COO-4041-13

Distribution Category UC-64

BIOFOULING AND CORROSION STUDIES

FINAL REPORT

PART II

MASTER

January 1, 1978 to October 31, 1979

by

J. G. FETKOVICH

Carnegie-Mellon University
Pittsburgh, PA 15213

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

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, 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 or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

COO-4041-13

Distribution Category UC-64

DOE/ET/20310--13

DE83 001011

BIOFOULING AND CORROSION STUDIES

FINAL REPORT

PART II

MASTER

January 1, 1978 to October 31, 1979

by

J. G. FETKOVICH

Carnegie-Mellon University
Pittsburgh, PA 15213

NOTICE

This report was prepared as an account of work sponsored by The United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, sub-contractors, 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.

This work was supported by the Department of Energy, Solar Energy Division, under Contract No. EY-76-S-02-4041.

DISCLAIMER

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, 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 or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

grrf
DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

CONTENTS

1. INTRODUCTION	1
2. BIOFOULING STUDIES	5
2.1 Keahole Point Series II	5
2.2 Keahole Point Series III	9
2.3 Gulf of Mexico Tugboat Operation	11
2.4 Gulf of México; First Buoy Operation (GBI)	13
2.5 Gulf of Mexico; Second Buoy Operation (GBII)	14
3. SOFTWARE DEVELOPMENT	70
3.1 New Analysis Program	70
3.2 TI-59 Programs	73
3.2.1 Comparing our Standard Fortran Analysis Program to the TI-59 Program	73
3.2.2 Comparing Results of the Two Analysis Programs	74
4. INTERPRETATION AND IMPLICATIONS OF OTEC BIOFOULING DATA	76
4.1 Observations	76
4.2 Interpretation	77
4.3 Conclusion	87

APPENDIX A	A1
APPENDIX B1	B1.1
APPENDIX B2	B2.1
APPENDIX B3	B3.1
APPENDIX B4	B4.1
APPENDIX B5	B5.1

LIST OF TABLES

2.1	Parameters of the KII Experiments*
2.2	Parameters of the KIII Experiments
2.3	Parameters of the GT Experiments
2.4	Parameters of the GBI Experiments
2.6a - 2.6d	Data from the KII Experiments for Units A, B, C, and D
2.7a - 2.7d	Data from the KIII Experiments for Units A, B, C, and D
2.8a, 2.8b	Data from the GT Experiments for Units A and B
2.9a - 2.9c	Data from the GBI Experiments for Units A, B, and C
4.1	Parameters of CMU OTEC Biofouling Experiments

* See Section 1.1 for an explanation of the nomenclature KII, etc.

LIST OF FIGURES

- 2.1a - 2.1d Graphs of R_f vs. time for the KII Experiments*
- 2.2a - 2.2d Graphs of R_f vs. time for the KIII Experiments
- 2.3a, 2.3b Graphs of R_f vs. time for the GT Experiments
- 2.4a - 2.4c Graphs of R_f vs. time for the GBI Experiments
- 2.6b - 2.6d Graphs of flow velocity vs. time for the KII Experiments
- 2.7a - 2.7d Graphs of flow velocity vs. time for the KIII Experiments
- 2.8a, 2.8b Graphs of flow velocity vs. time for the GT Experiments
- 2.11 Graph of seawater temperature vs. time for the KII Experiments
- 2.12 Graph of seawater temperature vs. time for the KIII Experiments
- 2.13 Graph of seawater temperature vs. time for the GT Experiments
- 4.1 Schematic representation of the general character of the thickness of the fouling film as a function of time
- 4.2 Graph of the biofouling rate vs. fractional down time of flow

* See Section 1.1 for an explanation of the nomenclature KII, etc.

1. INTRODUCTION

This report covers the activities of the CMU biofouling group during the period January 1, 1978, through September 30, 1979. The members of the group during this time were J. G. Fetkovich ($\sim \frac{1}{4}$ time), P. I., and G. N. Grannemann. For most of the period (until May 31, 1979) L. Mahalingam also contributed to the work.

The report is organized into sections covering the various aspects of the work as described in the following subsections.

1.1 Biofouling Data

During the period covered by this report, data were analyzed from the KII, KIII, GT, GBI, and GBII experiments. These data are presented in Section 2, along with additional relevant information which bears on the interpretation of the results. (Experiments KII and KIII are the second and third series done on the submerged buoy at Keahole Point, Hawaii. Experiment GT is the one done in the Gulf of Mexico with the heat transfer monitors mounted on the deck of a tugboat. GBI and GBII refer to the first and second series done on a surface buoy in the Gulf.) Our group did not do the primary data analysis for the Gulf of Mexico experiments except for certain checks that were requested by DOE. The full data are discussed here, however, for completeness.

1.2 Software Development

The computer program used to analyze the data from the Keahole experiments was modified to facilitate the rejection of bad data and analysis of good data with much less need for manual editing.

A program was written, for the TI-59 programmable calculator, to allow technicians in the field to quickly calculate heat transfer coefficients from the raw data.

These results are discussed in Section 3.

1.3 Interpretation of Biofouling Data

We take a broad look at all of the biofouling data obtained through September 1979, and consider its interpretation. We have developed a simple physical model which, we show, represents the data very well.

The model and its implications are discussed in Section 4.

1.4 Consultation

Considerable time was spent during this period providing assistance to various groups involved in using the CMU heat transfer monitoring devices in the biofouling studies.

Naturally, we worked very closely with the Hawaii groups for which we were doing the primary data analysis. However, much work was done for other groups as well. This included:

1.4.1 Consultation with the NDBO group and others associated with the Gulf of Mexico experiments. After the Gulf tugboat data had been obtained and analyzed, we were asked to check it. Our analysis indicated

that the data did not behave as expected from previous experience. The rather severe problems could not be traced to their source because by the time we saw the data, the experiment was over and the apparatus had been disassembled. We suspect improper assembly procedures which resulted in poor thermal contact between the heater cylinder and test pipe. Our conclusion was that the data were not usable.

Subsequently, we were asked to check the test data from the GBI experiments which had already been deployed. We confirmed that the test results appeared normal.

Later, we were involved in the GBII experiments from the beginning. One of us (Mahalingam) was physically present and contributed to the assembly of the apparatus, initial test, calibrations, etc. He found a serious problem of poor thermal contact between the heater cylinders and the test pipes. It is very likely the same problem that doomed the earlier GT experiments. In this instance, however, the problems were found and fixed before deployment.

1.4.2 Consultation with the ANL group. This included involvement in a number of matters during this period. The most significant involved the modifications in the design of the heat transfer monitor units.

1.4.3 Consultation with the Puerto Rico group. This group recently initiated biofouling studies off the coast of Puerto Rico. We were involved from the initial planning stages. Our contributions included assembly, testing, and calibration of the apparatus in our laboratory at CMU.

1.4.4 Studies of the NSRDC Panama City cleaning data and apparatus.

After the cleaning experiments had been in progress for a long period, we were asked to check the results. The data were highly erratic and the results are probably not useful except perhaps in qualitative terms. One of us (G. N. Grannemann) went to the site to study the apparatus and procedures. The principal problems were identified and solutions suggested. We concluded that the masses of bad data probably could be salvaged through suitable reanalysis. To our knowledge, this has not been done.

We do not further discuss these various consulting activities in this report except insofar as they relate to the subjects of the following sections.

2. BIOFOULING STUDIES

2.1 Keahole Point Series II

This experimental series is the second done with heat transfer monitors mounted on a submarine buoy off Keahole Point on the western coast of Hawaii. The details of the buoy system and operation may be found in part I of this report.

The series involved four units in which the simulated heat exchanger tube materials were titanium (2) and aluminum (2), all operated at nominal flow velocities of 6 ft/sec. The parameters of the experiments are given in Table 2.1. Tables 2.6a - 2.6d are complete records of the measured values of heat transfer coefficient (h) and thermal resistance (R_f) of the fouling layer during the run. These same data are shown graphically in Figures 2.1a - 2.1d. The flow velocities for three units* are plotted in Figures 2.6b - 2.6d, and the seawater temperature during the experiment is given in Figure 2.11. The experiment started on December 6, 1977, and continued to July 24, 1978, a period of 33 weeks.

In previous work at Keahole Point, the CMU group was intimately involved in all aspects of the work associated with the heat transfer monitors (construction, assembly, testing, etc.). In KII, for the first time, this part of the work was turned over to the local University of Hawaii group. When an operation like this is turned over to a new group, one expects "bugs" to develop in the system, leading to problems. These problems were aggravated because of untested changes in procedure which were first used in these experiments. Some of the more serious problems which

* Velocity data are not available for Unit A due to failure of the flowmeter.

developed were:

1. Construction of certain parts was faulty. In particular, the test pipes for units B and C were machined to wrong dimensions.
2. Assembly procedures were incorrect. This led to bad Wilson Plot parameters on units B and C (see Table 2.1).
3. Quality control was poor. The assembly problems leading to the poor Wilson Plot parameters on units B and C should have been resolved before the units were mounted on the buoy.
4. Untested changes in data-taking procedures were initiated. In particular, it was decided not to take flowmeter zero readings regularly. This (along with cable and connector problems) resulted in degradation of the velocity data.
5. There were a number of pump outages and flow velocity changes (indicated in Figures 2.1a - 2.1d and 2.6a - 2.6d). The pump on unit B gave out entirely, and B was then connected on the downstream side of unit D at week 10.
6. The submarine cable and connectors went bad leading to a long data outage (weeks 14 to 22) during repairs, and to continuing problems with the flowmeter readings. Flow velocity uncertainties are at about 10%.

Items 1, 2, 3, and 4, above, are perhaps most serious in that they have caused the quality of all the data to be compromised.

Pump outages (Item 5) would not be serious if the equipment included a simple warning system which would alert an operator when a shutdown occurred. Past evidence shows that as long as a shutdown does not last

more than a few hours, subsequent data are apparently not seriously affected. When it lasts of the order of a day or more, however, macro-foulers such as barnacles attach and grow in the system, probably seriously affecting the flow conditions and the interpretation of results.

Apart from the cost and inconvenience of having to repair or replace a cable or connectors, Item 6 is not a serious problem. Data are lost, of course, as long as the data transmission is interrupted, but as long as pumping is continuous, data acquired after the outage are good.

Because of the problems listed above, the data from these runs must be considered qualitatively, but not quantitatively useful. Until the time of the first pump outages, the data behave as expected from earlier results. The cleaning tests again show (roughly) that MAN brushes are effective in removing the accumulated fouling, at least if the system has not been run for more than a fraction of a year. However, it is not possible to know precisely how effective the cleaning was because of the large uncertainties in the velocity data.

We see unusual variations in R_f which are not associated with cleaning tests (Unit B weeks 10-11 and 22-24, Unit C weeks 21-26, Unit D weeks 25-33). It is possible these are due to pump stoppages and large flow velocity changes, either directly (as with Unit B at 10 weeks) or indirectly (due to effects of growing macrofoulers days or weeks after the stoppages).

In the past, we have assumed an error of about 1% on measured values of h . The random errors of measurement are usually smaller than this, but we expect systematic errors of about this amount due to uncertainty in the absolute calibration in the flowmeters. We have adhered to this

practice here (Tables 2.6a - 2.6d and Figures 2.1a - 2.1d). However, due to the problems with obtaining flow velocity data in this experiment, we are sure the actual velocity errors are much bigger than this. However, the quoted errors in R_f , as calculated assuming the 1% errors in h , are probably not far wrong in the early parts of the run. They will be approximately correct even if the velocity is poorly known as long as the velocity didn't change. Unfortunately, we do not know how long this condition held, except to say that it is certainly not longer than 9-10 weeks for units A, B, and D, and not longer than 15 weeks for Unit C.

2.2 Keahole Point Series III

This series of experiments involved four units with operating characteristics shown in Table 2.2. They were Aluminum (5052), Titanium (grade 2), 90/10 Copper-Nickel, and Stainless Steel, all at nominal 6 ft/sec. Temperatures and velocities are plotted in Figures 2.11, 2.7a - 2.7d.

In these experiments, the first cleaning tests were done early, so that the biofouling levels barely got out of the induction phase. Past data show that the linear growth phase begins at a fouling level of almost 20 hr. ft² F/BTU. The KIII data, shown plotted in Figures 2.2a through 2.2d, indicate that only unit B was allowed to progress significantly beyond the induction region.

The data are also tabulated in Tables 2.7a through 2.7d.

These experiments were also plagued with equipment problems. Due to a malfunction, unit B operated only for eighteen weeks. Furthermore, the Ramapo flowmeter readings were seriously erratic beginning in the second week. This was apparently due to seawater leakage into the submarine cables and/or connectors. As a result, there were no usable velocity readings through weeks 2 to 13. Beginning week 14 periodic velocity readings were taken by divers who took manually operated TSK flowmeters to the equipment. During weeks 18 through 29, there was a complete shutdown of data transmission while the cables and electronics were brought up, fixed, and replaced. During this period, however, the pumping continued on the units. After the equipment was replaced (week 30), Ramapo flowmeter readings were again available. However, the flow data were erratic. This is presumed to be due to flow stoppages, which occurred during equipment

replacement. We expect that macrofoulers which attach during flow stoppages will grow and affect the flow pattern. If they attach on or near the flowmeter element, they can seriously affect the readings.

Another datum of significance is that there is considerably more scatter in the heat transfer units than usual. This may be due to the fact that these units were "tuned" before deployment by decreasing the torque applied to the heater cylinders below the standard value. This has the effect of reducing the thermal contact between the cylinder and test pipe. We expect that this "looser" contact will lead to increased sensitivity to fluctuations in the seawater temperature (due to differential thermal expansion). We recommend that this method of tuning not be used.

We note too that the Wilson Plot data for these units are not as good as usual. Units A, B, and C have Wilson Plot slopes which are about 5% high. Unit D had a large intercept, and yielded rather noisy Wilson Plot data. These effects may also be associated with the detorquing of these units.

Finally, we note that there were rather large (and sometimes unknown) flow velocity changes during the course of this experiment.

The comment on errors at the end of Section 2.1 also applies to this KIII data.

Given all these problems, we feel that while one may safely draw qualitative conclusions from the KIII data, their quantitative significance must be considered suspect. In qualitative terms we feel that the data are in essential agreement with previous studies. Probably the most significant point about the results is that 90-10 Cu/Ni does not appear to inhibit significantly the growth of biofouling at the levels observed here.

2.3 Gulf of Mexico Tugboat Operation

In November 1977, two heat transfer monitor units were mounted on a tugboat moored in the Gulf of Mexico for a first study of biofouling in the Gulf waters. This "Gulf Tug" (GT) experiment was carried out by other groups. However, after it ended, we were asked to reanalyze the data, which appeared abnormal.

Our information is that there were two units in operation. Both used simulated heat exchanger tubes of aluminum (5052). One was operated with a nominal flow velocity of 6 ft/sec, and the other was 3 ft/sec. The parameters of the run are shown in Table 2.3. The results of our analysis of the data are shown in Tables 2.8a and 2.8b and in Figures 2.3a and 2.3b. The flow velocities are shown in Figures 2.8a and 2.8b, while the seawater temperature is shown in Figure 2.13. The experiment ran for 4.7 weeks; from November 7 through December 10.

We found a number of serious problems in the data. Perhaps the most significant oversight was in not taking Wilson Plot data to check the operation of the apparatus before beginning the run. In the case of the 3 ft/sec unit, the initial (clean tube) value of h is 521 BTU/hr ft² F compared to an expected value of 667. This is an enormous discrepancy. It indicates that if a Wilson Plot had been done, the parameters would have been found to be seriously wrong. Such very poor Wilson Plot results usually are related to incorrect assembly procedures (e.g. not torquing the heater cylinder down to specifications) or construction errors (e.g. machining parts to wrong dimensions). Perhaps related to this problem was a greater-than-usual scatter in the h

data. The data were so bad in the case of the 3 ft/sec unit that there were no usable results after November 26. In the same unit, we find the heat transfer coefficient increasing with time, contrary to expectation and past experience.

There were also various less serious problems in the data, such as short-duration cooling curves and numerous formatting errors.

Another anomaly in this data, which was not seen in previous data (e.g. NOII, K-I, and St. Croix), was very large fluctuations in the flowmeter-read velocities. We do not now know whether these fluctuations are real (due to large accelerations of the tugboat) or only apparent (resulting, e.g., from inertial forces acting on the target of the Ramapo fluctuations).

Our studies of the data have not led to any means for extracting useful information from them. In any case, the short duration of the experiment (less than five weeks) would not have allowed significant fouling to build up.

2.4 Gulf of Mexico; First Buoy Operation (GBI)

The GBI experiments were carried out by other groups, who also did the data analysis. We consulted with these groups during the experiments, and were asked to study the data.

In this experiment, three heat transfer monitors were mounted on a buoy in the Gulf of Mexico. The parameters of the experiment are shown in Table 2.4. The data are tabulated in Tables 2.9a - 2.9c and are plotted in Figures 2.4a - 2.4c. The velocity and seawater temperature data for these runs were not provided to us.

The Wilson Plot data (Table 2.4) show moderately large intercepts (units B and C), and slopes which are not in good agreement with expectations (units A and C). Then data were obtained at sea, after the experiment had begun. Since we did not receive the Wilson Plot data until the run was well underway, we judged that these discrepancies were not large enough to warrant halting the run and starting over.

We note that the data (including the Wilson Plots) may be systematically in error by several percent because certain information needed in the data analysis (e.g., pipe dimensions) were not available to us. Also, the data are tabulated and plotted without associated errors. We were not in a position, due to lack of data, to estimate errors in this experiment.

We note that there are several unusual aspects in the conduct of this experiment. The Wilson Plot data were taken at sea after the experiment had started, rather than in the lab. Then, the pipes were cleaned after the experiment had run for one week. Finally, for reasons

unknown to us, the run was terminated after less than ten weeks, so that only the induction period was studied; the linear biofouling region had not been penetrated in these data.

In the short duration of this experiment, we find no differences between the data obtained here and those obtained in previous work.

2.5 Gulf of Mexico Second Buoy Operation (GBII)

Our group was not directly involved with these experiments, but were asked to analyze the data. We did so, and found no important differences compared to the analysis done by the NOAA/Data Buoy Office group originally. We note here, however, that the experiments suffered severe equipment problems which resulted in serious disturbances in the flow beginning about one month after start up. We feel that the results of the heat transfer rate measurements after this must be considered suspect. A complete report of the heat transfer and other results of this experiment may be found in the report of Little et al.¹

TABLE 2.1

Unit	A	B	C	D
Material	Ti(Grade 2)	Al(5052)	Al(5052)	Ti(Grade 2)
Nominal Flow ¹	6	6	6	6
Wilson Plot Intercept ²	3.8 ± 2.4	41.7 ± 1.3	-5.0 ± 1.4	3.8 ± 1.4
Wilson Plot Slope ^{3,4}	349 ± 7	340 ± 4	361 ± 4	349 ± 4

Footnotes to Table 2.1

1. feet/sec.
2. $(\text{hr ft}^2 \text{ F/BTU}) \times 10^{-5}$
3. $(\text{hr ft}^2 \text{ F/BTU})(\text{ft/sec})^{0.8} \times 10^{-5}$
4. Expected slope is 344 (fresh water)

TABLE 2.2

Unit	A	B	C	D
Material	Al-5052	Ti	90/10 CuNi	SS(AL - 6X)
Nominal Flow ¹	6	6	6	6
Wilson Plot Intercept ²	5.2 ± 0.3	-3 ± 2	-4.3 ± 0.5	13 ± 2
Wilson Plot Slope ^{3,4}	359 ± 1	359 ± 6	351 ± 3	347 ± 7

Footnotes to Table 2.2

1. feet/sec.
2. $(\text{hr ft}^2 \text{ F/BTU}) \times 10^{-5}$
3. $(\text{hr ft}^2 \text{ F/BTU})(\text{ft/sec})^{0.8} \times 10^{-5}$
4. Expected slope is 344 (fresh water).

TABLE 2.3

Unit	A	B
Material	A1-5052	A1-5052
Nominal Flow Velocity	3 ft/sec.	6 ft/sec.
Wilson Plot Intercept	Not Available	Not Available
Wilson Plot Slope	Not Available	Not Available

TABLE 2.4

Unit	A	B	C
Material	Ti(grade 2)	Al(5052)	Al(5052)
Nominal Flow ¹	6	6	6
Wilson Plot Intercept ²	-0.7 ± 1.0	9.9 ± 0.8	9.6 ± 1.7
Wilson Plot Slope ^{3,4}	384.6 ± 3.4	366.0 ± 2.5	380.8 ± 5.6

Footnotes to Table 2.4

1. feet/sec.
2. $(\text{hr ft}^2 \text{ F/BTU}) \times 10^{-5}$
3. $(\text{hr ft}^2 \text{ F/BTU})(\text{ft/sec})^{0.8} \times 10^{-5}$
4. Expected slope is 364 (seawater).

SUMMARY OF DATA FROM SUBMERGED BUOY AT KEAHOLE POINT, HAWAII

Data normalized to 6 ft/sec and 70 °F.

Unit A: Titanium (Grade 2)

Flow Velocity¹: 5.54 ± 0.09 ft/sec.Wilson Plot²: $1/h = (3.8 \pm 2.4) \times 10^{-5} + (0.349 \pm 0.007) \times 10^{-2} V^{-.8} \text{ hr ft}^2 \text{ °F/BTU}$ (V in ft/sec)

Flow Started: Dec. 2, 1977

Date	Time Weeks	$\langle h \rangle^3$	$\langle 1/h \rangle$	$\Delta \langle 1/h \rangle \equiv R_f$
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ °F}}$	$\frac{\text{hr ft}^2 \text{ °F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ °F}}{\text{BTU}} \times 10^5$
Dec. 6, 1977	.57	1166 ± 12 (2)	85.8 ± 0.9	0 ± 1.2
Dec. 12, 1977	1.43	1177 ± 12 (3)	85.0 ± 0.8	-0.8 ± 1.2
Dec. 19, 1977	2.43	1161 ± 12 (3)	86.2 ± 0.9	0.4 ± 1.2
Dec. 27, 1977	3.57	1140 ± 11 (1)	87.8 ± 0.9	1.9 ± 1.2
Jan. 3, 1978	4.57	1124 ± 11 (2)	89.0 ± 0.9	3.2 ± 1.2
Jan. 10, 1978	5.57	1098 ± 11 (3)	91.0 ± 0.9	5.3 ± 1.3
Jan. 17, 1978	6.57	1050 ± 11 (3)	95.2 ± 0.9	9.4 ± 1.3
Jan. 24, 1978	7.57	987 ± 10 (4)	101.4 ± 1.0	15.6 ± 1.3
Jan. 31, 1978	8.57	910 ± 9 (2)	109.9 ± 1.1	24.1 ± 1.4
Feb. 4, 1978	9.14	864 ⁴ ± 9 (11)	115.7 ± 1.2	29.9 ± 1.7
Feb. 7, 1978	9.57	872 ± 9 (4)	114.7 ± 1.1	28.9 ± 1.4
Feb. 16, 1978	10.86	836 ± 8 (7)	119.6 ± 1.2	33.8 ± 1.5
Feb. 21, 1978	11.57	794 ± 8 (1)	125.9 ± 1.3	40.2 ± 1.5
Feb. 28, 1978	12.57	732 ± 7 (3)	136.7 ± 1.4	50.9 ± 1.6
Mar. 7, 1978	13.57	729 ± 7 (1)	137.2 ± 1.4	51.5 ± 1.6

TABLE 2.6a cont.

Date	Time	$\langle h \rangle^3$	$\langle l/h \rangle$	$\Delta \langle l/h \rangle \equiv R_f$
	Weeks	$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
Mar. 8, 1978	13.71	1079 $\pm$ 11 (5)	92.7 $\pm$ 0.9	6.9 ⁵ $\pm$ 1.3
Mar. 14, 1978	14.57	1036 $\pm$ 10 (4)	96.6 $\pm$ 1.0	10.8 $\pm$ 1.3
May 4, 1978	21.86	693 $\pm$ 7 (1)	144.3 $\pm$ 1.4	58.5 $\pm$ 1.7
May 12, 1978	23.00	1166 $\pm$ 12 (3)	85.8 $\pm$ 0.9	0.0 ⁵ $\pm$ 1.2
May 17, 1978	23.71	1119 $\pm$ 11 (4)	89.4 $\pm$ 0.9	3.6 $\pm$ 1.2
May 23, 1978	24.57	1002 $\pm$ 10 (3)	99.8 $\pm$ 1.0	14.0 $\pm$ 1.3
May 31, 1978	25.71	775 $\pm$ 8 (2)	129.0 $\pm$ 1.3	43.2 $\pm$ 1.5
June 6, 1978	26.57	663 $\pm$ 7 (3)	150.9 $\pm$ 1.5	65.1 $\pm$ 1.7
June 14, 1978	27.71	593 $\pm$ 6 (2)	168.7 $\pm$ 1.7	82.9 $\pm$ 1.9
June 20, 1978	28.57	539 $\pm$ 5 (1)	185.6 $\pm$ 1.9	99.9 $\pm$ 2.0
June 26, 1978	29.43	1139 $\pm$ 11 (13)	87.8 $\pm$ 0.9	2.0 ⁴ $\pm$ 1.3
June 27, 1978	29.57	1123 $\pm$ 11 (9)	89.1 $\pm$ 0.9	3.3 ⁵ $\pm$ 1.2
July 5, 1978	30.71	1013 $\pm$ 10 (8)	98.7 $\pm$ 1.0	13.0 $\pm$ 1.3
July 11, 1978	31.57	972 $\pm$ 10 (1)	102.9 $\pm$ 1.0	17.1 $\pm$ 1.3
July 18, 1978	32.57	903 $\pm$ 9 (1)	110.7 $\pm$ 1.1	24.9 $\pm$ 1.4
July 24, 1978	33.43	840 $\pm$ 8 (4)	119.0 $\pm$ 1.2	33.2 $\pm$ 1.5

FOOTNOTES FOR TABLE 2.6a

- 1) Flow velocity is an average of all TSK data taken to date. Ramapo flow meter inoperative in this unit.
- 2) The Wilson plot data were taken before deployment using fresh water and the buoy electronics. Data were taken at two velocities, four cooling curves at each velocity. Errors on the Wilson plot are determined by assuming 1% errors on the velocity measurement. Values to be expected are: $1/h = 0.344 \times 10^{-2} V^{-.8} \text{ hr ft}^2 \text{ }^\circ\text{F/BTU}$ (V in ft/sec).
- 3) Errors on h are given as 1% followed by the error on the mean of ~16 cooling curves. Errors on $1/h$ and R_f are calculated from 1% value.
- 4) Average of 2 cooling curves.
- 5) Unit cleaned using M.A.N. brushes on Mar. 8, May 11, and June 26, 1978.

TABLE 2.6b

SUMMARY OF DATA FROM SUBMERGED BUOY AT KEAHOLE POINT, HAWAII

Data normalized to 6 ft/sec and 70°F.

Unit B: Aluminum (5052)^cFlow Velocity¹:

12/ 2/77 - 2/ 8/78 7.33 ± 0.03 ft/sec

2/14/78⁷ - 3/15/78 8.29 ± 0.07 ft/sec

5/ 2/78 - 7/24/78 7.62 ± 0.02 ft/sec

Wilson Plot²: $1/h = (41.73 \pm 1.3) \times 10^{-5} + (0.340 \pm 0.004) \times 10^{-2} V^{-0.8} \text{ hr ft}^2 \text{ }^\circ\text{F/BTU}$ (V in ft/sec)

Flow Started: Dec. 2, 1977

Date	Time Weeks	$\langle h \rangle^4$	$\langle 1/h \rangle$	$\Delta \langle 1/h \rangle \equiv R_f$
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ }^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ }^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ }^\circ\text{F}}{\text{BTU}} \times 10^5$
Dec. 7, 1977	0.71	736 ± 7 (1)	135.8 ± 1.4	0 ± 1.9
Dec. 13, 1977	1.57	738 ± 7 (1)	135.5 ± 1.4	-0.3 ± 1.9
Dec. 20, 1977	2.57	688 ± 7 (7)	145.4 ± 1.5	9.6 ± 2.0
Dec. 28, 1977	3.71	681 ± 7 (2)	146.9 ± 1.5	11.1 ± 2.0
Jan. 4, 1978	4.71	664 ± 7 (1)	150.7 ± 1.5	14.9 ± 2.0
Jan. 11, 1978	5.71	646 ± 6 (1)	154.8 ± 1.5	18.9 ± 2.1
Jan. 18, 1978	6.71	625 ± 6 (2)	159.9 ± 1.6	24.1 ± 2.1
Jan. 25, 1978	7.71	598 ± 6 (2)	167.2 ± 1.7	31.4 ± 2.2
Feb. 1, 1978	8.71	578 ± 6 (1)	172.9 ± 1.7	37.1 ± 2.2
Feb. 4, 1978	9.14	545 ⁵ ± 5 (1)	183.6 ± 1.8	47.8 ± 2.3
Feb. 9, 1978	9.86	511 ± 5 (4)	195.8 ± 2.0	60.0 ± 2.4
Feb. 14, 1978	10.57	591 ± 6 (1)	169.2 ± 1.7	33.4 ± 2.2
Feb. 15, 1978	10.71	565 ± 6 (2)	177.0 ± 1.8	41.2 ± 2.2

TABLE 2.00 cont.

Date	Time Weeks	$\langle h \rangle^4$ $\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\langle 1/h \rangle$ $\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\Delta \langle 1/h \rangle \equiv R_f$ $\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
Feb. 22, 1978	11.71	560 $\pm$ 6 (1)	178.6 $\pm$ 1.8	42.8 $\pm$ 2.2
Mar. 1, 1978	12.71	519 $\pm$ 5 (2)	192.7 $\pm$ 1.9	56.9 $\pm$ 2.4
Mar. 9, 1978	13.86	518 $\pm$ 5 (2)	193.2 $\pm$ 1.9	57.4 $\pm$ 2.4
Mar. 15, 1978	14.71	507 $\pm$ 5 (1)	197.4 $\pm$ 2.0	61.6 $\pm$ 2.4
May 5, 1978	22.00	538 $\pm$ 5 (8)	185.8 $\pm$ 1.9	50.0 $\pm$ 3.2
May 9, 1978	22.57	547 $\pm$ 5 (3)	182.9 $\pm$ 1.8	47.0 $\pm$ 2.3
May 12, 1978	23.00	567 $\pm$ 6 (9)	176.3 $\pm$ 1.8	40.5 $\pm$ 3.2
May 18, 1978	23.86	597 $\pm$ 6 (7)	167.4 $\pm$ 1.7	31.6 $\pm$ 2.4
May 24, 1978	24.71	576 $\pm$ 6 (3)	173.5 $\pm$ 1.7	37.7 $\pm$ 2.2
June 1, 1978	25.86	548 $\pm$ 5 (3)	182.6 $\pm$ 1.8	46.8 $\pm$ 2.3
June 7, 1978	26.71	530 $\pm$ 5 (4)	188.8 $\pm$ 1.9	53.0 $\pm$ 2.3
June 15, 1978	27.86	500 $\pm$ 5 (2)	200.0 $\pm$ 2.0	64.2 $\pm$ 2.4
June 21, 1978	28.71	479 $\pm$ 5 (1)	208.8 $\pm$ 2.1	73.0 $\pm$ 2.5
June 28, 1978	29.71	470 $\pm$ 5 (6)	212.7 $\pm$ 2.1	76.9 $\pm$ 3.1
July 6, 1978	30.86	443 $\pm$ 4 (4)	225.7 $\pm$ 2.3	90.0 $\pm$ 2.6
July 12, 1978	31.71	450 $\pm$ 5 (2)	222.4 $\pm$ 2.2	86.5 $\pm$ 2.6
July 19, 1978	32.71	423 $\pm$ 4 (1)	236.2 $\pm$ 2.4	100.4 $\pm$ 2.7

FOOTNOTES FOR TABLE 2.6b

- 1) Flow velocities are averages of Ramapo and TSK data taken in the time intervals indicated. Prior to 6/2/78 the Ramapo flow meters worked intermittently. Since 6/2/78 the Ramapo data has been used to normalize the data. Flow velocity changes were caused by changes in plumbing necessitated by pump failures.
- 2) The Wilson plot data were taken before deployment using fresh water and the buoy electronics. Data were taken at two velocities, four cooling curves at each velocity. Errors on the Wilson plot are determined by assuming 1% errors on the velocity measurement. Values to be expected are: $1/h = 0.344 \times 10^{-2} \text{ V}^{-.8} \text{ hr ft}^2 \text{ }^{\circ}\text{F/BTU}$ (V in ft/sec).
- 3) This unit has an abnormally high thermal resistance in series with the heater cylinders. Any change in this resistance during the experiment would be interpreted as a change in R_f .
- 4) Errors on h are given as 1% followed by the error on the mean of ~16 cooling curves. Errors on $1/h$ and R_f are calculated from 1% value.
- 5) Average of 2 cooling curves.
- 6) Unit plumbed on discharge side of pump.
- 7) Pump failure about 2/8/78 required that Unit B be plumbed on the discharge side of the pump for Unit D. Thus water flow for remainder of the experiment was Unit D(Ti) $\rightarrow$ Pump $\rightarrow$ Unit B(A1).

TABLE 2.6c

SUMMARY OF DATA FROM SUBMERGED BUOY AT KEAHOE POINT, HAWAII

Data normalized to 6 ft/sec and 70°F.

Unit C: Aluminum (5052)

Flow Velocity¹:

12/ 2/77 - 3/ 8/78 7.19 ± 0.02

3/ 9/78 - 7/24/78 7.41 ± 0.03

Wilson Plot²: $1/h = (-5.0 \pm 1.4) \times 10^{-5} + (0.361 \pm .004) V^{-.8} \text{ hr ft}^2 \text{ }^\circ\text{F/BTU}$ (V in ft/sec)

Flow Started: Dec. 2, 1977

Date	Time Weeks	$\langle h \rangle^3$	$\langle 1/h \rangle$	$\Delta \langle 1/h \rangle \equiv R_f$
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ }^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ }^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ }^\circ\text{F}}{\text{BTU}} \times 10^5$
Dec. 6, 1977	0.57	1120 ± 11 (2)	89.3 ± 0.9	0 ± 1.3
Dec. 12, 1977	1.43	1118 ± 11 (4)	89.4 ± 0.9	0.2 ± 1.3
Dec. 19, 1977	2.43	1102 ± 11 (3)	90.7 ± 0.9	1.5 ± 1.3
Dec. 26, 1977	3.43	1056 ± 11 (1)	94.7 ± 0.9	5.5 ± 1.3
Jan. 2, 1978	4.43	1029 ± 10 (1)	97.1 ± 1.0	7.9 ± 1.3
Jan. 9, 1978	5.43	992 ± 10 (1)	100.8 ± 1.0	11.6 ± 1.3
Jan. 16, 1978	6.43	940 ± 9 (1)	106.4 ± 1.1	17.1 ± 1.4
Jan. 23, 1978	7.43	868 ± 9 (2)	115.2 ± 1.2	25.9 ± 1.5
Jan. 30, 1978	8.43	811 ± 8 (3)	123.3 ± 1.2	34.0 ± 1.5
Feb. 4, 1978	9.14	760 ⁴ ± 8 (2)	131.6 ± 1.3	42.3 ± 1.6
Feb. 6, 1978	9.43	733 ± 7 (2)	136.4 ± 1.4	47.2 ± 1.6
Feb. 16, 1978	10.86	714 ± 7 (1)	140.0 ± 1.4	50.7 ± 1.7
Feb. 20, 1978	11.43	645 ± 6 (2)	154.9 ± 1.5	65.7 ± 1.8
Feb. 27, 1978	12.43	592 ± 6 (1)	168.9 ± 1.7	79.6 ± 1.9

TABLE 2.6c (cont.)

Date	Time Weeks	$\langle h \rangle^3$	$\langle 1/h \rangle$	$\Delta \langle 1/h \rangle \equiv R_f$
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
Mar. 7, 1978	13.57	551 ⁴ ± 6 (2)	181.6 ± 1.8	92.3 ± 2.0
Mar. 8, 1978	13.71	533 ⁴ ± 5 (2)	187.6 ± 1.9	98.4 ± 2.1
Mar. 8, 1978	13.71	1076 ⁴ ± 11 (8)	92.9 ± 0.9	3.7 ⁵ ± 1.3
Mar. 10, 1978	14.00	1068 ± 11 (3)	93.6 ± 0.9	4.3 ± 1.3
Mar. 13, 1978	14.43	1066 ± 11 (3)	93.8 ± 0.9	4.5 ± 1.3
May 2, 1978	21.57	826 ± 8 (2)	121.1 ± 1.2	31.8 ± 1.5
May 11, 1978	22.86	1113 ± 11 (6)	89.8 ± 0.9	0.6 ⁵ ± 1.3
May 16, 1978	23.57	1084 ± 11 (12)	92.2 ± 1.0	3.0 ± 1.3
May 22, 1978	24.43	1060 ± 11 (3)	94.3 ± 0.9	5.1 ± 1.3
May 30, 1978	25.57	958 ± 10 (6)	104.4 ± 1.0	15.1 ± 1.4
June 4, 1978	26.29	867 ± 9 (14)	115.3 ± 1.8	26.0 ± 2.0
June 13, 1978	27.57	725 ± 7 (2)	138.0 ± 1.4	48.8 ± 1.6
June 19, 1978	28.43	687 ± 7 (2)	145.6 ± 1.5	56.4 ± 1.7
June 26, 1978	29.43	583 ± 6 (1)	171.7 ± 1.7	82.4 ⁶ ± 1.9
June 26, 1978	29.43	1117 ± 11 (16)	89.5 ± 1.3	0.2 ⁵ ± 1.6
July 3, 1978	30.43	1023 ± 10 (7)	97.8 ± 1.0	8.5 ± 1.3

Date	Time Weeks	$\frac{\overset{3}{\langle h \rangle}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\langle 1/h \rangle}{\text{BTU}} \times 10^5$	$\Delta \langle 1/h \rangle \equiv R_f$ $\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
July 10, 1978	31.43	982 $\pm$ 10 (1)	101.9 $\pm$ 1.0	12.6 $\pm$ 1.4
July 17, 1978	32.43	910 $\pm$ 9 (3)	109.8 $\pm$ 1.1	20.6 $\pm$ 1.4
July 24, 1978	33.43	800 $\pm$ 8 (4)	125.0 $\pm$ 1.2	35.8 $\pm$ 1.5

FOOTNOTES FOR TABLE 2.6c

- 1) Flow velocity is an average of Ramapo and TSK data taken in the time interval indicated. Prior to 6/2/78 the Ramapo flow meters worked intermittently. Since 6/2/78 the Ramapo value has been used to normalize the data. Flow velocity changes appear to be related to clearing.
- 2) The Wilson plot data were taken before deployment using fresh water and the buoy electronics. Data were taken at two velocities, four cooling curves at each velocity. Errors on the Wilson plot are determined by assuming 1% errors on the velocity measurement. Values to be expected are: $1/h = 0.344 \times 10^{-2} V^{-.8} \text{ hr ft}^2 \text{ }^\circ\text{F/BTU}$ (V in ft/sec).
- 3) Errors on h are given as 1% followed by the error on the mean of ~16 cooling curves. Errors on $1/h$ and R_f are calculated from 1% value.
- 4) Average of two cooling curves.
- 5) Unit cleaned using M.A.N. brushes on Mar. 8, May 11, and June 26, 1973.
- 6) Only one cooling curve taken.

TABLE 2.6d

SUMMARY OF DATA FROM SUBMERGED BUOY AT KEAHOLE POINT, HAWAII

Data normalized to 6 ft/sec and 70°F.

Unit D: Titanium (Grade 2)

Flow Velocity¹:

12/ 2/77 - 2/ 8/78 6.23 ± 0.01 ft/sec

2/14/78 - 3/15/78 8.20 ± 0.07 ft/sec

5/ 2/78 - 7/24/78 7.50 ± 0.02 ft/sec

Wilson Plot²: $1/h = (3.8 \pm 1.4) \times 10^{-5} + (0.349 \pm .004) \times 10^{-2} V^{-.8} \text{ hr ft}^2 \text{ }^\circ\text{F/BTU}$ (V in ft/sec)

Flow Started: Dec. 2, 1977

Date	Time	$\langle h \rangle^3$	$\langle 1/h \rangle$	$\Delta \langle 1/h \rangle \equiv R_f$
	Weeks	$\frac{\text{BTU}}{\text{hr ft}^2 \text{ }^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ }^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ }^\circ\text{F}}{\text{BTU}} \times 10^5$
Dec. 8, 1977	0.86	1190 ± 12 (1)	84.0 ± 0.8	0 ± 1.2
Dec. 14, 1977	1.71	1189 ± 12 (10)	84.1 ± 0.8	0.1 ± 1.2
Dec. 21, 1977	2.71	1134 ± 11 (7)	88.2 ± 0.9	4.2 ± 1.2
Dec. 28, 1977	3.71	1305 ± 13 (4)	76.7 ± 0.8	-7.4 ± 1.1
Jan. 5, 1978	4.86	1082 ± 11 (1)	92.4 ± 0.9	8.4 ± 1.2
Jan. 12, 1978	5.86	1049 ± 10 (8)	95.3 ± 1.0	11.3 ± 1.3
Jan. 19, 1978	6.86	980 ± 10 (1)	102.0 ± 1.0	18.0 ± 1.3
Jan. 26, 1978	7.86	885 ± 9 (4)	113.0 ± 1.1	29.0 ± 1.4
Feb. 1, 1978	8.71	820 ± 8 (1)	121.9 ± 1.2	37.9 ± 1.5
Feb. 4, 1978	9.14	770 ⁴ ± 8 (3)	129.9 ± 1.3	45.9 ± 1.5
Feb. 9, 1978	9.86	746 ± 7 (2)	134.0 ± 1.3	50.0 ± 1.6
Feb. 17, 1978	11.00	733 ± 7 (1)	136.5 ± 1.4	52.5 ± 1.6
Feb. 24, 1978	12.00	713 ± 7 (1)	140.3 ± 1.4	56.3 ± 1.6

TABLE 2.6d (cont.)

Date	Time	$\langle h \rangle^3$	$\langle l/h \rangle$	$\Delta \langle l/h \rangle \equiv R_f$
	Weeks	$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
Mar. 2, 1978	12.86	664 $\pm$ 7 (1)	150.5 $\pm$ 1.5	66.5 $\pm$ 1.7
Mar. 11, 1978	14.14	614 $\pm$ 6 (1)	162.8 $\pm$ 1.6	78.8 $\pm$ 1.8
May 8, 1978	22.43	501 $\pm$ 5 (2)	199.7 $\pm$ 2.0	115.7 $\pm$ 2.2
May 15, 1978	23.43	490 $\pm$ 5 (1)	204.2 $\pm$ 2.0	120.2 $\pm$ 2.2
May 19, 1978	24.00	491 $\pm$ 5 (1)	203.6 $\pm$ 2.0	119.6 $\pm$ 2.2
May 25, 1978	24.86	502 $\pm$ 5 (1)	199.1 $\pm$ 2.0	115.1 $\pm$ 2.2
June 2, 1978	26.00	616 $\pm$ 6 (2)	162.2 $\pm$ 1.6	78.2 ⁵ $\pm$ 1.8
June 8, 1978	26.86	625 $\pm$ 6 (1)	159.9 $\pm$ 1.6	75.9 $\pm$ 1.8
June 16, 1978	28.00	608 $\pm$ 6 (1)	164.4 $\pm$ 1.6	80.4 $\pm$ 1.8
June 22, 1978	28.86	611 $\pm$ 6 (1)	163.6 $\pm$ 1.6	79.6 $\pm$ 1.8
June 29, 1978	29.86	604 $\pm$ 6 (5)	165.4 $\pm$ 1.7	81.4 $\pm$ 1.9
June 30, 1978	30.00	613 $\pm$ 6 (2)	163.1 $\pm$ 1.6	79.1 $\pm$ 1.8
July 7, 1978	31.00	616 $\pm$ 6 (3)	162.2 $\pm$ 1.6	78.2 $\pm$ 1.8
July 13, 1978	31.86	647 $\pm$ 6 (1)	154.5 $\pm$ 1.5	70.5 $\pm$ 1.8
July 20, 1978	32.86	638 $\pm$ 6 (1)	156.6 $\pm$ 1.6	72.6 $\pm$ 1.8

FOOTNOTES FOR TABLE 2.6d

- 1) Flow velocities are averages of Ramapo and TSK data taken in the time intervals indicated. The Ramapo flow meter for Unit D has behaved erratically throughout the experiment. Some of the data were normalized using the data from the Unit B Ramapo flow meter (Units B and D are plumbed in series). Flow velocity changes were caused by changes in plumbing necessitated by pump failures.
- 2) The Wilson plot data were taken before deployment using fresh water and the buoy electronics. Data were taken at two velocities, four cooling curves at each velocity. Errors on the Wilson plot are determined by assuming 1% errors on the velocity measurement. Values to be expected are: $1/h = 0.344 \times 10^{-2} V^{-.8}$ hr ft² °F/BTU (V in the ft/sec).
- 3) Errors on h are given as 1% followed by the error on the mean of ~16 cooling curves. Errors on 1/h and R_f are calculated from 1% value.
- 4) Average of two cooling curves.
- 5) The change in R_f between May 25 and June 2 is very large and unexpected. Some possible sources of this change are:
 - 1) Changes in the fouling layer caused by changes in flow conditions
 - 2) Changes in the fouling layer caused by some other unknown factor

We are hopeful that further analysis will enable us to understand this change.

TABLE 2.7a

SUMMARY OF DATA FROM SUBMERGED BUOY AT KEAHOE POINT, HAWAII

Date	Time Weeks	$\langle h \rangle$	$\langle l/h \rangle$	$\Delta \langle l/h \rangle \equiv R_f$
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
8 Nov 78	.29	1018 $\pm$ 3	98.2 $\pm$.3	0.0 $\pm$.3
14 Nov 78	1.14	1029 $\pm$ 3	97.2 $\pm$.3	-1 $\pm$.4
20 Nov 78	2.00	1014 $\pm$ 2	98.6 $\pm$.2	0.4 $\pm$.4
27 Nov 78	3.00	808 $\pm$ 1	123.7 $\pm$.2	25.5 $\pm$.4
3 Dec 78	3.86	951 $\pm$ 4	105.2 $\pm$.4	7.0 $\pm$.5
11 Dec 78	5.00	922 $\pm$ 5	108.5 $\pm$.6	10.3 $\pm$.7
18 Dec 78	6.00	887 $\pm$ 2	112.8 $\pm$.3	14.6 $\pm$.4
26 Dec 78	7.14	849 $\pm$ 5	117.7 $\pm$.6	19.5 $\pm$.7
2 Jan 79	8.14	798 $\pm$ 1	125.2 $\pm$.2	27.0 $\pm$.4
5 Jan 79	8.57	1033 $\pm$ 1	96.8 $\pm$.1	-1.4 $\pm$.3
8 Jan 79	9.00	1047 $\pm$ 3	95.5 $\pm$.3	-2.7 $\pm$.4
16 Jan 79	10.14	980 $\pm$ 3	102.1 $\pm$.4	3.9 $\pm$.5
22 Jan 79	11.00	925 $\pm$ 2	108.0 $\pm$.2	9.8 $\pm$.4
30 Jan 79	12.14	855 $\pm$ 2	117.0 $\pm$.3	18.8 $\pm$.4

TABLE 2.7a (cont.)

Date	Time Weeks	$\langle h \rangle$	$\langle l/h \rangle$	$\Delta \langle l/h \rangle \equiv R_f$
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
5 Feb 79	13.00	800 $\pm$ 1	124.9 $\pm$.2	26.7 $\pm$.4
13 Feb 79	14.14	989 $\pm$ 4	101.1 $\pm$.4	2.9 $\pm$.5
22 Feb 79	15.43	871 $\pm$ 2	114.8 $\pm$.3	16.6 $\pm$.4
6 Mar 79	17.14	740 $\pm$ 7	135.9 $\pm$ 1.4	36.8 $\pm$ 1.4
7 Mar 79	17.29	726 $\pm$ 2	137.6 $\pm$.4	39.4 $\pm$.5
31 May 79	29.43	602 $\pm$ 4	166 $\pm$ 1	67.8 $\pm$ 1
4 Jun 79	30.00	582 $\pm$ 2	171.8 $\pm$ 6	73.6 $\pm$.7
11 Jun 79	31.00	561 $\pm$ 13	178.3 $\pm$ 4	80.1 $\pm$ 4
18 Jun 79	32.00	556 $\pm$ 1	179.9 $\pm$.4	81.7 $\pm$.5
25 Jun 79	33.00	510 $\pm$ 3	195.9 $\pm$ 1	97.7 $\pm$ 1
9 Jul 79	35.00	439 $\pm$ 10	228 $\pm$ 5	130 $\pm$ 5
16 Jul 79	36.00	818 $\pm$ 7	122.3 $\pm$ 1	24.1 $\pm$ 1
27 Jul 79	37.57	575 $\pm$ 13	174 $\pm$ 4	75.8 $\pm$ 4
30 Jul 79	38.00	593 $\pm$ 4	168.5 $\pm$ 1	70.3 $\pm$ 1

TABLE 2.7b

SUMMARY OF DATA FROM SUBMERGED BUOY AT KEAHOLE POINT, HAWAII

Date	Time Weeks	<h>		<l/h>		$\Delta<l/h> \equiv R_f$	
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$		$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$		$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	
8 Nov 78	.29	1177	4	85.0	.3	0.0	.3
15 Nov 78	1.29	1177	7	85.0	.5	0.0	.8
20 Nov 78	2.00	1200	12	83.5	.8	-1.5	.9
29 Nov 78	3.29	1143	7	87.1	.5	2.1	.8
5 Dec 78	4.14	1126	4	88.8	.3	3.8	.6
12 Dec 78	5.14	1080	6	92.6	.5	7.6	.8
19 Dec 78	6.14	1075	5	93.0	.4	8.0	.7
26 Dec 78	7.14	1048	8	95.4	.7	10.4	.9
3 Jan 79	8.29	1023	6	97.8	.6	12.8	.8
9 Jan 79	9.14	973	3	102.8	.3	17.8	.6
17 Jan 79	10.29	895	4	111.8	.5	26.8	.8
23 Jan 79	11.14	858	3	116.6	.4	31.6	.7
31 Jan 79	12.29	779	2	128.3	.3	43.3	.6
6 Feb 79	13.14	749	2	133.6	.3	48.6	.6
13 Feb 79	14.14	1168	1	85.6	.1	0.6	.4
8 Feb 79	17.43	858	5	116.5	.7	31.5	

TABLE 2.7c

SUMMARY OF DATA FROM SUBMERGED BUOY AT KEAHOLE POINT, HAWAII

Date	Time Weeks	$\langle h \rangle$	$\langle 1/h \rangle$	$\Delta \langle 1/h \rangle \equiv R_f$
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
9 Nov 78	.43	1271 $\pm$ 8	78.7 $\pm$.5	0.0 $\pm$.5
16 Nov 78	1.43	1245 $\pm$ 3	80.3 $\pm$.2	1.6 $\pm$.5
21 Nov 78	2.14	1205 $\pm$ 4	83.0 $\pm$.3	4.3 $\pm$.6
30 Nov 78	3.43	1065 $\pm$ 3	93.9 $\pm$.3	15.2 $\pm$.6
7 Dec 78	4.43	1146 $\pm$ 4	87.3 $\pm$.3	8.6 $\pm$.6
13 Dec 78	5.29	1117 $\pm$ 3	89.5 $\pm$.2	10.8 $\pm$.5
20 Dec 78	6.29	1088 $\pm$ 4	91.9 $\pm$.3	13.2 $\pm$.6
27 Dec 78	7.29	1049 $\pm$ 4	95.3 $\pm$.4	16.6 $\pm$.6
3 Jan 79	8.29	1058 $\pm$ 2	94.5 $\pm$.2	15.8 $\pm$.5
10 Jan 79	9.29	1263 $\pm$ 6	79.2 $\pm$.4	0.5 $\pm$.6
18 Jan 79	10.43	1244 $\pm$ 3	80.4 $\pm$.2	1.7 $\pm$.5
25 Jan 79	11.43	1222 $\pm$ 3	81.8 $\pm$.2	3.1 $\pm$.5
11 Feb 79	12.43	1196 $\pm$ 3	83.6 $\pm$.2	4.9 $\pm$.5
15 Feb 79	14.43	1043 $\pm$ 5	95.8 $\pm$.5	17.1 $\pm$.7

TABLE 2.7c (cont.)

Date	Time Weeks	$\langle h \rangle$	$\langle 1/h \rangle$	$\Delta \langle 1/h \rangle \equiv R_f$
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
9 Mar 79	17.57	1079 $\pm$ 7	92.6 $\pm$.6	13.9 $\pm$.8
31 May 79	29.43	814 $\pm$ 1	122.8 $\pm$.2	44.1 $\pm$.5
5 Jun 79	30.14	782 $\pm$ 3	127.9 $\pm$.5	49.2 $\pm$.7
12 Jun 79	31.14	711 $\pm$ 4	140.6 $\pm$.7	61.9 $\pm$.9
19 Jun 79	32.14	620 $\pm$ 2	159.0 $\pm$.4	80.3 $\pm$.6
26 Jun 79	33.14	577 $\pm$ 3	173.2 $\pm$.8	94.5 $\pm$.9
17 Jul 79	36.14	931 $\pm$ 7	107.4 $\pm$.8	28.7 $\pm$.9
10 Jul 79	35.14	527 $\pm$ 2	189.7 $\pm$.8	111.0 $\pm$.9
25 Jul 79	37.29	830 $\pm$ 10	120.5 $\pm$ 1.4	41.8 $\pm$ 1.5
31 Jul 79	38.14	867 $\pm$ 4	115.3 $\pm$.5	36.6 $\pm$.7

TABLE 2.7d

SUMMARY OF DATA FROM SUBMERGED BUOY AT KEAHOLE POINT, HAWAII

Date	Time Weeks	$\langle h \rangle$	$\langle 1/h \rangle$	$\Delta \langle 1/h \rangle \equiv R_f$
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
13 Nov 78	1	1022 $\pm$ 4	97.9 $\pm$.4	0.0 $\pm$.4
17 Nov 78	1.57	1001 $\pm$ 5	99.9 $\pm$.5	2.0 $\pm$.6
1 Dec 78	3.57	946 $\pm$ 5	105.7 $\pm$.5	7.8 $\pm$.6
14 Dec 78	5.43	887 $\pm$ 2	112.8 $\pm$.2	14.9 $\pm$.4
21 Dec 78	6.43	864 $\pm$ 2	115.8 $\pm$.3	17.9 $\pm$.5
27 Dec 78	7.29	814 $\pm$ 3	122.8 $\pm$.5	24.9 $\pm$.6
4 Jan 79	8.43	765 $\pm$ 2	130.7 $\pm$.3	32.8 $\pm$.5
5 Jan 79	8.57	955 $\pm$ 5	104.7 $\pm$.5	6.8 $\pm$.6
11 Jan 79	9.43	889 $\pm$ 3	112.5 $\pm$.4	14.6 $\pm$.6
19 Jan 79	10.57	765 $\pm$ 8	130.7 $\pm$ 1.3	32.8 $\pm$ 1.4
27 Jan 79	11.71	723 $\pm$ 3	138.3 $\pm$.5	40.4 $\pm$.6
2 Feb 79	12.57	688 $\pm$ 4	145.3 $\pm$.8	47.4 $\pm$.9
8 Feb 79	13.43	678 $\pm$ 1	147.4 $\pm$.3	49.5 $\pm$.5
14 Feb 79	14.29	876 $\pm$ 5	114.2 $\pm$.6	16.3 $\pm$.7

TABLE 2.7d (cont.)

Date	Time Weeks	$\langle h \rangle$	$\langle 1/h \rangle$	$\Delta \langle 1/h \rangle \equiv R_f$
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
1 Jun 79	29.57	451 $\pm$ 2	221.5 $\pm$ 1	123.7 $\pm$ 1.1
6 Jun 79	30.29	411 $\pm$ 8	244 $\pm$ 5	146 $\pm$ 5
13 Jun 79	31.29	412 $\pm$ 5	243 $\pm$ 3	145 $\pm$ 3
27 Jun 79	33.29	476 $\pm$ 5	210 $\pm$ 2	112 $\pm$ 2
6 Jul 79	34.57	466 $\pm$ 2	214 $\pm$.8	117 $\pm$.9
11 Jul 79	35.29	383 $\pm$ 2	261 $\pm$ 1.7	163.2 $\pm$ 1.7
18 Jul 79	36.29	950 $\pm$ 9	105.3 $\pm$ 1.0	7.5 $\pm$ 1.1
27 Jul 79	37.57	1011 $\pm$ 11	98.9 $\pm$ 1.1	1.1 $\pm$ 1.2

TABLE 2.8a

TUGBOAT DATA GULF OF MEXICO (GT)

Al 5052 Alloy: Nominal Flow = 3 ft/sec.

Period of Experiments: November 7, 1977 thru December 8, 1977

For seawater at 70° F flowing thru a 1" id
heat exchanger tube (Sch 40) at a nominal
velocity of 3'/sec.

$$h_{\text{expected}} = 662 \text{ BTU/hr ft}^{20}\text{F}$$

Date	Time Weeks	$\langle h \rangle$	$\langle 1/h \rangle \times 10^5$	$R_f \times 10^5$
		$\frac{\text{BTU}}{\text{hr ft}^{20}\text{F}}$	$\frac{\text{hr ft}^{20}\text{F}}{\text{BTU}}$	$\frac{\text{hr ft}^{20}\text{F}}{\text{BTU}}$
9 Nov 77	0.29	521.2	191.87	0.0
13 Nov 77	0.86	530.9 ± 4.7	188.40 ± 1.68	- 3.47
15 Nov 77	1.14	531.9	188.0	- 3.87
18 Nov 77	1.57	548.9 ± 10.7	182.25 ± 3.55	- 9.62
21 Nov 77	2.00	637.5 ± 0.9	156.90 ± .21	- 34.97
26 Nov 77	2.71	540.7 ± 2.6	184.94 ± .90	- 6.93
30 Nov 77	3.29			
5 Dec 77	4.00			
8 Dec 77	4.43			

TABLE 2.8b

TUGBOAT DATA GULF OF MEXICO (GT)

Al 5052 Alloy

Nominal Velocity: 6 ft/sec.

Period of Experiments: November 7 to December 10, 1977

Time (Weeks)	<h>	<1/h>	R_f
	$\frac{\text{BTU}}{\text{hr ft}^{20}\text{F}}$	$\frac{\text{hr ft}^{20}\text{F}}{\text{BTU} \times 10^5}$	$\frac{\text{hr ft}^{20}\text{F}}{\text{BTU} \times 10^5}$
0.4	954.3 $\pm$ 16.6	104.85 $\pm$ 1.80	0.0 $\pm$ 2.5
1.0	970.0 $\pm$ 2.5	103.09 $\pm$ 0.27	-1.76 $\pm$ 1.82
1.4	968.5 $\pm$ 6.5	103.27 $\pm$ 0.69	-1.58 $\pm$ 1.93
1.9	942.8 $\pm$ 4.1	106.08 $\pm$ 0.46	1.23 $\pm$ 1.86
3.0	997.3 $\pm$ 4.7	100.29 $\pm$ 0.47	-4.56 $\pm$ 1.86
3.9	975.2 $\pm$ 0.15	102.55 $\pm$ 0.02	-2.3 $\pm$ 1.80
4.7	967.9 $\pm$ 5.2	103.33 $\pm$ 0.55	-1.52 $\pm$ 1.88

TABLE 2.9a

SUMMARY OF DATA FROM OTEC-2 GULF OF MEXICO (GBI)

Nominal Flow Velocity: 6.0 ft/sec.

Material: Titanium (Grade 2)

Date 1978	Time (Weeks)	$\langle h \rangle$	$\langle 1/h \rangle$	R_f
		$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}}$
Mar 24	0.0	1099 $\pm$ 45	90.99 $\pm$ 3.73	0.0 $\pm$ 5.3
Mar 29	0.7	1101	90.83	-0.2
Pipes Cleaned				
Apr 6	1.9	1081	92.51	1.5
Apr 22	4.1	1085	92.34	1.4
Pump Failed				

TABLE 2.9b

SUMMARY OF DATA FROM OTEC-2 GULF OF MEXICO (GEI)

Nominal Flow Velocity: 6.0 ft/sec.

Material: Aluminum 5052

Date	Time	$\langle h \rangle$	$\langle 1/h \rangle$	R_f
1978	(Weeks)	$\frac{\text{BTU}}{\text{hr. ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
Mar 24	0.0	1074 $\pm$ 9.5	93.13 $\pm$ 0.82	0.0 $\pm$ 1.2
Mar 30	0.9	1007	99.30	6.2
Pipes cleaned				
Apr 4	1.6	1072	93.28	0.2
Apr 20	3.9	1060	94.34	1.2
Apr 25	4.6	1058	94.52	1.4
May 2	5.6	1049	95.33	2.2
May 10	6.7	1039	96.25	3.1
May 16	7.6	977	102.4	9.2

TABLE 2.9c

SUMMARY OF DATA FROM OTEC-2, GULF OF MEXICO (GBI)

Nominal Flow Velocity: 6.0 ft/sec.

Material: Aluminum 5052

Date	Time	$\langle h \rangle$	$\langle 1/h \rangle$	R_f
1978	(Weeks)	$\frac{\text{BTU}}{\text{hr ft}^2 \text{ } ^\circ\text{F}}$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$	$\frac{\text{hr ft}^2 \text{ } ^\circ\text{F}}{\text{BTU}} \times 10^5$
Mar 24	0.0	1031 $\pm$ 42	96.97 $\pm$ 3.99	0.0 $\pm$ 5.6
Mar 28	0.6	1003	99.70	2.7
Pipes Cleaned				
Apr 5	1.7	984	101.63	4.7
Apr 21	4.0	986	101.42	4.5
Apr 26	4.7	966	103.52	6.6
May 4	5.9	950	105.26	8.3
May 11	6.9	931	107.41	10.4
May 18	7.9	888	112.61	15.6
May 20-21	8.1	Pump off for 20-40 hours		
May 24	8.7	868	115.21	18.2
May 31	9.7	854	117.10	20.1

FIG 2.1A KII DATA UNIT A TI 6 FT/SEC

FIG 2.1B KII DATA UNIT B AL(5052) 6 FT/SEC

FIG 2.1C KII DATA UNIT C AL(5052) 6 FT/SEC

FIG 2.1D KII DATA UNIT D TI 6 FT/SEC

FIG 2.2A KIII DATA UNIT A AL(5052) 6 FT/SEC

FIG 2.2B KIII DATA UNIT B TI 6 FT/SEC

FIG 2.2C KIII DATA UNIT C 90/10 CU-NI 6 FT/SEC

FIG 2.2D KIII DATA UNIT D SS(AL-6X) 6 FT/SEC

FIG 2.3A GT DATA UNIT A AL(5052) 3 FT/SEC

FIG 2.3B GT DATA UNIT B AL(5052) 6 FT/SEC

FIG 2.4A GBI DATA UNIT A TI 6 FT/SEC

FIG 2.4B GBI DATA UNIT B AL(5052) 6 FT/SEC

FIG 2.4C GBI DATA UNIT C AL(5052) 6 FT/SEC

FIG 2.6b KII DATA UNIT B AL(5052) 6FT/SEC

FIG 2.6c KII DATA UNIT C AL(5052) 6FT/SEC

FIG 2.6d KII DATA UNIT D T1 6FT/SEC

FIG 2.7a KIII DATA UNIT A AL(5052) 6FT/SEC

FIG 2.7b KIII DATA UNIT B T1 6FT/SEC

FIG 2.7c KIII DATA UNIT C 90/10 CU-NI 6FT/SEC

FIG 2.7d KIII DATA UNIT D SS(AL-6X) 6FT/SEC

FIG 2.8a GT DATA UNIT A AL(5052) 3FT/SEC

FIG 2.8b GT DATA UNIT B AL(5052) 6FT/SEC

FIG 2.11 SEAWATER TEMPERATURE (KII)

FIG 2.12 SEAWATER TEMPERATURE (KIII)

FIG 2.13 SEAWATER TEMPERATURE (GT)

Section 2 References

1. B. Little, J. Morse, S. Loeb, and F. Spiehler, "A Biofouling and Corrosion Study of Ocean Thermal Energy Conversion (OTEC) Heat Exchanger Candidate Metals", presented at the Biofouling, Corrosion, and Materials Workshop, Washington D.C. (1979).

3. SOFTWARE DEVELOPMENT

3.1 New Analysis Program

A new version of our main analysis program was written. Two capabilities were added that make the program better able to handle the biofouling data. They are:

- 1) The program was given the ability to dynamically choose the starting point for the analysis window within a certain range. (The analysis window is 1 τ long.) Two and three parameter fits to the thermocouple data are made for each starting point and summary data are printed. The program then carries through the conversion to h, correction normalization, etc., for the two and three parameter τ resulting from the best of these fits.

The best of the fits is chosen according to specified criteria. For the first use of this program, the quantity used as a figure of merit is $\frac{\Delta\tau}{\tau} = \frac{\tau_2 - \tau_3}{\tau_3}$ where $\tau_{2(3)}$ is the time constant resulting from the 2(3) parameter fit to the thermocouple data.

- 2) The program produces a file of reduced data that is designed to be computer readable. This file contains all important results from each cooling curve analyzed and all the calibration constants used to reduce voltage signals to values of temperature, flow velocity, and $1/h$. These reduced data files are designed to be used for future correlation and statistical analysis of the data. Such analyses could be done using simple programs that run quickly and efficiently, and do not need the full power of the main analysis program.

Previous versions of the analysis program have used an analysis window 1.5 sec long which started 12 sec after the heat was turned off. Two and three parameter fits were made, and values of h reported were based on the two parameter τ . Cooling curves were eliminated if they were more than two standard deviations away from the average. We have not completed statistical studies to determine how is the best way to make cuts on the data produced by the new program, but as a working criterion we have been eliminating those curves where for the best fit $\frac{\Delta\tau}{\tau} > 2\%$. We find that very few curves are eliminated by this cut, and those that are have a large RMS difference between fitted and measured voltages.

The following table compares results obtained from the new program with those that would have been obtained from the old program. The following items are tabulated:

$\tau_2(12)$:	2 parameter τ for analysis window starting at $T = 12$ sec.
$\tau_3(12)$:	3 parameter τ for analysis window starting at $T = 12$ sec.
$\tau_2(\text{Best})$:	2 parameter τ for best analysis window.
$\tau_3(\text{Best})$:	3 parameter τ for best analysis window.
FRMS(12):	2 parameter RMS voltage difference between fit and measured values for window beginning at 12 sec.
FRMS(Best):	2 parameter RMS voltage difference between fit and measured values for best window.

Averages are from ≈ 16 cooling curves, errors are error on the mean for ≈ 16 cooling curves. Data from Keahole Point Series III.

Unit Date	$\tau_2(12)$	$\tau_3(12)$	FRMS(12)	$\tau_2(\text{Best})$	$\tau_3(\text{Best})$	FRMS(Best)
A 18 Dec 78	35.24 $\pm .04$	35.4 $\pm .1$	1.6	35.23 $\pm .05$	35.17 $\pm .08$	1.0
B 12 Dec 78	42.1 $\pm .1$	42.5 $\pm .3$	3.9	42.0 $\pm .1$	42.0 $\pm .1$	1.9
C 20 Dec 78	35.11 $\pm .04$	35.51 $\pm .07$	2.3	35.06 $\pm .04$	35.08 $\pm .06$	1.3
H 14 Dec 78	63.07 $\pm .07$	63.2 $\pm .3$	3.2	63.10 $\pm .07$	63.13 $\pm .06$	2.3

As can be seen, the $\tau_2(12)$, $\tau_2(\text{Best})$ are $\tau_3(\text{Best})$ values are in agreement in all cases. However, they disagree with the $\tau_3(12)$ value for units A, B, and C. In addition, the FRMS is reduced in all cases.

This indicates that with the new program we can place much more confidence in the results of the fitting routine. We also have a firm criterion to use in selecting the best starting point for the analysis window and also for selecting those curves where spurious data (caused perhaps by water temperature fluctuations) have affected the results. When we find large (>2%) differences between the best two and three parameter fits, we know something is wrong with the data for that curve, but when the two and three parameter fits agree to 1% or better, we can have confidence that we have extracted the true time constant from the data.

A complete program listing is included with this report (Appendix A). It is written in ANSI standard Fortran, with only a few (DEC-System 20) machine-dependent routines.

3.2 TI-59 Programs

A program was written for the TI-59 calculator/printer which allows one to quickly reduce field data and obtain an estimate of the value of $1/h$ normalized to 70°F and some standard velocity. Copies of three versions of the program are included. Version 1.1 (Appendix B1) gives a complete description of the program and its use. The constants stored in the data registers are for the KP II experiments. Version 1.2 (Appendix B2) has the appropriate constants for the NDBO experiments begun in September, 1978. (Use the handwritten values for the coefficients for the aluminum unit.) Versions 1.1 and 1.2 assume that the thermocouple information is input in the form of zero-corrected logs $[\ln(V_i - V_{\infty})]$. Version 2.0 (Appendix B3) shows how Version 1.2 is modified to accept voltages instead of zero-corrected logs.

3.2.1 Comparing our Standard Fortran Analysis Program to the TI-59 Program.

There are two major differences between the reductions done by the TI-59 program and our standard analysis program. The first is the way the log data is weighted in the fit. The TI-59 program weights all logs equally. In the standard analysis program the logs are weighted by the expression $(V_i - V_{\infty})^{-2}$. The other major difference between the programs is the way air and wall corrections are calculated. For a description of how this is done in our standard analysis program, see our design document (DOE Report No. C00-4041-10). In the TI-59 program,

$R_{\text{air}} = \alpha/h$ and $R_{\text{walls}} = \beta/\sqrt{h}$. Appendices B4 and B5 give values of α

and β for each unit and show how these approximations are justified. All other calculations by the TI-59 program are done exactly as they are done in the standard analysis program and with equal or higher precision.

3.2.2 Comparing results of the two analysis programs

The following table shows the results of using both programs to analyze four runs from the DP II experiments. Column (4) gives the results from the TI-59 program. Column (5) shows the results of the standard analysis program run on the DEC-20. Column (6) shows the $1/h$ obtained by the TI-59 program using the τ calculated by the standard analysis program.

(1) Unit	(2) Date	(3) Run #	(4) $\frac{1}{h} \times 10^5$ (TI-59)	(5) $\frac{1}{h} \times 10^5$ (standard)	(6) $\frac{1}{h} \times 10^5$ (standard τ and TI-59)
A(Ti)	7 Dec 77	10	88.0	88.7	88.3
A(Ti)	7 Mar 78	9	139.4	139.5	139.2
C(Al)	6 Dec 77	7	84.8	84.3	85.0
C(Al)	27 Feb 78	5	162.1	162.8	162.8

The results of these two different analysis programs can be compared in another way. Using the data in the table above we can calculate and compare $\Delta 1/h$ for each method of analysis. The last column shows $\Delta 1/h$ for six cooling curves on each of the two days using the standard analysis program. All the values are for one cooling curve per day.

Unit	$\Delta \frac{1}{h} \times 10^5$ (TI-59)	$\Delta \frac{1}{h} \times 10^5$ (standard)	$\Delta \frac{1}{h} \times 10^5$ (standard τ and TI-59)	$\Delta \frac{1}{h} \times 10^5$ (standard; 6 cooling curves)
A	51.4	50.8	50.9	51.5 $\pm$ 1.6
B	77.3	78.5	77.8	79.6 $\pm$ 1.9

Because of our lack of knowledge about flow velocity during the KP II experiments, the error on h for the standard analysis of a full day's cooling curves is set at 1%. The errors given in the last column above follow from the error on h .

4. INTERPRETATION AND IMPLICATIONS OF OTEC BIOFOULING DATA

In this section we wish to discuss the nature of the biofouling data which have so far been obtained, and to consider its implications. It is important to try to obtain an understanding of the underlying phenomena which determine the fouling process. This is so for several reasons. First, it could lead us to devise effective means of inhibiting and removing fouling, as well as help us to avoid wasting effort on methods which clearly will not work. Second, it could help us to anticipate unexpected future problems. Third, it should enable us to know which are the critical experiments to do, and how they should be conducted.

The experimental apparatus and procedures have been discussed at length previously, and will not be described here. Similarly, the data have been described in detail elsewhere. We will therefore consider only its dominant qualitative aspects. Those who are not familiar with the apparatus and the data are referred to earlier reports.¹

4.1 Observations

In studying the data, we consider, for the moment, only that portion of an experiment before any long-term (≥ 2 hrs) pump stoppages. (We consider the effect of such stoppages later.) When we do this, we find a remarkable simplicity in the results. The fouling thickness builds generally as shown in Fig. 4.1. In each case, starting with a new heat exchanger tube at $t=0$, we find the fouling building at rates which vary from experiment to experiment, and, within an experiment, from time to time. However, we find that invariably, once the fouling

thickness reaches $\sim 20\mu\text{m}$, the growth becomes linear with time.

Furthermore, the rate of growth in the linear region is roughly independent of all parameters which have been studied so far. The ranges of variation of these parameters are indicated in Table 4-1. It is worth emphasizing especially that the linear fouling rate is independent of flow velocity from 3-6 ft/sec, and of seawater bacterial levels from 10 per ml to 20,000 per ml.

Next, we look at the effects of pump shut-downs. We first exclude all data coming after a long pump shut-down. The effect on fouling growth of short (≤ 2 hrs) shut-downs is illustrated in Fig. 4.2. We find that short shut-downs do not significantly change the fouling phenomenon, except to increase slightly the rate of fouling. The increase is greater the greater is the fractional down time.

Long pump shut-downs (≥ 1 day) are found to lead to a new phenomenon: macrofouling. In all cases when there were no long pump shut-downs, we have found, upon disassembly of the apparatus, no evidence of macrofouling (e.g., barnacles). On the other hand, in every case when macrofouling has been found, it was after such a long shut-down.

Finally, of course, one of the most important observations made so far is that mechanical cleaning works. It is apparently easy to keep a tube nearly free of biofouling by at least one standard method of mechanical cleaning (M.A.N. brushes).

4.2 Interpretation

As we will show, it appears that most of the observed characteristics of biofouling can be explained by a simple physical model of the process.

Of course, one cannot say, just because the model is consonant with the data, that it is correct in all its details. Nevertheless, it works within the limits of the available data; it can be useful in planning and predicting; and it no doubt has at least some truth imbedded in it. For these reasons we present it here in spite of the very limited amount of data available.

In the initial period of the fouling of a clean surface, biologists believe, organisms do not attach effectively. It is apparently first necessary that a microscopic layer of non-living but organically derived material be adsorbed first. Only then can small organisms (such as bacteria) attach effectively.

In this initial period, often called the induction period, two processes are clearly important in fouling buildup: adsorption and multiplication (i.e., growth of organisms already attached). It is possible to write a mathematical expression, representing the physical and biological processes involved. However, the model is found to be dependent on many factors which we do not have enough data to evaluate at this stage. Therefore, we do not discuss the model quantitatively (however, see below), merely pointing out that at this stage, the rate of fouling buildup should depend on the character of the seawater, especially on its living and non-living organic content, on the flow velocity, and on many other factors.

At the end of the induction period the phase of linear fouling growth is initiated. This process is independent of most parameters, which implies it is a very simple and basic process, and one which should be amenable to quantitative understanding.

In developing a physical model, we again consider, as in the induction period, that adsorption and multiplication will be the important phenomena in the growth of the fouling layer. In general, of course, both will be happening simultaneously. However, to simplify the discussion, we will consider each as if it were occurring alone. To simplify further, we will represent the variety of organisms involved by a single "average" one having characteristics (such as diameter, etc.) typical of the group. (These simplifications are not essential. However, it is not worthwhile here to consider the most general form of the model, especially in light of the small amount of available data.)

First consider adsorption. It is easy to show* that the fouling should, in this process, build up according to the relation:

$$F = b \ln \left[\frac{\epsilon_o}{\epsilon_f} e^{t/\tau} + \left(1 - \frac{\epsilon_o}{\epsilon_f} \right) \right] . \quad (1)$$

Here, F is the thickness of the fouling layer, b is the dimension of a typical fouling organism measured perpendicular to the wall. The factors ϵ_o and ϵ_f are, respectively, the "sticking factors" for a bare, and for a fouled surface. (That is, ϵ measures the likelihood that an organism striking the wall will stick. If $\epsilon=0$, it never sticks; if $\epsilon=1$, it always sticks. In general, $0 \leq \epsilon \leq 1$, and $\epsilon_f > \epsilon_o$.) And

* We do not here present the detailed calculations leading to this and later results, in order not to complicate the discussion excessively. The complete details will appear in a paper now being prepared for publication. The author will be happy to supply these before publication to anyone interested.

τ is given by

$$\tau = D_p / 0.024 a b \epsilon_f D n_b (Pr) (Re)^{0.8} \quad (2)$$

where D_p is the diameter of the pipe or tube being studied, a is the cross-sectional area of the pipe wall taken up by one fouling organism*, D is the diffusion constant of an organism through water, n_b is the volume concentration of organisms in the local water, Pr is the Prandtl number and Re is the Reynolds number of the flow.

We see that Eq.(1) is not a linear function of time. However, it is seen to become linear asymptotically, where it reduces to

$$F = (0.024 a b \epsilon_f D n_b / D_p) (Pr) (Re)^{0.8} t + b \ln (\epsilon_o / \epsilon_f). \quad (3)$$

Of primary interest in this expression are the following observations. The rate of fouling in the linear region (i.e., the coefficient of t in Eq.(3)) depends on the characteristics of the responsible organisms through the factors a , b , ϵ_f , and D , none of which are very precisely known. Further, it is proportional to the volume concentration of organisms (n_b), and to the 0.8 power of the flow velocity (through the factor $(Re)^{0.8}$). These dependences are in direct contradiction to the observed results. We suspect, therefore, that in the linear region of biofouling growth, adsorption of new organisms onto the wall is unimportant. The dominant mechanism of biofouling growth, then, must be multiplication, or growth, of those already attached.

* Note that the space taken up by one organism may be larger than its size as a result, e.g., of exudation of material.

Now we consider the multiplication of attached organisms. We assume that a typical organism will multiply with a generation time τ_2 . We expect that, because of the availability of nutrients and the ability to be rid of waste products, the organisms nearest the surface will multiply fastest. The value of τ_2 , then, will depend upon depth in the fouling film. Again, we simplify by assuming that all organisms within a depth d_m multiply with the same generation time τ_2 , and all those at greater depth to not multiply at all.* This picture is, of course, appropriate only after the fouling thickness is greater than d_m ; i.e., in the linear growth region. One easily determines then that the fouling will build according to the relation

$$F = d_m (1 + t/\tau_2). \quad (4)$$

This result has several important characteristics. Perhaps most importantly, it does not depend on n_b , the concentration of organisms in the water. Also, it will not depend upon flow velocity if τ_2 is not nutrient-limited at the lowest observed flow velocities. Eq.(4) is clearly in agreement with the remarkable observation that the rate of fouling growth does not depend on any of the factors given in Table 4-1.

To summarize, the observed data lead us to a model of biofouling in which we divide the process into two periods. In the first, the induction period, the clean surface is covered with living and non-living organic material up to a depth of at least d_m . The fouling rate during this time depends on many factors including the nature of the organisms, the nature

* It is easy to see that not all the attached organisms can multiply at the same rate, for this leads to an exponential growth contrary to what is observed.

of the surface, the concentration of organisms, and the flow velocity. In the second, the linear period, the fouling rate becomes constant at a level which depends only on two factors: the depth, d_m , to which the organisms remain prolific (multiply), and their average generation time, τ_2 .

We turn now to consider what we can deduce about biofouling under OTEC conditions using this model.

First, by comparing with the data we can obtain at least approximate values of the important numbers d_m and τ_2 . The value of d_m must be of the order of the fouling thickness at the point of transition between the induction and linear periods. This is found to be (very roughly)

$$d_m \cong 20 \text{ } \mu\text{m} .$$

Since the universal slope of the fouling curve is (very roughly) 10 $\mu\text{m}/\text{week}$, we conclude that the average generation time is

$$\tau_2 \cong 2 \text{ weeks} .$$

We reemphasize that this is an average generation time within the outer 20 μm of the fouling layer. At the top of the layer it is presumably much shorter than this, while it becomes very long deeper in.

We note that this model allows us to say something about the effective size* of the responsible organisms. A typical dimension must be less than 20 μm . If the actual size of the organisms is 20 μm , then the prolific layer is just one organism deep. On the other hand, they may be

* Here, "effective size" refers to the mean separation between organisms in the fouling layer. Because of exudations, possible water spaces, etc., this may be significantly larger than the "bare size" of the organism as it floats freely.

smaller. If for example, they are 2 μm in size, then the prolific layer is 10 organisms deep.

In fact, the preliminary microscopic observations² of the fouling layer are consistent with the dominant organisms being bacteria with typical bare sizes well below 20 μm .

We have already alluded to the fact that apparently macrofoulers, such as barnacles, do not attach to the wall as long as the water is flowing (at least at speeds greater than 3 ft/sec). How can we understand this? In fact, it is easy to understand at least semiquantitatively through the application of some simple physical principles.

Next to the wall, the flow is laminar with a velocity profile approximately linear near the wall (with, of course, $v=0$ at the wall). The laminar layer is, for example, at a bulk flow velocity of 6 ft/sec, about 100 μm thick. When an organism attaches to the wall, the water exerts a drag force on it, parallel to the flow, of approximately

$$F_d = 6 \pi r^2 \mu \frac{dv}{dy} , \quad (5)$$

where r is a "typical radius" of the organism, μ is the viscosity of water, and $\frac{dv}{dy}$ is the velocity gradient at the wall. To counteract this, the wall will exert a force of adhesion on the organism. It is a reasonable guess that this force of adhesion is proportional to roughly the first power of r (think, for example, of stripping a piece of tape from a surface). So we write for the force of adhesion

$$F_a = sr . \quad (6)$$

Here, s is a "stickiness factor" (not to be confused with the sticking factor, ϵ , defined earlier) which determines how tightly the organism

can cling to the wall. The important thing is that if $F_d > F_a$, the organism will not stick. We get the limiting case then by considering $F_d = F_a$ which yields

$$\frac{dv}{dy} = \frac{s}{6\pi\mu r} \quad (7)$$

But we also have, from hydrodynamic theory,

$$\frac{dv}{dy} = e v_b^{1.8}, \quad (8)$$

where e is a numerical constant depending on the properties of water, and v_b is the bulk flow velocity.

Combining Eq.(7) and (8), and using the appropriate numerical values, we get

$$\left[\frac{v_b}{10 \text{ ft/sec}} \right]^{1.8} = \frac{1}{4.18 \times 10^3 \text{ dynes/cm}^2} \frac{s}{r} \quad (9)$$

It is worthwhile, perhaps, to emphasize the significance of Eq.(9). It says that for a given organism (i.e., given s and r), there is a critical flow velocity given by v_b in Eq.(9), such that, if the flow velocity exceeds v_b , the organism cannot stick, and if it is less than v_b , it will stick. Given, for example, barnacle larvae of known size r , and stickiness s , Eq.(9) could be used to calculate the highest velocity, v_b , at which barnacles could attach.

Of course, we do not know the stickiness of barnacle larvae. If we did the right experiments we could determine s . That is, for barnacle larvae of known size r , vary the velocity v_b until the critical value is found. Then Eq.(9) could be used to calculate s .

Or, we could look at Eq.(9) from another point of view. Consider a variety of organisms of various sizes r , all having roughly the same stickiness s . At a given flow velocity v_b , we could then solve Eq.(9)

for the critical size, r . Organisms larger than this could not stick, while those smaller would. Thus, the fact that we have a continuous flow past the wall serves as a "size filter" for sticking organisms, much as an ordinary filter works: those larger than the critical size pass through while the smaller ones stick.

The preceding discussion leads naturally to consideration of the observation that if the flow is interrupted (by pump shut-downs) for a sufficiently long time (much longer than an hour, less than 2-3 days), barnacles are observed to attach and grow. This is easily explained in terms of Eq.(9), since, when the pumps are off, $v_b = 0$, which is obviously below the critical velocity for barnacle larvae whatever it may be.

On the other hand, we observe that some pump shut-downs do not lead to barnacle growth: namely, those which are short (e.g., ≤ 2 hrs). According to Eq.(9), barnacle larvae will surely attach at such times. The apparent anomaly is removed if we invoke a phenomenon well known to biologists. That is, that a barnacle (as well as other organisms) holds tighter to a surface the longer it has been attached. This is presumed to be due to exudates which serve to bind the organism to the surface. That is to say, in terms of Eq.(9), that the "stickiness", s , of an organism is a function of time: $s = s(t)$. It is interesting to note that we could measure this time dependence if we did the obvious experiments. All that we know from the limited data so far available is that if $v_b = 0$ for a time less than 1-2 hrs, any attached barnacles are washed off when flow is resumed if $v_b \geq 3$ ft/sec, while if $v_b = 0$ for several

days, they are not washed off afterwards if $v_b \leq 6$ ft/sec.

All of this clearly also relates to another phenomenon mentioned earlier. That is that the rate of fouling in the linear period increases with the fractional pump down-time, for short pump stoppages. Again, this can be understood in terms of Eq.(9). During the stoppage ($v_b=0$) all organisms can attach regardless of size. During the brief stoppage, their stickiness, s , can increase only a little. When the pumps are restarted, then the very large organisms are washed off. But some, somewhat larger than those which normally attach, have sufficiently larger values of s , so that they remain even after the pumps are started. In effect, then, frequent short pump stoppages have the result that a greater fraction of the local organisms, in general larger ones, contribute to the overall fouling rate. The new population of organisms apparently either increases d_m , or decreases τ_2 , or both. Why this happens is as yet unclear.

There have been suggestions that chemotaxis might be important in the process of fouling in these tests. This now appears not to be so. We have seen that in the important period, when the fouling is growing linearly, adsorption is not important; the growth is dominantly by multiplication of those already attached. Even in the induction period, when adsorption is important, it is unlikely that chemotaxis plays a role. The reason for this is to be found in the dynamics of the motion of small objects (such as bacteria) through the laminar flow layer near the wall. Since the object finds itself in a shear field due to the velocity gradient, it will be rotating. The rate of rotation will be

approximately

$$\omega = \frac{1}{2} \frac{dv}{dy} \quad . \quad (10)$$

We note that, at a bulk velocity of 6 ft/sec, for example, this corresponds to an angular speed of about $10,000 \text{ sec}^{-1}$. It would be remarkable if a bacterium spinning at this rate could tell where it was going, or do anything about it if it could.

4.3 Future Plans

There are many experiments we would like to see done in order to test the model, to extend its range of validity, to refine it, and to evaluate some of the unknown factors. Some of the more interesting, for example, would be to measure the stickiness s and its time dependence, to extend the range of velocity covered in the experiments, to study the temperature dependence of these phenomena, and in general to learn more about the characteristics of the organisms involved, such as type, bare size, effective size, concentration in the water, etc. It should be noted that many of these experiments could be done more cheaply and easily, and with more controlled conditions, in the laboratory rather than in the field.

We turn now to consider some potentially important problems for OTEC which are suggested by the data and by this picture.

It would be of interest to have data over a wider velocity range. The velocity/critical-size effect discussed here implies that there is a velocity above which even the very small organisms responsible for microfouling will not attach. If this velocity is not too high, it

might be feasible to run heat exchangers above the critical velocity if, in the overall optimization, the increased costs of high velocity operation are counterbalanced by decreased costs of antifouling methods.

More information on the stickiness vs. time of barnacle larvae (and other macrofoulers) might be useful. We know that if flow is interrupted for less than about two hours, barnacles will not permanently attach, while if the interruption is longer than several days, they will. Operators of an OTEC plant will need guidelines to tell them how long is a safe shut-down before antimacrofouling measures must be instituted.

Studies at lower velocities might be important in order to determine the critical velocity for barnacles and other macrofoulers. This is so because there may be locations within a heat exchanger where, even at high nominal velocities, the actual local velocity may be very low. This would be so, for example, at the downstream end of a heat exchanger tube if there is a sudden step in tube size. Barnacles attaching at such points could grow to sizes which would close the end of the tube. Measures such as M.A.N. brushing are not likely to remove such large organisms. If the critical velocity is known, one could design the flow paths to avoid the dangerous regions. If this is not possible, then specific antimacrofouling measures may be needed in the affected areas.

There are related problems involved in using non-cylindrical heat exchanger geometries such as plate-fin. Depending on the geometry, there can be regions (e.g., in corners) where the shear stress in the flow

becomes very low. If so, macrofoulers can attach and grow. This would obviously complicate the antifouling problem.

So far, we have no information at all on deep (cold) water biofouling. A popular opinion is that biofouling in the condenser will be minimal or non-existent because there are few organisms living at 1000 meters depth. However, the concentration of organisms affects only the rate of biofouling in the induction period. In the more important linear period, the fouling rate depends only on the generation time of the attached organisms, not on their concentration in local waters. Therefore the build-up after the induction period is over will likely be as fast as in the surface waters.

4.4 Conclusion

We have seen that a simple physical model appears to explain most of the characteristics of OTEC heat exchanger biofouling. Naturally, with so little data in hand, we cannot have great confidence that all aspects of the model are correct. Still, it can serve as a useful guide in planning. It would be desirable to get some of the basic data necessary to test the model, and to refine it. It could help to save much effort and perhaps avoid future mistakes.

TABLE 4-I
PARAMETERS OF
CMU OTEC BIOFOULING
EXPERIMENTS

Expt.	Location	Date	Nominal Velocity (ft/sec)	Material	Pressure Change (ft of H ₂ O)	Bacterial Counts
NOI'I	Keahole Point	Jul-Sep 1976	3	Al 6061-T6	35	High*
NOI'I	Keahole Point	Aug-Sep 1976	6	Al 6061-T6	35	High*
BUOY SERIES I	Keahole Point	Feb-Jul 1977	3	Al 6061-T6	3	High**
BUOY SERIES I	Keahole Point	Feb-Jul 1977	6	Al 6061-T6	6	High**
BUOY SERIES I	Keahole Point	Feb-Apr 1977	6	Ti Grade 2	6	High**
ST. CROIX	St. Croix	Jul-Sep 1977	3	Al 6061-T6	75	Low ⁺
ST. CROIX	St. Croix	Jul-Sep 1977	6	Al 6061-T6	75	Low ⁺

* Harvey² reports approximately 20,000 bacteria/ml in addition to diatoms, glass fibers, and glass chips in water taken from near the intakes for the Noi'i experiments.

** No measurements have been reported for water from the vicinity of the buoy, but we expect the values to be similar to those found near Noi'i.

+ Hirshman et al.³ report bacterial populations of below 10/ml in the St. Croix intake water. See also Aftring et al.⁴

FIGURE CAPTIONS

- Fig 4.1 A qualitative representation of typical biofouling data. F represents the apparent thickness of the biofilm, t is time. It is generally observed that the fouling grows at a rate which varies with conditions until it is about $20\text{ }\mu\text{m}$ thick. Thereafter the growth is approximately linear for as long as has so far been observed.
- Fig 4.2 The dependence of fouling rate on short (≤ 2 hr) pump stoppages. The ordinate, dR_f/dt , is the slope of the fouling growth curves in the linear region (corresponding to the parts of the curves above the dashed line in Fig. 1). The ordinate is the total pump down time per month. The experiments from which the data were taken are indicated. We see that the fouling rate increases with down time. However, as long as the pumps are down less than a total of 1-2 hrs/mo, the effect on the results is very small.

FIGURE 4.1

FIGURE 4.2

REFERENCES TO SECTION 4

1. J. G. Fetkovich, G. N. Grannemann, D. L. Meier and C. W. Fette, "A Novel Method of Measuring Heat Transfer Coefficients with High Precision," published in Proceedings of the Twelfth Annual SouthEastern Seminar on Thermal Sciences, Charlottesville, VA, 1976.

J. G. Fetkovich, G. N. Grannemann, D. L. Meier and F. C. Munchmeyer, "Heat Transfer Problems in an Ocean Thermal Power Plant," published in Proceedings of the Twelfth Annual SouthEastern Seminar on Thermal Sciences, Charlottesville, VA, 1976.

J. G. Fetkovich, C. W. Fette, R. W. Findley, G. N. Grannemann, L. M. Mahalingam, D. L. Meier and P. D. Runco, A System for Measuring the Effect of Fouling and Corrosion on Heat Transfer Under Simulated OTEC Conditions, DOE Report No. COO-4041-10 (1976).

J. G. Fetkovich, G. N. Grannemann, L. M. Mahalingam, D. L. Meier and F. C. Munchmeyer, "Degradation of Heat Transfer Rates Due to Biofouling and Corrosion at Keahole Point, Hawaii," published in Proceedings of the Ocean Thermal Energy Conversion (OTEC) Biofouling and Corrosion Symposium, Seattle, WA, 1977.

J. G. Fetkovich, G. N. Grannemann, L. M. Mahalingam and D. L. Meier, "Measurements of Biofouling in OTEC Heat Exchangers," published in Proceedings of: The Fifth Ocean Thermal Energy Conversion Conference, Miami, FL, 1978.

2. G. W. Harvey, "Biofouling Experiments at Keahole Point, Hawaii, 1976 Biological Studies Progress Report," unpublished.

L. R. Berger and F. C. Munchmeyer, Summary Report of OTEC Biology October 1, 1977 to January 15, 1978, submitted to Battelle Pacific Northwest Laboratories, 1978.

3. J. Hirshman, D. L. Meier, R. S. C. Munier and B. F. Taylor, "Introduction to the St. Croix Biofouling and Corrosion Study," published in Proceedings of the Ocean Thermal Energy Conversion (OTEC) Biofouling and Corrosion Symposium, Seattle, WA, 1977.

4. R. P. Aftring, D. C. Capone, L. Duguay, J. W. Fell, I. M. Master and B. F. Taylor, "Biofouling and Site Characterization Studies in an Ocean Thermal Energy Conversion (OTEC) Experiment at St. Croix, U. S. Virgin Islands," published in Proceedings of: The Fifth Ocean Thermal Energy Conversion Conference, Miami, FL, 1978.

APPENDIX A

A copy of the latest version of the data
analysis (Fortran) program

[illegible][illegible]

0378

· 0378

0378

P474
GG20

GG20

20-JUL-79 13:27:23

PS: <GG20>BIG. FOR. 2

GG20

P474
: GG20

0378

0378

```

*** BEGIN ***
**** BEGIN ****
***** BEGIN *****
***** BEGIN *****
***** BEGIN *****
***** BEGIN *****
***** BEGIN *****
***** BEGIN *****
***** BEGIN *****

```

[illegible]

```
##### BEGIN #####
##### Carnegie-Mellon University Tops-C, TOP5-20 Monitor 4(2772)-2 #####
```

Y.

*****BRXNUM*****

```

SUBROUTINE BRNEWR(ICODE, ICHAN, IDATA, IPASS, AVEFMZ, NBR, M, ISET, NBR1)
DIMENSION AKEEP(30,16,3)
DIMENSION KEEP(30,16,3)
EQUIVALENCE(KEEP, AKEEP)
COMMON /BROUT/AKEEP
DIMENSION ICHAN(1500), IDATA(1500) !Channel in parallel with data
DIMENSION KSTORE(500) !Storage for FMZ data
DIMENSION N(80)
INTEGER IUNIT !Temporary unit number
INTEGER IFMZC !Number of sets of FMZ's read
INTEGER K !Number of FMZ points in KSTORE
INTEGER KKOUT !No apparent purpose
INTEGER NPCARD !Number of data points on one card
INTEGER NPOINT !Number of data points read
INTEGER IBEGIN !Temporary, first data point to read from card
INTEGER ISAVE !Final count of FMZ points saved
INTEGER IPASS !Pass 1 or 2
INTEGER ICODE !Returns code indicating errors
! 0 for no error
! 1 for severe error (give up on file)
! 10 for unexpected EOF

INTEGER ISET !Number of cooling curve to read, indexes KEEP
INTEGER M !Input unit number
INTEGER NBR !Returned unit number
INTEGER NBR1 !Returned number of data points
INTEGER COUNT !Number of
INTEGER LBLANK !A Blank ( ' ' )
INTEGER ATCT
INTEGER AUNT
INTEGER AX, AX1, AX2
INTEGER ACOM, ATID, ABTM, AFMZ, AFME, ASSV, ARUN, ACCU, AUC, AUB, AUA
INTEGER AU1, AU2
INTEGER ASIT
INTEGER AUD, AUE, AUF, AERR, A999, AEND, ATIM
INTEGER SANC
INTEGER ATCA
INTEGER AFVW
DATA AFVW/4HFVW=/
DATA ATCA/4HTCA=/
DATA LBLANK/1H /
DATA L0/1H0/
DATA L1/1H1/
DATA L2/1H2/
DATA L9/1H9/
DATA ATID/4HTID / !Marks tape ID
DATA AUA/4HA / !Unit labels A-F, 1-2
DATA AUB/4HB /
DATA AUC/4HC /
DATA AUD/4HD /
DATA AUE/4HE /
DATA AUF/4HF /
DATA AU1/4H 1 /
DATA AU2/4H 2 /
DATA ASIT/4HSIT / !Site
DATA AUNT/4HUNT / !Marks Unit label
DATA ABTM/4HBTM. / !Marks BTM data
DATA ATCT/4HTCT / !Marks TXT data
DATA AFMZ/4HFMZ / !Marks FMZ data

```

DATA ARME/4HFME /	IMarks FME data
DATA ASSV/4HSSV /	
DATA ARUN/4HRUN /	IMarks a RUN
DATA AERR/4HERR /	IMarks an ERROR
DATA A999/4HA999/	IMarks end of data
DATA AEND/4HEND /	IMarks end of data
DATA ACCU/4HCCU /	IMarks CCL
DATA ATIM/4HTIM /	IMarks the TIME
DATA ACOM/4HCOM /	IMarks comments
DATA SANC/4HST. /	

GO TO (100,1800),IPASS

C Pass 1 handles data with labels BTM, TCT, TMZ, TME, SSV

100 CONTINUE

KKOUT=0

Is always 0

L=ISET

ICODE =0

Initially, no errors

LOCATE=1

IUNIT=1

C
C
C

LCOK FOR TAPE ID TID

K=0

IClear FMZ array

IFMZC=0

land count of number of FMZ sets read

C READ CARDS UNTIL ONE BEGINS WITH 'TID' OR 'TIM' OR 'UNT' OR 'SIT' (BEGIN)

C OR ONE BEGINS WITH 'A999' OR 'END' (RETURN) OR WE REACH THE END OF FILE

200 CONTINUE

IUNIT=10

READ(M,200,ERR=3100,END=3700) AX,(N(J),J=1,76)

300 FORMAT(A4,76A1)

IF(AX.EQ.ATID .OR. AX.EQ.ATIM .OR. AX.EQ.AUNT .OR. AX.EQ.ASIT)

GO TO 400

IF(AX.EQ.A999) IPASS=3

IF(IPASS.EQ.3) RETURN

IF(AX.EQ.AEND) RETURN

GO TO 200

400 CONTINUE

IF(KKOUT.EQ.0) WRITE(6,600) AX,(N(J),J=1,76) IKKOUT is always 0

READ(M,300,ERR=3100,END=3900) AX,(N(J),J=1,76) IRead a card

500 CONTINUE

C If we see 'RUN' or 'CCU' or 'TCA=' or 'FVW=' then enter pass 2

IF(AX.EQ.ARUN

1 .OR. AX.EQ.ACCU

2 .OR. AX.EQ.ATCA

3 .OR. AX.EQ.AFVW)

4 GO TO 1800

IF(AX.EQ.SANC) LOCATE=2

IF(AX.EQ.AUA) IUNIT=1

IF(AX.EQ.AUB) IUNIT=2

IF(AX.EQ.AUC) IUNIT=3

IF(AX.EQ.AUD) IUNIT=4

IF(AX.EQ.AU1) IUNIT=7

IF(AX.EQ.AU2) IUNIT=8

```

C THIS MODIFICATION IS MADE TEMPORARILY TO PUSH THIS SET OF DATA
C   IUNIT=8
C   LOCATE=2
C   NBR=IUNIT

C If we just read a Data-Marker, then process data (at label 700)
  IF(AX.EQ.ABTM
1    .OR.AX.EQ.ATCT
2    .OR.AX.EQ.AFMZ
3    .OR.AX.EQ.AFME
4    .OR.AX.EQ.ASSV) GO TO 700

  IF(AX.EQ.ACOM) WRITE(6,600) AX,(N(J),J=1,76)      !A comment card
600  FORMAT(10X,A4,76A1)

  IF(AX.EQ.AEND) RETURN      !The end of something
  IF(AX.EQ.A999) RETURN      !Another end

C Otherwise, get another card
  GO TO 400

C Clear average and RMS
700  CONTINUE
  AV=0.0
  RMS=0.0
  ICOUNT=0

750  CONTINUE

C
C   READ IN DATA
C
  IF(LOCATE.EQ.1) READ(M,800,ERR=3300,END=4100)(ICHAN(J),IDATA(J),
CJ=1,8)
  IF(LOCATE.EQ.2) READ(M,900,ERR=3300,END=4100)(ICHAN(J),IDATA(J),
CJ=1,9)
900  FORMAT(9(A1,1X,15,1X))
800  FORMAT(8(A1,1X,16,1X))

C
C   CHECK FOR END OF DATA
C
  IF(ICHAN(1).EQ.L9) GO TO 1100      !'9' signals end of data
  DO 1000 J=1,8

C
C   DELETE BLANKS
C
  IF(ICHAN(J).EQ.L0) ICHAN(J)=0      !Convert character for channel to
  IF(ICHAN(J).EQ.L1) ICHAN(J)=1      !to channel number
  IF(ICHAN(J).EQ.L2) ICHAN(J)=2
  IF(IDATA(J).GT.0) GO TO 3300      !Data should always be negative?
  IF(ICHAN(J).EQ.LBLANK) GO TO 1000 !Skip if channel was a blank
  AV=AV+IDATA(J)
  RMS=RMS+IDATA(J)**2
  ICOUNT=ICOUNT+1
  IF(AX.EQ.AFMZ) K=K+1
  IF(AX.EQ.AFMZ) KSTORE(K)=IDATA(J)

1000 CONTINUE
  GO TO 750

C
C   END OF DATA ENCOUNTERED
C

```

```

1100 CONTINUE
C
C   STORE LATEST READ IN CODE
C
C   AX1=AX
C
C   CHECK FOR ERR
C
C   READ(M,300,ERR=3100,END=3900) AX,(N(J),J=1,76)
C
C READ FOLLOWING CARD. 'ERR' INDICATES PRECEDING SET OF DATA
C WAS ERRONEOUS. IN THAT CASE, CONTINUE SEARCHING
C
C   IF(AX.EQ.AERR) GO TO 500
C   AX2=AX
C   AX=AX1
C
C   PROCESS IF DATA OK
C
C   IF(AX.EQ.ATCT.OR.AX.EQ.AFMZ.OR.AX.EQ.AFME.OR.AX.EQ.ASSV.OR.AX.EQ.
CARUN.OR.AX.EQ.ABTM) GO TO 1200    Why is this here?
C   AX=AX2
C   GO TO 500
1200 CONTINUE
C
C   CODE BEING PROCESSED
C
C   GO FIND AV OF FMZ
C   COUNT=ICOUNT
C   AV=AV/ COUNT
C   RMS=(RMS/ COUNT)-AV**2
C   RMS=ABS(RMS)
C   RMS=SQRT(RMS)
C   AX=AX1
C
C   PRINT AV RMS
C
C   KEEPIX = 0
C   IF (AX .NE. ABTM) GOTO 1210
C   WRITE(6,2600) AV,RMS,ICOUNT
C   KEEPIX = 1
C   GOTO 1250
1210 IF (AX .NE. ATCT) GOTO 1220
C   WRITE(6,2700) AV,RMS,ICOUNT
C   KEEPIX = 2
C   GOTO 1250
1220 IF (AX .NE. AFMZ) GOTO 1230
C   WRITE(6,2800) AV,RMS,ICOUNT
C   IFMZC = IFMZC + 1
C   AVEFMZ=AV
C   IF (IFMZC .EQ. 1) KEEPIX = 3
C   IF (IFMZC .EQ. 2) KEEPIX = 6
C   GOTO 1250
1230 IF (AX .NE. AFME) GOTO 1240
C   WRITE(6,2900) AV,RMS,ICOUNT
C   KEEPIX = 4
C   GOTO 1250
1240 IF (AX .NE. ASSV) GOTO 1250
C   WRITE(6,3000) AV,RMS,ICOUNT
C   KEEPIX = 5

```



```

1250 CCNTINUE
    AKEEP(L, KEEPIX, 1) = AV
    AKEEP(L, KEEPIX, 2) = RMS
    AKEEP(L, KEEPIX, 3) = ICOUNT

    IF(IFMZC.EQ.2) GO TO 1400
    AX=AX2
    GO TO 1700
1400 CONTINUE
    SAVRMS=0.0
    SAVEAV=0.0
    DO 1500 I=1,K
    SAVRMS=SAVRMS+KSTORE(I)**2
    SAVEAV=SAVEAV+KSTORE(I)
1500 CONTINUE
    SAVEAV=SAVEAV/K
    SAVRMS=(SAVRMS/K)-SAVEAV**2
    SAVRMS=ARS(SAVRMS)
    SAVRMS=SQRT(SAVRMS)
    ISAVE=K
    AKEEP(L,7,1)=SAVEAV
    AKEEP(L,7,2)=SAVRMS
    KEEP(L,7,3)=ISAVE
    AVEFMZ=SAVEAV
    WRITE(6,1600) SAVEAV,SAVRMS,ISAVE
1600 FORMAT(20X,'**FMZAV=',F10.3,5X,'FMZRMS=',F10.3,5X,'NUM PTS=',I8)
C
C   REWIND AND SET ICODE =0    PASS=1 IS OK
C
    RETURN
1700 CONTINUE
C
C   CONTINUE    LAST FMZ NOT INCOUNTED YET
C   GO TO 500
C
C   ***** ENTER PASS 2 *****
C
1800 CONTINUE
    IF(AX.EQ.ARUN) GO TO 2000
    IF(ICODE.EQ.10) RETURN
    ICODE=0
C   READ CARDS UNTIL WE FIND 'RUN' OR 'FVW=' (GOTO #2000)
C   OR UNTIL WE FIND 'TCA=' OR 'CCU' (GOTO #2100)
C   OR UNTIL WE FIND 'END' (SET ICODE=10 AND RETURN)
C   OR UNTIL WE FIND 'FMZ' (GO TO #500)
C
1900 CONTINUE
    READ(M,300,ERR=3100,END=4300) AX,(N(J),J=1,76)
    IF(AX.EQ.ACOM) WRITE(6,690) AX,(N(J),J=1,76)
    IF(AX.EQ.AFMZ) GO TO 500
    IF(AX.EQ.ARUN) GO TO 2000
    IF(AX.EQ.AFVW) GO TO 2000
    IF(AX.EQ.A999) GO TO 4300
    IF(AX.EQ.ACCU) GO TO 2100
    IF(AX.EQ.ATCA) GO TO 2100
    IF(AX.EQ.AEND) ICODE=10
    IF(ICODE.EQ.10) RETURN

```

GO 1- 300

2000 CONTINUE

```
WRITE(6,600) AX,(N(J),J=1,76)
READ(M,300,ERR=3100,END=4300) AX,(N(J),J=1,76)
IF(AX.EQ.ACOM) WRITE(6,600) AX,(N(J),J=1,76)
IF(AX.EQ.A999) GO TO 4300
IF(AX.EQ.ACCU) GO TO 2100
IF(AX.EQ.ATCA) GO TO 2100
IF(AX.EQ.AEND) ICODE=10
IF(ICODE.EQ.10) RETURN
GO TO 2000
```

2100 CONTINUE

```
NPOINT=0
NPCARD =8
IF(LOCATE.EQ.2) NPCARD=9
```

C

C READ AND COUNT DATA POINTS FOR THIS SET (ASSUMES NPCARD DATA POINTS/CARD)

2200 CONTINUE

```
IBEGIN=NPOINT+1
IEND=IBEGIN +NPCARD-1
IF(LOCATE.EQ.1) READ(M,800,ERR=2400,END=4500) (
CICHAN(J),IDATA(J),J=IBEGIN,IEND)
IF(LOCATE.EQ.2) READ(M,900,ERR=2400,END=4500) (
CICHAN(J),IDATA(J),J=IBEGIN,IEND)
```

C IF BLANKS ARE FOUND IN CCU DATA SKIP THIS CCU

```
DO 2300 J=IBEGIN,IEND
IF (ICHAN(2) .EQ. LBLANK ) GO TO 2100
IF(IDATA(J).GT.0) GO TO 3500
IF(ICHAN(J).EQ.L0) ICHAN(J)=0
IF(ICHAN(J).EQ.L1) ICHAN(J)=1
IF(ICHAN(J).EQ.L2) ICHAN(J)=2
IF(ICHAN(J).EQ.L9) GO TO 2400
NPOINT=J
IF(ICHAN(J).EQ.LBLANK) GO TO 4700
```

2300 CONTINUE

GO TO 2200

2400 CONTINUE

```
READ(M,300,ERR=3100,END=3900) AX,(N(J),J=1,76)
NBR1=NPOINT
ICODE=0
IF(NPOINT.EQ.0) ICODE=10
IF(AX.EQ.AEND) ICODE=10
IF(AX.EQ.A999) ICODE=10
IF(AX.NE.AERR) RETURN
GO TO 1900
```

C THESE FORMAT STATEMENTS ARE NO LONGER USED

```
2500 FORMAT(20X,' CAN NOT FIND RUN CODE ****LAST CODE READ=',A4)
2600 FORMAT(20X,'AVBTM=',F10.3,5X,'RMSBTM=',F10.3,5X,'NUM OFPTS=',I8)
2700 FORMAT(20X,'AVTCT=',F10.3,5X,'RMSTCT=',F10.3,5X,'NUM OF PTS=',I8)
2800 FORMAT(20X,'AVFMZ=',F10.3,5X,'RMSFMZ=',F10.3,5X,'NUM OF PTS=',I8)
2900 FORMAT(20X,'AVFME=',F10.3,5X,'RMSFME=',F10.3,5X,'NUM OF PTS=',I8)
3000 FORMAT(20X,'AVSSV=',F10.3,5X,'RMSSSV=',F10.3,5X,'NUM OF PTS=',I8)
```

C PROGRAM WILL STOP UPON RETURN SINCE NO RUN WAS FOUND

C We should never execute the following code, since we don't read from MAGTAPE

```
3100 CONTINUE
ICODE=1
```

```

      WRITE(6,3200)
3200 FORMAT(20X,' *** TAPE ERROR **** STOP *****')
      RETURN
C Handle errors in data read in pass 1 by skipping that set of data
3300 CONTINUE
      WRITE(6,3400) AX
3400 FORMAT(20X,' *** ERROR IN ',A4,' DATA ***')
      GO TO 400

3500 CONTINUE
      WRITE(6,3600)
3600 FORMAT(20X,' ERROR IN DATA SKIPPING THIS CCU DATA SET')
      GO TO 1800

3700 CONTINUE
      WRITE(6,3800)
3800 FORMAT(20X,' NO TID OR END SKIP THIS SET *****')
      ICCODE=10
      RETURN

3900 CONTINUE
      WRITE(6,4000)
4000 FORMAT(20X,' HIT END FILE OF DATA SET GO TO NEXT ONE **')
      ICCODE=10
      RETURN

4100 CONTINUE
      WRITE(6,4200)
4200 FORMAT(20X,' *** HIT END OF DATA NO 9 SKIP AND CONTINUE')
      ICCODE=10
      RETURN

4300 CONTINUE
      ICCODE=10
      WRITE(6,4400)
4400 FORMAT(20X,' HIT END OF FILE NO RUN OR CCU FORMD CONTINUE ')
      RETURN

4500 CONTINUE
      ICCODE=10
      WRITE(6,4600)
4600 FORMAT(20X,' HIT END OF FILE WHILE READING CCU DATA SKIP')
      RETURN

4700 CONTINUE
      WRITE(6,4800)
4800 FORMAT(20X,' BLANKS IN DATA - SKIP THIS COOLING CURVE ')
      GO TO 1800
      END

```

*****CCN*****

```

C      CONVERSION TO TOPS-20 BY KCOA 7-20-78
C
C      SUBROUTINE CCN(TAU,TWATER,FLOVEL,HRNOM,HNOM)
C      COMMON /LOOK/THINT,THSLP,AVOLFM,VEXCIT,A,B,C,D;
C      1RAD1,RAD2,AKTUBE,TTHWLL,HINTR,TUBEID,FMGAIN,VNOM,HRSLPE
C      COMMON /CONST/ TIMIVL,NZERO,WINDOW,NPCARD,RAD3,HAIR,AKCYL,ALNGTH
C
C      TAU IS CONVERTED TO H BY THE RELATION
C       $LN(H) = A + B*LN(TAU) + C*(LN(TAU))^{**2} + D*LN(TAU))^{**3}$ 
C      THIS RELATION IS A FIT TO THE NUMBERS THAT WERE GENERATED BY HTAU (FETTE'S
C      TWO-CYLINDER PROGRAM)
C
C      IF(TAU.LT.0.0001) RETURN
C      FNTAU=ALOG(TAU)
C      FNH=A+B*FNTAU+C*FNTAU**2+D*FNTAU**3
C      HUNCOR=EXP(FNH)
C      HWATRR=1.0/HUNCOR-HINTR
C
C      CORRECTIONS FOR HEAT LOSS TO AIR AND TO WALLS OF HEAT
C      EXCHANGER TUBE. SEE PAGE 110 OF LAB BOOK II.
C
C      HCYL=(AKCYL/RAD1*12.0)/ALOG(RAD3/RAD2)
C      HCYLR=1.0/HCYL
C      HTUBE=(AKTUBE/RAD1*12.0)/ALOG(RAD2/RAD1)
C      HTUBER=1.0/HTUBE
C      RAIR=(HCYLR+HINTR+HTUBER+HWATRR)*HAIR*(RAD3/RAD1)
C      HCORA=HUNCOR*(1.0-RAIR)
C
C      HWTUWR=1.0/HCORA-HINTR
C      HWTUW=1.0/HWTUWR
C      WLLCON=(2.0/ALNGTH*12.0)*SQRT(AKTUBE*TTHWLL/12.0)
C      RWALLS=WLLCON/SQRT(HWTUW)
C      HCORAW=HWTUW*(1.0-RWALLS)
C      HCORR=1.0/HCORAW+HINTR
C      HCOR=1.0/HCORR
C
C      NORMALIZE HCOR TO 70 F BY THE FACTOR (1 + .012*TWATER)
C
C      ANORM=(1.0+.012*70.)/(1.0+.012*(1.8*TWATER+32.))
C      HCRAWN=ANORM*HCORAW
C      HCONNR=1.0/HCRAWN+HINTR
C      HCORN=1.0/HCONNR
C      WRITE(6,600)
600 FORMAT(/,20X,'TIME CONSTANT (TAU) CONVERTED TO HEAT TRANSFER COEF
1FICIENT (H) WITH CORRECTIONS FOR HEAT LOSS',/,20X,
2 'TO AIR AND TO HEAT EXCHANGER TUBE WALLS')
C      WRITE(6,605) HUNCOR,HINTR,RAIR,RWALLS,HCOR
C      WRITE(9,606) HUNCOR,RAIR,RWALLS,HCOR
605 FORMAT(25X,'H(UNCORRECTED) =',F8.2,' BTU/(HR FT**2 F)',/,
1 25X,'1.0/HINTERCEPT =',F9.6,' (BTU/HR FT**2 F)**-1',/,
2 25X,'RAIR =',F8.5,/,25X,'RWALLS =',F8.5,/,
3 25X,'H(CORRECTED) =',F8.2,' BTU/(HR FT**2 F)')
606 FORMAT(1PE13.6,4(/E13.6))
C      WRITE(6,610)
610 FORMAT(/,20X,'H(CORRECTED) NORMALIZED TO 70 F (NORMALIZATION STRIC
1TLY NOT CORRECT SINCE THE TEMPERATURE-INDEPENDENT',/,20X,
2 'FOULING CONTRIBUION IS NOT SUBTRACTED FROM H(CORRECTED)')
C      WRITE(6,615) ANORM,HCORN,HCONNR
C      WRITE(9,620) ANORM,HCORN,HCONNR

```

```

615 FCRMA, 25X, 'NORMALIZATION FACTOR =', F6.3, /, 25X, 'H(CORRECTED, NORMAL
11ZED) =', F8.2, ' BTU/(HR FT**2 F)', /, 25X, '1.C/H(CORRECTED, NORMALIZE
2D) =', F9.7)
620 FORMAT(1PE13.6, 2(/E13.6))

```

```

C NORMALISE FOR NOMINAL VELOCITY

```

```

V8R= FLOVEL ** (-0.8)
VNOM8R= VNOM **(-0.8)
VELCOR= HRSLPE * (V8R - VNOM8R )
HRNOM= HCONR - VELCOR
HNOM= 1. / HRNOM
WRITE (6,630) VNOM, HRSLPE
630 FORMAT(/, 20X, 'NOMINAL VELOCITY =', F8.3, /,
120X, 'SLOPE USED FOR VELOCITY NORMALISATION =', E12.5, //)
WRITE (6,640) VELCOR, HRNOM, HNOM
WRITE (9,645) VELCOR, HRNOM, HNOM
640 FORMAT(20X, 'CORRECTION IN 1/H DUE TO VELOCITY NORMALISATION =',
1E12.5, /, 20X, '1/H AFTER VELOCITY NORMALISATION =', E12.5, /,
220X, 'H AFTER VELOCITY NORMALISATION =', E12.5, //)
645 FORMAT (1PE13.6, 2(/E13.6))
RETURN
END

```

*****CHI*****

```

C     MODULE CHI
C
C     CONVERSION TO FORTRAN-20 BY KEVIN CLINE 27-JUL-78
C
C     CHI WAS MODIFIED TO SET A LOGICAL VARIABLE, OVFL, TO .TRUE.
C     INSTEAD OF GIVING AN ABNORMAL RETURN
C
C     1108 DECLARATION
C     REAL FUNCTION CHI(X,N,OVFL)
C     FORTRAN-20 DECLARATION
C     REAL FUNCTION CHI(X,N,OVFL)
C     LOGICAL OVFL
C-----
C     TO COMPUTE THE VALUE OF THE CHI-SQUARED DISTRIBUTION WITH N-D.F
C-----
C     IF(X.GT.0.0) GO TO 5
C     CHI=0.0
C     RETURN
C-----
C     CHECK TO SEE IF THE DEGREES OF FREEDOM IS EVEN
C-----
C     5 IF(MOD(N,2).EQ.0) GO TO 1
C-----
C     CALCULATION OF CHI FOR 1 DEGREE OF FREEDOM
C-----
C     G=SQRT(X)
C     CHI=2.0*RNORM(G):-1.0
C     G=G/1.25331414
C     IN=3
C     GO TO 2
C-----
C     CALCULATION OF CHI FOR 2 DEGREES OF FREEDOM
C-----
C     1 IN=4
C     G=X/2.0
C     CHI=1.0-EXP(-G)
C     2 IF(N.LT.3) RETURN
C     G=G*EXP(-X/2.0)
C-----
C     CALCULATION OF CHI FOR N-GT-2 DEGREES OF FREEDOM
C-----
C     DO 3 I=IN,N,2
C     CHI=CHI-G
C     G=G*X/I
C     CALL OVERFL(INDCT)
C     1108 SPECIAL RETURN
C     IF(INDCT.EQ.1)RETURN 3
C     FORTRAN-20 EQUIVALENT
C     IF (INDCT.EQ.1) GOTO 4
C     3 CONTINUE
C     RETURN
C     4 OVFL=.TRUE.
C     RETURN
C     END

```


*****CONLEV*****

```

C      MODULE CONLEV (FORMERLY INTERNAL TO PETITE)
C
C      CONVERSION TO FORTRAN-20 BY KEVIN CLINE
C      SEPARATION 25-JUL-78
C
C      EDIT BY KCOA 30-AUG-78 21:45
C      CHANGED PRINT STATEMENT TO WRITE(6,...)
C
C      EDIT BY KCOA 00:15 26-JUL-78
C      CONLEV AND CHI CHANGED TO ELIMINATE "RETURN N" STATEMENTS, THROUGH
C      USE OF LOGICAL VARIABLE OVFL. INSTEAD OF "RETURN 3" STATEMENT,
C      OVFL IS SET TO .TRUE. THEN A TEST IS DONE IN CONLEV, WHICH BRANCHES
C      TO 10 IF OVFL IS SET. IN FORTRAN-20 LABEL ARGUMENTS ARE LEGAL
C      IF PASSED TO SUBROUTINES, BUT MAY NOT BE PASSED TO FUNCTIONS.
C
C      FUNCTION CONLEV( NFREE , CHISQ )
C      LOGICAL OVFL
C      . 1108 CALL
C      CONLEV = 1. - CHI (CHISQ , NFREE , $10 )
C      . FORTRAN-20 CALL
C      CONLEV = 1. - CHI ( CHISQ , NFREE , OVFL )
C      IF (OVFL) GOTO 10
C      RETURN
10  CONTINUE
C      WRITE(6,20)
20  FORMAT(1H , ' OVERFLOW IN FUNCTION CHI . CONLEV SET EQUAL TO 0 ')
C      CONLEV = 0.
C      RETURN
C      END

```

*****FCHISQ*****

```

C      MODULE FCHISQ (FORMERLY INTERNAL TO PETITE)
C
C      EDIT BY KCOA 6-SEP-78
C      FIXED DECLARATION TO INCLUDE NPTS (ALSO CHANGED IN PETITE)
C
C      EDIT BY KCOA 29-AUG-78
C      DELETED TYPE STATEMENT, SINCE THINGS SEEMED TO BE WORKING;
C      DIVIDE CHECK WENT AWAY AFTER DIMENSIONING PAR.
C
C      EDIT BY KCOA 29-AUG-78 15:25
C      DIMENSIONED PAR, SINCE IT IS AN ARRAY.
C      ADDED TYPE STATEMENT TO CHECK PAR ON ENTRY.
C
C      CONVERSION TO FORTRAN-20 BY KEVIN CLINE
C      SEPARATION ON 25-JULY
C
C      1108 DECLARATION
C      SUBROUTINE FCHISQ ( PAR )
C
C      DEC-20 DECLARATION
C      SUBROUTINE FCHISQ (PAR,X,Y,WEIGHT,YFIT,CHISQR,NPTS)
C      COMMON/PETTY/FCALL,NFIX,FIX,B,NIT2,FLAMDA,CHISQ1,DEL
C      INTEGER NFIX
C      REAL FCALL, B(10), FLAMDA, CHISQ1, DEL
C      LOGICAL FIX(10)
C
C      REAL PAR(1), X(1), Y(1), WEIGHT(1), YFIT(1)
C
C      EVALUATE THE FITTING FUNCTION AND CHISQR FOR THE SET OF
C      PARAMETERS PAR
C
C      CHISQR = 0.
C      DO 10 I=1, NPTS
C      IF ( WEIGHT(I) .LE. 0. ) GO TO 10
C      YFIT(I) = FUNCTN ( X(I) , PAR )
C      FCALL = FCALL + 1.
C      CHISQR = CHISQR + WEIGHT(I) * ( Y(I) - YFIT(I) ) * * 2
10    CONTINUE
C      RETURN
C      END

```

*****FDERIV*****

C CONVERSION TO TOPS-20 BY KCOA 20-JUL-78

C

```
      SUBROUTINE FDERIV( TIME,APAR,DERIV )  
      DIMENSION APAR(3),DERIV(3)  
C  APAR(1) = VTC0  
C  APAR(2)= VTCT0 ( IE VTC AT T=0 )  
C  APAR(3)= TAU  
      VTC0=APAR(1)  
      VTCT0=APAR(2)  
      TAU=APAR(3)  
      XX=TIME / TAU  
      XXX= EXP( - XX )  
      DERIV(1)= 1. - XXX  
      DERIV(2)= XXX  
      DERIV(3)=( TIME/(TAU*TAU) ) * (VTCT0-VTC0) * XXX  
      RETURN  
      END
```

PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE

*****FITRES*****

SUBROUTINE FITRES(TMIN,TMAX,TAU,TAUERR,NPTFIT,WINTAU,AVETW,
1 STDW,AVEFV,STDFV)

WRITE(6,4700)

4700 FORMAT(/,20X,'FIT DATA (ADJUSTED FOR ZERO POINT) TO STRAIGHT LINE
10F THE FORM $Y=A+B \cdot X$, WHERE',/,25X,' $X = T$ ',/,25X,' $Y = \ln(VTC(T))$ ',

2/,25X,' $A = \ln(V(T=0))$ ',/,25X,' $B = -1/\tau$ ')

WINTAU=(TMAX-TMIN)/TAU

WRITE(6,4800) NPTFIT,TMIN,TMAX,WINTAU

WRITE(6,5100) TAU,TAUERR

4800 FORMAT(/,20X,'ONLY',IS,' DATA POINTS WHICH SATISFY THE CRITERIA T.

1GT.',F7.2,' SEC .AND. T.LT.',F8.2,' SEC',/,20X,'ARE INCLUDED IN TH

2E FIT',/,20X,'THIS WINDOW IS',F6.2,' TIME CONSTANTS WIDE')

5100 FORMAT(25X,' $\tau =$ ',F8.3,12X,' $\pm$ ',F5.2,'%')

WRITE(9,5110) NPTFIT,TMIN,TMAX,WINTAU,TAU,TAUERR

5110 FORMAT(113/,1PE13.6,4(/E13.6))

WRITE(6,5200) AVETW,STDW

5200 FORMAT(' AVERAGE WATER TEMPERATURE IS',F9.3' PLUS OR MINUS',F6.3)

WRITE(6,5300) AVEFV,STDFV

5300 FORMAT(' AVERAGE FLOW VELOCITY IS',F9.3' PLUS OR MINUS',F6.3)

WRITE(9,5905) AVETW,STDW,AVEFV,STDFV

5905 FORMAT(1PE13.6,3(/E13.6))

RETURN

END

*****FUNCTN*****

C CONVERSION TO TOPS-20 BY KCOA 2C-JUL-78

C

```
      FUNCTION FUNCTN ( TIME,APAR )  
      DIMENSION APAR(3)  
C  APAR(1) = VTCT0  
C  APAR(2)= VTCT0 ( 1E VTC AT T=0 )  
C  APAR(3)= TAU  
      VTCT0=APAR(1)  
      VTCT0=APAR(2)  
      TAU=APAR(3)  
      XX=TIME / TAU  
      XXX= EXP( - XX )  
      FUNCTN= VTCT0 + ( VTCT0 - VTCT0 ) * XXX  
      RETURN  
      END
```

PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE
PETITE

*****<US01>FIXSPC.HLP*****

FixSPC - A FORTRAN CALLABLE FILE SPEC GETTER

FixSPC gets file specs from the terminal in a format suitable for subsequent use in a FORTRAN OPEN call. FixSPC is one routine with two entry points: FIISPC for getting an input file specification and FIOSPC for getting an output file specification. An example of the use of the FixSPC routines is the following.

```
IMPLICIT INTEGER (a-z)
DOUBLE PRECISION fil
DIMENSION ppn (3)
CALL FIISPC('/Input file? >/',dev,fil,ppn)
OPEN(UNIT=1,DEVICE=dev,FILE=fil,DIRECTORY=ppn)
CALL FIOSPC('/Output file?>/',dev,fil,ppn)
OPEN(UNIT=2,DEVICE=dev,FILE=fil,DIRECTORY=ppn)
...
```

The first argument to FixSPC is a prompt string. Since FORTRAN does not pass its delimiters, the single quotes, to FixSPC, you must supply another set. FixSPC considers the first character it sees to be the delimiter, and the string extends until either the second such character is encountered or until 39 prompt characters have been encountered, whichever happens first. No attempt is made to write a null string, so CALL FIISPC(0,...) will not produce any prompt.

The second argument to FixSPC is a single word into which will be placed the device name obtained from the file spec. This will be five characters (nul filled); usually the returned value is something like TTY: or PS:.

The third argument is a doubleword into which will be placed the file name and extension. FixSPC will only accept file specs in which the name is six or fewer characters and the type is three or fewer (this limitation is because the FORTRAN OPEN, which is written for a DEC-10, would die otherwise). So what may typically be returned might be MYFILE.DAT or FRED.FOR. If what was typed, as is often the case, had no file name, e.g., the responder typed TTY:<return> or LPT:<return>, then the file name will be nul. Fortunately, FORTRAN OPEN will be gracious and not require that there be anything in the FILE field in such cases.

The DIRECTORY field must occupy at least three words. It is filled with the "project number", "programmer number", and zero respectively (again, a relic of DEC-10 software). This is necessary if you wish to access another user's directory. If there is no directory associated with the device (e.g., LPT:), then all words are returned as zero; again, OPEN is gracious if no directory is possible.

All the arguments to FixSPC are optional. No arguments (e.g., CALL FIOSPC) is wasteful but will generate no error. CALL FIOSPC(0) is pretty much the same. CALL FIISPC(!%there are easier ways%) will write the message and make no attempt to get a file specification. So much for silliness. A reasonable thing might be:

```
CALL FIOSPC ('!Output to LPT: or TTY: ? > ',dev)
```

The only difference between FIISPC and FIOSPC is that FIISPC requires that the file be able to be used for input, while FIOSPC requires that the file be able to be used for output. (Consider

the responses LPT: (the line printer) and CDR: (the card reader)).
Therefore, ACCESS='SEQIN' may be used in OPENing a FIISPCed file, and
ACCESS='SEQOUT' may be used in OPENing a FIOSPCed file.

There are errors that the FixSPC routines will not detect
since they are not caused until an attempt is made to open the file.
One of the more common errors at th's time is a protection violation.
Thus, you may wish to include the ERR= transfer in the OPEN statement
and return to the FixSPC routine in such an event.

FixSPC exhibits reasonable behavior in response to CTRL-U and
CTRL-R, but is pretty awful on CTRL-W's with an empty line. One last
note. FixSPC is persistent; once in the routine you can't get out
(except by CTRL-C, of course) until you've given a file spec that
makes it happy.

*****GXFILE*****

```

SUBROUTINE GIFILE(PROMPT, UNIT)
  INTEGER PPN(3), DEVICE, UNIT
  DOUBLE PRECISION FILE
  INTEGER PROMPT(1)      !A DUMMY DIMENSION
100  CALL FIISPC(PROMPT, DEVICE, FILE, PPN)
  OPEN(UNIT=UNIT, FILE=FILE, ACCESS='SEQIN', DEVICE=DEVICE,
    1 DIRECTORY=PPN, ERR=100)
  RETURN
  END

SUBROUTINE GOFIL(PROMPT, UNIT)
  INTEGER PPN(3), DEVICE, UNIT
  DOUBLE PRECISION FILE
  INTEGER PROMPT(1)      !A DUMMY DIMENSION
100  CALL FIOSPC(PROMPT, DEVICE, FILE, PPN)
  OPEN(UNIT=UNIT, FILE=FILE, ACCESS='SEQOUT', DEVICE=DEVICE,
    1 DIRECTORY=PPN, ERR=100)
  RETURN
  END

```

*****KXNUM*****

MAIN PROGRAM FOR SQLAR SEA FOULING

CONVERSION TO FORTRAN-20 BY KEVIN CLINE

GENERAL COMMENT:

I/O UNITS USED:

- 5 - ORIGINAL CONTROL CARD FILE. APPROXIMATELY 6 CARD IMAGES.
SHOULD BE USED ONLY BY THIS SUBPROGRAM.
- 6 - GENERAL OUTPUT FILE.
- 7 - WILSON PLOT DATA FILE.
- 9 - SEMI-REDUCED DATA FILE.

EXTENSIVE EDITING BY KCOA 22-MAY TO 30-MAY

Created subroutine LINEAR to perform the dirty work of calling LINFIT. This routine is a hack, but it works. It is used so that it could be called in a loop, varying TMIN and TMAX for each call, to allow the two and three parameter fits to be done over a number of windows. The initial fit is done once. Only the data from the window with the smallest relative difference between the two and three parameter TAU estimators is printed and saved.

Added code to write some of the numbers to a file of semi-reduced data on unit 9. This was at first scattered around the subroutines, but is now only in the main.

Created common blocks LOOK, FITDTA, CONST to eliminate long argument strings.

EDIT BY KCOA 31-AUG-78 18:25

ADDED TYPE STATEMENT SO THAT I COULD CHECK THERMISTOR DATA RETURNED BY BRNEW.

EDIT BY KCOA 29-AUG-78 11:45

DELETED CODE THAT PRINTS DETAILS OF FIT FOR AN ENTIRE SET OF DATA POINTS.

EDIT BY KCOA 27-AUG-78 16:40

FIXED TYPE STATEMENT FORMATS TO BEGIN WITH 1X.

EDIT BY KCOA 27-AUG-78 16:25

DELETED CALL TO ERRTRP. ADDED PROMPTS FOR FILE SPECS, SINCE FXSPEC WON'T ANY MORE. ALSO ASKED FOR OUTPUT FILE ON UNIT 6.

EDIT BY KCOA 27-AUG-78 15:55

ADDED "CALL ERRTRP" STATEMENT AT BEGINNING OF PROGRAM SO THAT I COULD FIND OUT WHICH ROUTINE WAS GETTING AN OVERFLOW ON A CALL TO EXP.

EDIT BY KCOA 27-AUG-78 15:45

DELETED "REWIND 17" STATEMENT INADVERTENTLY LEFT IN AFTER LAST EDIT.

EDIT BY KCOA 6-AUG-78

CHANGED FORTRAN A4 FORMATS TO A5, SINCE FIVE ASCII CHARACTERS FIT IN A PDP-10 WORD.

DELETED CODE TO SKIP TAPE FILES BY CALLING BR1108(0), AND INSTEAD CALLED GFILE(M) INSTEAD. GFILE OBTAINS A FILESPEC FROM THE TTY, AND OPENS THAT FILE ON THE UNIT

C .T TAKES AS ARGUMENT. NOTE THIS MEANS THAT NSKIP MUST NOT BE
C INPUT.

C ALSO MOVED PETITE DECLARATIONS TO PRECEDE OTHER DECLARATIONS
C SO THAT PARAMETER STATEMENT COULD OCCUR IN PROPER ORDER.

C EDIT BY KCOA 31-JUL-78 22:15
C MOVED TWO DATA STATEMENTS TO FOLLOW ALL DIMENSION AND
C EQUIVALENCE STATEMENTS.

C *****
C THIS GROUP OF STATEMENTS MOVED TO BEGINNING, SO PARAMETER STATEMENT
C COULD BE IN PROPER ORDER

C
C PARAMETER NPAR=3
C DIMENSION XPETIT(1000),YPETIT(1000),WTPTIT(1000)
C DIMENSION APAR(3),APARL1(3),APARL2(3),SIGPAR(3)
C EQUIVALENCE (TAU3,APAR(3)) ITAU from 3-parameter fit
C DIMENSION VTCPTT(1000),DIFPTT(1000),CSQPTT(1000)

C *****
C ORDER OF DATA CARDS FOR EACH SET OF DATA

C
C LABEL CHANGED TO A5 FORMAT 4-AUG
C LABEL IDENTIFICATION OF HEAT EXCHANGER TUBE (16A5)
C NPRELM NUMBER OF THERMOCOUPLE POINTS TAKEN BEFORE HEAT IS CUT (13)
C ICHAN(I),IDATA(I) CHANNEL NUMBER AND THERMOCOUPLE, THERMISTOR, OR FLOW
C METER DATA (I1,1X,I6,7(1X,I1,1X,I6)

C
C DIMENSION ICHAN(1500),IDATA(1500)

C
C COMMON /BLK2/ TITLE(20)

C *****
C VALUES FOR THE FOLLOWING VARIABLES ARE MADE AVAILABLE IN THE MAIN PROGRAM
C TIMIVL TIME INTERVAL BETWEEN SUCCESSIVE PIECES OF DATA. FOR EXAMPLE, IF
C THERMOPILE DATA ARE TAKEN AT T=0 SEC, THERMISTOR DATA AT T=2 SEC,
C AND FLOW METER DATA AT T=4 SEC, THEN TIMIVL=2.0 SEC.
C NZERO NUMBER OF THERMOPILE DATA POINTS AT THE END OF THE
C DECAY WHICH WILL BE USED IN CALCULATING THE ZERO POINT FOR THE
C THERMOPILE AS WELL AS IN ESTIMATING A CONSTANT IN THE WEIGHTING OF
C THE DECAY DATA FOR THE FIT OF TAU.
C TMININ TIME (MEASURED AFTER HEAT HAS BEEN CUT) WHICH MARKS THE BEGINNING
C OF THE ANALYSIS WINDOW FOR FITTING TAU.
C WINDOW NUMBER OF TIME CONSTANTS AFTER TMININ DURING WHICH THE THERMOPILE
C DECAY IS TO BE ANALYZED TO EXTRACT A TAU.
C NPCARD NUMBER OF DATA POINTS PER CARD.
C VFMZRC FLOW METER VOLTAGE READING WITH NO FLOW (MV).
C RAD3 1/2 OD OF HEATER CYLINDERS (IN).
C ISLT SELECTED UNIT FOR ANALYSIS

C
C COMMON /CONST/ TIMIVL,NZERO,WINDOW,NPCARD,RAD3,HAIR,AKCYL,ALNGTH

C *****
C
C *****
C VALUES FOR THE FOLLOWING UNIT DEPENDENT VARIABLES ARE MADE AVAILABLE
C IN THE SUBROUTINE LKTBLE

```

C THINT INTERCEPT IN CENTIGRADE DEGREES OF THE THERMISTOR CALIBRATION CURVE
C (TWATER(I)=THINT+THSLP*VTH(I))
C THSLP SLOPE OF THERMISTOR CALIBRATION CURVE ABOVE.
C VEXCIT EXCITATION VOLTAGE IN VOLTS OF RAMAPO FLOW METER STRAIN GAUGE BRIDGE.
C FMGAIN= GAIN OF FLOW METER AMPLIFIER
C AVOLFM CALIBRATION CONSTANT OF RAMAPO FLOW METER FOR USE IN CALIBRATION
C EQUATION FLOW(GAL/MIN)=AVOLFM*SQRT((VFM-VFMZRO)/VEXCIT)
C TUBEID ID OF HEAT EXCHANGER TUBE (IN).
C HINTR INTERCEPT ON ORDINATE AXIS IN 1/H VERSUS 1/V**0.8 PLOT.
C ACCN, BCCN, CCCN, DCCN COEFFICIENTS IN THE RELATION WHICH CONVERTS TAU TO
C H ACCORDING TO A TWO-CYLINDER MODEL WITH NO AIR, WALL, OR INTERFACE
C CORRECTIONS. WITH TAU IN SECONDS AND H IN BTU/HR FT**2 F, THE
C RELATION IS
C LN(H)=ACCN+BCCN*(LN(TAU))+CCCN*(LN(TAU))**2+DCCN*(LN(TAU))**3.
C RAD1 1/2 ID OF TUBE (IN).
C RAD2 1/2 OD OF TUBE (IN).
C AKTUBE THERMAL CONDUCTIVITY OF TUBE (BTU/HR FT F).
C AKCY THERMAL CONDUCTIVITY OF HEATER CYLINDERS (BTU/HR FT F).
C ALNGTH LENGTH OF HEATER CYLINDER SET (IN).
C TTHWLL WALL THICKNESS OF THIN-WALLED SECTION OF THE TUBE (IN).
C
C Common block /LOOK/ used in MAIN, LKTELE, LINEAR, CCN
C COMMON /LOOK/THINT,THSLP,AVOLFM,VEXCIT,ACCN,BCCN,CCCN,DCCN,
C 1RAD1,RAD2,AKTUBE,TTHWLL,HINTR,TUBEID,FMGAIN,VNOM,HRSLPE

C THIS COMMON BLOCK HOLDS DATA USED BY LINEAR IN COMPUTING A TWO-PARAMETER TAU.
C IT CONTAINS:
C FROM BRNEW
C TFM - TIME OF FLOW METER READINGS
C VFM - VOLTAGE FROM FLOW METER
C TTC - TIME OF THERMOCOUPLE READINGS
C VTC - VOLTAGE FROM THERMOCOUPLE
C TTH - TIME OF THERMISTOR READINGS
C VTH - VOLTAGE FROM THERMISTOR READINGS
C ITCEND - NUMBER OF VTC POINTS
C
C COMPUTED FROM THE BRNEW DATA
C VTCZRO - THERMOCOUPLE ZERO
C STDZRO - STANDARD DEVIATION OF THERMOCOUPLE DATA
C VFMZRO - FLOW METER ZERO
C
C READ FROM THE CARD FILE
C TEMFB - FALL-BACK TEMPERATURE
C VELFB - FALL-BACK VELOCITY

COMMON /FITOTA/TFM(500),VFM(500),TTC(500),VTC(500),TTH(500),
1 VTH(500),TEMFB,VELFB,VFMZRO,STDZRO,VTCZRO,ITCEND

C Common block for data returned by LINEAR or PETITE. This data is saved so
C that we can later decide which set of data to print & save. It is not
C really necessary to save more than one set at a time, but you never can
C tell what these Physicists might want tomorrow.

COMMON /FITSV/FSAVE(50) IROOM FOR 50 VARIABLES
DIMENSION IFSAVE(50)
EQUIVALENCE (IFSAVE,FSAVE)

```

```

C *****

```

INTEGER IPASS	IPass for BRNEW
INTEGER ISLT	INumber of experimental unit (location)
INTEGER IBEGIN	ITemporary for shuffling BRNEW data
INTEGER ISET	ICounts cooling curves
INTEGER ICODE	IErron code returned by BRNEW
INTEGER NUM	IObsolete counter for number of tapes
INTEGER NPCARD	INumber of data points per card (obsolete)
INTEGER NBR	IReturned unit number
INTEGER NBR1	INumber of data points returned by
	Iscond pass of BRNEW
INTEGER NPOINT	IOne less than NBF1, for some reason
INTEGER M	II/C unit number for main data
INTEGER ITC	IIndex for filling TTC and VTC
INTEGER ITCEND	INumber of TC values (final value of ITC)
INTEGER LEND	IIndex of last point to include in 2 & 3
	Iparameter fit
INTEGER IFIT	INumber of fit being done (1 = all data)
	(2 = small window)
INTEGER JBEGIN	IUsed for shuffling data from BRNEW
INTEGER IZEROB	IFirst point for calculation of asymptotic temp
INTEGER IZEROE	ILast point used for same
INTEGER MAXTIM	ITime of last TTC data point
INTEGER NPTFIT	INumber of points used in various fits
INTEGER IDIRCT	IUsed for separating IDATA by channel

```

C BR CHANGE 6/27/77 *****
C DIMENSION AKEEP(30,16,3)
C DIMENSION KEEP(30,16,3)
C EQUIVALENCE(KEEP,AKEEP)
C COMMON /BROUT/AKEEP
C DIMENSION YBR(500)

```

```

C DATA STATEMENTS MOVED TO HERE 31-JUL
C *****
C CHANGE BY GNG 30 JAN 79
C TAKE 10 POINTS FOR V(ING) NOT 15 (NZERO/10/)
C DATA TIMIVL/2.0/,NZERO/10/,WINDOW/1.0/,NPCARD/8/
C DATA RAD3/1.50 /,HAIR/1.57/,AKCYL/228./,ALNGTH/12.0/
C *****
C

```

```

C SAVE=0.0
C M=17 REFERS TO THE DRUM ON 1108
C M=17
C ISET=1

```

```

C ADDED CODE TO GET FILE SPECS FOR UNITS 5 & 6 27-AUG
C CALL GFILE('/FILE OF DATA CARDS: /',5)

```

```

C CALL GOFIL('OUTPUT FILE: /',6)
C END CHANGE KCOA 27-AUG
C CALL TO GOFIL (7) ADDED BY GNG TO
C OUTPUT WILSON PLOT DATA
C

```

```

C CALL GOFIL('/WILSON PLOT DATA FILE: /',7)
C END CHANGE GNG 22 FEB 79

```

```

      CALL GJFILE('/REDUCED DATA OUT: /', 9)
      READ (5,300) TITLE          !Save this
300  FORMAT (20A4)
C
C
C GNG CHANGE 29 JAN 79 TO READ DATA WITH POOR V AND T DATA
C READS FALL BACK VELOCITY AND TEMPERATURE AND USES THEM AS
C EXPLAINED NEAR CALL TO CCN
C
C Program reads START, WINDOW, number of windows to fit (NWINDO)
C It first regresses using the whole time series to get a rough
C estimator for TAU. Then it performs two fits for each of
C A series of windows, paying attention to the window with the
C best agreement between TAU2, obtained from a two-parameter
C regression, and TAU3, obtained from a three parameter regression.
C
C Each window has width TAU*WINDOW. The first window begins at the
C first reading taken after START seconds. Each succeeding window
C begins TIMIVL seconds later.
C
C *****END CHANGE*****
      READ (5,*)WINDOW,START,VELFB,TEMPB,NWINDO
      IF (WINDOW.LT.1.)WINDOW=1.

500  CONTINUE
      ICODE=0
      IPASS=1

C      CALL TO BR1108 REPLACED BY CALL TO GJFILE BY KCOA 4-AUG-78
CC      THIS CALL TO BR1108 READS IN ONE FILE AND CONVERTS IT TO BCD
C      CALL BR1108(1)

C      PROMPT ADDED BY KCOA 27-AUG-78
      TYPE 600
600  FORMAT(1X'DATA FILE: '$)
      CALL GJFILE('M)
      REWIND M

C
C FIRST PASS TO BRNEW R FINDS AVERAGES AND RMS VALUES FOR FMZ ETC
C CALL BRNEW R(ICODE,ICHAN,IDATA,IPASS,SAVE,NBR,M,ISLT,NBR1)
      REWIND M
      IF ( NBR .EQ. 7 ) SAVE=1163.4
      IF ( NBR .EQ. 8 ) SAVE=1418.7
      IF ( NBR .EQ. 10 ) GO TO 700
      GO TO 900
700  WRITE ( 6,800 )
800  FORMAT(1X,'UNIT INFORMATION NOT FOUND. - SKIP THIS FILE ')
      STOP

900  CONTINUE

      WRITE(9, 950)
950  FORMAT('BEGTAP01')          !TAPE, version 1

      ISLT=NBR

C      THIS CALL TO LKTBLE READS ALL THE LOCATION-DEPENDENT PARAMETERS

```

```

C      ...O COMMON BLOCK LOOK
      CALL LKTBLE (ISLT)

      WRITE(9,970)ISLT,THINT,THSLP,AVOLFM,VEXCIT,ACCN,BCCN,CCCN,
      1DCCN,RAD1,RAD2,AKTUBE,TTHWLL,HINTR,TUBEID,FMGAIN
970  FORMAT(I13/,1PE13.6,14(/E13.6))

      IF(IPASS.EQ.3) GO TO 7000
      IF(ICODE.NE.0) STOP

C START OF LOOP DONE ONCE FOR EACH COOLING CURVE.  ENDS AT #7000
1000 CONTINUE
      IPASS=2

C  THIS CALL TO BRNEWWR  READS IN ONE COOLING CURVE
      CALL BRNEWWR(ICODE,ICHAN,IDATA,IPASS,SAVE,NBR,M,ISET,NBR1)
      SAVE=ABS(SAVE)
      VFMZRO=SAVE
      NPJINT=NBR1-1
      IF(ICODE.EQ. 10) GOTO 7000
      IF(ICODE.NE.0) STOP

C  BR CHANGE 6/27/77  END *****

      WRITE(9, 1200) ISET
1200  FORMAT('BEGCRV01'/I13)  !Data for cooling curve, version 01

C
C BEGIN ANALYSIS WITH THE FIRST OCCURRENCE OF CHANNEL 0. (BLANKS CAN APPEAR
C ON LEFT OF FIRST DATA CARD).
C
      I=0
1300  I=I+1
      IF(ICHAN(I).NE.0) GO TO 1300
      IBEGIN=I

C
C FILL THE (REAL) VTC, VTH, AND VFM ARRAYS BEGINNING WITH THE ZERO TIME POINT
C
C  BR CHANGE 6/27/77 *****
      JBEGIN=0
      IND=NPOINT
      DO 1900 J=IBEGIN,IND,3
      BR1=IDATA(J)
      BR1=ABS(BR1)*.95
      BR2=IDATA(J+3)
      BR2=ABS(BR2)
      IF(BR1.GT.BR2) JBEGIN=J
      IF(JBEGIN.NE.0) GO TO 2000
1900 CONTINUE
2000 CONTINUE
C  BR CHANGE 6/27/77  END *****
C (SKIP INITIAL STEADY STATE POINTS).  THE FOLLOWING CODE ASSUMES THE DATA
C POINTS ARE NOW IN PROPER SEQUENCE (0,1,2)
C
      ITC=0
      ITH=0
      IFM=0
      DO 2400 J=JBEGIN,NPOINT
      IDIRCT=ICHAN(J)+1
      GO TO (2100,2200,2300),IDIRCT
2100 ITC=ITC+1

```

```

      TTC(ITC)=(J-JBEGIN)*TIMIVL
      VTC(ITC)=-IDATA(J)
      GC TO 2400
2200 ITH=ITH+1
      VTH(ITH)=-IDATA(J)
      TTH(ITH)=(J-JBEGIN)*TIMIVL
      GC TO 2400
2300 IFM=IFM+1
      VFM(IFM)=-IDATA(J)
      TFM(IFM)=(J-JBEGIN)*TIMIVL
2400 CONTINUE
      ITCEND=ITC
      ITHEND=ITH
      IFMEND=IFM

C  DETERMINE THERMOCOUPLE ZERO-POINT AND ITS RMS CEVIATION TO BE USED
C  IN WEIGHTING FACTOR
C
      IZEROB=ITCEND-NZERO+1
      IZEROE=ITCEND
      SUMZRO=0.0
      DO 3000 I=IZEROB, IZEROE
      SUMZRO=SUMZRO+VTC(I)
3000 CONTINUE
      VTCZRO=SUMZRO/NZERO
      SUMVAR=0.0
      DO 3050 I=IZEROB, IZEROE
      SUMVAR=SUMVAR+(VTC(I)-VTCZRO)**2
3050 CONTINUE
      VARZRO=SUMVAR/(NZERO-1)
      STDZRO=SQRT(VARZRO)

C
C  ADJUSTING RAW DATA FOR ZERO POINT
C
      DO 3100 I=1, ITCEND
      VTC(I)=VTC(I)-VTCZRO
3100 CONTINUE

      WRITE(6,3200) NZERO, VTCZRO
3200 FORMAT(//,20X, 'ZERO POINT FOR THIS MEASUREMENT DETERMINED FROM THE
1 LAST', I4, ' THERMOPILE DATA POINTS =', F9.2)
      WRITE(6,3300) STDZRO, VARZRO
3300 FORMAT(20X, 'RMS DEVIATION FROM ZERO POINT =', F8.2,/, 20X,
1 'VARIANCE (USED IN WEIGHTING OF DATA FOR FI) =', F9.2)
      WRITE(9,3350) NZERO, VTCZRO, STDZRO, VARZRO
3350 FORMAT(I13/, 1PE13.6, 2(/E13.6))

C  BR CHANGE    6/27/77    *****
      AKEEP(ISET,11,1)=VTCZRO
      AKEEP(ISET,11,2)=STDZRO
      KEEP(ISET,11,3)=NZERO
      BR CHANGE    6/27/77    END    *****

C
C  FITTING THE DATA OVER SPECIFIED TIME 'WINDOWS'
C
3400 TMIN=-1.0
      TMAX=-1.0
      TMININ = 10.0

```

C FIRST FIT OVER WHOLE DATA SET.

CALL LINEAR(TMAX,TMIN,TMININ,TAU1,TAUERR,AVEFV,AVETW,
1 IFIT,ISET,NPTFIT,STDFV,STDW,V0,YBRRMS,LBEGIN)

CALL FITRES(TMIN,TMAX,TAU1,TAUERR,NPTFIT,WINTAU,AVETW,
1 STDW,AVEFV,STDFV)

C

BESDIF = 1.0E10 |Effectively Infinity

DO 6350 IWIN = 1, NWINDOW

3500 WRITE(9, 3500)
 FORMAT('BEGWIN01')

 TMININ=START+(IWIN-1)*TIMIVL*3 |this is garbage, 3 TIMIVL's
 |between thermocouple readings
 | - KC

 TMININ=IFIX(TMININ/6.0)*6.0-0.1

 TMIN=TMININ
 TMAX=TMIN+WINDOW*TAU1
 IF(TMAX.GT.TMIN) GO TO 3600
 TMXSAV=TMAX
 TMAX=TMIN
 TMIN=TMXSAV

3600 CONTINUE

CALL LINEAR(TMAX,TMIN,TMININ,TAU2,TAUERR,AVEFV,AVETW,
1 IFIT,ISET,NPTFIT,STDFV,STDW,V0,YBRRMS,LBEGIN)

 WRITE (9,3700)TMIN,TMAX,TAU2,TAUERR
3700 FORMAT(1PE13.6,3(/E13.6))

C *****

C NOW ENTER INTO AN ANALYSIS USING PETITE

C AND FIT A FUNCTION $VTC(T)=VTC0 + (VTC(T=0) - VTC0) * \exp(-T/TAU)$

C $\exp(-T/TAU)$

C FILL IN THE ARRAYS XPETIT AND YPETIT

C XPETIT=TIME ARRAY

C YPETIT=' THERMOPILE VOLTAGE ' ARRAY

C WTPIT= WEIGHT FOR YPETIT DATA PTS. SET EQUAL TO 1.0

CALL PET(LBEGIN,NPTFIT,TAU2,V0,XPETIT,YPETIT,
1 WTPIT,APAR,APARL1,APARL2)

CALL PETITE(XPETIT,YPETIT,WTPIT,NPTFIT,3,APAR,SIGPAR,
1 APARL1,APARL2,VTCPTT,CHIFIT,0) |con't print

C 3-PARAMETER TAU = APAR(3)

RELDIF = ABS((APAR(3)-TAU2)/TAU2)

3800 WRITE(9,3800)APAR(3),SIGPAR(3),RELDIF,YBRRMS
 FORMAT(1PE13.6,3(/E13.6))

3900 WRITE(6,3900)TMIN, TAU2, TAU3, RELDIF,YBRRMS
 FORMAT('RESULTS FOR WINDOW BEGINNING AT T = 'F6.2/
1 TWO-PARAMETER TAU: 'F10.4')


```

2      THREE-PARAMETER TAU: 'F10.4/
3      RELATIVE DIFFERENCE: 'F10.4/
4      TWO-PARAMETER FITRMS: 'F10.4/)

      IF (RELDIF .GT. BESDIF) GOTO 6320

      BESDIF = RELDIF
      CALL SAVFIT(LBEGIN,TMAX,TMIN,TAU2,TAUERR,AVEFW,AVETW,NPTFIT,
1          STDFB,STDW,V0,YBRMS,APAR,SIGPAR,RELDIF,NIT1,NIT2)

      WRITE(9,4000)
4000  FORMAT('ENDWIN')

6320  CONTINUE

6350  CONTINUE

      WRITE(6,6400) ISET
6400  FORMAT(' BEST FIT FOR DATA SET',I3)

      CALL GETFIT(LBEGIN,TMAX,TMIN,TAU2,TAUERR,AVEFW,AVETW,NPTFIT,
1          STDFB,STDW,V0,YBRMS,APAR,SIGPAR,RELDIF,NIT1,NIT2)

      CALL STORE (ISET,AVETW,STDW,AVEFV,STDFV,TAU2,HCORNR,HCORN,YBRMS)

      AKEEP(ISET,8,1)=TAU2
      AKEEP(ISET,8,2)=YBRMS
      AKEEP(ISET, 16, 1) = TMIN
      AKEEP(ISET, 16, 2) = TMAX
      KEEP(ISET, 16, 3) = NPTFIT

      CALL FITRES(TMIN,TMAX,TAU2,TAUERR,NPTFIT,WIMTAU,AVETW,
1          STDW,AVEFV,STDFV)

      CALL CCN(TAU2,AVETW,AVEFV,HCORNR,HCORN)

      CALL PET(LBEGIN,NPTFIT,TAU2,V0,XPETIT,YPETIT,
1          WTPTIT,APAR,APARL1,APARL2)

      CALL PETITE(XPETIT,YPETIT,WTPTIT,NPTFIT,3,AFAR,SIGPAR,
1          APARL1,APARL2,VTCPIT,CHIFIT,1)  !print now

      WRITE (6,6500)
6500  FORMAT (1X,/,36X, 'PARAMETER',8X, 'ERROR')
      DO 6600 I=1,NPAR
      WRITE (6,6650) I,APAR(I),SIGPAR(I)
6650  FORMAT(21X,I10,E15.6,E15.6)
      WRITE (9,6655) APAR(I), SIGPAR(I)
6655  FORMAT(1PE13.6/E13.6)
6600  CONTINUE

C COMPUTE THE DIFFERENCE BETWEEN OBSERVED AND FITTED VTC FOR EACH DATA
C COMPUTE ALSO THE CHISQUARE FOR EACH DATA PT. AND THE SUM
      SUMCSQ=0.0
      DO 6700 J=1,NPTFIT
      DIFPTT(J)= ( VTCPTT(J) - YPETIT(J) )
      CSQPTT(J)= WTPTIT(J) * (DIFPTT(J) ) **2
      SUMCSQ=SUMCSQ + CSQPTT(J)
6700  CONTINUE

```

```

AX=APAR(3)
TAU=AX
RMSFIT=SQRT( SUMCSQ / NPTFIT )
WRITE(9,6655)YBRRMS,RMSFIT  IPRINT FITRMS FOR 2 AND 3 PAR FITS
CALL CCN(TAU,AVETW,AVEFV,HCORNR,HCORN)
CALL STORE3(ISET,AVETW,STDTW,AVEFV,STDFV,TAU,HCORNR,HCORN,RMSFIT)
DELVOO=VTCZRO - APAR(1)
AKEEP(ISET,13,1)=TAU
AKEEP(ISET,13,2)=SIGPAR(3)
AKEEP(ISET,14,3)=APAR(1)
AKEEP(ISET,15,3)=(AKEEP(ISET,8,1)-TAU)/TAU
AKEEP(ISET,14,1)=DELVOO
AKEEP(ISET,14,2)=RMSFIT
AKEEP(ISET,15,1)=HCORNR

6800  WRITE(9,6800)
      FORMAT('ENDCRV')

      ISET = ISET + 1          Iincrement ISET
      GOTO 1000

7000  CONTINUE

      WRITE(9,7100)-----
7100  FORMAT('ENDTAP')

      ISET=ISET - 1          IDecrement to get the count right
      CALL RESULT
      CALL RESULT3(ISET)

C OUT OF INFORMATION ABOUT 2 AND 3 PARAMETER COMPARISONS.
C
C CODE ADDED TO ALLOW DATA FOR WILSON PLOTS
C TO BE PRINTED OUT.
C
C
      WRITE(6,7900)TITLE
      WRITE(6,7800)START
7800  FORMAT(20X,' START='F7.3,' STARTING POINT = START SECONDS')
7900  FORMAT(1H1,20X,20A4)
      WRITE(6,9000)
8000  FORMAT(X,'RUN',5X,'TMIN',5X,'TAU 2',7X,'FITRMS 2',6X,'TAU 3',
1,7X,'FITRMS 3',5X,'ERROR 3',5X,'(T2-T3)/T3',4X,
2'V INF 2',6X,'V INF 3',7X,'V2-V3')
      DO8100I=1,ISET
8100  WRITE(6,8200)I,AKEEP(I,16,1),AKEEP(I,8,1),AKEEP(I,8,2),
1AKEEP(I,13,1),AKEEP(I,14,2),AKEEP(I,13,2),AKEEP(I,15,3),
2AKEEP(I,11,1),AKEEP(I,14,3),AKEEP(I,14,1)
8200  FORMAT(X,I3,4X,F5.1,F8.3,3F13.3,2X,6G13.3)
      WRITE(7,7900)TITLE
      WRITE(7,8400)
8300  FORMAT(I3,' =NUMBER OF DATA POINTS')
8400  FORMAT(1X,'WILSON PLOT DATA')
      WRITE(7,8300)ISET
      WRITE(7,8500)
8500  FORMAT(4X,'VEL',4X,'1/H(2PAR)',X,'1/H(3PAR)',7X,'RUN')

```

```

1,5X,'(T2-T3)/T3',4X,'FITRMS3')
DO 8600 I=1,ISET
VBR=AKEEP(I,10,1)**(-.8)
VNOMBR=6.**(-.8)
VELUNC=.00364*(VNOMBR-VBR)
R2=AKEEP(I,15,2)-VELUNC
R3=AKEEP(I,15,1)-VELUNC
8600 WRITE(7,8700)KEEP(I,10,1),R2,R3,1,AKEEP(I,15,3),AKEEP(I,14,2)
8700 FORMAT(F10.4,2E10.4,I10,5X,E10.4,E10.4)
CALL OUTBR(ISET)
CALL PLOTBR(ISET)
C CALL NTRAN(1,11)
STOP
END

```

*****LIMCHK*****

```

C      MODULE LIMCHK (FORMERLY INTERNAL TO PETITE)
C
C      CONVERSION TO FORTRAN-20 BY KEVIN CLINE
C      SEPARATION 25-JUL-78
C
C      1108 DECLARATION
C      SUBROUTINE LIMCHK
C      DEC-20 DECLARATION
C      SUBROUTINE LIMCHK(NTERMS, ALIM1, ALIM2)
C      COMMON/PETTY/FCALL,NFIX, FIX,B,NIT2,FLAMDA,CHISQ1,DEL
C      INTEGER NFIX
C      REAL FCALL, B(10), FLAMDA, CHISQ1, DEL
C      LOGICAL FIX(10)
C
C      INTEGER NTERMS
C      REAL ALIM1(1), ALIM2(1)
C
C      CHECK PARAMETERS AGAINST LIMITS . BRING AN OUT OF RANGE
C      PARAMETER BACK BY REFLECTING IT ABOUT IT S LIMIT POINTS
C
C      DO 10 I=1, NTERMS
C      IF ( FIX(I) ) GO TO 10
C      P = B(I)
C      PA = ALIM1(I)
C      PB = ALIM2(I)
C      IF ( P .GE. PA ) GO TO 4
C      P = 2. * PA - P
C      IF ( P .LE. PB ) GO TO 6
C      P = 2. * PB - P
C      GO TO 2
C      B(I) = P
C      CONTINUE
C      RETURN
C      END

```

*****LIMSET*****

```

C  MODULE LIMSET (FORMERLY INTERNAL TO PETITE)
C
C  CONVERSION TO TOPS-20 BY KEVIN CLINE
C  SEPARATION ON 25-JUL-78
C
C  EDIT BY KCOA 27-JUL-78
C  CORRECTED CALL TO INCLUDE NTERMS, ALIM1, AND ALIM2
C
C  1108 DECLARATION
C  SUBROUTINE LIMSET
C  DEC-20 DECLARATION
C  SUBROUTINE LIMSET(NTERMS,ALIM1,ALIM2)
C  DIMENSION ALIM1(1), ALIM2(2)
C  COMMON/PETTY/FCALL,NFIX,FIX,B,NIT2,FLAMDA,CHISQ1,DEL
C  INTEGER NFIX
C  REAL FCALL, B(10), FLAMDA, CHISQ1, DEL
C  LOGICAL FIX(10)
C
C  INTEGER NTERMS
C
C  REMOVE A PARAMETER FROM THE MINIMIZATION IF IT HAS EQUAL OR
C  INVERTED LIMITS
C
C  NFIX = 0
C  DO 10 I=1, NTERMS
C  FIX(I) = .FALSE.
C  IF ( ALIM1(I) .LT. ALIM2(I) ) GO TO 10
C  NFIX = NFIX + 1
C  FIX(I) = .TRUE.
10 CONTINUE
C  RETURN
C  END

```

*****LINEAR*****


```

SUBROUTINE LINEAR(TMAX,TMIN,TMININ,TAU,TAUERR,AVEFV,AVETW,
1 IFIT,ISET,NPTFIT,STDFV,STDW,V0,YBRMS,LBEGIN)

C ACCEPTS: TMAX, TMIN, TMININ, IFIT, ISET
C RETURNS: TAU, TAUERR, AVEFV, AVETW, NPTFIT, STDFV, STDW, V0, YBRMS,
C LBEGIN

      DIMENSION VTC SHF(500),TTHSHF(500),VTHSHF(500)
      DIMENSION TFM SHF(500),VFM SHF(500)
      DIMENSION X(500),Y(500),SIGMAY(500),CHISQR(500),WT(500),
D DIFF( 500),YFIT( 500)
      DIMENSION TWATER( 500),FLOVEL( 500)

      COMMON /BLK2/ TITLE(20)
C CONSTANTS DEFINED IN MAIN
      COMMON /CONST/ TIMIVL,NZERO,WINDOW,NPCARD,RAD3,HAIR,AKCYL,ALNGTH
C CONSTANTS FROM LKTBLE
      COMMON /LOOK/THINT,THSLP,AVOLFM,VEXCIT,ACCN,BCCN,CCCN,CCCN,
1 RAD1,RAD2,AKTUBE,TTHWLL,HINTR,TUBEID,FMGAIN,VNOM,HRSLE
C DATA FROM MAIN, CONSTANT OVER FITTINGS
      COMMON /FITDIA/TFM(500),VFM(500),TTC(500),VTC(500),TTH(500),
1 VTH(500),TEMFB,VELFB,VFMZRO,STDZRO,VTCZRO,ITCEND

      DIMENSION AKEEP(30,15,3)
      DIMENSION KEEP(30,15,3)
      EQUIVALENCE(KEEP,AKEEP)
      COMMON /BROUT/AKEEP

      DIMENSION YBR(500)
      MAXTIM=TTC(ITCEND)
D TYPE 100,TMIN,TMAX
D100 FORMAT(' TMIN='F6.2,' TMAX='F6.2)

      IF(TMIN.GE.MAXTIM) TMIN=-1.0
      IF(TMIN.GT.0.0) GO TO 3700
      TMIN=0.0
      LBEGIN=1
      GO TO 3900
3700 CONTINUE
      DO 3800 I=1,ITCEND
      IF(TMIN.GT.TTC(I)) GO TO 3800
      LBEGIN=I
      GO TO 3900
3800 CONTINUE
3900 CONTINUE
      IF(TMAX.GE.MAXTIM) TMAX=-1.0
      IF(TMAX.GT.0.0) GO TO 4000
      TMAX=MAXTIM
      LEND=ITCEND
      NPTFIT=LEND-LBEGIN+1
      GO TO 4200
4000 CONTINUE
      DO 4100 I=LBEGIN,ITCEND
      IF(TTC(I).LT.TMAX) GO TO 4100
      LEND=I-1
      NPTFIT=LEND-LBEGIN+1
      GO TO 4200
4100 CONTINUE

```

```

4200 CONTINUE
C
C FILLING ARRAY WITH VARIABLES FROM TIME 'WINDOWS'
C
DO 4300 I=1,NPTFIT
L=LBEGIN+I-1
X(I)=TTC(L)
VTC SHF(I)=VTC(L)
TTH SHF(I)=TTH(L)
VTH SHF(I)=VTH(L)
TFM SHF(I)=TFM(L)
VFM SHF(I)=VFM(L)
ARG1=ABS(VTC(L))
IF(ARG1.LT.0.001) ARG1=0.001
Y(I)=ALOG(ARG1)
SIGMAY(I)=STDZRO/VTC(L)
4300 CONTINUE
CALL LINFIT(X,Y,SIGMAY,NPTFIT,1,A,SIGMAA,B,SIGMAB,R,WT,CHISQR,
1 CHITOT,STD)
DO 4400 I=1,NPTFIT
YFIT(I)=A+B*X(I)
ARG2=ABS(VTC SHF(I))
IF(ARG2.LT.0.001) ARG2=0.001
DIFF(I)=YFIT(I)-ALOG(ARG2)
4400 CONTINUE
C
C ADDITION BY B.R. 6/24/77 *****
C
YBRDIF=0.0
DO 4500 I=1,NPTFIT
YBR(I)=EXP(A+X(I)*B)
YBRDIF=YBRDIF+(YBR(I)-VTC SHF(I))**2
4500 CONTINUE
PTFIT=NPTFIT
YBR RMS=SQRT(YBRDIF/PTFIT)
AKEEP(ISET,8,1)=-1.0/B
AKEEP(ISET,8,2)=YBR RMS
KEEP(ISET,8,3)=PTFIT
IF(IFIT.EQ.1) AKEEP(ISET,12,1)=-1.0/B
IF(IFIT.EQ.1) AKEEP(ISET,12,2)=YBR RMS
IF(IFIT.EQ.1) KEEP(ISET,12,3)=NPTFIT
C
C END BR CHANGE
C
C FITTED VARIABLES AND ERRORS TRANSFORMED TO VARIABLES OF THE PROBLEM AND
C THEIR ERRORS
C
VO=EXP(A)
VOHIGH=EXP(A+SIGMAA)
VOLOW=EXP(A-SIGMAA)
TAU=-1.0/B
TAUERR=-100*SIGMAB/B lin percent
C
C CONVERTING THERMISTOR SIGNAL TO WATER TEMPERATURE (C) AND
C FLOW METER SIGNAL TO FLOW VELOCITY (FT/SEC).
C
SUMTW=0.0
SUMFV=0.0

```

```

FMRI = 'FMRU'      IFLOW meter zero indicator = Flow meter reading Used
ALINFM=AVOLFMSQRT(1.0/VEXCIT)*0.4085/TUBEID**2
NPTAVE=NPTFIT-1
DO 5500 I=1,NPTAVE
  TWATER(I)=THINT+THSLP*(0.001*VTHSHF(I))
  SUMTW=SUMTW+TWATER(I)
  XYX=VFMSHF(I) - VFMZRO
  IF ( XYX .GE. 0. )GOTO 5400
  FMRI = 'FBVU'      IFall back velocity used
  XYX=(VELFB/ALINFM)**2*FMGAIN

```

```

5400 CONTINUE
  FLOVEL(I)=ALINFM * SQRT( XYX / FMGAIN )
  SUMFV=SUMFV+FLOVEL(I)
5500 CONTINUE

```

```

  IF (FMRI.EQ.'FBVU'). WRITE(6,5300)
5300 FCRMAT (10X,'FLOW METER READING LESS THAN FMZ USE FALL BACK
1VELOCITY')

```

```

  AVETW=SUMTW/NPTAVE
  AVEFV=SUMFV/NPTAVE

```

```

C
C STANDARD DEVIATION OF WATER TEMPERATURE AND FLOW VELOCITY OVER ANALYSIS
C WINDOW
C

```

```

  SUMSQT=0.0
  SUMSQF=0.0
  DO 5600 I=1,NPTAVE
    SUMSQT=SUMSQT+(TWATER(I)-AVETW)**2
    SUMSQF=SUMSQF+(FLOVEL(I)-AVEFV)**2
5600 CONTINUE

```

```

  STDTW=SQRT(SUMSQT/(NPTAVE-1))
  STDFV=SQRT(SUMSQF/(NPTAVE-1))
C BR CHANGE 6/27/77 *****
  AKEEP(ISET,9,1)=AVETW
  AKEEP(ISET,9,2)=STDTW
  AKEEP(ISET,10,1)=AVEFV
  AKEEP(ISET,10,2)=STDFV
  KEEP(ISET,9,3)=NPTAVE
  KEEP(ISET,10,3)=NPTAVE
C BR CHANGE 6/27/77 END *****

```

```

C
C CCN CONVERTS TAU TO H, CORRECTS H FOR HEAT LOSSES TO THE AIR AND TO THE TUBE
C WALLS, AND NORMALIZES THE RESULT TO 70 F.
C

```

```

C GNG CHANGE 29 JAN 79
C

```

```

C THIS CHANGE READS IN A FALL BACK VELOCITY AND A FALL BACK
C TEMPERATURE. THE PROGRAM CALCULATES VELOCITY AND TEMPERATURE
C IN THE USUAL WAY AND COMPARES THE AVERAGE VELOCITY AND TEMPERATURE
C WITH THE FALL BACK VALUES. IF THE TWO VELOCITIES ARE
C DIFFERENT BY MORE THAN .5 FT/SEC THE FALL BACK VALUE IS USED.
C IF THE TWO TEMPERATURES ARE DIFFERENT BY MORE THAN .5K AGAIN THE
C FALL BACK VALUE IS USED.
C
C
C

```

```

  IF(VELFB.LT.1.)GO TO 5920

```

```

      FVTEST = ABS(AVEFV-VELFB)
      FMRI = 'FMRU'
      IF(FVTEST.LT..5)GO TO 5920
      AKEEP(ISET,10,1)=AVEFV
      WRITE(6,5910)AVEFV,VELFB
5910   FORMAT(10X,'AVE FLOW VELOCITY FROM THE DATA=' ,F6.3,
      1' FALL BACK VELOCITY =',F6.3,' FALL BACK VELOCITY USED IN'
      2ANALYSIS')
      AVEFV=VELFB
      FMRI = 'FBVU'
5920   CONTINUE
      WRITE(9,5925)FMRI
5925   FORMAT(A5)

      KEEP(ISET,13,3)=AVEFV
      WRITE(6,5930)VFMZRO
5930   FORMAT(10X,' FLOWMETER ZERO USED IN ANALYSIS',F8.2)
      WRITE(9,5935)VFMZRO
5935   FORMAT(1PE13.6)

      TMRI = 'TMRU'   !TeMperature Reading Indicator = Temp reading used
      IF(TEMFB.LT.20.)GO TO 5950
      TWTEST=ABS(AVETW-TEMFB)
      IF(TWTEST.LT..5)GO TO 5950
      WRITE (6,5940)AVETW,TEMFB
      AVETW=TEMFB
5940   FORMAT(10X,'AVERAGE WATER TEMPERATURE = ' ,F10.4,
      1' FALL BACK TEMPERATURE = ' ,F10.4,' FALL BACK TEMPERATURE
      2USED IN ANALYSIS')
      TMRI = 'FBTU'   !Fall back temperature used
5950   CONTINUE
      WRITE(9, 5960)TMRI
5960   FORMAT(A5)

```

C
C
C
C
C

****END CHANGE****

RETURN
END

C CONVERSION TO TOPS-20 BY KCOA 20-JUL-78
C
C LINFIT WAS LIFTED FROM 'DATA REDUCTION AND ERROR ANALYSIS FOR THE PHYSICAL
C SCIENCES' BY BEVINGTON, PAGE 104
C

 SUBROUTINE LINFIT(X,Y,SIGMAY,NPTS,MODE,A,SIGMAA,B,SIGMAB,R,WT,
 1 CHISQR,CHITOT,STD).
 DOUBLE PRECISION SUM,SUMX,SUMY,SUMX2,SUMY2
 DOUBLE PRECISION XI,YI,WEIGHT,DELTA,VARNCE
 DIMENSION X(1),Y(1),SIGMAY(1),CHISQR(1),WT(*)

C
C ACCUMULATE WEIGHTED SUMS
C

11 SUM=0.
 SUMX=0.
 SUMY=0.
 SUMX2=0.
 SUMXY=0.
 SUMY2=0.
21 DO 50 I=1,NPTS
 XI=X(I)
 YI=Y(I)
 IF(MODE) 31,36,38
31 IF(YI) 34,36,32
32 WEIGHT=1./YI
 GO TO 41
34 WEIGHT=1./(-YI)
 GO TO 41
36 WEIGHT=1.
 WT(I)=WEIGHT
 GO TO 41
38 WEIGHT=1./SIGMAY(I)**2
 WT(I)=WEIGHT
41 SUM=SUM+WEIGHT
 SUMX=SUMX+WEIGHT*XI
 SUMY=SUMY+WEIGHT*YI
 SUMX2=SUMX2+WEIGHT*XI*XI
 SUMXY=SUMXY+WEIGHT*XI*YI
 SUMY2=SUMY2+WEIGHT*YI*YI
50 CONTINUE

C
C CALCULATE COEFFICIENTS AND STANDARD DEVIATIONS
C

51 DELTA=SUM*SUMX2 - SUMX*SUMX
 A=(SUMX2*SUMY - SUMX*SUMXY)/DELTA
53 B=(SUMXY*SUM - SUMX*SUMY)/DELTA
 CHITOT=0.0
 DO 55 I=1,NPTS
 CHISQR(I)=WT(I)*(A+B*X(I)-Y(I))**2
 CHITOT=CHITOT+CHISQR(I)
55 CONTINUE
61 IF(MODE) 62,64,62
62 VARNCE=1.
 GO TO 67
64 C=NPTS-2
 VARNCE=(SUMY2 + A*A*SUM + B*B*SUMX2
 1 -2.*(A*SUMY + B*SUMXY - A*B*SUMX))/C
67 SIGMAA=DSQRT(VARNCE*SUMX2/DELTA)
68 SIGMAB=DSQRT(VARNCE*SUM/DELTA)

```
71 R=(SUM*SUMXY - SUMX*SUMY)/  
1 DSQRT(DELTA*(SUM*SUMY2 - SUMY*SUMY))  
STD=SQRT((NPTS/(NPTS-2))*(CHITOT/SUM))  
RETURN  
END
```

SUBROUTINE LKTBLE(ISLT)

COMMON /LOOK/THINT,THSLP,AVOLFM,VEXCIT,ACCN,BCCN,CCCN,DCCN,
1RAD1,RAD2,AKTUBE,TTHWLL,HINTR,TUBEID,FMGAIN,VNOM,HRSLPE

```

30 TO (10,20,30,40,50,60,70,80) ,ISLT
10  CONTINUE
C UNIT A KEAHOLE NOV 78 RAMAPO 4028 TM 14 AL 5052 TUBE 13 PORT 1
  WRITE(6,1001)
1001 FORMAT(10X,' UNIT SELECTED IS UNIT A KEAHOLE POINT NOV 78
1SEA WATER FM CALIB.')
  VEXCIT=5.101
  FMGAIN=1961.4
  C      +-18
  AVOLFM=44.10
  C      +-0.1
  ACCN=13.623865
  ECCN=-2.961494
  CCCN=0.393788
  ECCN=-0.025913
  C      RMS=2.1
  AKTUBE=79.83
  TTHWLL=0.058
  TUBEID=1.049
  C      +-0.001
  RAD1=0.5247
  RAD2=0.6496
  HINTR=.00009
  C      +-0.00001
  VNOM=6.0
  HRSLPE=.00364
  THINT=20.91
  C      +-1
  THSLP=1.230
  C      +-0.1
  C UNIT A THERMISTOR NOT CALIBRATED IN LAB. CALIBRATION DETERMINED BY
  C ASSUMING A SLOPE AND CALCULATING AN INTERCEPT WITH T DETERMINED
  C USING UNIT B THERMISTOR.
  GO TO 66
20  CONTINUE
C UNIT B KEAHOLE NOV 78 RAMAPO 4411 TM 12 TI GRADE 2 TUBE 14 PORT 2
  WRITE(6,1002)
1002 FORMAT(10X,' UNIT SELECTED IS UNIT B KEAHOLE POINT NOV 78
1SEA WATER FM CALIB USED')
  VEXCIT=5.101
  FMGAIN=1961.4
  C      +-18
  AVOLFM=51.10
  C      +-0.2
  ACCN=44.842129
  BCCN=-22.953427
  CCCN=4.74917
  DCCN=-.347970
  C      RMS=0.4
  AKTUBE =12.7
  TTHWLL=.1224
  TUBEID=1.056
  C      +-0.001

```

```

RAD1=.5276
RAD2=.6500
HINTR=-.000007
C   +- .00001
VNOM=6.0
HRSLE=.00364
THINT=18.58
C   +-0.04
T-SLP=1.215
C   +-0.007
GO TO 66
30  CONTINUE
C UNIT C KEAHOLE NOV 78 RAMAPO 4030 TM 2 90CU/10NI TUBE 15 PORT 3
WRITE(6,1003)
1003 FORMAT(10X,' UNIT SELECTED IS UNIT C KEAHOLE POINT NOV 78
1SEA WATER FM CALIB USED')
VEXCIT=5.101
FMGAIN=1961.4
C   +-19
AVOLFM = 44.51
C   +-0.01
ACCN=21.231570
BCCN=-7.462704
CCCN=1.299938
DCCN=-.088456
C   RMS=5.3
AKTUBE =22.98
TTHWLL=.119
TLBEID=1.060
C   +- .001
RAD1=.530
RAD2=.649
HINTR=-.000033
C   +- .000004
VNOM=6.0
HRSLE=.00364
THINT=19.21
C   +- .02
THSLP=1.238
C   +- .003
GO TO 66
40  CONTINUE
C UNIT D KEAHOLE POINT NOV 78 SS AL8X #16 RAMAPO 4969 TM 11 PORT 5
WRITE(6,1004)
1004 FORMAT(10X,' UNIT SELECTED IS UNIT D KEAHOLE POINT NOV 78
1SEA WATER FM CALIB USED')
VEXCIT=5.101
C   +-18
FMGAIN=1961.4
AVOLFM=48.57
C   +- .04
ACCN=69.869702
BCCN=-35.738367
CCCN=6.827357
DCCN=-.450269
C   RMS=14.3
AKTUBE=7.98
TTHWLL=.129
TUBEID=1.041
C   +- .004

```



```

RAD1=.521
RAD2=.6500
VNOM=6.0
HINTR=.00016
C   +- .00002
HRSLPE=0.00364
THINT=18.77
C   +- .02
THSLP=1.255
C   +- .005
GO TO 66
50  CONTINUE
C   UNIT A KEAHOLE POINT DEC 77 WITH UNIT C TM AND FM
    WRITE (6,2001)
    2001 FORMAT(10X,' TC FROM UNIT A TM AND FM FROM UNIT C')
    VEXCIT=5.101
    FMGAIN=1951.4
    AVOLFM = 44.677
    ACCN=44.842129
    BCCN=-22.953427
    CCCN=4.749417
    DCCN=-.347970
    AKTUBE=12.7
    TTHWLL=.1224
    TUBEID=1.055
    RAD1=.5276
    RAD2=.6500
    HINTR=.00004
    VNOM=6.0
    HRSLPE=.003632
    THINT=18.919
    THSLP=1.277
    GO TO 66
60  CONTINUE
C   UNIT B KEAHOLE POINT DEC 77 WITH UNIT C TM AND FM
    WRITE (6,2002)
    2002 FORMAT(10X,' UNIT B KEAHOLE POINT WITH UNIT C TM AND FM')
    VEXCIT=5.101
    FMGAIN=1961.4
    AVOLFM=44.567
    ACCN=13.996615
    BCCN=-3.297064
    CCCN=.432248
    DCCN=-.0036349
    AKTUBE =79.83
    TTHWLL=.058
    TUBEID=1.049
    RAD1=.5247
    RAD2=.6496
    HINTR=.00042
    VNOM=6.0
    HRSLPE=.003632
    THINT=18.919
    THSLP=1.277
    GO TO 66
70  CONTINUE
C   TC FROM UNIT D TM AND FM FROM UNIT C
    WRITE (6,2004)
    2004 FORMAT(10X,' UNIT D WITH UNIT C TM AND FM KEAHOLE DEC 77')
    VEXCIT=5.101

```

FMGAIN=1961.4
AVOLFM = 44.677
ACCN=44.842129
BCCN=-22.953427
CCCN=4.749417
DCCN=-.347970
AKTUBE=12.7
TTHWLL=.1224
TUBEID=1.055
RAD1=.5276
RAD2=.6500
VNOM=6.0
HINTR=.00004
HRSLPE=0.003632
THINT=18.919
THSLP=1.277
GO TO 66

80 CONTINUE

C THE SELECTED UNIT IS AL 6061-T6 OPERATING AT ST.CROIX
C NOMINAL FLOW VELOCITY IS 3.0 FT/SEC

THINT= 21.363
THSLP= 1.1926
VEXCIT=5.025

C FMGAIN=2019.5
AVOLFM=35.229 FOR SEA WATER (PAGE 59, BOOK II)

AVOLFM=35.229
ACCN= 13.328
BCCN= -2.7895
CCCN= 0.35925
DCCN= -0.024562
AKTUBE=89.5

T-HWLL= 0.059
TUBEID=1.052

RAD1= 0.526
RAD2= 0.650

HINTR= 0.0
VNOM=3.0

C HRSLPE=0.003625 FOR SEA WATER (PAGE 57, BOOK II)
HRSLPE=0.003625

GO TO 66

66 CONTINUE
RETURN
END

*****MATINV*****

```

C  PURPOSE
C  INVERT A SYMMETRIC MATRIX AND CALCULATE ITS DETERMINANT
C
C  USAGE
C  CALL MATINV (ARRAY, NORDER, DET)
C
C  DESCRIPTION OF PARAMETERS
C  ARRAY - INPUT MATRIX WHICH IS REPLACED BY ITS INVERSE
C  NORDER - DEGREE OF MATRIX (ORDER OF DETERMINANT)
C  DET - DETERMINANT OF INPUT MATRIX
C
C  SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C  NONE
C
C  COMMENTS
C  DIMENSION STATEMENT VALID FOR NORDER UP TO 10
C
C  SUBROUTINE MATINV (ARRAY, NORDER, DET)
C  DOUBLE PRECISION ARRAY, AMAX, SAVE
C  DIMENSION ARRAY(10,10), IK(10), JK(10)
C
C  10 DET = 1.
C  11 DO 100 K=1, NORDER
C
C      FIND LARGEST ELEMENT ARRAY(I,J) IN REST OF MATRIX
C
C      AMAX = 0.
C  21 DO 30 I=K, NORDER
C      DO 30 J=K, NORDER
C  23 IF ( DABS ( AMAX ) - DABS ( ARRAY(I,J) ) ) 24, 24, 30
C  24 AMAX = ARRAY(I,J)
C      IK(K) = I
C      JK(K) = J
C  30 CONTINUE
C
C      INTERCHANGE ROWS AND COLUMNS TO PUT AMAX IN ARRAY(K,K)
C
C  31 IF (AMAX) 41, 32, 41
C  32 DET = 0.
C      GO TO 140
C  41 I = IK(K)
C      IF (I-K) 21, 51, 43
C  43 DO 50 J=1, NORDER
C      SAVE = ARRAY(K,J)
C      ARRAY(K,J) = ARRAY(I,J)
C  50 ARRAY(I,J) = -SAVE
C  51 J = JK(K)
C      IF (J-K) 21, 61, 53
C  53 DO 60 I=1, NORDER
C      SAVE = ARRAY(I,K)
C      ARRAY(I,K) = ARRAY(I,J)
C  60 ARRAY(I,J) = -SAVE
C
C      ACCUMULATE ELEMENTS OF INVERSE MATRIX
C
C  61 DO 70 I=1, NORDER
C      IF (I-K) 63, 70, 63
C  63 ARRAY(I,K) = -ARRAY(I,K) / AMAX
C  70 CONTINUE

```

```

71 DO 80 I=1, NORDER
   DO 80 J=1, NORDER
     IF (I-K) 74, 80, 74
74 IF (J-K) 75, 80, 75
75 ARRAY(I,J) = ARRAY(I,J) + ARRAY(I,K)*ARRAY(K,J)
80 CONTINUE
81 DO 90 J=1, NORDER
   IF (J-K) 83, 90, 83
83 ARRAY(K,J) = ARRAY(K,J) / AMAX
90 CONTINUE
   ARRAY(K,K) = 1. / AMAX
100 DET = DET * AMAX

```

C
C
C

RESTORE ORDERING OF MATRIX

```

101 DO 130 L=1, NORDER
   K = NORDER - L + 1
   J = IK(K)
   IF (J-K) 111, 111, 105
105 DO 110 I=1, NORDER
   SAVE = ARRAY(I,K)
   ARRAY(I,K) = -ARRAY(I,J)
110 ARRAY(I,J) = SAVE
111 I = JK(K)
   IF (I-K) 130, 130, 113
113 DO 120 J=1, NORDER
   SAVE = ARRAY(K,J)
   ARRAY(K,J) = -ARRAY(I,J)
120 ARRAY(I,J) = SAVE
130 CONTINUE
140 RETURN
   END

```

*****OUTBR*****

C CONVERSION TO TOPS-20 BY KCOA 20-JUL-78

C

```
SUBROUTINE OUT8R(ISET)
DIMENSION AKEEP(30,15,3)
DIMENSION KEEP(30,15,3)
COMMON /BROUT/AKEEP
EQUIVALENCE(KEEP,AKEEP)
COMMON /BLK2/TITLE(20)
WRITE(6,501)
501 FORMAT(1H1,/)
WRITE(6,700) (TITLE(I),I=1,20)
700 FORMAT(30X,20A4)
DO 10 I=1,ISET
WRITE(6,600) I
600 FORMAT(/,50X,'ISET=',I5)
WRITE(6,502) AKEEP(I,1,1),AKEEP(I,1,2),KEEP(I,1,3)
C,AKEEP(I,2,1),AKEEP(I,2,2),KEEP(I,2,3)
WRITE(6,503) AKEEP(I,3,1),AKEEP(I,3,2),KEEP(I,3,3)
C,AKEEP(I,4,1),AKEEP(I,4,2),KEEP(I,4,3)
WRITE(6,504) AKEEP(I,5,1),AKEEP(I,5,2),KEEP(I,5,3)
C,AKEEP(I,6,1),AKEEP(I,6,2),KEEP(I,6,3)
WRITE(6,505) AKEEP(I,7,1),AKEEP(I,7,2),KEEP(I,7,3)
C,AKEEP(I,8,1),AKEEP(I,8,2),KEEP(I,8,3)
WRITE(6,506) AKEEP(I,9,1),AKEEP(I,9,2),KEEP(I,9,3)
C,AKEEP(I,10,1),AKEEP(I,10,2),KEEP(I,10,3)
WRITE(6,507) AKEEP(I,11,1),AKEEP(I,11,2),KEEP(I,11,3)
507 FORMAT(1X,'TCZRO=',F10.2,2X,'STDZRO=',F8.2,2X,'NPTS=',I5)
WRITE(6,508) AKEEP(I,12,1),AKEEP(I,12,2),KEEP(I,12,3)
508 FORMAT(1X,'TAU1 =',F10.2,2X,'FT1RMS=',F8.4,2X,'NPTS=',I5)
10 CONTINUE
502 FORMAT(1X,'BTMAV=',F10.2,2X,'BTMRMS=',F8.2,2X,'NPTS=',I5,
C2X,'TCTAV=',F10.2,2X,'TCTRMS=',F8.2,2X,'NPTS=',I5)
503 FORMAT(1X,'FMZAV=',F10.2,2X,'FMZRMS=',F8.2,2X,'NPTS=',I5,
C2X,'FMEAV=',F10.2,2X,'FMERMS=',F8.2,2X,'NPTS=',I5)
504 FORMAT(1X,'SSVAV=',F10.2,2X,'SSVRMS=',F8.2,2X,'NPTS=',I5,
C2X,'FMZAV=',F10.2,2X,'FMZRMS=',F8.2,2X,'NPTS=',I5)
505 FORMAT(1X,'AVFMZ=',F10.2,2X,'AVRMS=',F8.2,2X,'NPTS=',I5,
C2X,'TAU =',F10.2,2X,'FITRMS=',F8.2,2X,'NPTS=',I5)
506 FORMAT(1X,'TEMAV=',F10.2,2X,'TEMRMS=',F8.4,2X,'NPTS=',I5,
C2X,'WTVAV=',F10.2,2X,'WTVRMS=',F8.4,2X,'NPTS=',I5)
RETURN
END
```

*****PASS1*****

C EDIT BY KCOA 31-MAR-78 17:10
 C SPLIT BRNEW in TWO, MAKING TWO SUBROUTINES: PASS1 AND PASS2.
 C PASS1 READS THE PRELIMINARY INFORMATION IN THE FILE; UPTO THE
 C FIRST RUN CARD. PASS 2 READS ONE COOLING CURVE AT A TIME.

C
 SUBROUTINE PASS1(ICODE, ICHAN, IDATA, IPASS, AVEFMZ, NBR, M, ISET, NBR1)
 DIMENSION AKEEP(30,15,3)
 DIMENSION KEEP(30,15,3)
 EQUIVALENCE(KEEP, AKEEP)
 COMMON /BROUT/AKEEP
 DIMENSION ICHAN(1500), IDATA(1500) !channel in parallel with data
 DIMENSION KSTORE(500) !Storage for FMZ data
 DIMENSION N(80)
 INTEGER IUNIT !Temporary unit number
 INTEGER IFMZC !Number of sets of FMZ's read
 INTEGER K !Number of FMZ points in KSTORE
 INTEGER KKOUT !No apparent purpose
 INTEGER NPCARD !Number of data points on one card
 INTEGER NPOINT !Number of data points read
 INTEGER IBEGIN !Temporary, first data point to read from card
 INTEGER ISAVE !Final count of FMZ points saved
 INTEGER IPASS !Pass 1 or 2
 INTEGER ICODE !Returns code indicating errors
 ! 0 for no error
 ! 1 for severe error (give up on file)
 ! 10 for unexpected EOF

 INTEGER ISET !Number of cooling curve to read, indexes KEEP
 INTEGER M !Input unit number
 INTEGER NBR !Returned unit number
 INTEGER NBR1 !Returned number of data points
 INTEGER COMBUF(200) !Storage for 10 comments
 INTEGER COMCNT !Number of comments found
 INTEGER COUNT !Number of
 INTEGER LBLANK !A Blank (' ')
 INTEGER ATCT
 INTEGER AUNT
 INTEGER AX, AX1, AX2
 INTEGER ACOM, ATID, ABTM, AFMZ, AFME, ASSV, ARUN, ACCU, AUC, AUB, AUA
 INTEGER AU1, AU2
 INTEGER ASIT
 INTEGER AUD, AUE, AUF, AERR, A999, AEND, ATIM
 INTEGER SANC
 INTEGER ATCA
 INTEGER AFVW
 DATA AFVW/4HFVW=/
 DATA ATCA/4HTCA=/
 DATA LBLANK/1H /
 DATA L0/1H0/
 DATA L1/1H1/
 DATA L2/1H2/
 DATA L9/1H9/
 DATA ATID/4HTID / !Marks tape ID
 DATA AUA/4HA / !Unit labels A-F, 1-2
 DATA AUB/4HB /
 DATA AUC/4HC /
 DATA AUD/4HD /
 DATA AUE/4HE /

DATA AUF/4HF /	
DATA AU1/4H 1 /	
DATA AU2/4H 2 /	
DATA ASIT/4HSIT /	!Site
DATA AUNT/4HUNT /	!Marks Unit label
DATA ABTM/4HBTM /	!Marks BTM data
DATA ATCT/4HTCT /	!Marks TXT data
DATA AFMZ/4HFMZ /	!Marks FMZ data
DATA AFME/4HFME /	!Marks FME data
DATA ASSV/4HSSV /	
DATA ARUN/4HRUN /	!Marks a RUN
DATA AERR/4HERR /	!Marks an ERROR
DATA A999/4HA999/	!Marks end of data
DATA AEND/4HEND /	!Also marks end of data
DATA ACCU/4HCCU /	!Marks CCU
DATA ATIM/4HTIM /	!Marks the TIME
DATA ACOM/4HCOM /	!Marks comments
DATA SANC/4HST. /	

C Subroutine to read data up to first run card

100 CONTINUE

KKOUT=0 !Is always 0!

L=ISET

ICODE = NUNIT+NOFMZ !Haven't read anything yet

LOCATE=1

IUNIT=1

C

C LOOK FOR TAPE ID TID

C

K=0 !Clear FMZ array

IFMZC=0 !and count of number of FMZ sets read

C READ CARDS UNTIL ONE BEGINS WITH 'TID' OR 'TIM' OR 'UNT' OR 'SIT' (BEGIN)

C OR ONE BEGINS WITH 'A999' OR 'END' (RETURN) OR WE REACH THE END OF FILE

200 CONTINUE

IUNIT=10

READ(M,300,END=3700) AX,(N(J),J=1,76)

300 FORMAT(A4,76A1)

IF(AX.EQ.ATID .OR. AX.EQ.ATIM .OR. AX.EQ.AUNT .OR. AX.EQ.ASIT)

GO TO 400

IF(AX.EQ.A999) IPASS=3

IF(IPASS.EQ.3) RETURN

IF(AX.EQ.AEND) RETURN

GO TO 200

400 CONTINUE

IF(KKOUT.EQ.0) WRITE(6,600) AX,(N(J),J=1,76) !KKOUT is always 0

READ(M,300,END=3900) AX,(N(J),J=1,76) !Read a card

500 CONTINUE

C If we see 'RUN' or 'CCU' or 'TCA=' or 'FVW=' then we're done

IF(AX.EQ.ARUN

1 .OR. AX.EQ.ACCU

2 .OR. AX.EQ.ATCA

3 .OR. AX.EQ.AFVW) RETURN

```

      ICODE=ICODE-NOUNIT
      IF(AX.EQ.SANC) LOCATE=2
      IF(AX.EQ.AUA) IUNIT=1
      IF(AX.EQ.AUB) IUNIT=2
      IF(AX.EQ.AUC) IUNIT=3
      IF(AX.EQ.AUD) IUNIT=4
      IF(AX.EQ.AU1) IUNIT=7
      IF(AX.EQ.AU2) IUNIT=8
      NBR=IUNIT

C If we just read a Data-Marker, then process data (at label 700)
      IF(AX.EQ.ASTM
1       .OR.AX.EQ.ATCT
2       .OR.AX.EQ.AFMZ
3       .OR.AX.EQ.AFME
4       .OR.AX.EQ.ASSV) GO TO 700

      IF(AX.NE.ACOM) GOTO 590 !Save comments into comment buffer
      ENCODE(80,580,COMBUF(COMCNT*20+1))AX,(N(J),J=1,76)
      COMCNT=COMCNT+1
580     FORMAT(A4,7GA1)

590     CONTINUE
      WRITE(6,600) AX,(N(J),J=1,76) !A comment card
600     FORMAT(10X,A4,7GA1)

      IF(AX.EQ.AEND) RETURN !The end of something
      IF(AX.EQ.A999) RETURN !Another end

C Otherwise, get another card
      GO TO 400

700     CALL INDATA(ICHAN,IDATA,LOCATE,AV,RMS,ICOUNT,NCODE)
      IF ((NCODE.AND.2).NE.0) GOTO 3900 !EOF
      IF ((NCODE.AND.1).NE.0) GOTO 3300 !ERROR

C     CHECK FOR ERR
C
      READ(M,300,END=3900) AX1,(N(J),J=1,76)

C READ FOLLOWING CARD. 'ERR' INDICATES PRECEDING SET OF DATA
C WAS ERRONEOUS. IN THAT CASE, CONTINUE SEARCHING

      IF(AX1.NE.AERR) GOTO 1200 !continue if no error
      AX=AX1 !else use last card to start new search
      GOTO 500

C
C
C     PROCESS IF DATA OK
C
C
C     CODE BEING PROCESSED
C
C
C
C     PRINT AV RMS
C
1200    KEEPIX = 0
      IF (AX.NE.ASTM) GOTO 1210
      WRITE(6,2600) AV,RMS,ICOUNT

```

```

      KEEPIX = 1
      GOTO 1250
1210 IF (AX .NE. ATCT) GOTO 1220
      WRITE(6,2700) AV,RMS,ICOUNT
      KEEPIX = 2
      GOTO 1250
1220 IF (AX .NE. AFMZ) GOTO 1230
C COPY THE FMZ DATA INTO KSTORE
      DO 1225 I=1, ICOUNT
        K=K+1
        KSTORE(K)=IDATA(I)
1225 CONTINUE
      WRITE(6,2800) AV,RMS,ICOUNT
      ICODE=ICODE-NOFMZ
      IF (IFMZC .EQ. 1) KEEPIX = 3
      IF (IFMZC .EQ. 2) KEEPIX = 6
      GOTO 1250
1230 IF (AX .NE. AFME) GOTO 1240
      WRITE(6,2900) AV,RMS,ICOUNT
      KEEPIX = 4
      GOTO 1250
1240 IF (AX .NE. ASSV) GOTO 1250
      WRITE(6,3000) AV,RMS,ICOUNT
      KEEPIX = 5

1250 CONTINUE
      AKEEP(ISET, KEEPIX, 1) = AV
      AKEEP(ISET, KEEPIX, 2) = RMS
      AKEEP(ISET, KEEPIX, 3) = ICOUNT


1400 CONTINUE
      SAVRMS=0.0
      SAVEAV=0.0
      DO 1500 I=1,K
        SAVRMS=SAVRMS+KSTORE(I)**2
        SAVEAV=SAVEAV+KSTORE(I)
1500 CONTINUE
      SAVEAV=SAVEAV/K
      SAVRMS=(SAVRMS/K)-SAVEAV**2
      SAVRMS=ABS(SAVRMS)
      SAVRMS=SQRT(SAVRMS)
      ISAVE=K
      AKEEP(ISET,7,1)=SAVEAV
      AKEEP(ISET,7,2)=SAVRMS
      KEEP(ISET,7,3)=ISAVE
      WRITE(6,1600) SAVEAV,SAVRMS,ISAVE
1600 FORMAT(20X,'**FMZAV=',F10.3,5X,'FMZRMS=',F10.3,5X,'NUM PTS=',I8)
C
C   REWIND AND SET ICODE =0   PASS=1 IS OK
C
      RETURN

C THESE FORMAT STATEMENTS ARE NO LONGER USED
2500 FORMAT(20X,' CAN NOT FIND RUN CODE   ****LAST CODE READ=',A4)
2600 FORMAT(20X,'AVBTM=',F10.3,5X,'RMSBTM=',F10.3,5X,'NUM OFPTS=',I8)
2700 FORMAT(20X,'AVTCT=',F10.3,5X,'RMSTCT=',F10.3,5X,'NUM OF PTS=',I8)
2800 FORMAT(20X,'AVFMZ=',F10.3,5X,'RMSFMZ=',F10.3,5X,'NUM OF PTS=',I8)
2900 FORMAT(20X,'AVFME=',F10.3,5X,'RMSFME=',F10.3,5X,'NUM OF PTS=',I8)

```

3000 FORMAT(20X,'AVSSV=',F10.3,5X,'RMSSSV=',F10.3,5X,'NUM OF PTS=',I8)

C Handle errors in data read in pass 1 by skipping that set of data

3300 CONTINUE

WRITE(6,3400) AX

3400 FORMAT(20X,' *** ERROR IN ',A4,' DATA ***')

GO TO 400

3700 CONTINUE

WRITE(6,3800)

3800 FORMAT(20X,' NO TID OR END SKIP THIS SET *****')

ICODE=10

RETURN

3900 CONTINUE

WRITE(6,4000)

4000 FORMAT(20X,' HIT END FILE OF DATA SET GO TO NEXT ONE **')

ICODE=10

RETURN

4100 CONTINUE

WRITE(6,4200)

4200 FORMAT(20X,' *** HIT END OF DATA NO 9 SKIP AND CONTINUE')

ICODE=10

RETURN

4300 CONTINUE

ICODE=10

WRITE(6,4400)

4400 FORMAT(20X,' HIT END OF FILE NO RUN OR CUU FORND CONTINUE ')

RETURN

4500 CONTINUE

ICODE=10

WRITE(6,4600)

4600 FORMAT(20X,' HIT END OF FILE WHILE READING CCU DATA SKIP')

RETURN

END

*****PET*****

```

C FILL IN THE ARRAYS XPETIT AND YPETIT
C XPETIT=TIME ARRAY
C YPETIT=' THERMOPILE VOLTAGE ' ARRAY
C WPTTIT= WEIGHT FOR YPETIT DATA PTS. SET EQUAL TO 1.0

```

```

SUBROUTINE PET(LBEGIN,NPTFIT,TAU2,V0,XPETIT,YPETIT,
1 WPTTIT,APAR,APARL1,APARL2)
DIMENSION XPETIT(NPTFIT),YPETIT(NPTFIT),WPTTIT(NPTFIT)
DIMENSION APAR(3),APARL1(3),APARL2(3)

```

```

COMMON /FITDTA/TFM(500),VFM(500),TTC(500),VTC(500),TTH(500),
1 VTH(500),TEMFB,VELFB,VFMZRO,STDZFO,VTCZRO,ITCEND

```

```

DO 6300 I=1,NPTFIT
L=LBEGIN + I -1
XPETIT(I)= TTC(L)
AVOL= VTC(L) + VTCZRO
YPETIT(I)= ABS( AVOL)
WPTTIT(I)= 1.0

```

```

6300 CONTINUE

```

```

C SET UP THE STARTING GUESS VALUES FOR PETITE

```

```

APAR(1)= VTCZRO
APAR(2)= V0 + VTCZRO
APAR(3)= TAU2
APARL1(1)= 0.0E00
APARL1(2)= 0.0E00
APARL1(3)= 0.0E00
APARL2(1) = 1.0E10
APARL2(2) = 1.0E10
APARL2(3) = 1.0E10

```

```

RETURN
END

```

*****PETITE*****

MODULE PETITE

EDIT BY KCOA 1-JUN-79

ADDED ARGUMENT IPRFLG , FLAG TO TELL WHETHER WE SHOULD
PRINT OR NOT

EDIT BY KCOA 6-SEP-78

FIXED CALLS TO FCHISQ TO INCLUDE NPTS

EDIT BY KCOA 29-AUG-78

CHANGED PRINT STATEMENTS TO WRITE(6,...)

EDIT BY KCOA 27-JUL-78

ADDED COMMON BLOCK PETTY2, AND LENGTHENED ARGUMENT STRING NEEDED
TO CALL PROGRS

EDIT BY KCOA 27-JUL-78 20:30

EXTENDED ARGUMENT STRING OF LIMSET TO INCLUDE ALIM1 AND ALIM2

CONVERSION TO FORTRAN-20 BY KEVIN CLINE

CONVERSION OF PETITE TO FORTRAN-20 MOSTLY REQUIRED THAT
THE USE OF INTERNAL SUBROUTINES (LEGAL ON THE 1108) BE
CONVERTED TO ANSI STANDARD, BY LENGTHENING SOME PARAMETER STRINGS
(FOR THOSE ROUTINES WHICH USE VARIABLES PASSED TO PETITE),
AND BY CREATING A COMMON AREA NAMED PETTY CONTAINING GLOBALS
IN PETITE NEEDED BY OTHER MODULES. SUBROUTINES INTERNAL TO PETITE
WERE: FCHISQ, LIMCHK, LIMSET, PROGRS.
END STATEMENTS WERE ALSO ADDED WHERE REQUIRED.

SUBROUTINE PETITE

PURPOSE

MAKE A LEAST-SQUARES FIT TO A NON-LINEAR FUNCTION
WITH A LINEARIZATION OF THE FITTING FUNCTION

USAGE

CALL PETITE (X, Y, WEIGHT, NPTS, NTERMS, A, SIGMAA, ALIM1, ALIM2, YFIT,
CHISQR, IPRFLG)

DESCRIPTION OF PARAMETERS

X - ARRAY OF DATA POINTS FOR INDEPENDENT VARIABLE
Y - ARRAY OF DATA POINTS FOR DEPENDENT VARIABLE
WEIGHT - ARRAY OF WEIGHTS FOR Y DATA POINTS . IF LESS THAN OR
EQUAL TO 0, THE POINT IS EXCLUDED FROM THE FIT
NPTS - NUMBER OF PAIRS OF DATA POINTS
NTERMS - NUMBER OF PARAMETERS
A - ARRAY OF PARAMETERS
SIGMAA - ARRAY OF STANDARD DEVIATIONS FOR PARAMETERS A
ALIM1 - LOWER PARAMETER LIMIT
ALIM2 - UPPER PARAMETER LIMIT
NOTE IF ALIM1 .GE. ALIM2 THE PARAMETER IS FIXED
YFIT - ARRAY OF CALCULATED VALUES OF Y
CHISQR - CHI SQUARE FOR FIT
FLAMDA - PROPORTION OF GRADIENT SEARCH INCLUDED
IPRFLG - 1 TO SHOW PROGRESS ON UNIT 6, 0 NOT TO

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

FUNCTN (X , A)

```

C      EVALUATES THE FITTING FUNCTION AT X FOR THE PARAMETERS A
C      FDERIV ( X , A , DERIV )
C      EVALUATES THE DERIVATIVES OF THE FITTING FUNCTION
C      AT X WITH RESPECT TO EACH PARAMETER
C      MATINV (ARRAY, NTERMS, DET)
C      INVERTS A SYMMETRIC TWO-DIMENSIONAL MATRIX OF DEGREE NTERMS
C      AND CALCULATES ITS DETERMINANT
C
C      COMMENTS
C      DIMENSION STATEMENT VALID FOR NTERMS UP TO 10
C
C      SUBROUTINE PETITE ( X, Y, WEIGHT, NPTS, NTERMS, A, SIGMAA, ALIM1,
$      ALIM2, YFIT, CHISQR, IPRFLG)
C
C      COMMON/PETTY/FCALL,NFIX,FIX,B,NIT2,FLAMDA,CHISQ1,DEL
C      DOUBLE PRECISION ARRAY
C      DIMENSION
C      D X(1), Y(1), WEIGHT(1), A(1), SIGMAA(1), ALIM1(1), ALIM2(1)
C      D,YFIT(1)
C      D,ALPHA(10,10), BETA(10), DERIV(10), ARRAY(10,10), B(10), SAVE(10)
C      D,FIX(10)
C
C      COMMON/PETTY2/NFREE,DFCALL,ARRAY,SAVE,DET,NVPAR
C
C      LOGICAL FIX
C
C      DATA MPRNT/2/
C      DATA MAXIT/80/
C      DATA CHICON/1.E-4/
C
C      FLAMDA = 0.01
C      NIT1 = 0
C      NIT2 = 0
C      FCALL = 0.
C      DFCALL = 0.
C
C      NN = 0
C      DO 10 I=1, NPTS
C      IF ( WEIGHT(I) .LE. 0. ) GO TO 10
C      NN = NN + 1
10  CONTINUE
C
C      DETERMINE WHICH PARAMETERS ARE FIXED
C
C      CALL LIMSET(NTERMS,ALIM1,ALIM2)
C      NVPAR = NTERMS - NFIX
C
C      NFREE = NN - NVPAR
C
C      IF (NFREE) 13, 13, 20
13  CHISQR = 0.
C      IF (IPRFLG.NE. 0) WRITE(6,199)NPTS,NN,NVPAR
199  FORMAT(1H0,/,1X,20(1H*), ' FIT NOT POSSIBLE ',21(1H*),/,1X, 3(1H*),
C      ' OUT OF ',15,' POINTS ONLY ',15,' HAVE POSITIVE WEIGHT '
C      3(1H*),/,1X,14(1H*), ' FIT REQUIRES',15,' PARAMETERS ',
C      15(1H*),/,1X,59(1H*))
C      RETURN
C

```

```

C      PRINT STARTING VALUES OF THE PARAMETERS AND LIMITS
C
20  CONTINUE
    LOOP = 1
    IF (IPRFLG.NE. 0)
1    WRITE(6,604) (I,A(I),ALIM1(I),ALIM2(I),I=1,NTERMS)
604  FORMAT(1H0,'INITIAL PARAMETER VALUES',12X,'LIMITS',
F      25(/,1X,16,3E15.5))
C
C      EVALUATE THE FITTING FUNCTION AND CHISQR
C
    CALL FCHISQ(A,X,Y,WEIGHT,YFIT,CHISQR,NPTS)
C
C
C      START MAIN LOOP
31  CONTINUE
    CHISQ1 = CHISQR
    FLAMDA = FLAMDA/10.
C
C      EVALUATE ALPHA AND BETA MATRICES
C
    DO 34 J=1, NVPAR
      BETA(J) = 0.
      DO 34 K=1, J
34    ALPHA(J,K) = 0.
C
    DO 50 I=1, NPTS
      IF ( WEIGHT(I) .LE. 0. ) GO TO 50
      WYDIF = WEIGHT(I) * ( Y(I) - YFIT(I) )
      CALL FDERIV ( X(I) , A , DERIV )
      DFCALL = DFCALL + 1.
C
      J = 0
      DO 47 JJ=1, NTERMS
        IF ( FIX(JJ) ) GO TO 47
        J = J + 1
        BETA(J) = BETA(J) + WYDIF * DERIV(JJ)
        K = 0
        DO 46 KK=1, JJ
          IF ( FIX(KK) ) GO TO 46
          K = K + 1
          ALPHA(J,K) = ALPHA(J,K) + WEIGHT(I) * DERIV(JJ) * DERIV(KK)
46      CONTINUE
47      CONTINUE
50      CONTINUE
C
      DO 53 J=2, NVPAR
        L = J - 1
        DO 53 K=1, L
53      ALPHA(K,J) = ALPHA(J,K)
C
C      INVERT MODIFIED CURVATURE MATRIX TO FIND NEW PARAMETERS
C
C      START SUB LOOP
70  CONTINUE
    DO 74 J=1, NVPAR
      DO 74 K=1, NVPAR
        IF ( J - K ) 71, 72, 73
71    ARRAY(J,K) = ALPHA(J,K) / SQRT (ALPHA(J,J) * ALPHA(K,K))

```

```

      GO TO 74
72  ARRAY(J,J) = 1. + FLAMDA
      GO TO 74
73  ARRAY(J,K) = ARRAY(K,J)
74  CONTINUE
C
      CALL MATINV ( ARRAY , NVPAR , DET )
C
      J = 0
      DO 85 JJ=1, NTERMS
        B(JJ) = A(JJ)
        IF ( FIX(JJ) ) GO TO 85
        J = J + 1
        DO 84 K=1, NVPAR
          IF ( J - K ) 81, 82, 83
81      ARRAY(J,K) = ARRAY(J,K) / SQRT ( ALPHA(J,J) * ALPHA(K,K) )
          GO TO 84
82      ARRAY(J,J) = ARRAY(J,J) / ABS ( ALPHA(J,J) )
          GO TO 84
83      ARRAY(J,K) = ARRAY(K,J)
84      B(JJ) = B(JJ) + BETA(K) * ARRAY(J,K)
85      CONTINUE
C
      MAKE SURE THAT THE PARAMETERS ARE IN RANGE
      CALL LIMCHK(NTERMS, ALIM1, ALIM2)
C
      EVALUATE THE FITTING FUNCTION AND CHISQR WITH THE NEW PARS
      CALL FCHISQ(B,X,Y,WEIGHT,YFIT,CHISQR,NPTS)
C
      IF CHI SQUARE INCREASED, INCREASE FLAMDA AND TRY AGAIN
      DEL = CHISQ1 - CHISQR
C  END SUB LOOP
      IF ( DEL ) 95, 101, 101
85  CONTINUE
      FLAMDA = 10. * FLAMDA
      NIT1 = NIT1 + 1
      IF (IPRFLG .NE. 0) WRITE(6,600) NIT2,NIT1,FLAMDA,CHISQ1,CHISQR,DEL,DET
600  FORMAT(1H0,10(1H*),2I3,' FLAMDA INCREASED TO',1PE8.2,' OLD CHISQ='
      F      ,1PE11.4,' NEW CHISQ=',1PE11.4,' DEL CHISQ=',1PE11.4,
      F      ,DET=',1PE11.4)
C  STAY IN SUB LOOP
      GO TO 70
C
C      EVALUATE NEW PARAMETERS
C
101  CONTINUE
      LOOP = 0
      DO 103 J=1, NTERMS
        SAVE(J) = A(J)
        A(J) = B(J)
103  CONTINUE
C
      NIT2 = NIT2 + 1
      IF(( MOD( NIT2 , MPRNT ) .EQ. 0).AND.(IPRFLG .EQ. 1))
1      CALL PROGRS(NTERMS,SIGMAA,A,CHISQR)
      IF ( NIT2 .GT. MAXIT ) GO TO 299

```

```

C   END MAIN LOOP

      IF ( DEL / CHISQ1 - CHICON ) 110, 110, 31

C   SUCCESS EXIT
110  CONTINUE
      IF (IPRFLG.NE.0) WRITE(6,166) NIT2
166  FORMAT(1H0,' PETITE HAS CONVERGED AFTER ',13,' ITERATIONS')
      IF (NIT1.NE.0.AND. (IPRFLG.NE.0)) WRITE(6,167) NIT1
167  FORMAT(1H+,43X,'AND ',13,' ADJUSTMENTS OF FLAMDA')

C   EVALUATE THE FITTING FUNCTION AT THE POINTS EXCLUDED FROM FIT
      DO 170 I=1, NPTS
      IF ( WEIGHT(I) .GT. 0. ) GO TO 170
      YFIT(I) = FUNCTN( X(I) , A.)
      FCALL = FCALL + 1.
170  CONTINUE
      IF( MOD( NIT2 , MPRNT ) .NE. 0 .AND. (IPRFLG.EQ. 1))
1    CALL PROGRS(NTERMS,SIGMAA,A,CHISQR)
      RETURN

C
C   LAST RESORT . USE PARAMETERS AND RESTART THE SEARCH
C
97   CONTINUE
      IF ( LOOP.EQ. 1 ) GO TO 299
      DO 98 J=1, NTERMS
      A(J) = SAVE(J)
98   CONTINUE
      WRITE(6,99)
99   FORMAT(1H0,'..... RESTART SEARCH .....')
      LOOP = 1
      FLAMDA = .8 * FLAMDA
      GO TO 20

C
C   FAILURE EXIT
C
299  WRITE(6,399) NIT2
399  FORMAT(1H0,20(1H*),' FAILURE TO CONVERGE AFTER ',13,' ITERATIONS',
      F      1X,20(1H*))
      IF( MOD( NIT2 , MPRNT ) .NE. 0 )
1    CALL PROGRS(NTERMS,SIGMAA,A,CHISQR)
      RETURN
      END

```

*****PLOTBR*****

```

C      EDIT BY KCOA 8-SEP-78
C      MODIFIED USPLX CALL TO FIT DEC IMSL.
C      RANGE AND SYM4 WERE TACKED ONTO THE END OF A WITH
C      AND EQUIVALENCE STATEMENT, AND A WAS REDIMENSIONED.
C
SUBROUTINE PLOTBR(ISET)
  DIMENSION AKEEP(30,15,3),KEEP(30,15,3)
  EQUIVALENCE(AKEEP,KEEP)
  COMMON /BROUT/AKEEP
  DIMENSION X(101),Y(51),IMAG4(101,51),RANGE(4)

C      ARRAY B CONTAINS A, RANGE, AND SYM4. B IS PASSED TO USPLH
C      IN THE STANDARD MANNER. THE MODIFICATION TO USPLH ON THE 1108
C      IS NO LONGER NECESSARY.

  INTEGER A(72,2),B(154),SYM4(10)

C      EQUIVALENCE STATEMENT TO LOCATE A, RANGE, SYM4 INSIDE B
  EQUIVALENCE (A(1,1),B(1)) , (B(145),RANGE(1)) , (B(149),SYM4(1))
  DO 30 J=1,4
    XMIN=1.0E20
    YMIN=1.0E20
    XMAX=-1.0E+20
    YMAX=-1.0E+20
    DO 10 I=1,ISET
      IF (J.EQ.1) X(I)=AKEEP(I,15,2)
      IF (J.EQ.1) Y(I)=100.
      IF (J.EQ.2) X(I)=AKEEP(I,15,1)
      IF (J.EQ.2) Y(I)=AKEEP(I,15,2)
      IF (J.EQ.3) X(I)=AKEEP(I,15,2)
      IF (J.EQ.3) Y(I)=AKEEP(I,8,2)
      IF (J.EQ.4) X(I)=AKEEP(I,14,1)
      IF (J.EQ.4) Y(I)=AKEEP(I,8,2)
    10 CONTINUE
    N=100
    M=1
    INC=1
    IA=100
    DO 11 I=1,ISET
      IF(X(I).GT.XMAX) XMAX=X(I)
      IF(Y(I).GT.YMAX) YMAX=Y(I)
      IF(Y(I).LT.YMIN) YMIN=Y(I)
      IF(X(I).LT.XMIN) XMIN=X(I)
    11 CONTINUE
    RANGE(1)=XMIN-.01*XMIN
    RANGE(2)=XMAX+.01*XMAX
    RANGE(3)=YMIN-.01*YMIN
    RANGE(4)=YMAX+.01*YMAX
    SYM4(1)=' '
    READ(5,20)(A(I,1),I=1,72)
    READ(5,20)(A(I,2),I=1,72)
  20 FORMAT(72A1)
    CALL USPLX(X,Y,N,M,INC,IA,B,IMAG4,IER)
  30 CONTINUE
  40 CONTINUE
    RETURN
  END

```

*****PROGRS*****


```

C      MODULE PROGRS (FORMERLY INTERNAL TO PETITE)
C
C      CONVERSION TO FORTRAN-20 BY KCOA
C
C      EDIT BY KCOA 8-SEP-78
C      FIXED PREVIOUS ERROR BY DIMENSIONING B(10)
C
C      EDIT BY KCOA 30-AUG-78 21:50
C      CHANGED PRINT STATEMENTS TO WRITE(6,...)
C
C      EDIT BY KCOA 27-JUL-78
C      PARAMETER STRING ADDED TO CALLING SEQUENCE, AND TWO
C      COMMON BLOCKS CREATED: PETTY, PETTY2.
C
C      1108 DECLARATION
C      SUBROUTINE PROGRS
C      FORTRAN-20 DECLARATION
C      SUBROUTINE PROGRS(NTERMS, SIGMAA, A, CHISQR)
C      INTEGER NTERMS
C      REAL CHISQR, A(1), SIGMAA(1)
C
C      NEW COMMON DECLARATIONS
C
C      CCOMMON/PETTY/FCALL,NFIX, FIX,B,NIT2,FLAMDA,CHISQ1,DEL
C      REAL FCALL, B(10), FLAMDA, CHISQ1, DEL
C      INTEGER NFIX, NIT2
C      LOGICAL FIX(10)
C      COMMON/PETTY2/NFREE,DFCALL,ARRAY,SAVE,DET,NVPAR
C      INTEGER NFREE,NVPAR
C      REAL DFCALL,SAVE(10),DET
C      DOUBLE PRECISION ARRAY(10,10)
C
C      WRITE(6,100)
100  FORMAT(1H ,130(1H=))
C      WRITE(6,150)NIT2,FLAMDA,CHISQR,CHISQ1,DEL
150  FORMAT(1H0,I3,' CURRENT VALUE OF FLAMDA=',1PE8.2,2X,' CHI SQ.=',
F      'TPE11.4,' PREVIOUS CHISQ=',1PE11.4,' DEL CHISQ=',1PE11.4)
C      CLEV = CONLEV ( NFREE , CHISQR )
C      WRITE(6,160),CLEV,NFREE
160  FORMAT(1H0,5X,' CONFIDENCE LEVEL ',F10.6,19,' DEGREES OF FREEDOM')
C
C      WRITE(6,170)FCALL,DFCALL
170  FORMAT(1H0,5X,' FUNCTN CALLED ', F8.1,' TIMES ; FDERIV CALLED ',
F      'F8.1,' TIMES')
C
C      EVALUATE PARAMETER UNCERTAINTIES
C
C      I = 0
C      DO 10 II=1, NTERMS
C      SIGMAA(II) = 0.
C      IF ( FIX(II) ) GO TO 10
C      I = I + 1
C      SIGMAA(II) = DSQRT ( ARRAY(I,I) )
10  CONTINUE
C      WRITE(6,200)
200  FORMAT(1H0,76X,'PARAMETER',8X,'ERROR',8X,'LAST STEP')
C      DO 15 J=1, NTERMS
C      STP = A(J) - SAVE(J)
C      WRITE(6,210)J,A(J),SIGMAA(J),STP

```

```
210 FORMAT(61X,I10,3E15.6)
15  CONTINUE
    WRITE(6,253)DET
250 FORMAT(1H , ' DETERMINANT = ',1PE11.4)
    DO 20 K=1, NVPAR
20  WRITE(6,300) (ARRAY(J,K),J=1,K)
300 FORMAT(1H0,10D13.5)
    WRITE(6,100)
    RETURN
    END
```

*****RESULT3*****

```

C      CONVERSION TO TOPS-20 BY KCOA 20-JUL-78
C
SUBROUTINE RESUL3(II)
  DIMENSION IRUN(25), TW(25), DELTW(25), VW(25), DELVW(25), CHISQ(25)
  DIMENSION TCON(25), H(25), HR(25)
  COMMON/FIT3/IRUN, TW, DELTW, VW, DELVW, CHISQ, TCON, H, HR
  COMMON /BLK2/ TITLE(20)
  COMMON /LOOK/THINT, THSLP, AVOLFM, VEXCIT, ACCN, BCCN, CCCN, DCCN,
  IRAD1, RAD2, AKTUBE, TTHWLL, HINTR, TUBEID, FMGAIN, VNOM, HRSLPE

  DIMENSION IRUNAC(25), TWAC(25), DELTWA(25), VWAC(25), DELVWA(25),
  1CHISQA(25), TCONAC(25), HACP(25), HRACP(25)
  TNOM=70.
  WRITE (6,199) TITLE
199  FORMAT(1H1,20X,20A4)
  WRITE (6,200)
200  FORMAT(1X,/,10X,'SUMMARY OF RESULTS -- OF ALL THE COOLING CURVES
  1',/))
  WRITE (6,201)
201  FORMAT (1X,'RUN NO.',5X,'WATER TEMP.',3X,'RMS WATER TEMP.',
  15X,'FLOW VEL.',3X,'RMS FLOW VEL.',5X,'CHISQ',6X,'TAU',8X,
  2'H',9X,'1/H',/))
  DO 13 K=1,II
  WRITE (6,202) IRUN(K),TW(K),DELTW(K),VW(K),DELVW(K),
  1CHISQ(K),TCON(K),H(K),HR(K)
202  FORMAT(1X,I4,10X,F7.4,8X,F8.4,10X,F7.3,6X,F8.4,6X,F9.3,2X,F7.3,2X,
  1E9.4,2X,E9.4)
13  CONTINUE
  IAC= 0
  DO 10 J=1,II
  IAC=IAC + 1
  IRUNAC(IAC)= J
  TWAC(IAC)=TW(J)
  DELTWA(IAC)=DELTW(J)
  VWAC(IAC)=VW(J)
  DELVWA(IAC)=DELVW(J)
  CHISQA(IAC)=CHISQ(J)
  TCONAC(IAC)=TCON(J)
  HACP(IAC)=H(J)
  HRACP(IAC)=HR(J)
10  CONTINUE
C  COMPUTATION OF AVERAGES AND ERRORS ON AVERAGES ---
C  ALL THE COOLING CURVES ARE ACCEPTED IN THIS PASS
  SUMT=0.
  SUMV=0.
  SUMH=0.
  SUMHR=0.
  DO 30 M=1, IAC
  SUMT=SUMT + TWAC(M)
  SUMV=SUMV + VWAC(M)
  SUMH=SUMH + HACP(M)
  SUMHR=SUMHR + HRACP(M)
30  CONTINUE
  AVET= SUMT / IAC
  AVEV= SUMV / IAC
  AVEH= SUMH / IAC
  AVEHR= SUMHR / IAC
  SUMSQT=0.
  SUMSQV=0.

```

```

SUMSQH=0.
SMSQHR=0.
DO 31 N=1,IAC
SUMSQT=SUMSQT + (TWAC(N) - AVET) ** 2
SUMSQV=SUMSQV + (VWAC(N) - AVEV) ** 2
SUMSQH=SUMSQH + (HACP(N) - AVEH) ** 2
SMSQHR= SMSQHR + (HRACP(N) - AVEHR) **2
31 CONTINUE
AC=IAC
ERRH=SQRT ( SUMSQH / (IAC-1) ) * (1.0 / SQRT( AC))
ERRV=SQRT ( SUMSQV / (IAC-1) ) * (1.0 / SQRT( AC))
ERRT=SQRT ( SUMSQT / (IAC-1) ) * (1.0 / SQRT( AC))
ERRHR=SQRT( SMSQHR / (IAC-1) ) * (1.0 / SQRT( AC))
WRITE (6,80) II
80 FORMAT(///,5X,'TOTAL NO OF COOLING CURVES ANALYSED' =',13)
WRITE (6,81) IAC
81 FORMAT(5X,'TOTAL NO OF ACCEPTED COOLING CURVES' =',13)
WRITE (6,78)
78 FORMAT(5X,' ALL THE COOLING CURVES ARE ACCEPTED IN THIS PASS' /)
WRITE (6,82) AVET,ERRT
82 FORMAT(5X,'TEMP. OF WATER (IN C)=' ,F6.3,2X,'PLUS OR MINUS',
11X,F6.4)
WRITE (6,83) AVEV,ERRV
83 FORMAT(5X,'VELOCITY OF WATER(IN FT./SEC)=' ,F6.3,2X,
1'PLUS OR MINUS',1X,F6.4,/)
WRITE (6,86) TNOM,VNOM,HRSLPE
86 FORMAT(5X,'NQRMALISATION TEMP.' =',F8.3,2X,'DEG.F',/,
15X,'NORMALISATION VELOCITY=' ,F8.3,2X,'FT./SEC.',/,
25X,'1/H VS 1/(V**.8) SLOPE=' ,E12.5,/)
WRITE(6,84) AVEH,ERRH
84 FORMAT(5X,' H (BTU/FT.**2 HR F )=' ,E12.5,2X,'PLUS OR MINUS',
11X,E12.5)
WRITE (6,85) AVEHR,ERRHR
85 FORMAT(5X,'1/H (FT.**2 HR F /BTU)=' ,E12.5,2X,
1'PLUS OR MINUS ',2X,E12.5,/)

C
C RE-DO THE ABOVE AVERAGING WITH THE NEW ACCEPTANCE CRITERION
C REJECT A COOLING CURVE IF THE DEVIATION IN H FROM ITS MEAN IS MORE THAN
C TWO STANDARD DEVIATIONS
C
HMEAN=AVEH
HSIGMA=SQRT(SUMSQH/(II-1))
IAC= 0
DO 110 J=1,II
HDV=ABS( H(J) - HMEAN)
TSTPAR=(HDV/HSIGMA)
IF (TSTPAR.GT. 2.0 ) GO TO 110
IAC=IAC + 1
IRUNAC(IAC)= J
TWAC(IAC)=TW(J)
DELTWA(IAC)=DELTW(J)
VWAC(IAC)=VW(J)
DELVWA(IAC)=DELVW(J)
CHISQA(IAC)=CHISQ(J)
TCONAC(IAC)=TCON(J)
HACP(IAC)=H(J)
HRACP(IAC)=HR(J)
110 CONTINUE
WRITE (6,199) TITLE
WRITE (6,70)

```

```

70  FOR... 1X, //, 10X, 'SUMMARY OF RESULTS -- OF ACCEPTED COOLING CURVE
    1S', // )
    WRITE (6,71)
71  FORMAT (1X, 'RUN NO.', 5X, 'WATER TEMP.', 3X, 'FMS WATER TEMP.',
    15X, 'FLOW VEL.', 3X, 'RMS FLOW VEL.', 5X, 'CHISQ', 6X, 'TAU', 8X,
    2'H', 9X, '1/H', //)
    DO 29 L=1, IAC
    WRITE(6,72) IRUNAC(L), TWAC(L), DELTWA(L), VWAC(L), DELVWA(L),
    1CHISQA(L), TCONAC(L), HACP(L), HRACP(L)
72  FORMAT(1X, 14, 10X, F7.4, 8X, F8.4, 10X, F7.3, 6X, FE.4, 6X, F9.3, 2X, F7.3, 2X,
    1E9.4, 2X, E9.4. )
29  CONTINUE
C  COMPUTATION OF AVERAGES AND ERRORS ON AVERAGES ---
C  ONLY ACCEPTED COOLING CURVES ARE USED
    SUMT=0.
    SUMV=0.
    SUMH=0.
    SUMHR=0.
    DO 130 M=1, IAC
    SUMT=SUMT + TWAC(M)
    SUMV=SUMV + VWAC(M)
    SUMH=SUMH + HACP(M)
    SUMHR=SUMHR + HRACP(M)
130 CONTINUE
    AVET= SUMT / IAC
    AVEV= SUMV / IAC
    AVEH= SUMH / IAC
    AVEHR= SUMHR / IAC
    SUMSQT=0.
    SUMSQV=0.
    SUMSQH=0.
    SMSQHR=0.
    DO 131 N=1, IAC
    SUMSQT=SUMSQT + (TWAC(N) - AVET) ** 2
    SUMSQV=SUMSQV + (VWAC(N) - AVEV) ** 2
    SUMSQH=SUMSQH + (HACP(N) - AVEH) ** 2
    SMSQHR= SMSQHR + (HRACP(N) - AVEHR) **2
131 CONTINUE
    AC=IAC
    ERRT=SQRT ( SUMSQT / (IAC-1) ) * (1.0 / SQRT( AC))
    ERRV=SQRT ( SUMSQV / (IAC-1) ) * (1.0 / SQRT( AC))
    ERRH=SQRT ( SUMSQH / (IAC-1) ) * (1.0 / SQRT( AC))
    ERRHR=SQRT( SMSQHR / (IAC-1) ) * (1.0 / SQRT( AC))
    WRITE (6,80) II
    WRITE (6,81) IAC
    WRITE (6,178)
178 FORMAT(5X, 'REJECT A COOLING CURVE IF THE DEVIATION IN H FROM ITS
    1MEAN IS MORE THAN TWO STANDARD DEVIATIONS', //)
    WRITE (6,82) AVET, ERRT
    WRITE (6,83) AVEV, ERRV
    WRITE (6,86) TNOM, VNOM, HRSLE
    WRITE(6,84) AVEH, ERRH
    WRITE (6,85) AVEHR, ERRHR
    RETURN
    END

```

*****RESULT*****

C
C

CONVERSION TO TOPS-20 BY KCOA 20-JUL-78

```

SUBROUTINE RESULT
DIMENSION IRUN(25),TW(25),DELTW(25),VW(25),DELVW(25),CHISQ(25)
DIMENSION TCON(25),H(25),HR(25)
COMMON IRUN,TW,DELTW,VW,DELVW,CHISQ,TCON,H,HR
COMMON /BLK1/ II
COMMON /BLK2/ TITLE(20)
COMMON /LOOK/THINT,THSLP,AVOLFM,VEXCIT,ACCN,BCCN,CCCN,DCCN,
1RAD1,RAD2,AKTUBE,TTHWLL,HINTR,TUBEID,FMGAIN,VNOM,HRSLEPE

DIMENSION IRUNAC(25),TWAC(25),DELTWA(25),VWAC(25),DELVWA(25),
1CHISQA(25),TCONAC(25),HACP(25),HRACP(25)
TNOM=70.
WRITE (6,199) TITLE
199 FORMAT(1H1,20X,20A4)
WRITE (6,200 )
200 FORMAT(1X,/,10X,'SUMMARY OF RESULTS -- OF ALL THE COOLING CURVES
1',/)
WRITE (6,201 )
201 FORMAT (1X,'RUN NO.',5X,'WATER TEMP.',3X,'RMS WATER TEMP.',
15X,'FLOW VEL.',3X,'RMS FLOW VEL.',5X,'CHISQ',6X,'TAU',8X,
2'H',9X,'1/H',/)
DO 13 K=1,II
WRITE (6,202) IRUN(K),TW(K),DELTW(K),VW(K),DELVW(K),
1CHISQ(K),TCON(K),H(K),HR(K)
202 FORMAT(1X,I4,10X,F7.4,8X,F8.4,10X,F7.3,6X,F8.4,6X,F9.3,2X,F7.3,2X,
1E9.4,2X,E9.4 )
13 CONTINUE
IAC= 0
DO 10 J=1,II
IAC=IAC + 1
IRUNAC(IAC)= J
TWAC(IAC)=TW(J)
DELTWA(IAC)=DELTW(J)
VWAC(IAC)=VW(J)
DELVWA(IAC)=DELVW(J)
CHISQA(IAC)=CHISQ(J)
TCONAC(IAC)=TCON(J)
HACP(IAC)=H(J)
HRACP(IAC)=HR(J)
10 CONTINUE
C COMPUTATION OF AVERAGES AND ERRORS ON AVERAGES ---
C ALL THE COOLING CURVES ARE ACCEPTED IN THIS PASS
SUMT=0.
SUMV=0.
SUMH=0.
SUMHR=0.
DO 30 M=1,IAC
SUMT=SUMT + TWAC(M)
SUMV=SUMV + VWAC(M)
SUMH=SUMH + HACP(M)
SUMHR=SUMHR + HRACP(M)
30 CONTINUE
AVET= SUMT / IAC
AVEV= SUMV / IAC
AVEH= SUMH / IAC
AVEHR= SUMHR / IAC
SUMSQT=0.

```



```

SUMSQV=0.
SUMSQH=0.
SMSQHR=0.
DO 31 N=1,IAC
SUMSQT=SUMSQT + (TWAC(N) - AVET) ** 2
SUMSQV=SUMSQV + (VWAC(N) - AVEV) ** 2
SUMSQH=SUMSQH + (HACP(N) - AVEH) ** 2
SMSQHR= SMSQHR + (HRACP(N) - AVEHR) **2
31 CONTINUE
AC=IAC
ERRH=SQRT ( SUMSQH / (IAC-1) ) * (1.0 / SQRT( AC))
ERRV=SQRT ( SUMSQV / (IAC-1) ) * (1.0 / SQRT( AC))
ERRT=SQRT ( SUMSQT / (IAC-1) ) * (1.0 / SQRT( AC))
ERRHR=SQRT( SMSQHR / (IAC-1) ) * (1.0 / SQRT( AC))
WRITE (6,80) II
80 FORMAT(///,5X,'TOTAL NO OF COOLING CURVES ANALYSED  =',I3)
WRITE (6,81) IAC
81 FORMAT(5X,'TOTAL NO OF ACCEPTED COOLING CURVES =',I3)
WRITE (6,78)
78 FORMAT(5X,' ALL THE COOLING CURVES ARE ACCEPTED IN THIS PASS ' //)
WRITE (6,82) AVET,ERRT
82 FORMAT(5X,'TEMP. OF WATER (IN C)=',F6.3,2X,'PLUS OR MINUS',
11X,F6.4)
WRITE (6,83) AVEV,ERRV
83 FORMAT(5X,'VELOCITY OF WATER(IN FT./SEC )=',F6.3,2X,
1'PLUS OR MINUS',1X,F6.4,/)
WRITE (6,86) TNOM,VNOM,HRSLPE
86 FORMAT(5X,'NORMALISATION TEMP.=',F8.3,2X,'DEG.F',/,
15X,'NORMALISATION VELOCITY=',F8.3,2X,'FT./SEC.',/,
25X,'1/H VS 1/(V**.8) SLOPE=',E12.5,/)
WRITE(6,84) AVEH,ERRH
84 FORMAT(5X,' H (BTU/FT.**2 HR F )=',E12.5,2X,'PLUS OR MINUS',
11X,E12.5)
WRITE (6,85) AVEHR,ERRHR
85 FORMAT(5X,'1/H (FT.**2 HR F /BTU )=',E12.5,2X,
1'PLUS OR MINUS ',2X,E12.5,/)
C
C RE-DO THE ABOVE AVERAGING WITH THE NEW ACCEPTANCE CRITERION
C REJECT A COOLING CURVE IF THE DEVIATION IN H FROM ITS MEAN IS MORE THAN
C TWO STANDARD DEVIATIONS
C
HMEAN=AVEH
HSIGMA=SQRT(SUMSQH/(II-1))
IAC= 0
DO 110 J=1,II
HDV=ABS( H(J) - HMEAN)
TSTPAR=(HDV/HSIGMA)
IF (TSTPAR .GT. 2.0 ) GO TO 110
IAC=IAC + 1
IRUNAC(IAC)= J
TWAC(IAC)=TW(J)
DELTWA(IAC)=DELTW(J)
VWAC(IAC)=VW(J)
DELVWA(IAC)=DELVW(J)
CHISQA(IAC)=CHISQ(J)
TCONAC(IAC)=TCON(J)
HACP(IAC)=H(J)
HRACP(IAC)=HR(J)
110 CONTINUE
WRITE (6,199) TITLE

```

```

WRITE (6,70)
70  FORMAT(1X,/,10X,'SUMMARY OF RESULTS -- OF ACCEPTED COOLING CURVE
15',/,)
WRITE (6,71)
71  FORMAT (1X,'RUN NO.',5X,'WATER TEMP.',3X,'RMS WATER TEMP.',
15X,'FLOW VEL.',3X,'RMS FLOW VEL.',5X,'CHISQ',6X,'TAU',8X,
2'H',9X,'1/H',/)
DO 29 L=1,IAC
WRITE(6,72) IRUNAC(L),TWAC(L),DELTWA(L),VWAC(L),DELVWA(L),
1CHISQA(L),TCONAC(L),HACP(L),HRACP(L)
72  FORMAT(1X,14,10X,F7.4,8X,F8.4,10X,F7.3,6X,F8.4,6X,F9.3,2X,F7.3,2X,
1E9.4,2X,E9.4.)
29  CONTINUE
C  COMPUTATION OF AVERAGES AND ERRORS ON AVERAGES ---
C  ONLY ACCEPTED COOLING CURVES ARE USED
SUMT=0.
SUMV=0.
SUMH=0.
SUMHR=0.
DO 130 M=1,IAC
SUMT=SUMT + TWAC(M)
SUMV=SUMV + VWAC(M)
SUMH=SUMH + HACP(M)
SUMHR=SUMHR + HRACP(M)
130 CONTINUE
AVET= SUMT / IAC
AVEV= SUMV / IAC
AVEH= SUMH / IAC
AVEHR= SUMHR / IAC
SUMSQT=0.
SUMSQV=0.
SUMSQH=0.
SMSQHR=0.
DO 131 N=1,IAC
SUMSQT=SUMSQT + (TWAC(N) - AVET) ** 2
SUMSQV=SUMSQV + (VWAC(N) - AVEV) ** 2
SUMSQH=SUMSQH + (HACP(N) - AVEH) ** 2
SMSQHR= SMSQHR + (HRACP(N) - AVEHR) **2
131 CONTINUE
AC=IAC
ERRT=SQRT ( SUMSQT / (IAC-1) ) * (1.0 / SQRT( AC))
ERRV=SQRT ( SUMSQV / (IAC-1) ) * (1.0 / SQRT( AC))
ERRH=SQRT ( SUMSQH / (IAC-1) ) * (1.0 / SQRT( AC))
ERRHR=SQRT( SMSQHR / (IAC-1) ) * (1.0 / SQRT( AC))
WRITE (6,80) II
WRITE (6,81) IAC
WRITE (6,178)
178  FORMAT(5X,'REJECT A COOLING CURVE IF THE DEVIATION IN H FROM ITS
1MEAN IS MORE THAN TWO STANDARD DEVIATIONS',/)
WRITE (6,82) AVET,ERRT
WRITE (6,83) AVEV,ERRV
WRITE (6,85) TNOM,VNOM,HRSLPE
WRITE(6,84) AVEH,ERRH
WRITE (6,85) AVEHR,ERRHR
RETURN
END

```

*****RNORM*****

```

FUNCTION RNORM(X)
  DIMENSION A(7)
  DATA(A(I),I=1,7)/.430638E-4,.2765672E-3,.1520143E-3,.92705272E-2,
1.422820123E-1,.705230784E-1,1.0/

```

```

C-----
C  TO COMPUTE A RATIONAL FUNCTION APPROXIMATION TO THE NORMAL
C  DISTRIBUTION
C-----

```

```

  Y=ABS(X)/1.41421356
  RNORM=0.0
  DO 1 I=1,7
    RNORM=RNORM*Y+A(I)
  CALL OVERFL(INDCT)
  IF(INDCT.EQ.1) GO TO 3
1  CONTINUE
  RNORM=.5*(((( 1.0/RNORM)**2)**2)**2)**2)
2  IF(X.GT.0.0) RNORM=1.0-RNORM
  RETURN
3  RNORM=0.0
  GO TO 2
END

```

*****SAVFIT*****

```
SUBROUTINE SAVFIT(LBEGIN,TMAX,TMIN,TAU2,TAUERR,AVEFW,AVETW,NPTFIT,  
1STDFB,STDTW,VO,YBRRMS,APAR,SIGPAR,RELDIF,NIT1,NIT2)
```

C Uses N as index into FSAVE, IFSAVE to save all the other parameters
C for later retrieval when the best window has been determined.

```
COMMON /FITSV/ FSAVE(50)  
DIMENSION IFSAVE(50)  
EQUIVALENCE (FSAVE,IFSAVE)  
  
DIMENSION APAR(3), SIGPAR(3)
```

```
FSAVE(1) = LBEGIN  
FSAVE(2) = TMAX  
FSAVE(3) = TMIN  
FSAVE(4) = TAU2  
FSAVE(5) = TAUERR  
FSAVE(6) = AVEFW  
FSAVE(7) = AVETW  
IFSAVE(8) = NPTFIT  
FSAVE(9) = STDFB  
FSAVE(10) = STDTW  
FSAVE(11) = VO  
FSAVE(12) = YBRRMS  
FSAVE(13) = APAR(1)  
FSAVE(14) = APAR(2)  
FSAVE(15) = APAR(3)  
FSAVE(16) = SIGPAR(1)  
FSAVE(17) = SIGPAR(2)  
FSAVE(18) = SIGPAR(3)  
FSAVE(19) = RELDIF  
IFSAVE(20) = NIT1  
IFSAVE(21) = NIT2
```

```
RETJRN  
END
```

SUBROUTINE GETFIT(LBEGIN,TMAX,TMIN,TAU2,TAUERR,AVEFW,AVETW,NPTFIT,
1STDFB,STDW,V0,YBRRMS,APAR,SIGPAR,RELDIF,NIT1,NIT2)

COMMON /FITSV/ FSAVE(50)
DIMENSION IFSAVE(50)
EQUIVALENCE (FSAVE,IFSAVE)

DIMENSION APAR(3), SIGPAR(3)

LBEGIN = FSAVE(1)
TMAX = FSAVE(2)
TMIN = FSAVE(3)
TAU2 = FSAVE(4)
TAUERR = FSAVE(5)
AVEFW = FSAVE(6)
AVETW = FSAVE(7)
NPTFIT = IFSAVE(8)
STDFB = FSAVE(9)
STDW = FSAVE(10)
V0 = FSAVE(11)
YBRRMS = FSAVE(12)
APAR(1) = FSAVE(13)
APAR(2) = FSAVE(14)
APAR(3) = FSAVE(15)
SIGPAR(1) = FSAVE(16)
SIGPAR(2) = FSAVE(17)
SIGPAR(3) = FSAVE(18)
RELDIF = FSAVE(19)
NIT1 = IFSAVE(20)
NIT2 = IFSAVE(21)

RETURN
END

*****STORE*****

C
C

CONVERSION TO TOPS-20 BY KC0A 20-JUL-78

```
SUBROUTINE STORE ( ISET,AVETW,STDTW,AVEFV,STDFV,TAU,HCORNR,  
1HCORN,CHITOT )  
DIMENSION IRUN(25),TW(25),DELTW(25),VW(25),DELVW(25),CHISQ(25)  
DIMENSION TCON(25),H(25),HR(25)  
COMMON IRUN,TW,DELTW,VW,DELVW,CHISQ,TCON,H,HR  
COMMON /BLK1/ II  
II=ISET  
IRUN(II)=ISET  
TW(II)=AVETW  
DELTW(II)=STDTW  
VW(II)=AVEFV  
DELVW(II)=STDFV  
CHISQ(II)=CHITOT  
TCON(II)=TAU  
H(II)=HCORN  
HR(II)=HCORNR  
RETURN  
END
```

*****STORE3*****

C CONVERSION TO TOPS-20 BY KCOA 20-JUL-78
C

```
SUBROUTINE STORE3(ISET,AVETW,STDTW,AVEFV,STDFV,TAU,HCORNR,  
1HCORN,CHITOT)  
DIMENSION IRUN(25),TW(25),DELTW(25),VW(25),DELVW(25),CHISQ(25)  
DIMENSION TCON(25),H(25),HR(25)  
COMMON/FIT3/IRUN,TW,DELTW,VW,DELVW,CHISQ,TCON,H,HR  
COMMON /BLK1/ II  
II=ISET  
IRUN(II)=ISET  
TW(II)=AVETW  
DELTW(II)=STDTW  
VW(II)=AVEFV  
DELVW(II)=STDFV  
CHISQ(II)=CHITOT  
TCON(II)=TAU  
H(II)=HCORN  
HR(II)=HCORNR  
RETURN  
END
```

END

```
***** END *****
***** END *****
***** END *****
***** END *****
***** END *****
***** END *****
***** END *****
```

0378

0378 P474 GG20
GG20
Pages printed 100

20-JUL-79 13:27:38
PS: <GG20>BIG.FOR.2

GG20 P474 0378
GG20

0378

0378

[illegible][illegible]

```

#####Carnegie-Mellon University Tops-C, TOPS-20 Monitor 4(2772)-2 #####

```

APPENDIX B1

Version 1.1 of the TI-59 analysis program

PROGRAM DESCRIPTION

Zero corrected logs ($\ln(V - V_0)$) are entered through the Σ^+ function. These logs are fit with an equally weighted least squares form giving $\ln(V_i - V_0) = A + Bi$. The parameters A and B and their errors are printed out, B is converted to τ ($\tau = -2/B$), then τ is converted to h using the form $\ln h = A_0 + A_1 \ln \tau + A_2 (\ln \tau)^2 + A_3 (\ln \tau)^3$ (continued on P 3)

USER INSTRUCTIONS

STEP	PROCEDURE	ENTER	PRESS	DISPLAY
1	Set Partitioning	5	2 nd OP 17	559.49
2	Read 4 card sides	0	INV 2 nd Write	1,2,3,4
3	Initialize statistics	2 nd , OP Pgm 01, SBR, CLR		0
4	Enter zero corrected logs.	$\ln(V_i - V_0)$	Σ^+	i
	Continue P 3			

7.6309	Σ^+	45.	A	Complete output from A, $\tau = 45$ entered
1.		1204.766786	HU	
7.5528	Σ^+	1173.082724	H	
2.		.0008524548	1/H	
7.4755	Σ^+	T?		
3.		25.		
7.3989	Σ^+	R(70)		
4.		.0008913712		
7.3198	Σ^+			
5.				
-0.07761	SLP			
.0002118962	+/-	V?		
7.6308	INT	7.		
.0005190376	+/-	VN?		
		6.		
25.76987502	TAU	R(VN)		
		.0009918691		

USER DEFINED KEYS	DATA REGISTERS (INV)	LABELS (Op 08)
Output from	3.80666249 00	162 12 B
ln entry and	1204.766786 Temp 01	204 14 D
T calculation	.0008524548 storage 02	247 11 A
(Trace on	.0008913712 used for Σ^+ 03	289 13 C
where Σ^+ is	7. and 04	331 22 INV
printed)	6. stop 05	420 23 LN
	73.9797 etc. 06	465 24 CE
	7.6308 07	
	-0.07761 08	
	3555423156. R(VN) 09	
FLAGS

NAME: AMMER

DATE

CODE	KEY	COMMENTS	LOC	CODE	KEY	COMMENTS	LOC	CODE	KEY	COMMENTS
000	69	DP		055	65	X		110	07	7
001	12	12		056	43	RCL		111	06	6
002	42	STD <i>Intercept</i>		057	07	07		112	03	3
003	07	07		058	65	X		113	02	2
004	32	X:T		059	43	RCL		114	00	0
005	42	STD <i>slope</i>		060	08	08		115	69	DP
006	08	08		061	65	X		116	04	04
007	43	RCL		062	02	2		117	42	STD
008	05	05 <i>Steps</i>		063	54	)		118	33	33
009	65	X		064	55	÷		119	43	RCL
010	43	RCL		065	53	(		120	32	32
011	03	03 <i>take sums</i>		066	43	RCL		121	69	DP
012	75	- <i>created</i>		067	03	03		122	06	06
013	43	RCL		068	75	-		123	02	2
014	04	04 <i>by 1/2 and</i>		069	02	2		124	04	4
015	33	X² <i>make a</i>		070	54	)		125	03	3
016	95	= <i>linear</i>		071	95	=		126	01	1
017	42	STD <i>ression</i>		072	42	STD		127	03	3
018	39	39		073	30	30		128	07	7
019	53	(<i>fit. and</i>		074	65	X		129	69	DP
020	43	RCL		075	43	RCL		130	04	04
021	02	02 <i>calculate</i>		076	05	.05		131	43	RCL
022	85	+ <i>the error</i>		077	55	÷		132	07	07
023	43	RCL		078	43	RCL		133	69	DP
024	07	07 <i>on the</i>		079	39	39		134	06	06
025	33	X² <i>slope and</i>		080	95	=		135	43	RCL
026	65	X <i>intercept.</i>		081	34	FX		136	33	33
027	43	RCL		082	42	STD		137	69	DP
028	03	03		083	31	31		138	04	04
029	85	+ <i>see</i>		084	32	X:T		139	43	RCL
030	43	RCL <i>Bevington</i>		085	43	RCL		140	31	31
031	05	05 <i>chapter 6.</i>		086	30	30		141	69	DP
032	65	X		087	65	X		142	06	06
033	43	RCL <i>The slope</i>		088	43	RCL		143	43	RCL
034	08	08		089	03	03		144	08	08
035	33	X² <i>is then</i>		090	55	÷		145	35	1/X
036	75	- <i>converted</i>		091	43	RCL		146	65	X
037	43	RCL		092	39	39		147	02	2
038	01	01 <i>to T.</i>		093	95	=		148	94	+/-
039	65	X		094	34	FX		149	95	=
040	43	RCL <i>and slope,</i>		095	42	STD		150	32	X:T
041	07	07 <i>intercept and</i>		096	32	32		151	43	RCL
042	65	X <i>their errors</i>		097	03	3		152	38	38
043	02	2 <i>are printed</i>		098	06	6		153	69	DP
044	75	- <i>out as</i>		099	02	2		154	04	04
045	43	RCL		100	07	7		155	98	ADV
046	06	06 <i>is T.</i>		101	03	3		156	32	X:T
047	65	X		102	03	3		157	69	DP
048	43	RCL		103	69	DP		158	06	06
049	08	08		104	04	04		159	98	ADV
050	65	X		105	43	RCL				
051	02	2		106	08	08				
052	85	+		107	69	DP				
053	43	RCL		108	06	06				
054	04	04		109	04	4				

MERGED CODES

62	72	83
63	73	84
64	74	92

TEXAS INSTRUMENTS
INCORPORATED

Correction is made for air and walls conductivity using the forms $RAIR = \alpha / h_u$ and $RWALLS = \beta / V_{nu}$ then $h = (1 - RAIR)(1 - RWALLS) h_u$. α and β are determined from complete analyses ~~to~~ using the 1108. h_u , α , $R = 1/h$ are then output. The program then asks for a temperature. Upon receiving a temp. in $^{\circ}C$ it then normalizes R to $70^{\circ}F$. (continue P5)

USER INSTRUCTIONS				
STEP	PROCEDURE	ENTER	PRESS	DISPLAY
5	After last log is entered calculate T	—	RST R/S	T , Prints 5 lines.
6	Convert T to h	—	A, B, C or D	3771 Prints 5 lines.
7	Normalize R to $70^{\circ}F$ from temperature of run.	$T (^{\circ}C)$	R/S	4271 Prints 4 lines.
8	Normalize to VN from V	V (ft/sec) VN (ft/sec)	R/S R/S	423121 $P(VN)$
—	To use for linear regression			Prints 5 lines
1.	Initialize statistics	2 nd Pgm 01, SBR, CLR		0
2.	Enter data	x_i y_i	Σx Σy	i
3	Calculate parameters & Errors. slope, intercept and errors are printed.	—	RST R/S	T (may not be meaningful)
	output for B $T = 25$ entered			25. 1439.067091 1385.564058 .0007217277

USER DEFINED KEYS		DATA REGISTERS (INV L3)	
A	44.842129	A ₀	10 Unit—
B	-22.953427	A ₁	11 A
C	4.749417	A ₂	12 coffe—
D	-0.34797	A ₃	13 cents
E	7.0356748	σ	14
A'	.7143000251	β	15 Constant for Temp
B'	3771.	T	16 normalizing
C'	0.012	T_{nu}	17 constant
D'	1.84	$R(70)$	18 $R(VN)$
E'	3555100156.	$R(70)$	19 0.000855174
FLAGS	0	1	2
	3	4	5
	6	7	8
	9		

CODE	KEY	COMMENTS	LOC	CODE	KEY	COMMENTS	LOC	CODE	KEY	COMMENTS
160	91	R/S		215	00	00		270	32	32
161	76	LBL		216	04	4		271	01	1
162	12	B		217	06	6		272	03	3
163	32	XIT	Jhl B sets up the calculation of h for Ti	218	42	STD		273	42	STD
164	01	1		219	30	30		274	33	33
165	04	4		220	04	4		275	01	1
166	69	DP		221	07	7		276	04	4
167	04	04		222	42	STD		277	42	STD
168	32	XIT		223	31	31		278	34	34
169	69	DP		224	04	4		279	01	1
170	06	06		225	08	8		280	05	5
171	23	LNK		226	42	STD		281	42	STD
172	42	STD		227	32	32		282	35	35
173	00	00		228	04	4		283	71	SER
174	04	4		229	09	9		284	22	INV
175	00	0		230	42	STD		285	98	ADV
176	42	STD		231	33	33		286	98	ADV
177	30	30		232	03	3		287	91	R/S
178	04	4		233	06	6		288	76	LBL
179	01	1		234	42	STD		289	13	C
180	42	STD		235	34	34		290	32	XIT
181	31	31		236	03	3		291	01	1
182	04	4		237	07	7		292	05	5
183	02	2		238	42	STD		293	69	DP
184	42	STD		239	35	35		294	04	04
185	32	32		240	71	SER		295	32	XIT
186	04	4		241	22	INV		296	69	DP
187	03	3		242	98	ADV		297	06	06
188	42	STD		243	98	ADV		298	23	LNK
189	33	33		244	91	R/S		299	42	STD
190	04	4		245	00	0		300	00	00
191	04	4		246	76	LBL		301	02	2
192	42	STD		247	11	A	Jhl A sets up the calculation of h for AL	302	00	0
193	34	34		248	32	XIT		303	42	STD
194	04	4		249	01	1		304	30	30
195	05	5		250	03	3		305	02	2
196	42	STD		251	69	DP		306	01	1
197	35	35		252	04	04		307	42	STD
198	71	SER		253	32	XIT		308	31	31
199	22	INV		254	69	DP		309	02	2
200	98	ADV		255	06	06		310	02	2
201	98	ADV		256	23	LNK		311	42	STD
202	91	R/S		257	42	STD		312	32	32
203	76	LBL		258	00	00		313	02	2
204	14	D		259	01	1		314	03	3
205	32	XIT	Jhl D sets up the calculation of h for copper Rel	260	00	0		315	42	STD
206	01	1		261	42	STD		316	33	33
207	06	6		262	30	30		317	02	2
208	69	DP		263	01	1		318	04	4
209	04	04		264	01	1		319	42	STD
210	32	XIT		265	42	STD				
211	69	DP		266	31	31				
212	06	06		267	01	1				
213	23	LNK		268	02	2				
214	42	STD		269	42	STD				

MERGED CODES

62	DP	72	STD	83	STD
63	DP	73	STD	84	STD
64	DP	74	STD	92	INV

TEXAS INSTRUMENTS
INCORPORATED

GRAMMER

DATE

LOC	CODE	KEY	COMMENTS	LOC	CODE	KEY	COMMENTS	LOC	CODE	KEY	COMMENTS
320	34	34		375	73	RC*		430	91	R/S	
321	02	2		376	34	34		431	99	PRT	
322	05	5		377	55	÷		432	36	PGM	
323	42	STD		378	43	RCL		433	25	25	calculate
324	35	35		379	01	01		434	16	A'	by Ino
325	71	SBR		380	54	)		435	42	STD	to 70°F.
326	22	INV		381	65	x		436	00	00	operator
327	98	ADV		382	53	(		437	43	RCL	must
328	98	ADV		383	01	1		438	02	02	input
329	91	R/S		384	75	-		439	65	x	experiments
330	76	LBL		385	73	RC*		440	53	(	temperature
331	22	INV	Jbl Ino	386	35	35		441	01	1	
332	53	(		387	55	÷		442	85	+	
333	73	RC*	calculates	388	43	RCL		443	43	RCL	
334	30	30		389	01	01		444	17	17	
335	85	+	h given	390	34	FX		445	65	x	
336	43	RCL		391	54	)		446	43	RCL	
337	00	00	a T	392	65	x		447	00	00	
338	65	x		393	43	RCL		448	54	)	
339	73	RC*	and 4	394	01	01		449	55	÷	
340	31	31		395	95	=		450	43	RCL	
341	85	+	coefficients	396	32	XIT		451	18	18	
342	73	RC*		397	25	CLR		452	95	=	
343	32	32	it then	398	02	2		453	32	XIT	
344	65	x	makes	399	03	3		454	43	RCL	
345	43	RCL		400	69	DP		455	19	19	
346	00	00	Air and	401	04	04		456	69	DP	
347	33	X²	walls	402	32	XIT		457	01	01	
348	85	+		403	69	DP		458	69	DP	
349	73	RC*	corrections	404	06	06		459	05	05	
350	33	33		405	35	1/X		460	32	XIT	
351	65	x	based on	406	32	XIT		461	99	PRT	
352	43	RCL	approximate	407	02	2		462	42	STD	
353	00	00		408	06	6		463	03	03	
354	45	YX	forms of	409	03	3		464	76	LBL	Jbl. CE
355	03	3	the	410	02	2.		465	24	CE	normalization
356	54	)	correction	411	03	3		466	98	ADV	has as
357	22	INV	for molas.	412	69	DP		467	43	RCL	calculate
358	23	LNx		413	04	04		468	26	26	by P In)
359	42	STD		414	32	XIT		469	69	DP	to V/N
360	01	01		415	69	DP		470	00	00	from V.
361	32	XIT		416	06	06		471	69	DP	
362	25	CLR		417	42	STD		472	01	01	
363	02	2		418	02	02		473	69	DP	
364	03	3		419	76	LBL		474	05	05	
365	04	4		420	23	LNx	Jbl ln x	475	91	R/S	
366	01	1		421	98	ADV	normalized	476	99	PRT	
367	69	DP		422	69	DP	h as	477	42	STD	
368	04	04		423	00	00		478	04	04	
369	32	XIT		424	43	RCL		479	43	RCL	
370	69	DP		425	16	16					
371	06	06		426	69	DP					
372	53	(		427	01	01					
373	01	1		428	69	DP					
374	75	-		429	05	05					

MERGED CODES

62	63	64	72	73	74	83	84	92
STO	RC	SUM	STO	RC	SUM	STO	RC	INV

TEXAS INSTRUMENTS
INCORPORATED

Version 1.1 (cont.)

PROGRAM DESCRIPTION

B1.7

RST, A/S gives linear regression slope and intercept with errors for data entered with ϵ^+ . A-D converts T to h for any of 4 different sets of ~~from~~ coefficients. GTO Lnx, A/S normalizes any P stored in loc. 02 to $70^\circ E$, and GTO CE, A/S normalizes any P stored in loc 03 to VN from V.

USER INSTRUCTIONS

STEP	PROCEDURE	ENTER	PRESS	DISPLAY
	<u>Note</u>			
	The form of the temperature and velocity correction depends on an empirical expression relating h , T and V . Several forms of this expression have been proposed. Writing h in English units, V in ft/sec and T in $^{\circ}\text{F}$ have h and D in inches, have h			
①	$h = 160 V^{.8} (1 + .012 T) / D^{.2}$			
②	$h = 150.4 V^{.8} (1 + .012 T) / D^{.2}$			
③	$h = 133.2 V^{.8} (1 + .0113 T) / D^{.2}$			
④	$h = 115.6 V^{.8} (1 + .0137 T) / D^{.2}$			
	Continue P 8	46. Storage	30	
		47. for	31	
		48. indirect	32	
		49. recall	33	
		36.	34	
		37. ↓	35	
	7.0356748 α unit		36	
	.7143000251 e_D		37	
	371341. TAU		38	
		constants		Partial
				output for D
				$T = 4$ entered

USER DEFINED KEYS		50. Temperature		43.			
A		13.996615	A ₀	40	1204.766786	HU	
B		-3.297064	A ₁	41	1173.082724	H	
C		0.492248	A ₂	42	.0008524548	1/H	
D		-0.036349	A ₃	43			
E		5.51	α	44			
F		1.27	β	45			
G		44.842129	A ₀	46			
H		-22.953427	A ₁	47			
I		4.749417	A ₂	48			
J		-0.34797	A ₃	49			
K							
L							
M							
N							
O							
P							
Q							
R							
S							
T							
U							
V							
W							
X							
Y							
Z							
[							
{							
~							
.							
/							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"							
#							
\$							
%							
&							
'							
"</

PROGRAMMER

DATE

Coding Form

CODE	KEY	COMMENTS	LOC	CODE	KEY	COMMENTS	LOC	CODE	KEY	COMMENTS
480	27	27				The source and applicability				
481	69	DP				of these expressions is as follows.				
482	01	01				① Standard expression for fresh				
483	69	DP				water from McAdams Heat				
484	05	05				Transmission 2.19 (1954)				
485	91	R/S				② ① modified for seawater				
486	99	PRT				properties, no modification to				
487	42	STD				Temperature dependence.				
488	05	05				③ Rederivation of ① from forced water from				
489	53	(				first principles, using temperature				
490	43	RCL				dependence of sea water properties				
491	05	05				done by Mariat CNU.				
492	45	YX				④ Rederivation of ① forced water				
493	43	RCL				done by Helen Kritzsch at				
494	28	28				NSRDC.				
495	75	-				The constants stored in				
496	43	RCL				locations 17, 18, and 29 are				
497	04	04				dependent on these expressions.				
498	45	YX				17 is the temperature dependence				
499	43	RCL				constant multiplying T, 18				
500	28	28				is the value of the expression				
501	54	)				in parenthesis and 29 is				
502	65	X				the slope of the 1/h vs V ^{1/2}				
503	43	RCL				curve evaluated at 70°F. 9m				
504	29	29				the case of ① the value of this				
505	95	=				constant is given by				
506	85	+				1.049 $1.049^2 / ((1 + 0.012 * 70) * 160)$				
507	43	RCL				= .00343.				
508	03	03				The values of these constants				
509	95	=				(evaluated by D = 1.049) are				
510	42	STD				given below				
511	00	00				Express for 17				
512	43	RCL				Temp Unit				
513	09	09				① 0.012				
514	69	DP				② 0.012				
515	01	01				③ 0.0113				
516	69	DP				④ 0.0137				
517	05	05								
518	43	RCL								
519	00	00								
520	99	PRT								
521	92	RTN								

CNU recommends the use of ① for fresh water and ③ for sea water, but use of the 2 different expressions will make a difference of only 0.004 at $R_f = .00050$ for the temperature and velocity variations observed during

MERGED CODES

62	STO	83	CTO
63	RCL	84	INV
64	SUM	92	INV (SBR)

TEXAS INSTRUMENTS
INCORPORATED

APPENDIX B2

Version 1.2 of the TI-59 analysis program

001 12 12
 002 42 STD
 003 07 07
 004 32 XIT
 005 42 STD
 006 08 08
 007 43 RCL
 008 05 05
 009 65 X
 010 43 RCL
 011 03 03
 012 75 -
 013 43 RCL
 014 04 04
 015 33 X²
 016 95 =
 017 42 STD
 018 39 39
 019 53 (
 020 43 RCL
 021 02 02
 022 85 +
 023 43 RCL
 024 07 07
 025 33 X²
 026 65 X
 027 43 RCL
 028 03 03
 029 85 +
 030 43 RCL
 031 05 05
 032 65 X
 033 43 RCL
 034 08 08
 035 33 X²
 036 75 -
 037 43 RCL
 038 01 01
 039 65 X
 040 43 RCL
 041 07 07
 042 65 X
 043 02 2
 044 75 -
 045 43 RCL
 046 06 06
 047 65 X
 048 43 RCL
 049 08 08
 050 65 X
 051 02 2
 052 85 +
 053 43 RCL
 054 04 04
 055 65 X
 056 43 RCL
 057 07 07
 058 65 X
 059 43 RCL
 060 08 08

Version 1.2
 Page 52

063 54)
 064 55 ÷
 065 53 (
 066 43 RCL
 067 03 03
 068 75 -
 069 02 2
 070 54)
 071 95 =
 072 42 STD
 073 30 30
 074 65 X
 075 43 RCL
 076 05 05
 077 55 ÷
 078 43 RCL
 079 39 39
 080 95 =
 081 34 FX
 082 42 STD
 083 31 31
 084 32 XIT
 085 43 RCL
 086 30 30
 087 65 X
 088 43 RCL
 089 03 03
 090 55 ÷
 091 43 RCL
 092 39 39
 093 95 =
 094 34 FX
 095 42 STD
 096 32 32
 097 03 3
 098 06 6
 099 02 2
 100 07 7
 101 03 3
 102 03 3
 103 69 DP
 104 04 04
 105 43 RCL
 106 08 08
 107 69 DP
 108 06 06
 109 04 4
 110 07 7
 111 06 6
 112 03 3
 113 02 2
 114 00 0
 115 69 DP
 116 04 04
 117 42 STD
 118 33 33
 119 43 RCL
 120 32 32

121 06 06
 122 02 2
 123 04 4
 124 04 4
 125 03 3
 126 01 1
 127 03 3
 128 07 7
 129 69 DP
 130 04 04
 131 43 RCL
 132 07 07
 133 69 DP
 134 06 06
 135 43 RCL
 136 33 33
 137 69 DP
 138 04 04
 139 43 RCL
 140 31 31
 141 69 DP
 142 06 06
 143 43 RCL
 144 08 08
 145 35 1/X
 146 65 X
 147 02 2
 148 94 +/-
 149 95 =
 150 32 XIT
 151 43 RCL
 152 38 38
 153 69 DP
 154 04 04
 155 98 ADV
 156 32 XIT
 157 69 DP
 158 06 06
 159 98 ADV
 160 91 R/S
 161 76 LBL
 162 12 B
 163 32 XIT
 164 03 3
 165 07 7
 166 02 2
 167 04 4
 168 69 DP
 169 04 04
 170 32 XIT
 171 69 DP
 172 06 06
 173 23 LNX
 174 42 STD
 175 00 00
 176 04 4
 177 00 0
 178 42 STD
 179 30 30
 180 04 4

B2.1.

VERSION
1.2
(cont.)

182	42	STD	242	03	3	302	06	6
183	31	31	243	07	7	303	69	DP B2.2
184	04	4	244	42	STD	304	04	04
185	02	2	245	35	35	305	32	X:IT
186	42	STD	246	71	SBR	306	69	DP
187	32	32	247	22	INV	307	06	06
188	04	4	248	98	ADV	308	23	LNK
189	03	3	249	98	ADV	309	42	STD
190	42	STD	250	91	R/S	310	00	00
191	33	33	251	00	0	311	02	2
192	04	4	252	76	LBL	312	00	0
193	04	4	253	11	A	313	42	STD
194	42	STD	254	32	X:IT	314	30	30
195	34	34	255	01	1	315	02	2
196	04	4	256	03	3	316	01	1
197	05	5	257	02	2	317	42	STD
198	42	STD	258	07	7	318	31	31
199	35	35	259	69	DP	319	02	2
200	71	SBR	260	04	04	320	02	2
201	22	INV	261	32	X:IT	321	42	STD
202	98	ADV	262	69	DP	322	32	32
203	98	ADV	263	06	06	323	02	2
204	91	R/S	264	23	LNK	324	03	3
205	76	LBL	265	42	STD	325	42	STD
206	14	D	266	00	00	326	33	33
207	32	X:IT	267	01	1	327	02	2
208	01	1	268	00	0	328	04	4
209	05	5	269	42	STD	329	42	STD
210	06	6	270	30	30	330	34	34
211	03	3	271	01	1	331	02	2
212	03	3	272	01	1	332	05	5
213	01	1	273	42	STD	333	42	STD
214	69	DP	274	31	31	334	35	35
215	04	04	275	01	1	335	71	SBR
216	32	X:IT	276	02	2	336	22	INV
217	69	DP	277	42	STD	337	98	ADV
218	06	06	278	32	32	338	98	ADV
219	23	LNK	279	01	1	339	91	R/S
220	42	STD	280	03	3	340	76	LBL
221	00	00	281	42	STD	341	22	INV
222	04	4	282	33	33	342	53	(
223	06	6	283	01	1	343	73	RC*
224	42	STD	284	04	4	344	30	30
225	30	30	285	42	STD	345	85	+
226	04	4	286	34	34	346	43	RCL
227	07	7	287	01	1	347	00	00
228	42	STD	288	05	5	348	65	x
229	31	31	289	42	STD	349	73	RC*
230	04	4	290	35	35	350	31	31
231	08	8	291	71	SBR	351	85	+
232	42	STD	292	22	INV	352	73	RC*
233	32	32	293	98	ADV	353	32	32
234	04	4	294	98	ADV	354	65	x
235	09	9	295	91	R/S	355	43	RCL
236	42	STD	296	76	LBL	356	00	00
237	33	33	297	13	C	357	33	X2
238	03	3	298	32	X:IT	358	85	+
239	06	6	299	03	3	359	73	RC*
240	42	STD	300	06	6	360	33	33

VERSION

1.2
(cont.)

363 00 00
364 45 YX
365 03 3
366 54)
367 22 INV
368 23 LNX
369 42 STD
370 01 01
371 32 XIT
372 25 CLR
373 02 2
374 03 3
375 04 4
376 01 1
377 69 DP
378 04 04
379 32 XIT
380 69 DP
381 06 06
382 53 (
383 01 1
384 75 -
385 73 RC#
386 34 34
387 55 +
388 43 RCL
389 01 01
390 54)
391 65 x
392 53 (
393 01 1
394 75 -
395 73 RC#
396 35 35
397 55 +
398 43 RCL
399 01 01
400 34 IX
401 54)
402 65 x
403 43 RCL
404 01 01
405 95 =
406 32 XIT
407 25 CLR
408 02 2
409 03 3
410 69 DP
411 04 04
412 32 XIT
413 69 DP
414 06 06
415 35 1/X
416 32 XIT
417 02 2
418 06 6
419 03 3
420 02 2

421 04 04
422 32 XIT
423 69 DP
424 06 06
425 42 STD
426 02 02
427 76 LBL
428 23 LNX
429 98 ADV
430 69 DP
431 00 00
432 43 RCL
433 16 16
434 69 DP
435 01 01
436 69 DP
437 05 05
438 91 R/S
439 99 PRT
440 36 PGM
441 25 25
442 16 A'
443 42 STD
444 00 00
445 43 RCL
446 02 02
447 65 x
448 53 (
449 01 1
450 85 +
451 43 RCL
452 17 17
453 65 x
454 43 RCL
455 00 00
456 54)
457 55 +
458 43 RCL
459 18 18
460 95 =
461 32 XIT
462 43 RCL
463 19 19
464 69 DP
465 01 01
466 69 DP
467 05 05
468 32 XIT
469 99 PRT
470 42 STD
471 03 03
472 76 LBL
473 24 CE
474 98 ADV
475 43 RCL
476 26 26
477 69 DP
478 00 00
479 00 00

480 01 01
481 69 DP
482 05 05
483 91 R/S
484 99 PRT
485 42 STD
486 04 04
487 43 RCL
488 27 27
489 69 DP
490 01 01
491 69 DP
492 05 05
493 91 R/S
494 99 PRT
495 42 STD
496 05 05
497 53 (
498 43 RCL
499 05 05
500 45 YX
501 43 RCL
502 28 28
503 75 -
504 43 RCL
505 04 04
506 45 YX
507 43 RCL
508 28 28
509 54)
510 65 x
511 43 RCL
512 29 29
513 95 =
514 85 +
515 43 RCL
516 03 03
517 95 =
518 42 STD
519 00 00
520 43 RCL
521 09 09
522 69 DP
523 01 01
524 69 DP
525 05 05
526 43 RCL
527 00 00
528 99 PRT
529 92 RTN

B2.3

.0012861804	00
48.4679	01
391.5655508	02
6.	03
30.	04
220.	05
240.6114	06
8.201419048	07
-.0246871429	08
3555423156.	09
623865 16.42955	10
61494 5.45687	11
03788 1.12304	12
00913 0.097956	13
4.52	14
1.3599	15
3771.	16
0.012	17
1.84	18
3555100156.	19
69.8696	20
-35.73837	21
6.82736	22
-0.450269	23
4.52	24
0.58589	25
4271.	26
423171.	27
-0.8	28
0.003632	29
.00000001293	30
.0002602295	31
.0000429755	32
476320.	33
36.	34
37.	35
4.44	36
0.9587	37
371341.	38
420.	39
36.58555	40
-16.56452	41
3.11874	42
-0.210451	43
4.48	44
0.7245	45
21.23157	46
-7.4627	47
1.29994	48
-0.088456	49

A1

SS

T1

C/N

VERSION 1.2 (cont.)

162	12	B
206	14	D
253	11	A
297	13	C
341	22	INV
430	23	LNK
475	24	CE

B2.4	0.	CLR
	0.	
	0.9	Σ+
	1.	
	0.8	Σ+
	2.	
	0.7	Σ+
	3.	
	0.601	Σ+
	4.	
	4.	RST
		OP
		12
	0.8998	
	0.8998	STD
	-0.0997	SLP
	.0001732051	+/-
	0.8998	INT
	0.000324037	+/-
	20.06018054	TAU
	20.06018054	AL
	1850.499181	HU
	1787.622669	H
	.0005594022	1/H
	T?	
	25.	
	R(70)	
	.0005849401	
	V?	
	7.	
	VN?	
	6.	
	R(VN)	
	.0006854379	
	45.	TI
	1229.00509	HU
	1199.218756	H
	.0008338762	1/H
	T?	
	60.	SS
	1201.31812	HU
	1176.567557	H
	.0008499299	1/H
	T?	
	50.	C/N
	761.8120198	HU
	731.0652167	H
	0.001362867	1/H

APPENDIX B3

Version 2.0 of the TI-59 analysis program

PROGRAM DESCRIPTION

Modification of LN ($V - V_o$) $\rightarrow R$ to enable the entry of voltages rather than zero corrected logs.

Version 2.0 of data analysis program.

USER INSTRUCTIONS

STEP	PROCEDURE	ENTER	PRESS	DISPLAY
1.	Follow Steps 1 & 2 for previous program			
2.	Modify Calculation of τ for new time interval (Z) between points (Put Z in step 147)	Z	GTO 147 L&N	147 02
3.	Create space for new program Insert 11 steps	—	RST L&N	000 69
4.	Key in program steps 0-10 as shown on P. 2	—	2 nd L&N	Repeat 11 times
5.	Exit from L&N	—	L&N	000 00
6.	Determine V_o using Σ^+ or other means.			
7.	Enter V_o	V_o	STO 00	V_o
8.	Initialize statistics	—	2 nd Pgm 01	SBR Clr 0
9.	Reset calculator (very important or Calc goes to FbLE of Pgm 01)		RST	0
10.	Enter voltages	V_i	E	i
11.	Evaluate τ and continue program	—	2 nd E	$\tau +$
12.	Recall Program is desired	1, 2, 3	2 nd Unit	print

USER DEFINED KEYS	DATA REGISTERS (INV L&R)	LABELS (Op 08)
A Must record 3 acids	0 To	INV — Inv — CE — CLR — x:1 — x:2 —
B get all the revised program	1	√ — 1/x — STO — RCL — SUM — 7 th —
C onto card 2.	2	EE — (—) — ÷ — GTO — X —
D	3	SBR — — — RST — + — R/S — —
E	4	0/ — = — CLR — ON — — — 0 —
A'	5	CF — P — M — W — — — —
B'	6	LC — M — CI — LC — — — —
C'	7	7 th — 2 nd — 2 nd — — — —
D'	8	III — — — — — — —
E'	9	IN — — — — — — —
FLAGS	0	W — — — — — — —
	1	
	2	
	3	
	4	
	5	
	6	
	7	
	8	
	9	

LOC	CODE	KEY	COMMENTS	LOC	CODE	KEY	COMMENTS	LOC	CODE	KEY	COMMENTS
76	2 nd	HL									
1	1	E						1			
2								2			
43	RCL							3			
00	00							4			
95	=							5			
23	2 nd	X		1	After Step 2			6			
78	2 nd	Σ+		2	14.5 - 14.8 should						
91	R/S			3	look like this			6			
76	2 nd	HL		4				9			
10	2 nd	E'		14	635 1/x			0			
69	2 nd	OP	OK	14	65 X			1			
12	12		program	10	0.2 0.2			2			
0			continues	14	94 +1-			3			
			from					4			
			here.					5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2				7			
				3				8			
				4				9			
				5				0			
				6				1			
				7				2			
				8				3			
				9				4			
				0				5			
				1				6			
				2							

APPENDIX B4

Approximate calculation of R_{walls}

Computation of R_{wells} for each tube material :

$$R_w = \frac{2}{L} \sqrt{\frac{R_a t'}{h}} = \frac{2}{L} \sqrt{R_a t'} \cdot \frac{1}{\sqrt{h}} = \beta \frac{1}{\sqrt{h}}$$

Al 5052 :

$$R_a = 79.83 \text{ EU}$$

$$t' = (0.0695/12)'$$

$$L = 1'$$

$$\beta = 1.3599$$

Ti grade :

$$R_a = 12.7 \text{ EU}$$

$$t' = (0.124/12)'$$

$$L = 1'$$

$$\beta = 0.7245$$

AL-6X SS :

$$R_a = 7.983 \text{ EU}$$

$$t' = (0.129/12)'$$

$$L = 1'$$

$$\beta = 0.58589$$

Cu-Ni 90/10 :

$$R_a = 22.98 \text{ EU}$$

$$t' = (0.120/12)'$$

$$L = 1'$$

$$\beta = 0.9587$$

For say $h \approx 1000 \text{ EU}$, the % wall corrections are :

Al-5052 ~ 4.3

Ti ~ 2.3

SS AL 6X ~ 1.9

Cu-Ni 90/10 ~ 3.0

APPENDIX B5

Approximate calculation of R_{air}

Approximate computation of R_{air} for each heat exchanger pipe material.

$$R_a = h_a \frac{r_3}{r_1} \left[\frac{1}{h_{block}} + \frac{1}{h_{pipe}} + \frac{r_1}{r_2} \frac{1}{h_i} + \frac{1}{h} \right]$$

say $1/h_i \approx 0$

further in the 1st approximation

$$1/h_{block} \ll 1/h$$

$$1/h_{pipe} \ll 1/h$$

$$\therefore R_{air} \approx h_a (r_3/r_1) 1/h = \alpha/h$$

calculate α

$$h_a = 1.57 \text{ English units}$$

Material	r_3/r_1	α	R_a for $h = 1000$ EU
Al 5052	(1.5" / .5215")	4.52	0.45
Ti	(1.5" / .526")	4.46	0.45
SS AL-6X	(1.5" / .5215")	4.52	0.45
Cu-Ni 90/10	(1.5" / .53")	4.44	0.44