot
890   PRINT "WHEN YOU SAVE THE FILE YOU MUST YOU A SIX-CHARACTER NAME.",LIN(3)
900   INPUT "WHAT IS THE NAME OF THE FILE WHERE DATA IS TO BE STORED ON DISK?",F
ile$[1,6]
910   File$[7,11]=":F8,0"
920   PRINT "IF THIS IS THE SAME NAME AS FILE THAT ALREADY EXISTS ON THE DISK",L
IN(1),"YOU WILL NEED TO PURGE THE OLD COPY USING: PURGE 'OLD FILE:F8' EXECUTE"
930   PRINT "WHERE THE NAME 'OLDFILE' STANDS FOR WHATEVER THE FILE NAME IS OF TH
E",LIN(1),"ORIGINAL FILE THAT NEEDS TO BE PURGED, E.G. HIS101 OR HIS321."
940   PRINT "THIS MUST BE DONE BEFORE THE NEW FILE CAN BE CREATED."
950   PRINT "IF THIS IS NECESSARY DO IT NOW. IN EITHER CASE, WHEN YOU ARE READY"
,LIN(1),"TO HAVE THE NEW FILE ACTUALLY CREATED THEN PRESS CONTINUE.",LIN(1)
960   PRINT "THE PROGRAM WILL NOW WAIT FOR YOU TO DO PURGING IF NECESSARY",LIN(1
),"AND WAIT FOR YOU TO PRESS CONTINUE BEFORE PROCEEDING FURTHER.",LIN(5)
970   PAUSE
980   CREATE File$,15,10000
990   ASSIGN #2 TO File$
1000  PRINT #2;Time$(*),Data(*),Tscan
1010  PRINT #2;END
1020  ASSIGN #2 TO *
1030 Plot:REM THIS BEGINS PLOTTING ROUTINE BY SELECTING TYPE OF PLOT DESIRED
```

```
1031  FIXED 8
1040 PRINTER IS 16
1050  PRINT PAGE
1060  PEN P
1070  PRINT "REMEMBER: CHANNELS NUMBERS TO BE PLOTTED NEED TO BE IN GROUP!",LIN(
1)
1080 INPUT "WHICH CHANNEL NUMBER DO YOU WISH TO PLOT?",Chan
1090 INPUT "DO YOU WANT A DIFFERENT SCALE ON THE RIGHT-HAND SIDE OF THE PLOT? Y
OR N",Diff$
1100 IF UPC$(Repeat$[1,1])="Y" THEN New_label
1110 INPUT "ARE YOU USING THE 9872A 4-PEN PLOTTER, Y OR N?",P1$
1120 History: REM SET UP TIME HISTORY OF SELECTED CHANNELS
1130 INPUT "WHAT IS J-DAY WHEN YOU WANT THE PLOT TO BEGIN? ",Tbegin
1140 INPUT "WHAT IS J-DAY WHEN YOU WANT THE PLOT TO END?",Tend
1150 ! Tend=Tend+1
1160  Time=Tend-Tbegin
1170  Miny=0
1180  PRINT " PROGRAM CAN DO AUTO SCALING OF Y-AXIS. IF YOU DON'T WANT IT THEN",
LIN(1),"YOU WILL NEED TO ADD YOUR OWN MAX AND MIN VALUES.",LIN(1)
1190 INPUT "DO YOU WANT AUTO SCALING OF Y-AXIS? Y OR N.",As$
1200 IF UPC$(As$[1,1])<>"Y" THEN Manual
1210 Maxy=10
1220 FOR K=1 TO Tscan
1230  IF Data(K,Chan)>Maxy THEN Maxy=Data(K,Chan)
1240 NEXT K
1250 Maxy=10*INT(Maxy/10)+10
1260 GOTO 1290
1270 Manual: INPUT "ENTER MAXIMUM VALUE DESIRED FOR LEFT-SIDE Y-AXIS PLOTTING",M
axy
1280  INPUT "ENTER MINIMUM VALUE DESIRED FOR LEFT-SIDE Y-AXIS PLOTTING",Miny
1290  INPUT "ENTER LABEL FOR UNITS ON LEFT-SIDE ON Y-AXIS",Units$
1300  IF UPC$(Diff$[1,1])<>"Y" THEN Scale
1310  INPUT "ENTER MAXIMUM VALUE DESIRED FOR RIGHT-SIDE Y-AXIS PLOTTING",Maxr
1320  INPUT "ENTER MINIMUM VALUE DESIRED FOR RIGHT-SIDE Y-AXIS PLOTTING",Minr
1330  INPUT "ENTER LABEL FOR UNITS ON RIGHT-SIDE ON Y-AXIS",Unitr$
1340 Scale: Xlim=110
1350 IF UPC$(P1$[1,1])<>"Y" THEN 1430
1360  PRINT "IN ORDER TO PLOT CORRECTLY ON 9872A PLOTTER THE PROGRAM NEEDS THE",
LIN(1),"DECIMAL FRACTION OF THE EXPRESSION 'PAPER WIDTH/PAPER HEIGHT'",LIN(1)
1370  INPUT "WHAT IS DECIMAL FRACTION Xscale/Yscale?",Xlim
1380  Xlim=100*Xlim
1390 PLOTTER IS 7,5,"9872A"
1400 PEN P
1410  LOCATE 0,Xlim,0,100
1420 GOTO 1460
1430 PLOTTER IS 13,"GRAPHICS"
1440 GRAPHICS
1450 LOCATE 5,Xlim,0,100
1460 SCALE Tbegin,Tend,Miny,Maxy
1470 CSIZE 2.5
1480 LORG 2
1490  IF Maxy-Miny>.1 THEN 1580
1500  AXES 1,.002,Tbegin,Miny-.01,5,5
1510  FOR V=Miny+.002 TO Maxy-.001 STEP .002
1520   MOVE Tbegin,V
1530   LABEL USING "MD.DDD";V
1540  NEXT V
1550  IF UPC$(Diff$[1,1])="Y" THEN New_scale
1560  AXES 1,.002,Tend,Miny-.01,5,5
1570  GOTO 1920
1580  IF Maxy-Miny>1 THEN 1670
1590  AXES 1,.05,Tbegin,Miny-.25,5,5
1600  FOR V=Miny+.05 TO Maxy-.01 STEP .05
1610   MOVE Tbegin,V
1620   LABEL USING "MD.DD,X";V
1630  NEXT V
```

```
1640  IF UPC$(Diff$[1,1])="Y" THEN New_scale
1650  AXES 1,.05,Tend,Miny-.25,5,5
1660  GOTO 1920
1670  IF Maxy-Miny>10 THEN 1760
1680  AXES 1,.2,Tbegin,Miny-1,5,5
1690  FOR V=Miny+1 TO Maxy-.1 STEP 1
1700   MOVE Tbegin,V
1710   LABEL USING "M3D.D,X";V
1720  NEXT V
1730  IF UPC$(Diff$[1,1])="Y" THEN New_scale
1740  AXES 1,.2,Tend,Miny-1,5,5
1750  GOTO 1920
1760 IF Maxy-Miny>100 THEN 1850
1770  AXES 1,2,Tbegin,Miny-10,5,5
1780 FOR V=Miny+10 TO Maxy-1 STEP 10
1790  MOVE Tbegin,V
1800  LABEL USING "M3D,X";V
1810 NEXT V
1820  IF UPC$(Diff$[1,1])="Y" THEN New_scale
1830  AXES 1,2,Tend,Miny-10,5,5
1840 GOTO 1920
1850  AXES 1,10,Tbegin,Miny-50,5,5
1860  FOR V=Miny+50 TO Maxy-5 STEP 50
1870   MOVE Tbegin,V
1880   LABEL USING "M3D,X";V
1890  NEXT V
1900  IF UPC$(Diff$[1,1])="Y" THEN New_scale
1910  AXES 1,10,Tend,Miny-50,5,5
1920 SCALE Tbegin,Tend,Miny,Maxy
1930 MOVE Tbegin+Time/12,(Maxy-Miny)/2+Miny
1940  LDIR PI/2
1950  LORG 6
1960 CSIZE 3
1970 LABEL USING "K";Units$
1971  CSIZE 2.4
1980 LORG 4
1990  LDIR 0
1991  IF Time>1/16 THEN 1998
1992  AXES 1/1440,10,Tbegin,Miny,5,5
1993  AXES 1/1440,10,Tbegin,Maxy,5,5
1994  FOR H=Tbegin+1/288 TO Tend-1/576 STEP 1/288
1995   MOVE H,(Maxy-Miny)/100+Miny
1996   LABEL USING "ZZ,K,ZZ";INT(24*FRACT(H)+.001),":",(H-Tbegin)*1440
1997  NEXT H
1998  IF Time>1/4 THEN 2009
1999  AXES 1/288,10,Tbegin,Miny,6,5
2000  AXES 1/288,10,Tbegin,Maxy,6,5
2001  FOR H=Tbegin+1/48 TO Tend-1/96 STEP 1/48
2002   MOVE H,(Maxy-Miny)/100+Miny
2003   LABEL USING "ZZ,K,ZZ";INT(24*FRACT(H)+.01),":",60*FRACT(24*FRACT(H)+.0001
)
2004  NEXT H
2009  IF Time>1 THEN 2140
2010  AXES 1/24,1,Tbegin-1,Miny,12,5
2020  AXES 1/24,1,Tbegin-1,Maxy,12,5
2030  MOVE Tbegin+.25,Miny
2040  DRAW Tbegin+.25,Maxy
2050  MOVE Tbegin+.5,Miny
2060  DRAW Tbegin+.5,Maxy
2070  MOVE Tbegin+.75,Miny
2080  DRAW Tbegin+.75,Maxy
2090  FOR H=Tbegin+.25 TO Tend-.1 STEP .25
2100   MOVE H,(Maxy-Miny)/100+Miny
2110   LABEL USING "M3D.DD";H
2120  NEXT H
2130  GOTO 2360
```

```
2140  IF Time>5 THEN 2220
2150  AXES 1/12,10,Tbegin-1,Miny,12,5
2160  AXES 1/12,10,Tbegin-1,Maxy,12,5
2170  FOR H=Tbegin+1 TO Tend-.1 STEP 1
2180   MOVE H,(Maxy-Miny)/100+Miny
2190   LABEL USING "M3D";H
2200  NEXT H
2210  GOTO 2360
2220 IF Time>30 THEN 2300
2230  AXES 1/2,10,Tbegin-2,Miny,2,5
2240  AXES 1/2,10,Tbegin-2,Maxy,2,5
2250 FOR H=Tbegin+1 TO Tend-.1 STEP 2
2260  MOVE H,(Maxy-Miny)/100+Miny
2270  LABEL USING "M3D";H
2280 NEXT H
2290 GOTO 2360
2300  AXES 2,2,Tbegin-10,Miny,5,5
2310  AXES 2,2,Tbegin-10,Maxy,5,5
2320 FOR H=Tbegin+10 TO Tend-1 STEP 10
2330  MOVE H,(Maxy-Miny)/100+Miny
2340  LABEL USING "M3D";H
2350 NEXT H
2360 MOVE Time/2+Tbegin,(Maxy-Miny)/25+Miny
2370 LABEL USING "K";"TIME(J-DAY)"
2380  GOTO Repeat
2390 New_scale: ! THIS SUBPROGRAM PUTS SCALE ON RIGHT-SIDE OF PLOT
2400  SCALE Tbegin,Tend,Minr,Maxr
2410  CSIZE 2.5
2420  LORG 8
2430  IF Maxr-Minr>.1 THEN 2500
2440  AXES 1,.002,Tend,Minr-.01,5,5
2450  FOR V=Minr+.002 TO Maxr-.001 STEP .002
2460   MOVE Tend,V
2470   LABEL USING "MD.DDD,X";V
2480  NEXT V
2490  GOTO Blabel
2500  IF Maxr-Minr>1 THEN 2570
2510  AXES 1,.1,Tend,Minr-.5,5,5
2520  FOR V=Minr+.1 TO Maxr-.01 STEP .1
2530   MOVE Tend,V
2540   LABEL USING "MD.DD,X";V
2550  NEXT V
2560  GOTO Blabel
2570  IF Maxr-Minr>10 THEN 2640
2580  AXES 1,.2,Tend,Minr-1,5,5
2590  FOR V=Minr+1 TO Maxr-.1 STEP 1
2600   MOVE Tend,V
2610   LABEL USING "M3D.D,X";V
2620  NEXT V
2630  GOTO Blabel
2640  IF Maxr-Minr>100 THEN 2710
2650  AXES 1,2,Tend,Minr-10,5,5
2660  FOR V=Minr+10 TO Maxr-1 STEP 10
2670   MOVE Tend,V
2680   LABEL USING "M3D,X";V
2690  NEXT V
2700  GOTO Blabel
2710  AXES 1,10,Tend,Minr-50,5,5
2720  FOR V=Minr+50 TO Maxr-5 STEP 50
2730   MOVE Tend,V
2740   LABEL USING "M3D,X";V
2750  NEXT V
2760 Blabel:  MOVE Tend-Time/12,(Maxr-Minr)/2+Minr
2770  LDIR PI/2
2780  LORG 4
2790  CSIZE 3
```

```
2800  LABEL USING "K";Unitr$
2810  LDIR 0
2820  GOTO 1920
2830 New_label:  REM PRINTS NEW LABEL FOR NEXT MONSTER FILE
2840  SCALE Tbegin,Tend,Miny,Maxy
2850  Q=Q+1
2860  IF UPC$(P1$[1,1])<>"N" THEN 2880
2870  GRAPHICS
2880  LORG 2
2890  CSIZE 3
2900 Repeat:LORG 4
2910  Step=1
2920  PRINT PAGE,"INSTEAD OF SLOW, TEDIOUS PLOTTING OF EVERY POINT YOU HAVE THE
OPTION",LIN(1),"OF ONLY PLOTTING EVERY FIFTH,TENTH, OR TWENTIETH POINT.",LIN(1)
2930  PRINT "WHICH OPTION DO YOU WANT? 1-EVERY POINT; 2-EVERY FIFTH; 3-EVERY TEN
TH;",LIN(1),"OR 4-EVERY TWENTIETH?";
2940  INPUT "",Freq
2950  INPUT "WHICH PEN DO YOU WANT FOR PLOTTING. 1,2,3, OR 4?",P
2960  PEN P
2970  IF Freq=1 THEN Step=1
2980  IF Freq=2 THEN Step=5
2990  IF Freq=3 THEN Step=10
3000  IF Freq=4 THEN Step=20
3010 FOR I=1 TO Tscan STEP Step
3020  Days=VAL(Time$(I)[4,5])
3030  Mo=VAL(Time$(I)[1,2])
3040  Hrs=VAL(Time$(I)[7,8])
3050  Min=VAL(Time$(I)[10,11])
3051  Sec=VAL(Time$(I)[13,14])
3060  ON Mo GOTO 3190,3180,3170,3160,3150,3140,3130,3120,3110,3100,3090,3080,307
0
3070  Days=Days+31
3080  Days=Days+30
3090  Days=Days+31
3100  Days=Days+30
3110  Days=Days+31
3120  Days=Days+31
3130  Days=Days+30
3140  Days=Days+31
3150  Days=Days+30
3160  Days=Days+31
3170  Days=Days+29
3180  Days=Days+31
3181  Min=Min+Sec/60
3190  Hrs=Hrs+Min/60
3200  Days=Days+Hrs/24
3210  Date=Days
3211  IF Date<Tbegin THEN 3260
3212  IF Date>Tend THEN 3270
3220  LINE TYPE 1
3230  CSIZE .5
3240  PLOT Date,Data(I,Chan-Group+1),1
3250  PENUP
3260 NEXT I
3270 CSIZE 3
3280 PRINT "AFTER CURVES ARE PLOTTED A FLASHING CURSOR WILL APPEAR",LIN(1),"AT T
HE LOWER LEFT HAND CORNER OF THE SCREEN. BY USING THE ARROWS"
3290 PRINT "IN THE DISPLAY SECTION OF THE CONSOLE THE CURSOR CAN BE MOVED TO",LI
N(1),"PRINTING LOCATION. THEN TYPE LABELS. WHEN FINISHED PRESS CONT.",LIN(1)
3300 PRINT "IF YOU HAVE THEORETICAL DATA TO PLOT YOU WILL NEED TO GIVE PROGRAM",
LIN(1),"TIME VALUES AND DATA VALUES AT EACH DISCRETE POINT.",LIN(1)
3310 INPUT "DO YOU HAVE THEORETICAL DATA TO PLOT, Y OR N?",T$
3320 IF UPC$(T$[1,1])<>"Y" THEN Letter
3330 INPUT "HOW MANY TIME POINTS DO YOU HAVE ON THEORETICAL CURVE?",Mscan
3340 FOR I=1 TO Mscan
3350  FIXED 0
```

```
3360  DISP "WHAT IS TIME(J-DAY) FOR TIME # ";I;
3370  INPUT Tyme(I)
3380 NEXT I
3390 FIXED 2
3400 FOR K=1 TO Mscan
3410  DISP "ENTER THEORETICAL VALUE ON J-DAY ";Tyme(K);
3420  INPUT Model(K)
3430 NEXT K
3440 FIXED 0
3450 INPUT "ARE THERE TWO DISTINCT SEGMENTS TO THEORETICAL CURVE,Y OR N?",S$
3460 IF UPC$(S$[1,1])="N" THEN Single
3470 INPUT "HOW MANY POINTS ARE IN THE FIRST SEGMENT",M
3480 CALL Spline2(M,Narg)
3490 FOR J=1 TO Narg
3500  PLOT Domain(J),Func(J)
3510 NEXT J
3520 P=Mscan-M+1
3530 FOR J=1 TO P
3540  Tyme(J)=Tyme(M+J-1)
3550  Model(J)=Model(M+J-1)
3560 NEXT J
3570 CALL Spline2(P,Narg)
3580 FOR J=1 TO Narg
3590  PLOT Domain(J),Func(J)
3600 NEXT J
3610 GOTO Letter
3620 Single: CALL Spline2(Mscan,Narg)
3630 FOR J=1 TO Narg
3640  PLOT Domain(J),Func(J)
3650 NEXT J
3660 Letter:IF UPC$(P1$[1,1])<>"Y" THEN GRAPHICS
3670 PEN P
3680 LETTER
3690 IF UPC$(P1$[1,1])="Y" THEN Question
3700 INPUT "DO YOU WANT TO ERASE OR CHANGE ANYTHING, Y OR N?",L$
3710 IF UPC$(L$[1,1])="N" THEN Question
3720 EXIT GRAPHICS
3730  PRINT "TO ERASE, RE-TYPE INCORRECT CHARACTERS OVER WHAT IS TO BE ERASED.",
LIN(1),"WHEN FINISHED MAKING ERASURES PRESS CONTINUE TO RE-ENABLE LETTERING"
3740  WAIT 9000
3750  GRAPHICS
3760 PEN -1
3770 LETTER
3780 GOTO Letter
3790 Question:PRINTER IS 16
3800  IF Different=1 THEN 3940
3810 INPUT "ARE MORE CURVES FROM THIS DATA TO GO ON PLOT LEFT-SIDE SCALING, Y OR
 N?",More$
3820 IF UPC$(More$[1,1])<>"Y" THEN 3870
3830 INPUT "WHAT IS NEXT CHANNEL NUMBER TO BE PLOTTED WITH LEFT-SIDE SCALE?"
,Chan
3840  P=P+1
3850  IF P>4 THEN P=1
3860 GOTO Repeat
3870  Different=1
3880  IF UPC$(Diff$[1,1])<>"Y" THEN 3980
3890  SCALE Tbegin,Tend,Minr,Maxr
3900  INPUT "WHICH CHANNEL NUMBER DO YOU WISH TO PLOT USING RIGHT-SIDE SCALE?"
,Chan
3910  P=P+1
3920  IF P>4 THEN P=1
3930  GOTO Repeat
3940  INPUT "ARE MORE CURVES FROM THIS DATA TO GO ON PLOT RIGHT-SIDE SCALE,Y OR
N?",More$
3950  IF UPC$(More$[1,1])<>"Y" THEN 3980
3960  INPUT "WHAT IS NEXT CHANNEL NUMBER TO BE PLOTTED WITH RIGHT-SIDE SCALE
```

```
?",Chan
3970  GOTO Repeat
3980  PRINT "YOU CAN EXPAND THE PLOTTING CAPABILITIES BY READING IN MORE DATA TO
",LIN(1),"EITHER EXTEND PLOT FOR OTHER TIMES OR MOVE TO OTHER GROUPS OF"
3990  PRINT "CHANNELS.",LIN(1)
4000  INPUT "DOES MORE DATA NEED TO BE READ IN TO ADD TO THIS PLOT? Y OR N",Repe
at$
4010  IF UPC$(Repeat$[1,1])<>"N" THEN Begin
4020 LOCATE 20,115,25,98
4030 PRINTER IS 0
4040  PRINT PAGE,LIN(12)
4050 DUMP GRAPHICS
4060 INPUT "DO YOU WANT ANOTHER COPY OF THIS PLOT, Y OR N?",Copy$
4070 IF UPC$(Copy$[1,1])<>"N" THEN 4040
4080  PLOTTER IS 13,"GRAPHICS"
4090 INPUT "DO YOU WANT SOME OTHER PLOTS FROM THIS DATA?",G$
4100  Different=0
4110  Diff$=""
4120 IF UPC$(G$[1,1])<>"N" THEN Plot
4130 Exit: INPUT "DO YOU WANT TO READ IN MORE DATA FROM DISK/TAPE, Y OR N?",Y$
4140 IF UPC$(Y$[1,1])<>"N" THEN 80
4141  PRINTER IS 16
4142  PRINT PAGE,LIN(5)
4150 PRINT "TERMINATING PROGRAM AT YOUR REQUEST. OVER AND OUT!!!"
4160 END
4170 SUB Spline(N,Narg,X(*),Y(*),Domain(*),Func(*),Deriv(*),Int,Eps)
4180 !
4190 ! ******************************************************************
4200 ! *** SPLINE FIT FOR FUNCTION VALUES, INTEGRATION AND DIFFERENTIATION.
4210 ! ******************************************************************
4220 !
4230 !
4240 ! *** BAD DATA CHECK.
4250 Baddta=(N<=0) OR (Eps<=0)
4260 IF Baddta=0 THEN 4360
4270 ! *** PRINT ERROR MESSAGE AND PAUSE.
4280 ! *** USER MAY CORRECT DATA AND CONTINUE.
4290 PRINT LIN(2),"ERROR IN SUBPROGRAM Spline."
4300 PRINT "N=";N,"Eps=";Eps,LIN(2)
4310 PAUSE
4320 GOTO 4250
4330 !
4340 !
4350 ! *** BEGIN SUBPROGRAM.
4360 OPTION BASE 1
4370 DIM S(N),G(N-1),Work(N-1)
4380 FOR I=2 TO N-1
4390    Xi=X(I)
4400    Xim1=X(I-1)
4410    Xip1=X(I+1)
4420    Yi=Y(I)
4430    Yim1=Y(I-1)
4440    Yip1=Y(I+1)
4450    X=Xi-Xim1
4460    H=Xip1-Xim1
4470    Work(I)=.5*X/H
4480    T=((Yip1-Yi)/(Xip1-Xi)-(Yi-Yim1)/X)/H
4490    S(I)=2*T
4500    G(I)=3*T
4510 NEXT I
4520 S(1)=S(N)=0
4530 ! *** W IS THE RELAXATION FACTOR FOR SUCCESSIVE OVER-RELAXATION.
4540 W=8-4*SQR(3)
4550 U=0
4560 FOR I=2 TO N-1
4570    T=W*(-S(I)-Work(I)*S(I-1)-(.5-Work(I))*S(I+1)+G(I))
```

```
4580    H=ABS(T)
4590    IF H>U THEN U=H
4600    S(I)=S(I)+T
4610 NEXT I
4620 IF U>=Eps THEN 4550
4630 FOR I=1 TO N-1
4640    G(I)=(S(I+1)-S(I))/(X(I+1)-X(I))
4650 NEXT I
4660 IF Narg=0 THEN 4980
4670 !
4680 !
4690 ! *** CALCULATE FUNCTION VALUES AND DERIVATIVES.
4700 FOR J=1 TO Narg
4710 Corrector:    I=1
4720    T=Domain(J)
4730    IF T>=X(1) THEN 4810
4740    PRINT LIN(2),"ERROR IN SUBPROGRAM Spline."
4750    PRINT "ARGUMENT OUT OF BOUNDS."
4760    PRINT "X(1)=";X(1),"X(N)=";X(N),"Domain(";J;")=";Domain(J),LIN(2)
4770    PAUSE
4780    GOTO 4710
4790    !
4800    !
4810    I=I+1
4820    IF I>N THEN 4740
4830    IF T>X(I) THEN 4810
4840    I=I-1
4850    H=Domain(J)-X(I)
4860    T=Domain(J)-X(I+1)
4870    X=H*T
4880    S=S(I)+H*G(I)
4890    Z=1/6
4900    U=Z*(S(I)+S(I+1)+S)
4910    W=(Y(I+1)-Y(I))/(X(I+1)-X(I))
4920    Func(J)=W*H+Y(I)+X*U
4930    Deriv(J)=W+(H+T)*U+Z*X*G(I)
4940 NEXT J
4950 !
4960 !
4970 ! *** CALCULATE INTEGRAL FROM X(1) TO X(N).
4980 Int=0
4990 FOR I=1 TO N-1
5000    H=X(I+1)-X(I)
5010    Int=Int+.5*H*(Y(I)+Y(I+1))-1/24*H^3*(S(I)+S(I+1))
5020 NEXT I
5030 SUBEND
5040 REM  ****************************************  Spline2  *******
5050 SUB Spline2(Q,Narg)
5060 OPTION BASE 1
5070 COM Table$(20),SHORT Table2(20,120),INTEGER Scans
5080 COM Time$(2500),SHORT Data(2500,10),INTEGER Tscan,Group,REAL Tbegin,Tend
5090 COM Tyme(15),Model(15),Domain(20),Func(20),P1$,Repeat$
5100 DIM Deriv(Narg)
5110 FOR I=1 TO Narg
5120  Domain(I)=I*(Tyme(Q)-Tyme(1))/Narg+Tyme(1)
5130 NEXT I
5140 CALL Spline(Q,Narg,Tyme(*),Model(*),Domain(*),Func(*),Deriv(*),Int,1E-6)
5150 SUBEND
```

APPENDIX E

PLOTE Program

```
10   REM THIS IS PROGRAM TO READ DATA TAPE/FLOPPY AND PLOT  5/09/80 *****
20    REM  ****************  REVISION E  *****************
30   OPTION BASE 1
40   COM Table$(20),SHORT Table2(20,120),INTEGER Scans
50   COM Time$(200),SHORT Data(200,120),INTEGER Tscan,Chan(5),Tim(5)
60   COM Isoth(5,15),A(15),P(15),Depth(5,15),Temp(5,15),Number(5),Pl$
70   COM Domain(40),Func(40)
80   COM Radius(15),Mod(15),Xp(15),Yp(15),Xi(15),Yi(15),Ch(5,15)
90   Narg=40
100   P=1
110  PRINTER IS 16
120  INPUT "IS DATA GOING TO BE READ IN FROM FLOPPY DISKS?",A$
130  IF UPC$(A$[1,1])="N" THEN 150
140  MASS STORAGE IS ":F8"
150  INPUT "WHAT IS FIRST DATA FILE TO BE READ?",F$[1,6]
160  ASSIGN #1 TO F$[1,6]
170  READ #1;Table$(*),Table2(*),Scans
180  FOR I=1 TO Scans
190   Time$(I)=Table$(I)
200   FOR J=1 TO 120
210    Data(I,J)=Table2(I,J)
220   NEXT J
230  NEXT I
240  Tscan=Scans
250 Read: Left=200-Tscan
260  FIXED 0
270  PRINT "ROOM FOR ";Left;" MORE SCANS."
280  INPUT "DO YOU WANT TO READ IN MORE DATA, Y OR N?",B$
290  IF UPC$(B$[1,1])="N" THEN Plot
300  INPUT "WHAT IS NEXT DATA FILE TO BE READ?",F$[1,6]
310  ASSIGN #1 TO F$[1,6]
320  READ #1;Table$(*),Table2(*),Scans
330  FOR I=1 TO Scans
340   Time$(I+Tscan)=Table$(I)
350   FOR J=1 TO 120
360    Data(I+Tscan,J)=Table2(I,J)
370   NEXT J
380  NEXT I
390  Tscan=Tscan+Scans
400  GOTO Read
410 Plot: REM THIS BEGINS PLOTTING ROUTINE BY SELECTING TYPE OF PLOT DESIRED
420  PRINTER IS 16
430  PRINT PAGE,"THE FOLLOWING OPTIONS ARE AVAILABLE FOR PLOTTING:"
440  PRINT LIN(1),"OPTION 1:  TEMPERATURE AS FUNCTION OF RADIUS FOR SEVERAL TIME
S"
450  PRINT "OPTION 2:  ISOTHERMS FOR A GIVEN TIME"
460  INPUT "PICK YOUR OPTION BY PRESSING 1 OR 2 THEN CONT",Type
470  INPUT "ARE YOU USING THE 9872A PLOTTER, Y OR N?",Pl$
480  ON Type GOTO Profile,Isotherm
490  INPUT "DO YOU WANT SOME MORE PLOTS FROM THIS DATA?",G$
500  IF UPC$(G$[1,1])<>"N" THEN Plot
510  GOTO Exit
520 Profile:  REM SET UP TEMPERATURE VS RADIUS PROFILES FOR SELECT CHANNELS
530  INPUT "HOW MANY DIFFERENT RADII ARE THERE?",Holes
```

```
540  INPUT "ENTER NUMBER OF TEMPERATURE SCANS (DIFFERENT TIMES) FOR PLOTTING",Ns
can
550  FOR I=1 TO Holes
560   FIXED 0
570   DISP "RADIUS TO HOLE # ";I;"IN METERS";
580   INPUT Radius(I)
590   DISP "CHANNEL NUMBER OF THERMOCOUPLE IN HOLE # ";I;
600   INPUT Chan(I)
610  NEXT I
620  FOR I=1 TO Nscan
630   DISP "WHAT IS THE SCAN NUMBER FOR TEMPERATURE SCAN # ";I;
640   INPUT Tim(I)
650  NEXT I
660  LOCATE 5,110,0,100
670  INPUT "DO YOU WANT AUTO SCALING OF Y-AXIS?",As$
680  IF UPC$(As$[1,1])<>"Y" THEN Noscale
690  Maxy=10
700  FOR K=1 TO Nscan
710   Chan=Chan(1)
720   Tym=Tim(K)
730   IF Data(Tym,Chan)>Maxy THEN Maxy=Data(Tym,Chan)
740  NEXT K
750  Maxy=10*INT(Maxy/10)+10
760  GOTO 780
770 Noscale:  INPUT "ENTER MAXIMUM VALUE DESIRED FOR Y-AXIS IN MULTIPLES OF 10",
Maxy
780  IF UPC$(P1$[1,1])<>"Y" THEN 830
790  PLOTTER IS 7,5,"9872A"
791   PRINTER IS 7,5
792   PRINT "VS5"
800   PEN P
810   LOCATE 10,110,5,95
820  GOTO 860
830  PLOTTER IS 13,"GRAPHICS"
840  GRAPHICS
850  LOCATE 5,110,0,100
860  SCALE 0,INT(2*Radius(Holes))/2+.5,0,Maxy
870  IF Radius(Holes)>1 THEN AXES .1,5,0,0,5,2
880  IF Radius(Holes)<=1 THEN AXES .1,5,0,0,2,2
890  IF Radius(Holes)>1 THEN AXES .1,5,INT(2*Radius(Holes))/2+.5,Maxy,5,2
900  IF Radius(Holes)<=1 THEN AXES .1,5,INT(2*Radius(Holes))/2+.5,Maxy,2,2
910  CSIZE 3
920  LORG 2
930  IF Maxy>100 THEN 990
940  FOR V=10 TO Maxy-1 STEP 10
950   MOVE 0,V
960   LABEL USING "M3D,X";V
970  NEXT V
980  GOTO 1030
990  FOR V=50 TO Maxy-5 STEP 50
1000  MOVE 0,V
1010  LABEL USING "M3D,X";V
1020 NEXT V
1030 MOVE Radius(Holes)/8,Maxy/2
1040 CSIZE 3
1050 LABEL USING "K";"T(C)"
1060 LORG 4
1070 CSIZE 3
1080 IF Radius(Holes)>1 THEN 1140
1090 FOR H=.2 TO .8 STEP .2
1100  MOVE H,Maxy/60
1110  LABEL USING "MD.DD";H
1120 NEXT H
1130 GOTO 1180
1140 FOR H=.5 TO Radius(Holes) STEP .5
1150  MOVE H,Maxy/60
```

```
1160  LABEL USING "MD.DD";H
1170 NEXT H
1180 CSIZE 4
1190  IF Radius(Holes)>1 THEN 1220
1200  MOVE .5,Maxy/15
1210  GOTO 1230
1220 MOVE Radius(Holes)/2,Maxy/15
1230 LABEL USING "K";"RADIUS(M)"
1240 LORG 2
1250 CSIZE 3
1260  INPUT "WHICH PEN DO YOU WANT TO USE FOR DATA POINTS. 1,2,3,OR 4?",P
1280 MOVE Radius(Holes)/20,.96*Maxy
1290 LABEL USING "K,K,K,K,K,K,K,K";"DATA CH # ";Chan(1);", ";Chan(2);", ";Chan(3
);", ";Chan(4)
1320 LORG 5
1330 N=1
1340 FOR J=1 TO Nscan
1341  PEN P
1350  Tim=Tim(J)
1360  FOR I=1 TO Holes
1370   CSIZE 2
1380   Chan=Chan(I)
1390   MOVE Radius(I),Data(Tim,Chan)
1400   LABEL USING "K";"X"
1410  NEXT I
1420  PRINTER IS 16
1430  CSIZE 3
1440  INPUT "DO YOU HAVE THEORETICAL PROFILE TO PLOT, Y OR N?",Th$
1450  PRINT LIN(2),"AFTER THEORETICAL CURVE IS PLOTTED A FLASHING CURSOR WILL AP
PEAR",LIN(1),"AT THE LOWER LEFT HAND CORNER OF THE SCREEN. BY USING THE ARROWS"
1460  PRINT "IN THE DISPLAY SECTION OF THE CONSOLE THE CURSOR CAN BE MOVED TO",L
IN(1),"PRINTING LOCATION. THEN TYPE LABELS. WHEN FINISHED PRESS CONT."
1470  IF UPC$(Th$[1,1])<>"Y" THEN Again
1480  FIXED 2
1490  FOR I=1 TO Holes
1500   DISP "WHAT IS TEMPERATURE VALUE AT TIME ";Time$(Tim);" AND AT RADIUS ";Ra
dius(I);
1510   INPUT Mod(I)
1520  NEXT I
1521  INPUT "WHICH PEN DO YOU WANT TO USE FOR THEORETICAL CURVES?",Pt
1522  PEN Pt
1530  CALL Spline1(Holes,Narg)
1540  FOR I=1 TO Narg
1550   PLOT Domain(I),Func(I)
1560  NEXT I
1561  PENUP
1570 Again:  NEXT J
1580 Pletter:PEN P
1590 LETTER
1600 IF UPC$(P1$[1,1])="Y" THEN Quest
1610 INPUT "DO YOU WANT TO ERASE OR CHANGE ANYTHING, Y OR N?",L$
1620 IF UPC$(L$[1,1])="N" THEN 1660
1630 PEN -1
1640 LETTER
1650 GOTO Pletter
1660 LOCATE 5,110,0,100
1661  PEN 0
1670 PRINTER IS 0
1680 PRINT PAGE,LIN(12)
1690 DUMP GRAPHICS
1700 INPUT "DO YOU WANT ANOTHER COPY OF THIS PLOT, Y OR N?",Copy$
1710 IF UPC$(Copy$[1,1])="N" THEN Quest
1720 GOTO 1680
1730 Quest: PRINTER IS 16
1740 PLOTTER IS 13,"GRAPHICS"
1750 GOTO 490
```

```
1760 Isotherm: REM SET UP FOR PLOTTING ISOTHERMS AT SELECTED TIMES
1770 PRINTER IS 16
1780 Que=1
1790 INPUT "WHAT IS THE SCAN NUMBER FOR THE ISOTHERM TO BE PLOTTED?",Tim
1800 Caption: Holes=3
1810 Number(1)=6
1820 Number(2)=15
1830 Number(3)=9
1840 Radius(1)=.0635
1850 Radius(2)=.3320! WHATEVER RADIUS TO FIRST HOLE IS *********
1860 Radius(3)=.5827! WHATEVER RADIUS TO SECOND HOLE IS ********
1870 IF Que<>1 THEN 2030
1880 DATA 16.4,17.1,18.29,18.75,19.2,20.5
1890 DATA 68,5,17,23,26,68
1900 DATA 16.6,17.22,17.83,18.29,18.44,18.59,18.75,18.90,19.05,19.2,19.35,19.66,
19.96,20.57,20.9
1910 DATA 68,53,54,55,56,57,61,62,63,64,65,66,67,68,44
1920 DATA 16.31,17.22,18.14,18.44,18.75,19.05,19.35,19.66,21.18
1930 DATA 44,45,46,47,48,49,50,51,52
1940 IF UPC$(G$[1,1])="Y" THEN 2030
1950 FOR I=1 TO Holes
1960  FOR J=1 TO Number(I)
1970   READ Depth(I,J)
1980  NEXT J
1990  FOR J=1 TO Number(I)
2000   READ Ch(I,J)
2010  NEXT J
2020 NEXT I
2030 FOR I=1 TO Holes
2040  FOR J=1 TO Number(I)
2050   K=Ch(I,J)
2060   Temp(I,J)=Data(Tim,K)
2070  NEXT J
2080 NEXT I
2090 Que=Que+1
2100 FIXED 2
2110 INPUT "DO YOU WANT THE TEMPERATURES PRINTED ON THE PLOT, Y OR N?",Pr$
2120 INPUT "DO YOU WANT THE ISOTHERMS PLOTTED, Y OR N?",Is$
2130 Maxim:IF UPC$(Pl$[1,1])<>"Y" THEN 2160
2140 PLOTTER IS 7,5,"9872A"
2141  PEN 1
2142  LOCATE 15,95,10,130
2143  PRINTER IS 7,5
2144  PRINT "VS5"
2150 GOTO 2190
2160 PLOTTER IS 13,"GRAPHICS"
2170 GRAPHICS
2180 Locate: LOCATE 25,89,2,98
2190 SCALE 0,2,17.5,20.5
2200 AXES .1,.1,0,17.5,5,5,3
2210 ! AXES .1,.5,-.2,16.3,2,2,3
2220 ! AXES .1,.5,-.2,21.5,2,2,3
2230 AXES .1,.1,2,20.5,5,5,3
2240 CSIZE 2.5
2250 LORG 5
2260 MOVE .023,18.75
2270 LDIR PI/2
2280 LABEL USING "K";"HEATER"
2290 LDIR 0
2300 MOVE 0,18.18
2310 DRAW .051,18.18
2320 DRAW .051,19.35
2330 DRAW 0,19.35
2340 CSIZE 3
2350 LORG 8
2360 FOR V=18 TO 20 STEP .5
```

```
2370  MOVE 0,V
2380  LABEL USING "M2D.DX";V
2390 NEXT V
2400 MOVE -.33,19
2410 CSIZE 4
2420 LORG 5
2430 LDIR PI/2
2440 LABEL USING "K";"DEPTH(M)"
2450 LDIR 0
2460 LORG 4
2470 CSIZE 2.5
2480 FOR H=.5 TO 1.5 STEP .5
2490  MOVE H,17.6
2500  LABEL USING "MD.D";H
2510 NEXT H
2520 CSIZE 3
2530 MOVE 1.6,17.75
2540 LABEL USING "K";"RADIUS(M)"
2550 LORG 3
2560 CSIZE 3
2570 MOVE 1,20.35
2580 LABEL USING "K,K";"TIME ";Time$(Tim)[1,11]
2581  PENUP
2590 Cycle:FIXED 0
2600 CSIZE 2.5
2601  GOTO 2790  ! SKIPS DRAWING LINES FOR TC PLACEMENT ******************
2610  INPUT "WHICH PEN DO YOU WANT TO USE TO DRAW HOLES. 1,2,3,OR 4?",P
2620  PEN P
2621  MOVE Radius(1),17.5
2622  DRAW Radius(1),19.35
2623  DRAW .0381,19.35
2624  DRAW .0381,19.873
2625  DRAW 0,19.873
2630 FOR K=2 TO Holes
2640  MOVE Radius(K),17.5
2650  DRAW Radius(K),20.5
2660  FOR I=1 TO Number(K)
2670   IF (K=1) AND (I=1) THEN 2770
2680   IF (K=1) AND (I=6) THEN 2770
2690   IF (K=2) AND (I=1) THEN 2770
2700   IF (K=2) AND (I=15) THEN 2770
2710   MOVE Radius(K),Depth(K,I)
2720   LORG 2
2730   IF UPC$(Pr$[1,1])="Y" THEN 2760
2740   LABEL USING "K";"-"
2750   GOTO 2770
2760   LABEL USING "K,3D.D,K";"-";Temp(K,I);" C"
2770  NEXT I
2780 NEXT K
2781  PENUP
2790 IF UPC$(Is$[1,1])="N" THEN Dump
2791  INPUT "WHICH PEN DO YOU WANT TO USE FOR DRAWING ISOTHERMS?",P
2800  FOR J=1 TO 3
2810 Least: X=X2=X3=X4=Xy=X2y=Y=0
2820   IF J<>1 THEN 2910
2830   X=Temp(1,3)+Temp(2,4)+(Temp(3,3)+Temp(3,4))/2
2840   X2=Temp(1,3)^2+Temp(2,4)^2+((Temp(3,3)+Temp(3,4))/2)^2
2850   X3=Temp(1,3)^3+Temp(2,4)^3+((Temp(3,3)+Temp(3,4))/2)^3
2860   X4=Temp(1,3)^4+Temp(2,4)^4+((Temp(3,3)+Temp(3,4))/2)^4
2870   Y=Radius(1)+Radius(2)+Radius(3)
2880   Xy=Radius(1)*Temp(1,3)+Radius(2)*Temp(2,4)+Radius(3)*(Temp(3,3)+Temp(3,4)
)/2
2890   X2y=Radius(1)*Temp(1,3)^2+Radius(2)*Temp(2,4)^2+Radius(3)*((Temp(3,3)+Tem
p(3,4))/2)^2
2900   GOTO 3070
2910   IF J<>2 THEN 3000
```

```
2920   X=Temp(1,4)+Temp(2,7)+Temp(3,5)
2930   X2=Temp(1,4)^2+Temp(2,7)^2+Temp(3,5)^2
2940   X3=Temp(1,4)^3+Temp(2,7)^3+Temp(3,5)^3
2950   X4=Temp(1,4)^4+Temp(2,7)^4+Temp(3,5)^4
2960   Y=Radius(1)+Radius(2)+Radius(3)
2970   Xy=Radius(1)*Temp(1,4)+Radius(2)*Temp(2,7)+Radius(3)*Temp(3,5)
2980   X2y=Radius(1)*Temp(1,4)^2+Radius(2)*Temp(2,7)^2+Radius(3)*Temp(3,5)^2
2990   GOTO 3070
3000   X=Temp(1,5)+Temp(2,10)+Temp(3,8)
3010   X2=Temp(1,5)^2+Temp(2,10)^2+Temp(3,8)^2
3020   X3=Temp(1,5)^3+Temp(2,10)^3+Temp(3,8)^3
3030   X4=Temp(1,5)^4+Temp(2,10)^4+Temp(3,8)^4
3040   Y=Radius(1)+Radius(2)+Radius(3)
3050   Xy=Radius(1)*Temp(1,5)+Radius(2)*Temp(2,10)+Radius(3)*Temp(3,8)
3060   X2y=Radius(1)*Temp(1,5)^2+Radius(2)*Temp(2,10)^2+Radius(3)*Temp(3,8)^2
3070   Num=3*(X2*X2y-Xy*X3)-X*(X*X2y-Xy*X2)+Y*(X*X3-X2*X2)
3080   Den=3*(X2*X4-X3*X3)-X*(X*X4-X3*X2)+X2*(X*X3-X2*X2)
3090   A2(J)=Num/Den
3100   N=Y-A2(J)*X2
3110   M=Xy-A2(J)*X3
3120   A1(J)=(N/3-M/X)/(X/3-X2/X)
3130   A0(J)=M/X-A1(J)*X2/X
3140  NEXT J
3150   A(1)=60
3160 A(2)=70
3170 A(3)=80
3180 A(4)=90
3190 A(5)=100
3200  PRINTER IS 16
3210  PRINT LIN(3),"I HAVE DEFAULT VALUES FOR ISOTHERM TEMPERATURES WHICH WILL B
E",LIN(1),"DRAWN.  CURRENT VALUES ARE 60,70,80,90,AND 100 DEGREES C.",LIN(1)
3220  PRINT "YOU CAN CHANGE THEM IF YOU DESIRE. HOWEVER, NOTE THAT THE VALUES",L
IN(1),"MUST BE AT LEAST 20 DEG.",LIN(3)
3240  INPUT "DO YOU WANT TO PUT IN YOUR OWN ISOTHERM VALUES, Y OR N?",Own$
3250 P(1)=P(2)=P(3)=P(4)=P(5)=0
3260  IF UPC$(Own$[1,1])="N" THEN 3311
3270  INPUT "WHAT IS THE FIRST (LOWEST TEMP) ISOTHERM VALUE YOU WANT?",A(1)
3280  INPUT "WHAT IS THE SECOND ISOTHERM VALUE? (YOU GET A TOTAL OF FIVE)?",A(2)
3290  INPUT "WHAT IS THE THIRD ISOTHERM VALUE?",A(3)
3300  INPUT "WHAT IS THE FOURTH ISOTHERM VALUE?",A(4)
3310  INPUT "WHAT IS THE FIFTH (HIGHEST) ISOTHERM VALUE?",A(5)
3311  PEN P
3320  MOVE 1.6,20.1
3330  LABEL USING "M3D,K";A(1);" C"
3340  MOVE 1.6,20
3350  LABEL USING "M3D,K";A(2);" C"
3360  MOVE 1.6,19.9
3370  LABEL USING "M3D,K";A(3);" C"
3380  MOVE 1.6,19.8
3390  LABEL USING "M3D,K";A(4);" C"
3400  MOVE 1.6,19.7
3410  LABEL USING "M3D,K";A(5);" C"
3420 FOR K=1 TO Holes
3430  M=0
3440  FOR I=1 TO Number(K)
3450   IF Temp(K,I)=0 THEN 3490
3460   M=M+1
3470   Yp(M)=Temp(K,I)
3480   Xp(M)=Depth(K,I)
3490  NEXT I
3500  PRINTER IS 0
3510  CALL Spline3(M,Narg)
3520  PRINTER IS 16
3530  FIXED 1
3540  FOR L=1 TO 5
3550   FOR I=1 TO Narg-1
```

```
3560    IF Func(I+1)<Func(I) THEN 3620
3570    IF Func(I)<A(L) THEN Onward
3580    Isoth(L,K)=Domain(I)
3590    P(L)=K
3600    GOTO 3620
3610 Onward:     NEXT I
3620   FOR I=1 TO Narg-1
3630   IF Func(I+1)>Func(I) THEN Reward
3640   IF Func(I)<=A(L) THEN Reward
3650   Isoth(L,2*Holes-K+1)=Domain(I)
3660 Reward:    NEXT I
3670  NEXT L
3680  Yi(K)=Radius(K)
3690 NEXT K
3700 FOR L=1 TO 5
3710  N=2*P(L)
3720  FOR I=1 TO P(L)
3730   Xi(I)=Isoth(L,I)
3740   Xi(N-I+1)=Isoth(L,2*Holes-I+1)
3750   Yi(N-I+1)=Yi(I)
3760  NEXT I
3770  IF N>=4 THEN 3870
3780  N=5
3790  Xi(5)=Xi(2)
3800  Xi(2)=18.29
3810  Xi(3)=18.75
3820  Xi(4)=19.20
3830  Yi(5)=Yi(2)
3840  Yi(2)=A0(1)+A1(1)*A(L)+A2(1)*A(L)^2
3850  Yi(3)=A0(2)+A1(2)*A(L)+A2(2)*A(L)^2
3860  Yi(4)=A0(3)+A1(3)*A(L)+A2(3)*A(L)^2
3870  CALL Spline4(N,Narg)
3880  MOVE Func(1),Domain(I)
3890  FOR I=1 TO Narg
3900   PLOT Func(I),Domain(I)
3910  NEXT I
3920  PENUP
3930 NEXT L
3931  PEN 0
3940 Dump:PRINTER IS 0
3960 IF UPC$(P1$[1,1])="Y" THEN Frage
3965  PRINT PAGE,LIN(8)
3970 DUMP GRAPHICS
3980 INPUT "DO YOU WANT ANOTHER COPY OF THIS PLOT, Y OR N?",Copy$
3990 IF UPC$(Copy$[1,1])="N" THEN Frage
4000 PRINT PAGE,LIN(12)
4010 DUMP GRAPHICS
4020 GOTO 3980
4030 Frage: PRINTER IS 16
4040 PLOTTER IS 13,"GRAPHICS"
4050 GOTO 490
4060 Exit: INPUT "DO YOU WANT TO READ IN MORE DATA FROM DISK/TAPE, Y OR N?",Y$
4070 IF UPC$(Y$[1,1])<>"N" THEN 120
4080 PRINT "TERMINATING PROGRAM AT YOUR REQUEST. OVER AND OUT!!!"
4090 END
4100 REM   ****************************************  Spline4  ********
4110 SUB Spline4(N,Narg)
4120 OPTION BASE 1
4130 COM Table$(20),SHORT Table2(20,120),INTEGER Scans
4140 COM Time$(200),SHORT Data(200,120),INTEGER Tscan,Chan(5),Tim(5)
4150 COM Isoth(5,15),A(15),P(15),Depth(5,15),Temp(5,15),Number(5),P1$
4160 COM Domain(40),Func(40)
4170 COM Radius(15),Mod(15),Xp(15),Yp(15),Xi(15),Yi(15),Ch(5,15)
4180 DIM Deriv(Narg)
4190 FOR I=1 TO Narg
4200  Domain(I)=I*(Xi(N)-Xi(1))/Narg+Xi(1)
```

```
4210 NEXT I
4220 PRINTER IS 0
4230 CALL Spline(N,Narg,Xi(*),Yi(*),Domain(*),Func(*),Deriv(*),Int,1E-6)
4240 SUBEND
4250 REM  ************************************  Spline3  ************
4260 SUB Spline3(M,Narg)
4270 OPTION BASE 1
4280 COM Table$(20),SHORT Table2(20,120),INTEGER Scans
4290 COM Time$(200),SHORT Data(200,120),INTEGER Tscan,Chan(5),Tim(5)
4300 COM Isoth(5,15),A(15),P(15),Depth(5,15),Temp(5,15),Number(5),P1$
4310 COM Domain(40),Func(40)
4320 COM Radius(15),Mod(15),Xp(15),Yp(15),Xi(15),Yi(15),Ch(5,15)
4330 DIM Deriv(Narg)
4340 FOR I=1 TO Narg
4350  Domain(I)=I*(Xp(M)-Xp(1))/Narg+Xp(1)
4360 NEXT I
4370 PRINTER IS 0
4380 CALL Spline(M,Narg,Xp(*),Yp(*),Domain(*),Func(*),Deriv(*),Int,1E-6)
4390 SUBEND
4400 REM  ************************************  Spline1  **********
4410 SUB Spline1(Holes,Narg)
4420 OPTION BASE 1
4430 COM Table$(20),SHORT Table2(20,120),INTEGER Scans
4440 COM Time$(200),SHORT Data(200,120),INTEGER Tscan,Chan(5),Tim(5)
4450 COM Isoth(5,15),A(15),P(15),Depth(5,15),Temp(5,15),Number(5),P1$
4460 COM Domain(40),Func(40)
4470 COM Radius(15),Mod(15),Xp(15),Yp(15),Xi(15),Yi(15),Ch(5,15)
4480 DIM Deriv(Narg)
4490 FOR I=1 TO Narg
4500  Domain(I)=I*(Radius(Holes)-Radius(1))/Narg+Radius(1)
4510 NEXT I
4520 PRINTER IS 0
4530 CALL Spline(Holes,Narg,Radius(*),Mod(*),Domain(*),Func(*),Deriv(*),Int,1E-6
)
4540 SUBEND
4550 SUB Spline(N,Narg,X(*),Y(*),Domain(*),Func(*),Deriv(*),Int,Eps)
4560 !
4570 ! ******************************************************************
4580 ! *** SPLINE FIT FOR FUNCTION VALUES, INTEGRATION AND DIFFERENTIATION.
4590 ! ******************************************************************
4600 !
4610 !
4620 ! *** BAD DATA CHECK.
4630 Baddta=(N<=0) OR (Eps<=0)
4640 IF Baddta=0 THEN 4740
4650 ! *** PRINT ERROR MESSAGE AND PAUSE.
4660 ! *** USER MAY CORRECT DATA AND CONTINUE.
4670 PRINT LIN(2),"ERROR IN SUBPROGRAM Spline."
4680 PRINT "N=";N,"Eps=";Eps,LIN(2)
4690 PAUSE
4700 GOTO 4630
4710 !
4720 !
4730 ! *** BEGIN SUBPROGRAM.
4740 OPTION BASE 1
4750 DIM S(N),G(N-1),Work(N-1)
4760 FOR I=2 TO N-1
4770    Xi=X(I)
4780    Xim1=X(I-1)
4790    Xip1=X(I+1)
4800    Yi=Y(I)
4810    Yim1=Y(I-1)
4820    Yip1=Y(I+1)
4830    X=Xi-Xim1
4840    H=Xip1-Xim1
4850    Work(I)=.5*X/H
```

```
4860    T=((Yip1-Yi)/(Xip1-Xi)-(Yi-Yim1)/X)/H
4870    S(I)=2*T
4880    G(I)=3*T
4890 NEXT I
4900 S(1)=S(N)=0
4910 ! *** W IS THE RELAXATION FACTOR FOR SUCCESSIVE OVER-RELAXATION.
4920 W=8-4*SQR(3)
4930 U=0
4940 FOR I=2 TO N-1
4950    T=W*(-S(I)-Work(I)*S(I-1)-(.5-Work(I))*S(I+1)+G(I))
4960    H=ABS(T)
4970    IF H>U THEN U=H
4980    S(I)=S(I)+T
4990 NEXT I
5000 IF U>=Eps THEN 4930
5010 FOR I=1 TO N-1
5020    G(I)=(S(I+1)-S(I))/(X(I+1)-X(I))
5030 NEXT I
5040 IF Narg=0 THEN 5360
5050 !
5060 !
5070 ! *** CALCULATE FUNCTION VALUES AND DERIVATIVES.
5080 FOR J=1 TO Narg
5090 Corrector:    I=1
5100    T=Domain(J)
5110    IF T>=X(1) THEN 5190
5120    PRINT LIN(2),"ERROR IN SUBPROGRAM Spline."
5130    PRINT "ARGUMENT OUT OF BOUNDS."
5140    PRINT "X(1)=";X(1),"X(N)=";X(N),"Domain(";J;")=";Domain(J),LIN(2)
5150    PAUSE
5160    GOTO 5090
5170    !
5180    !
5190    I=I+1
5200    IF I>N THEN 5120
5210    IF T>X(I) THEN 5190
5220    I=I-1
5230    H=Domain(J)-X(I)
5240    T=Domain(J)-X(I+1)
5250    X=H*T
5260    S=S(I)+H*G(I)
5270    Z=1/6
5280    U=Z*(S(I)+S(I+1)+S)
5290    W=(Y(I+1)-Y(I))/(X(I+1)-X(I))
5300    Func(J)=W*H+Y(I)+X*U
5310    Deriv(J)=W+(H+T)*U+Z*X*G(I)
5320 NEXT J
5330 !
5340 !
5350 ! *** CALCULATE INTEGRAL FROM X(1) TO X(N).
5360 Int=0
5370 FOR I=1 TO N-1
5380    H=X(I+1)-X(I)
5390    Int=Int+.5*H*(Y(I)+Y(I+1))-1/24*H^3*(S(I)+S(I+1))
5400 NEXT I
5410 SUBEND
```

APPENDIX F

SEMLGC Program

```
10   REM PROGRAM TO READ PERMEABILITY DATA AND PLOT SEMILOG  9/10/80 *****
20    REM  ****************  REVISION C  *****************
30   OPTION BASE 1
40   COM Time$(1000),Tbegin,Tend,SHORT Data(1000,5),INTEGER Tscan,Start
50   COM Tyme(15),Model(15),Domain(20),Func(20),P1$,Repeat$,Id$(5)[20]
60   Narg=20
70    Tscan=0
80    Diff$=""
90    Repeat$=""
100   P=1
110   Q=0
120   DATA INJECTION PRESSURE,INJECTION FLOW,TOTAL GALLONS,BL,BL
130   FOR I=1 TO 5
140    READ Id$
150   NEXT I
160 Begin:  Different=0
170   Tscan=0
180   ON KEY #3 GOTO 350
190   ON KEY #4 GOTO 720
200   ON KEY #5 GOTO 530
210  PRINTER IS 16
220   INPUT "IS DATA GOING TO BE READ IN FROM A FLOPPY DISK? Y OR N",A$
230   EXIT GRAPHICS
240   PRINT PAGE
250  IF UPC$(A$[1,1])="N" THEN 280
260  MASS STORAGE IS ":F8"
270   GOTO 290
280   MASS STORAGE IS ":T15"
290   PRINT PAGE
300   PRINT "PUT TAPE OR FLOPPY (WHICHEVER WAS SPECIFIED AS BEING THE MEDIUM",LI
N(1),"FROM WHICH DATA WAS TO BE READ) INTO THE PROPER DRIVE."
310   PRINT "....THEN PRESS CONTINUE. THE CATALOG OF THIS TAPE/FLOPPY WILL THE
N",LIN(1),"BE PRINTED ON CRT DISPLAY FOR YOU TO SEE WHAT FILES ARE THERE",LIN(1)
320   PAUSE
330   PRINT PAGE
340   CAT
350 Read: INPUT "WHAT IS THE DATA FILE NAME TO BE READ?",F$
360  ASSIGN #1 TO F$[1,6]
370  READ #1;Time$(*),Data(*),Tscan
380   PRINT PAGE,LIN(5)
390   INPUT "DO YOU WANT TO STORE UPGRADED FILES, Y OR N?",Store$
400   IF UPC$(Store$[1,1])="N" THEN 490
410   PRINT LIN(4),"PUT IN DISK FOR STORING NEW DATA. PRESS CONT WHEN READY"
420   PAUSE
430   CREATE F$,5,10000
440   ASSIGN #3 TO F$
450   PRINT #3;Time$(*),Data(*),Tscan,Start
460   PRINT #3;END
470   ASSIGN #3 TO *
480   PAUSE
490   FOR I=1 TO Tscan
```

```
500     Data(I,1)=Data(I,1)*80
510     Data(I,2)=Data(I,2)/.5
520    NEXT I
530 Plot: REM THIS BEGINS PLOTTING ROUTINE BY SELECTING TYPE OF PLOT DESIRED
540   PRINTER IS 16
550    PRINT PAGE
560    PEN P
570    INPUT "WHAT IS BEGINNING SCAN NUMBER FOR PLOT?",Tb
580    INPUT "WHAT IS FINAL SCAN NUMBER FOR PLOT?",Te
590   INPUT "WHICH CHANNEL NUMBER DO YOU WISH TO PLOT?",Chan
600   INPUT "ARE YOU USING THE 9872A 4-PEN PLOTTER, Y OR N?",P1$
610   P=1
620 Log:  INPUT "HOW MANY CYCLES IS LOG PLOT?",Cycle
630    INPUT "WHAT IS MINIMUM PRESSURE TO BE PLOTTED?",Minp
640    INPUT "WHAT IS MAXIMUM PRESSURE TO BE PLOTTED?",Maxp
650    IF UPC$(P1$[1,1])<>"Y" THEN 700
660    PLOTTER IS 7,5,"9872A"
670    PEN P
680    LOCATE 0,1000/7,0,100
690    GOTO 730
700    PLOTTER IS 13,"GRAPHICS"
710    GRAPHICS
720    LOCATE 0,120,0,100
730    Tbegin=0
740    Tend=Cycle
750    SCALE Tbegin,Tend,Minp,Maxp
760    FRAME
770    FOR C=0 TO Cycle-1
780    FOR I=1 TO 10 STEP 1
790     Q=C+LGT(I)
800     MOVE Q,Minp
810     DRAW Q,Minp+(Maxp-Minp)/50
820     MOVE Q,Maxp
830     DRAW Q,Maxp-(Maxp-Minp)/50
840     IF I=10 THEN DRAW Q,Minp
850    NEXT I
860    NEXT C
870    LORG 4
880    FOR I=0 TO Cycle
890     MOVE I,Minp+(Maxp-Minp)/40
900     LABEL USING "4D,X";10^I
910    NEXT I
920    MOVE Cycle/2,Minp+(Maxp-Minp)/20
930    LABEL USING "K";"TIME (SEC)"
940    IF Maxp-Minp>10 THEN 1020
950    AXES 100,.5,Tbegin,Minp-2,10,2,4
960    AXES 100,.5,Tend,Minp-2,10,2,4
970    FOR V=Minp+1 TO Maxp-1 STEP 1
980     MOVE Tbegin+.04*(Tend-Tbegin),V
990     LABEL USING "2D";V
1000   NEXT V
1010   GOTO 1240
1020   IF Maxp-Minp>25 THEN 1100
1030   AXES 100,1,Tbegin,Minp-4,10,2,4
1040   AXES 100,1,Tend,Minp-4,10,2,4
1050   FOR V=Minp+2 TO Maxp-2 STEP 2
1060    MOVE Tbegin+.04*(Tend-Tbegin),V
1070    LABEL USING "MDD";V
1080   NEXT V
1090   GOTO 1240
1100   IF Maxp-Minp>80 THEN 1180
1110   AXES 100,2.5,Tbegin,Minp-5,10,2,4
1120   AXES 100,2.5,Tend,Minp-5,10,2,4
1130   FOR V=Minp+5 TO Maxp-5 STEP 5
1140    MOVE Tbegin+.04*(Tend-Tbegin),V
1150    LABEL USING "MDD";V
```

```
1160  NEXT V
1170  GOTO 1240
1180  AXES 100,5,Tbegin,Minp-20,10,2,4
1190  AXES 100,5,Tend,Minp-20,10,2,4
1200  FOR V=Minp+20 TO Maxp-20 STEP 20
1210   MOVE Tbegin+.04*(Tend-Tbegin),V
1220   LABEL USING "M3D";V
1230  NEXT V
1240  MOVE Tbegin+.08*(Tend-Tbegin),Minp+(Maxp-Minp)/2
1250  LDIR PI/2
1260  LORG 5
1270  LABEL USING "K";"PRESSURE (PSI)"
1280  LDIR 0
1281  PEN 1
1290  FOR I=Tb TO Te STEP 1
1300  Hrs=VAL(Time$(I)[7,8])
1310  Min=VAL(Time$(I)[10,11])
1320  Sec=VAL(Time$(I)[13,14])
1330  Min=Min+Hrs*60
1340  Sec=Sec+Min*60
1350  Date=Sec
1360  IF I=Tb THEN Begin=Date
1370  Date=Date-Begin
1380  IF Date<Tbegin THEN 1440
1390  IF Date>10^Tend THEN 1470
1400  LINE TYPE 1
1410  IF I=Tb THEN 1430
1420  PLOT LGT(Date),Data(I,Chan),1
1430  PENUP
1440  NEXT I
1450  CSIZE 3.5
1460  LORG 2
1470  MOVE Cycle/2,.9*(Maxp-Minp)+Minp
1480  LABEL USING "K";"POSTTEST PERMEABILITY"
1490  MOVE Cycle/2,.85*(Maxp-Minp)+Minp
1500  LABEL USING "K,K";"FROM ",Time$(Tb)
1510  MOVE Cycle/2,.82*(Maxp-Minp)+Minp
1520  LABEL USING "K,K";"TO ",Time$(Te)
1521  PENUP
1522  PEN 0
1530  INPUT "ARE MORE CURVES FROM THIS DATA TO GO ON PLOT? Y OR N",More$
1540  IF UPC$(More$[1,1])="N" THEN 1570
1550  INPUT "WHAT IS THE NEW CHANNEL TO BE PLOTTED?",Chan
1560  GOTO 1281
1570  PRINT "YOU CAN EXPAND THE PLOTTING CAPABILITIES BY READING IN MORE DATA TO
",LIN(1),"EITHER EXTEND PLOT FOR OTHER TIMES OR MOVE TO OTHER GROUPS OF"
1580  PRINT "CHANNELS.",LIN(1)
1590  INPUT "DOES MORE DATA NEED TO BE READ IN TO ADD TO THIS PLOT? Y OR N",Repe
at$
1600  IF UPC$(Repeat$[1,1])<>"N" THEN Begin
1601  IF UPC$(P1$[1,1])="Y" THEN 1670
1610 LOCATE 0,120,0,100
1620 PRINTER IS 0
1630  PRINT PAGE,LIN(12)
1640 DUMP GRAPHICS
1650 INPUT "DO YOU WANT ANOTHER COPY OF THIS PLOT, Y OR N?",Copy$
1660 IF UPC$(Copy$[1,1])<>"N" THEN 1630
1670  PLOTTER IS 13,"GRAPHICS"
1680 INPUT "DO YOU WANT SOME OTHER PLOTS FROM THIS DATA?",G$
1690  Different=0
1700  Diff$=""
1710 IF UPC$(G$[1,1])<>"N" THEN Plot
1720 Exit: INPUT "DO YOU WANT TO READ IN MORE DATA FROM DISK/TAPE, Y OR N?",Y$
1730 IF UPC$(Y$[1,1])<>"N" THEN 70
1740 PRINT "TERMINATING PROGRAM AT YOUR REQUEST. OVER AND OUT!!!"
1750 END
```

DISTRIBUTION:

DOE/TIC-4500 (Rev 69) U-70 (319)

US Department of Energy (3)
Office of Waste Isolation
Germantown, MD 20767
Attn: C. A. Heath, Director
Rm B-207
C. R. Cooley, Deputy Director
Rm B-214
D. L. Vieth, Actg Team Leader
Rm B-220

US Department of Energy
National Waste Terminal
Storage Program Office
505 King Ave.
Columbus, OH 43201
Attn: J. O. Neff, Program Manager

Lawrence Livermore National Lab (2)
University of California
P. O. Box 808
Livermore, CA 94550
Attn: L. D. Ramspott, Technical
Project Officer, MS L-204
K. Street, Jr., MS L-209

Los Alamos National Lab (2)
University of California
P. O. Box 1663
Los Alamos, NM 87545
Attn: B. R. Erdal, Technical
Project Officer, MS 514
D. C. Hoffman, MS 760

Westinghouse - AESD
P. O. Box 708
Mail Stop 703
Mercury, NV 89023
Attn: A. R. Hakl, Site Manager

US Geological Survey (2)
P. O. Box 25046
Mail Stop 954
Federal Center
Denver, CO 80301
Attn: G. L. Dixon, Tech Prjct Officer
W. E. Wilson

W. S. Twenhofel
820 Estes St
Lakewood, CO 80226

US Department of Energy
Richland Operations Office
P. O. Box 550
Richland, WA 99352
Attn: R. G. Goranson

Rockwell International Atomics
International Division
Rockwell Hanford Operations
Richland, WA 99352
Attn: R. Deju

US Department of Energy (13)
P. O. Box 14100
Las Vegas, NV 89114
Attn: R. M. Nelson, Jr., Dir. (3)
Waste Mgmt Project Office
D. F. Miller, Director
Office of Public Affairs
R. H. Marks, CP-1, M/S 210
B. W. Church, Director
Health Physics Division
R. R. Loux (7)

Holmes & Narver, Inc
P. O. Box 14340
Las Vegas, NV 89114
Attn: A. E. Gurrola

Battelle (8)
Office of Nuclear Waste Isolation
505 King Ave.
Columbus, OH 43201
Attn: N. E. Carter
W. A. Carbiener
ONWI Library (5)

State of Nevada
Capitol Complex
Governor's Office of Planning Coordination
Carson City, NV 89023
Attn: R. M. Hill, State Planning Coordinator

State of Nevada
Capitol Complex
Department of Energy
Carson City, NV 89710
Attn: N. A. Clark

DISTRIBUTION (cont):

International Atomic Energy Agency
Division of Nuclear Power & Reactors
Karnter Ring 11
P. O. Box 590, A-1011
Vienna, Austria
Attn: J. P. Colton

Reynolds Electrical & Engineering Co., Inc
P. O. Box 14400
Las Vegas, NV 89114
Attn: H. D. Cunningham, MS 555

Fenix & Scisson, Inc
P. O. Box 15408
Las Vegas, NV 89114
Attn: J. A. Cross

Argonne National Labs
9700 S. Cass Ave.
Argonne, IL 60439
Attn: A. M. Friedman

1000 G. A. Fowler
1100 C. D. Broyles
1110 J. D. Plimpton
1112 C. R. Mehl
1112 D. R. Waymire (10)
1116 W. Barrett
1116 J. I. Greenwoll
1116 K. E. Scranton
1120 T. L. Pace
1123 J. T. Lindman
1123 J. W. Schwarz
1124 J. W. Wister
1124 C. O. Duimstra (5)
1124-1 M. E. Burke
1125 J. J. Loukota
1125 J. T. McIlmoyle
1126 T. J. Stetz
1131 R. J. Dye
1417 F. W. Muller
1417 G. F. Rudolfo (2)
4500 E. H. Beckner
4510 W. D. Weart
4512 T. O. Hunter
4530 R. W. Lynch
4531 L. W. Scully
4537 L. D. Tyler
4537 J. K. Johnstone (10)
4537 D. R. Fortney
4537 B. S. Langkopf
4537 A. R. Lappin
4537 R. M. Zimmerman
4537 QA/QC Files
4538 R. C. Lincoln (7)
4538 A. E. Stephenson
4540 M. L. Kramm
4550 R. M. Jefferson
5511 J. W. Nunziato
5511 G. R. Hadley
5512 D. F. McVey
5533 W. T. Brown
8214 M. A. Pound
3141 L. J. Erickson (5)
3151 W. L. Garner (3)
For DOE/TIC (Unlimited Release)

Org.	Bldg.	Name	Rec'd by	Org.	Bldg.	Name	Rec'd by

Sandia National Laboratories