NPS ARCHIVE
1969
CASTER, W.

NEAR-BOTTOM CURRENTS IN MONTEREY
SUBMARINE CANYON AND ON THE AD-
JACENT SHELF

by

William Albert Caster

XVI

United States Naval Postgraduate School

THESIS

NEAR-BOTTOM CURRENTS IN MONTEREY SUBMARINE
CANYON AND ON THE ADJACENT SHELF

by

William Albert Caster

October 1969

*This document has been approved for public re-
lease and sale; its distribution is unlimited.*

LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIF. 93940

Near-Bottom Currents in Monterey Submarine Canyon
and on the Adjacent Shelf

by

William Albert Caster
Lieutenant, United States Coast Guard
B.S., United States Coast Guard Academy, 1963

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OCEANOGRAPHY

from the

NAVAL POSTGRADUATE SCHOOL
October 1969

SPS ARCHIVE
9169
CASTER, W.

ABSTRACT

Near-bottom currents in Monterey Submarine Canyon and the adjacent shelf were collected using Savonius current meters. Simultaneous measurements were made with one current meter on the shelf at a depth of 91 meters and one meter located in the Canyon at 366 meters. Another record was taken in the Canyon at the 366-meter location. Current speed, current direction and water temperature were recorded continuously for approximately seven days in each record. First-order statistics were calculated and plotted for these time-series data. Scatter diagrams, progressive vector diagrams and power spectra were also computed and analyzed for the records collected during this study and for records of sufficient length from previous investigations.

Net current set was in a cross-canyon direction for many of the records; however, the currents in the Canyon oscillated as reported in previous investigations. The oscillations were not as evident on the shelf record. Mean and maximum current speeds recorded in the Canyon were 10 and 51 cm/sec, respectively. On the shelf these values were 7 and 25 cm/sec. Observed values of net current set and volume transport are related to Monterey Bay seasonal water conditions.

TABLE OF CONTENTS

I.	INTRODUCTION -----	11
A.	SUBMARINE CANYONS -----	11
B.	PREVIOUS INVESTIGATIONS OF NEAR-BOTTOM CURRENTS IN SUBMARINE CANYONS -----	12
C.	PURPOSE OF THIS INVESTIGATION -----	15
II.	REGION OF OBSERVATION -----	16
A.	MONTEREY BAY -----	16
B.	MONTEREY SUBMARINE CANYON -----	16
III.	MEASURING EQUIPMENT AND PROCEDURES -----	19
A.	CURRENT MEASURING SYSTEM -----	19
B.	RELEASE MECHANISM -----	21
C.	TAUT-LINE MOORING AND MOORING PROCEDURES -----	21
D.	TIDE RECORDING EQUIPMENT -----	23
IV.	CONTINUOUS TIME-SERIES CURRENT OBSERVATIONS -----	24
V.	DATA ANALYSIS -----	28
A.	DATA SAMPLING AND REDUCTION OF CURRENT RECORDS PROCEDURES -----	28
B.	SAMPLING PROCEDURE AND DATA REDUCTION OF TIDAL RECORDS -----	28
C.	ELEMENTARY STATISTICS -----	28
D.	POWER SPECTRA ANALYSIS -----	29
VI.	DISCUSSION -----	31
A.	GRAPHIC PLOTS AND STATISTICAL ANALYSIS -----	31
B.	PROGRESSIVE VECTOR DIAGRAM -----	40
C.	SPECTRAL ANALYSIS -----	52
VII.	CONCLUSIONS AND RECOMMENDATIONS -----	60
A.	CONCLUSIONS -----	60
B.	RECOMMENDATIONS -----	61

APPENDIX A	DIFFICULTIES WITH TAUT-LINE MOORING -----	63
APPENDIX B	GRAPHIC PLOTS OF TIME-SERIES DATA -----	66
APPENDIX C	SCATTER DIAGRAMS OF DOOLEY'S AND NJUS' DATA -----	94
APPENDIX D	PROGRESSIVE VECTOR DIAGRAMS OF DOOLEY'S AND NJUS' DATA -----	101
APPENDIX E	SPECTRAL ESTIMATES OF DOOLEY'S AND NJUS' DATA -----	108
APPENDIX F	PROGRAM CONVERT -----	115
APPENDIX G	PROGRAM TIDE -----	118
APPENDIX H	PROGRAM CURRENT -----	120
APPENDIX I	PROGRAM COMPOSITE DRAW -----	128
APPENDIX J	PROGRAM TEMPERATURE DRAW -----	130
APPENDIX K	UNIVERSITY OF BRITISH COLUMBIA PROGRAMS: FTOR, SCOR AND FCPLT -----	132
APPENDIX L	PROGRAM VECTOR DRAW -----	195
LITERATURE CITED	-----	198
INITIAL DISTRIBUTION LIST	-----	200
FORM DD 1473	-----	203

LIST OF TABLES

I.	OBSERVATION PROGRAM -----	26
II.	SUMMARY OF CURRENT STATISTICS -----	39
III.	SUMMARY OF RESULTS OF PROGRAM VECTOR DRAW -----	51
IV.	SUMMARY OF POWER SPECTRUM PEAKS -----	58
V.	SUMMARY OF SLOPE OF SPECTRUM DATA -----	59

LIST OF FIGURES

1.	Monterey Bay -----	17
2.	Monterey Submarine Canyon -----	18
3.	Strip Chart -----	20
4.	Taut-Line Mooring -----	22
5.	Mooring Sites -----	25
6.	Histograms - 1148 11 April to 1503 18 April 1969 -----	33
7.	Histograms - 1215 11 April to 1149 18 April 1969 -----	34
8.	Histograms - 1118 25 April to 0714 26 April 1969 -----	35
9.	Histograms - 1038 26 April to 1357 26 April 1969 -----	36
10.	Histograms - 1533 26 April to 0052 27 April 1969 -----	37
11.	Histograms - 0148 27 April to 0033 3 May 1969 -----	38
12.	Scatter Diagram - 1148 11 April to 1503 18 April 1969 -----	42
13.	Scatter Diagram - 1215 11 April to 1149 18 April 1969 -----	43
14.	Scatter Diagram - 0148 27 April to 0033 3 May 1969 -----	44
15.	Progressive Vector Diagram - 1148 11 April to 1503 18 April 1969 -----	46
16.	Progressive Vector Diagram - 1215 11 April to 1149 18 April 1969 -----	47
17.	Progressive Vector Diagram - Simultaneous 91- and 366- Meter Arrays -----	48
18.	Progressive Vector Diagram - 0148 27 April to 0033 3 May 1969 -----	49
19.	Volume Transport and Current Set vs. Months -----	50
20.	Spectral Analysis - 1148 11 April to 1503 18 April 1969 ----	53
21.	Spectral Analysis - 1215 11 April to 1149 18 April 1969 ----	54
22.	Spectral Analysis - 0148 27 April to 0033 3 May 1969 -----	55
23.	Hours vs. Log ₁₀ Frequency -----	56
24.	Composite Draw - 1148 11 April to 0848 13 April 1969 -----	66

25.	Composite Draw - 0848 13 April to 0548 15 April 1969 -----	68
26.	Composite Draw - 0548 15 April to 0248 17 April 1969 -----	70
27.	Composite Draw - 0248 17 April to 1503 18 April 1969 -----	72
28.	Composite Draw - 1215 11 April to 0915 13 April 1969 -----	74
29.	Composite Draw - 0915 13 April to 0615 15 April 1969 -----	76
30.	Composite Draw - 0615 15 April to 0315 17 April 1969 -----	78
31.	Composite Draw - 0315 17 April to 1149 18 April 1969 -----	80
32.	Composite Draw - 1118 25 April to 0714 26 April and 1038 26 April to 1357 26 April 1969 -----	82
33.	Composite Draw - 1533 26 April to 0052 27 April 1969 -----	84
34.	Composite Draw - 0148 27 April to 2248 28 April 1969 -----	86
35.	Composite Draw - 2248 28 April to 1948 30 April 1969 -----	88
36.	Composite Draw - 1948 30 April to 1648 2 May 1969 -----	90
37.	Composite Draw - 1648 2 May to 0033 3 May 1969 -----	92
38.	Scatter Diagram - 1000 22 March to 1340 24 March 1967 -----	94
39.	Scatter Diagram - 1110 24 Nov to 2210 26 Nov 1967 -----	95
40.	Scatter Diagram - 1426 15 January to 2326 21 January 1968 ---	96
41.	Scatter Diagram - 0945 29 February to 1410 2 March 1968 -----	97
42.	Scatter Diagram - 1520 14 May to 1526 21 May 1968 -----	98
43.	Scatter Diagram - 1230 26 September to 1250 3 October 1968 --	99
44.	Scatter Diagram - 1050 17 October to 0950 24 October 1968 ---	100
45.	Progressive Vector Diagram - 1000 22 March to 1340 24 March 1967 -----	100
46.	Progressive Vector Diagram - 1110 24 November to 2210 26 November 1967 -----	102
47.	Progressive Vector Diagram - 1426 15 January to 2326 21 January 1968 -----	103
48.	Progressive Vector Diagram - 0945 29 February to 1410 2 March 1968 -----	104

49.	Progressive Vector Diagram - 1520 14 May to 1526 21 May 1968 -----	105
50.	Progressive Vector Diagram - 1230 26 September to 1250 3 October 1968 -----	106
51.	Progressive Vector Diagram - 1050 17 October to 0950 24 October 1968 -----	107
52.	Spectral Analysis - 1000 22 March to 1340 24 March 1967 -----	108
53.	Spectral Analysis - 1110 24 November to 2210 26 Nov 1967 -----	109
54.	Spectral Analysis - 1426 15 January to 2326 21 January 1968 -	110
55.	Spectral Analysis - 0945 29 February to 1410 2 March 1968 ---	111
56.	Spectral Analysis - 1520 14 May to 1526 21 May 1968 -----	112
57.	Spectral Analysis - 1230 26 September to 1250 3 October 1968 -----	113
58.	Spectral Analysis - 1050 17 October to 0950 24 October 1968 -----	114

ACKNOWLEDGEMENTS

The author expresses his sincere appreciation to Professor Warren W. Denner, Department of Oceanography, Naval Postgraduate School, for his advice during the course of this investigation. Thanks are due the oceanographic support personnel who assisted in this study, especially Mr. Jack Mellor; the staffs of the Computer Facility and the Machine Facility, Naval Postgraduate School, and the crew of the School's research vessel. The assistance of Mr. J. R. Wilson of the Institute of Oceanography of the University of British Columbia in using the power spectrum program is appreciated. A special thanks is due my wife for her assistance in reducing the data from the strip-chart records.

I. INTRODUCTION

A. SUBMARINE CANYONS

Submarine canyons were first studied toward the end of the nineteenth century; however, detailed studies were not made until the 1930s [Kuenen, 1960]. Veatch and Smith [1939] compiled a set of detailed contour charts of the submarine canyons of the east coast of the United States and the Congo Submarine Canyon. Work on the canyons of the west coast of the United States was begun by Francis Shepard during this same period. The report by Shepard and Emery [1941] on California canyons is considered to be one of the most important papers in the field [Martin, 1964].

The formation of canyons has been a much debated subject. Kuenen [1960] suggested several possible mechanisms for canyon formation including: (1) diastrophic origin, (2) warping of the continental borderland, (3) artesian spring sapping, (4) mudflows and landslides, (5) tsunamis, (6) hydraulic and tidal currents and (7) effects due to the Ice Age related to excessive lowering of sea level and turbidity currents.

Shepard [1963] suggests that no one process alone can explain the formation of submarine canyons. He feels that some were caused by diastrophism, turbidity currents and stream erosion from the lowering of sea level; however, he concludes that more study is required.

Submarine canyons are a unique feature of the continental margins and play an important part in local ocean processes such as sediment transport, circulation, oscillations, biological activity and weather

conditions. Current measurements thus have an important part in the complete understanding of the processes occurring in a canyon.

B. PREVIOUS INVESTIGATORS OF CURRENTS IN SUBMARINE CANYONS

Early investigators of near-bottom currents in submarine canyons used Ekman type current meters. Stetson [1937] measured currents in three canyons on Georges Bank and found the maximum currents to be 11.8 cm/sec during flood tide and 10.37 cm/sec during the ebb tide. These canyon currents were higher than the shelf currents in the same area. He felt that the currents could preserve the canyons but not form them. Shepard, Revelle and Dietz [1939] found maximum current speeds of 26.9 cm/sec in Monterey Submarine Canyon and speeds in excess of 20 cm/sec in canyons off Southern California. These measurements were taken approximately one meter off the bottom at depths of 42 to 840 meters. Although Stetson felt the currents were strongly related to the tide, Shepard, et al, [1964] using a diving saucer estimated down-canyon currents of 10 cm/sec and suggested that the currents might be due to internal waves, surf beat and seaward return of water carried inshore by swell.

Gatje and Pizinger [1965] made 75 observations using an Ekman current meter on nine different days in February and March 1965 in the Monterey Submarine Canyon. The current meter was placed 4.9 meters above the bottom in 132 meters of water. The most frequently observed speed was 22.7 cm/sec. There were eight readings higher than 25.7 cm/sec with the highest being 41.8 cm/sec and the second highest 30.9 cm/sec. The median speed was 10.3 cm/sec. Current direction showed a bimodal distribution with the modes separated by approximately 180°. By comparing current direction with the Canyon axis, they found

that the flow was predominately up- and down-canyon. Current direction reversal coincided with high and low tide but the flow was out of phase with the tide, often by 180° ; that is, down-canyon flow was associated with rising tide and up-canyon flow with falling tide.

Dooley [1968] continued, in Monterey Submarine Canyon, investigation of near-bottom currents using Savonius current meters recording speed, direction and water temperature. Observations were made in March, April and November 1967 and January, February, April and May 1968. Measurements were taken at depths of 118 to 165 meters with the current meters 16 meters above the bottom.

Current speeds averaged from 5.7 to 15.8 cm/sec and ranged from zero to 51.5 cm/sec. Direction of flow seemed to be predominately along the axis of the Canyon; however, other current directions were observed. Direction showed a bimodal tendency for all records except November's which was unimodal.

Spectral estimates of speed and temperature showed a prominent peak at 12 hours. Speed also had peaks from four to six hours and two to three hours. The direction reversed frequently at 12-hour intervals. The consistent strong energy peak at 12 hours suggested an association with the semidiurnal tidal component; however, no clear relationship between tidal movement and Canyon current was seen by Dooley from correlation of the Canyon currents and the tide marigram of Monterey Harbor.

Interestingly, the currents exhibited rapid bursts of speed, sometimes increasing 41.2 cm/sec in 1.5 minutes. The maximum bursts would take only three minutes to develop and slowly decrease after 20 to 30 minutes. On several occasions sustained speeds of 41.2

cm/sec were seen to last one hour. Dooley found that an increase in current speed, in most cases, accompanied a decrease in temperature.

Njus [1968] continued the study of near-bottom currents in the Monterey Submarine Canyon. He made three moorings during September and October 1968, each 12.2 meters above the bottom and at depths of 150, 165 and 201 meters. He found periodic reversal of current direction which was generally along the Canyon axis. Direction reversals occurred primarily at high and low tide. The up-canyon flow of cold water was associated with a falling tide. Speed increased from zero to approximately 51.5 cm/sec and then decreased. On the rising tide, the direction reversed and warm water flowed down-canyon increasing from zero to a maximum and then decreasing to a minimum to start the cycle again. Average current speeds for his good records obtained at 165 and 201 meters were 25.7 and 34.5 cm/sec. Power spectrum of temperature and speed showed peaks at 12.5, six and four hours, and the direction of the flow exhibited several reversals at a tidal frequency. There was good correlation between current speed and direction and water temperature at tidal periods.

Squire [1969] used aircraft-launched seabed drifters to study the bottom drift in Monterey Bay. He made two launches, one on 25 May 1962 consisting of 115 seabed drifters and one on 15 February 1963 using 172 drifters. Nine and fifteen drifters were recovered representing a recovery rate of 7.5% and 15% respectively. Three recoveries from the first launch were from the Monterey Submarine Canyon. One was at a depth of 915 meters and two were at a depth of 549 meters. Cumulative drift rate was reported as 0.07 km/day (0.08 cm/sec) to the northeast for one of the 549-meter drifters and for the 915-meter drifter. The other 549-meter drifter was recovered

1,456 days after launch; therefore, the cumulative drift rate was not calculated. Drifters in the Canyon from the second launch showed drifts of less than 0.06 km/day (0.07 cm/sec) to the east. In general, the lowest drift rates for the Bay were reported from the drifters dropped into the Canyon.

Shepard and Marshall [1969], using the Isaacs-Schick continuous recording current meter system, measured currents in submarine canyons off La Jolla, California. Their measurements, 3.6 meters above the bottom, of three to six days, included simultaneous stations of up to six current meters and in La Jolla Canyon a vertical array of three current meters 3.6, 19 and 34 meters above the bottom. They reported that there is no relationship between the tide and current flow at this location. Although correlation between two stations was high, flows in opposite directions did occur. They felt that this suggested cellular flow.

C. PURPOSE OF THIS INVESTIGATION

The purpose of this research was to: (1) continue near-bottom current measurements in the Monterey Submarine Canyon, extending the work of previous investigators into deeper regions of the Canyon and onto the surrounding shelf, (2) calculate the power spectra and volume transport for all continuous records of sufficient length using new programs specifically designed for the purpose and (3) analyze all continuous records to determine significant peaks in the energy spectrum and trends in the volume transport, define more clearly the relationship between tidal forces and the currents in the Canyon, investigate the relationship between the currents on the shelf and in the Canyon, and study the relationship of observed current set and volume transport with the Monterey Bay seasonal periods.

II. REGION OF OBSERVATION

A. MONTEREY BAY

Monterey Bay is located 121 km south of San Francisco (Figure 1) and is in open communications with the Pacific Ocean, thus subject to oceanic variations. Skogsberg [1936] has divided the Bay's water conditions into three periods: (1) the Davidson Period - from the middle of November to the middle of February, when the Davidson Current brings relatively warm water northward near shore, (2) the upwelling period - between the middle of February and the end of July when cold waters upwell in the Bay and (3) the oceanic period - from the end of July until the middle of November, when the California Current transports cooler water southward.

B. MONTEREY SUBMARINE CANYON

The bottom of the Bay is cut by a large V-shaped, meandering canyon called the Monterey Submarine Canyon (Figure 2). This Canyon is one of the largest in the world and has two tributaries, the Carmel Canyon to the south and the smaller Soquel Canyon to the north. The Canyon extends seaward some 53 km into the Monterey Trough which continues 51.5 km further with diminishing relief. Beyond the Trough, deep channels have been traced for 250 km giving a total of 354.5 km for the extent of the Monterey Canyon, Trough and Deep-sea Channels. Side slopes for the Monterey Canyon range from 5 to 34 degrees; slopes for the Soquel and Carmel Canyons range from 8 to 40 degrees and 1 to 25 degrees respectively. Monterey Canyon's axial gradient flattens from 8.5 degrees at its head to 0.5 degrees at the seaward limit [Martin and Emery, 1967].

III. MEASURING EQUIPMENT AND PROCEDURES

A. CURRENT MEASURING SYSTEM

All the current data analyzed in this investigation were collected with Hydro Products Model 501B Savonius current meters. This meter is a self-contained instrument package measuring current speed and direction as well as water temperature. The current sensor is a Savonius rotor, the direction sensor is a coupled potentiometer reading relative to a magnetic compass and the temperature sensor is a thermistor. The readout package is a solid-state circuit connected to a two-inch Rustrak paper chart recorder. A self-contained, rechargeable 6-volt nickel cadmium battery pack is used for the system's power.

On the scale used in this investigation, the speed range is zero to one knot (51.5 cm/sec) with a threshold of 0.05 knot and an accuracy $\pm 3\%$ of the reading. The direction has an accuracy of ± 5 degrees. Temperature between zero and 40°C can be recorded to an accuracy of $\pm 3\%$ of the reading. A seven-day continuous record of current speed is possible, but the temperature and direction use the same channel and the system switches between these sensors. The three readouts recorded (Figure 3) on the two-inch wide strip chart are distinguished by their appearance as a series of dots whose lengths are proportional to the recorded time for each one of the inputs.

The switching sequence for the seven-day recording interval is continuous speed, direction for five minutes, temperature for $1\frac{1}{2}$ minutes and neither direction nor temperature for one minute. The sequence then begins again Franz, [1967]. The sensors and the recorder electronics

STRIP CHART

Figure 3.

are mounted in a bird-cage frame. A 102 by 61 cm aluminum vane was attached to the current meter frame to give better directional stability.

B. RELEASE MECHANISM

The timed release mechanism used was the Braincon Model 422 release. The release mechanism allowed the measuring system to return to the surface for recovery. After the timing mechanism has counted down from its preset time, an explosive squib fires which allows the release to disengage the line connecting the release and the mooring weights.

C. TAUT-LINE MOORING SYSTEM AND MOORING PROCEDURE

Three moorings were successfully established during the course of this investigation. The measurements were made as close as practicable to the bottom to allow convenient setting of the mooring and to be above any rock outcrops. The taut-line mooring system (Figure 4) used was very similar to that developed by Dooley [1968] and later used by Njus [1968]. Appendix A discusses problems encountered with the taut-line moor.

Navigation was by visual or radar fixes from the Naval Postgraduate School (NPS) sixty-three foot research boat. Depths were determined using the vessel's fathometer. As the mooring site was approached, the recoverable section was paid out and allowed to stream behind the boat. When the proper location was reached, the current meter and release were lowered over the side and then the mooring weights were dropped.

Taut-Line Mooring

Figure 4.

D. TIDE RECORDING EQUIPMENT

Tidal records were obtained from a standard recording tide gauge located on Wharf Number 2 in Monterey Harbor. The heights of the tide are the same in Monterey as at Moss Landing and the time difference of three minutes was ignored [Dooley, 1968].

IV. CONTINUOUS TIME-SERIES CURRENT OBSERVATIONS

Six moorings were attempted during this investigation and three moorings were successfully completed. Two were set simultaneously on 11 April 1969, one in the Canyon at a depth of 366 meters and another on the north shelf of the Canyon at a depth of 91 meters. A third moor was made on 25 April 1969 at the previous 366-meter site.

Dooley [1968] made nine moors. Five yielded data in excess of 50 hours and were reanalyzed during the course of this investigation. Njus [1968] made three successful moors. The first did not record current speed; hence, only the latter two were reanalyzed in this study. Figure 5 indicates the sites occupied in this and previous investigations and Table I summarizes the successful observations of this and earlier work.

Figure 5.

TABLE I
OBSERVATION PROGRAM

Dooley 1968

START/FINISH TIME (Local)	RECORD LENGTH (Hours)	POSITION	DEPTH (Meters)	REMARKS / MALFUNCTIONS
1000 22 Mar 67 to 1340 24 Mar 67	50	36°48.09'N 121°48.53'W	119	Testing Taut-Moor
1048 10 Apr 67 to 1454 11 Apr 67	28	36°48.10'N 121°48.60'W	121	Testing Taut-Moor with surface recovery buoy.
1110 24 Nov 67 to 2210 26 Nov 67	56	36°48.15'N 121°48.40'W	128	Ceased recording at 262210, normal recovery at 272310 Nov 67.
1426 15 Jan 68 to 2326 21 Jan 68	153	36°48.10'N 121°48.60'W	133	Recovered buoyancy float by dragging 201500 Jan 68. Re- covered meter by dragging 081400 Feb 68. Squib failed to fire after clock time expired.
0945 29 Feb 68 to 1410 2 Mar 68	52	36°47.92'N 121°51.05'W	165	Recovered in surf off Moss Landing 20 Mar 68. Time release failed to disengage after squib fired. Speed rotor malfunc- tion after 52 hrs.
1610 18 Apr 68 to 1439 25 Apr 68	5	36°48.18'N 121°48.18'W	110	Meter malfunction, launched with diffi- culty in heavy seas.
1307 30 Apr 68 to 1436 7 May 68	0	36°48.18'N 121°48.15'W	95	Meter malfunction, electronics failure.
1515 7 May 68 to 1428 14 May 68	0	36°48.19'N 121°48.20'W	108	Meter malfunction, electronic failure.
1520 14 May 68 to 1526 21 May 68	162	36°48.08'N 121°48.52'W	104	Normal recovery at 211535 May 68.

TABLE I (Continued)

Njus 1968

START/FINISH TIME (Local)	RECORD LENGTH (Hours)	POSITION	DEPTH (Meters)	REMARKS
1030 10 Sep 68 1000 18 Sep 68	89.25	36° 47.90'N 121° 48.90'W	165	1. No speed data obtained. 2. Rustrak recorder malfunctioned.
1230 26 Sep 68 1250 3 Oct 68	168.3	36° 48.05'N 121° 48.70'W	160	
1050 17 Oct 68 0950 24 Oct 68	168	36° 47.50'N 121° 49.35'W	202	

Caster 1969

START/FINISH TIME (Local)	RECORD LENGTH (Hours)	POSITION	DEPTH (Meters)	REMARKS
1215 11 Apr 69 to 1149 18 Apr 69	167.56	36° 47.98'N 121° 51.58'W	366	
1148 11 Apr 69 to 1503 18 Apr 69	171.25	36° 48.76'N 121° 51.98'W	91	
1118 25 Apr 69 to 0714 26 Apr 69	19.94	36° 47.98'N 121° 51.58'W	366	The current meter chart recorder had blank periods indicating no data had been recorded. The record was broken into the intervals indicated in the far left column and each period containing data was handled as a separate record.
1038 26 Apr 69 to 1357 26 Apr 69	3.37			
1533 26 Apr 69 to 0052 27 Apr 69	9.32			
0148 27 Apr 69 to 0033 3 May 69	142.75			

V. DATA ANALYSIS

A. DATA SAMPLING AND REDUCTION OF CURRENT RECORD PROCEDURES

The Hydro Products current meter provided a record of current speed and direction and water temperature. These records were digitized at a sampling interval of 3.75 minutes. Since the temperature was recorded every 7.5 minutes, the intermediate values were interpolated. The current meter data was processed on the NPS Computer Facility IBM 360 computer using programs discussed in section C of this chapter.

B. SAMPLING PROCEDURE AND DATA REDUCTION OF TIDAL RECORDS

The marigrams for the observational periods were digitized using a Calma Company Model 480 digitizer at the Pt. Pinos Annex of the Fleet Numerical Weather Central (FNWC). PROGRAM CONVERT (Appendix F) was used on the FNWC CDC 6500 computer to convert the digitized information stored on magnetic tape to punched computer cards. Procedures for digitizing the records are outlined in Appendix A of Robinson [1969]. The sampling interval of the digitizer was 0.01 inch; however, for this investigation, only every 0.07 inch was printed as output and punched onto computer cards giving a sampling interval of approximately 3.86 minutes. These data were then input to PROGRAM TIDE (Appendix G) which provided a graph of tidal fluctuations.

C. ELEMENTARY STATISTICS

First order statistics for the current meter records were computed using standard techniques in PROGRAM CURRENT (Appendix H) written by Dooley [1968] and modified by Njus [1968]. A transformation of current

direction suggested by Webster [1964a] was employed to facilitate the computations for direction.

PROGRAM CURRENT provides:

- (1) Hourly means
- (2) Daily means, medians, modes and frequency distribution
- (3) Time series means, medians, modes, variances, standard deviations and frequency distribution
- (4) Graphic histograms.

PROGRAM COMPOSITE DRAW (Appendix I) was used to give a graphic display of current speed, current direction and water temperature. This program was modified to PROGRAM TEMPERATURE DRAW (Appendix J) to give a better display of temperature.

D. POWER SPECTRA ANALYSIS

Power Spectra analysis was performed on all the available records of sufficient length using programs originally developed at the Institute of Oceanography of the University of British Columbia (UBC). The UBC system consists of three programs: FTOR, SCOR and FCPLLOT (Appendix K). The data is initially read onto a magnetic tape using the following program:

```
C      CARD TO TAPE FOR TIME SERIES PROGRAMS
C
      DIMENSION TEMP(5), SPEED(5), DIR(5)
      JCTR = 15
      NCHANS = 3
10  READ (5,2001, END=20) (TEMP(I),SPEED(I),DIR(I),I=1,5)
      WRITE (6,100) JCTR, NCHANS, (TEMP(I),SPEED(I),DIR(I),I=1,5)
100  FORMAT (2I5,15F8.2)
      GO TO 10
20  END FILE 4
      REWIND 4
2001  FORMAT (5(F5.1,F5.2,F4.0))
      END
```

Data blocks of 2^n points, where n is a positive integer less than or equal to 13, can be used and two or more blocks are required. After the data has been read onto the tape, FTOR computes the Fourier

coefficients for the time series and writes them on another tape. SCOR then reads the output of FTOR and computes spectrum, cospectrum and quadrature spectrum for the series. FCPLLOT is used to give a Calcomp plot of the spectrum. These procedures are described in greater detail by Wilson, et al, [1969].

VI. DISCUSSION

A. GRAPHIC PLOT AND STATISTICAL ANALYSIS

Plots from programs COMPOSITE DRAW, TEMPERATURE DRAW and TIDE are shown in Appendix B. Figures 24 through 27 are for the ^{300-ft} 91-meter array located on the north shelf of the Canyon on 11 April 1969 while results from the ^{1200-ft} 366-meter array in the Canyon of the same date are shown in Figures 28 to 31. These are the first simultaneous records of near-bottom currents in Monterey Submarine Canyon and on the adjacent shelf. In the shelf record, the general flow is to the southeast, into the Canyon, particularly on the rising tide. At high tide and for a few hours into falling tide the flow fluctuates, sometimes changing to a northerly direction. Speeds are quite low, seldom over 11 cm/sec, and the highest speeds occur near high tide.

In contrast with the shelf record, records from the Canyon for the same period show that falling tide is associated with up-canyon flow and rising tide with down-canyon flow. This oscillatory flow has been observed in previous studies in the Canyon. Maximum speeds reached in the Canyon were in excess of 35 cm/sec and usually occurred with down-canyon flow, although this was not always the case.

The 366-meter array of 25 April 1969 is shown in Figures 32 through 37. The direction is nearly constant at 165°T. Blank spaces in the record indicate periods of no data on the strip-chart recorder. Dooley [1968] reported a single mode for a record collected on 24 November 1967 at a depth of 128 meters in the Canyon. His record covered only a 56-hour period and no further discussion was given to explain this anomaly. Undoubtedly, local irregularities in the bottom topography

of the Canyon could cause variations from the expected; however, for this record, it is felt that the current meter was not properly recording direction. The speed record for the 25 April record shows that maximum speeds normally occur at high tide as seen in the shallow record of 11 April. Speeds often reach 51.5 cm/sec and remain over 25 cm/sec for several hours. The speeds build rapidly to a maximum and then gradually decrease to a minimum.

Figures 6 through 11 show the frequency distributions computed by PROGRAM CURRENT of temperature, current speed and current direction. Table II summarizes the elementary statistics for records from the 91- and 366-meter arrays set simultaneously on 11 April 1969 and the 366-meter array of 25 April 1969. Histograms for the arrays placed at 91 and 366 meters are shown in Figures 6 and 7 respectively. At 91 meters, the speed is unimodal with no values over 25 cm/sec and direction is scattered with a predominate peak between 105 and 155°T. Mean speed and direction for this observation are 6.7 cm/sec and 137°T, with standard deviations of 2.6 cm/sec and 63.7°. The 366-meter array shows that current speed and direction exhibit characteristics similar to those previously observed in Monterey Submarine Canyon by Dooley and Njus. Speeds range from zero to greater than 25 cm/sec and current direction is bimodal with the modes approximately 180° apart. Mean speed and direction are 10.8 cm/sec and 156°T, while standard deviations are calculated to be 7.2 cm/sec and 89.6°.

Histograms from the 366-meter array of 25 April 1969 are shown in Figures 8 through 11. For the first 38½ hours of the record, speed remains less than 25 cm/sec; however, for the portion starting at 0148 on the 27th of April speeds of 51.5 cm/sec are observed.

Figure 6. 1148 11 April to 1503 18 April 1969 Frequency Distributions

Figure 7. 1215 11 April to 1149 18 April 1969 Frequency Distributions

Figure 8. 1118 25 April to 0714 26 April 1969 Frequency Distributions

Figure 9. 1038 26 April to 1357 26 April 1969 Frequency Distributions

Figure 10. 1533 26 April to 0052 27 April 1969 Frequency Distributions

Figure 11. 0148 27 April to 0033 3 May 1969 Frequency Distributions

TABLE II
SUMMARY OF CURRENT STATISTICS

START/FINISH TIME (Local)		WATER TEMPERATURE	CURRENT SPEED	CURRENT DIRECTION
1148 11 April to 1503 18 April 1969	Mean Variance Standard Deviation Median Mode	8.38°C 0.04(°C) ² 0.19°C 8.40°C 8.45°C	6.67 cm/sec 6.59(cm/sec) ² 2.57 cm/sec 6.2 cm/sec 7.7 cm/sec	132°T 4057(°T) ² 63.2°T 130°T 115°T
1215 11 April to 1149 18 April 1969	Mean Variance Standard Deviation Median Mode	6.86°C 0.07(°C) ² 0.27°C 6.90°C 7.12°C	10.8 cm/sec 51.1(cm/sec) ² 7.16 cm/sec 10.3 cm/sec 4.12 cm/sec	156°T 8033.5(°T) ² 89.6°T 180°T 190°T
1118 25 April to 0714 26 April 1969	Mean Variance Standard Deviation Median Mode	6.96°C 0.03(°C) ² 0.16°C 7.00°C 7.12°C	4.12 cm/sec 32.1(cm/sec) ² 5.67 cm/sec 2.08 cm/sec 1.55 cm/sec	196°T 1252(°T) ² 35.4°T 190°T 165°T
1038 26 April to 1357 26 April 1969	Mean Variance Standard Deviation Median Mode	6.73°C 0.04(°C) ² 0.19°C 6.70°C 6.63°C	14.95 cm/sec 4.24(cm/sec) ² 2.06 cm/sec 13.9 cm/sec 15.5 cm/sec	164°T 19(°T) ² 4.4°T 165°T 165°T
1533 26 April to 0052 27 April 1969	Mean Variance Standard Deviation Median Mode	6.67°C 0.03(°C) ² 0.16°C 6.80°C 6.75°C	6.70 cm/sec 21.4(cm/sec) ² 4.63 cm/sec 5.66 cm/sec 4.12 cm/sec	166°T 582(°T) ² 24.2°T 168°T 170°T
0148 27 April to 0033 3 May 1969	Mean Variance Standard Deviation Median Mode	6.82°C 0.11(°C) ² 0.33°C 6.87°C 7.08°C	15.4 cm/sec 108.2(cm/sec) ² 10.8 cm/sec 13.9 cm/sec 5.15 cm/sec	162°T 398(°T) ² 19.9°T 162°T 165°T

Direction is markedly constant and is believed to be in error. Mean speed and current direction for the longest portion of the record is 15.4 cm/sec and 162° T with standard deviations of 10.8 cm/sec and 19.9° . These compare to a mean of 10.8 cm/sec and 156° T for the 11 April array at the same depth. Tidal range for the last 80 hours of the 25 April 366-meter array is much higher than for the 11 April array at the same location. This may account for the higher speeds reported for the 25 April array. Dooley's and Njus' time-series data revealed this effect in isolated cases.

Temperatures for all records show narrow unimodal peaks with the shallower record having warmer waters as expected. Means for the simultaneous 91- and 366-meter arrays are 8.4 and 6.9°C with standard deviations of 0.2 and 0.3°C . For the 366-meter array on 25 April, the mean is 6.8°C with the standard deviation being 0.3°C .

B. PROGRESSIVE VECTOR DIAGRAM

A progressive vector diagram (PVD) is one graphic method of showing a time series of current velocity. The diagram is constructed by adding successive current vectors. As stated by Webster [1964b], the temptation is to think of the diagram as representing the path of a particle's trajectory; however, this can be misleading, since the PVD represents particle trajectory only when the field of motion is independent of position over the spatial scale. This seldom occurs in nature, particularly in shallow water.

PROGRAM VECTOR DIAGRAM (Appendix L) was written to (1) provide a PVD from the hourly values calculated by PROGRAM CURRENT and (2) calculate the volume transport at the array location. A scatter diagram (SD) of the velocity vectors was obtained to compliment the

PVD using the PLOTP subroutine available in the NPS computer program library. These presentations are new to the analysis of Monterey Submarine Canyon current data. Because of this fact, the data collected by Njus and Dooley are also analyzed and presented in Appendix C and D.

The scatter diagrams of the data collected in this investigation are shown in Figures 12 through 14. The first SD is for the 91-meter array of 11 April. It shows the predominate mode towards the south-east. Figure 13, for the 366-meter array of 11 April, shows the expected bimodal tendency of currents in the Canyon; however, both northeastward and southwestward modes are skewed towards a southerly direction. The longest portion of the 25 April 366-meter array is shown in Figure 14.

The progressive vector diagrams for the same data are shown in Figures 15 to 18. Successive 24-hour periods are marked with a "X." Figure 15 is the PVD for the shelf record and Figure 16 is for the Canyon record of the same date. Both of these figures show very clearly a feature of the Canyon circulation not previously reported. The PVD from the deeper array shows the cyclic up-canyon and down-canyon flow previously reported but also a net set to the southeast across the Canyon. As expected from the SD of the shelf array, the net current set is also southeast into the Canyon. The combination of the two simultaneous arrays are shown in Figure 17. The net set for both sites is to the southeast, particularly for the first $3\frac{1}{2}$ days of the record. Since the Canyon axis for the 366-meter site is 085-265°T, the set is cross canyon..

For the period of these observations, winds in the Monterey Bay area were very strong from the west to northwest. This would cause

Scatter Diagram

N

Figure 12. 1148 11 April to 1503 18 April 1969

**Scatter
Diagram**

N →

Figure 13. 1215 11 April to 1149 18 April 1969

Scatter Diagram

Figure 14. 0148 27 April to 0033 3 May 1969

a longshore current to the south in the Bay. The bottom topography of the Canyon near the 91-meter array (Figure 5) would funnel the south-flowing longshore current into the Canyon in a cross-canyon direction. While the cyclic up-canyon and down-canyon tidal currents would persist, this inflow from the north could cause the 366-meter array in the Canyon to exhibit a net set toward the south.

The PVD (Figure 18) for the 366-meter array of 25 April also shows a southeastward set. As mentioned before, this record is believed to be in error; however, it is interesting to note that the net set shown is in the same general direction as the set from the earlier array at this depth.

Dooley's and Njus' records of SD are shown in Appendix C (Figures 38 to 44). The PVDs of their records are shown in Appendix D (Figures 45 to 51) and display the net current set for the periods of observation. Njus' record of 26 September 1968 (Figure 46) at a depth of 160 meters shows the cyclic tidal fluctuations and the net current set to the southeast as do the records of this investigation. The remainder of their records, however, show various directions of set.

Total volume transport for each array was calculated and divided by the number of hours each array recorded data, giving volume transport in cubic meters per hour. A summary of net current set and volume transport for all records is given in Table III. These volume transports and current sets are plotted in Figure 19 as a function of months in an attempt to relate the current set and volume transport to the ocean seasons of the Monterey Bay as defined by Skogsberg [1936]. The volume transport and net current set for the 25 April 1969 record was not plotted because as previously discussed, this record is in question.

PROGRESSIVE VECTOR DIAGRAM

91 meters

N

0 1 2 3 4 5
Kilometers

* end

Figure 15. 1148 11 April to 1503 18 April 1969

PROGRESSIVE VECTOR DIAGRAM

Figure 17. Simultaneous 91- and 366-meter Arrays

Volume Transport and Current Set vs. Months

Figure 19.

TABLE III
SUMMARY OF RESULTS OF PROGRAM VECTOR DRAW

Figure Number	Date (Start of Record)	Depth (Meters)	Net Current Direction Set ($^{\circ}$ T)	Volume Transport (Meters ³ /Hour)
45	1000 22 March 1967	119	294	171
46	1110 24 November 1967	128	228	486
47	1426 15 January 1968	133	222	117
48	1010 29 February 1968	165	011	122
49	1507 14 May 1968	104	138	85
50	1230 26 September 1968	160	140	355
51	1050 17 October 1968	202	190	369
15	1148 11 April 1969	91	121	168
16	1215 11 April 1969	366	147	110
18	0148 27 April 1969	366	160	548

Current set is predominately south for the oceanic period and the latter part of the upwelling period. North set is observed for the transition between the Davidson Current and upwelling periods. Although the set would be expected to be northerly during the Davidson Current period and southerly for the oceanic and upwelling periods, the dividing line between the seasons is not well defined and during the periods reported in this investigation, set in the Canyon was to the south during the Davidson Current period.

Volume transport changes significantly from the oceanic to the Davidson Current period and through the upwelling period. Transports of approximately $400 \text{ m}^3/\text{hr}$ are seen toward the latter part of the oceanic period and then decrease to less than $200 \text{ m}^3/\text{hr}$ during the remaining two seasons.

C. SPECTRAL ANALYSIS

Spectral estimates of temperature and speed for the records collected during this investigation are shown in Figures 20 to 22. Figure 23 can be used to convert $\log_{10}$ frequency to the period in hours. The short portions of the 25 April 1969 record were not analyzed.

For the 91-meter array, temperature peaks occurred at 34.5, 12 and 5 hours and speed peaks at 16, 5 and 0.5 hours. The 366-meter array on the same date shows periods of 31.5 and 12 hours for temperature and 35, 16 and 5 hours for speed. For the longer portion of the 25 April 366-meter array, periods of 12 and 7 hours for temperature and 22.75, 11 and 7 hours for speed are observed. The consistent peak at 12 hours is obviously tidal.

Figure 20. 1148 11 April to 1503 18 April 1969

Figure 21. 1215 11 April to 1149 18 April 1969

Figure 22. 0148 27 April to 0033 3 May 1969

Dooley's and Njus' records are shown in Appendix E (Figures 52 to 58). Table IV combines these records and those collected during this investigation. For the longer records, periods of 22.75, 22.5, 12 and 11 hours are probably tidal periods. Periods of 35, 34.5, 31.5 and 31 hours are also noted and the driving forces for these periods are as of yet unknown. Dooley [1968] used spectral program BMD02T and reported a peak at 12 hours for temperature and peaks at 12, 4-6 and 2-3 hours for speed. Njus [1968] used spectral program GHOST and reported temperature peaks at 12.5, 6 and 4 hours and speed peaks at 12.5, 6, 4 and 2.5 hours.

Shepard and Marshall [1969] reported that periods of four hours might be associated with internal waves. The inertial period for the latitude of Monterey Bay is approximately 20 hours; however, no peaks at this period were observed.

Using Monterey Harbor tide records, Ranines [1967] found peaks from 19 to 39 minutes, while Robinson [1969] using Monterey Harbor and Santa Cruz Harbor tide records found peaks of approximately 30 minutes.

After all the records were plotted, it was noticed that the slope of the higher frequency portions were decreasing in a similar manner. Measurements of the slope of the fall-off portion of the spectrum yields, for most cases, a slope of $-1.732/1$ (60°). Table V summarizes the values for all records. Batchelor [1959] discusses the inertial-convective sub-range (ISR) for turbulent measurements. A slope of $-5/3$ (-1.66) has been observed for the fall-off of the ISR.

The slopes for the temperature spectrum are approximately the same for all records taken from the Canyon. The spread is from $-1.54/1$ to

-2.14/1. The record from the shelf has a slope of -1.33/1. For speed, the slopes are -1.54/1 to -2.25/1 for Canyon records and -1.07/1 for the shelf record. This may indicate a constant method of energy transfer for the Canyon records, but a different method of energy transfer for the shelf record.

TABLE IV
SUMMARY OF POWER SPECTRUM

Figure Number	Date (Start of Record)	Length of Record (Hours)	Periods (Hours)	
			Temperature	Speed
52	1000 22 March 1967	50	None	5.5, 1.75
53	1110 24 November 1967	56	None	4.5
54	1426 15 January 1968	154	20.5, 11	31, 11, 4.5, 0.5
55	0945 29 February 1968	56	4.5	5.5, 0.75
56	1520 14 May 1968	81	9.75, 6, 2.25	9.75, 4.5
57	1230 26 September 1968	159	11, 3.5	22.5, 12, 7
58	1050 17 October 1968	167	22.5, 12	12, 7, 4
20	1148 11 April 1969	171	34.5, 12, 5	16, 5, 0.5
21	1215 11 April 1969	168	31.5, 12	35, 16, 5
22	0148 27 April 1969	142	12, 7	22.75, 11, 7

TABLE V
SUMMARY OF SLOPE OF SPECTRUM DATA

Date (Start of Record)	Slope	
	Temperature	Speed
1000 22 March 1967	-1.54/1	-1.60/1
1110 24 November 1967	-1.66/1	-2.05/1
1426 15 January 1968	-1.73/1	-1.73/1
0945 27 February 1968	-1.73/1	-1.88/1
1520 14 May 1968	-1.88/1	-2.25/1
1230 26 September 1968	-1.80/1	-1.88/1
1050 17 October 1968	-2.14/1	-1.96/1
1148 11 April 1969	-1.33/1	-1.07/1
1215 11 April 1969	-1.73/1	-1.73/1
0148 27 April 1969	-1.60/1	-1.54/1

VII. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

Currents in the Monterey Submarine Canyon reach speeds of 51.5 cm/sec (the instrument's cut-off) and obviously go higher. Normally the speeds build up rapidly on a rising tide and decrease slowly on the falling tide. Motion is warm, down-canyon flow on rising tide and cold, up-canyon flow on the falling tide. Superimposed on the tidal up- and down-canyon flow is a flow probably due to longshore current across the Canyon. Scatter diagrams and progressive vector diagrams show this effect, indicating a net current set across the Canyon.

Currents on the shelf show lower speeds, seldom reaching 25 cm/sec. Data from the array north of the Canyon indicate flow into the Canyon in a southeastern direction. This is believed to be due to a pre-dominate southerly longshore current caused by strong west to northwest winds during the observational period and the bottom topography of the Canyon which funnels the flow into and across the Canyon.

An attempt was made to correlate the direction of set and volume transport with Monterey Bay seasonal water conditions. However, only nine sets of data were used and they were of differing length, depth and location. Nevertheless, there is some indication of a relationship between currents in the Canyon and ocean season.

Power spectra of all data available show peak periods that correspond to the tide; however, other periods exist and their cause is not defined as of yet. The power spectrum program used computed

a raw spectrum; since the tidal influence is very strong, other frequencies may be blanked out by the tide.

Measurement of the slope of the power spectrum revealed that this slope is nearly constant at $-1.73/1$. This suggests a uniform transfer of energy from low frequency to high frequency.

B. RECOMMENDATIONS

Further study of bottom currents is warranted. Continuous measurements at several locations in the Monterey Submarine Canyon, on the shelf area and at different depths would be desirable in order to better define the flow characteristics in the Bay. A simultaneous bathymetric and bottom composition study, combined with a study of the water's physical parameters by either STD or Nansen casts at the location of current measurements, would more completely define the oceanographic conditions.

Ten records each longer than 50 hours are now available for further analysis. Dooley [1968] and Njus [1968] have suggested re-evaluation of their data by filtering out the tidal components. While more data is needed, the available data should be reanalyzed in accordance with their suggestions.

Two limitations now inhibit study of the near-bottom currents. First, the amount of line necessary for the taut-line moor in deeper sites is reaching proportions that cannot be handled on the Naval Postgraduate School research boat. The system should either be redesigned to accommodate the boat or research should be done from a larger vessel.

Secondly, data reduction is a time-consuming task and limits the amount of information one can collect and effectively handle. The

digitizer at the Fleet Numerical Weather Facility Pt. Pinos Annex offers a solution to this problem. Not only would data reduction be faster using this device but the sampling interval would be smaller, allowing the investigator to look at higher frequencies.

APPENDIX A

DIFFICULTIES WITH TAUT LINE MOORING

On 19 February 1969, three current meters were placed in the following locations:

<u>Position</u>				<u>Depth</u>
36° 48.8'N	121° 51.9'W			91 meters
36° 47.9'N	121° 51.7'W			366 meters
36° 46.9'N	121° 51.6'W			91 meters

These positions are approximately 5.5 km west of Moss Landing. The R/V OCEANEER was used for launching. The southern-most array used a subsurface float and release mechanism that had not been used at sea. All surface floats were made of styrafoam, colored international orange, and banded together between sheets of plywood with a short wooden staff with a red flag attached to it.

The OCEANEER was to recover the meters on the 25th of February. During the period the meters were in the water, extremely high winds were experienced in the Bay. On the day of recovery, the center and northern meters were found (one without a surface float) and the southern meter and its surface float were not found.

On the 26th of February, both of the remaining releases failed upon cocking due to breakage of the cocking pawl, so attempts to reset the meters were aborted. Of the two meters recovered, one had malfunctioned in the chart recorder and the other had little usable data because it did not have the 102 by 61 cm vane attached and it had been set for high scale speed of zero to 360.4 cm/sec. A decision was then made to install vanes on the two meters and have the Naval Postgraduate School Machine Facility make new cocking pawls.

On the 27th, the USCGC CAPE WASH (95310) attempted to drag the area of the missing meter. Due to rough seas, high wind and the Cutter's high minimum speed, the drag was of questionable value. An Army helicopter from U.S. Army Base, Fort Ord was used the next day to search the beach area from Fort Ord to Santa Cruz and back. Neither the meter nor the two missing floats were seen. Winds had been onshore and it was felt that if the equipment had been missed at sea it would have drifted ashore.

As soon as the meter vanes were installed, the cocking pawls built and the Naval Postgraduate School's sixty-three footer available, the meters were reset with the assistance of Lt. Dana W. Starkweather, USCG and Major James Neehan, USMC. This took place on 12 April with the meters being placed at the center and northern positions attempted previously. After setting the meters, a drag was made for the missing meter with no results.

The above two meters were recovered by the author and Mr. Jack Mellor on 18 April. Although plans were originally made to reset the meters immediately after they were recovered, sea conditions dictated that the meters be reset under more favorable circumstances. One styrafoam float was missing.

Two meters were to be reset by the author and Mr. Bob Middleburg on 25 April. One meter was placed at the center location and another was to be placed in 549 meters (300 fathoms) of water. As the meter was being lifted over the boat's lifelines, the instrument package cover snapped off and fell into the sea, so only the 366-meter position was occupied.

The meter was due to surface on 2 May; however, it was not found. On the 3rd of May an Army spotter plane searched the beach areas with no results. The meter was finally found on the 16th of May in approximately 274 meters of water (the length of line from the release to the subsurface float is approximately this distance).

The cocking nut on the release apparently had not turned completely when the release squib fired. It appeared that the edge of the nut had finally worn away and released the recoverable portion of the mooring, which drifted ashore until anchoring itself with the release mechanism. The styrafoam surface float was never found.

Lessons learned were:

- (1) The aluminum vane is needed on the Model 501B current meter.
- (2) Only the low speed scale should be used for bottom current measurements.
- (3) After the current meter has been turned on, the instrument package cover should be wired closed.
- (4) The surface float should have little freeboard to cut down the effects of wind, and if the releases are to be trusted the surface floats are of little use.
- (5) If at all possible, pick a period of light winds to moor and recover the arrays.

Figure 24a. 1148 11 April to 0848 13 April 1969

Figure 24b. 1148 11 April to 0848 13 April 1969

Figure 25a. 0848 13 April to 0548 15 April 1969

Figure 25b. 0848 13 April to 0548 15 April 1969

Figure 26b. 0548 15 April to 0248 17 April 1969

Figure 27a. 0248 17 April to 1503 18 April 1969

Figure 27b. 0248 17 April to 1503 18 April 1969

Figure 28a. 1215 11 April to 0915 13 April 1969

Figure 28b: 1215 11 April to 0915 13 April 1969

Figure 29a. 0915 13 April to 0615 15 April 1969

Figure 29b. 0915 13 April to 0615 15 April 1969

Figure 30b. 0615 15 April to 0315 17 April 1969

Figure 31a. 0315 17 April to 1149 18 April 1969

Figure 3lb. 0315 17 April to 1149 18 April 1969

Figure 32a. 1118 25 April to 0714 26 April and 1038 26 April to 1357 26 April 1969

Figure 32b. 1118 25 April to 0714 26 April and 1038 26 April to 1357 26 April 1969

Figure 33a. 1533 26 April to 0052 27 April 1969

Figure 33b. 1533 26 April to 0052 27 April 1969

Figure 34a. 0148 27 April to 2248 28 April 1969

Figure 34b. 0148 27 April to 2248 28 April 1969

Figure 35a. 2248 28 April to 1948 30 April 1969

Figure 35b. 2248 28 April to 1948 30 April 1969

Figure 36a. 1948 30 April to 1648 2 May 1969

Figure 36b. 1948 30 April to 1648 2 May 1969

Figure 37a. 1648 2 May to 0033 3 May 1969

Figure 37b. 1648 2 May to 0033 3 May 1969

SCATTER DIAGRAMS OF DOOLEY'S AND NJUS' DATA

Scatter
Diagrams

Figure 38. 1000 22 March to 1340 24 March 1967

N

Figure 39. 24 November to 26 November 1968

Figure 40. 1426 15 January to 2326 21 January 1968

Z ←

Figure 41. 0945 29 February to 1410 2 March 1968

Figure 42. 1520 14 May to 1526 21 May 1968

Figure 43. 1230 26 September to 1250 3 October 1968

Figure 44. 1050 17 October to 0950 24 October 1968

APPENDIX D
PROGRESSIVE VECTOR DIAGRAMS OF
DOOLBY'S AND NJUS' DATA

PROGRESSIVE VECTOR DIAGRAM

N

0 1 2 3 4 5
Kilometers

Figure 46. 1110 24 November to 2210 26 November 1967

Figure 47. 1426 15 January to 2326 21 January 1968

PROGRESSIVE VECTOR DIAGRAM

Figure 48. 0945 29 February to 1410 2 March 1968

PROGRESSIVE VECTOR DIAGRAM

Figure 49. 1520 14 May to 1526 21 May 1968

PROGRESSIVE VECTOR DIAGRAM

Figure 50. 1230 26 September to 1250 3 Oct 1968

PROGRESSIVE VECTOR DIAGRAM

Figure 51. 1050 17 October to 0950 24 October 1968

APPENDIX E
SPECTRAL ESTIMATES OF
DOOLEY'S AND NJUS' DATA

Figure 52. 1000 22 March to 1340 24 March 1967

Figure 53. 1110 24 November to 2210 26 November 1967

Figure 54. 1426 15 January to 2326 21 January 1968

Figure 55. 0945 29 February to 1410 2 March 1968

Figure 56. 1520 14 May to 1526 21 May 1968

Figure 57. 1230 26 September to 1250 3 October 1968

Figure 58. 1050 17 October to 0950 24 October 1968

APPENDIX F

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
PROGRAM CONVERT
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C

```

THIS PROGRAM CONVERTS THE DIGITIZED RECORDS OF THE NAVAL POSTGRADUATE SCHOOL TIDE GAUGE, LOCATED ON WHARF NO. 2 IN MONTEREY HARBOR, FROM EXTERNAL BCD TO AMPLITUDE VALUES FOR A CONSTANT SAMPLING RATE OF APPROXIMATELY 34 SECONDS. AVGX IS THE AVERAGE DISTANCE THE RECORD ADVANCES PER ONE HOUR REAL TIME, I.E., THE DATA SAMPLING RATE. AVGX IS APPROXIMATELY 1.1 INCH PER HOUR.

REFERENCE: CALMA COMPANY MODEL 480 DIGITIZER INSTRUCTION MANUAL.

THIS PROGRAM IS DESIGNED TO BE USED ON THE FLEET NUMERICAL WEATHER FACILITY CDC 6500 COMPUTER AND THE CARDS USED MUST PUNCHED ON CARD PUNCH MACHINES AT FNUF.

```

          DIMENSION U(5000),V(5000),N(80),NK(80),IBUFF(3000)
          READ 97,I2)
          FORMAT(IJ,JJ=1,L
          READ 98,AVGX
          FORMAT(F10.0)
          READ 50,A,B,C,D
          FORMAT(4A10)
          READ 51,E,F,G,H
          FORMAT(4A10)
          PRINT 52,A,B,C,D
          FORMAT(IH1,39X,4A10)
          PRINT 53,E,F,G,H
          FORMAT(IH0,37X,4A10)
          DELT=36.0/AVGX
          PRINT 99,DELT
          FORMAT(IH0,2X,25H SAMPLING INTERVAL EQUALS,1X,E15.7,1X,7HSECONDS)
          PRINT 700,JJ
          FORMAT(IH0,9HJJ EQUALS,2X,I2)
          DO 100 I=1,5000
          U(I)=0.0
          V(I)=0.0
          DO 202 I=1,3000
          IBUFF(I)=0
          COUNTX=0.0
          COUNTY=0.0
          NUM=3000
          N=1
          K8=0

```

```

100 CALL LICF (5LRBC01,IBUFF,NUM,NPAR,NECF)
101 IF(NECF) 602,200
102 K=-7
103 KF=0
104 K=K+8
105 KA=K+8
106 KC=0
107 DO 102 KB=K,KA
108 KC=KC+1
109 NK(KC)=IBUFF(KB)
110 IF(NK(KC).EQ.0) KF=1
111 DECODE(80,103,NK) (N(I),I=1,80)
112 FORMAT(80R1)
113 DO 104 I9=1,80
114 IF(N(I9).EQ.508) GO TC 106
115 IF(N(I9).EQ.558) GO TC 107
116 IF(N(I9).EQ.348) GO TC 108
117 IF(N(I9).EQ.478) GO TC 105
118
119 SYMBOL /, (50B), REPRESENTS AN INCREMENT TRAVEL IN THE MINUS X OR Y DIRECTION
120 SYMBOL /, (50B), REPRESENTS AN INCREMENT TRAVEL IN THE MINUS X OR Y
121 DIRECTION, BY THE DIGITIZER. A ZERO INCREMENT TRAVEL IN THE X OR Y
122 SYMBOL 0, (55B), REPRESENTS A ZERO INCREMENT TRAVEL IN THE X OR Y
123 DIRECTION, BY THE DIGITIZER. AN INCREMENT TRAVEL IN THE POSITIVE X OR Y
124 SYMBOL 1, (34B), REPRESENTS AN INCREMENT TRAVEL IN THE POSITIVE X OR Y
125 DIRECTION, BY THE DIGITIZER.
126 SYMBOL *, (47B), IS A FLAG INSERTED IN THE RECORD BY THE PERSON
127 DIGITIZING.
128
129 GO TC 104
130 PRINT 205,M,I9
131 FORMAT(1H0,2I10)
132 IF(M.GT.10) GO TC 113
133 GO TC 104
134 RX=-0.01
135 K8=K8+1
136 GO TC 109
137 RX=0.0
138 K8=K8+1
139 GO TC 109
140 RX=0.01
141 K8=K8+1
142 K3=K8/2
143 K3=2*K3
144 IF(K3.EQ.K8) GO TO 111
145 CCUNTX=CCUNTX+RX
146 IF(CCUNTX.EQ.0.07) GO TC 110
147 GO TC 104

```

CCCCCCCC

```

110 U(M)=CCOUNTX
    GO TO 104
111 COUNTY=CCOUNTY+RX
    IF(COUNTX.EQ.0.07) GO TO 112
112 GO TO 104
    V(M)=COUNTY
    COUNTX=0.0
    M=M+1
    IF(M.GT.5000) GO TO 600
104 CONTINUE
    GO TO 201
113 MAX=M-7

C
C
C
C
TOTAL TIME OF THE RECCRD EQUALS THE NUMBER OF DATA POINTS TIMES THE
SAMPLING INTERVAL, DELT AND MULTIPLIED BY 7.0 BECAUSE THIS PROGRAM IS
TAKING EVERY 0.07 INCH INSTEAD OF EVERY 0.01 INCH.

TIME=((M*DELT)/3600.0)*7.0
PRINT 115,TIME
115 FORMAT(1H,10X,28H TOTAL TIME OF RECORD EQUALS,2X,E15.7,2X,
    *7H HOURS.)
215 PRINT 215,(V(I),I=1,M)
    FORMAT(1H,10X,14F7.2)
213 PUNCH 213,(V(I),I=1,M)
116 FORMAT(14F5.2)
    CONTINUE
    STOP
600 PRINT 601
601 FORMAT(1H1,20X,37H***** U AND V SPACE INADEQUATE *****,)
602 STOP
END

```

THIS PROGRAM PROVIDES GRAPH OF TIDAL FLUCTUATIONS.

CIMENSION TIME(3000), TIDE(3000)

REAL*8 TITLE(12)

REAL LABEL/4H

LINK=NUMBER OF DATA SETS TO BE GRAPHED.

READ(5,10)NK

一
二
三
四
五
六
七
八
九
十

CONTINUE

READ(5,105)

FORMAT(6A8)

NUMBER ONE

MUST BE LESS THAN OR EQUAL TO 144)

READ(5,10)

FORMAT(I6)

41, N, 1, 14

```
READ(5,2,END=3)(
```

FORMAT(1)

CONTINUE

CONTINUE

COMARIGRAM REFERENCE LEVEL OF 13 FEET = 9.8 FT ABOVE MLL WATER.

REF. LINE FOR APRIL = 6.5 FT. OR 3.3 FT. ABOVE MLL WATER

ADJUST ZERO REF. TO MILL WATER REF.

WRITE (6,14)

```

WRITE (9,14)
FORMAT (9X, 'TIME', 7X, 'TIDE', 5X, 'DATA POINT NO.')
```

DO 7 I=1, N

$$TIDE(I) = TIDE(I) + 3.30$$

CONTINUE

INTERVAL = 236.25 SECONDS = 0.0658 HOURS FOR 0.07 INCH

0051-1. N

TIME(1)=C00C

$$TIME(I+1) = I * 0.0658$$

CONTINUE

WRITE(6,6)TIME(I),TIDE(I),I

FORMAT(5X,F10.4,5X,F6.2,5X,I6)

IF (I.E. 720) GO TO 5
IF (I.E. 720) GO TO 5

SET SCALES FOR GRAPHICAL DISPLAY

$$\text{TIME} (I) = \text{TIME} (I) + 8.0$$
[illegible]

CONTINUE

CONTINUE
INVERT GRAPHICAL DISPLAY.

U. S. 1916

```

TIDE(I)=-TIDE(I)
19 CONTINUE
IF (N.LE.720) GO TO 72
M=720
GO TO 73
72 CONTINUE
M=N
73 CONTINUE
CALL DRAW ( M, TIDE( 1), TIME( 1), 1, 0, LABEL, ITITLE, 2., 3., C, 8,
, 2, 2, 9, 15, 1, LAST)
IF (N.LE.720) GO TO 720
M=N-720
CALL DRAW ( M, TIDE( 721), TIME( 721), 3, 0, LABEL, ITITLE, 2., 3., C, 8,
, 2, 2, 9, 15, 1, LAST)
720 CONTINUE
J=J+1
IF (J.LE.NK) GO TO 13
STOP
END
//GO.FTJ06FC01 DD SPACE=(CYL,(30,1))
//GO.SYSIN DD *

```



```

PROGRAM CURRENT
      THIS PROGRAM PROVIDES ELEMENTARY STATISTICS FOR THREE TIME
      SERIES OF CURRENT TEMPERATURE, SPEED, AND DIRECTION. MEANS ARE
      COMPUTED HOURLY, DAILY AND FOR THE ENTIRE SERIES. A HISTOGRAM
      IS PROVIDED DAILY AND FOR THE SERIES. VARIANCE AND STANDARD
      DEVIATION ARE COMPUTED FOR THE SERIES. OPTIONAL GRAPHICAL OUTPUTS
      HISTOGRAMS ARE PROVIDED.

      DIMENSION DATE(3),DATO(3),TIME(3000),T(3000),S(3000),D(3000),
      1M(3000)
      PEAL LABEL/4H /
      IND=NUMBER OF DATA SETS
      IND=1
      DO 1313 III=1,IND
      1313 CLOCK=I*TIME(C)*0.01
      CONTROL CARD INPUT:
      PEAD(5,202)NN,NO,NT,NA,NH,NP,PN,NW,PW,ICOR,IMG,DATE,DATO
      NN=NO. OF DATA POINTS; CALL DRAW MUST BE ADJUSTED
      NO=TOTAL OF NN POINTS(I4)
      NT=DAY OF FIRST OBSERVATION(I2)
      NA=LARGEST NT LESS THAN 2400(I4)
      NH=HOUR FRACTION(IN MIN.) OF FIRST OBS. (I2)
      NP=LATITUDE DEGREES(I2)
      PN=LATITUDE MINUTES (F6.2)
      NW=LONGITUDE DEGREES(I3)
      PW=LONGITUDE MINUTES (F6.2)
      ICOR=DIRECTION CORR. FOR MAGNETIC VAP. OR DEV. (I3)
      IMO=LAST DAY OF MONTH(I3)
      DATE=MONTH OF OBSERVATION(3A4)
      DATO=DATE PLUS ONE MONTH(3A4)
      DEL=3.75
      DEL IS THE SAMPLING INTERVAL IN MINUTES
      TIME(I)=0.
      NODRW=0
      NODRW=1 IF GRAPHS ARE NOT DESIRED
      L=0
      NZ=0
      IDAY=1
      WRITE(6,104)NP,PN,NW,PW,NO,NT,DATE
      WRITE(6,100)

```

```

C      WRITE(6,106)NC,NT,DATE
      INPUT TEMP, SPEED, AND DIRECTION TIME SERIES
      DO 5 I=1,NN,5
      READ(5,200,END=55)(T(I+K),S(I+K),M(I+K),K=0,4)
      CONTINUE
      55 CONTINUE
      C      ADJUST DIRECTION IF REQUIRED
      DO 10 I=1,NN
      M(I)=M(I)+ICOR
      IF(M(I)-0.)42,43,43
      42 M(I)=M(I)+360
      43 CONTINUE
      N=I
      IF(T(I).EQ.1.)GO TO 10
      7 WRITE(6,101)TIME(I),T(I),S(I),M(I),I
      IF(L.LT.16)GO TO 8
      NT=NT+100
      IF(NT.LE.NA)GO TO 56
      NT=0000+NH
      NO=NO+1
      IMN=I-NZ
      AT END OF EACH DAY COMPUTE STATISTICS
      CALL DAY(T,S,M,IMN,I,1,DTAV,DSAV,DDAV)
      IDDAV=IFIX(DDAV)
      WRITE(6,88)IDAY,DTAV,DSAV,IDDAV
      88 FORMAT(5X,'DAY',I2,' STATISTICS:      AVGDAYTEMP=',F6.2,3X,
      1,'AVGDAYSPD=',F5.2,3X,'AVGDAYDIR=',I4,/)
      IDAY=IDAY+1
      87 CONTINUE
      NZ=0
      C      IF MONTH CHANGES REPLACE THE DATE
      55 IF(NO.LE.IMO)GO TO 3
      NO=1
      DO 60 J=1,3
      60 DATE(J)=DATE(J)
      3 CONTINUE
      HR = TIME (I)/60.
      IF(I.GE.18)GO TO 11
      IML=I-L
      GO TO 12
      11 CONTINUE
      IML=I-L+1
      12 CONTINUE
      C      SUBROUTINES FOR HOURLY MEANS
      CALL SUBROUTINE(T,IML,I,AT)
      CALL MEAN(S,IML,I,AS)
      CALL MEAN(M,IML,I,S,AVG)
      MD=IFIX(AVG)

```

```

IF(MD.GE.0)GO TO 59
MD=360+MD
CONTINUE
58 WRITE OUT THE HOURLY MEANS
WRITE(6,103) NC,NT,DATE,HR,AT,AS,MD
L=0
8 TIME(I+1)=TIME(I)+DEL
NZ=NZ+1
L=L+1
1C CONTINUE
N=NN
C COMPUTE THE TIME SERIES MEANS
CALL MEAN(T,1,N,TAT)
CALL MEAN(S,1,N,TAS)
CALL MEAND(M,1,N,S,AVG)
MMD=IFIX(AVG)
IF(MMD.GE.0)GO TO 57
MMD=360+MMD
57 CONTINUE
WRITE(6,107)TAT,TAS,MMD
DO 16 I=1,N
15 D(I)=FLCAT(M(I))
16 CONTINUE
C SUBROUTINE FOR TIME SERIES HISTOGRAM
CALL HISTO(5,16,244,T,NN,1,NODRW)
CALL HISTO(6,1,22,S,NN,1,NODRW)
CALL HISTO(7,360,36,D,NN,1,NODRW)
C CALL POWER SUBROUTINE FOR RMS, VARIANCE, AND STANDARD DEVIATIONS
CALL POWER (DEL,NN,T,S,D,100)
CLOCK=ITIME(0)*2.01-CLOCK
WRITE(6,999)CLOCK
105 FORMAT(6A8)
202 FORMAT(14,12,214,212,F6.2,I3,F6.2,I3,F6.2,2I3,3A4,3A4)
1C4 FORMAT(1H1,1T40,'CURRENT DATA OBTAINED IN MONTEREY CANYON',///,5X,
5,METER F6.2,'N',F6.2,'W',/5X,'ZERO TIME: ',I2,I4,'(LOCAL)',
1I2,I4,///,5X,'TIME IN MINUTES',/5X,'TEMPERATURE IN DEGREES CENTIGRA
2,3A4,///,5X,'CURRENT SPEED IN KNOTS ',/5X,'CURRENT DIRECTION IN DEGR
3DES,///,5X,'TRUE',/5X)
4EES,TRUE,///)
100 FORMAT(15X,'TEMPERATURE',6X,'CURRENT SPEED',
14X,'CURRENT DIRECTION')
106 FORMAT(1H0,5X,I2,I4,3A4)
200 FORMAT(5(F5.1X,F10.2,13X,F4.1,15X,F5.2,14X,I3,5X,I5)
1C1 FORMAT(1H0,I3,I4,3A4,2X,F5.1,'HRS',4X,'MEAN TEMP:',
1C3 F6.2,4X,'MEAN SPEED:',F5.2,2X,'MEAN DIR:',I3,/)
107 FORMAT(1H0,11X,'MEANS FOR THIS RECORD:
1F5.2,2X,'DIRECTION:',I3)

```

```

999 FORMAT(///,T80,'ELAPSED COMPUTING TIME=',F15.7,'SECONDS')
1313 CONTINUE
      STOP
      END

```

```

C
C
      SUBROUTINE MEAN(A,J,M,AVG)
      THIS SUBROUTINE COMPUTES THE ARITHMETIC MEAN OF
      ANY INPUT PARAMETER.
      DIMENSION A(3000)
      AVG=0.0
      SUM=C.0
      KK=M-J+1
      FKK=FLOAT(KK)
      DO 1 K=J,M
      1 SUM=SUM+A(K)
      AVG=SUM/FKK
      RETURN
      END

```

```

C
C
C
C
      SUBROUTINE MEAN(LL,J,M,S,AVG)
      THIS SUBROUTINE COMPUTES THE MEANS OF DIRECTIONS INPUT AS
      COMPASS HEADINGS BY SUMMING SINES AND COSINES, CONVERT
      TO VECTOR WITH SPEED, THEN NORMALIZE AND FIND THE
      ARCTANGENT. THIS PROCEDURE NECESSARY TO CIRCUMVENT THE
      00 TO 360 DISCONTINUITY.
      DIMENSION LL(3000),S(3000)
      SUM1=C.0
      SUM2=C.0
      DO 1 K=J,M
      AK=LL(K)*0.01745329
      IML=M-J+1
      IF (S(K).GT.0.0) GO TO 13
      SPEED NOT AVAILABLE. READ FROM DATA CARDS AS S(K)=0.0
      INSERT FALSE VALUE TO ELIMINATE DIVISION BY ZERO.
      S(K)=0.001
      CONTINUE
      SUM1=SUM1+S(K)*SIN(AK)
      SUM2=SUM2+S(K)*COS(AK)
      CONTINUE
      1 U=SUM1/FLOAT(IML)
      V=SUM2/FLOAT(IML)
      USQ=U**2

```

I ONLY
I ONLY

```

VSQ=V**2
SUMSQ=USQ+VSQ
VAVG=SQRT(SUMSQ)
X=U/V
AVG=ATAN2(U,V)
AVG=AVG/.01745329
RETURN
END

```

```

SUBROUTINE HISTO(A,Z,K,X,L,M,NODRW)
THIS SUBROUTINE OBTAINS HISTOGRAM, PERCENT OF DATA PER
INTERVAL, PROBABILITY DENSITY, AND PROBABILITY DISTRIBUTION.
ENTER WITH: A=LOWER BOUND

```

CCCCCCCC

```

Z=UPPER BOUND
K=NO. INTERVALS (MAX=50)
X=VARIABLE NAME
L=NO. VARIABLES (MAX=3000)
NODRW=1 (IF GRAPH NOT DESIRED)
DIMENSION N(3000),X(3000),P(55),PP(55),D(55),B(3000)
REAL*8 ITITLA(12) /
REAL LABEL/4H /
IF(NODRW.EQ.1)GO TO 12
READ(5,105)(ITITLA(I),I=1,12)
CONTINUE

```

12

```

WRITE(6,101)
LL=0
C=(Z-A)/FLOAT(K)
D(1)=A

```

```

KP1=K+1
KP2=K+2
DO 1 I=2,KP2
IM1=I-1
D(I)=A+IM1*C
SUMP=0.0
DO 5 I=1,KP2
B(I)=C

```

1

5

```

LLL=M+L
DO 2 J=M,LLL
DO 3 I=1,KP2
IP1=I+1
IF(X(J)).LT.D(I))GO TO 3
IF(X(J)).GE.D(I))GO TO 4
IF(X(J)).LT.D(IP1))GO TO 10
GO TO 3
B(I)=B(I)+1.

```

4

10


```

SUM1=SUM1+X(I)
SUM2=SUM2+Y(I)
SUM3=SUM3+Z(I)
2  XBAR=SUM1/N
   YBAR=SUM2/N
   ZBAR=SUM3/N
   WRITE(6,3)XBAR,YBAR,ZBAR
3  FORMAT(10,'SAMPLE MEANS:',//T35,'XBAR= ',F10.3,/,T35,'YBAR= ',
1  F10.3,/,T35,'ZBAR= ',F10.3,///)
   TRANSFORMATICN TO ZERC MEAN VALUE
DO 4  I=1,N
   XM(I)=X(I)-XBAR
4  CONTINUE
   CALL MEAN (XM,1,N,AVG)
   WRITE (6,15)AVG
DO 44 I=1,N
   XM(I)=Y(I)-YBAR
44 CONTINUE
   CALL MEAN (XM,1,N,AVG)
   WRITE (6,15)AVG
DO 444 I=1,N
   XM(I)=Z(I)-ZBAR
444 CONTINUE
   CALL MEAN (XM,1,N,AVG)
   WRITE (6,15)AVG
15  FORMAT(140,F30.10)
   CALCULATION OF THE MEAN SQUARE VALUE
SUM1=0.0
SUM2=0.0
SUM3=0.0
DO 5  I=1,N
   SUM1=SUM1+(X(I)-XBAR)**2
   SUM2=SUM2+(Y(I)-YBAR)**2
   SUM3=SUM3+(Z(I)-ZBAR)**2
5  CONTINUE
   XMS=SUM1/N
   YMS=SUM2/N
   ZMS=SUM3/N
   WRITE(6,6)XMS,YMS,ZMS
6  FORMAT(10,'MEAN SQUARE VALUES:',//T35,'XMSQ= ',F10.3,/,T35,
1  F10.3,/,T35,'ZMSQ= ',F10.3,///)
   CALCULATION OF THE STANDARD DEVIATION
   XTEMP=SUM1/(N-1)
   YTEMP=SUM2/(N-1)
   ZTEMP=SUM3/(N-1)
   SX=SQRT(XTEMP)
   YX=SQRT(YTEMP)
   ZX=SQRT(ZTEMP)

```

```

WRITE(6,7)SX,YX,ZX
FORMAT(10,'STANDARD DEVIATIONS:',/,T35,'XSTDDEV= ',F10.5,/,T35,
1,YSTDDEV= ',F10.5,/,T35,'ZSTDDEV= ',F10.5,///)
RETURN
END

```

```

SUBROUTINE DAY(T,S,M,J,K,NODRW,DTAV,DSAV,DDAV)
THIS SUBROUTINE UTILIZES PREVIOUS SUBROUTINES TO OBTAIN
DAILY STATISTICS.
DIMENSION DH(3000),T(3000),S(3000),M(3000)
ENTER WITH:

```

```

T=TEMPERATURE SERIES
S=SPEED SERIES
M=DIR ECTION SERIES
J=DAY START OBSERVATION NUMBER
K=TOTAL OBSERVATIONS IN THE SERIES
NODRW=1 (IF GRAPH NOT DESIRED)

```

```

CALL MEAN(T,J,K,DTAV)
CALL MEAN(S,J,K,DSAV)
CALL MEAN(M,J,K,DDAV)
IF(DDAV.GE.0)GO TO 2
DDAV=DDAV+360.
2 CONTINUE

```

```

JP1=J+1
KK=K-JP1

```

```

WRITE(6,3)JP1,K,KK
FORMAT(2X,'DAILY HISTO FROM',I4,' TO ',I4,' TOTAL: ',I4,'PTS',/)
DO 1 I=JP1,K

```

```

1 DH(I)=FLCAT(M(I))
CALL HISTO(5.,16.,44,T,KK,J,NODRW)
CALL HISTO(1.,20,S,KK,J,NODRW)
CALL HISTO(0.,360.,36,DH,KK,J,NODRW)
RETURN
END

```

```

//GO.FT06F001 DD SPACE=(CYL,5)
//GO.SYSIN DD *

```

APPENDIX I

本 本
本 本
本 本

✱ ✱
✱ ✱
✱ ✱

APPENDIX J

```

PROGRAM TEMPERATURE DRAW

THIS PROGRAM PROVIDES A TEMPERATURE GRAPH.  THE PROGRAM IS BASICALLY THE
SAME AS PROGRAM COMPOSITE DRAW; HOWEVER, THIS PROGRAM DRAWS TEMPERATURE TO
AN EXPANDED SCALE.

DIMENSION T(800),TIME(800),S(800),M(800),A(800),B(800),C(800)
REAL*8 ITITLE(12)
REAL LABEL /4H
ND=NUMBER OF DATA SETS
READ(5,20)ND
WRITE(6,22)ND
FORMAT(//,5X,I4,' DATA SETS')
DO 999 J=1,ND
WRITE(6,8)J
FORMAT(//,5X,'DATA SET NUMBER',I4)
PROVIDE GRAPH TITLE.  TITLE IS 2 CARDS WITH NO MORE THAN 48
CHARACTERS EACH (648)
READ(5,105)(ITITLE(I),I=1,12)
FORMAT(648)
NN=NUMBER OF DATA POINTS
NN MUST NOT EXCEED 720
READ(5,20)NN
FORMAT(16)
WRITE(6,1) NN
FORMAT(//,5X,'NN=',I4)
DEL IS THE SWAPING INTERVAL IN HOURS (3.75 MIN.= .0625 HOURS)
DEL=.0625
WRITE(6,2) DEL
FORMAT(5X,'DEL=',F7.4)
TIME(1)=0
NODRW=0
WRITE(6,4) NODRW
FORMAT(5X,'NODRW=',I4)
DO 5 I=1,NN
READ(5,200)(T(I+K),M(I+K),M(I+K),K=0,4)
FORMAT(5(F5.1,F5.2,I4))
CONTINUE
WRITE(6,6)
FORMAT(//,5X,'DATA INPUT COMPLETED')
SET SCALES FOR GRAPHICAL DISPLAY
DO 10 I=1,NN

```



```

TIME(I+1)=I*DEL
C(I)=T(I)
A(I)=(FLOAT(M(I))/24.)
B(I)=-((S(I)*15.)
10 CONTINUE
20 WRITE(6,7) 5X,'DATA CONVERTED FOR GRAPHICAL DISPLAY')
7 FORMAT(//,NN,TIME,C,O,LABEL,I,TITLE,6.1,O,O,2,2,9,8,C,LAST)
C CALL DRAW STATEMENTS FOR SPEED, B, AND DIRECTION, A, HAVE BEEN DELETED IN
C ORDER TO DRAW ONLY TEMPERATURE TO AN EXPANDED SCALE
999 CONTINUE
RETURN
END
//GO.SYSIN DD *

```


FOURIER TRANSFORM OF TIME SERIES DATA SUPPLIED THROUGH
'OCEAN' SUBROUTINES USING P-K FORT FAST FOURIER TRANSFORM
SUBROUTINE (SHARE SDA3465) IN THE IBM SYSTEM 360 MODEL 67.

LAST REVISION JANUARY 24, 1969

JOHN GARRETT

NBLOCK BLOCKS OF 2**NPOW SAMPLES EACH ARE READ FOR KCHAN OF THE
NCHAN CHANNELS AVAILABLE TO THE 'OCEAN' SUBROUTINES. THE CONTENTS OF
ANY CHANNEL MAY BE REPLACED BY A LINEAR COMBINATION OF ITSELF WITH ANY
OTHER CHANNEL.

FOR EACH BLOCK 2**NPOW-1 COMPLEX FOURIER COEFFICIENTS ARE
COMPUTED FOR EACH OF THE KCHAN CHANNELS. THESE ARE THEN WRITTEN ON THE
OUTPUT (TAPE) 03 IN THE FORMAT DESCRIBED BELOW. IN ADDITION THE
COEFFICIENTS MAY BE SUMMED IN GROUPS OF (2**NPOW-1)/32 AND
PRINTED OUT.

ADDITIONAL OPTIONS ARE DESCRIBED UNDER THE RELEVANT CONTROL
PARAMETERS BELOW.

IT SHOULD BE NOTED THAT THE COEFFICIENTS PRODUCED ARE THOSE OF THE
FOURIER SERIES

$$Y(J) = \sum_{K=0}^{N-1} \text{OVER } K = 0, N/2 \text{ OF REAL PARTS OF} \\ (C(K) * \exp((2 * \pi * I / N) * J * K))$$

$$\text{WITH } J = 0, N-1, Y(J) \text{ REAL, AND } I = \sqrt{-1}$$

THE FOLLOWING SUBROUTINES ARE REQUIRED

OCEAN1, OCEAN2, OCEAN3, RWUNLD

SKPFL

CONVCL

USCRMB

P-K FORT

THE FOLLOWING LOGICAL INPUT/OUTPUT DEVICES ARE USED IN THIS PROGRAM
2 = SCRATCH TAPE FOR TEMPORARY STORAGE OF COEFFICIENTS IF

LOFR=1 BELOW

3 = OUTPUT (TAPE) FOR COEFFICIENTS

5 = (CARD) INPUT FOR CONTROL PARAMETERS

6 = PRINTED OUTPUT

INUNIT = INPUT TAPE OF TIME SERIES DATA FOR 'OCEAN' SUBROUTINES

INPUT INFORMATION REQUIRED

FIRST DATA CARD IN COLUMN NUMBER

1-9 ICUSER = USER IDENTIFICATION NUMBER (9-DIGIT INTEGER)

14-15 NCHAN = NUMBER OF CHANNELS DIGITIZED ON OCEAN TAPE

25 NTYPE = (NOT RELEVANT. SET TO ZERO.)

34-35 INFILE = FILE NUMBER OF DATA ON OCEAN TAPE

```

44-45 INUNIT = NUMBER OF UNIT ON WHICH INPUT TAPE IS MOUNTED
55 NSEARH = (NOT RELEVANT. SET TO ZERO.)
61-70 SAMFRQ = SAMPLING FREQUENCY OF DIGITIZING (SAMPLES/SECOND)
(MUST INCLUDE A DECIMAL POINT)
SECOND DATA CARD
4-5 NBLOCK = NUMBER OF BLOCKS DESIRED
14-15 NPOW- MAXIMUM NUMBERS OF SAMPLES PER BLOCK WILL BE 2**NPOW,
MAX NPOW IS 13 (8192 SAMPLES/BLOCK) BUT NPOW WILL BE
REDUCED UNTIL (2**NPOW)*KCHAN WILL FIT IN MEMORY. FOR
KCHAN OF 10 THIS WILL GIVE NPOW = 10 (1024 SAMPLES/BLOCK).
24-25 MTAPE =+1 FOR OUTPUT TAPE TO BE WRITTEN
= -1 FOR NO OUTPUT TAPE
34-35 NFILE - OUTPUT COEFFICIENTS WILL BE NFILE-TH FILE ON TAPE
43-45 MAXERR = (NOT RELEVANT. SET TO ZERO.)
54-55 MPRINT = -1 SUPPRESSES SUMMARY COEFFICIENTS PRINT OUT
= 0 OR GREATER PERMITS PRINT OUT
THIRD DATA CARD
4-5 KCHAN = NUMBER OF CHANNELS TO BE TRANSFORMED (MAX 10)
15 LOFR = 2 IF COEFFICIENTS TO BE COMPUTED FROM DATA SMOOTHED
AND SUBSAMPLED (DECIMATED) USING CONVOL SUBROUTINE.
WEIGHTS USED IN SMOOTHING AND DECIMATING FACTOR ARE
DETERMINED BY CHOICE OF CONVOL USED.
= 1 IF ALTERNATE BLOCKS TO BE MADE UP OF SAMPLES FROM DATA
SMOOTHED AND DECIMATED USING CONVOL SUBROUTINE.
COEFFICIENTS WILL APPEAR ON OUTPUT TAPE IN FILE
IMMEDIATELY FOLLOWING THAT CONTAINING RESULTS FROM
UNSMOOTHED DATA
25 IHANN = 0 IF DATA TO BE LEFT ALONE
= 1 IF FOURIER COEFFICIENTS TO BE HANNED AND
NORMALIZED (*SQRT(8/3))
= 0 IF NOT

```

```

NEXT KCHAN CARDS
1-5 NO. OF PRIMARY A TO D CHANNEL
6-10 NO. OF SECONDARY A TO D CHANNEL
11-20 CALIBRATION ASSOCIATED WITH THE PRIMARY A TO D CHANNEL
21-30 CALIBRATION ASSOCIATED WITH THE SECONDARY A TO D CHANNEL
31-66 ALPHAMERIC NAME OF RESULTING PRIMARY CHANNEL
71-78 8 CHARACTER NAME OF UNITS FOR RESULTING PRIMARY CHANNEL

```

THE DATA TRANSFORMED AS THE PRIMARY CHANNEL IS THEN

```

CALIBRATION1 X VALUE OF PRIMARY + CALIBRATION2 X VALUE OF
SECONDARY CHANNEL

```

A SECOND SET OF DATA CARDS WILL PRODUCE A SECOND ANALYSIS. A BLANK
CARD TERMINATES THE RUN.

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

```

FORMAT OF OUTPUT TAPE FOR EACH BLOCK TRANSFORMED IS AS FOLLOWS
FIRST LOGICAL RECORD IS AN ARRAY OF 256 WORDS CALLED
ARTAPE(4) = NTAP (SEE BELOW)
ARTAPE(1) = IDUSER
ARTAPE
ARTAPE(3) = NUMBER OF SAMPLES /BLOCK
ARTAPE(2) = BLOCK NUMBER
ARTAPE(10) = IHANN
ARTAPE(5) = NUMBER 6F CHANNELS TRANSFORMED
ARTAPE(6) = SAMPLING FREQUENCY
ARTAPE(10+K) = MCHAN(K)
ARTAPE(31+9(K-1)) = ACHNAM(K) (9A4)
ARTAPE(121+2(K-1)) = AUNITS(1,K) (2A4)
ARTAPE(140+K) = CAL(K)

NEXT NTAP LOGICAL RECORDS OF 256 WORDS EACH CALLED
TAPRAY AND CONSIST OF
1 WORD CONTAINING INTEGER HARMONIC NO (0 TO IBLOCK/2) FOLLOWED BY
2*KCHAN WORDS CONTAINING FOR EACH OF THE KCHAN CHANNELS THE
REAL PART OF THE FOURIER COEFF AT THIS HARMONIC(FIRST WORD)
FOLLOWED BY THE IMAGINARY PART(SECOND WORD).
ONLY COMPLETE SEQUENCES OF 1+(2*KCHAN) WORDS ARE INCLUDED IN A
TAPRAY SO THAT THE LAST FEW WORDS MAY CONTAIN ZEROS

```

AN END OF FILE IS WRITTEN ON THE OUTPUT TAPE AT THE END OF A SEQUENCE OF BLOCKS

J.F. GARRETT
J.R. WILSON

INSTITUTE OF OCEANOGRAPHY
UNIVERSITY OF BRITISH COLUMBIA
VANCOUVER 8, CANADA

```

DIMENSION W(10242), S(4100), ARTAPE(256), TAPRAY(256), MCHAN(10),
1 CAL(10), AUNITS(2,10), ACHNAM(9,10), INDCH(12), I2TAB(14), INCR(10),
2 INDATA(12), W1(1024), W2(1024), W3(1024), W4(1024), W5(1024),
3 W6(1024), W7(1024), W8(1024), W9(1024), W10(1024), A(256), A1A(12)
4 , VAL(50,10)
5 DIMENSION MCHAN2(10), CAL2(10), DDFREQ(20)
COMMON W, S, NAN, KOUNT
EQUIVALENCE (W(1), W1(1)), (W(1025), W2(1)), (W(2049), W3(1)),
1 (W(3073), W4(1)), (W(4097), W5(1)), (W(5121), W6(1)), (W(6145), W7(1)),
2 (W(7169), W8(1)), (W(8193), W9(1)), (W(9217), W10(1)), (ARTAPE(1), A(1))
3, (A(1), IDUSER), (A(2), MBLOCK), (A(3), IBLOCK), (A(4), NTAP),
4 (A(5), KCHAN), (A(6), SAMFRQ), (A(11), MCHAN(1)),
5 (A(31), ACHNAM(1,1)), (A(121), AUNITS(1,1)), (A(141), CAL(1)),
6 (A(151), INDCH(1)), (A(163), I2TAB(1)), (A(180), INCR(1)),
7 (A(190), INDATA(1)), (A(210), TAPRAY(1)), (A(210), IAR),
8 (A(190), A1A(1))

```



```

NZBLK = 0
CORNB = 1./SQRT(6.0)
CORNA = 2.0 * CORNB
IOFPS = 0
6000 READ (5,5,END=6010) IDUSER,NCHAN,NTYPE,INFILE,INUNIT,NSEARH,SAMFRQ
      IF (INFILE.EQ.0) GO TO 6010
      DFREQ = SAMFRQ
      DO 10010 I=1,10
      MCHAN(I) = 0
      MCHAN2(I) = 0
      CAL(I) = 0
      CAL2(I) = 0
      DO 10009 J=1,9
10009 ACHNAM(J,I) = 0
      DO 10010 J=1,2
10010 AUNIT(J,I) = 0
      CONTINUE
10012 DO 10012 I=1,12
      INDCH(I) = 0
      READ (5,6) NBLOCK,NPOW,MTAPE,KFILE,MAXERR,MPRINT,IRWCTL
      IF (IOFPS.EQ.0) NFILE = KFILE
      READ(5,6) KCHAN,LOFR,IHANN
      ARTAPE(10) = IHANN
      IF (LOFR.EQ.1) REWIND 2
      DO 8 K=1,KCHAN
8      KCH = K
      READ (5,7) MCHAN(K),MCHAN2(K),CAL(K),CAL2(K),(ACHNAM(L,K),L=1,9),
1(AUNIT(L,K),L=1,2)
      IF (MCHAN2(K).EQ.0) MCHAN2(K) = 1
      IF (MCHAN(K).EQ.0) GO TO 9
      KCH = MCHAN(K)
      INDCH(KCH) = K
      KCH = K
9      KCHAN = KCH
      C SET UP TWOS TABLE
      DO 10 N=1,14
10      I2TAB(N) = 2**N
      C DETERMINE BLOCK SIZE
11      LOKTES = KCHAN*I2TAB(NPOW)
      IF (LOKTES.LE.10242) GO TO 15
      NPOW = NPOW-1
      GO TO 11
15      IBLOCK = I2TAB(NPOW)
      KBLOCK = IBLOCK/2
      KCH = 256/(1+(2*KCHAN))
      NTAP = (KBLOCK + KCH -1)/KCH
      C COMPUTE SINE TABLE
      CALL FORT(W,NPOW,S,0,IFERR)
      C POSITION OUTPUT TAPE

```

```

IF (MTAPE.LT.0) GO TO 31
IF (IOFPOS.NE.0) GO TO 31
REWIND 3
IOFPOS = 1
IF (KFILE.GT.1) CALL SKPFL (KFILE-1,3)
C SET 31 CALL OCEAN1 (NCHAN,INUNIT,MAXERR,INFILE,NTYPE,NSEARH)
WRITE(6,35) IDUSER,NCHAN,KCHAN,NBLOCK,IBLOCK,SAMFRQ
IF (IHANN.GT.0) WRITE(6,36)
IF (LOFR.GT.0) CALL CONVOL(VAL,MXWAY,LDEC,1,SUM)
DECIM = LDEC
LINC = IBLOCK
IF (IBLOCK.LT.1024) LINC = 1024
C SET 40 UP STORAGE REFERENCES
DO 40 K = 1,KCHAN
  KCH = K - 1
  INCR(K) = LINC * KCH
  MINCR = LINC
  LOBLO = LOFR
  MBLOCK = 1
C READ 12 SOME DATA
  NPAR = 0
  IADD = 0
  N111 = 0
  7000 NOLD = 1
  SAMFRQ = DFREQ
  IF (LOBLO.GT.0) SAMFRQ = DFREQ/DECIM
  DO 289 IB = 1,IBLOCK
    IF (LOBLO) 49,49,101
  DO 150 LSAMP=1,LDEC
    CALL OCEAN2(NIND,KBLOCK,INDATA(1))
    NIND = NIND+1
    GO TO (140,70,116,105,140),NIND
  N111 = N111+1
  KCH = INDATA(1)/NCHAN
  ICH = INDATA(1)-(KCH*NCHAN)
  IF (ICH) 51,106,51
  DO 108 IC = 1,KCH
    DO 107 K = 1,KCHAN
      VAL(NOLD,K) = 512.
    NOLD = NOLD+1
  CONTINUE
  GO TO 150
  116 NPAR = NPAR + 1
  NZBLOCK = KBLOCK
  IF (NPAR.GT.MAXERR) GO TO 5000
  GO TO 7000

```

```

140 DO 145 K = 1,KCHAN
    IDSUB = MCHAN(K)
    IDSUB2 = MCHAN2(K)
    IF (NIND.NE.5) GO TO 7020
    DATA = ATA(IDSUB)*CAL(K) + ATA(IDSUB2)*CAL2(K)
    GO TO 7030
7020 DATA = (FLOAT(INDATA(IDSUB))/100. - 5.11)*CAL(K)
7030 VAL(NOLD,K) = DATA
145 CONTINUE
    NOLD = NOLD+1
    IF (NOLD.GT.MXWAY) NOLD = 1
150 CONTINUE
    IF (IADD.EQ.0.AND.NOLD.GT.1) GO TO 101
    DO 160 K = 1,KCHAN
        IDSUB = MCHAN(K)
        CALL CCNVCL(VAL,NOLD,K,2,SUM)
        ATA(IDSUB) = SUM
160 CONTINUE
        NIND = 6
        GO TO 100
49 CALL OCEAN2(NIND,KBLOCK,INDATA(1))
C NIND = 1 MEANS END OF FILE, 2 MEANS PARITY ERROR, 3 MEANS ONES SEARCH
C NIND = 4 MEANS OCEAN2 WILL RETURN A FLOATING POINT NUMBER BETWEEN
C 0. AND 1023.
        NIND = NIND + 1
        GO TO (100,70,60,50,100), NIND
50 N111 = N111 + 1
        KCH = INDATA(1)/NCHAN
        ICH = INDATA(1) - (KCH*NCHAN)
        IF (ICH) 51,56,51
51 WRITE(6,55) MBLOCK,KBLOCK,IADD
        GO TO 5000
56 DO 58 IC = 1,KCH
        IADD = IADD + 1
        DO 58 K = 1,KCHAN
        IWSUB = INCR(K) + IADD
        W(IWSUB) = 0.0
58 GO TO 100
        60 IF (NZBLOCK.EQ.KBLOCK) GO TO 7000
        NPAR = NPAR + 1
        NZBLOCK = KBLOCK
        IF (NPAR.GT.MAXERR) GO TO 5000
        GO TO 7000
70 WRITE(6,75) MBLOCK,KBLOCK
        GO TO 5000
C STORE DATA IN PSEUDO 2-D ARRAY
100 IADD = IADD + 1

```



```

DO 200 K = 1, KCHAN
  IDSUB = MCHAN(K)
  IDSUB2 = MCHAN2(K)
  IWSUB = INCR(K)+IADD
  IF (NIND.LT.5) GO TO 7040
  IF (NIND.EQ.5) DATA = ATA(IDSUB)*CAL(K) + ATA(IDSUB2)*CAL2(K)
  IF (NIND.EQ.6) DATA = ATA(IDSUB)
  GO TO 7050
7040 DATA = (FLOAT(INDATA(IDSUB))/100. - 5.11)*CAL(K)
7050 W(IWSUB) = DATA
200 CONTINUE
  IF (IADD.GE.IBLOCK) GO TO 299
  GET FOURIER TRANSFORM
  MPOW = NPOW - 1
  CALL FORT(W1,MPOW,S,-2,IFERR)
  CALL USCRMB(W1,IBLOCK,S)
  IF (KCHAN.LE.1) GO TO 340
  KCH = MINCR/1024
  GO TO (320,300,300,301),KCH
300 KCH = KCHAN - 1
  GO TO (310,309,308,307), KCH
301 CALL FORT(W5,MPOW,S,-2,IFERR)
  CALL USCRMB(W5,IBLOCK,S)
  GO TO 340
  KCH = KCHAN/2
  GO TO (306,305,304,303,302), KCH
302 CALL FORT(W10,MPOW,S,-2,IFERR)
  CALL USCRMB(W10,IBLOCK,S)
303 CALL FORT(W8,MPOW,S,-2,IFERR)
  CALL USCRMB(W8,IBLOCK,S)
304 CALL FORT(W6,MPOW,S,-2,IFERR)
  CALL USCRMB(W6,IBLOCK,S)
305 CALL FORT(W4,MPOW,S,-2,IFERR)
  CALL USCRMB(W4,IBLOCK,S)
306 CALL FORT(W2,MPOW,S,-2,IFERR)
  CALL USCRMB(W2,IBLOCK,S)
  KCH = (KCHAN+1)/2
  GO TO (340,310,309,308,307),KCH
330 KCH = (KCHAN-1)/2
  GO TO (310,309,308,307), KCH
307 CALL FORT(W9,MPOW,S,-2,IFERR)
  CALL USCRMB(W9,IBLOCK,S)
308 CALL FORT(W7,MPOW,S,-2,IFERR)
  CALL USCRMB(W7,IBLOCK,S)
309 CALL FORT(W5,MPOW,S,-2,IFERR)
  CALL USCRMB(W5,IBLOCK,S)

```

```

310 CALL FORT(W3,MPOW,S,-2,IFERR)
CALL USCRMB(W3,IBLOCK,S)
340 IF (IFERR) 350,500,350
350 WRITE(6,355) IFERR
C COEFFICIENTS NOW UNSCRAMBLED,NOW BEGIN SUM AND PRINT
500 IF(IHANN.LE.0) GO TO 880
DO 800 K = 1,KCHAN
ISUB = INCR(K) + 1
ISTAR = INCR(K) + 5
ILSUB = INCR(K) + IBLOCK - 2
KLR = ISUB + 2
KLM = ISUB + 3
AR3 = 0.0
AR2 = W(ISUB)
AR1 = W(KLR)
AM3 = 0.0
AM2 = 0.0
AM1 = W(KLM)
W(ISUB) = CORNA*(AR2-AR1)
DO 700 KLP = ISTAR,ILSUB,2
KM = KLP+1
AR3 = AR2
AR2 = AR1
AR1 = W(KLP)
AM3 = AM2
AM2 = AM1
AM1 = W(KM)
W(KLR) = CORNB*((2.*AR2)-AR3-AR1)
W(KLM) = CORNB*((2.*AM2)-AM3-AM1)
KLR = KLP
KLM = KM
CONTINUE
ISUB = INCR(K)+2
AR3 = AR2
AR2 = AR1
AR1 = W(ISUB)
W(KLR) = CORNB*((2.*AR2)-AR3-AR1)
W(ISUB) = CCORNA*(AR1-AR2)
AM3 = AM2
AM2 = AM1
AM1 = 0.
W(KLM) = CORNB*((2.*AR2)-AR3-AR1)
CONTINUE
IF(MPRINT.LT.0) GO TO 1300
KCH = NPOW-6
IF (KCH.LE.0) KCH = 1
ISUM = I2TAB(KCH)
KFIRST = 1

```

```

900      KLAST = 5
      IF(KCHAN.LT.KLAST)KLAST = KCHAN
      WRITE(6,905)IDUSER,MBLOCK,IBLOCK,TODAY,NPAR,N111
      IF(IHANN.GT.0) WRITE(6,36)
      DO 920 K = KFIRST,KLAST
      WRITE(6,915) K,MCHAN(K),(ACHNAM(L,K),L= 1,9),(AUNITS(L,K),L=1,2),CAL(K)
      CONTINUE
      WRITE(6,925) (MCHAN(K),K=KFIRST,KLAST)
      WRITE (6,935)
      SU = 1
      IHAR = -1
      DO 1200 ICO = 1,33
      DO 950 K= KFIRST,KLAST
      KL = 2*K
      TAPRAY(KL) = 0.0
      TAPRAY(KL+1) = 0.0
      DO 990 IS = 1,ISUM
      IHAR = IHAR+1
      IADD = 2*IHAR
      DO 980 K=KFIRST,KLAST
      KL = 2*K
      LINCR = INCR(K)
      IISTOR = LINCR+IADD
      IF(IHAR.EQ.(IBLOCK/2)) GO TO 970
      TAPRAY(KL) = TAPRAY(KL) + W(IISTOR +1)
      IF (IHAR.EQ.0) GO TO 980
      TAPRAY(KL+1) = TAPRAY(KL+1)+W(IISTOR+2)
      GO TO 980
      TAPRAY(KL) = TAPRAY(KL)+W(LINCR+2)
      CONTINUE
      IF (IHAR.EQ.0) GO TO 1000
      CONTINUE
      SU = ISUM
      KONE = 2*KFIRST
      KTWO = (2*KLAST)+1
      DO 1010 KL = KONE,KTWO
      TAPRAY(KL) = TAPRAY(KL)/SU
      WRITE(6,1105) IHAR,(TAPRAY(KL),KL=KONE,KTWO)
      CONTINUE
      IF (KLAST.GE.KCHAN) GO TO 1300
      KFIRST = KLAST + 1
      KLAST = KFIRST + 4
      GO TO 900
      C END OF SUM AND PRINT LOOP, NOW WRITE OUT PUT TAPE
1300 IF(MTAPE)1350,1301,1301
1301 IF(LOBLO-1) 1302,1303,1302
      WRITE (3) ARTAPE
      GO TO 1304

```

```

1303 WRITE (3) ARTAPE
1304 CONTINUE
C SET UP 256 WORD TAPRAY
IHARM=(IBLOCK/2)-1
MAXTAP = 256-(2*KCHAN)
C ZERO TH HARMONIC IS A SPECIAL CASE
IHAR = 1
IAR = 0
DO 1310 K = 1, KCHAN
ISUB = INCR(K)
IHAR = IHAR+1
TAPRAY(IHAR) = W(ISUB+1)
IHAR = IHAR+1
TAPRAY(IHAR) = 0.0
DO 1330 IH = 1, IHARM
IHH = 2*IH
IHAR = IHAR+1
TAPRAY(IHAR) = IH
DO 1320 K=1, KCHAN
ISUB = IHH + INCR(K)
TAPRAY(IHAR+1)=W(ISUB+1)
TAPRAY(IHAR+2)=W(ISUB+2)
IHAR = IHAR+2
1310 CONTINUE
IF(IHAR.LT.MAXTAP)GO TO 1330
IHAR=0
IF (LOBLO-1) 1322,1323,1322
1322 WRITE (3) TAPRAY
1323 GO TO 1330
1330 CONTINUE
C (IBLOCK/2)TH HARMONIC IS ALSO A SPECIAL CASE
IHAR = IHAR+1
TAPRAY(IHAR) = IHARM+1
DO 1340 K=1, KCHAN
ISUB = INCR(K)
TAPRAY(IHAR+1) = W(ISUB +2)
TAPRAY(IHAR+2) = 0.0
IHAR = IHAR+2
1340 IF (LOBLO-1) 1341,1342,1341
1341 WRITE (3) TAPRAY
WRITE(6,1355) MBLOCK,NFILE
GO TO 1343
1342 WRITE (3) TAPRAY
WRITE(6,13425) MBLOCK,IDUSER
CONTINUE
1343 IF(LOBLO.NE.1) MBLOCK = MBLOCK+1
1350 IF(MBLOCK.GT.NBLOCK) GO TO 5000

```

```

1351 IF(LOFR.EQ.0) GO TO 12
      IF(LOBLO-1)1351,1352,12
1352 LOBLO = 1
      GO TO 12
1352 LOBLC = 0
      GO TO 12
C WRAP UP ROUTINES
5000 IF(MTAPE.LT.0) GO TO 5500
      END FILE 3
      IF(LOFR.NE.1) GO TO 5500
      END FILE 2
      REWIND 2
      IDTEST = IDUSER
5200 READ (3,END=5300) ARTAPE
      WRITE (3) ARTAPE
      GO TO 5200
5300 END FILE 3
      NFILE = NFILE + 1
      IDUSER = IDTEST
      WRITE(6,5305) IDUSER,NFILE
5500 CONTINUE
      IF (MTAPE.GE.0) NFILE = NFILE + 1
      WRITE (6,8001)
      WRITE (6,8002)
      CALL OCEAN3
      GO TO 6000
6010 CALL RWUNLD
      IF (MTAPE.LT.0) GO TO 6020
      IF (IRWCTL.NE.0) GO TO 6030
      REWIND 3
6020 WRITE (6,5005)
      CALL EXIT
6030 REWIND 3
      GO TO 6020
6070 WRITE(6,6075) MBLOCK,NFILE,LOBLO
6075 FORMAT(5X,30H END OF OUTPUT TAPE, MBLOCK = ,15,9H NFILE = ,
      1 15,9H LOBLO = ,15/5X,21HPROCESSING TERMINATED )
      GO TO 6010
      5 FORMAT (19,1X,5(15,5X),F10.0)
      6 FORMAT (7(15,5X))
      7 FORMAT (215,2F10.0,9A4,4X,2A4)
35 8 FORMAT(46H1 FOURIER TRANSFORM OF OCEAN TIME SERIES DATA /14H USE
      1R NUMBER,11/16H OCEAN TAPE HAS,15,1X,18HCHANNELS, OF WHICH, /
      2 15,16H WILL BE DONE IN,15,10H BLOCKS OF,16,13H SAMPLES EACH, /
      3 39H THE FREQUENCY OF DIGITAL SAMPLING WAS, F10.2,9H SAMP/SEC/)
36 4 FORMAT(64H0 FOURIER COEFFICIENTS TO BE HANNED AND NCRMALIZED (*SQR
      1T(8/3))
55 5 FORMAT(32H CHANNEL IDENTITY LOST IN BLOCK ,15, 7H,RECORD,15,

```



```

75 1 6H,SWEEP,I7 )
355 FORMAT(36H ) END OF FILE ON INPUT TAPE IN BLOCK,I5,7H,RECORD,I5)
905 FORMAT(15H TOTAL IFERR=,I10)
1 17,8H SAMPLES,10X, /5X,37HIN THIS BLOCK PARITY ERRORS ON TAPE
2= ,I3,32H,ALL ONES,NOT FOUND IN CHANNEL 1,I4,6H,TIMES //16X }
3 1HK,4X,7HCHANNEL,17X,4HNAME,25X,5HUNITS,3X,2H,AL.FACTOR }
915 FORMAT( 15X,I3,4X,I3, 5X,9A4,8X, 2A4,5X,1PE9.2,
925 FORMAT( 2X,4HLAST,1X, 5(8X,8HCHANNEL,12,4X))
935 FORMAT( 2X,4HHARM,8X, 5(4HREAL,6X,4HIMAG,8X))
1105 FORMAT( 2X,I4,1X,5(3X,1PE9.2,1X,1PE9.2))
13425 FORMAT(24H COEFFICIENTS FOR BLOCK ,I5,36H OF SMOOTHED AND DECIMA
1355 1TED DATA FOR ,I9,14H WRITTEN ON 02 )
1 15H ON OUTPUT TAPE )
5005 FORMAT(13HONORMAL EXIT )
80 1 15H ON OUTPUT TAPE )
5305 FORMAT(20(1X,F5.2))
8001 10PIED FROM 02 ONTO FILE ,I2,14H OF TAPE ON 03 //
8002 10PIED FROM 02 ONTO FILE ,I2,14H OF TAPE ON 03 //
1 15H ON OUTPUT TAPE )
END

```

```

10 SUBROUTINE CONVOL(VAL,NOLD,KCHAN,MCODE,SUM)
DIMENSION VAL(50,1),WEIGHT(50)
IF(MODE-1) 10,10,20
LDEC = 10
MXWAY = 21
WAY = 1.0/21.0
DO 11 KM = 1,21
WEIGHT(KM) = WAY
KCHAN = LDEC
NOLD = MXWAY
WRITE(6,5) LDEC,MXWAY,(WEIGHT(K),K=1,MXWAY)
RETURN
SUM = 0.
NCOUNT = 0
NFIR = NOLD-1
NFIR = NFIR+1
NCOUNT = NCOUNT +1
IF(NCOUNT-MXWAY)17,17,21
IF(NFIR-MXWAY)40,40,18
NFIR = 1
SUM = SUM+VAL(NFIR,KCHAN)
GO TO 30

```



```

21
5
SUM = SUM*WEIGHT(1)
RETURN
FORMAT(50H1 ALTERNATE BLOCKS OF DATA DECIMATED BY FACTOR OF , I3/
1 2X,22H AFTER SMOOTHING WITH ,I5,35H WEIGHTS WITH THE FOLLOWING VA
2LUES, / (10X,1PE14.5))
END

SUBROUTINE USCRMB (C,M,S)
DIMENSION C(1),S(1)
N = M
ST = C(1)
C(1) = 0.5*(C(1) + C(2))
C(2) = 0.5*(ST - C(2))
K = N/2 - 1
MSIN = N/4
C(K+3) = -C(K+3)
DO 10 I=3,K,2
IS = (I-1)/2
IC = MSIN - IS
ST = S(IC)
AL = C(I)
B1 = C(I+1)
L = N - I
A2 = C(L+2)
B2 = C(L+3)
C(I) = 0.5*(A1+A2+(B1+B2)*CT-(A1-A2)*ST)
C(I+1) = 0.5*(B1-B2-(B1+B2)*ST-(A1-A2)*CT)
C(L+2) = 0.5*(A1+A2 -(B1+B2)*CT+(A1-A2)*ST)
C(L+3) = 0.5*(B2-B1-(B1+B2)*ST-(A1-A2)*CT)
10 RETURN
END

FORTRAN OCEAN PACKAGE

THE FORTRAN OCEAN PACKAGE IS DESIGNED TO DUPLICATE THE
'OCEAN' AND 'OCEANB' SUBROUTINE PACKAGES WITH A TAPE WHICH WAS
EITHER CREATED BY THE DIGITAL FILTERING SYSTEM OR IS IN THE SAME
FORMAT. THE CALLS ARE EQUIVALENT.

THE CALLING SEQUENCES ARE:

CALL OCEAN1 (KCHAN,NUNIT,MAXERR,NFILE,NTYPE,NSEARCH)

WHERE,

```

```

NCHAN = NO. OF CHANNELS DIGITIZED ON THE FORTRAN OCEAN TAPE
NUNIT = NO. OF UNIT ON WHICH THE FORTRAN OCEAN TAPE IS
MOUNTED
MAXERR (THIS PARAMETER IS NOT APPLICABLE TO FORTRAN OCEAN
BUT MUST BE INCLUDED FOR COMPATIBILITY)
NFILE = NO. OF DESIRED FILE ON THE FORTRAN OCEAN TAPE
NTYPE (NOT APPLICABLE BUT IS INCLUDED FOR COMPATIBILITY)
NSEARH (NOT APPLICABLE BUT IS INCLUDED FOR COMPATIBILITY)

CALL OCEAN2 (IND,KBLOCK,N)
WHERE,
    IND = 0 FOR NORMAL RETURN (RETURNED)
        = 1 FOR END OF INPUT FILE (RETURNED)
        = 4 IF THE VECTOR RETURNED IS IN FLOATING POINT (FORTRAN
          VERSION OF OCEAN ONLY) (RETURNED)
    KBLOCK = THE SEQUENTIAL NUMBER OF THE PHYSICAL TAPE BLOCK OF
              THE NEXT DATA WHICH WILL BE RETURNED BY THE NEXT OCEAN2 CALL.
              (RETURNED)
    N = FIRST LOCATION OF THE ARRAY INTO WHICH THE CROSS CHANNEL
        SEQUENCE IS TO BE RETURNED

CALL OCEAN3
THIS CALL RESULTS IN A 1 LINE SUMMARY BEING LISTED ON LOGICAL
UNIT 6. THIS MAY BE CALLED AT ANY TIME.

```

CC

```

SUBROUTINE OCEAN1 (NCHAN,NUNIT,MAXERR,NFILE,NTYPE,NSEARH)
DATA KFILE/0/
KUNIT = NUNIT
IF (KFILE.GT.0) GO TO 20
10 REWIND KUNIT
KFILE = 1
KSTART = 1
20 IF (KFILE.GT.NFILE) GO TO 10
IF (KFILE.EQ.NFILE.AND.KSTART.EQ.0) GO TO 10
KSKIP = NFILE - KFILE
IF (KSKIP.LE.0) GO TO 50
DO 40 I=1,KSKIP
30 READ (NUNIT,END=40) IDUMMY
GO TO 30
40 CONTINUE
50 KFILE = NFILE
KSTART = 1
KMAX = 256
KCTR = KMAX + 1
KREAD = 0
RETURN
ENTRY OCEAN2 (IND,KBLOCK,X)
DIMENSION X(1),DATA(256)
IF (KCTR.GT.KMAX) GO TO 70
55 DO 60 I=1,NCHAN
X(I) = DATA(KCTR)
60 KCTR = KCTR + 1
IND = 4
KBLOCK = KREAD
RETURN
70 READ (KUNIT,END=80) KMAX,NCHAN,(DATA(J),J=1,KMAX)
KSTART = 0
IF (KMAX.EQ.0) GO TO 70
KCTR = 1
KREAD = KREAD + 1
GO TO 55
80 KFILE = KFILE + 1
KSTART = 1
IND = 1
RETURN
ENTRY OCEAN3
WRITE (6,4001) KREAD
RETURN
4001 FORMAT ('OCEAN 3 CALLED (FORTRAN VERSION).',I10,' TAPE BLOCKS PRO
1CESSED.')
ENTRY RWUNLD
REWIND KUNIT
KFILE = 1

```

```

KSTART = 1
RETURN
END

```

```

THIS SUBROUTINE WHEN CALLED MOVES THE TAPE ON LOGICAL UNIT NUNIT
PAST KSKIP END OF FILE MARKS. THE RECORDS SKIPPED OVER MUST BE IN
FORTRAN BINARY. IF KSKIP IS ZERO OR NEGATIVE THE ROUTINE RETURNS
WITHOUT MOVING THE TAPE.

```

```

SUBROUTINE SKPFL (KSKIP,NUNIT)
IF (KSKIP.LT.1) RETURN
DO 20 I=1,KSKIP
10 READ (NUNIT,END=20) IDUMMY
20 GO TO 10
CONTINUE
RETURN
END

```

FORT, ONE-DIMENSIONAL FINITE COMPLEX FOURIER TRANSFORM.

FORT 002
FORT 003

```

SUBROUTINE FORT(A,M,S,IFS,IFERR)

```

```

FOURIER TRANSFORM SUBROUTINE, PROGRAMMED IN SYSTEM/360,
BASIC PROGRAMMING SUPPORT, FORTRAN IV. FORM C28-6504
THIS DECK SET UP FOR IBSYS ON IBM 7094.

```

FORT 004
FORT 005
FORT 006
FORT 007
FORT 008

```

THIS DECK MODIFIED TO ALLOW COMPUTATION OF SINE TABLE ( S(J) )
WITH M=14, FOR USE WITH SERIES OF 2*14 REAL VALUES
BY ADDITION OF STATEMENTS 6 AND 7 AND CHANGING 3 FROM
IF(M-13) 5,5,2 TO IF(M-13) 5,5,6

```

```

DOES EITHER FOURIER SYNTHESIS, I.E., COMPUTES COMPLEX FOURIER SERIES
GIVEN A VECTOR OF N COMPLEX FOURIER AMPLITUDES, OR, GIVEN A VECTOR
OF COMPLEX DATA X DOES FOURIER ANALYSIS, COMPUTING AMPLITUDES.
A IS A COMPLEX VECTOR OF LENGTH N=2**M COMPLEX NOS. OR 2*N REAL
NUMBERS. A IS TO BE SET BY USER.
M IS AN INTEGER 0.LT.M.LE.13, SET BY USER.
S IS A VECTOR S(J)= SIN(2*PI#J/NP), J=1,2,...,NP/4-1,
COMPUTED BY PROGRAM.
IFS IS A PARAMETER TO BE SET BY USER AS FOLLOWS-
IFS=0 TO SET NP=2**M AND SET UP SINE TABLE.

```

FORT 009
FORT 010
FORT 011
FORT 012
FORT 013
FORT 014
FORT 015
FORT 016
FORT 017
FORT 018
FORT 019


```

020 IFS=1 TO SET N=NP=2**M, SET UP SIN TABLE, AND DC FOURIER
021 SYNTHESIS, REPLACING THE VECTOR A BY
022 X(J)= SUM OVER K=0,N-1 OF A(K)*EXP(2*PI*I/N)**(J*K),
023 J=0,N-1, WHERE I=SQRT(-1)
024
025 THE X'S ARE STORED WITH RE X(J) IN CELL 2*J+1
026 AND IM X(J) IN CELL 2*J+2 FOR J=0,1,2,...,N-1.
027 THE A'S ARE STORED IN THE SAME MANNER.
028
029 IFS=-1 TO SET N=NP=2**M,SET UP SIN TABLE, AND DO FOURIER
030 ANALYSIS, TAKING THE INPUT VECTOR A AS X AND
031 REPLACING IT BY THE A SATISFYING THE ABOVE FOURIER SERIES.
032 IFS=+2 TO DO FOURIER SYNTHESIS ONLY, WITH A PRE-COMPUTED S.
033 IFS=-2 TO DO FOURIER ANALYSIS ONLY, WITH A PRE-COMPUTED S.
034 IFERR IS SET BY PROGRAM TO-
035 =0 IF NO ERROR DETECTED.
036 =1 IF M IS OUT OF RANGE., OR, WHEN IFS=+2,-2, THE
037 PRE-COMPUTED S TABLE IS NOT LARGE ENOUGH.
038 =-1 WHEN IFS =+1,-1, MEANS ONE IS RECOMPUTING S TABLE
039 UNNECESSARILY.
040
041 NOTE- AS STATED ABOVE, THE MAXIMUM VALUE OF M FOR THIS PROGRAM
042 ON THE IBM 7094 IS 13. FOR 360 MACHINES HAVING GREATER STORAGE
043 CAPACITY, ONE MAY INCREASE THIS LIMIT BY REPLACING 13 IN
044 STATEMENT 3 BELOW BY LOG2 N, WHERE N IS THE MAX. NO. OF
045 COMPLEX NUMBERS ONE CAN STORE IN HIGH-SPEED CORE. ONE MUST
046 ALSO ADD MORE DO STATEMENTS TO THE BINARY SORT ROUTINE
047 FOLLOWING STATEMENT 24 AND CHANGE THE EQUIVALENCE STATEMENTS
048 FOR THE K'S.
049
050 DIMENSION A(1), S(1), K(14)
051 EQUIVALENCE (K(13),K1), (K(12),K2), (K(11),K3), (K(10),K4)
052 EQUIVALENCE (K( 9),K5), (K( 8),K6), (K( 7),K7), (K( 6),K8)
053 EQUIVALENCE (K( 5),K9 ), (K( 4),K10), (K( 3),K11), (K( 2),K12)
054 EQUIVALENCE (K( 1),K13), ( K(11),N2)
055 IF(M)2,2,3
056 IF(M-13)5,5,6
057 FOLLOWING TWO STATEMENTS ADDED BY J GARRETT , SEE ABOVE
058 IF(M-14)7,7,2
059 IF(M-14)7,7,2
060 IF(IIFS)2,5,2
061 IFERR=1
062 RETURN
063 IFERR=0
064 N=2**M
065 IF( IABS(IFS) - 1 ) 200,200,10
066 WE ARE DOING TRANSFORM ONLY. SEE IF PRE-COMPUTED
067 S TABLE IS SUFFICIENTLY LARGE

```

```

10 IF( N-NP )20,20,12
12 IFERR=1
C   GO TO 200
   SCRAMBLE A, BY SANDE'S METHOD
20 K(1)=2*N
22 DO 22 L=2,M
   K(L)=K(L-1)/2
24 DO 24 L=M,12
   K(L+1)=2
C   NOTE EQUIVALENCE OF KL AND K(14-L)
C   BINARY SORT-
  IJ=2
  DO 30 J1=2,K1,2
    DO 30 J2=J1,K2,K1
    DO 30 J3=J2,K3,K2
    DO 30 J4=J3,K4,K3
    DO 30 J5=J4,K5,K4
    DO 30 J6=J5,K6,K5
    DO 30 J7=J6,K7,K6
    DO 30 J8=J7,K8,K7
    DO 30 J9=J8,K9,K8
    DO 30 J10=J9,K10,K9
    DO 30 J11=J10,K11,K10
    DO 30 J12=J11,K12,K11
    DO 30 J1=J12,K13,K12
    IF(IJ-J1-1)28,30,30
    T=A(IJ-1)
28  A(IJ-1)=A(J1-1)
    A(J1-1)=T
    T=A(IJ)
    A(IJ)=A(J1)
    A(J1)=T
    IJ=IJ+2
30  IF(IFS)32,2,36
C   DOING FOURIER ANALYSIS,SO DIV. BY N AND CONJUGATE.
32  FN = N
   DO 34 I=1,N
     A(2*I-1) = A(2*I-1)/FN
34  A(2*I) = -A(2*I)/FN
C   SPECIAL CASE- L=1
36  DO 40 I=1,N,2
     T = A(2*I-1)
     A(2*I-1) = T + A(2*I+1)
     T = A(2*I)
     A(2*I+1) = T - A(2*I+1)
     A(2*I) = T + A(2*I+2)
     A(2*I+2) = T - A(2*I+2)
40  IF(M-1) 2,1 ,50

```

```

FORT 065
FORT 066
FORT 067
FORT 068
FORT 069
FORT 070
FORT 071
FORT 072
FORT 073
FORT 074
FORT 075
FORT 076
FORT 077
FORT 078
FORT 079
FORT 080
FORT 081
FORT 082
FORT 083
FORT 084
FORT 085
FORT 086
FORT 087
FORT 088
FORT 089
FORT 090
FORT 091
FORT 092
FORT 093
FORT 094
FORT 095
FORT 096
FORT 097
FORT 098
FORT 099
FORT 100
FORT 101
FORT 102
FORT 103
FORT 104
FORT 105
FORT 106
FORT 107
FORT 108
FORT 109
FORT 110
FORT 111
FORT 112

```



```

C      50      SET FOR L=2
C      LEXPI=2**{(L-1)}
C      LEXP=8
C      LEXP=2**{(L+1)}
C      NPL= 2**MT 2**L
C      DO 130 L=2,M
C      SPECIAL CASE- J=0
C      DO 80 I=2,N2,LEXP
C      I1=I1+ LEXPI
C      I2=I1+ LEXPI
C      I3=I2+LEXPI
C      T=A(I-1)
C      A(I-1)= T +A(I2-1)
C      A(I2-1)= T-A(I2-1)
C      T=A(I)
C      A(I)= T+A(I2)
C      T= -A(I3)
C      TI= A(I3-1)
C      A(I3-1)= A(I1-1) - T - TI
C      A(I3)= A(I1)= A(I1-1) +T
C      A(I1-1)= A(I1-1) +T
C      A(I1)= A(I1)
C      IF(L-2) 120,120,90
C      80      KLAST=N2-LEXP
C      90      JJ=NPL
C      DO 110 J=4, LEXPI,2
C      NPJJ=NT-JJ
C      UR=S(NPJJ)
C      UI=S(JJ)
C      ILAST=J+KLAST
C      DO 100 I= J,ILAST,LEXP
C      I1=I1+LEXPI
C      I2=I1+LEXPI
C      I3=I2+LEXPI
C      T=A(I2-1)*UR-A(I2)*UI
C      TI=A(I2-1)*UI+A(I2)*UR
C      A(I2-1)=A(I-1)-T
C      A(I2)=A(I) - TI
C      A(I-1)=A(I-1)+T
C      A(I)=A(I)+TI
C      T=-A(I3-1)*UI-A(I3)*UR
C      TI=A(I3-1)*UR-A(I3)*UI
C      A(I3-1)=A(I1-1)-T
C      A(I3)=A(I1) -TI
C      A(I1-1)=A(I1-1)+T

```

```

113 FORT
114 FORT
115 FORT
116 FORT
117 FORT
118 FORT
119 FORT
120 FORT
121 FORT
122 FORT
123 FORT
124 FORT
125 FORT
126 FORT
127 FORT
128 FORT
129 FORT
130 FORT
131 FORT
132 FORT
133 FORT
134 FORT
135 FORT
136 FORT
137 FORT
138 FORT
139 FORT
140 FORT
141 FORT
142 FORT
143 FORT
144 FORT
145 FORT
146 FORT
147 FORT
148 FORT
149 FORT
150 FORT
151 FORT
152 FORT
153 FORT
154 FORT
155 FORT
156 FORT
157 FORT
158 FORT
159 FORT
160 FORT

```

```

100 A(I1) =A(I1)      +TI
C   END OF I LOOP
110 JJ=JJ+NPL
C   END OF J LOOP
120 LEXP1=2*LEXPI
C   LEXP = 2*LEXP
130 NPL=NPL/2
C   END OF L LOOP
140 IF(IFS)145,2,1
C   DOING FOURIER ANALYSIS. REPLACE A BY CONJUGATE.
145 DO 150 I=1,N
150 A(2*I) =-A(2*I)
160 GO TO 1
C   RETURN
C   MAKE TABLE OF S(J)=SIN(2*PI*J/NP), J=1,2,...,NT-1,NT=NP/4
200 NP=N
C   NP=M
C   NT=N/4
C   MT=M-2
205 IF(MT) 260,260,205
C   THETA=.7853981634
C   THETA=PI/2**(L+1)      FOR L=1
210 JSTEP = NT
C   JSTEP = 2** ( MT-L+1 ) FOR L=1
C   JDIF = NT/2
C   JDIF = 2** (MT-L) FOR L=1
C   S(JDIF) = SIN(THETA)
220 DO 250 L=2,MT
C   THETA = THETA/2.
C   JSTEP2 = JSTEP
C   JDIF = JDIF/2
C   S(JDIF)=SIN(THETA)
C   JC1=NT-JDIF
C   S(JC1)=COS(THETA)
C   JLAST=NT-JSTEP2
230 DO 240 J=JSTEP,JLAST,JSTEP
C   JC=NT-J
C   JD=J+JDIF
240 S(JD)=S(J)*S(JC1)+S(JDIF)*S(JC)
250 CONTINUE
260 IF(IFS)20,1,20
END

```

```

FORT 161
FORT 162
FORT 163
FORT 164
FORT 165
FORT 166
FORT 167
FORT 168
FORT 169
FORT 170
FORT 171
FORT 172
FORT 173
FORT 174
FORT 175
FORT 176
FORT 177
FORT 178
FORT 179
FORT 180
FORT 181
FORT 182
FORT 183
FORT 184
FORT 185
FORT 186
FORT 187
FORT 188
FORT 189
FORT 190
FORT 191
FORT 192
FORT 193
FORT 194
FORT 195
FORT 196
FORT 197
FORT 198
FORT 199
FORT 200
FORT 201
FORT 202
FORT 203
FORT 204
FORT 205

```

```

C SPECTRUM AND CROSS SPECTRUM STATISTICS FROM FOURIER
C COEFFICIENT TAPE PRODUCED BY 'FTOR' PROGRAM ON IBM
C SYSTEM 360 MODEL 67
C
C LAST REVISION JANUARY 27,1969          JOHN GARRETT
C
C THIS PROGRAM READS FOURIER COEFFICIENTS FROM TAPE PRODUCED BY FTOR
C PROGRAM, AND FROM THE APPROPRIATE SUMS OF THESE PRODUCE SPECTRA AND
C COSPECTRA. THE SUMS USED MAY BE FIXED WITH FREQUENCY OR MAY GO IN HALF
C OCTAVES FROM A SPECIFIED LOW FREQUENCY. AN AVERAGE, STANDARD DEVIATION
C AND LINEAR COEFFICIENT OF REGRESSION OVER THE ICMAX BLOCKS USED (SEE
C FTOR DESCRIPTION) ARE GIVEN FOR EACH VALUE OF SPECTRAL DENSITY.
C AT EACH FREQUENCY, THE COSPECTRUM BETWEEN (1) AND (2) IS GIVEN BY
C  $(R(1)*R(2) + I(1)*I(2))/2.0$ 
C AND THE QUADRATURE SPECTRUM BY
C  $(R(2)*I(1) - R(1)*I(2))/2.0$ 
C WHERE  $(R(N) + (I(N)*SQRT(-1)))$  IS THE COMPLEX FOURIER COEFFICIENT
C OF SERIES (N) AT THAT FREQUENCY
C
C A VARIETY OF PLOTTED OUTPUT IS AVAILABLE. IN ALL A HORIZONTAL BAR
C INDICATES THE FREQUENCY INTERVAL INCLUDED IN THE ESTIMATE PLOTTED, AND
C A VERTICAL BAR INDICATES THE EXPECTED STANDARD DEVIATION OF THE
C ESTIMATE (= STD.DEV. OF BLOCKS AVERAGED TO GIVE ESTIMATE/ SQRT(NUMBER
C OF BLOCKS))
C
C THE FOLLOWING SUBROUTINES MUST BE SUPPLIED BY USER
C PHASES
C PLVAL,TIC,LABEL
C
C THE FOLLOWING LOGICAL INPUT/OUTPUT UNITS ARE USED BY THIS PROGRAM
C 3= (TAPE) SUPPLYING COEFFICIENTS AND IDENTIFICATION AS
C    PRODUCED BY FTOR
C 5= (CARDS) CONTROL PARAMETERS
C 6= PRINTED OUTPUT
C
C THE FOLLOWING INPUT IS REQUIRED
C
C A CARD IS REQUIRED TO IDENTIFY YOUR GRAPHICAL OUTPUT FOR THE COMPUTING
C CENTRE STAFF. IT MUST BE PRESENT WHETHER PLOTS ARE PRODUCED OR NOT.
C THE FIRST 72 COLUMNS OF THIS CARD WILL BE REPRODUCED ON THE BEGINNING
C OF YOUR PLOT. THIS CARD APPEARS ONLY ONCE IN THE JOB AND IS THE FIRST
C DATA CARD. THE FOLLOWING SET OF CARDS IS PRESENT FOR EACH FILE OF
C FOURIER COEFFICIENTS TO BE PROCESSED.
C
C FIRST CARD, IN COLUMN
C 1-9 IDUSER = USER IDENTIFICATION NUMBER FOR DATA DESIRED
C 14-15 ICMAX = NUMBER OF CHANNELS TO BE USED (MAX 10)

```

```

24-25 IBMAX = NUMBER OF DATA BLOCKS FOR STATISTICS
34-35 IBSTAR = STATISTICS START WITH BLOCK NO. IBSTAR
44-45 NFILE = DATA IS IN NFILE-TH FILE ON FOURIER COEFFICIENT TAPE
54 IAXIS = 0 (BLANK) IF PLOT AXES TO BE SET BY PROGRAM
      = 1 IF AXES TO BE SET BY USER
55 IPLOT = 0 IF NO PLOTTED OUTPUT DESIRED
      = 1 IF SPECTRAL DENSITY TO BE PLOTTED AGAINST FREQUENCY
      = 2 IF LOG10 SPECTRAL DENSITY TO BE PLOTTED AGAINST FREQUENCY
      = 3 IF LOG10 SPECTRAL DENSITY TO BE PLOTTED AGAINST LOG10
        FREQUENCY
      = 4 IF (FREQ*SPECTRAL DENSITY) TO BE PLOTTED
        AGAINST LOG10 FREQUENCY
      = 5 SAME AS IPLOT=1 EXCEPT COHERENCE AND PHASE PLOTTED
        INSTEAD OF CO AND QUAD SPECTRA
      = 6 SAME AS = 2 EXCEPT COHERENCE AND PHASE PLOTTED INSTEAD
        OF CO AND QUAD
      = 7 SAME AS 3 EXCEPT COHERENCE AND PHASE INSTEAD OF CO+QU
      = 8 SAME AS 4 WITH COHERENCE AND PHASE INSTEAD OF CO + QU
65 IPHASE = 0 FOR NO PHASE CORRECTIONS
      = 1 FOR PHASES TO BE CORRECTED.

```

CARD INSERTED ONLY IF IAXIS = 1

```

IN COLUMN
1-10 VALUE OF ORIGIN FOR SPECTRUM AND COSPECTRUM AXIS (F10.)
11-20 UNITS PER INCH FOR SPECTRUM AND COSPECTRUM AXIS (F10.)
      AXIS WILL BE 5.00 INCHES LONG
21-30 VALUE OF ORIGIN FOR QUAD SPECTRUM AXIS (F10.)
31-40 UNITS PER INCH FOR QUAD SPECTRUM AXIS (F10.)
      AXIS WILL BE
      5.00 INCHES LONG. (COHERENCE AND PHASE WILL BE
      SCALED BY PROGRAM IF PLOTTED INSTEAD OF CO AND QUAD.)
41-50 VALUE OF ORIGIN FOR FREQUENCY AXIS (F10.)
51-60 UNITS PER INCH FOR FREQUENCY AXIS (F10.)
61-70 LENGTH OF FREQUENCY AXIS IN INCHES (F10.)
N.B. WHEN LOG10 PLOTS HAVE BEEN REQUESTED NUMBERS ABOVE MUST ALL
REFER TO LOG10, E.G. UNITS PER INCH = 1.00 MEANS 1 DECADE PER INCH

```

```

NEXT CARD, IN COLUMN
5 LINOCT = 0 FOR CONSTANT BANDWIDTH, GIVEN BY BANDW BELOW
      = 1 FOR EXPONENTIAL BANDWIDTHS ( GIVES LOG10 FREQ.
        SCALE IN ALL PLOTS )
6-15 STRFRQ = APPROXIMATE CENTER FREQUENCY OF FIRST POINT OF
      SPECTRUM, IN HERTZ. (MUST INCLUDE A DECIMAL POINT)
      FIRST BAND WILL INCLUDE ZEROH HARMONIC IF STRFRQ
      IS LESS THAN BANDW
16-25 BANDW = APPROX. BANDWIDTH FOR FIXED BANDWIDTH SPECTRA,
      IN HERTZ. (MUST INCLUDE A DECIMAL POINT)
30 INDOV = 1 IF FOURIER COEFFICIENTS TO BE HANNED BEFORE
      SPECTRA COMPUTED

```


CC

SUBSEQUENT IC MAX CARDS
4-5 CHANNEL NUMBER OF A CHANNEL FOR WHICH SPECTRA ARE WANTED
14-15, 19-20, 24-25, 29-30 TO 59-60, CHANNEL NUMBERS OF UP TO IC MAX
CHANNELS FOR WHICH CROSS SPECTRA WITH CHANNEL GIVEN IN
COLS. 4-5 ARE DESIRED. SPECTRUM IS HERE CONSIDERED AS CROSS
SPECTRUM OF CHANNEL WITH ITSELF, I.E. THE NUMBER IN 4-5
MUST REAPPEAR IF SPECTRUM IS TO BE OBTAINED. ALSO, CROSS
SPECTRA ARE DONE ONLY IF EACH CHANNEL APPEARS IN THE LIST
ON THE OTHER CHANNELS CARD. THUS TO GET SPECTRUM FOR
CHANNEL 2 AND CROSS SPECTRUM BETWEEN 2 AND CHANNEL 8, THERE
MUST BE A CARD WITH 2 IN COL. 5 AND BOTH 2 AND 8 IN THE LIST
AND ANOTHER CARD WITH 8 IN COL. 5 AND 2 IN THE LIST.

PHASE CORRECTION DECK(INSERT ONLY IF IPHASE NOT ZERO)
PHASE CORRECTIONS APPLIED WILL BE OBTAINED BY LINEAR INTERPOLATION
BETWEEN VALUES SUPPLIED AT FREQUENCIES LISTED BELOW
FIRST K CARDS(K LESS THAN 6) FREQUENCIES(F10.4) AT WHICH PHASE
1-10, 11-20, ..., 71-80, FREQUENCIES(F10.4) AT WHICH PHASE
CORRECTIONS ARE TO BE SUPPLIED (UP TO 48 OF THEM)
IF LESS THAN THE FULL 48 FREQUENCIES ARE SUPPLIED, THEN
THE LAST FREQUENCY MUST BE LEFT BLANK.

SUBSEQUENT IC MAX SETS OF K CARDS EACH
1-10, 11-20, ..., 71-80, PHASE CORRECTIONS(F10.4) TO BE APPLIED
AT THE ABOVE FREQUENCIES. EACH SET CONTAINS THE CORRECTIONS
FOR A GIVEN CHANNEL, AND THE SETS ARE IN THE SAME ORDER AS THE
SPECTRUM CARDS. IF NO CORRECTIONS ARE TO BE APPLIED TO A
CHANNEL, THE K CARDS FOR THAT CHANNEL SHOULD BE BLANK.
CORRECTIONS MUST BE IN RADIAN. A POSITIVE CORRECTION WILL
CAUSE THE CORRECTED PHASE TO LEAD THE UNCORRECTED ONE.

LAST CARD MUST BE BLANK UNLESS ANOTHER FILE IS TO BE PROCESSED, IN
WHICH CASE A COMPLETE NEW SEQUENCE OF CARDS APPROPRIATE TO THE NEW
FILE SHOULD FOLLOW.

DIMENSION ARTAPE(256), TAPRAY(256), PHI(10,10,32), PHISQ(10,10,32),
1 PHIBL(10,10,32), KCHA(10), KCHB(10,10), ICH(10,10), FREQ(32), NEND(33)
2, BANDW(32), A(32), AD(32), B(32), BD(32), ACHNAM(9,10), AUNITS(2,10),
3 CAL(10), MCHAN(10), INDCH(12), Z(256), I2TAB(14), AE(32), BE(32),
4 POS(32), DC(10,10), DCSQ(10,10), DCBL(10,10), PHASE(50,12)
5, AR1(10), AR2(10), AR3(10), AM1(10), AM2(10), AM3(10), BR1(10,10),
6, BR2(10,10), BR3(10,10), BM1(10,10), BM2(10,10), BM3(10,10),
7, COSP(10,10), QUSP(10,10), POSL(32), POSU(32), NLOGBW(9)
DIMENSION GTITLE(18)
EQUIVALENCE (TAPRAY(100), ICH(1,1), (ARTAPE(1), Z(1)), IDUSER)
1, (Z(2), MBLOCK), (Z(3), IBLOCK), (Z(4), NTAP), (Z(5), KCHAN),
2 (Z(6), SAMFRQ), (Z(11), MCHAN(1)), (Z(31), ACHNAM(1,1))

```

3 ) , (Z(121), AUNITS(1,1)), (Z(141), CAL(1)), (Z(151), INDCH(1)),
4 (Z(163), I2TAB(1)), (Z(210), IAPRAY(1))
5 , (Z(10), IHANN), (Z(210), IAR)
DATA NLOGBW(1), NLOGBW(2), NLOGBW(3), NLOGBW(4), NLOGBW(5), NLOGBW(6),
1 NLOGBW(7), NLOGBW(8), NLOGBW(9)/1,2,3,4,6,8,11,15,20 /
CORNA = 1./2.00000
CORNA = SQRT(CORNA)
CORNB = 1.00/6.000
CORNB = SQRT(CORNB)
LOPT = 0
REWIND 3
IFILE = 1
KBEGIN = 1
YREORG = 11
READ(5,6) GTITLE
READ(5,15,END=5020) IDTEST, ICMAX, IBSTAR, NFILE, IPLOT, IPHASE
IF (IDTEST.EQ.0) GO TO 5020
IAXIS = IPLOT/10
IPLOT = IPLOT - (10*IAXIS)
FRAXLN = 4.00
IF (IAXIS.GT.0) READ(5,5) XRES1, DXRES1, XRES2, DXRES2, XMIN3, DX3,
1 FRAXLN
TRANS = FRAXLN+4.00
IF (IFILE.LT.NFILE) GO TO 19
IF (IFILE.EQ.NFILE.AND.KBEGIN.EQ.1) GO TO 19
REWIND 3
IFILE = 1
KBEGIN = 1
READ(5,16) LINDCT, STRFRQ, BAND, INDOX
IF (IPLOT.LE.0) GO TO 20
IF (LOPT.GT.0) GO TO 20
CALL PLOTS
CALL SYMBOL (0.,0.,0.14, GTITLE, 0.,72)
CALL PLOT (0.,3.,-3)
LOPT = 1
DO 21 I = 1,10
KCHA(I) = 0
DO 21 J = 1,10
ICH(I,J) = 0
KCHB(I,J) = 0
CONTINUE
DO 30 ICD = 1, ICMAX
READ(5,25) KCHA(ICD), (ICH(ICD,J), J=1,10)
IF (KCHA(ICD).GT.0) GO TO 30
ICMAX = ICD
GO TO 31
CONTINUE
DO 40 KA = 1, ICMAX

```



```

KB = 0
DO 39 J = 1,10
IF (KCHA(KA,J).GT.ICH(KA,J)) GO TO 39
KB = KB+1
KCHB(KA,KB) = ICH(KA,J)
CONTINUE
IF (IPHASE.EQ.0) GO TO 41
CALL PHASES( 0,Z,CO,QU,IFB,FREQ,NEND,BANDW,PHASE,KCA,KCB,FUNDEF,
1 ICMAX,MCHAN,MAXIF,RAD , KCHA)
41 KSKIP = NFILE - IFILE
IF (KSKIP.GT.0) CALL SKPFL (KSKIP,3)
IFILE = NFILE
READ (3,END=1108) ARTAPE
KBEGIN = 0
IF (IDTEST - IDUSER) 4900,54,4900
IF (IBSTAR - MBLOCK) 59,65,55
DO 56 N = 1,NTAP
56 READ (3,END=1108) TAPRAY
GO TO 54
IB = (IBSTAR + IBMAX-1) - MBLOCK
IF (IB) 4700,60,60
IBMAX = IBMAX - (MBLOCK - IBSTAR)
IBSTAR = MBLOCK
C PRINT INTENTIONS
65 IS = 0
WRITE (6,62) IDUSER,IFILE,ICMAX,IBMAX,IBSTAR
IF (IHANN.GT.0) INDOV = 0
C EXCHANGE REAL CHANNEL NUMBERS FOR INDICES USED ON TAPE
DO 67 K = 1,10
IT = 0
KC = KCHA(K)
KIS = K - IS
KCHA(KIS) = 0
IF (KC.LE.0) GO TO 659
KCHA(KIS) = INDCH(KC)
IF (INDCH(KC).EQ.0) IS = IS+1
DO 66 I = 1,10
IC = KCHB(K,I)
KIT = I - IT
KCHB(KIS,KIT) = 0
IF (IC.LE.0) GO TO 66
KCHB (KIS,KIT) = INDCH(IC)
IF (INDCH(IC).EQ.0) IT = IT+1
CONTINUE
66 NHAR = 256/((2*KCHAN)+1)
67

```

C SET UP SUMMING ROUTINE, FIRST GETTING FUNDAMENTAL FREQUENCY

```
JBLOCK = IBLOCK/2
TBLOCK = IBLOCK
FUNDNR = SAMFRQ/TBLOCK
BLKLNQ = TBLOCK/SAMFRQ
NSTART = STRFRQ/FUNDR
IFRSCL=0
```

70 C GET NUMBER OF HARMONICS FOR FIXED BANDWIDTH

```
IF(LINOC1) 70,70,76
NBAND = BAND/FUNDR
IF(NBAND.EQ.0) NBAND=1
BAND = NBAND
BANHAF = BAND*FUNDR
J = NBAND/2
```

C SET

```
NSTART = NSTART - J
IF(NSTART.LT.0) NSTART = 0
NUMBER OF FIXED BANDS
MAXIF = (JBLOCK - NSTART)/NBAND
IF (MAXIF.GT.32) MAXIF = 32
STRFRQ = NSTART
STRFRQ = STRFRQ*FUNDR
FREQ(1) = STRFRQ+((BAND-FUNDR)/2.)
POS(1)=FREQ(1)
NEND(1) = NSTART +NBAND-1
BANDW(1) = BAND
POSL(1) = FREQ(1)-BANHAF
POSU(1) = FREQ(1)+BANHAF
DO 73 I = 2,MAXIF
```

```
J = I-1
FREQ(I) = FREQ(J) + BAND
POS(I)=FREQ(I)
POSU(I) = FREQ(I)+BANHAF
POSL(I) = POSU(J)
BANDW(I) = BAND
NEND(I) = NEND(J) + NBAND
GO TO 80
```

73 C SET UP FOR LOGARITHMIC SUMMING

```
IF(NSTART.NE.1) NSTART=1
```

```
NEND(1) = 1
BANDW(1) = FUNDR
POSL(1) = FUNDR/2.0
POSU(1) = FUNDR+POSL(1)
FREQ(1) = SORT(POSL(1)*POSU(1))
POS(1)=FREQ(1)
DO 74 I = 2,9
NEND(I) = NEND(9)
```

74 BAND = NEND(9)

```

77      BAND = BAND+0.50
        I = 9
        I = I+1
        BAND = BAND*1.33352
        NEND(I) = BAND
        MAXIF = I
        IF(NEND(I).LT.JBLOCK) GO TO 77
        NEND(I) = JBLOCK
        DO 78 I = 2,MAXIF
          J = I-1
          BAND = NEND(I)-NEND(J)
          BANDW(I) = BAND*FUNDFR
          BAND = NEND(I)
          POSU(I) = (BAND+0.50)*FUNDFR
          POSL(I) = POSU(J)
          FREQ(I)=SQRT(POSL(I)*POSU(I))
          POS(I)=FREQ(I)
78      C NOW GET SET TO READ SOME DATA
80      KBLOCK = 1
        IF (IHANN.GT.0.AND.NEND(MAXIF).EQ.JBLOCK) NEND(MAXIF) = JBLOCK - 1
81      NSTARW = NSTART - INDOU
        DO 100 IFR = 1,MAXIF
          DO 99 KA = 1,10
            DO 98 KB = 1,10
              DC(KA,KB) = 0.0
              DCSQ(KA,KB) = 0.0
              DCBL(KA,KB) = 0.0
              PHI(KA,KB,IFR) = 0.0
              PHISQ(KA,KB,IFR) = 0.0
              PHIBL(KA,KB,IFR) = 0.0
98      CONTINUE
99      CONTINUE
100     BL = 0.0
        BLSQ = 0.0
        C BEGINNING OF BLOCK LOOP
180     BLOCK = MBLOCK
        DO 181 KBC = 1,10
          AR1(KBC) = 0.0
          AR2(KBC) = 0.0
          AR3(KBC) = 0.0
          AM1(KBC) = 0.0
          AM2(KBC) = 0.0
          AM3(KBC) = 0.0
          DO 181 KAC = 1,10
            COSP(KAC,KBC) = 0.0
            QUSP(KAC,KBC) = 0.0
            BR1(KAC,KBC) = 0.0
            BR2(KAC,KBC) = 0.0

```

```

181      BR3(KBC,KAC) = 0.0
      BM1(KBC,KAC) = 0.0
      BM2(KAC,KBC) = 0.0
      BM3(KAC,KBC) = 0.0
      CONTINUE
      BL = BL + BLOCK
      BLSQ = BLSQ + (BLOCK*BLOCK)
      IFR = 1
      IHARM = -1
185      DO 1000 NT = 1,NTAP
      NCAR = NT
      READ (3,END=1108) TAPRAY
190      DO 990 IH = 1,NHAR
      IF (IFR.GT.MAXIF) GO TO 1000
19005      IHARM = IHARM + 1
      IHARM2 = IHARM - INDOW
      IF (IHARM2.EQ.0) GO TO 191
      IF (IHARM.LT.NSTARW) GO TO 990
      IF (IHARM2.LE.NEND(IFR)) GO TO 191
      DO 850 KA = 1,ICMAX
      KCA = KCHA(KA)
      IF (KCA.LE.0) GO TO 850
      DO 850 KB = 1,ICMAX
      KCB = KCHB(KA,KB)
      IF (KCB.LE.0) GO TO 850
      PHI(KCA,KCB,IFR) = COSP(KA,KB)+PHI(KCA,KCB,IFR)
      PHIBL(KCA,KCB,IFR)=COSP(KA,KB)*BLOCK+PHIBL(KCA,KCB,IFR)
      PHISQ(KCA,KCB,IFR)=COSP(KA,KB)*COSP(KA,KB)+PHISQ(KCA,KCB,IFR)
      COSP(KA,KB) = 0.0
      IF (KCA.EQ.KCB) GO TO 850
      PHI(KCB,KCA,IFR)=QUSP(KA,KB)+PHI(KCB,KCA,IFR)
      PHIBL(KCB,KCA,IFR)=QUSP(KA,KB)*BLOCK+PHIBL(KCB,KCA,IFR)
      PHISQ(KCB,KCA,IFR)=QUSP(KA,KB)*QUSP(KA,KB)+PHISQ(KCB,KCA,IFR)
      QUSP(KA,KB) = 0.0
      CONTINUE
850      IFR = IFR+1
      IF (IFR.GT.MAXIF) GO TO 1000
      ISUB = (IH-1)*(1+(2*KCHAN))
      THARM = IHARM
      IF (.NOT.(THARM.EQ.TAPRAY(ISUB+1))) GO TO 4800
191      DO 980 KA = 1,ICMAX
      KCA = KCHA(KA)
      IF (KCA.LE.0) GO TO 980
      IS = ISUB + (2*KCA)
      IF (INDOW.GT.0) GO TO 196
      ARL = TAPRAY(IS)
      AMG = TAPRAY(IS+1)
      GO TO 199

```

```

196  AR3(KA) = AR2(KA)
      AR2(KA) = AR1(KA)
      AR1(KA) = TAPRAY(IS)
      AM3(KA) = AM2(KA)
      AM2(KA) = AM1(KA)
      AM1(KA) = TAPRAY(IS+1)
      IF(IHARM2.EQ.0) GO TO 197
      IF(IHARM2.LT.NSTART) GO TO 199
      ARL = CORNB*((2.0*AR2(KA))-AR1(KA)-AR3(KA))
      AMG = CORNB*((2.0*AM2(KA))-AM1(KA)-AM3(KA))
      GO TO 199
197  ARL = 2.0*CORNB*(AR2(KA)-AR1(KA))
      AMG = 0.0
199  CONTINUE
      DO 970 KB = 1,ICMAX
      KCB = KCHB(KA,KB)
      IF(KCB.LE.0) GO TO 970
      IT = ISUB+(2*KCB)
      IF(INDOW.GT.0) GO TO 206
      BRL = TAPRAY(IT)
      BMG = TAPRAY(IT+1)
      GO TO 209
206  BR3(KA,KB) = BR2(KA,KB)
      BR2(KA,KB) = BR1(KA,KB)
      BR1(KA,KB) = TAPRAY(IT)
      BM3(KA,KB) = BM2(KA,KB)
      BM2(KA,KB) = BM1(KA,KB)
      BM1(KA,KB) = TAPRAY(IT+1)
      IF(IHARM2.EQ.0) GO TO 207
      IF(IHARM2.LT.NSTART) GO TO 209
      BRL = CORNB*((2.0*BR2(KA,KB))-BR1(KA,KB)-BR3(KA,KB))
      BMG = CORNB*((2.0*BM2(KA,KB))-BM1(KA,KB)-BM3(KA,KB))
      GO TO 209
207  BRL = 2.0*CORNB*(BR2(KA,KB)-BR1(KA,KB))
      BMG = 0.0
209  CONTINUE
      CO = ((ARL*BMG)+(AMG*BMG))/2.0
      QU = ((BRL*AMG)-(ARL*BMG))/2.0
      C STORE REAL PARTS
      IF(IHARM2.NE.0) GO TO 801
      IF(KCA.EQ.KCB) GO TO 250
      DC(KCA,KCB) = ARL*BRL+DC(KCA,KCB)
      DCSQ(KCA,KCB) = 4.00*CO*CO + DCSQ(KCA,KCB)
      DCBL(KCA,KCB) = ARL*BRL*BLOCK + DCBL(KCA,KCB)
      GO TO 256
250  DC(KCA,KCB) = ARL+DC(KCA,KCB)
      DCSQ(KCA,KCB) = ARL*ARL+DCSQ(KCA,KCB)
      DCBL(KCA,KCB) = (ARL*BLOCK) + DCBL(KCA,KCB)

```



```

256 CONTINUE
801 IF(IHARM2.LT.NSTART) GO TO 970
   COSP(KA,KB) = CO + COSP(KA,KB)
   QUSP(KA,KB) = QU + QUSP(KA,KB)
970 CONTINUE
980 CONTINUE
990 IF (IHARM2.EQ.NEND(MAXIF)) GO TO 19005
1000 CONTINUE
   KBLOCK = KBLOCK + 1
   IF(KBLOCK.GT.IBMAX) GO TO 1110
   READ (3,END=1100) ARTAPE
   GO TO 180
C IF END OF FILE, PROCEEDS WITH COMPUTATION
1100 IFILE = IFILE + 1
   KBEGIN = 1
   IBMAX = KBLCK - 1
   WRITE (6,1105) IBMAX
   GO TO 1110
1108 IFILE = IFILE + 1
   KBEGIN = 1
   IBMAX = KBLCK
   WRITE (6,1109) IBMAX,NCAR
1110 BLMAX = IBMAX
   BLROGT = SQRT(BLMAX)
C DATA IS ALL STORED, NOW CONVERT FOR STATISTICS AND WRITE OUT
   IF( I PHASES.EQ.0 ) GO TO 1120
   CALL PHASES( 1,Z,CO,QU,IFB,FREQ,NEND,BANDW,PHASE,KCA,KCB,FUNDFR,
1 ICMAX,MCHAN, MAXIF, RAD , KCHA)
1120 DO 1130 KA = 1,ICMAX
   KCA = KCHA(KA)
   IF(KCA.LE.0) GO TO 1130
   DO 1130 KB = 1,ICMAX
   KCB = KCHB(KA,KB)
   IF(KCB.LE.0) GO TO 1130
   DDC = DC(KCA,KCB) / BLMAX
   DDCSQ = (DCSQ(KCA,KCB) - (BLMAX*DDC*DDC)) / (BLMAX-1.)
   DDCSQ = ABS(DDCSQ)
   IF(KCA.NE.KCB) ODCSQ = SQRT(DDCSQ)
   DCOR = (DCBL(KCA,KCB) - (BL*ODC)) / (BLSQ - ((BL*BL)/BLMAX))
   DC(KCA,KCB) = ODC
   DCSQ(KCA,KCB) = DDCSQ
   DCBL(KCA,KCB) = DCOR
CONTINUE
1130 DO 3000 KA = 1,ICMAX
   KCA = KCHA(KA)
   IF (KCA.LE.0) GO TO 3000
   DO 2900 KB = 1,ICMAX

```



```

KCB = KCHB(KA,KB)
IF (KCB.LE.0) GO TO 2900
SUMCO = 0.0
SUMQU = 0.0
IF (KCB-KCA) 1200,1150,1200
C WRITE HEADING FOR SPECTRUM
1150 WRITE(6,1155) IDUSER,MCHAN(KCA),(ACHNAM(L,KCA),L=1,9)
WRITE(6,1165) IBMAX,IBLOCK,SAMFRQ,BLKNG
IF(IHANN.GT.0) WRITE(6,14)
IF(INDOW.GT.0) WRITE(6,194)
WRITE(6,1175)CAL(KCA),(AUNITS(KL,KCA),KL=1,2),(AUNITS(KL,KCA),
1 KL=1,2)
GO TO 1209
C WRITE HEADING FOR COSPECTRUM
1200 WRITE(6,1205) IDUSER,MCHAN(KCA),(ACHNAM(L,KCA),L=1,9),MCHAN(KCB),
2 (ACHNAM(L,KCB),L=1,9)
WRITE(6,1165) IBMAX,IBLOCK,SAMFRQ,BLKNG
IF(IHANN.GT.0) WRITE(6,14)
IF(INDOW.GT.0) WRITE(6,194)
WRITE(6,1206)CAL(KCA),CAL(KCB),(AUNITS(KL,KCA),KL=1,2),
1 (AUNITS(KL,KCB),KL=1,2),(AUNITS(KL,KCA),KL=1,2)
2 (AUNITS(KL,KCB),KL=1,2),(AUNITS(KL,KCA),KL=1,2),
3 (AUNITS(KL,KCB),KL=1,2),(AUNITS(KL,KCA),KL=1,2),
C NOW GET AV. DENSITY, STD.DEV., REGRESSION COEFF
1209 DO 1500 IFB=1,MAXIF
VID = BANDW(IFB)
DIV = BLMAX*VID
C FIRST FOR REAL PARTS
CO = PHI(KCA,KCB,IFB)/DIV
FCO = CO*FREQ(IFB)
COSQ = ((PHISQ(KCA,KCB,IFB)/(VID*VID))-(BLMAX*CO*CO))/(BLMAX-1.)
COSQ = ABS(COSQ)
COSQ = SQRT(COSQ)
A(IFB)=CO
AD(IFB) = CO + (COSQ/BLROOT)
AE(IFB) = CO - (COSQ/BLROOT)
IF(IPLT.NE.4) GO TO 12091
A(IFB) = A(IFB)*FREQ(IFB)
AD(IFB) = AD(IFB)*FREQ(IFB)
AE(IFB) = AE(IFB)*FREQ(IFB)
C CONTINUE
12091 COR=((PHIBL(KCA,KCB,IFB)/VID)-(BL*CO))/(BLSQ-((BL*BL)/BLMAX))
C IF THIS WAS SPECTRUM, WRITE OUT
IF(KCA.EQ.KCB) GO TO 1300
C IF NOT, DO IMAGINARY PARTS
QU = PHI(KCB,KCA,IFB)/DIV
QUSQ = ((PHISQ(KCB,KCA,IFB)/(VID*VID))-(BLMAX*QU*QU))/(BLMAX-1.)
QUSQ = ABS(QUSQ)

```

```

QUSQ = SQRT(QUSQ)
QR=((PHIBL(KCB,KCA,IFB)/VID)-(8L*QU))/(BLSQ-((8L*BL)/BLMAX))
ARL = PHI(KCA,KCA,IFB)/DIV
BRL = PHI(KCB,KCB,IFB)/DIV
C = CO * CO
D = QU * QU
RAD = SQRT(C+D)
ARL = ARL*BRL
ARL = ABS(ARL)
CORCOF = RAD/SQRT(ARL)
IF( IPHASE=EQ.0 ) GO TO 1211
CALL PHASES( 2,Z,CO,QU,IFB,FREQ,NEND,BANDW,PHASE,KCA,KCB,FUNDFR,
1 ICMAX,MCHAN, MAXIF, RAD , KCHA)
1211 ARG = QU/CO
FQU = QU*FREQ(IFB)
FCO = CO*FREQ(IFB)
A(IFB)=CO
AD(IFB) = CO + (COSQ/BLROOT)
AE(IFB) = CO - (COSQ/BLROOT)
B(IFB) = QU
BD(IFB) = QU + (QUSQ/BLROOT)
BE(IFB) = QU - (QUSQ/BLROOT)
IF(IPLT.LT.4) GO TO 12092
IF(IPLT.GT.4) GO TO 12093
A(IFB) = A(IFB)*FREQ(IFB)
AD(IFB) = AD(IFB)*FREQ(IFB)
AE(IFB) = AE(IFB)*FREQ(IFB)
B(IFB) = B(IFB)*FREQ(IFB)
BD(IFB) = BD(IFB)*FREQ(IFB)
BE(IFB) = BE(IFB)*FREQ(IFB)
GO TO 12092
12093 A(IFB) = CORCOF
AD(IFB) = CORCOF
AE(IFB) = CORCOF
B(IFB) = BMG
BD(IFB) = BMG
BE(IFB) = BMG
CONTINUE
12092 IF(ARG.LT.10000.) GO TO 1210
BMG = SIGN(1.,QU)*90.0
GO TO 1212
1210 BMG = 57.296 * ATAN2(QU,CO)
C WRITE OUT CROSS SPECTRA
1212 WRITE(6,1215)IFB,FREQ(IFB),CO,CCSQ,COR,QU,QUSQ,QUR,FCO,FQU,BMG,
1 CORCOF
GO TO 1499
1300 WRITE(6,1305)IFB,FREQ(IFB),BANDW(IFB),CO,COSQ,COR,FCO,NEND(IFB)
1499 SUMCO = SUMCO+(CO*VID)

```

```

1500 SUMQU = SUMQU + (QU*VID)
      CONTINUE
      IF(KCA.EQ.KCB) GO TO 1501
      WRITE(6,1535)SUMCO, (AUNIT(S(KL,KCA),KL=1,2), (AUNIT(S(KL,KCB),KL=1,2)
1      1,SUMQU, (AUNIT(S(KL,KCA),KL=1,2), (AUNIT(S(KL,KCB),KL=1,2)
      GO TO 1502
1501 WRITE(6,1545) SUMCO, (AUNIT(S(KL,KCA),KL=1,2)
1502 CONTINUE
      IF(KCA.EQ.KCB) GO TO 1503
      DCDF = DC(KCA,KCB) - (DC(KCA,KCA)*DC(KCB,KCB))
      WRITE(6,1525)DC(KCA,KCB), (AUNIT(S(KL,KCA),KL=1,2), (AUNIT(S(KL,KCB),
1      1,KL=1,2),DCSQ(KCA,KCB), (AUNIT(S(KL,KCA),KL=1,2), (AUNIT(S(KL,KCB),KL=
2      2,1,2),DCBL(KCA,KCB), (AUNIT(S(KL,KCA),KL=1,2), (AUNIT(S(KL,KCB),KL=1,
3      3,2),DCDF, (AUNIT(S(KL,KCA),KL=1,2), (AUNIT(S(KL,KCB),KL=1,2)
      GO TO 1504
1503 WRITE(6,1526)DC(KCA,KCB), (AUNIT(S(KL,KCA),KL=1,2),DCSQ(KCA,KCB),
1      1 (AUNIT(S(KL,KCA),KL=1,2),DCBL(KCA,KCB), (AUNIT(S(KL,KCA),KL=1,2)
1504 CONTINUE
      IF (IPLOT.EQ.0) GO TO 2900
      LPL = 1
      GO TO (1740,1740,1750,1750,1740,1740,1750,1750),IPLCT
1740 IF(LINDCT) 1770,1770,1750
1750 IF(POS(1).NE.0.0) GO TO 17505
      POSL(1) = POSL(1)+POS(2)-POS(1)
      POSU(1) = POSU(1)+POS(2)-POS(1)
      POS(1) = POS(2)
      IF(IFRSCL.GT.0) GO TO 1770
      DO 1751 IL=1,MAXIF
      POSU(IL) = ALOG10(POSU(IL))
      IF(POSL(IL).LE.0.0)POSL(IL) = 0.001*POS(IL)
      POSL(IL) = ALOG10(POSL(IL))
      POS(IL) = ALOG10(POS(IL))
      GO TO (7100,7001,7001,7100,7100,7001,7001,7100), IPLOT
1751 IF(IPLOT.LE.4) GO TO 70015
1770 IF(KCA.NE.KCB) GO TO 7100
7001 DO 7003 IL=1,MAXIF
70015 IF(A(IL))8200,8100,8110
8100 A(IL) = AD(IL)
8110 IF(AE(IL).LE.0.0) AE(IL) = A(IL)
      GO TO 8300
8200 A(IL) = -A(IL)
      AE(IL) = -AE(IL)
      AD(IL) = -AD(IL)
      IF (AD(IL).LE.0.0) AD(IL) = A(IL)
8300 A(IL) = ALOG10(A(IL))
      AD(IL) = ALOG10(AD(IL))
      AE(IL) = ALOG10(AE(IL))
      IF (KCA.EQ.KCB) GO TO 7003

```

8400	IF (B(IL)) 8500,8400,8400
8500	IF (BE(IL).LT.0.0) BE(IL) = B(IL) GO TO 8600
	B(IL) = -B(IL) BE(IL) = -BE(IL) BD(IL) = -BD(IL)
8600	IF (BD(IL).LE.0.0) BD(IL) = B(IL) B(IL) = ALOG10(B(IL)) BD(IL) = ALOG10(BD(IL)) BE(IL) = ALOG10(BE(IL))
7003	CONTINUE
7100	XMIN1 = XRES1 DX1 = DXRES1 XMIN2 = XRES2 DX2 = DXRES2 ISCL = IAXIS IF (IPLT.GT.4.AND.KCA.NE.KCB) ISCL = 0 IF (ISCL.GT.0) GO TO 7102 CALL SCALE(A,MAXIF,5.00,XMIN1,DX1,1) CONTINUE
7102	DO 7005 IL = 1,MAXIF IF (ISCL.GT.0) A(IL)=(A(IL)-XMIN1)/DX1 AD(IL) = (AD(IL)-XMIN1)/DX1 AE(IL) = (AE(IL) - XMIN1)/DX1 IF (A(IL).GT.6.50) A(IL) = 6.50 IF (A(IL).LT.0.0) A(IL) = 0.0 IF (AD(IL).GT.6.50) AD(IL) = 6.50 IF (AD(IL).LT.0.0) AD(IL) = 0.0 IF (AE(IL).GT.6.50) AE(IL) = 6.50 IF (AE(IL).LT.0.0) AE(IL) = 0.00 CONTINUE
7005	IF (KCA.EQ.KCB) GO TO 7110 IF (ISCL.GT.0) GO TO 7104 CALL SCALE(B,MAXIF,5.00,XMIN2,DX2,1) CONTINUE
7104	DO 7105 IL = 1,MAXIF IF (ISCL.GT.0) B(IL) = (B(IL)-XMIN2)/DX2 BD(IL) = (BD(IL) - XMIN2)/DX2 BE(IL) = (BE(IL)-XMIN2)/DX2 IF (B(IL).GT.6.50) B(IL) = 6.50 IF (B(IL).LT.0.0) B(IL)=0.0 IF (BD(IL).GT.6.50) BD(IL)=6.50 IF (BD(IL).LT.0.0) BD(IL) = 0.0 IF (BE(IL).GT.6.50) BE(IL) = 6.50 IF (BE(IL).LT.0.0) BE(IL) = 0.0 CONTINUE
7105	KC = KCA
7110	ADDY = 0.0


```

7112 DO 7113 ID = 1,9
      ADDX = ID-1
      XS = 2.00+(0.48*ADDX)
      YS = 2.75 - ADDY
      CALL SYMBOL(XS,YS,0.14,ACHNAM(ID,KC),0.00,4)
      CONTINUE
7113 IF (KC.EQ.KCB) GO TO 7120
      KC = KCB
      ADDY = 0.25
      GO TO 7112
7120 IF(IFRSCLE.GT.0) GO TO 7122
      IFRSCL=1
      IF(IAXIS.GT.0) GO TO 7121
      CALL SCALE(POS,MAXIF,4.00,XMIN3,DX3,1)
      DO 71215 KIF=1,MAXIF
7121 IF(IAXIS.NE.0) POS(KIF) = (PCS(KIF)-XMIN3)/DX3
      POSU(KIF) = (POSU(KIF)-XMIN3)/DX3
      POSL(KIF) = (POSL(KIF)-XMIN3)/DX3
      IF(POS(KIF).GT.FRAXLN) POS(KIF) = FRAXLN
      IF(PCS(KIF).LT.0.0) POS(KIF) = 0.0
      IF(POSU(KIF).GT.FRAXLN) POSU(KIF)=FRAXLN
      IF(POSU(KIF).LT.0.0) POSU(KIF)=0.0
      IF(POSL(KIF).GT.FRAXLN) POSL(KIF) = FRAXLN
      IF(POSL(KIF).LT.0.0) POSL(KIF) = 0.0
      CONTINUE
71215 IF(LINOCLE.EQ.1) GO TO 71225
7122 GO TO (7130,7130,71225,71225,7130,7130,71225,71225),IPLOT
71225 CALL AXIS(2.00,3.50,15HLOG10 FREQUENCY,-15,FRAXLN,0.0,XMIN3,DX3)
      GO TO 7140
7130 CALL AXIS(2.00,3.50,9HFREQUENCY,-9,FRAXLN,0.0,XMIN3,DX3)
7140 IF(KCA.EQ.KCB) GO TO 7300
      IF (LPL.GT.1) GO TO 7170
7142 GO TO (7150,7145,7145,7142,7155,7155,7155),IPLOT
      CALL AXIS(2.00,3.50,13H FREQ X COSP ,+19,5.00,90.0,XMIN1,
1 DX1)
      GO TO 7180
7145 CONTINUE
      CALL AXIS(2.00,3.50,17HLOG10 CO-SPECTRUM,+17,5.00,90.0,XMIN1,
1 DX1)
      GO TO 7180
7150 CALL AXIS(2.00,3.50,11HCO-SPECTRUM,+11,5.00,90.0,XMIN1,DX1)
      GO TO 7180
7155 CALL AXIS(2.00,3.50,9HCOHERENCE,+9,5.00,90.0,XMIN1,DX1)
      GO TO 7180
7170 GO TO (7175,7171,7171,7175,7172,7172,7172,7172),IPLOT
71715 CALL AXIS(2.00,3.50,13H FREQ X QUSP ,+19,5.00,90.0,XMIN2 ,
1 DX2)
      GO TO 7180

```

```

7171      CONTINUE
      CALL AXIS(2.00,3.50,19HLOG10 QUAD-SPECTRUM,+19,5.00,90.0,XMIN2,
1 DX2)
      GO TO 7180
7175      CALL AXIS(2.00,3.50,13HQAD-SPECTRUM,+13,5.00,90.0,XMIN2,DX2)
      GO TO 7180
7172      CALL AXIS(2.00,3.50,5HPHASE,+5,5.00,90.0,XMIN2,DX2)
7180      IF(IPLT.EQ.4) GO TO 7181
      CALL LABEL(AUNITS(1,KCA),AUNITS(1,KCB),2)
7181      CONTINUE
      IF (LPL.GT.1) GO TO 7190
      CALL PLVAL(A,AD,AE,POS,POSU,POSL,MAXIF)
      LPL = 2
      CALL PLOT (0.,YREORG,-3)
      GO TO 7110
7190      CALL PLVAL(B,8D,BE,POS,POSU,POSL,MAXIF)
      LPL = 1
      GO TO 2800
7300      GO TO (7310,7305,7302,7310,7305,7302
7302      1 DX1 ),IPLT
      CALL AXIS(2.00,3.50,14H FREQ X SPECT ,+20,5.00,90.0,XMIN1,
      GO TO 7320
7305      CONTINUE
      CALL AXIS(2.00,3.50,14HLOG10 SPECTRUM,+14,5.00,90.0,XMIN1,DX1)
      GO TO 7320
7310      CALL AXIS(2.00,3.50,8HSPECTRUM,+8,5.00,90.0,XMIN1,DX1)
7320      IF(IPLT.EQ.4.OR.IPLT.EQ.8) GO TO 7327
      CALL LABEL(AUNITS(1,KCA),AUNITS(1,KCB),1)
7327      CALL PLVAL(A,AD,AE,POS,POSU,POSL,MAXIF)
2800      CALL PLOT (0.,YREORG,-3)
2900      CONTINUE
3000      CONTINUE
      GO TO 5000
4700      WRITE (6,4705) MBLOCK,IBSTAR,IBMAX,IFILE
      GO TO 5000
4800      WRITE (6,4805) IHARM,TAPRAY(I SUB+1), NT, IH, KCHAN
      GO TO 5000
4900      WRITE (6,4905) IFILE,IDUSER,IDTEST
5000      GO TO 1
5020      IF (LOPT.LE.0) GO TO 5015
      CALL PLOT
      REWIND 3
5015      CALL EXIT
      FORMAT(7F10.0)
5      FORMAT(18A4)
6      FORMAT(45H0 FOURIER COEFFICIENTS WERE HANNED BY FTOR /)
14      FORMAT(19,1X,6(15,5X))
15      FORMAT(15,2F10.0,15)
16

```



```

5005  FORMAT (37H0 RUN TERMINATED IN AN ORDERLY MANNER )
      CONTINUE
      END

```

```

      SUBROUTINE PLVAL(A,AD,AE,POS,POSU,POSL,NVAL)
      C  SPLOT = PLVAL AND LABEL
      C  AUGUST 6 1968
      DIMENSION A(1),AE(1),AD(1),POS(1),POSU(1),POSL(1)
      DO 20 K = 1,NVAL
      Y = A(K) + 3.50
      X = POS(K) + 2.00
      XP = POSL(K) + 2.00
      CALL PLCT(XP,Y,+3)
      XP = POSU(K) + 2.00
      CALL PLOT (XP,Y,+2)
      Y = AD(K) + 3.50
      CALL PLOT(X,Y,+3)
      CALL TIC(X,Y)
      Y = AE(K) + 3.50
      CALL PLOT (X,Y,+2)
      CALL TIC(X,Y)
      CONTINUE
      RETURN
      END

```

JOHN GARRETT

20

```

      SUBROUTINE TIC(X,Y)
      X = X+0.02
      CALL PLOT(X,Y,+2)
      X = X-0.04
      CALL PLOT(X,Y,+2)
      X = X+0.02
      CALL PLOT(X,Y,+2)
      RETURN
      END

```

C SUBROUTINE LABEL (U1,U2,ISY)
C VERSION OF APRIL 26 1968

JOHN GARRETT

DIMENSION U1(2),U2(2)

X = 1.70

Y = 7.30

HT = 0.14

CALL SYMBOL(X,Y,HT,1H(,90.0,1)

Y = Y+0.12

CALL SYMBOL(X,Y,HT,U1,90.0,8)

Y = Y+1.00

GO TO (10,20), ISY

CALL SYMBOL(X,Y,HT,1HX,90.0,1)

Y = Y+0.20

CALL SYMBOL(X,Y,HT,U2,90.0,8)

Y = Y+0.96

GO TO 30

XSUP = X-0.10

CALL SYMBOL(XSUP,Y,HT,1H2,90.0,1)

Y = Y+0.18

CALL SYMBOL(X,Y,HT,7H/HERTZ),90.0,7)

RETURN

END

SUBROUTINE PHASES(N,A,CO,QU,IFB,F,ND,BW,PHASE,KCA,KCB,FFR,ICM,
1 MCHAN, MAXIF, RAD, KCHA)

DIMENSION A(1),F(1),ND(1),BW(1),MCHAN(1),PHASE(50,1),PHCOR(50,1),
1 SUMP(12), MWR(10), KCHA(1)

KTRL = N + 1,100,500) , KTRL

GO TO (10,100,500) , KTRL

DO 9800 MZERO = 1,48

DO 9800 NZERO = 1,11

PHCOR(MZERC,NZERO) = 0.0

CONTINUE

MFRE = 0

DO 9820 KARD = 1,6

READ(5,9825) (PHASE(JFRE,1), JFRE = 1,8)

DO 9820 LFRE = 1,8

MFRE = MFRE + 1

FIN = PHASE(LFRE,1)

IF(FIN.EQ.0.) GO TO 9830

PHCOR(MFRE,1) = FIN

CONTINUE

NFR = 48

```

9830 GO TO 9835
9835 NFR = MERE - 1
      MCH = ICM + 1
      DO 9840 NCHS = 2, MCH
      MCOR = 0
      DO 9840 KRD = 1, KARD
      READ( 5, 9825 ) ( PHASE(JFRE, NCHS), JFRE = 1, 8 )
      DO 9840 LCR = 1, 8
      MCOR = MCOR + 1
      PHCOR( MCOR, NCHS ) = PHASE( LCR, NCHS )
9840 CONTINUE
      RETURN
100  NPH = 1
      DO 450 I = 1, MAXIF
      DO 110 LS = 1, 12
      SUMP(LS) = 0.
110  CONTINUE
      ANIB = BW(I)/FFR
      NIB = ANIB + 0.000001
      IF( NIB.EQ.0 ) NIB = 1
      PHASE(I, 1) = F(I)
      B = ND(I) - NIB
      DO 400 J = 1, NIB
      D = J
      C = B + D
      FRS = C*FFR
      FRS = PHCOR(NPH, 1 )
      IF( FRS.GT.FR ) GO TO 400
      NPL = NPH + 1
      IF( NPL.GT.NFR ) GO TO 460
      FRL = PHCOR( NPL, 1 )
      IF( FR.GT.FRL ) GO TO 130
      GO TO 140
130  NPH = NPH + 1
      GO TO 120
140  FRD = FRL - FRS
      FDIF = FR - FRS
      FRFR = FDIF/FRD
      DO 390 M = 2, MCH
      PHIL = PHCOR(NPL, M)
      PHIS = PHCOR(NPH, M)
      PHID = PHIL - PHIS
      SUMP(M) = SUMP(M) + PHIS + PHID*FRFR
390  CONTINUE
400  IF( ANIB.EQ.0. ) ANIB = 1.
      DO 450 MP = 2, MCH
      PHASE( I, MP ) = SUMP( MP )/(ANIB)
450  CONTINUE

```



```

460      GO TO 600
      IAD = I
      DO 480 IZ = IAD, MAXIF
      PHASE( IZ, 1 ) = F( IZ )
      DO 480 MZ = 2, MCH
      PHASE( IZ, MZ ) = 0.
      CONTINUE
480      WRITE( 6, 55 )
600      DO 490 K = 1, 10
      KY = KCHA( K )
      IF( KY.EQ.0 ) GO TO 495
      MWR( K ) = MCHAN( KY )
      GO TO 490
495      MWR( K ) = 0
490      CONTINUE
      WRITE( 6, 10904 ) ( MWR( KCS ), KCS = 1, 10 )
      DO 50 IWR = 1, NFR
      WRITE( 6, 10915 ) ( PHCOR( IWR, JWR ), JWR = 1, MCH )
      CONTINUE
50      WRITE( 6, 10905 ) ( MWR( KCS ), KCS = 1, 10 )
      DO 601 IRO = 1, MAXIF
      WRITE( 6, 10915 ) ( PHASE( IRO, JCO ), JCO = 1, MCH )
      CONTINUE
601      RETURN
500      ARG = QU/CO
      KCA = KCA + 1
      KDB = KCB + 1
      PHICOR = PHASE( IFB, KDA ) - PHASE( IFB, KDB )
      IF( ARG.GT.10000. ) GO TO 9300
      BNG = ATAN2( QU, CO ) + PHICOR
      GO TO 9400
9300      BNG = ( 1.5708 ) * SIGN( 1., QU ) + PHICOR
9400      TB = TAN( BNG )
      SB = SIN( BNG )
      QU = RAD*SB
      CO = RAD*SB/TB
      RETURN
55      FORMAT( 1H0, 93H FREQUENCIES ENCOUNTERED WHICH ARE GREATER THAN HIGH
      1EST AVAILABLE PHASE CORRECTION FREQUENCY./57H CORRECTIONS ABOVE T
      2HAT FREQUENCY HAVE BEEN SET TO ZERO. )
9825      FORMAT( 8F10.4 )
10904      FORMAT( 1H0, 52X, 26HPHASE CORRECTIONS ON CARDS // 1X, 10HFREQ.(HZ.) ,
      15X, 4HCH. , 12, 5X, 4HCH. , 12, 5X, 4HCH. , 12, 5X, 4HCH. , 12,
      25X, 4HCH. , 12, 5X, 4HCH. , 12, 5X, 4HCH. , 12, 5X, 4HCH. , 12, / )
10905      FORMAT( 1H0, 50X, 28HPHASE CORRECTIONS IN RADIANS// 1X, 10HFREQ.(HZ.) ,
      15X, 4HCH. , 12, 5X, 4HCH. , 12, 5X, 4HCH. , 12, 5X, 4HCH. , 12,
      25X, 4HCH. , 12, 5X, 4HCH. , 12, 5X, 4HCH. , 12, 5X, 4HCH. , 12, / )
10915      FORMAT( 1X, 1PE10.2, 9( 3X, 0PF8.4 ) )

```

```

C
C
C
C
C
C
C
END
THIS SUBROUTINE WHEN CALLED MOVES THE TAPE ON LOGICAL UNIT NUNIT
PAST KSKIP END OF FILE MARKS. THE RECORDS SKIPPED OVER MUST BE IN
FORTRAN BINARY. IF KSKIP IS ZERO OR NEGATIVE THE ROUTINE RETURNS
WITHOUT MOVING THE TAPE.

```

```

SUBROUTINE SKPFL (KSKIP,NUNIT)
IF (KSKIP.LT.1) RETURN
DO 20 I=1,KSKIP
10 READ (NUNIT,END=20) IDUMMY
20 CONTINUE
RETURN
END

```

```

SUBROUTINE SCALE (X,N,S,YMIN,DY,K)
DIMENSION X(2)
LOGICAL FLAG
FLAG = .FALSE.
YMAX=X(1)
YMIN=YMAX
NP=N*K
DO 10 I=1,NP,K
IF (YMAX-X(I)) 5,6,6
5 YMAX=X(I)
6 IF (X(I)-YMIN) 7,10,10
7 YMIN=X(I)
10 CONTINUE
105 RANGE = YMAX - YMIN
IF (RANGE.NE.0.0) GO TO 11
DY = 0.0
DO 12 I = 1,NP,K
12 X(I) = 0.0
RETURN
11 SFAC = RANGE / (S)
SFLOG = ALOG10(SFAC)
IF (SFLOG.GE.0.0) GO TO 13
13 SFLOG = SFLOG - 1.
ISFLOG = SFLOG
ISFAC = SFAC / 10.** ISFLOG

```

SCAL0020
SCAL0030
SCAL0034
SCAL0036
SCAL0040
SCAL0050
SCAL0060
SCAL0070
SCAL0080
SCAL0090
SCAL0100
SCAL0110
SCAL0120
SCAL0130
SCAL0140
SCAL0150
SCAL0160
SCAL0170
SCAL0180
SCAL0190
SCAL0200
SCAL0210
SCAL0220
SCAL0230
SCAL0240

SCAL0250
SCAL0260
SCAL0270
SCAL0280
SCAL0290
SCAL0300
SCAL0310
SCAL0320
SCAL0330
SCAL0340
SCAL0350
SCAL0360
SCAL0370
SCAL0380
SCAL0390
SCAL0400
SCAL0410
SCAL0415
SCAL0420
SCAL0430
SCAL0440
SCAL0444
SCAL0446
SCAL0447
SCAL0448
SCAL0450
SCAL0460
SCAL0470
SCAL0480

```

IF(SFACT.GT.1.0) GO TO 14
DY = 10.** ISFLOG
GO TO 15
14 IF(SFACT.GT.2.0) GO TO 16
DY = 2.** ISFLOG
GO TO 15
16 IF(SFACT.GT.4.0) GO TO 17
DY = 4.** ISFLOG
GO TO 15
17 IF(SFACT.GT.5.0) GO TO 18
DY = 5.** ISFLOG
GO TO 15
18 IF(SFACT.GT.8.0) GO TO 19
DY = 8.0 * 10.** ISFLOG
GO TO 15
19 DY = 10.** ISFLOG + 1)
TEMP = YMIN / DY
IF( FLAG ) GO TO 21
IF(TEMP.LT.0.) TEMP = TEMP - 1.0
IYMIN = TEMP
YMIN = FLOAT(IYMIN)* DY
FLAG = .TRUE.
TEMP = YMIN + S * DY
C SCALE AGAIN IF THE RANGE HAS INCREASED TOO MUCH
IF( YMAX .GT. TEMP ) GO TO 105
DO 20 I=1,NP,K
21 20 X(I)=(X(I)-YMIN)/DY
RETURN
END

```

SUBROUTINE AXIS(X,Y,BCD,NC,SIZE,THETA,YMIN,DY)

```

DIMENSION BCD( 2 )
SIG = 1.0
IF( NC ) 1 , 2 , 2
SIG = -1.0
NAC = IABS( NC )
TH = THETA * 0.017453294
N = SIZE + 0.5
TN = N
CTH = COS( TH )
STH = SIN( TH )
SCALE THE AXIS *****
ADY = DY
ABSU = YMIN
EXP = 0.0
C*****

```

```

9      IF( ADY ) 9, 18, 9
12     IF( ABS( ADY ) - 100.0 ) 10, 12, 12
      ABSU = ABSU * 0.1
      EXP = EXP + 1.0
      GO TO 9
14     ADY = ADY * 10.0
      ABSU = ABSU * 10.0
      EXP = EXP - 1.0
10     IF( ABS( ADY ) - 0.01 ) 14, 18, 18
C$**** INITIALIZE THE MAIN LOOP *****
C      XL, YL ETC. ARE FOR THE LINE
C      XN, YN ETC. ARE FOR THE NUMBERS
C      XT, YT ETC. ARE FOR THE TITLE
C$****
18     XS = X
      YS = Y
      TEMP = 0.2*SIG - 0.05
      XNA = XS - TEMP*STH - 0.0857*CTH
      YNA = YS + TEMP*CTH - 0.0857*STH
      NAFT = 3
      ICK = ABS( ABSU ) * 1000.0 + 0.5
      IF( MOD( ICK, 10 ) .EQ. 0 ) NAFT = 2
      IF( MOD( ICK, 100 ) .EQ. 0 ) NAFT = 1
      CALL NUMBER( XNA, YNA, 0.10, ABSU, THETA, NAFT )
      XLB = XS + IN * CTH
      YLB = YS + IN * STH
      TEMP = 0.1*SIG
      XLA = XLB - TEMP * STH
      YLA = YLB + TEMP * CTH
      NBY2 = N/2
C$**** THIS IS THE MAIN LOOP *****
      DO 20 I = 1, N *****
C$**** NEXT THE NUMBER *****
      XNA = XNA + CTH
      YNA = YNA + STH
      ABSU = ABSU + ADY
      NAFT = 3
      ICK = ABS( ABSU ) * 1000.0 + 0.5
      IF( MOD( ICK, 10 ) .EQ. 0 ) NAFT = 2
      IF( MOD( ICK, 100 ) .EQ. 0 ) NAFT = 1
      CALL NUMBER( XNA, YNA, 0.10, ABSU, THETA, NAFT )
C$**** NOW PERHAPS THE TITLE *****
      IF( I .NE. NBY2 ) GO TO 19
      TNC = NAC + 7
      TEMP = SIZE*0.5 - 0.06*TNC
      TEMPB = (-0.07 + SIG*0.36)
      XT = XS + TEMP*CTH - TEMPB*STH

```

176

PRINTER PLOT OF AMPLITUDE FROM FOURIER COEFFICIENT TAPE

THE FIRST INPUT CARD SHOULD CONTAIN UP TO 72 CHARACTERS OF PLOT IDENTIFICATION FOR USE BY THE COMPUTING CENTER STAFF AND IS REPRODUCED ON THE PLOTTER OUTPUT. THIS CARD MUST BE PRESENT IN THE DECK, THE FOLLOWING SET OF CARDS IS PRESENT ONLY ONCE IN THE DECK, THE FOURIER COEFFICIENTS PROCESSED.

CONTROL CARD DECK MAKEUP

1) KFILE,KCHAN,KSTART,KSTCP,KPLOT,KRUL,KLIST,KPUNCH,LOGLIN,
LOGLOG (1915)

WHERE,

KFILE = NO. OF FILE ON FOURIER COEFFICIENT TAPE

KCHAN = NO. OF FOURIER COEFFICIENT TAPE CHANNEL DESIRED

KSTART = FIRST BLOCK TO BE INCLUDED IN ANALYSIS

KSTOP = LAST BLOCK TO BE INCLUDED IN ANALYSIS

KPLOT = 1 FOR CALCOMP PLOT
KPLOT = 0 FOR NO CALCOMP PLOT

KRULKE = 1 FOR A COLUMN OF PERIODS PER INCH ON THE PLOT
= 0 IF THIS IS NOT DESIRED

```
KLIST = 1 FOR A PRINTER PLOT
        = 0 TO SUPPRESS THE PRINTER PLOT
```

KPUNCH = 1 TO PUNCH A BINARY DECK OF THE INTERMEDIATE SUMS
= 0 OTHERWISE

LOGLIN = 1 TO OBTAIN A LOG VS LINEAR SPECTRUM PLOT
LOGLIN = 0 OTHERWISELOGLOG = 1 TO OBTAIN A LOG LOG SPECTRUM PLOT
= 0 OTHERWISE

2) TITLE CARD WITH UP TO 72 COLUMNS OF IDENTIFICATION FOR PRINTER OUTPUT AND PLOTS

THE FOLLOWING CARD IS PRESENT ONLY IF LOGLIN = 1 ON CARD 1)

3) NDCDE,NPDCDE,FSTART,DPINCH (2I10,2F10.0)

WHERE,

NDCDE = NO. OF FREQUENCY DECADES TO BE PLOTTED

NPDCDE = MAXIMUM DESIRED NO. OF SPECTRAL ESTIMATES PER DECADE

FSTART = ANOTATION TO APPEAR ON FIRST FREQUENCY DECADE ON
PLOT

DPINCH = NO. OF FREQUENCY DECADES TO BE PLOTTED PER INCH

THE FOLLOWING CARD IS PRESENT ONLY IF LOGLOG = 1 ON CARD 1)

4) NDCDE,NPDCDE,FSTART,DPINCH,NYDCDE,DY (2110,2F10.0,110,F10.0)

WHERE,

NDCDE = NO. OF FREQUENCY DECADES TO BE PLOTTED

NPDCDE = MAXIMUM DESIRED NO. OF SPECTRAL ESTIMATES PER DECADE

FSTART = ANOTATION TO APPEAR ON FIRST FREQUENCY DECADE ON
PLOT

DPINCH = NO. OF FREQUENCY DECADES TO BE PLOTTED PER INCH

NYDCDE = NO. OF DECADES TO BE PLOTTED ON SPECTRAL DENSITY
AXIS

DY = NO. OF DECADES PER INCH TO BE PLOTTED ON SPECTRAL
DENSITY AXIS

INPUT TAPE IS ON LOGICAL UNIT 9

A BLANK CARD WILL TERMINATE THE RUN OR ANOTHER COMPLETE SET
OF CARDS WILL DO A SECOND ANALYSIS

THE PROGRAM DOES AN ANALYSIS ON THE FOURIER COEFFICIENTS
DEFINED BY THE FIRST CONTROL CARD. THE FOURIER COEFFICIENT NO.,
ITS FREQUENCY, ITS MEAN AMPLITUDE AND ITS 95 PERCENT CONFIDENCE
INTERVAL ARE PRINTED. THE MEAN AND 95 CONFIDENCE INTERVAL ARE
PLOTTED ON THE PRINTER. ONLY THE MEAN IS PLOTTED ON THE CALCOMP
PLOTTER.

THE PROGRAM WILL HANDLE ONLY ONE CHANNEL AT A TIME FROM THE
FOURIER COEFFICIENT TAPE

C

```

DIMENSION DATA(4098),AMPL(2049),VAR(2049),KPLOT(101),KTITLE(18),AP
1  LQOT(3),VAR2(2049),JUNITS(2),JDUMMY(2),GTITLE(18)
  KUNIT = 9
  KMODE = 1
  KPCALL = 0
  NPAGE = 0
  JRCHAN = 1
  REWIND KUNIT
  IFILE = 1
  KBEGIN = 1
5  READ (5,2002) GTITLE
  READ (5,2001) KFILE,KCHAN,KSTART,KSTOP,KPLOTI,KRULE,KLIST,KPUNCH,L
  LOGLIN,LOGLOG,LPUNCH
  IF (KFILE.LE.0) GO TO 135
  NPAGE = NPAGE + 1
  WRITE (6,4001) NPAGE,KCHAN,KSTART,KSTOP
  IF (KPLCTI.EQ.1) WRITE (6,4003)
  IF (KLIST.EQ.1) WRITE (6,4011)
  IF (KPUNCH.EQ.1) WRITE (6,4017)
  IF (LOGLIN.EQ.1) WRITE (6,4015)
  IF (LOGLOG.EQ.1) WRITE (6,4016)
  READ (5,2002) (KTITLE(I),I=1,18)
  WRITE (6,4007) (KTITLE(I),I=1,18)
6  KSKIP = KFILE - IFILE
  IF (KSKIP.GT.0) GO TO 8
  IF (KSKIP.LT.0) GO TO 7
  IF (KBEGIN.EQ.1) GO TO 9
7  REWIND KUNIT
  IFILE = 1
  KBEGIN = 1
  GO TO 6
8  CALL SKPFL (KSKIP,KUNIT)
  IFILE = IFILE + KSKIP
  KBEGIN = 1
9  CALL FCINPT (-KUNIT,IDUSER,KBLOCK,NSAMPL,JRCHAN,KCHAN,DFREQ,JUNITS
1  ,DATA,IND)
  IF (IND.EQ.1) GO TO 160
  KSKIP = KSTART - 1
  IF (KSKIP.LE.0) GO TO 15
  DO 10 I=1,KSKIP
  CALL FCINPT (KUNIT,IDUSER,KBLOCK,KSAMPL,JRCHAN,KCHAN,XFREQ,JDUMMY,
  1  IDATA,IND)
  KBEGIN = 0
  WRITE (6,4018) I
  IF (IND.EQ.1) GO TO 160
10 CONTINUE

```



```

15 KPTS = (NSAMPL + 2)/2
   IF (KPTS.GT.2049) GO TO 170
   DO 20 I=1,KPTS
     AMPL(I) = 0.
     VAR(I) = 0.
20   VAR2(I) = 0.
     LBLOCK = 0
     CALL FCINPT (KUNIT, IDUSER, KBLOCK, KSAMPL, JRCHAN, KCHAN, XFREQ, JDUMMY,
1     IDATA, IND)
     KBEGIN = 0
     WRITE (6,4019) I
     IF (IND.EQ.1) GO TO 180
     LBLOCK = LBLOCK + 1
     DO 30 J=1,KPTS
       A = DATA(2*J-1)
       B = DATA(2*J)
       X = A*A + B*B
       AMPL(J) = AMPL(J) + Sqrt(X)
       VAR(J) = VAR(J) + X
30   VAR2(J) = VAR2(J) + X*X
40   WRITE (6,4009) LBLOCK
       IF (KPUNCH.EQ.0) GO TO 45
       WRITE (7,2002) KPTS, LBLOCK
       WRITE (7,2002) (AMPL(J), J=1, KPTS)
       WRITE (7,2002) (VAR(J), J=1, KPTS)
       WRITE (7,2002) (VAR2(J), J=1, KPTS)
45   XBLOCK = LBLOCK
     DO 50 J=1,KPTS
       AMPL(J) = AMPL(J)/XBLOCK
       VAR(J) = VAR(J)/XBLOCK
50   VAR2(J) = VAR2(J)/XBLOCK
       IF (KPCALL.EQ.1) GO TO 52
       IF (LOGLIN.EQ.0.AND.LOGLOG.EQ.0) GO TO 55
       CALL PLOTS
       CALL SYMBOL (0,0,0,0,14,GTITLE,0,72)
       CALL PLOT (0,3,-3)
       KPCALL = 1
52   IF (LCGLIN.EQ.1) CALL LOGPLT (FLCAT(NSAMPL), DFREQ, VAR(2), VAR2(2), X
1     BLOCK, KTITLE, MOFLOW)
       IF (LCGLOG.EQ.1) CALL LLPLOT (FLOAT(NSAMPL), DFREQ, VAR(2), VAR2(2), X
1     BLOCK, KTITLE, MOFLOW, LPUNCH)
55   DO 56 J=1,KPTS
56   VAR(J) = 1.96*SQRT((VAR(J) - AMPL(J)*AMPL(J))/(XBLOCK - 1.))
       CALL SCALE (AMPL(2), KPTS-1, 10, XMIN, DX, 1)
       WRITE (6,4014) AMPL(1), JUNITS(1), JUNITS(2), VAR(1), JUNITS(1), JUNITS
1(2)
       IF (KLIST.EQ.0) GO TO 132

```

```

CALL REPLCE (KPERD,1H.)
CALL REPLCE (KRECMK,1H )
CALL REPLCE (KAST,1H*)
CALL REPLCE (KMINUS,1H-)
CALL REPLCE (KBLANK,1H )
XSAMPL = NSAMPL
XMAX = 10.*DX + XMIN
WRITE (6,4010) XMIN,(JUNITS(K),K=1,2),XMAX,(JUNITS(K),K=1,2),DX,(J
1UNITS(K),K=1,2)
KPAGE = 0
CALL GRHEAD (100,KPAGE)
DO 130 I=2,KPTS
NCOEFF = I - 1
FREQ = NCOEFF
FREQ = FREQ*DFREQ/XSAMPL
XAMPL = AMPL(I)*DX + XMIN
XVAR = VAR(I)
SVAR = VAR(I)/DX
APLOT(1) = AMPL(I) - SVAR
APLOT(2) = AMPL(I) + SVAR
APLOT(3) = AMPL(I) + SVAR
KMIN = 10.*APLOT(1) + 1.5
KPOINT = 10.*APLOT(2) + 1.5
KMAX = 10.*APLOT(3) + 1.5
KERRP = KBLANK
IF (KMIN.LT.1.OR.KMAX.GT.101) KERRP = KRECMK
KI = KMIN + 1
KF = KMAX - 1
IF (KI.LT.1) KI = 1
IF (KF.GT.101) KF = 101
DO 60 INDEX=1,101
KPLOT(INDEX) = KBLANK
KPLOT(1) = KPERD
KPLOT(101) = KPERD
IF (KRULE.EQ.0) GO TO 80 + 1
LS = ((KF-1)/10 + 1)*10 + 1
IF (LS.GT.91) GO TO 80
DO 70 INDEX = LS,101,10
KPLOT(INDEX) = KPERD
IF (KI.GT.KF) GO TO 100
DO 90 INDEX=KI,KF
KPLOT(INDEX) = KMINUS
IF (KMIN.GT.0) KPLOT(KMIN) = KAST
KPLOT(KPOINT) = KAST
IF (KMAX.LT.102) KPLOT(KMAX) = KAST
WRITE (6,4004) NCOEFF,FREQ,XAMPL,XVAR,KERRP,(KPLOT(K),K=1,101)
CALL GRHEAD (1,KPAGE)
130 IF (KPLOT1.EQ.1) GO TO 140

```

```

135 GO TO 5
    REWIND KUNIT
    IF (KPCALL.EQ.1) CALL PLOT (0.,3.,-3)
    IF (KPCALL.EQ.1) CALL PLOTE
    WRITE (6,4005)
    WRITE (6,4006)
    CALL EXIT
140 IF (KPCALL.EQ.1) GO TO 145
    CALL PLOTS
    CALL SYMBOL (C,0.,0.14,GTITLE,0.,72)
    CALL PLOT (0.,3.,-3)
    KPCALL = 1
145 S = KPTS/100 + 1
    IF (S.GT.9.) S = 9.
    IF (LGGLOG.EQ.1.OR.LOGLIN.EQ.1) CALL PLOT (0.,2.,-3)
    CALL AXIS (0.,0.,19HFOURIER COEFFICIENT,-19,S,0.,0.,100.)
    CALL AXIS (0.,0.,19HAMPLITUDE,9,10.,90.,XMIN,DX)
    CALL SYMBOL (0.,10.1,0.14,KTITLE,0.,72)
    KPTS = KPTS
    IF (KGPTS.GT.900) KGPTS = 900
    IPEN = 3
    DO 150 I=2,KGPTS
      X = I - 1
      X = 0.01*X
      CALL PLOT (X,AMPL(I),IPEN)
150 IPEN = 2
      CALL PLOT (0.,14.,-3)
      GO TO 5
160 WRITE (6,4008)
170 GO TO 5
170 WRITE (6,4012)
180 GO TO 5
    IFILE = IFILE + 1
    KBEGIN = 1
    GO TO 40
2001 FORMAT (11I5)
2002 FORMAT (18A4)
4001 FORMAT (58H1 AVERAGED AMPLITUDE SPECTRUM FROM FOURIER COEFFICIENT T
1APE,65X,4HPAGE,13)
4002 FORMAT (9HOF FILE NO.,13/15H A TO D CHANNEL,13/27H ANALYSIS STARTS W
11TH BLOCK,15,20H AND ENDS WITH BLOCK,15)
4003 FORMAT (28HOA CALCOMP PLOT IS REQUESTED)
4004 FORMAT (15,F8.2,1PE9.2,E8.1,102A1)
4005 FORMAT (11HOEND OF JOB)
4006 FORMAT (1H1)
4007 FORMAT (1H0,18A4)
4008 FORMAT (45HOUNEXPECTED END OF FILE WHILE SPACING FORWARD)
4009 FORMAT (1H0,15,16H BLOCKS ANALYSED)

```

```

4010 FORMAT (25HOPRINTER PLOT DEFINITIONS//4H 0 =,1PE10.3,1H ,2A4/6H 10
10 =,E10.3,1H ,2A4/9H 1 INCH =,E10.3,1H ,2A4/)
4011 FORMAT (28H0A PRINTER PLOT IS REQUESTED)
4012 FORMAT (52H0THE MAXIMUM TRANSFORM SIZE (4096) HAS BEEN EXCEEDED)
4014 FORMAT (18H0MEAN OF SERIES IS,1PE11.3,1H ,2A4//60H 95 PER CENT CON
1FIDENCE INTERVAL ABOUT MEAN IS PLUS OR MINUS,E11.3,1H ,2A4/)
4015 FORMAT (34H0A LOG VS LINEAR PLOT IS REQUESTED)
4016 FORMAT (28H0A LOG LOG PLOT IS REQUESTED)
4017 FORMAT (10AN EBCD DECK OF SUMMED COEFFICIENTS IS REQUESTED')
4018 FORMAT (18H SKIPPING IN BLOCK,I4)
4019 FORMAT (20H PROCESSING IN BLOCK,I4)
END

```

```

SUBROUTINE LLPLOT (XSAMPL,DFREQ,DATA,VAR,XBLOCK,NXANOT,MOFLCW,LPU
1CH)
DIMENSION DATA(1),VAR(1),Y(100),NC(100),NXANOT(1)
INTEGER YERROR
READ (5,2001) NDCODE,NPDCODE,FSTART,EPINCH,NYDCODE,DY
WRITE (6,4001) (NXANOT(K),K=1,18)
MOFLOW = 0
FACR = DFREQ/XSAMPL
HBAND = 0.5*FACR
NCMAX = IFIX(XSAMPL)/2
XDECODE = ALOG10(XSAMPL*FACR/(2.*HBAND) + 1.)
NPTS = XDECODE*FLOAT(NPDCODE) + 0.5
FLINCR = XDECODE/FLOAT(NPTS)
I = 1
ISTART = 1
NXTMIN = 0
FBASE = ALOG10(HBAND)
DO 30 K=1,NPTS
NCOEFF = (10.*(FBASE + FLINCR*FLOAT(K)))/FACR + 0.5
IF (NCOEFF.LT.ISTART) GO TO 30
IF (NCOEFF.LT.NXTMIN) NCOEFF = NXTMIN
IF (I.GT.100) GO TO 60
NC(I) = NCOEFF
Y(I) = 0.
DO 10 J=ISTART,NCOEFF
Y(I) = Y(I) + DATA(J)
Y(I) = Y(I)/(2.*FLOAT(NCOEFF-ISTART+1)*FACR)
NXTMIN = 2*NCOEFF - ISTART + 1
IF (NXTMIN.GT.NCMAX) NXTMIN = NCMAX
ISTART = NCOEFF + 1
I = I + 1
30 CONTINUE

```



```

NPTS = I - 1
S = FLOAT(NDCDE)/DPINCH
CALL AXIS (0.,2.,13HLCG FREQUENCY,-13,S,0.,FSTART,DPINCH)
DENMAX = 0.
DO 32 I=1,NPTS
  IF (DENMAX.LT.Y(I)) DENMAX = Y(I)
  YLMAX = ALOG10(DENMAX)
  IF (YLMAX.LT.0.) GO TO 34
  YLMAX = FLOAT(IFIX(YLMAX + 1.))
GO TO 36
34 YLMAX = FLOAT(IFIX(YLMAX))
36 YMIN = YLMAX - FLOAT(NYDCDE)
  YULMT = 10.**YLMAX
  YLLMT = 10.**YMIN
  YLNG = FLOAT(NYDCDE)/DY
CALL AXIS (0.,2.,23HLCG OF SPECTRAL DENSITY,23,YLNG,90.,YMIN,DY)
CALL SYMBOL (0.,YLNG + 2.5,0.14,NXANOT,0.,72)
ISTART = 1
DO 50 I=1,NPTS
  NCOEFF = NC(I)
  FRLOW = FLOAT(ISTART)*FACR - HBAND
  FRHIGH = FLOAT(NCOEFF)*FACR + HBAND
  BANDW = FRHIGH - FRLOW
  FRMID = (FRHIGH + FRLOW)/2.
  XPOS = (ALOG10(FRMID) - FSTART)/DPINCH
  YERROR = 0.
  IF (Y(I).LT.YLLMT) GO TO 70
  YPOS = (ALOG10(Y(I)) - YMIN)/DY + 2.
  CALL PLOT (XPOS,YPOS,3)
  CALL PLOT ((ALOG10(FRLOW) - FSTART)/DPINCH,YPOS,2)
  CALL PLOT ((ALOG10(FRHIGH) - FSTART)/DPINCH,YPOS,2)
  CALL PLOT (XPOS,YPOS,2)
  YVAR = 0.
DO 40 K=ISTART,NCOEFF
  YVAR = YVAR + VAR(K)
  YVAR = 1.96*SQRT((YVAR/(4.*BANDW*FACR) - Y(I)*Y(I))/(FLOAT(NCOEFF
1-ISTART+1)*XBLOCK - 1.))
  YUPPER = Y(I) + YVAR
  IF (YUPPER.GT.YULMT) GO TO 80
  CALL PLOT (XPOS,(ALOG10(YUPPER)-YMIN)/DY+2.,2)
  CALL HORTIC
42 YLOWER = Y(I) - YVAR
  IF (YLOWER.LT.YLLMT) GO TO 90
  CALL PLOT (XPOS,(ALOG10(YLOWER)-YMIN)/DY+2.,2)
  CALL HORTIC
44 IF (YERROR.EQ.0) WRITE (6,4002) I,ISTART,NCOEFF,FRMID,BANDW,Y(I),Y
1VAR
  IF (YERROR.EQ.1) WRITE (6,4004) I,ISTART,NCOEFF,FRMID,BANDW,Y(I),Y

```

```

1VAR (LPUNCH.EQ.1) WRITE (7,4002) I, ISTART, NCOEFF, FRMID, BANDW, Y(I), Y
1VAR IF (YERROR.EQ.1) CALL SYMBOL (XPOS+0.03, 1., 0.07, 1HX, 0., 1)
46 ISTART = NCOEFF + 1
50 CALL PLOT (0., YLNG+3., -3)
RETURN
60 MOFLOW = 1
WRITE (6,4003)
RETURN
70 YERROR = 1
GO TO 44
80 YERROR = 1
CALL PLOT (XPOS, YLNG+2., 2)
GO TO 42
90 YERROR = 1
CALL PLOT (XPOS, 2., 2)
GO TO 44
2001 FORMAT (2I10, 2F10.0, I10, F10.0)
4001 FORMAT (55H1LOG FREQUENCY VS LOG SPECTRAL DENSITY PLOT INFORMATION
1//1H, 18A4//80H POINT START STOP MID FREQUENCY BANDWIDTH
2 SPEC. DENSITY 95 CONFID.)
4002 FORMAT (1H, I4, I8, I7, 1P4E15.4)
4003 FORMAT (40H100 MANY POINTS REQUESTED. PLOT SKIPPED)
4004 FORMAT (1H, I4, I8, I7, 1P4E15.4, 10X, 21HPLOTTER PEN OFF SCALE)
END

```

```

SUBROUTINE LOGPLT (XSAMPL, DFREQ, DATA, VAR, XBLOCK, NXANOT, MOFLOW)
DIMENSION DATA(1), VAR(1), Y(100), NC(100), YMEAN(100), NXANCT(1)
READ (5,2001) NDCDE, NPDCE, FSTART, DPINCH
WRITE (6,4001) (NXANOT(K), K=1, 18)
MOFLOW = 0
FACTR = DFREQ/XSAMPL
HBAND = 0.5*FACTR
NCMAX = IFIX(XSAMPL)/2
XDECODE = ALG10(XSAMPL*FACTR/(2.*HBAND) + 1.)
NPTS = XDECODE*FLOAT(NPDCE) + 0.5
FLINCR = XDECODE/FLOAT(NPTS)
I = 1
ISTART = 1
NXTMIN = 0
FBASE = ALG10(HBAND)
DO 30 K=1, NPTS
NCOEFF = I*10.**((FBASE + FLINCR*FLOAT(K)))/FACTR + 0.5
IF (NCOEFF.LT. ISTART) GO TO 30

```



```

IF (NCOEFF.LT.NXTMIN) NCOEFF = NXTMIN
IF (I.GT.100) GO TO 60
NC(I) = NCOEFF
Y(I) = 0.
DO 10 J=I,ISTART,NCOEFF.
  Y(I) = Y(I) + DATA(J)
  Y(I) = Y(I)/(2.*FLOAT(NCOEFF-ISTART+1)*FACR)
  YMEAN(I) = Y(I)
  NXTMIN = 2*NCOEFF - ISTART + 1
IF (NXTMIN.GT.NCMAX) NXTMIN = NCMAX
  ISTART = NCOEFF + 1
  I = I + 1
30 CONTINUE
NPTS = I - 1
S = FLOAT(NCODE)/DPINCH
CALL AXIS (0.,2.,13HLCG FREQUENCY,-13,S,0.,FSTART,DPINCH)
CALL SCALE (Y,NPTS,6.,YMIN,DY,1)
CALL AXIS (0.,2.,27HSPECTRAL DENSITY - MV**2/HZ,27,6.,90.,YMIN,DY)
CALL SYMBOL (0.,8.5,0.14,NXANOT,0.,72)
  ISTART = 1
DO 50 I=1,NPTS
  NCOEFF = NC(I)
  FRLOW = FLOAT(ISTART)*FACR - HBAND
  FRHIGH = FLGAT(NCOEFF)*FACR + HBAND
  BANDW = FRHIGH - FRLOW
  FRMID = (FRHIGH + FRLOW)/2.
  XPOS = (ALOG10(FRMID) - FSTART)/DPINCH
  YPOS = Y(I) + 2.
  CALL PLOT (XPCS,YPOS,3)
  CALL PLOT ((ALOG10(FRLOW) - FSTART)/DPINCH,YPOS,2)
  CALL PLOT ((ALOG10(FRHIGH) - FSTART)/DPINCH,YPOS,2)
  YVAR = 0.
DO 40 K=ISTART,NCOEFF
  YVAR = YVAR + VAR(K)
  1NCOEFF-ISTART+1)*XBLOCK-1.))
  IERTIC = 0
  JERTIC = 0
  YPOS = (YMEAN(I) + YVAR - YMIN)/DY + 2.
  IF (YPOS.LE.8.) GO TO 42
  YPOS = 8.
  IERTIC = 1
  42 CALL PLOT (XPOS,YPOS,2)
  IF (IERTIC.EQ.0) CALL HORTIC
  YPOS = (YMEAN(I) - YVAR - YMIN)/DY + 2.
  IF (YPOS.GE.2.) GO TO 44
  YPOS = 2.

```

```

JERTIC = 1 (XPCS, YPOS, 2)
CALL PLOT (XPCS, YPOS, 2)
IF (JERTIC.EQ.0) CALL HORTIC
IF (JERTIC.EQ.1) OR (JERTIC.EQ.1) GO TO 46
WRITE (6,4002) I, ISTART, NCOEFF, FRMID, BANDW, YMEAN(I), YVAR
GO TO 50
46 WRITE (6,4004) I, ISTART, NCOEFF, FRMID, BANDW, YMEAN(I), YVAR
CALL SYMBOL (XPOS+0.03, 1., 0.07, IHX, 1)
50 ISTART = NCOEFF + 1
CALL PLOT (0., 9., -3)
RETURN
60 MOWFLOW = 1
WRITE (6,4003)
RETURN
2001 FORMAT (2I10, 2F10.0)
4001 FORMAT (4I10, LOG FREQUENCY VS LINEAR PLOT INFORMATION, // 1H, 18A4// 8
10H POINT START STOP MID FREQUENCY BANDWIDTH SPEC. DENSI
2TY 95 CONFID.//)
4002 FORMAT (1H, 14, 18, 17, 1P4E15.4)
4003 FORMAT (40H, TOO MANY POINTS REQUESTED. PLOT SKIPPED)
4004 FORMAT (1H, 14, 18, 17, 1P4E15.4, 10X, 21HPLOTTER PEN OFF SCALE)
END

```

SUBROUTINE GRHEAD (INCRMT, NPAGE)

```

DATA LNCNT/0/
LNCNT = LNCNT + INCRMT
IF (LNCNT.LE.57) RETURN
IF (NPAGE.NE.0) WRITE (6,4001) (I, I=10, 100, 10)
NPAGE = NPAGE + 1
WRITE (6,4002) (I, I=10, 100, 10)
LNCNT = 1
RETURN
4001 FORMAT (1H, 30X, 9H0....., I2, 8(8H....., I2), 7H....., I3)
4002 FORMAT (31H1COEF FREQ AMPL 95 C , 9H0....., I2, 8(8H.....
1....., I2), 7H....., I3)
END
FOURIER COEFFICIENT INPUT SUBROUTINE

```

THIS SUBPROGRAM RETRIEVES BLOCKS OF FOURIER COEFFICIENTS FROM THE TAPE PRODUCED BY THE FTOR PROGRAM. FOR EACH CALL ONE GROUP OF K BLOCKS OF N/2 + 1 COEFFICIENTS WILL BE RETURNED IN AN ARRAY.

WHERE,

K = NO. OF CHANNELS TO BE RETURNED FROM THE FTOR TAPE AND N = NO.

CC

OF DATA PCINTS IN EACH FOURIER TRANSFORM
CALLING SEQUENCE
CALL FCINPT (NUNIT, IDUSER, KBLOCK, NSAMPL, JRCHAN, LCHN, DFREQ, JUNITS, A
1, IND)
WHERE,
NUNIT = NO. OF TAPE UNIT CONTAINING THE COCEAN TAPE
IDUSER = 9 DIGIT USER IDENTIFICATION NUMBER (RETURNED)
KBLOCK = CONSECUTIVE NUMBER OF BLOCK OF COEFFICIENTS ON THE
COCEAN TAPE (RETURNED)
NSAMPL = NO. OF POINTS IN THE FOURIER TRANSFORM (RETURNED)
JRCHAN = NO. CF TRANSFORMED CHANNELS TO BE RETURNED
LCHN = FIRST LOCATION OF AN ARRAY CONTAINING THE LOCATIONS OF
THE JRCHAN CHANNELS TO BE RETURNED
DFREQ = FREQUENCY FO DIGITAL DATA TRANSFORMED (RETURNED)
JUNITS = 8 CHARACTER NAMES OF UNITS OF DATA (RETURNED)
A = ARRAY INTO WHICH COEFFICIENTS ARE TO BE RETURNED
IND = 0 FOR NORMAL RETURN
=1 FOR END OF INPUT FILE
IF THE CALLING SEQUENCE
CALL FCINPT (-NUNIT, IDUSER, KBLOCK, NSAMPL, JRCHAN, LCHN, DFREQ, A, IND)
IS EXECUTED, TITLE INFORMATION ABOUT THE TAPE WILL BE PRINTED AND
ALL INFORMATION EXCEPT THE VECTOR A WILL BE RETURNED
THE ADDRESS OF THE FIRST COEFFICIENT FOR EACH CF THE JRCHAN
TRANSFORMS RETURNED WILL BE GIVEN BY (NSAMPL+2)*(KCHAN-1) + 1
WHERE KCHAN = NO. OF CHANNEL REQUIRED

```

SUBROUTINE FCINPT (NUNIT, IDUSER, KBLOCK, NSAMPL, JRCHAN, LCHN, DFREQ, JU
NITS, A, IND)
DIMENSION MTAPE(256), ATAPE(256), A(1), LCHN(1), JUNIT(1)
EQUIVALENCE (MTAPE(1), ATAPE(1))
KUNIT = IABS(NUNIT)
READ (KUNIT, END=70) MTAPE
IF (NUNIT.LT.0) GO TO 50
IDUSER = MTAPE(1)
KBLOCK = MTAPE(2)
NSAMPL = MTAPE(3)
DFREQ = MTAPE(6)
IND = 0
READ (KUNIT, END=70) MTAPE
K = 1
DO 30 N=1, NCOEFF
  NOLOC = 2*N - 1
  IF (K.LE.NCRSEQ) GO TO 10
  READ (KUNIT, END=70) MTAPE
  K = 1
10 NOLOC = K*KINCR + KOFF
  DO 20 J=1, JRCHAN
    JOLOC = (J-1)*JINCR + NOLOC
    JILOC = NOLOC + 2*LCHN(J)
    A(JOLOC) = ATAPE(JILOC)
20 A(JOLOC+1) = ATAPE(JILOC+1)
30 K = K + 1
  RETURN
50 IDUSER = MTAPE(1)
  KBLOCK = MTAPE(2)
  NSAMPL = MTAPE(3)
  DFREQ = ATAPE(6)
  IND = 0
  NCOEFF = MTAPE(3)/2 + 1
  JCHANS = MTAPE(5)
  DO 55 I=1, JRCHAN
    J = LCHN(I)
55 JUNIT(2*I-1) = MTAPE(2*J+119)
    JUNIT(2*I) = MTAPE(2*J+120)
    KINCR = 2*JCHANS + 1
    KOFF = - KINCR
    NCRSEQ = 256/KINCR
    JINCR = 2*NCOEFF
    WRITE (6,4001) IDUSER, MTAPE(3), JCHANS, DFREQ
    WRITE (6,4002)
    DO 60 J=1, JCHANS
    WRITE (6,4003) J, MTAPE(J+10)
    LSTART = 9*J + 22
    LSTOP = LSTART + 8

```



```

WRITE (6,4004) (MTAPE(K),K=LSTART,LSTOP)
WRITE (6,4005) MTAPE(2*J+119),MTAPE(2*J+120),ATAPE(J+140)
BACKSPACE KUNIT
RETURN
70 IND = 1
RETURN
4001 FORMAT (27HOCOCEAN TAPE SPECIFICATIONS/20HOUSER IDENTIFICATION,I10
1/15H TRANSFORM SIZE,I6/32H NO. OF CHANNELS TRANSFORMED WAS,I3/23H
2SAMPLING FREQUENCY WAS,F10.4,15H SAMPLES/SECOND)
4002 FORMAT (20HCHANNEL INFORMATION//)
4003 FORMAT (8HCHANNEL,I3,24H FROM OCEAN TAPE CHANNEL,I3)
4004 FORMAT (1H,9A4)
4005 FORMAT (19H UNITS OF DATA ARE ,2A4/24H CALIBRATION FACTOR WAS ,1PE
116.7)
END

```

C
C
C
C
C

THIS SUBROUTINE DRAWS A HORIZONTAL TIC ON THE PLOT AT THE POSITION OF THE PEN AT THE TIME OF THE CALL. THE TIC IS 0.04 INCHES LONG. THE PEN IS RETURNED TO ITS POSITION AT THE TIME OF THE CALL.

```

SUBROUTINE HORTIC
CALL WHERE (X,Y)
CALL PLOT (X-0.02,Y,2)
CALL PLOT (X+0.02,Y,2)
CALL PLOT (X,Y,2)
RETURN
END

```

```

SUBROUTINE REPLCE (M,N)
M = N
RETURN
END

```

C
C
C
C
C
C

THIS SUBROUTINE WHEN CALLED MOVES THE TAPE ON LOGICAL UNIT NUNIT PAST KSKIP END OF FILE MARKS. THE RECORDS SKIPPED OVER MUST BE IN FORTRAN BINARY. IF KSKIP IS ZERO OR NEGATIVE THE ROUTINE RETURNS WITHOUT MOVING THE TAPE.


```

SUBROUTINE SKPFL (KSKIP,NUNIT)
IF (KSKIP.LI.1) RETURN
DO 20 I=1,KSKIP
10 READ (NUNIT,END=20) IDUMMY
GO TO 10
20 CONTINUE
RETURN
END

```

```

SUBROUTINE SCALE (X,N,S,YMIN,DY,K)
DIMENSION X(2)
LOGICAL FLAG
FLAG = .FALSE.
YMAX=X(1)
YMIN=YMAX
NP=N*K
DO 10 I=1,NP,K
IF(YMAX-X(I)) 5,6,6
5 YMAX=X(I)
6 IF(X(I)-YMIN) 7,10,10
7 YMIN=X(I)
10 CONTINUE
105 RANGE = YMAX - YMIN
IF(RANGE.NE.0.0) GO TO 11
DY = 0.0
DO 12 I = 1,NP,K
12 X(I) = 0.0
11 RETURN = RANGE / (S)
SFLOG = ALOG10(SFACT)
IF(SFLOG.GE.0.0) GO TO 13
13 SFLOG = SFLOG - 1.
SFLOG = SFLOG
SFAC = SFLOG / 10.** ISFLOG
IF(SFACT.GT.1.0) GO TO 14
DY = 10.
GO TO 15
14 IF(SFACT.GT.2.0) GO TO 16
DY = 2.
GO TO 15
16 IF(SFACT.GT.4.0) GO TO 17
DY = 4.
GO TO 15
17 IF(SFACT.GT.5.0) GO TO 18
DY = 5.

```

SCAL00020
SCAL00030
SCAL00034
SCAL00036
SCAL00040
SCAL00050
SCAL00060
SCAL00070
SCAL00080
SCAL00090
SCAL00100
SCAL00110
SCAL00120
SCAL00130
SCAL00140
SCAL00150
SCAL00160
SCAL00170
SCAL00180
SCAL00190
SCAL00200
SCAL00210
SCAL00220
SCAL00230
SCAL00240
SCAL00250
SCAL00260
SCAL00270
SCAL00280
SCAL00290
SCAL00300
SCAL00310
SCAL00320
SCAL00330
SCAL00340
SCAL00350

SCAL0360
SCAL0370
SCAL0380
SCAL0390
SCAL0400
SCAL0410
SCAL0415
SCAL0420
SCAL0430
SCAL0440
SCAL0444
SCAL0446
SCAL0447
SCAL0448
SCAL0450
SCAL0460
SCAL0470
SCAL0480

```

GO TO 15
18 IF(SFAC1.GT.8.0) GO TO 19
   DY = 8.0 * 10.** ISFLOG
GO TO 15
19 DY = 10.** (ISFLOG + 1)
   TEMP = YMIN / DY
   IF( FLAG ) GO TO 21
   IF(TEMP.LT.0.) TEMP = TEMP - 1.0
   IYMIN = TEMP
   YMIN = FLOAT(IYMIN)* DY
   FLAG = .TRUE.
   TEMP = YMIN + S * DY
   SCALE AGAIN IF THE RANGE HAS INCREASED TOO MUCH
   IF( YMAX .GT. TEMP ) GO TO 105
21 DO 20 I=1,NP,K
   X(I)=(X(I)-YMIN)/DY
20 RETURN
END

```

SUBROUTINE AXIS(X,Y,BCD,NC,SIZE,THETA,YMIN,DY)

```

DIMENSION BCD( 2 )
SIG = 1.0
IF( NC ) 1 , 2 , 2
SIG = -1.0
NAC = IABS( NC )
TH = THETA * 0.017453294
N = SIZE + 0.5
TN = N
CTH = COS( TH )
STH = SIN( TH )
SCALE THE AXIS *****
ADY = DY
ABSU = YMIN
EXP = 0.0
IF( ADY ) 9 , 18 , 9
IF( ABS( ADY ) - 100.0 ) 10 , 12 , 12
ADY = ADY * 0.1
ABSU = ABSU * 0.1
EXP = EXP + 1.0
GO TO 9
ADY = ADY * 10.0
ABSU = ABSU * 10.0
EXP = EXP - 1.0
IF( ABS( ADY ) - 0.01 ) 14 , 18 , 18
10 ***** INITIALIZE THE MAIN LOOP *****

```

```

C      XL, YL      ETC. ARE FOR THE LINE
C      XN, YN      ETC. ARE FOR THE NUMBERS
C      XT, YT      ETC. ARE FOR THE TITLE
C$*****
18      XS = X
      YS = Y
      TEMP = 0.2*SIG - 0.05
      XNA = XS - TEMP*STH - 0.0857*CTH
      YNA = YS + TEMP*CTH - 0.0857*STH
      NAFT = 3
      ICK = ABS( ABSU ) * 1000.0 + 0.5
      IF( MOD( ICK, 10 ) .EQ. 0 ) NAFT = 2
      IF( MOD( ICK, 100 ) .EQ. 0 ) NAFT = 1
      CALL NUMBER( XNA, YNA, 0.10, ABSU, THETA, NAFT )
      XLB = XS + IN * CTH
      YLB = YS + IN * STH
      TEMP = 0.1*SIG
      XLA = XLB - TEMP * STH
      YLA = YLB + TEMP * CTH
      NBY2 = N/2
C$***** THIS IS THE MAIN LOOP *****
      DO 20 I = 1, N *****
C$***** NEXT THE NUMBER *****
      XNA = XNA + CTH
      YNA = YNA + STH
      ABSU = ABSU + ADY
      NAFT = 3
      ICK = ABS( ABSU ) * 1000.0 + 0.5
      IF( MOD( ICK, 10 ) .EQ. 0 ) NAFT = 2
      IF( MOD( ICK, 100 ) .EQ. 0 ) NAFT = 1
      CALL NUMBER( XNA, YNA, 0.10, ABSU, THETA, NAFT )
C$***** NOW PERHAPS THE TITLE *****
      IF( I .NE. NBY2 ) GO TO 19
      TNC = NAC + 7
      TEMP = SIZE*0.5 - 0.06*TNC
      TEMPB = (-0.07 + SIG*0.36 )
      XT = XS + TEMP*CTH - TEMPB*STH
      YT = YS + TEMP*STH + TEMPB*CTH
      CALL SYMBOL( XT, YT, 0.14, BCD, THETA, NAC )
      IF( EXP .EQ. 0. ) GO TO 19
      TEMP = ( TNC - 6.0 ) * 0.12
      XT = XT + TEMP*CTH
      YT = YT + TEMP*STH
      CALL SYMBOL( XT, YT, 0.14, 7H(X10 ) , THETA, 7 )
      XT = XT + 0.48*CTH - 0.07*STH
      YT = YT + 0.48*STH + 0.07*CTH
      CALL NUMBER( XT, YT, 0.10, EXP, THETA, -1 )
CONTINUE

```

```

20      CONTINUE THE LINE BACKWARDS TO THE ORIGIN *****
C$*****
      CALL PLOT( XLA , YLA , +3 )
      CALL PLOT( XLB , YLB , +2 )
      DO 44 I = 1, N, TIC *****
C$*****
      BOTTOM OF NEXT TIC *****
      XLB = XLB - CTH
      YLB = YLB - STH
      CALL PLOT( XLB , YLB , +2 )
C$*****
      TOP OF NEXT TIC *****
      XLA = XLA - CTH
      YLA = YLA - STH
      CALL PLOT( XLA , YLA , +2 )
      CALL PLOT( XLB , YLB , +2 )
      CONTINUE
      RETURN
      END
44

```

```

PROGRAM VECTOR
THIS PROGRAM IS WRITTEN TO PROVIDE A PROGRESSIVE VECTOR DIAGRAM. THIS
DIAGRAM WILL SHOW THE SUMMATION OF VECTORS MADE OF CURRENT SPEED AND THE
CURRENT DIRECTION FOR ONE HOUR PERIODS. THE VECTORS ARE FOUND FROM THE
OUTPUT OF PROGRAM CURRENT. THESE VALUES ARE TAKEN FROM PROGRAM CURRENT,
PUNCHED ONTO COMPUTER CARDS AND THEN USED IN THE PROGRAM. SPORT OF WATER AT
THE PROGRESSIVE VECTOR DIAGRAM WILL SHOW THE VOLUME TRANSPORT AT THIS DEPTH.
A GIVEN LOCATION AND DEPTH. THIS WILL INDICATE WHETHER THE DIRECTION OF MOVEMENT.
THERE IS A NET TRANSPORT OF WATER AND WHAT IS THE DIRECTION OF MOVEMENT.

WRITTEN BY: LT. W.A. CASTER
DATE: 4 AUGUST 1969

```

WRITTEN BY: LT. W. A. CASTER
DATE: 4 AUGUST 1969

DIMENSION SPD(200),DIR(200),X(200),Y(200),A(200),B(200),XA(200),

```

*YB(200) TVT
  REAL*8 N = NUMBER OF DATA POINTS
  READ(5,10) N
  FORMAT(5,10)
  WRITE(6,11) N
  FORMAT(6,11)
  WRITE(6,12) IHI,6X,'NO. OF DATA POINTS = ',I6,///)
  REAL*8 LABEL(4H END)
  READ(5,20) I TITLE
  FORMAT(5,20)
  REAL*8 LABEA(8H
  READ IN DATA
  READ(5,21) (SPD(I),DIR(I),I = 1,N)
  FORMAT(8(F5.2,F4.0))
  PRINT OUT DATA
  WRITE(6,12)
  WRITE(6,13) 'SPEED',3X,'DIRECTION',4X,'DATA PT.
  WRITE(6,13) (SPD(I),DIR(I),I = 1,N)
  FORMAT(6X,F5.2,5X,F4.0,9X,I4)
  INITIALIZE
  X(1) = 0.
  Y(1) = 0.
  A(1) = 0.
  B(1) = 0.
  N1 = N + 1
  WRITE(6,22)
  FORMAT(1HI)
  DO 1 I=2,N1

```



```

CONVERT DIRECTION IN DEGREES TO DIRECTION IN RADIANS BY MULTIPLYING BY
0.01745.
C CONVERT CURRENT SPEED AND DIRECTION VECTOR INTO X AND Y COMPONENTS.
X(I) = SPD(I-1)*SIN(DIR(I-1)*0.01745)
Y(I) = SPD(I-1)*COS(DIR(I-1)*0.01745)
C A AND B ARE THE SUM OF THE INDIVIDUAL X AND Y VALUES.
A(I) = X(I) + A(I-1)
B(I) = Y(I) + B(I-1)
WRITE(6,23) X(I),Y(I),A(I),B(I)
23 *F10.4)
FORMAT (5X,'X= ',F10.4,5X,'Y= ',F10.4,5X,'A= ',F10.4,5X,'B= ',
1 CONTINUE
DO 2 I=1,N1,24
XA AND YB WILL BE THE VALUES OF A AND B AT THE END OF A 24 HOUR PERIOD.
XA(K) = A(I)
YB(K) = B(I)
K = K+1
2 CONTINUE
N2 = N1/24
CALL DRAW WILL DRAW PROGRESSIVE VECTOR DIAGRAMS USING THE CALCOMP PLOTTER
IF THE NUMBER OF DATA POINTS ARE LESS THAN 25. THE SECOND DRAW
STATEMENT SHOULD BE DELETED, BECAUSE THIS STATEMENT INDICATES A 'X' EVERY
24 HOURS AND DRAW MUST HAVE AT LEAST TWO POINTS IN ORDER TO DRAW.
CALL DRAW (N1,A,B,1,0,LABEL,ITITLE,1,1,0,0,0,9,10,0,C,LA$T)
CALL DRAW (N2,XA,YB,3,1,LABELA,ITITLE,1,1,0,0,0,9,10,0,C,LA$T)
WRITE(6,22)
CALL PLOTP (X,Y,N1,0)
WRITE(6,22)
CALL PLOTP (A,B,N1,1)
CALL PLOTP (XA,YB,N2,3)
WRITE(6,24) A(N1),B(N1)
24 *F10.4,1X,'N.M.',//)
FORMAT (1H,5X,'FINAL A VALUE= ',F10.4,1X,'N.M.',5X,'FINAL B VALUE
*= ',F10.4,1X,'N.M.',//)
VECTOR SUM IN NAUTICAL MILES.
VECSUM = SORT(A(N1)+B(N1)*R(N1))
CONVERT TO METERS.
VECSUM = VECSUM*(1853.248)
WRITE(6,25) VECSUM
25 FORMAT (5X,'VECTOR SUM= ',F10.4,1X,'METERS',///)
AREA = ONE SQUARE METER
AREA = 1.0
WRITE(6,27) AREA
27 FORMAT (5X,'AREA= ',F10.4,1X,'M**2',//)
TOTAL VOLUME TRANSPORT IN M**3.
TVT = AREA*VECSUM
WRITE(6,28) TVT
28 FORMAT (5X,'TOTAL VOLUME TRANSPORT= ',D20.13,1X,'M**3',//)

```

```

C      VOLUME TRANSPORT PER HOUR.
      ANT = N
      XVT = TVT/AN
      WRITE (6,29) XVT
29  FORMAT (5X,'VOLUME TRANSPORT PER HOUR= ',F20.4,1X,'M**3/HR',///)
      STOP
      END

```

LITERATURE CITED

- Batchelor, G.K., "Small-scale Variation of Convected Quantities Like Temperature in Turbulent Fluid. Part 1. General Discussion and the Case of Small Conductivity," *Journal of Fluid Mechanics*, Vol. 5, Part 1, pp. 113-133, 1959.
- Dooley, John J., "An Investigation of Near-Bottom Currents in the Monterey Submarine Canyon," Unpublished Master's Thesis, Naval Postgraduate School, Monterey, California, 1968.
- Franz, F., "Model 501 in situ Current Speed, Current Direction, and Temperature Recording System, Operation and Maintenance Instructions," Hydro Products, San Diego, California, 1967.
- Gatje, P.H. and Pizinger, D.D., "Bottom Current Measurements in the Head of Monterey Submarine Canyon," Unpublished Master's Thesis, Naval Postgraduate School, Monterey, California, 1965.
- Kuenen, P. H., Marine Geology, pp. 485-526, Wiley, 1950.
- Martin, Bruce D., "Monterey Submarine Canyon, California: Genesis and Relationship to Continental Geology," Unpublished Ph.D. Thesis, University of Southern California, 1964.
- Martin, B. D. and Emery, K.O., "Geology of Monterey Canyon, California," *The American Association of Petroleum Geologists Bulletin*, Vol. 51, No. 11, pp. 2281-2304, 1967.
- Njus, Ingamar J., "An Investigation of Environmental Factors Affecting the Near-Bottom Currents in the Monterey Submarine Canyon," Unpublished Master's Thesis, Naval Postgraduate School, Monterey, California, 1968.
- Raines, William A., "Sub-Tidal Oscillations in Monterey Harbor," Unpublished Master's Thesis, Naval Postgraduate School, Monterey, California, 1967.
- Robinson, David B., "Seiching in Monterey Bay," Unpublished Master's Thesis, Naval Postgraduate School, Monterey, California, 1969.
- Shepard, Francis P., Submarine Geology, 2d ed., Harbor & Row, 1963.
- Shepard, F.P., Curray, J.R., Inman, D.L., Murray, E.A., Winterer, E.L. and Dill, R.F., "Submarine Geology by Diving Saucer," *Science*, Vol. 145, No. 3636, pp. 1042-1045, 1964.
- Shepard, F.P. and Emery, K.O., "Submarine Topography Off the California Coast: Canyons....," *Geological Society of American Special Papers*, No. 31, pp. 72-78, 1941.

- Shepard, F.P. and Marshall, N.F., "Currents in La Jolla and Scripps Submarine Canyons," *Science*, Vol. 165, No. 3891, pp. 177-178, 1969.
- Shepard, F.P., Revelle, R.R. and Dietz, R.S., "Ocean Bottom Currents Off the California Coast," *Science*, Vol. 89, No. 2317, pp. 488-489, 1939.
- Skogsberg, Tage, "Hydrography of Monterey Bay, California. Thermal Conditions, 1929-1933," *Transactions of the American Philosophical Society*, Vol. XXIX, 1936.
- Squire, James L., "Observations on Cumulative Bottom Drift in Monterey Bay Using Seabed Drifters," *Limnology and Oceanography*, Vol. 14, No. 1, pp. 163-167, 1969.
- Stetson, H.C., "Current Measurements in the Georges Bank Canyons," *Transactions of the American Geophysical Union*, Vol. 18, pp. 216-219, 1937.
- Veatch, A.C. and Smith, P.A., "Atlantic Submarine Valleys of the United States and the Congo Submarine Valley," *Geological Society of America Special Papers Number 7*, 1939.
- Webster, Ferris, "Some Perils of Measurements from Moored Ocean Buoys," *Technical Report No. 64-18, Unpublished Manuscript*, Woods Hole Oceanographic Institute, Woods Hole Massachusetts, 1964a.
- Webster, F., "Processing Moored Current Meter Data," *Technical Report No. 64-55, Unpublished Manuscript*, Woods Hole Oceanographic Institute, Woods Hole, Massachusetts, 1964b.
- Wilson, J.R., Boston, N.E.J. and Denner, W.W., "Digital Analysis of Turbulence Data on the IBM 360/67 at the Naval Postgraduate School," *Unpublished Manuscript*, Naval Postgraduate School, Monterey, California, 1969.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Documentation Center Cameron Station Alexandria, Virginia 22314	20
2. Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3. Department of Oceanography, Code 58 Naval Postgraduate School Monterey, California 93940	3
4. Professor Warren W. Denner, Code 58Dw Naval Postgraduate School Monterey, California 93940	10
5. Oceanographer of the Navy The Madison Building 732 N. Washington Street Alexandria, Virginia 22314	1
6. Superintendent U. S. Naval Academy Annapolis, Maryland 21402	1
7. Commandant (PTP) U. S. Coast Guard Washington, D. C. 20226	2
8. Commandant (OMS) U. S. Coast Guard Washington, D. C. 20226	1
9. Commanding Officer U. S. Coast Guard Oceanographic Unit Bldg. 159-E, Navy Yard Annex Washington, D. C. 20390	1
10. MST School U. S. Coast Guard Training Center U. S. Coast Guard Base Governors Island New York, N. Y. 10004	1
11. Department of Oceanography U. S. Coast Guard Academy New London, Connecticut 06320	1

12.	Commander (oms)
Eastern Area, U.S. Coast Guard
U. S. Coast Guard Base
Governors Island
New York, N. Y. 10004	1
13.	Commander (oms)
Western Area, U.S. Coast Guard
630 Sansome Street
San Francisco, California 94126	1
14.	Lt. William A. Caster, USCG
USCG ROCKAWAY (WAGO-377)
U. S. Coast Guard Base
Governors Island
New York, N. Y. 10004	3

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Naval Postgraduate School Monterey, California 93940		2a. REPORT SECURITY CLASSIFICATION Unclassified	
		2b. GROUP	
3. REPORT TITLE Near-Bottom Currents in Monterey Submarine Canyon and on the Adjacent Shelf			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Master's Thesis; October 1969			
5. AUTHOR(S) (First name, middle initial, last name) William Albert Caster			
6. REPORT DATE October 1969		7a. TOTAL NO. OF PAGES 202	7b. NO. OF REFS 22
8a. CONTRACT OR GRANT NO.		9a. ORIGINATOR'S REPORT NUMBER(S)	
b. PROJECT NO.			
c.		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
d.			
10. DISTRIBUTION STATEMENT This document has been approved for public release and sale; its distribution is unlimited.			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY Naval Postgraduate School Monterey, California 93940	
13. ABSTRACT <p>Near-bottom currents in Monterey Submarine Canyon and the adjacent shelf were collected using Savonius current meters. Simultaneous measurements were made with one current meter on the shelf at a depth of 91 meters and one meter located in the Canyon at 366 meters. Another record was taken in the Canyon at the 366-meter location. Current speed, current direction and water temperature were recorded continuously for approximately seven days in each record. First-order statistics were calculated and plotted for these time-series data. Scatter diagrams, progressive vector diagrams and power spectra were also computed and analyzed for the records collected during this study and for records of sufficient length from previous investigations.</p> <p>Net current set was in a cross-canyon direction for many of the records; however, the currents in the Canyon oscillated as reported in previous investigations. The oscillations were not as evident on the shelf record. Mean and maximum current speeds recorded in the Canyon were 10 and 51 cm/sec, respectively. On the shelf these values were 7 and 25 cm/sec. Observed values of net current set and volume transport are related to Monterey Bay seasonal water conditions.</p>			

14.

KEY WORDS

LINK A

LINK B

LINK C

ROLE

WT

ROLE

WT

ROLE

WT

Near-bottom Currents

Monterey Bay

Monterey Submarine Canyon

Submarine Canyon Currents

Ocean Currents

thesC3

Near-bottom currents in Monterey submari

3 2768 002 09094 6

DUDLEY KNOX LIBRARY