WORKING NOTES ON CURRENT-METER DEPLOYMENT, RECOVERY MAINTENANCE, AND DATA PROCESSING FOR SAN FRANCISCO BAY

by Ralph T. Cheng and Jeffrey W. Gartner

U.S. Geological Survey

Open-File Report #80-1297

Prepared as part of a continuing
San Francisco Bay estuarine study

December 1980

UNITED STATES DEPARTMENT OF THE INTERIOR

CECIL D. ANDRUS, Secretary

GEOLOGICAL SURVEY

H. William Menard, Director

OPEN-FILE REPORT

For additional information write to:

Regional Hydrologist
Water Resources Division
U.S.Geological Survey
345 Middlefield Road
Menlo Park, CA 94025

CONTENTS

ILLUSTRATIONS

WORKING NOTES ON CURRENT-METER DEPLOYMENT, RECOVERY, MAINTENANCE, AND DATA PROCESSING FOR SAN FRANCISCO BAY

By Ralph T. Cheng and Jeffrey W. Gartner

ABSTRACT

A current-metering program has been initiated by the U.S. Geological Survey as a part of an interdisciplinary study to collect basic hydrodynamic data for the San Francisco Bay estuarine system. The current meters employed for the study record current speed and direction, water temperature, and conductivity at a preselected time interval on endless-loop magnetic tape. This report documents operational procedures, maintenance, and computer programs for data reduction for these current meters.

INTRODUCTION

The complex interactive processes which affect an estuary such as San Francisco Bay, California, include tides, freshwater input, wind, and mixing. To enhance understanding of the hydrodynamic and ecological processes at work in the Bay, mathematical hydrodynamic models have been developed (Conomos and others, 1978; Walters and Cheng, 1979; and Cheng and Walters, 1978). Basic hydrodynamic data are needed to describe estuarine hydrodynamic processes and to support mathematical modeling research. At present there are no long-term (monthly) records of current data available for San Francisco Bay. To fill this gap and to complement other hydrographical, biological, and chemical data-collection programs, a research program to collect current-meter data of San Francisco Bay has been established.

Factors that must be considered in establishing such a data-collection program include the selection of suitable hardware and the preparation, maintenance, and repair of these instruments, as well as the deployment and recovery procedures used in the field. Other important and often underestimated efforts are the development of data-reduction procedures and software. This report addresses these considerations. It also documents our experience while working with these instruments to enable better understanding and interpretation of current-meter data to be presented in future reports.

FIELD WORK

The initial phase of the field work consisted of selecting the current meter best suited to the intended program. Primary considerations in making the selection were ease of handling, accuracy of measurement over an extended period of 30 to 60 days, and availability of efficient data-reduction procedures. Of the several meters examined the ENDECO-174[1/] most closely met the needs of the Bay studies (Cheng, 1978).

The typical current-meter mooring employed by the U.S. Geological Survey is a subsurface taut-wire mooring (see Fig. 1), which consists of floats with sufficient buoyancy to suspend the weight of equipment and mooring cable, the current meter(s), an acoustic release, and a suitable anchor (normally 90-115 kg). All of these components are shackled in series by 0.6-cm stainless steel cable. Length of the mooring cable is determined by water depth and is so chosen that the top of the subsurface float is a minimum of 3 m below the surface of the mean lower low water (MLLW). Current-meter arrays are normally deployed from the R/V *Polaris*, the Geological Survey's 97-foot research vessel which is equipped with power winches, capstans, booms and davits to facilitate deployment and recovery of equipment. Prior to deployment, the current meters are serviced in the laboratory following the predeployment check list (see Appendix A). Immediately before deployment components of the assembly are shackled together and placed on deck. The deployment sequence is an anchor-in-last procedure. The entire

[1/]Mention of a commercial company or product in this report does not constitute an endorsement by the authors or the U.S. Geological Survey.

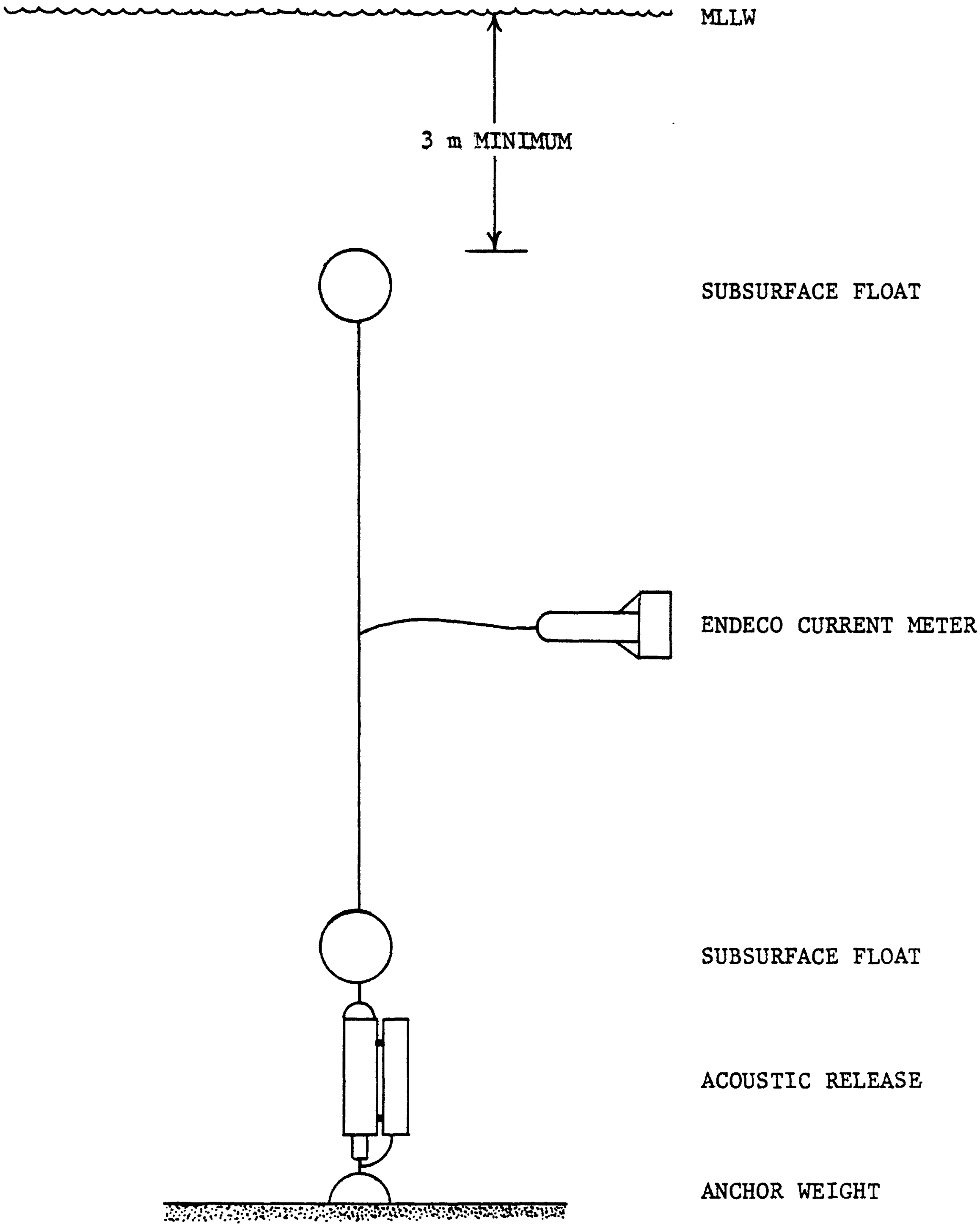

FIGURE 1. Schematic diagram of a typical current meter array.

assembly, with the exception of the anchor, is lowered into the water and held away from the anchor. The anchor weight, which is attached to a winch through a quick-release mechanism, is then lifted up, swung free of the ship, and suspended above water. While the array is held away from the anchor weight, the mooring assembly is slowly lowered to the bottom. The winch cable is detached by triggering the quick release, thus completing the deployment. The position of the array is determined by use of a sextant fix or radar ranges and bearings from known points.

The deployment periods are normally scheduled to cover a minimum of 30 days because currents in the Bay are primarily driven by the tides, which include semidiurnal, diurnal, fortnightly, monthly, and annual variations. Where water depth permits, three meters have been used in an array to give a more complete picture of the velocity profile. In shallower areas only one or two meters are used and the meters are placed as close to the bottom as practicable (2 m) or at middepth.

Equipment recovery is initiated by positioning the R/V *Polaris* in the vicinity of the current-meter array. Then by activating the command unit of the acoustic release from shipboard, an acoustic signal is transmitted from the submerged transducer to the receiver transducer on the release unit. If the correct code is identified by the acoustic release, it responds by separating the anchor and release unit. Due to buoyancy the entire mooring assembly, except the anchor, floats to the surface. The array assembly is still attached to the anchor by a 0.6-cm nylon rope which was originally housed in a canister attached to the acoustic release. After recovering the instruments, the anchor is recovered whenever possible by retrieving the nylon line. See Appendix B for postdeployment procedures.

Our experience indicates that the acoustic release sometimes may not work as described owing to marine growth or other obstruction at the release shackle, electronic failure, or other unknown reason. If this situation occurs, it is necessary to search for the current meter array for recovery by dragging. The dragging procedure involves running a weighted dragline between two small boats to search the area of deployment until the array is circled and snagged. Sometimes when water depth and current speed are favorable, the assembly can be recovered without a diver's assistance.

INSTRUMENT DESCRIPTION

The ENDECO-174 current meter (also see ENDECO Service Manual) is an axial-flow, ducted-impeller current-meter which records digital data on 0.6-cm magnetic tape (endless-loop 8-track cartridge). Recorded parameters include current speed and direction, and water temperature and conductivity. Data can be recorded up to approximately 40 days on one tape cartridge with the recording rate selected at 30 per hour. At each observation eight four-bit data words are written in series on the tape; that is, each of the four parameters is represented by a byte to be reconstructed from two four-bit words. Current speed is determined by measuring the displacement of an encoder disc which is driven by the impeller through a magnetic coupler and a 500:1 reduction gear. Heading of the meter is determined and digitized from the output of a damped magnetic compass. Temperature is determined by a thermolinear thermistor encased in stainless steel, and conductivity by an induction-type electrodeless conductivity probe. Current speed and direction are recorded in Gray code, and temperature and conductivity in binary code (Richards, 1971). Gray code is similar to binary code but is designed so that two bits never change value in successive Gray code sequence.

At the end of every hour a record mark is given which is represented by a record gap between the recorded data blocks. Every 24 hours, a 24-hour mark, which is signified by all "1's" in data, is written on the tape. Two conductivity reference values are generated internally every 24 hours and are recorded as reference data immediately following the 24-hour mark. These reference data are to be incorporated in the data-reduction program to construct a regression curve for conductivity computation to compensate for any electronic drift due to change of ambient temperature or battery voltage. These reference data, in conjunction with hourly data gaps and 24-hour marks, are used to check the accuracy of timing of the data series, and to assist, when necessary, in reconstructing partially lost data.

The acoustic release used in the data collection program is the Innerspace Technology Model 431 (see Service Manual), which uses FM coding on a 22 kHz carrier frequency. The release unit contains an expandable link within the pressure case. Once the link is broken after receipt of the correct acoustic signal, a spring-loaded obstruction pin is retracted to permit free rotation of an output hook. When the output hook is turned it allows the shackle, which is attached to the anchor, to fall free, and thus permits separation of the release unit and anchor.

PARAMETER ACCURACY

Accuracy specifications given by the manufacturer for ENDECO-174 current meters are as follows: speed, ± 3.0 percent of full scale (257 cm/s) above 2.57 cm/s; direction, $\pm 7.2^{\circ}$; temperature, $\pm 0.2^{\circ}C$; and conductivity, ± 0.55 mmho/cm.

TEMPERATURE

The accuracy of temperature measurements of all meters was checked periodically in a test tank (1 x 1 x 2 m) in the laboratory. The mean temperature difference between recorded and actual values (two sets of readings for each meter) after computer correction was $-0.01^{\circ}C$ with a standard deviation of $0.14^{\circ}C$. Maximum temperature differences were -0.3 and $+0.2^{\circ}C$. The tests were conducted at room temperature (approximately 20° - $24^{\circ}C$).

CONDUCTIVITY

Conductivity accuracy was also examined in the same test tank at three conductivity values (approximately 15, 30, and 45 mmho/cm). Corresponding binary data and actual conductivity values (determined by a Beckman Induction Salinometer), as well as binary conductivity reference values, were used in a linear regression analysis to determine two fixed "standard" conductances for each meter. A conductivity rating curve is established by combination of meter-generated conductivity reference values and "standard" conductances. Conductance values used for subsequent data reduction were either supplied by the manufacturer or generated in the U.S. Geological Survey laboratory, depending on which set gave closer agreement to induction salinometer-rated values. Once rating curves were established, all meters were retested again at two different conductivities to verify the predictability of the rating curves. Using this method, the mean conductivity difference for all meters was placed at 0.13 mmho/cm. The standard deviation was 0.85 mmho/cm, and the maximum differences were -2.76 and +1.27 mmho/cm. The extremely large difference of -2.76 mmho/cm between the recorded and actual value was for only one meter, and may be erroneous. Unfortunately, this current meter leaked and was destroyed during a subsequent deployment, and no further analysis was possible. Recomputing the statistics by excluding that meter results in a mean of 0.25 mmho/cm, a standard deviation of 0.58 mmho/cm and maximum differences of -1.06 and +1.27 mmho/cm.

SPEED

The accuracy for measurement of current speed bv ENDECO-174 current meter was examined at the U.S. Geological Survey Gulf Coast Hydroscience Center, Bay St. Louis, Mississippi. Generally, the impeller rating is linear in the test range of 0-250 cm/s. Each impeller supplied by ENDECO was rated in a continuous water tunnel at Johns Hopkins University before shipment. The speed coefficient recommended by the manufacturer for a perfect impeller is 50.1 rpm/knot. There is a 500:1 gear reduction between impeller and the encoder disc which is digitized and represented by 256 divisions; that is, a byte. Accordingly, the current velocity, V, or speed rating curve is worked out to be

$$V(cm/s) = 2.009 \frac{\Delta N}{\Delta t}$$

where ΔN is the difference between successive speed readings, and Δt is the sampling interval in minutes. To account for imperfection of individual impellers, the rating curve becomes

$$V \text{ (cm/s)} = 2.009 \ (749/C_I) \frac{\Delta N}{\Delta t}$$

where C_I is the impeller coefficient obtained in the water tunnel at the Johns Hopkins University. The impeller coefficients supplied by the manufacturer have been found to be consistently in error by about 1 to 2 percent (too low) when compared with Geological Survey tow-tank results; this deviation appears to be the wall effect of the water tunnel. The U.S. Geological Survey tow-tank results are used in the computation. of speed. Another problem which may affect current-speed readings is marine growth and/or sea grass in contact with the impeller or bearing shaft. There is a tendency for sea grass to wrap around the impeller shaft between the impeller and current meter base, significantly hampering free movement of the impeller. At present, there is no remedy for the problem. The manufacturer is working on a new type of mold-injected impeller which we have tested in the field. These impellers are less susceptible to problems of this kind. New mold-injected impellers currently supplied by the manufacturer were tested in the Geological Survey tow tank and speed computations were found to be in error by about 15 percent (too high).

DIRECTION

The ENDECO-174 current meters use the Model-213 Digicourse compass for meters with serial numbers smaller than 133. The manufacturer has since replaced the compass with the Model-218 Digicourse compass for their meters with serial numbers greater than 132. The directional calibration was performed at the factory on a swing table.

While the meter was turned on and recording, the swing table rotated 15^{o} every 10 minutes for at least two complete revolutions for a total test duration of no less than 8 hours. The calibration data were supplied to us, and this information was incorporated into our data-reduction program in the calculation for current direction. At this time verification of the calibration data for direction supplied by the manufacturer has not been attempted. However, we have encountered some difficulty in certain meters for which directional readings were affected by the lower tolerance limits associated with rolling and pitching of the current meter. This problem is further discussed in the next section.

HARDWARE RELIABILITY

Several problems encountered during the first 2 years of use of the ENDECO-174 current meters are documented here. The most damaging was leakage into the housings of several meters during deployment. Subsequent testing by the manufacturer indicated that one meter leaked through a fastening screw on the conductivity probe due to a faulty manufacturing procedure. This problem can be corrected by using a retrofit kit supplied by the manufacturer. The repair involves replacing one of the screws on the end cap of the conductivity probe with a special fitting to prevent leakage. The second meter presumably leaked at the end cap. This was probably caused by a worn or damaged "O" ring. To eliminate the possibility of leakage through the end cap, the end-cap "O" rings are now replaced routinely prior to each deployment. Two other meters which were completely flooded were either hit by vessels or entangled with the mooring line. The conductivity probes were sheared off both meter housings, thus permitting water to flood the meter through a wiring access hole. The manufacturer has remedied this problem by retrofitting the current meters with a PVC block underneath the conductivity probe to prevent the mooring line from being entangled with the conductivity probe. Less serious damage to several other meters included lost or broken shrouds, vanes, and impellers, also probably caused by passing vessels or mooring-line entanglement.

A less catastrophic problem which may affect data quality is the loss of impeller blades. While the impellers are specified to withstand currents to about 5 knots, one blade was lost at a low velocity (less than 2 knots), and six of eight blades were lost on another occasion at a higher velocity. In the latter case it was impossible to determine the current velocity encountered because the data-tape cartridge was defective and no data were recorded. The expected maximum velocity at the site was 4 to 5 knots. Tow-tank evaluation of the damaged impeller indicated that the accuracy of speed is not severely affected at high current speed with loss of one or two blades; however, the threshold velocity of the current meter is certainly affected because the impeller is no longer balanced. This problem was brought to the attention of the manufacturer, whose solution is to replace the old impellers with a single-piece, mold-injected impeller of greater strength.

In addition to physical damage of current meters, a significant cause of lost data is related to the magnetic tape cartridge. Of the 14-percent data loss during the first 2 years of operation, tape-cartridge malfunctions (tapes not feeding out of hubs, improper tension and so forth) were responsible for approximately 60 percent of the total loss. Thirty-five percent of the data loss was due to physical damage of meters, and 5 percent for miscellaneous reasons. A possible reason for tape jamming is the swelling of tape caused by residual moisture within the current-meter housing. This is a particular problem when the meters are loaded in warm, moist air and subsequently deployed in colder water. The recommended solution to this problem consists of storing the tapes in desiccated containers and purging the instrument environment with Freon-12 to displace the moist air just before sealing it. Large or multiple desiccant bags should be used in the current meters.

In addition to the complete loss of data, there were several instances of poor performance in measuring one or more parameters. Four current meters showed faulty temperature readings outside the range of manufacturer's specification: measurements with temperatures too high or too low, or a condition in which the temperatures changed by one or two degrees between readings. These problems may have been due to defects in specific integrated-circuit boards. Three of the four meters were returned to the manufacturer for repair and recalibration and one meter mysteriously appeared to have fixed itself. Subsequent calibration checks in a Geological Survey test tank showed that all meters measure temperature within acceptable limits of $\pm 0.2^{\circ}C$.

A problem relating to conductivity measurements became apparent in low-salinity water. While the meter was specified to have a low conductivity limit of 5 mmho/cm, 2 of the 11 meters did not indicate any conductivity lower than about 7 mmho/cm. This also was a problem with individual integrated circuit boards, and has since been corrected.

Directional information appears to suffer from two separate and distinct limitations. These are the allowable tilt angle of the compass and the lack of certain binary readings for specific headings on certain meters. As previously mentioned, the earlier model ENDECO-174 current meters utilize the Model-213 Digicourse compass. The Model-213 compass has limits on its pitch and roll axes of 5° of arc. While the neutrally buoyant meter and tethered attachment of meter to mooring line keep large fluctuations in pitch and roll to a minimum, long periods of repeating bearings do occasionally occur. The other factor affecting directional accuracy is the consistent lack of certain binary number readings (headings) with some meters. However, missing values seldom occur for more than two or three sequential binary numbers. Because the heading resolution is 1.4^{o} (increment between sequential binary numbers), it is assumed that this defect does not significantly affect the manufacturer's stated directional accuracy limit of $\pm 7.2^{o}$.

The largest area of uncertainty remains in the measurement of conductivity values. Because the documentation from the manufacturer is limited, a short explanation of the design principle for measuring conductivity is included here. In order to compensate conductivity readings for electronic drift and reduced battery voltage over a long deployment, the conductivity circuitry contains two fixed-frequency outputs which are switched into the circuitry in place of the conductivity probe outputs once every 24 hours to generate a set of "synthetic" conductance values. The "synthetic" conductance values for each current meter are deduced from calibration tank test values in a linear regression. These "synthetic" conductance values which are assumed known and unchanged are used in conjunction with the conductance reference values recorded on the tape every 24th hour to generate a rating curve for computation of conductivity for the next 24-hour period. The conductance reference values generated every 24th hour should vary only slightly as the battery voltage changes. In some meters these values change radically, especially in the first and last days, thus rendering the conductivity calibration meaningless. The problem is apparently in the circuits of the individual instruments, and requires manufacturer servicing.

DATA PROCESSING

While working with fairly delicate recording instruments, numerous problems, often related to only one specific meter, may periodically occur. In order to enable rapid identification of malfunctions, a data-processing system which is structured around a microcomputer and tape translator (ENDECO-173) has been developed (Cheng and Gartner, 1979). The data-acquisition system has proven to be convenient and cost effective. The system consists of an LSI-11 microcomputer, two floppy disc drives, a 9-track tape drive, a line printer, a graphic cathode-ray-tube (CRT) terminal, and a data-tape translator as shown in Fig. 2. The CRT terminal performs the main Input-Output (I/O) control of the system, and any hard copy of output is produced on the line printer. The central processing unit (CPU) system-software resides on one floppy disc, whereas the other is user's working space. In this case, the raw-data tape recovered from the current meter is translated and stored on the user's floppy disc in binary numbers.

An ENDECO-173 data tape translator consisting of a microprocessor and limited memory is used to translate raw data and to communicate with the microcomputer. Upon command from the microcomputer, the translator engages its tape-drive mechanism and reads a block of four-bit data into its memory. The microprocessor in the translator combines pairs of four-bit data series into bytes and sends the block of data to the microcomputer as a block of 125 bytes of data for further processing. The tape translator is then set to standby awaiting the next instruction from the microcomputer.

The microcomputer includes an LSI-11 processor, a power supply, two memory boards with 28K words of onboard refreshed memory, a number

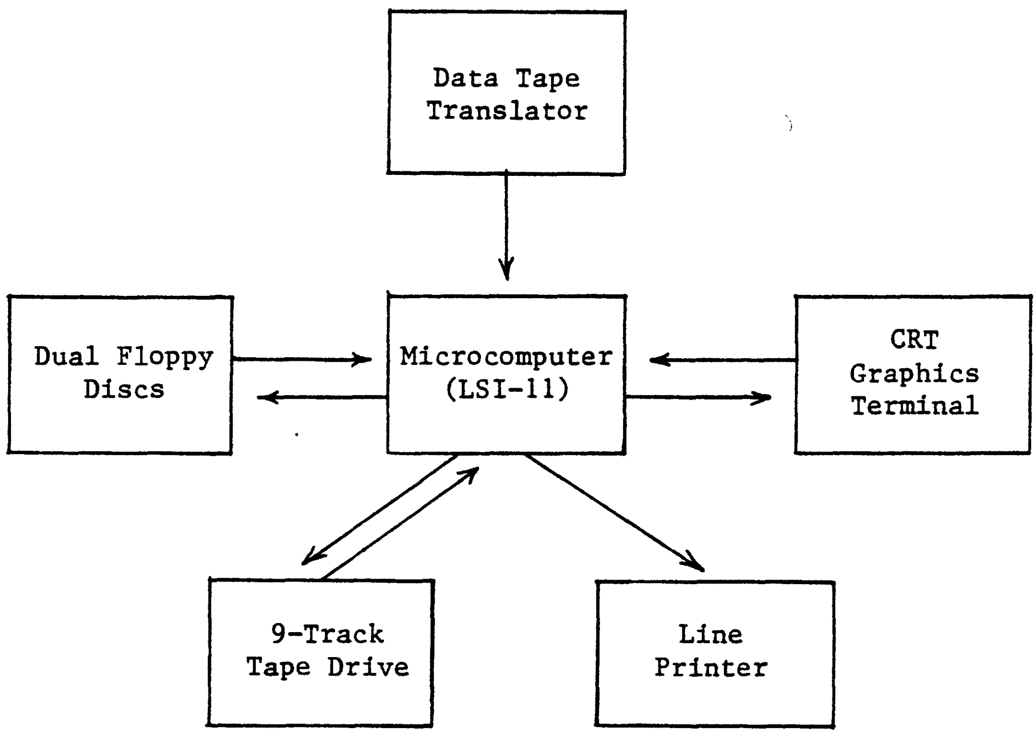

FIGURE 2. A microcomputer based current-meter data-acquisition system.

of RS-232 serial interface boards for communication with peripheral devices, and printer and floppy-disc controllers. The LSI-11 microprocessor has computing capability virtually equivalent to the CPU on the PDP-11/03 microcomputer and is supported by an extensive software library including the RT-11 operating system and FORTRAN and BASIC compilers.

The mass storage media are a dual floppy-disc-drive subsystem, and a 9-track tape drive. The floppy-disc subsystem contains two disc drives, a formatter, and a disc controller to interface with the microcomputer. Diskettes employed may be recorded on both sides in double density to provide a usable capacity of 1.2 mega-bytes. The data-reduction programs are written in FORTRAN by the users and reside on the system disc to provide maximum working space for data on the other drive.

A Tektronix-4010 graphics terminal is the primary input/output control for the system. Since the terminal is a graphics terminal, the Tektronix PLOT-10 plotting software package is used to produce various computer-generated graphics (Figures 3 to 8). Additionally, the data-processing system includes a Printronix-300 line printer for any hard copy of data, tables, or graphics.

Data processing procedures include the translation of 8-track tapes to binary file for storage on disc. The binary data file is further processed to produce engineering unit files, as well as files of various averages suitable for graphics and statistical analysis. This is accomplished through running a series of data-reduction programs (see Appendix C). The initial program, "Step 1", is used in conjunction with the data translator (ENDECO-173) to translate the 8-track tapes into binary form and store the data on flexible disc. "Step 2" is then run to check values of the conductivity corrections and to check for missing hour blocks between the 24-hour readings. "Step 2A" is used to check for data completeness. This program searches the data file for any incomplete hour blocks and checks the file beginning and ending times against the logged times recorded at deployment and recovery of the current meter. "Step 3" may be executed to check for questionable temperature and conductivity data by searching for and flagging any data changes between consecutive readings greater than 5 percent of the sensor range. The data files can then be edited for obvious mistakes before the data are translated to engineering units. The engineering-unit data file is produced from program "Step 4", which converts binary data into engineering units, incorporating calibration data of individual sensors. This program outputs the engineering-unit file to either disc or 9-track tape. The program "Step 4A" was written to produce a time-averaged data file (speed and direction are vector averaged). Normally a 30-minute data file, which was produced from the original 2-minute sampling, is used for all subsequent plotting routines or tabulations.

A collection of other miscellaneous data-reduction programs is included in this report. They include "Step 5" and "Step 5A", which are plotting routines for parameter vs. time for 1 week of data and for the entire tape respectively. "Step 6" is a plotting routine for the tidal ellipse; "Step 7" and "Step 7A" are for plotting progressive vector diagrams for 1 week of data at a time or for the entire tape and "Step 8" plots a histogram of percent of observations for speed and direction.

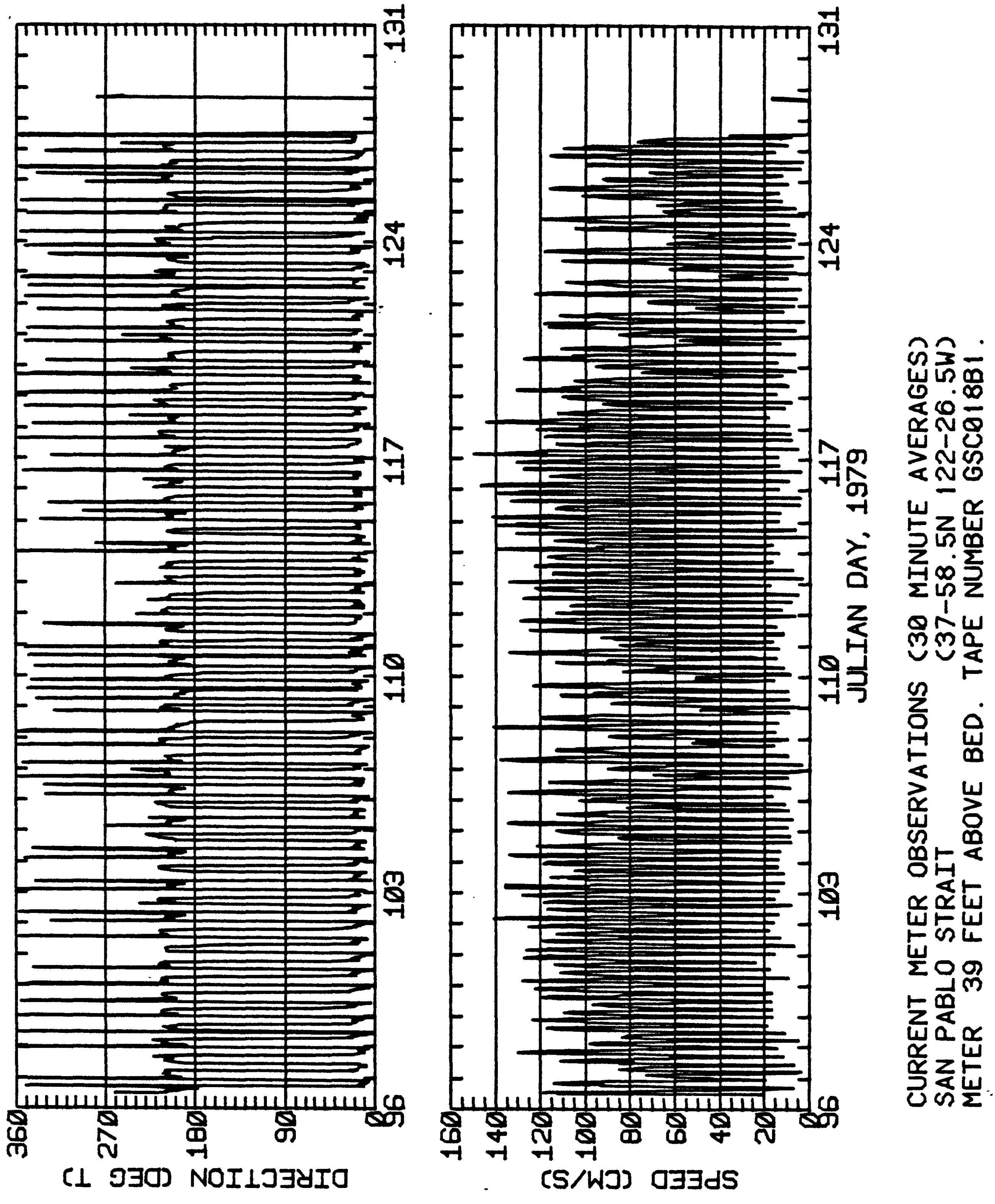

Fig. 3. Typical speed and direction data for entire deployment

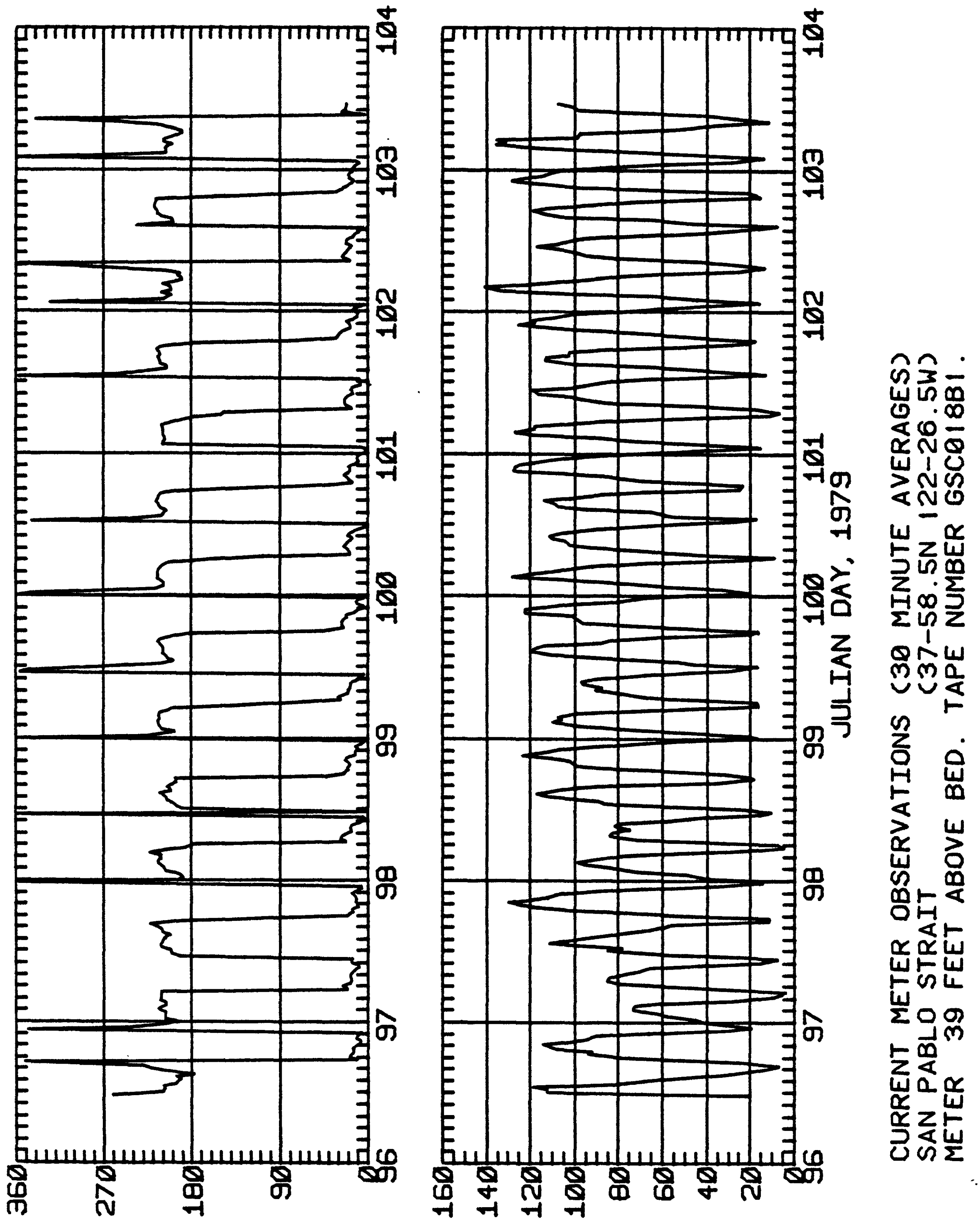

Fig. 4. Typical speed and direction data for 1 week of deployment

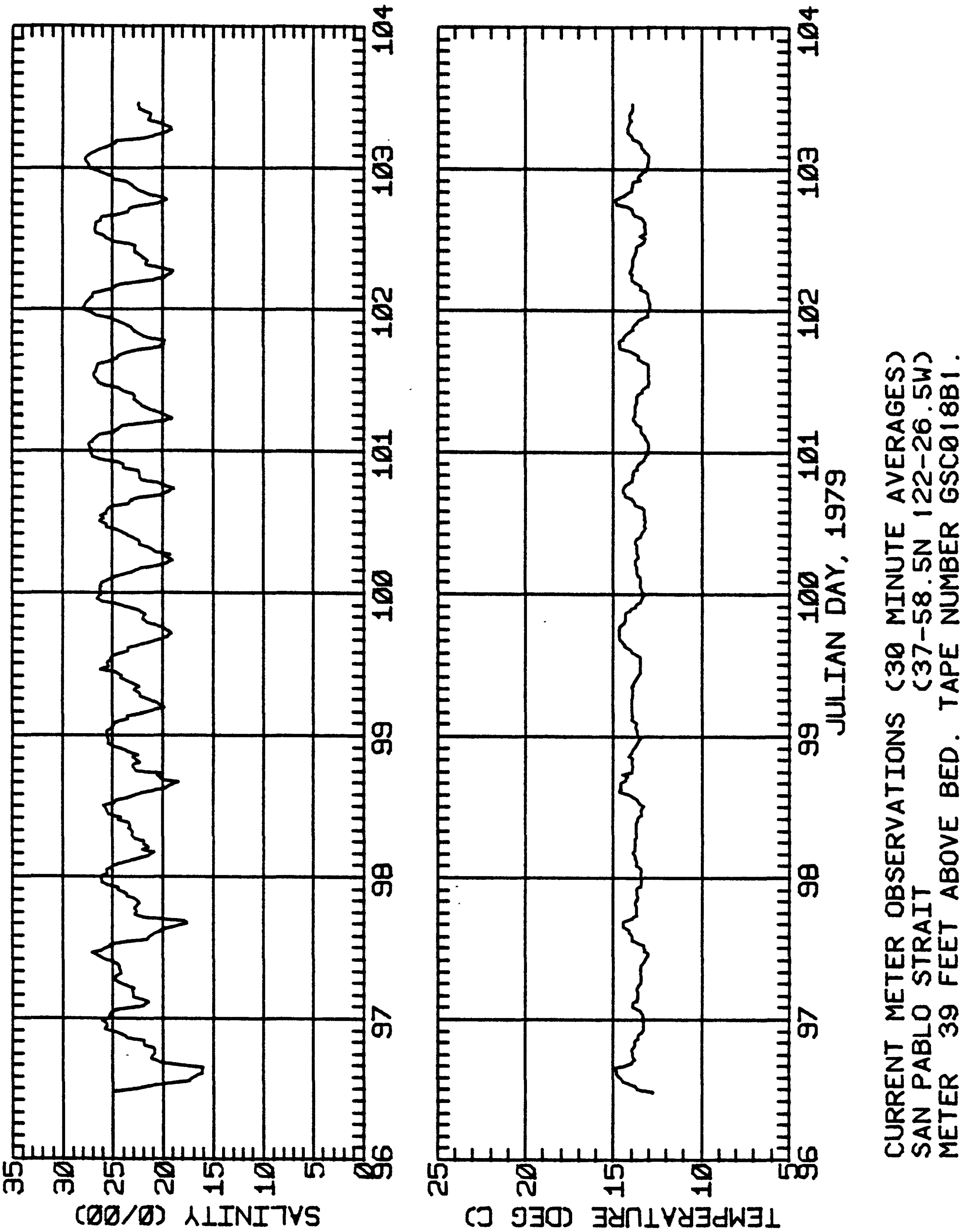

Fig. 5. Typical temperature and salinity data for 1 week of deployment

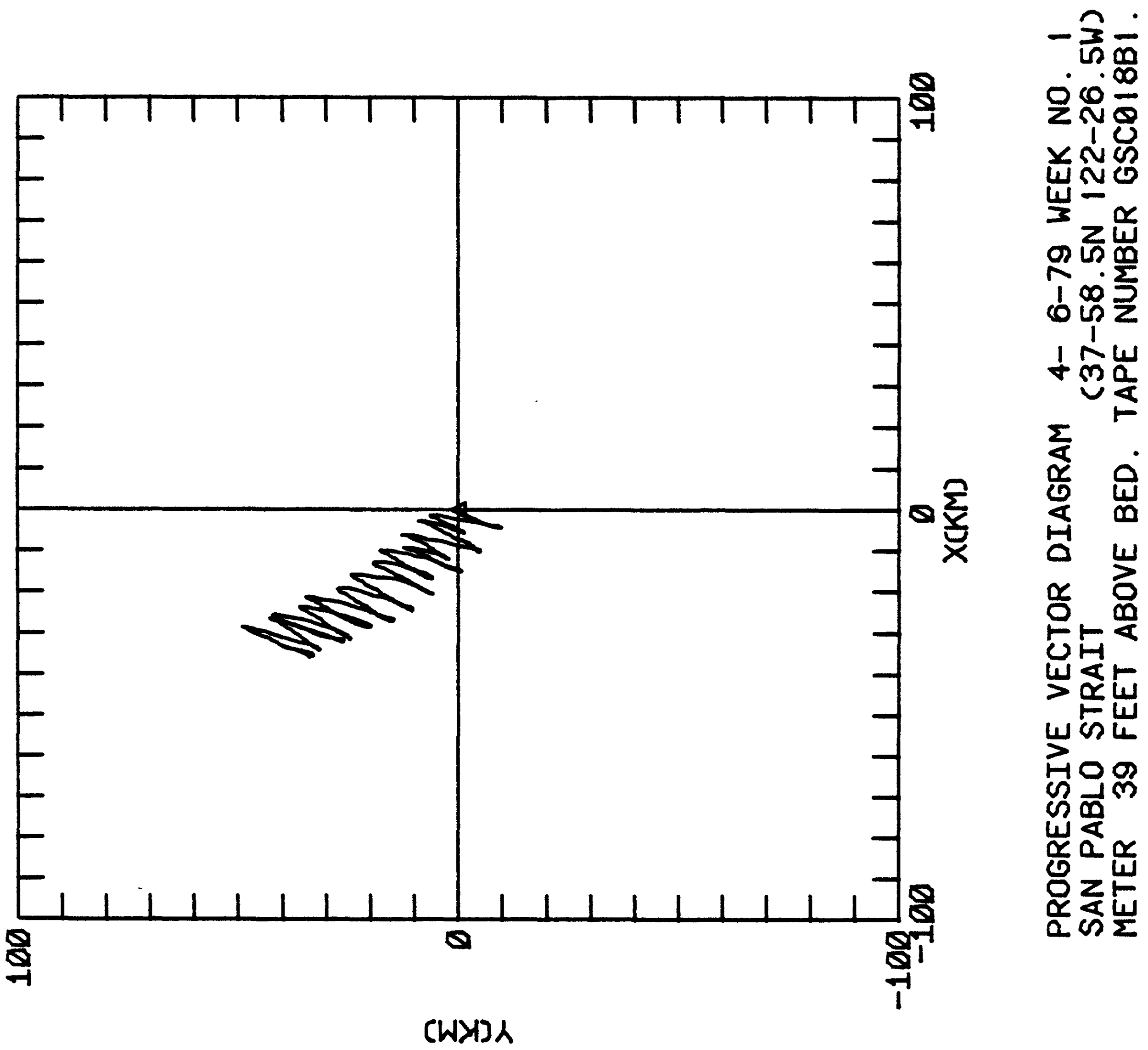

Fig. 6. Progressive vector diagram utilizing 1 week of data

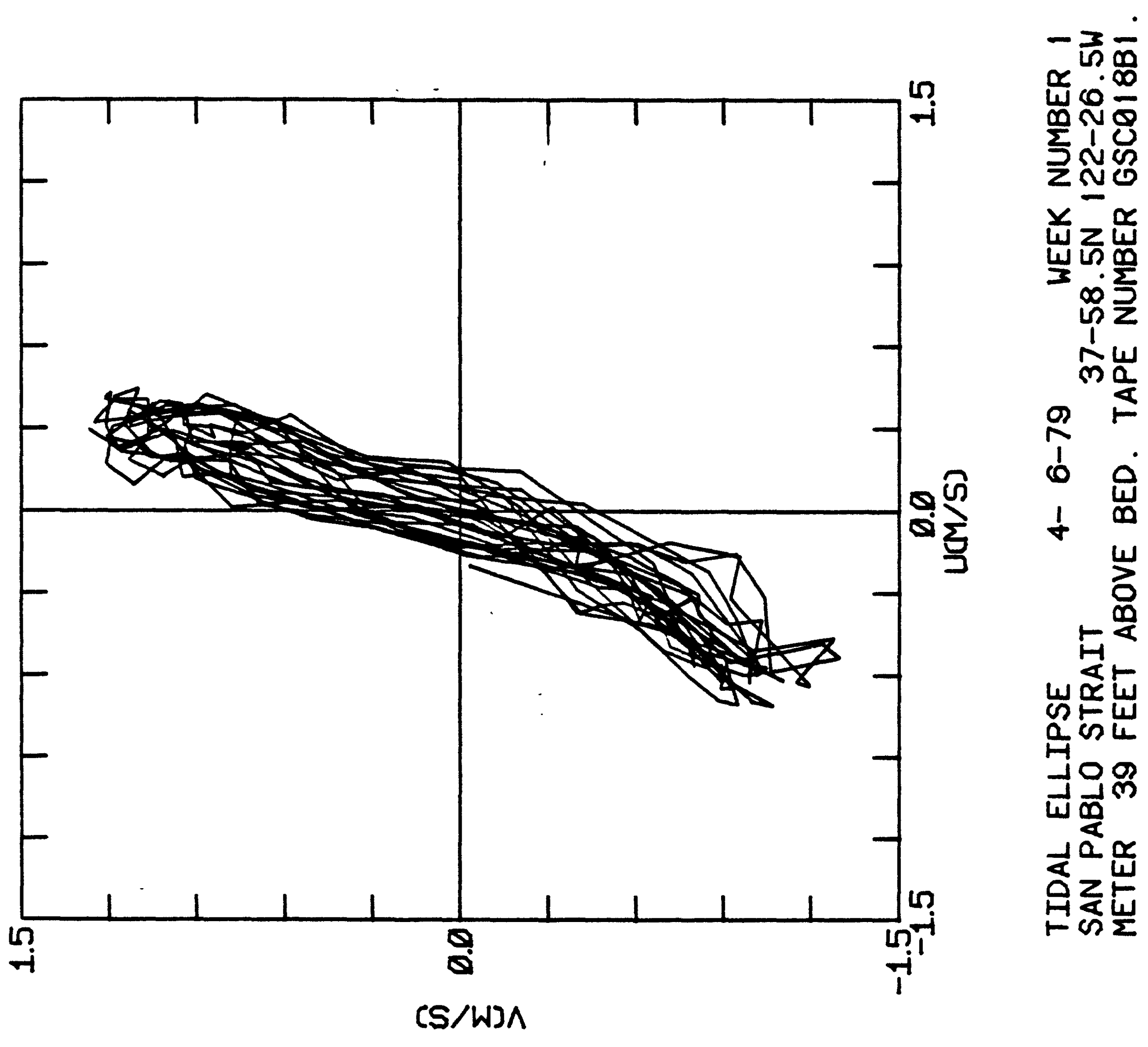

Fig. 7. Tidal ellipse utilizing 1 week of data

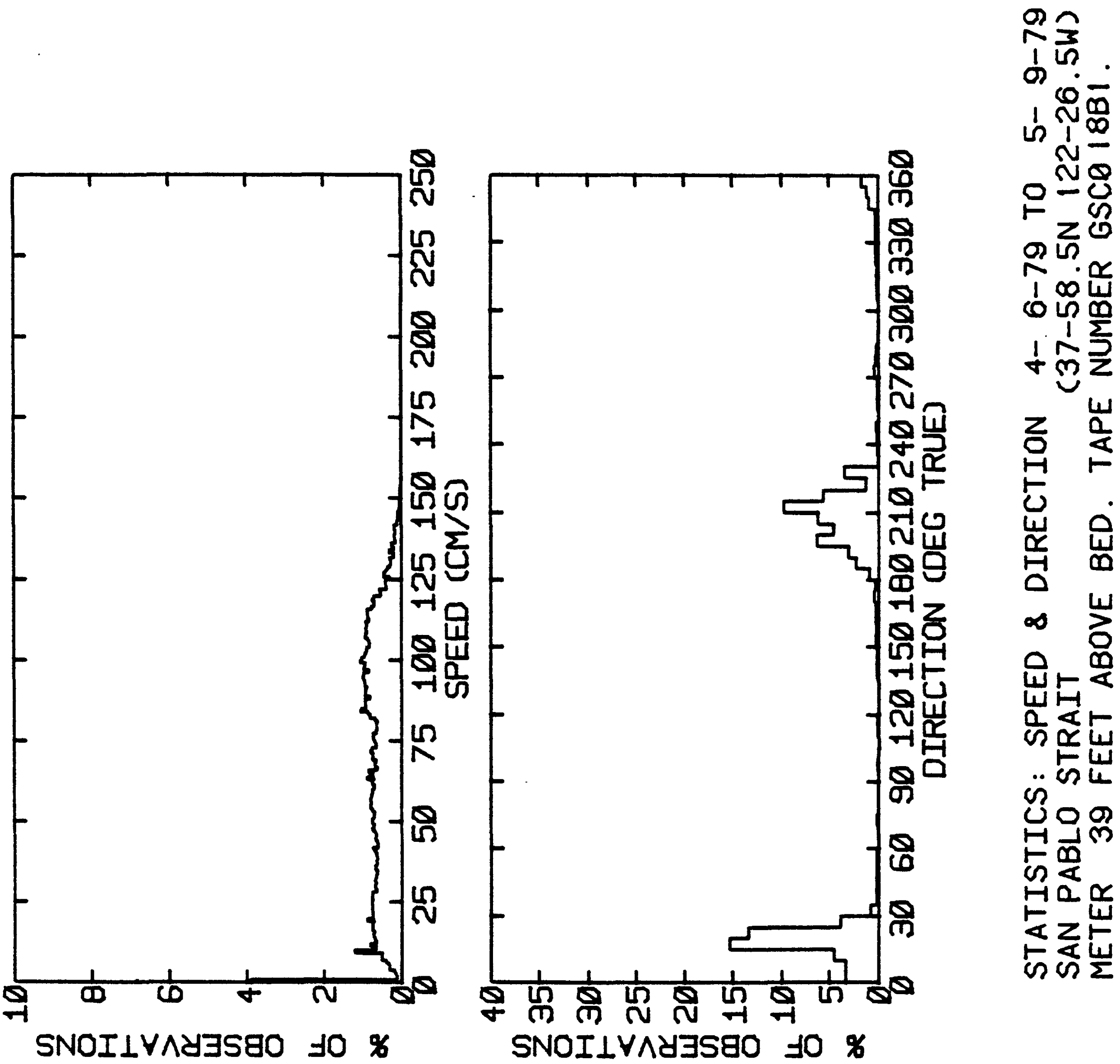

Fig. 8. Histogram for speed and direction utilizing data from entire deployment

DISCUSSION

The ENDECO-174 current meter is a relatively new instrument designed for current measurements in an estuarine or near-shore environment. While current measurement in an estuary is a relatively new task for us, we have successfully worked out our current-meter deployment and recovery procedures and streamlined our data-processing methods for handling large volumes of current-meter data. Employment of the ENDECO-174 meters by the Geological Survey in our data-collection program has also served as an extensive field evaluation of the instrument. During the early stage of working with the ENDECO-174 current-meters, many suggestions for modification of the current meter have been made to the manufacturer. The suggestions have led to subsequent retromodification of the meters and, in turn, improvements in their reliability and performance. Many facts about the instruments, learned through personal contacts with the manufacturer, have proved valuable additions to documentation provided by the manufacturer. This open-file report was prepared to document our two-year working experience with ENDECO-174 current-meters and to assist other ENDECO-174 current-meter users.

REFERENCES

Cheng, R. T., 1978, Comparison of a few recording current meters in San Francisco Bay, California, *in* Woodward, W., Mooers, C. N. K., and Jensen, K. eds. Proceedings of a working conference on current measurement: Technical Report DEL-SG-3-78, University of Delaware, Newark, Del., January 11-13, 1978, p. 293-301.

Cheng, R. T., and Gartner, J. W., 1979, A microcomputer based system for current-meter data acquisition, *in* Marine Technology Society-Institute of Electrical and Electronics Engineers Oceans '79 Conference, San Diego, California, September 17-19, 1979, Proceedings, p. 288-291.

Cheng, R. T., and Walters, R. A., 1978, Some finite element applications in environmental hydrodynamics: United States - Japan Seminar on Interdisciplinary Finite Element Analysis, Cornell University, August 7-11, 1978, 33 p.

Conomos, T. J., Nichols, F. H., Cheng, R. T., and Peterson, D. H., 1978, Field and modeling studies of San Francisco Bay, *in* Coastal Zone '78: Symposium on Technical, Environmental, Socioeconomic and Regulatory Aspects of Coastal Zone Management, ASCE, March 14-16, 1978, San Francisco, Calif., Proceedings, p. 1917-1927.

Environmental Devices Corporation, 1978, ENDECO Type 174 digital magnetic tape recording tethered current meter; Service Manual: Marion, Mass., 13 p.

Innerspace Technology Incorporated, 1978, Operation and maintenance manual, Model 430/431, acoustic release system: Waldwick, N.J., 41 p.

McCullough, J. R., 1977, Problems in measuring currents near the ocean surface, *in* Marine Technology Society-Institute of Electrical and Electronics Engineers '77 Conference, Los Angeles, Calif., Proceedings, p. 46A1-46A7.

Richards, R. K., 1971, Digital design, New York, John Wiley, 571 p.

Walters, R. A., and Cheng, R. T., 1979, A two-dimensional hydrodynamic model of a tidal estuary: Advances in water resources, Vol. 2, December 1979, p. 177-184.

APPENDIX A

ENDECO CURRENT METER PREDEPLOYMENT CHECK LIST

1. Remove impeller to access bearing assembly. Remove bearing assembly, clean and check bearings for free movement and replace if necessary. Reassemble bearing assembly and impeller.
2. Check external appearance of meter. Repair and paint as necessary.
3. Insert new batteries (Eveready No. 1150 Size D) and check polarity. Record battery voltage.
4. Check pinger operation.
5. Place cardboard spacer between batteries and contacts.
6. Fast forward magnetic tape on a regular 8-track player at least once to ascertain that tape is free to feed out of hub without jamming. Degauss tape and place in disiccator. The tape used is Scotch Dynarange (low noise) 90-minute length (Scotch S-8TR-90).
7. Remove old "O" ring from end cap and clean "O" ring groove. Install new, greased "O" ring in groove after visually checking "O" ring for any nicks or damage.
8. Clean and demagnetize recording head.

   The following steps should be completed within 24 hours of deployment.
9. Remove cardboard circuit breaker from between batteries and contacts.
10. Center internal trim weight ($\sim$ 15 cm from end).
11. Remove tape from desiccator. Advance sensing foil on tape 6 inches past capstan and load tape in current meter. Close tape cartridge retainer.
12. Install desiccant bags in meter.
13. Press red reset button on meter. After motor stops press reset again. When motor stops, enter tape-on time on deployment log sheet.

14. Purge ambient air from meter with Freon-12 and immediately replace end cap making sure that the "O" ring is not damaged in the process.
15. Tighten knurled-nylon nuts finger tight, advancing all three nuts at same rate to avoid stripping the nuts.

NOTE: When replacing any metal parts (nuts, cotter pins, and so forth) use only stainless steel replacements where metal contacts metal.

APPENDIX B

ENDECO CURRENT METER POSTDEPLOYMENT CHECK LIST

1. Check external appearance of meter. Record any condition which might affect data quality on the deployment log-sheet (loss of impeller blades, growth on impeller, sea grass around impeller shaft, and so forth).
2. Remove end cap and press reset button. Enter time (tape-off time) on log-sheet. When motor stops, press reset button again.
3. Unload 8-track tape.
4. Replace end cap and clean all external surfaces of any marine growth or sediment.
5. Inspect tether line for any unusual wear.
6. Remove impeller, disassemble bearing assembly for cleaning and inspection. When complete, reassemble.
7. If time permits, reduce data tape to verify condition of recording quality prior to next deployment.

APPENDIX C

```
C       PROGRAM STEP1
C       THIS COMPUTER PROGRAM IS USED TO TRANSLATE ENDECO CURRENT
C       METER DATA.  THE DATA TAPE IS PLAYED BACK BY ENDECO-173.
C       THE RAW DATA IS NOW STORED IN THE FILE CALLED
C       'RAW.DAT'.  THE FIRST RECORD IS DESCRIPTIVE INFORMATION.
C       THE DATA ARE GIVEN IN BINARY; I.E.  0-255.

C
        COMMON /RBLOCK/ B,G,NBYTE,IBLKND,NBLK
        INTEGER DAY,DDAY,HOUR,RECLGH
        INTEGER G(300),B(300)
        EXTERNAL RBYTE
        LOGICAL RAW
        DIMENSION LINE(32),TAPE(4)
        TYPE 2001
 2001   FORMAT('  INITIALIZING ENDECO-173 DATA TRANSLATION',/,
       1'  PROGRAM.****DO NOT FORGET TO TURN ON THE TIMER',/,
       2'  IS THE FILE RAW.DAT TO BE SAVED? (T/F)')
        ACCEPT 1005,RAW
 1005   FORMAT(L1)
        IF(RAW) GO TO  5000
        RECLGH=120
        TYPE 2012
 2012   FORMAT('  ESTIMATE THE TOTAL AMOUNT OF DATA TO BE TRANS',
       1'LATED',/,'  DEFINE THE NUMBER OF RECORDS FOR FILE RAW.DAT'
       2,/,'  NRECD=(I3)?')
        ACCEPT 1000,NRECD
        OPEN(UNIT=1,NAME='RAW.DAT',RECORDSIZE=120,
       1INITIALSIZE=NRECD,FORM='FORMATTED')
        NBLK=1
 1000   FORMAT(I3)
        TYPE 2004
 2004   FORMAT(' WHAT WAS THE METER NUMBER? (I4)')
        ACCEPT 2003,ISN
 2003   FORMAT(I4)
        TYPE 2002
 2002   FORMAT (' WHAT IS THE TAPE SERIAL NUMBER? (4A2)  ' )
        ACCEPT 1001,TAPE
 1001   FORMAT (4A2)
        TYPE 2006
 2006   FORMAT (' ENTER THE DATA  STARTING DATE AND HOUR  '
       1'(MMDDYY HHMM) (3I2,1X,I4)')
        ACCEPT 1003,MON,DAY,IYR,HOUR
 1003   FORMAT (3I2,1X,I4)
        TYPE 2007
 2007   FORMAT(' ENTER THE DATA ENDING DATE (3I2)')
        ACCEPT 1007,MMON,DDAY,IIYR
 1007   FORMAT(3I2)
        TYPE 3001
 3001   FORMAT(' WHAT IS THE DATA INTERVAL?(I4)')
        ACCEPT 3002,INT
 3002   FORMAT(I4)
        TYPE 3003
 3003   FORMAT(' WHAT IS THE UTM GRID POSITION? (F5.1,1X,F5.1)')
        ACCEPT 3004,UTMN,UTME
 3004   FORMAT(F5.1,1X,F5.1)
        TYPE 2009
 2009   FORMAT(' WHAT WAS THE STATION LATITUDE AND LONGITUDE?
       1(DD,1X,MM.M,1X,DDD,1X,MM.M)')
        ACCEPT 1009,LAT,ALAT,LONG,ALONG
```

```
1009  FORMAT(I2,1X,F4.1,1X,I3,1X,F4.1)
      TYPE 3007
3007  FORMAT(' WHAT WAS THE MLLW WATER DEPTH? (I3)')
      ACCEPT 3008,IDEPTH
3008  FORMAT(I3)
      TYPE 2005
2005  FORMAT(' WHAT WAS THE METER HEIGHT ABOVE BOTTOM? (I3)')
      ACCEPT 3005,IHITE
3005  FORMAT(I3)
      TYPE 2008
2008  FORMAT(' TYPE IN ANY LINE OF INFORMATION YOU WANT ',/,
     1' TO BE PLACED IN THE FIRST RECORD OF THE  FILE.'/)
      ACCEPT 1004,(LINE(I),I=1,32)
1004  FORMAT(32A2)
      WRITE(1,2017) (LINE(I),I=1,32),TAPE,ISN,LAT,ALAT,LONG,ALONG,
     1UTMN,UTME,IHITE,IDEPTH,MON,DAY,IYR,HOUR,MMON,DDAY,IIYR,INT
C     NBLK=2
      WRITE (6,2017) (LINE(I),I=1,32),TAPE,ISN,LAT,ALAT,LONG,ALONG,
     1UTMN,UTME,IHITE,IDEPTH,MON,DAY,IYR,HOUR,MMON,DDAY,IIYR,INT
2017  FORMAT(' REMARKS: ',32A2,//,
     1' TAPE NUMBER:   ',4A2,/,
     2' METER NUMBER:  ',I4,/,
     3' POSITION:      ',I3,F5.1,'N',1X,I3,F5.1,'W',/,
     4' UTM POSITION:  ',F8.1,'N',1X,F8.1,'E',/,
     5' METER HEIGHT:  ',I4,' FEET',/,
     6' WATER DEPTH:   ',I4,' FEET',/,
     7' STARTING DATE:',I2,2('-',I2),I6,1X,'PST',/,
     8' RECOVERY DATE:',I2,2('-',I2),/,
     9' DATA INTERVAL:',I4,' MIN')
      DO 555 I=1,120
      B(I)=0
555   CONTINUE
      CALL TRANS
      GO TO 6000
5000  TYPE 1006
1006  FORMAT(' TRANSFER FILE RAW.DAT TO OTHER NAME AND RERUN THIS',/,
     1' PROGRAM ')
6000  RETURN
      END
      SUBROUTINE RBYTE(ID)
      COMMON /RBLOCK/B,G,NBYTE,IBLKND,NBLK
      INTEGER G(300),B(300)
      NBYTE=NBYTE+1
      G(NBYTE)=IPEEK("176512)
      IF(G(NBYTE).NE."377) GO TO 1
      IF(G(NBYTE-1).NE."12) GO TO 1
      IF(G(NBYTE-2).NE."215) GO TO 1
      IBLKND=1
1     RETURN
      END
      SUBROUTINE TRANS
      COMMON /RBLOCK/ B,G,NBYTE,IBLKND,NBLK
      EXTERNAL RBYTE
      INTEGER G(300),B(300)
C
C     SET UP INTERRUPT VECTOR
C
      I=INTSET("310,7,0,RBYTE)
      TYPE 1001
5     ACCEPT 1002,I
```

```
      IF(I.EQ.0) GO TO 20
10    TYPE 1005
      GO TO 5
20    CONTINUE
      LBLOCK=0
      NBLOCK=0
      MBLOCK=1
      NBYTE=0
      IBLKND=0
      I=IPEEK("176512)
      ITPEND=0
C
C     ENABLE INTERRUPT
C
      CALL IPOKE("176510,"100)
C
C     SEND ASCII ACK
C
100   CALL IPOKE("176516,"6)
101   CONTINUE
      I=IPEEK("176516)
      TO=SECNDS(0.)
200   T=SECNDS(TO)
      IF(IBLKND.EQ.1) GO TO 300
      IF(T.LT.10.) GO TO 200
      TYPE 1006
      IF(ITPEND.EQ.1) GO TO 400
      IBLKND=0
      ITPEND=1
      GO TO 100
300   CALL GRYBIN
      MBYTE=NBYTE-5
      OBYTE=NBYTE-3
      IF(MBYTE.EQ.120) GO TO 310
      TYPE 1015
310   IF(LBLOCK.GT.G(OBYTE)) MBLOCK=MBLOCK+256
      NBLOCK=MBLOCK+G(OBYTE)
      LBLOCK=G(OBYTE)
      IF(NBLOCK.GT.999)NBLOCK=NBLOCK-999
      IF(MBYTE.EQ.120)GO TO 315
      TYPE 103,NBLOCK,MBYTE
      TYPE 103,(B(I),I=1,MBYTE)
315   WRITE(1,1003) NBLOCK,MBYTE
      MBYTE=120
320   WRITE(1,1003)(B(I),I=1,MBYTE)
      NBYTE=0
      NTRY=0
      NTRY=0
      IBLKND=0
      ITPEND=0
      GO TO 100
1001  FORMAT(' ENTER 0 WHEN ENDECO-173 IS READY')
1002  FORMAT(I1)
1003  FORMAT(20I3)
1005  FORMAT('****WAKE UP ,TYPE 0 AND <CR>****')
1006  FORMAT(' END OF DATA TAPE OR TIMEOUT ERROR')
103   FORMAT(1X,8I7)
1015  FORMAT('  WARNING: POSSIBLE MISSING DATA!     ')
400   RETURN
      END
```

```
      SUBROUTINE GRYBIN
      COMMON /RBLOCK/ B,G,NBYTE,IBLKND,NBLK
      INTEGER GRAY
      INTEGER G(300),B(300)
      DIMENSION GRAY(16)
      DATA GRAY/0,1,3,2,7,6,4,5,15,14,12,13,8,9,11,10/
      MBYTE=NBYTE-5
      DO 100 I=1,MBYTE,4
      NGH=G(I)/16
      NGL=G(I)-NGH*16
      NBH=GRAY(NGH+1)
      EVEN=NBH/2.0
      IEVEN=NBH/2
      IF(EVEN.EQ.IEVEN) GO TO 50
      NBL=15-GRAY(NGL+1)
      GO TO 60
50    NBL=GRAY(NGL+1)
60    B(I)=NBH*16+NBL
      NGH=G(I+1)/16
      NGL=G(I+1)-NGH*16
      NBH=GRAY(NGH+1)
      EVEN=NBH/2.0
      IEVEN=NBH/2
      IF(EVEN.EQ.IEVEN) GO TO 70
      NBL=15-GRAY(NGL+1)
      GO TO 80
70    NBL=GRAY(NGL+1)
80    B(I+1)=NBH*16+NBL
      B(I+2)=G(I+2)
      B(I+3)=G(I+3)
100   CONTINUE
      IF(B(1).EQ.170.AND.B(2).EQ.170) GO TO 150
      GO TO 200
150   DO 160 I=3,4
      NGH=B(I)/16
      NGL=B(I)-NGH*16
      NBH=GRAY(NGH+1)
      EVEN=NBH/2.0
      IEVEN=NBH/2
      IF(EVEN.EQ.IEVEN) GO TO 155
      NBL=15-GRAY(NGL+1)
      GO TO 156
155   NBL=GRAY(NGL+1)
156   B(I)=NBH*16+NBL
160   CONTINUE
200   CONTINUE
      RETURN
      END
```

```
C      PROGRAM STEP2
C      THE PROGRAM IS DESIGNED TO READ AND WRITE ALL THE
C      CONDUCTIVITY CORRECTIONS FOR ANY TAPE.
C
C      DATA(I)=BINARY NUMBERS
C      IDATA(1)=BLOCK NUMBER; IDATA(2)=NUMBER OF DATA IN BLOCK
C
       PAUSE' INSERT THE DATA DISK AND TYPE  <CR>. '
       DIMENSION IDATA(2),DATA(120),LINE(32)
       INTEGER DATA
       CALL ASSIGN(1,'RAW.DAT')
3001   FORMAT(2I3)
3002   FORMAT(/,2X,'THE BLOCK NUMBER IS',1X,I3,/,2X,'THE NUMBER
      * OF DATA IN THE BLOCK ARE',1X,I3)
3003   FORMAT(20I3)
3004   FORMAT(2X,'THE LOW CONDUCTIVITY CHECK IS',1X,I3,/,2X,'THE
      * HIGH CONDUCTIVITY CHECK IS',1X,I3)
3005   FORMAT(80A2,/)
       DO 1000 J=1,11
       READ(1,3005)(LINE(I),I=1,32)
       WRITE(6,3005)(LINE(I),I=1,32)
1000   CONTINUE
       K=0
       DO 2000 J=1,1000
       READ(1,3001)(IDATA(I),I=1,2)
       READ(1,3003)(DATA(I),I=1,120)
C       THIS PERMITS THE FIRST CONDUCTIVITY CORRECTIONS TO BE WRITTEN
       IF(IDATA(2).GT.60.AND.K.EQ.0)GO TO 2002
C       THE LAST BLOCK IN ANY 24 HOUR PERIOD ENDS WITH 170170255255
C       THE NEXT 2 CONDUCTIVITY VALUES ARE THE CONDUCTIVITY CORRECTIONS.
       IF(DATA(117).EQ.170.AND.DATA(118).EQ.170)GO TO 2001
       GO TO 2000
2001   IF(DATA(119).EQ.255.AND.DATA(120).EQ.255)GO TO 2004
       GO TO 2000
2004   READ(1,3001)(IDATA(I),I=1,2)
       READ(1,3003)(DATA(I),I=1,120)
2002   WRITE(6,3002)IDATA(1),IDATA(2)
       WRITE(6,3004)DATA(4),DATA(8)
       K=1
2000   CONTINUE
       STOP
       END
```

```
C       PROGRAM STEP2A
C       THE PROGRAM IS DESIGNED TO CHECK DATA COMPLETENESS
C
C       DATA(I)=BINARY NUMBERS
C       IDATA(1)=BLOCK NUMBER; IDATA(2)=NUMBER OF DATA IN BLOCK
C
        TYPE 2003
2003    FORMAT(' ENTER THE TAPE START JULIAN DATE AND TIME (I3,
       *1X,2I2)')
        ACCEPT 2005,IDATE,IHR,MIN
C       INITIALIZE ISTORE
        DO 10 J=1,10
10      ISTORE(J)=0
2005    FORMAT(I3,1X,2I2)
        IHR=IHR-1
        MIN=MIN+2
        IDATE=MOD(IDATE-1+(60*IHR+MIN)/1440,365)+1
        IHR=MOD(IHR+MIN/60,24)
        MIN=MOD(MIN,60)
        DIMENSION IDATA(2),DATA(120),LINE(32),ISTORE(10)
        INTEGER DATA
        CALL ASSIGN(1,'RAW.DAT')
3001    FORMAT(2I3)
3002    FORMAT(/,2X,'THE BLOCK NUMBER IS',1X,I3,/,2X,'THE NUMBER
       * OF DATA IN THE BLOCK ARE',1X,I3)
3003    FORMAT(20I3)
3005    FORMAT(80A2,/)
        DO 1000 J=1,11
        READ(1,3005)(LINE(I),I=1,32)
        WRITE(6,3005)(LINE(I),I=1,32)
1000    CONTINUE
2001    DO 2000 J=1,1000
        READ(1,3001,END=400)(IDATA(I),I=1,2)
        READ(1,3003,END=400)(DATA(I),I=1,120)
        IF(IDATA(2).LT.60.AND.J.EQ.1)GO TO 2001
        IHR=IHR+1
        IDATE=MOD(IDATE-1+(60*IHR+MIN)/1440,365)+1
        IHR=MOD(IHR+MIN/60,24)
        IF(IDATA(2).LT.120)WRITE(6,3002)IDATA(1),IDATA(2)
        IF(IDATA(2).GT.120)WRITE(6,3002)IDATA(1),IDATA(2)
        DO 20 K=2,10
20      ISTORE(K-1)=ISTORE(K)
        ISTORE(10)=IDATA(2)
2000    CONTINUE
C       BECAUSE THE TAPES END WITH 2 RESETS AND THE SECOND
C       ONE ALWAYS BEGINS A NEW DATA BLOCK, IT IS NECESSARY
C       TO USE THE FIRST ONE TO COMPUTE THE TIME.
400     DO 30 K=1,10
        IDX=11-K
        IF(ISTORE(IDX).NE.4)GO TO 40
30      CONTINUE
        WRITE(6,6001)
6001    FORMAT(' MORE THAN 10 RESETS')
40      MIN=MIN+(ISTORE(IDX)/2)
        IHR=IHR-10+IDX
        IDATE=MOD(IDATE-1+(60+IHR+MIN)/1440,365)+1
        IHR=MOD(IHR+MIN/60,24)
        MIN=MOD(MIN,60)
        WRITE(6,2002)IDATE,IHR,MIN
2002    FORMAT(/,' THE TAPE END DATE AND TIME IS'1X,I3,1X,2I2)
```

```
STOP
END
```

```
C       PROGRAM STEP3
C       THIS PROGRAM CHECKS THE BINARY NUMBERS IN
C       TEMPERATURE AND CONDUCTIVITY FOR A CHANGE
C       OF 5% OR MORE
C
C       DATA(I)=BINARY NUMBERS
C       FDATA(I)=PHYSICAL PARAMETERS
C       IDATA(1)=BLOCK NUMBER; IDATA(2)=NUMBER OF DATA IN BLOCK
C
        PAUSE' INSERT THE DATA DISK AND TYPE  <CR>. '
        DIMENSION IDATA(2),DATA(120),LINE(32)
        INTEGER DATA
        CALL ASSIGN(1,'RAW.DAT')
3001    FORMAT(2I3)
3002    FORMAT(/,2X,'THE BLOCK NUMBER IS',1X,I3,/,2X,'THE NUMBER
       1 OF DATA IN THE BLOCK ARE',1X,I3)
3003    FORMAT(1X,20I3)
3004    FORMAT(20I3)
3005    FORMAT(80A2,/)
        DO 2003 J=1,11
        READ(1,3005)(LINE(I),I=1,32)
        WRITE(6,3005)(LINE(I),I=1,32)
2003    CONTINUE
2050    DO 2000 J=1,1000
        JJ=J
        READ(1,3001)(IDATA(I),I=1,2)
        READ(1,3004)(DATA(I),I=1,120)
        IF(IDATA(2).LT.60.AND.J.EQ.1)GO TO 2050
        DO 1000 I=1,113,4
        IF(IABS(DATA(I+2)-DATA(I+6)).GT.26)GO TO 2001
        GO TO 1000
C
C       THESE CHECK TO SEE IF DIFFERENCE IS SCALE
C       CHANGE OR 24 HOUR MARK
C
2001    IF(DATA(I+2).GE.229.AND.DATA(I+6).LE.26)GO TO 1000
        IF(DATA(I+2).LE.26.AND.DATA(I+6).GE.229)GO TO 1000
        IF(DATA(I+6).EQ.255.AND.DATA(I+2).NE.255)GO TO 1000
        WRITE(6,3002)IDATA(1), IDATA(2)
        WRITE(6,3003)(DATA(II),II=1,120)
        GO TO 1001
1000    CONTINUE
1001    DO 1050 I=1,113,4
        IF(IABS(DATA(I+3)-DATA(I+7)).GT.26)GO TO 2002
        GO TO 1050
2002    IF(I.EQ.1.AND.MOD(JJ,24).EQ.1)GO TO 1050
        IF(I.EQ.5.AND.MOD(JJ,24).EQ.1)GO TO 1050
C
C       THESE CHECK TO SEE IF DIFFERENCE IS SCALE CHANGE OR
C       24 HOUR MARK.
C
        IF(DATA(I+3).GE.229.AND.DATA(I+7).LE.26)GO TO 1050
        IF(DATA(I+3).LE.26.AND.DATA(I+7).GE.229)GO TO 1050
        IF(DATA(I+7).EQ.255.AND.DATA(I+3).NE.255)GO TO 1050
        WRITE(6,3002)IDATA(1),IDATA(2)
        WRITE(6,3003)(DATA(II),II=1,120)
        GO TO 2000
1050    CONTINUE
2000    CONTINUE
        STOP
```

END

```
C      PROGRAM STEP4
C      THIS PROGRAM TRANSLATES RAW DATA TO REAL UNITS,
C      APPLIES CORRECTIONS, AND CREATES A FILE CONSISTING
C      OF THE TWO MINUTE VALUES .
C
       PAUSE ' FOR DATA TAPES RECORDED PRIOR TO JUNE 1979,
      * TERMINATE PROGRAM & EDIT STEP4.FOR AS FOLLOWS:
      * PLACE A "C" IN COLUMN 1 PRIOR TO STATEMENT 9500
      *AND ITS CONTINUATION.  FOR TAPES AFTER
      * JUNE 1979, REMOVE THE COMMENT "C".  IF PROGRAM
      * IS CORRECT, TYPE <CR>. '
       DIMENSION DEV(24),BIN(3),BATH(3),TSTCON(2)
       DIMENSION LINE(32),DATA(120),IFILE(14)
       LOGICAL PRNT
       COMMON FDATA(120),IDATA(2),SAL(120)
       INTEGER DATA,DEV
1001   FORMAT(I4)
       TYPE 0002
0002   FORMAT(' WHEN * APPPEARS, TYPE MT: AND 6  CHARACTER
      *OUTPUT FILE NAME.',/)
       CALL ASSIGN(2,IFILE,-10)
       CALL ASSIGN(4,'CORR.DAT')
       CALL ASSIGN(1,'RAW.DAT')
       TYPE 0003
0003   FORMAT ('  WHAT WAS THE INSTRUMENT NUMBER (I3)?')
       ACCEPT 1003,ISN
1003   FORMAT(I3)
       TYPE 0004
0004   FORMAT ('  WHAT IS THE TOTAL NUMBER OF HOURS OF DATA? (I3)')
       ACCEPT 1004, IHRS
1004   FORMAT (I3)
       TYPE 0005
0005   FORMAT('  ENTER START JULIAN DATE, TIME, & LAST 2 DIGITS OF YEAR?
      1 (I3,1X,I4,1X,I2)')
       ACCEPT 1005,IDATE,IHR,MIN,IYEAR
1005   FORMAT(I3,1X,I2,I2,1X,I2)
       NM=365
       IF(MOD(IYEAR,4).EQ.0)NM=366
       MIN=MIN+02
       IDATE=MOD(IDATE-1+(60*IHR+MIN)/1440,NM)+1
       IHR=MOD(IHR+MIN/60,24)
       MIN=MOD(MIN,60)
       TYPE 0006
0006   FORMAT('  WHAT IS THE HIGHEST EXPECTED CONDUCTIVITY (I2)?')
       ACCEPT 1006, COND
1006   FORMAT(F2.0)
       TYPE 0007
0007   FORMAT('  SKIP HOW MANY HOURS UNTIL DATA IS DESIRED (I3)?')
       ACCEPT 1007,JHOURS
1007   FORMAT(I3)
       TYPE 0009
0009   FORMAT('  WHAT IS THE VARIATION?  ADD SIGN
      1 FOR DIRECTION.  (EAST=+;WEST=-) (F5.2)')
       ACCEPT 1009, VAR
1009   FORMAT(F5.2)
C
C      READ FILE HEADER
C
       DO 150 J=1,11
3004   FORMAT(80A2,/)
```

```
      READ(1,3004)(LINE(I),I=1,32)
      WRITE(2,3004)(LINE(I),I=1,32)
150   CONTINUE
3001  FORMAT(2I3)
3003  FORMAT(20I3)
      MDATA=0
3006  FORMAT(F3.0)
3007  FORMAT(24I3)
3008  FORMAT(3F4.1)
3009  FORMAT(3F3.0)
3010  FORMAT(2F5.2)
C
C     READ CURRENT METER CALIBRATION VALUES
C
      READ(4,3006)COUNT
      READ(4,3007)(DEV(I),I=1,24)
      READ(4,3009)(BIN(I),I=1,3)
      READ(4,3008)(BATH(I),I=1,3)
      READ(4,3010)(TSTCON(I),I=1,2)
      DO 250 J=1,IHRS
      MJ=J
151   READ(1,3001)(IDATA(I),I=1,2)
      IF(IDATA(2).GT.120)IDATA(2)=120
      READ(1,3003)(DATA(I),I=1,120)
      IF(IDATA(2).LT.60.AND.J.EQ.1)GO TO 151
      IF(MOD(MJ,24).EQ.1)GO TO 153
      GO TO 154
153   ALCON=DATA(4)
      HCON=DATA(8)
C
C     THIS PUTS CONDUCTIVITY CALIBRATION VALUES ON CORRECT
C     SCALE DEPENDING ON METER NUMBER.
C
C9500  IF(ISN.EQ.31.OR.ISN.EQ.37.OR.ISN.EQ.40.OR.
C     *ISN.EQ.41.OR.ISN.EQ.42)GO TO 9501
      IF(ALCON.GT.191.)ALCON=ALCON-256
      IF(HCON.LT.64.)HCON=HCON+256
      HCON=HCON-256.
      GO TO 9502
9501  IF(HCON.GT.191.)HCON=HCON-256
      IF(ALCON.LT.64.)ALCON=ALCON+256
9502  IF(MOD(MJ,24).EQ.1)ALCON=ALCON+256
      DATA(4)=DATA(12)
      DATA(8)=DATA(12)
C
C     THE FIRST TWO CONDUCTIVITY READINGS ARE CONDUCTIVITY CHECKS AND
C     THE DATA VALUES USED ARE MEARLY FROM THE NEXT REAL DATA VALUE
C
154   IF(MOD(MJ,24).EQ.0)GO TO 156
      GO TO 157
156   DATA(117)=(DATA(113)-DATA(109))+DATA(113)
      DATA(118)=(DATA(114)-DATA(110))+DATA(114)
      DATA(119)=(DATA(115)-DATA(111))+DATA(115)
      DATA(120)=(DATA(116)-DATA(112))+DATA(116)
C
C     THE LAST READINGS IN 24 HOUR BLOCK ARE EXTRAPOLATED
C
157   CONTINUE
      IF(J.LT.JHOURS)GO TO 257
C
```

```
C       THE FOLLOWING SECTION COMPUTES THE SPEED FROM BINARY NUMBERS
C
        DO 100 I=1,120,4
        IF(J.EQ.1.AND.I.EQ.1)DATA(I-4)=0
        IF(J.NE.1.AND.I.EQ.1)DATA(I-4)=MDATA
        IF(DATA(I-4).EQ.999.AND.DATA(I).NE.999)DATA(I-4)=0
        IDIF=(DATA(I)-DATA(I-4))
        IF(IDIF.LT.0)GO TO 101
        IF(IDIF.GT.230)GO TO 102
C
C       IF THE CHANGE IN BINARY NUMBERS IS POSITIVE AND IS GREATER
C       THAN 90% OF THE SCALE THE PROGRAM ASSUMES A CURRENT REVERSAL.
C
        GO TO 103
102     FDATA(I)=257.2/256.*(IDIF-255)*(749./COUNT)
C
C       THESE CORRECT FOR 2 TYPES OF IMPELLERS USED
C
        IF(IYEAR.LT.80)FDATA(I)=FDATA(I)+0.01*FDATA(I)
        IF(IYEAR.GE.80)FDATA(I)=FDATA(I)-0.15*FDATA(I)
        IF(DATA(I).EQ.999)FDATA(I)=999.9
        GO TO 100
101     IF(IDIF.GE.(-26))GO TO 103
C
C       IF THE CHANGE IN BINARY NUMBERS IS NEGATIVE AND LESS THAN
C       10% OF THE SCALE THE PROGRAM SENSES A CURRENT REVERSAL.  IF
C       THE NUMBER IS NEGATIVE BUT MORE THAN 10% OF THE SCALE THE
C       PROGRAM ASSUMES A ROLLOVER (SCALE REPEATS).
C
        FDATA(I)=257.2/256.*(IDIF+255)*(749./COUNT)
        IF(IYEAR.LT.80)FDATA(I)=FDATA(I)+0.01*FDATA(I)
        IF(IYEAR.GE.80)FDATA(I)=FDATA(I)-0.15*FDATA(I)
        IF(DATA(I).EQ.999)FDATA(I)=999.9
        GO TO 100
103     FDATA(I)=257.2/256.*IDIF*(749./COUNT)
        IF(IYEAR.LT.80)FDATA(I)=FDATA(I)+0.01*FDATA(I)
        IF(IYEAR.GE.80)FDATA(I)=FDATA(I)-0.15*FDATA(I)
        IF(DATA(I).EQ.999)FDATA(I)=999.9
100     CONTINUE
        MDATA=DATA(IDATA(2)-3)
C
C       THE FOLLOWING SECTION COMPUTES THE DIRECTION FROM BINARY NUMBERS
C
        DO 200 I=1,120,4
        FDATA(I+1)=DATA(I+1)*1.40625
        IF(FDATA(I+1).GT.352.5.AND.FDATA(I+1).LT.7.5)GO TO 203
        ANGLE=22.5
        DO 201 K=2,24
        IF(FDATA(I+1).LT.ANGLE)DDEV=-DEV(K)
        IF(FDATA(I+1).LT.ANGLE)GO TO 202
        ANGLE=ANGLE+15
201     CONTINUE
203     DDEV=-DEV(1)
202     FDATA(I+1)=FDATA(I+1)+VAR+DDEV
        IF(FDATA(I+1).GT.360.)FDATA(I+1)=FDATA(I+1)-360.
        IF(DATA(I+1).EQ.999)FDATA(I+1)=999.9
200     CONTINUE
C
C        THE FOLLOWING COMPUTES THE CURVE FIT FOR TEMPERATURE
C
```

```
      SBIN=BIN(1)+BIN(2)+BIN(3)
      SBATH=BATH(1)+BATH(2)+BATH(3)
      SBNBTH=BIN(1)*BATH(1)+BIN(2)*BATH(2)+BIN(3)*BATH(3)
      SBIN2=BIN(1)*BIN(1)+BIN(2)*BIN(2)+BIN(3)*BIN(3)
      B=(SBNBTH-(SBIN*SBATH/3.))/(SBIN2-(SBIN*SBIN/3.))
      A=(SBATH/3.)-(B*SBIN/3.)
C     THE FOLLOWING SECTION COMPUTES THE TEMPERATURE FROM BINARY NUMBERS
C     (RANGE LIMITED TO 5 DEGREES TO 30 DEGREES CENTIGRADE)
C
      DO 300 I=1,120,4
      IF(DATA(I+2).LT.153)DATA(I+2)=DATA(I+2)+256
      FDATA(I+2)=A+B*DATA(I+2)
      IF(DATA(I+2).EQ.999)FDATA(I+2)=999.9
300   CONTINUE
C
C     THE FOLLOWING COMPUTES THE CURVE FIT FOR CONDUCTIVITY
C
      SUMBIN=ALCON + HCON
      SUMCON=TSTCON(1)+TSTCON(2)
      SUMBC=ALCON*TSTCON(1)+HCON*TSTCON(2)
      SMBIN2=ALCON*ALCON+HCON*HCON
      BB=(SUMBC-(SUMBIN*SUMCON/2.))/(SMBIN2-(SUMBIN*SUMBIN/2.))
      AA=(SUMCON/2.)-(BB*SUMBIN/2.)
C
C     THE FOLLOWING SECTION COMPUTES THE COND FROM BINARY NUMBERS
C     RANGE LIMITED TO 25MMHOS/CM
C
      DO 400 I=1,120,4
      CCON=-10.24*COND+562.2
      IF(DATA(I+3).LE.CCON)DATA(I+3)=DATA(I+3)+256.
      FDATA(I+3)=AA+BB*DATA(I+3)
      IF(DATA(I+3).EQ.999)FDATA(I+3)=999.9
400   CONTINUE
      CALL SALIN
      DO 256 I=1,120,4
3011  FORMAT(1X,I3,1X,2I2,5F10.1)
      WRITE(2,3011)IDATE,IHR,MIN,FDATA(I),FDATA(I+1),
     *FDATA(I+2),FDATA(I+3),SAL(I)
      MIN=MIN+2
      IDATE=MOD(IDATE-1+(60*IHR+MIN)/1440,NM)+1
      IHR=MOD(IHR+MIN/60,24)
      MIN=MOD(MIN,60)

256   CONTINUE
      GO TO 250
257   IHR=IHR+1
      IDATE=MOD(IDATE-1+(60*IHR+MIN)/1440,NM)+1
      IHR=MOD(IHR+MIN/60,24)
      MIN=MOD(MIN,60)
250   CONTINUE
      END
C
C
      SUBROUTINE SALIN
C
C     THIS SUBROUTINE IS BASED ON THE NOTES OF
C     D. W. PRITCHARD (APRIL, 1978)
C
      COMMON FDATA(120),IDATA(2),SAL(120)
      DO 25 I=1,120,4
```

```
      TF=FDATA(I+2)-15.0
      CKT=29.0473+0.86062*FDATA(I+2)+4.6704E-03*FDATA(I+2)**2-
     12.5813E-05*FDATA(I+2)**3-2.217E-08*FDATA(I+2)**4
      FT=-1.024*TF-2.843E-02*TF**2+4.35E-05*TF**3+1.481E-04*TF**4-
     13.0E-06*TF**5
      RT=FDATA(I+3)/CKT
      RD=RT-1.0
      BR=67.1-8.52*RD+3.09*RD**2-26.9*RD**3
      FRT=-0.25*RD*TF
      RF=RT+RT*RD*TF*(BR+FT+FRT)*1.0E-05
      SAL(I)=-0.08996+28.29720*RF+12.80832*RF**2-10.67869*RF**3+
     15.98624*RF**4-1.32311*RF**5
      IF(SAL(I).LE.0.)SAL(I)=.10
      IF(SAL(I).GT.0..AND.SAL(I).LE.44.)SAL(I)=.99667*SAL(I)+.1
      IF(SAL(I).GT.44.)SAL(I)=999.9
      IF(FDATA(I+3).EQ.999.9)SAL(I)=999.9
25    CONTINUE
      RETURN
      END
```

```
C      PROGRAM STEP4A
C      THIS PROGRAM AVERAGES THE ENGINEERING UNIT DATA
C
       LOGICAL PRNT
       DIMENSION SUME(6),SUMN(6),FDATA(128),ADATA(2,12),RSUM(6),
      *SUMPHI(6),SAL(120),ASAL(6),LINE(32),LLINE(32)
       INTEGER DAY,DDAY,HOUR,TAPE(4)
       TYPE 0010
0010   FORMAT(' WHEN * APPEARS, TYPE MT: AND 6 DIGIT INPUT
      *DATA FILE NAME.',/)
       CALL ASSIGN(1,IFILE,-10)
       CALL ASSIGN(3,'PLOT.DAT')
       TYPE 0001
0001   FORMAT(' WHAT IS THE AVERAGING INTERVAL DESIRED? (I3)')
       ACCEPT 0002,INT
0002   FORMAT(I3)
       TYPE 0008
0008   FORMAT(' WHAT IS THE TOTAL NUMBER OF HOURS OF DATA? (I3)')
       ACCEPT 0009,IHRS
0009   FORMAT(I3)
       TYPE 0011
0011   FORMAT(' DO YOU DESIRE A HARDCOPY OF THE DATA? (T OR F)')
       ACCEPT 1011,PRNT
1011   FORMAT(L1)
       READ(1,1013)(LINE(I),I=1,32)
1013   FORMAT(9X,32A2)
1000   FORMAT(80A2)
       READ(1,1000)(LLINE(I),I=1,32)
       READ(1,1001)TAPE
1001   FORMAT(15X,4A2)
       READ(1,1002)ISN
1002   FORMAT(15X,I4)
       READ(1,1003)LAT,ALAT,LONG,ALONG
1003   FORMAT(15X,I3,F5.1,2X,I3,F5.1)
       READ(1,1004)UTMN,UTME
1004   FORMAT(15X,F8.1,2X,F8.1)
       READ(1,1002)IHITE
       READ(1,1002)IDEPTH
       READ(1,1005)MON,DAY,IYR,HOUR
1005   FORMAT(15X,I2,1X,I2,1X,I2,2X,I4)
       READ(1,1006)MMON,DDAY,IIYR
1006   FORMAT(15X,I2,1X,I2,1X,I2)
       READ(1,1002)
       WRITE(3,0007)(LINE(I),I=1,32),(LLINE(I),I=1,32),
      *TAPE,ISN,LAT,ALAT,LONG,ALONG,
      1UTMN,UTME,IHITE,IDEPTH,MON,DAY,IYR,HOUR,MMON,DDAY,IIYR,INT
       IF(PRNT)WRITE(6,0007)(LINE(I),I=1,32),(LLINE(I),I=1,32),
      *TAPE,ISN,LAT,ALAT,LONG,ALONG,
      1UTMN,UTME,IHITE,IDEPTH,MON,DAY,IYR,HOUR,MMON,DDAY,IIYR,INT
0007   FORMAT(' REMARKS:  ',32A2,/,
      *1X,32A2,/,
      1' TAPE NUMBER:   ',4A2,/,
      2' METER NUMBER:  ',I4,/,
      3' POSITION:      ',I3,F5.1,'N',1X,I3,F5.1,'W',/,
      4' UTM POSITION:  ',F8.1,'N',1X,F8.1,'E',/,
      5' METER HEIGHT:  ',I4,' FEET',/,
      6' WATER DEPTH:   ',I4,' FEET',/,
      7' STARTING DATE: ',I2,2('-',I2),I6,1X,'PST',/,
      8' RECOVERY DATE: ',I2,2('-',I2),/,
      9' DATA INTERVAL: ',I4,' MIN')
```

```
      DO 900 N=1,IHRS
      NN=N
      IF(PRNT)GO TO 850
      GO TO 875
850   IF(MOD(NN,24).EQ.1)WRITE(6,0004)
875   DO 100 J=1,30
      READ(1,0003)IDATE,IHR,IMIN,FDATA(4*J-3),FDATA(4*J-2),
     *FDATA(4*J-1),FDATA(4*J),SAL(4*J-3)
0003  FORMAT(1X,I3,1X,2I2,5F10.1)
      IF(J.EQ.1)JDATE=IDATE
      IF(J.EQ.1)JHR=IHR
      IF(J.EQ.1)JMIN=IMIN
100   CONTINUE
      JJ=60/INT
      DIV=INT/2
      KK=2*INT
      LL=1
      K=KK
      DO 200 II=1,JJ
      RSUM(II)=0
      SUMPHI(II)=0
      SUME(II)=0
      SUMN(II)=0
      DO 300 I=LL,KK,4
      RPHI=FDATA(I+1)/57.29578
      SPDN=(COS(RPHI))*FDATA(I)
      SPDE=(SIN(RPHI))*FDATA(I)
      SUME(II)=SUME(II)+SPDE
      SUMN(II)=SUMN(II)+SPDN
      IF(FDATA(I+1).EQ.999.9)BAD=999.9
300   CONTINUE
      RSUM(II)=SQRT(SUME(II)*SUME(II)+SUMN(II)*SUMN(II))/DIV
      IF(ABS(SUMN(II)).LE.1.E-7)GO TO 700
      SUMPHI(II)=(ATAN(SUME(II)/SUMN(II)))*57.29578
      IF(SUMPHI(II).LE.0.)GO TO 750
      IF(SUMN(II).LT.0.)SUMPHI(II)=180.+SUMPHI(II)
      GO TO 775
750   IF(SUMN(II).GT.0..AND.SUME(II).LE.0.)SUMPHI(II)=360.+SUMPHI(II)
      IF(SUMN(II).LT.0..AND.SUME(II).GE.0.)SUMPHI(II)=180.+SUMPHI(II)
      IF(BAD.EQ.999.9)SUMPHI(II)=999.9
      BAD=0
      GO TO 775
700   IF(SUME(II).LT.0.)SUMPHI(II)=270.
      IF(SUME(II).GT.0.)SUMPHI(II)=90.
775   CONTINUE
      SSAL=0
      DO 400 I=LL,KK,4
      SSAL=SSAL+SAL(I)
400   CONTINUE
      ASAL(II)=SSAL/DIV
      IF(ASAL(II).GT.36)ASAL(II)=999.9
      SDATA=0
      DO 500 I=LL,KK,4
      SDATA=SDATA+FDATA(I+2)
500   CONTINUE
      ADATA(3,II)=SDATA/DIV
      SDATA=0
      DO 600 I=LL,KK,4
      SDATA=SDATA+FDATA(I+3)
600   CONTINUE
```

```
      ADATA(4,II)=SDATA/DIV
      IF(ADATA(4,II).GT.60)ADATA(4,II)=999.9
      LL=KK+1
      KK=K*(II+1)
200   CONTINUE
      JMIN=JMIN+(INT/2)-1
      JDATE=MOD(JDATE-1+(60*JHR+JMIN)/1440,365)+1
      JHR=MOD(JHR+JMIN/60,24)
      JMIN=MOD(JMIN,60)
0004  FORMAT(//,2X,'DATE',2X,'TIME',08X,'SPD',09X,'DIR',09X,
     *'TMP',09X,'COND',9X,'SAL',/,18X,'CM/SEC',7X,'DEG T',
     *7X,'DEG C',6X,'MMHO/CM',7X,'0/00')
      DO 800 II=1,JJ
      IF(PRNT)WRITE(6,0006)JDATE,JHR,JMIN
     *,RSUM(II),SUMPHI(II),ADATA(3,II),ADATA(4,II),ASAL(II)
      AMIN=JMIN
      AHR=JHR
      DATE=JDATE+AHR/24.+AMIN/1440.
      SPD=RSUM(II)
      DIR=SUMPHI(II)
      TMP=ADATA(3,II)
      CON=ADATA(4,II)
      SSAL=ASAL(II)
      WRITE(3,0005)DATE,SPD,DIR,TMP,CON,SSAL
0005  FORMAT(F10.2,5F10.1)
      JMIN=JMIN+INT
      JDATE=MOD(JDATE-1+(60*JHR+JMIN)/1440,365)+1
      JHR=MOD(JHR+JMIN/60,24)
      JMIN=MOD(JMIN,60)
800   CONTINUE
0006  FORMAT(2X,I3,3X,2I2,5F12.1)
900   CONTINUE
      STOP
      END
```

```
C      PROGRAM STEP5
C      PLOTTING ROUTINE: PARAMETER VS TIME (1 WEEK/PLOT)
C
       INTEGER TAPE(4)
       DIMENSION IADE(16),JADE(19),KADE(15),LADE(12),MADE(17)
       DIMENSION DATE(336),SPD(336),DIR(336),TMP(336),CON(336),
      1SSAL(336),POSIT(13)
       CALL ASSIGN(3,'PLOT.DAT')
       COMMON/ADEO/IXY
       COMMON/PLOT/XMIN,XMAX,YMIN,YMAX,MINX,MAXX,MINY,MAXY,
      *IMIN,IMAX,JMIN,JMAX
       COMMON/TEKLET/NCHAR,XSIZE,YSIZE,SLOPE
       DATA IADE/74,85,76,73,65,
      *78,32,68,65,89,44,32,49,57,56,48/
       DATA JADE/84,69,77,80,69,82,65,84,85,82,69,32,40,
      *68,69,71,32,67,41/
       DATA KADE/83,65,76,73,78,73,84,89,32,40,48,47,48,48,41/
       DATA LADE/83,80,69,69,68,32,40,67,77,47,83,41/
       DATA MADE/68,73,82,69,67,84,73,79,78,32,40,68,69,71,32,84,41/
       TYPE 0001
0001   FORMAT('  HOW MANY WEEKS OF DATA TO PLOT?')
       ACCEPT 1001,IWEEKS
1001   FORMAT(I2)
       TYPE 0002
0002   FORMAT('  HOW MANY AVERAGES ARE THERE EVERY HOUR? (I2) ')
       ACCEPT 1002,ITIMES
       INT=60/ITIMES
1002   FORMAT(I2)
       TYPE 0005
0005   FORMAT('  TYPE POSITION DESCRIPTION (UP TO 26 CHARACTERS).')
       ACCEPT 1005,(POSIT(NN),NN=1,13)
1005   FORMAT(13A2)
       TYPE 0003
0003   FORMAT(' DO YOU WANT TO USE THE X/Y PLOTTER (YES=1; NO=0)')
       ACCEPT 1003,IXY
1003   FORMAT(I1)
       READ(3,1001)
       READ(3,1001)
       READ(3,0004)TAPE
0004   FORMAT(15X,4A2)
       READ(3,1001)
       READ(3,0006)LAT,ALAT,LONG,ALONG
0006   FORMAT(15X,I3,F5.1,2X,I3,F5.1)
       READ(3,1001)
       READ(3,0007)IHITE
0007   FORMAT(15X,I4)
       READ(3,1001)
       READ(3,0008)IYR
0008   FORMAT(21X,I2)
       READ(3,1001)
       READ(3,1001)
4000   FORMAT(07X,'CURRENT METER OBSERVATIONS (',I2,1X,
      1'MINUTE AVERAGES)')
4001   FORMAT(07X,13A2,1X,'(',,I2,'-',F4.1,'N',1X,I3,'-',F4.1,'W)')
4002   FORMAT(07X,'METER',1X,I3,1X,'FEET ABOVE BED.  TAPE
      1NUMBER ',4A2,'.')
       IPTS=168*ITIMES
       DO 10 J=1,IWEEKS
100    FORMAT(F10.2,5F10.1)
       JJP=0
```

```
      DO 4501 JP=1,IPTS
      READ(3,100,END=4500)DATE(JP),SPD(JP),DIR(JP),TMP(JP),CON(JP)
     1,SSAL(JP)
      JJP=JP
4501  CONTINUE
4500  IIDATE=DATE(1)
      XMIN=IIDATE
      XMAX=IIDATE+8
      YMIN=0.0
      YMAX=160.
      MINX=80
      MAXX=1000
      MINY=140
      MAXY=425
      PAUSE 'PAGE SCREEN AND <CR>'
      CALL INITT(120)
      CALL GRID(8,12,8,4,1.0,1)
      CALL TINPUT(LA)
      CALL PLOT(DATE,SPD,JJP,0)
      CALL TINPUT(LA)
      CALL ASCALE(1,8,0)
      CALL ASCALE(3,8,0)
      NCHAR=16
      IADE(15)=IYR/10+48
      IADE(16)=IYR-10*(IYR/10)+48
      CALL LABEL(IADE,1)
      NCHAR=12
      CALL LABEL(LADE,3)
      CALL RROTAT(0.)
      YMIN=0.0
      YMAX=360.0
      MINY=485
      MAXY=770
      CALL TINPUT(LA)
      CALL GRID(8,12,4,9,1.0,1)
      CALL TINPUT(LA)
      CALL PLOT(DATE,DIR,JJP,0)
      CALL TINPUT(LA)
      CALL ASCALE(1,8,0)
      CALL ASCALE(3,4,0)
      NCHAR=17
      CALL LABEL(MADE,3)
      CALL FINITT(0,70)
      WRITE (7,4000)INT
      WRITE(7,4001)(POSIT(NN),NN=1,13),LAT,ALAT,LONG,ALONG
      WRITE(7,4002)IHITE,TAPE
      READ(5,999)JI
999   FORMAT(I5)
      YMIN=5.
      YMAX=25.
      MINY=140
      MAXY=425
      PAUSE 'PAGE SCREEN AND <CR>'
      CALL INITT(120)
      CALL GRID(8,12,4,5,1.0,1)
      CALL TINPUT(LA)
      CALL PLOT(DATE,TMP,JJP,0)
      CALL TINPUT(LA)
      CALL ASCALE(1,8,0)
      CALL ASCALE(3,4,0)
```

```
      NCHAR=16
      CALL LABEL(IADE,1)
      NCHAR=19
      CALL LABEL(JADE,3)
      CALL RROTAT(0.)
      YMIN=0.
      YMAX=35.
      MINY=485
      MAXY=770
      CALL TINPUT(LA)
      CALL GRID(8,12,7,5,1.0,1)
      CALL TINPUT(LA)
      CALL PLOT(DATE,SSAL,JJP,0)
      CALL TINPUT(LA)
      CALL ASCALE(1,8,0)
      CALL ASCALE(3,7,0)
      NCHAR=15
      CALL LABEL(KADE,3)
      CALL FINITT(0,70)
      WRITE(7,4000)INT
      WRITE(7,4001)(POSIT(NN),NN=1,13),LAT,ALAT,LONG,ALONG
      WRITE(7,4002)IHITE,TAPE
      READ(5,999)JI
10    CONTINUE
      STOP
      END
```

```
C      PROGRAM STEP5A
C      PLOTTING ROUTINE: PARAMETER VS TIME (ENTIRE TAPE)
C
       DIMENSION IADE(16),JADE(19),KADE(15),LADE(12),MADE(17)
       DIMENSION DATE(336),Y(336),POSIT(13)
       CALL ASSIGN(3,'PLOT.DAT')
       COMMON/ADEO/IXY
       COMMON/PLOT/XMIN,XMAX,YMIN,YMAX,MINX,MAXX,MINY,MAXY,
      *IMIN,IMAX,JMIN,JMAX
       COMMON/TEKLET/NCHAR,XSIZE,YSIZE,SLOPE
       INTEGER TAPE(4)
       DATA IADE/74,85,76,73,65,
      *78,32,68,65,89,44,32,49,57,56,48/
       DATA JADE/84,69,77,80,69,82,65,84,85,82,69,32,40,
      *68,69,71,32,67,41/
       DATA KADE/83,65,76,73,78,73,84,89,32,40,48,47,48,48,41/
       DATA LADE/83,80,69,69,68,32,40,67,77,47,83,41/
       DATA MADE/68,73,82,69,67,84,73,79,78,32,40,68,69,71,32,84,41/
       TYPE 0001
0001   FORMAT('  HOW MANY WEEKS OF DATA TO PLOT?')
       ACCEPT 1001,IWEEKS
1001   FORMAT(I2)
       TYPE 0002
0002   FORMAT('  HOW MANY AVERAGES ARE THERE EVERY HOUR? (I2) ')
       ACCEPT 1002,ITIMES
       INT=60/ITIMES
1002   FORMAT(I2)
       TYPE 0003
0003   FORMAT('  TYPE POSITION DESCRIPTION (UP TO 26 CHARACTERS).')
       ACCEPT 1003,(POSIT(NN),NN=1,13)
1003   FORMAT(13A2)
       TYPE 0004
0004   FORMAT('  DO YOU WANT TO USE THE X/Y PLOTTER? (YES=1; NO=0)')
       ACCEPT 1004,IXY
1004   FORMAT(I1)
       READ(3,1001)
       READ(3,1001)
       READ(3,0005)TAPE
0005   FORMAT(15X,4A2)
       READ(3,1001)
       READ(3,0006)LAT,ALAT,LONG,ALONG
0006   FORMAT(15X,I3,F5.1,2X,I3,F5.1)
       READ(3,1001)
       READ(3,0007)IHITE
0007   FORMAT(15X,I4)
       READ(3,1001)
       READ(3,0008)IYR
0008   FORMAT(21X,I2)
       READ(3,1001)
       READ(3,1001)
4000   FORMAT(07X,'CURRENT METER OBSERVATIONS (',I2,1X,
      1'MINUTE AVERAGES)')
4001   FORMAT(07X,13A2,1X,'(',,I2,'-',F4.1,'N',1X,I3,'-',F4.1,'W)')
4002   FORMAT(07X,'METER',1X,I3,1X,'FEET ABOVE BED.  TAPE
      1NUMBER ',4A2,'.')
       IPTS=168*ITIMES
       JJP=0
       ND=IWEEKS*7
       XMIN=0
       XMAX=XMIN+ND
```

```
      YMIN=0.0
      YMAX=160.
      MINX=80
      MAXX=1000
      MINY=140
      MAXY=425
      PAUSE 'PAGE SCREEN AND <CR>'
      CALL INITT(120)
      CALL GRID(1,ND,8,4,1.0,1)
      DO 10 J=1,IWEEKS
      DO 15 JP=1,IPTS
      READ(3,100,END=4400)DATE(JP),Y(JP)
      JJP=JP
15    CONTINUE
100   FORMAT(F10.2,F10.1)
4400  IF(J.NE.1)GO TO 4500
      IIDATE=DATE(1)
      XMIN=IIDATE
      XMAX=IIDATE+ND
      CALL TINPUT(LA)
4500  CALL PLOT(DATE,Y,JJP,0)
10    CONTINUE
      CALL TINPUT(LA)
      CALL ASCALE(1,ND/7,0)
      CALL ASCALE(3,8,0)
      NCHAR=16
      IADE(15)=IYR/10+48
      IADE(16)=IYR-10*(IYR/10)+48
      CALL LABEL(IADE,1)
      NCHAR=12
      CALL LABEL(LADE,3)
      REWIND 3
C
C     THE DO LOOPS FOLLOWING REWINDS ARE TO REREAD PAST
C     THE HEADER TO THE DATA PORTION OF FILE.
C
      DO 20 K=1,11
      READ(3,1001)
20    CONTINUE
      CALL RROTAT(0.)
      YMIN=0.0
      YMAX=360.0
      MINY=485
      MAXY=770
      CALL TINPUT(LA)
      CALL GRID(1,ND,4,9,1.0,1)
      JJP=0
      DO 11 J=1,IWEEKS
      DO 16 JP=1,IPTS
      READ(3,101,END=4501)DATE(JP),Y(JP)
      JJP=JP
16    CONTINUE
101   FORMAT(F10.2,10X,F10.1)
      CALL TINPUT(LA)
4501  CALL PLOT(DATE,Y,JJP,0)
11    CONTINUE
      CALL TINPUT(LA)
      CALL ASCALE(1,ND/7,0)
      CALL ASCALE(3,4,0)
      NCHAR=17
```

```
      CALL LABEL(MADE,3)
      REWIND 3
      DO 21 K=1,11
      READ(3,1001)
21    CONTINUE
      CALL TINPUT(LA)
      CALL FINITT(0,70)
      WRITE (7,4000)INT
      WRITE(7,4001)(POSIT(NN),NN=1,13),LAT,ALAT,LONG,ALONG
      WRITE(7,4002)IHITE,TAPE
      READ(5,999)JI
999   FORMAT(I5)
      YMIN=5.
      YMAX=25.
      MINY=140
      MAXY=425
      PAUSE 'PAGE SCREEN AND <CR>'
      CALL INITT(120)
      CALL GRID(1,ND,4,5,1.0,1)
      JJP=0
      DO 12 J=1,IWEEKS
      DO 17 JP=1,IPTS
      READ(3,102,END=4502)DATE(JP),Y(JP)
      JJP=JP
17    CONTINUE
102   FORMAT(F10.2,20X,F10.1)
      CALL TINPUT(LA)
4502  CALL PLOT(DATE,Y,JJP,0)
12    CONTINUE
      CALL TINPUT(LA)
      CALL ASCALE(1,ND/7,0)
      CALL ASCALE(3,4,0)
      NCHAR=16
      CALL LABEL(IADE,1)
      NCHAR=19
      CALL LABEL(JADE,3)
      REWIND 3
      DO 22 K=1,11
      READ(3,1001)
22    CONTINUE
      CALL RROTAT(0.)
      YMIN=0.
      YMAX=35.
      MINY=485
      MAXY=770
      CALL TINPUT(LA)
      CALL GRID(1,ND,7,5,1.0,1)
      JJP=0
      DO 13 J=1,IWEEKS
      DO 18 JP=1,IPTS
      READ(3,103,END=4503)DATE(JP),Y(JP)
      JJP=JP
18    CONTINUE
103   FORMAT(F10.2,40X,F10.1)
      CALL TINPUT(LA)
4503  CALL PLOT(DATE,Y,JJP,0)
13    CONTINUE
      CALL TINPUT(LA)
      CALL ASCALE(1,ND/7,0)
      CALL ASCALE(3,7,0)
```

```
NCHAR=15
CALL LABEL(KADE,3)
REWIND 3
CALL FINITT(0,70)
WRITE(7,4000)INT
WRITE(7,4001)(POSIT(NN),NN=1,13),LAT,ALAT,LONG,ALONG
WRITE(7,4002)IHITE,TAPE
READ(5,999)JI
STOP
END
```

```
      PROGRAM STEP6
C     PLOTTING ROUTINE: TIDAL ELLIPSE
C
      COMMON /PLOT/ XMIN, XMAX, YMIN, YMAX, MINX, MAXX, MINY, MAXY,
     *              IMIN, IMAX, JMIN, JMAX
      COMMON /TEKLET/ NCHAR, XSIZE, YSIZE, SLOPE
      COMMON/ADEO/IXY
      DIMENSION U(340), V(340)
      DIMENSION IADE(6), POSIT(13)
      DATA IADE/85, 40, 77, 47, 83, 41/
      DATA MINX, MAXX, MINY, MAXY/220, 800, 160, 740/
      INTEGER DAY, TAPE(4)
      TYPE 602
602   FORMAT('  TYPE POSITION DESCRIPTION (UP TO 26 CHARACTERS)')
      ACCEPT 603, (POSIT(NN), NN=1, 13)
603   FORMAT(13A2)
      TYPE 600
600   FORMAT(' HOW MANY AVERAGES ARE THERE EVERY HOUR? (I2)')
      ACCEPT 601, ITIMES
601   FORMAT(I2)
      TYPE 604
604   FORMAT(' HOW MANY TOTAL WEEKS OF DATA? (I2)')
      ACCEPT 605, IWEEK
605   FORMAT(I2)
      TYPE 606
606   FORMAT(' DO YOU WANT TO USE THE X/Y PLOTTER (YES=1; NO=0)')
      ACCEPT 607, IXY
607   FORMAT(I1)
      CALL ASSIGN (1, 'PLOT.DAT')
      READ(1, 601)
      READ(1, 601)
      READ(1, 700)TAPE
700   FORMAT(15X, 4A2)
      READ(1, 601)
      READ(1, 701)LAT, ALAT, LONG, ALONG
701   FORMAT(15X, I3, F5.1, 2X, I3, F5.1)
      READ(1, 601)
      READ(1, 702)IHITE
702   FORMAT(15X, I4)
      READ(1, 601)
      READ(1, 703)MON, DAY, IYR
703   FORMAT(15X, I2, 1X, I2, 1X, I2)
      READ(1, 601)
      READ(1, 601)
C     READ DATA POINTS
      DO 13 K=1, IWEEK
      JT=0
      M=ITIMES*168
      DO 72 L=1, M
      READ (1, 1000, END=81) TET, SPEED, DIR
1000  FORMAT(F10.2, 2F10.1)
      JT=JT+1
      SPEED=SPEED/100.
      U(JT)=SPEED*SIN(DIR/57.3)
      V(JT)=SPEED*COS(DIR/57.3)
72    CONTINUE
      GO TO 35
81    JT=JT+1
      SPEED=SPEED/100.
      U(JT)=SPEED*SIN(DIR/57.3)
```

```
      V(JT)=SPEED*COS(DIR/57.3)
85    CONTINUE
C     SET VIRTUAL WINDOW
      IDX=0
      XMAX=0.
      YMAX=0.
      IDX=IDX+1
      DO 110 J=1,JT
      YMAX=ABS(U(J))
      IF(XMAX.LT.YMAX) XMAX=YMAX
      YMAX=ABS(V(J))
110   IF(XMAX.LT.YMAX) XMAX=YMAX
      IF(XMAX.LT..20) GO TO 125
      I=2.*XMAX+1.
      XMAX=FLOAT(I)*.5
      GO TO 130
125   XMAX=.20
130   XMIN=-XMAX
      YMAX=XMAX
      YMIN=XMIN
      PAUSE 'PAGE SCREEN AND TYPE <CR>'
      CALL INITT(120)
      CALL GRID(2,5,2,5,1.,1)
      CALL TINPUT(LA)
      CALL PLOT(U,V,JT,0)
      SLOPE=0.
      CALL TINPUT(LA)
      CALL ASCALE(1,2,1)
      CALL ASCALE(3,2,1)
      NCHAR=6
      IADE(1)=85
      CALL LABEL(IADE,1)
      NCHAR=6
      IADE(1)=86
      CALL LABEL(IADE,3)
      CALL FINITT(0,70)
      WRITE(7,1009)MON,DAY,IYR,K
1009  FORMAT(16X,'TIDAL ELLIPSE',6X,I2,'-',I2,'-',I2,
     *6X,'WEEK NUMBER',1X,I1)
      WRITE(7,1010)(POSIT(NN),NN=1,13),LAT,ALAT,LONG,ALONG
1010  FORMAT(16X,13A2,2X,I2,'-',F4.1,'N',1X,I3,'-',F4.1,'W')
      WRITE(7,1011)IHITE,TAPE
1011  FORMAT(16X,'METER',1X,I3,1X,'FEET ABOVE BED.  TAPE
     1 NUMBER',1X,4A2,'.')
      READ(5,605) J
      TYPE 500
500   FORMAT('  ENTER BEGINNING DATE FOR NEXT WEEK
     * OF DATA (3I2)')
      ACCEPT 510,MON,DAY,IYR
510   FORMAT(3I2)
10    CONTINUE
      STOP
      END
```

```
C       PROGRAM STEP7
C       PLOTTING ROUTINE: PROGRESSIVE VECTOR DIAGRAM (1 WEEK
C       PER PLOT)
C
        DIMENSION IADE(5),POSIT(13)
        DIMENSION DATE(340),SPD(340),DIR(340),X(340),Y(340)
        CALL ASSIGN(3,'PLOT.DAT')
        COMMON/ADEO/IXY
        COMMON/PLOT/XMIN,XMAX,YMIN,YMAX,MINX,MAXX,MINY,MAXY,
       *IMIN,IMAX,JMIN,JMAX
        COMMON/TEKLET/NCHAR,XSIZE,YSIZE,SLOPE
        INTEGER TAPE(4),DAY
        DATA IADE/88,40,75,77,41/
        TYPE 0001
0001    FORMAT('  HOW MANY AVERAGES ARE THERE EACH HOUR? (I2)')
        ACCEPT 1001,ITIMES
1001    FORMAT(I2)
        INT=60/ITIMES
        TYPE 0002
0002    FORMAT('  HOW MANY TOTAL WEEKS OF DATA? (I2)')
        ACCEPT 1002,IWEEK
1002    FORMAT(I2)
        TYPE 0003
0003    FORMAT('  WHAT IS THE DESIRED N/S PLOT RANGE IN KILOMETERS?
       *(I4,1X,I4)')
        ACCEPT 1003,IYMIN,IYMAX
1003    FORMAT(I4,1X,I4)
        TYPE 0004
0004    FORMAT('  WHAT IS THE DESIRED E/W PLOT RANGE IN KILOMETERS?
       * (I4,1X,I4)')
        ACCEPT 1004,IXMIN,IXMAX
1004    FORMAT(I4,1X,I4)
        TYPE 0005
0005    FORMAT(' DO YOU WANT TO USE THE X/Y PLOTTER? (YES=1; NO=0)')
        ACCEPT 1005,IXY
1005    FORMAT(I1)
        TYPE 0007
0007    FORMAT('  TYPE POSITION DESCRIPTION (UP TO 26 CHARACTERS)')
        ACCEPT 1007,(POSIT(NN),NN=1,13)
1007    FORMAT(13A2)
        READ (3,1001)
        READ(3,1001)
        READ(3,0006)TAPE
0006    FORMAT(15X,4A2)
        READ(3,1001)
        READ(3,0008)LAT,ALAT,LONG,ALONG
0008    FORMAT(15X,I3,F5.1,2X,I3,F5.1)
        READ(3,1001)
        READ(3,0009)IHITE
0009    FORMAT(15X,I4)
        READ(3,1001)
        READ(3,0010)MON,DAY,IYR
0010    FORMAT(15X,I2,1X,I2,1X,I2)
        READ(3,1001)
        READ(3,1001)
        XMIN=IXMIN
        XMAX=IXMAX
        YMIN=IYMIN
        YMAX=IYMAX
        MINX=220
```

```
      MAXX=800
      MINY=160
      MAXY=740
      X(1)=0.0
      Y(1)=0.0
      M=ITIMES*168
      DO 13 K=1,IWEEK
      PAUSE 'PAGE SCREEN AND TYPE <CR>'
      CALL INITT(120)
      CALL GRID(2,10,2,10,1.,1)
      DO 102 J=2,M+1
100   FORMAT(F10.2,2F10.1)
101   READ(3,100,END=102)DATE(J),SPD(J),DIR(J)
      SPD(J)=SPD(J)*1.0E-5
      JJ=J
102   CONTINUE
103   DO 11 J=2,JJ
      DT=60/ITIMES*60
      X(J)=SIN(DIR(J)/57.3)*SPD(J)*DT+X(J-1)
      Y(J)=COS(DIR(J)/57.3)*SPD(J)*DT+Y(J-1)
201   IF(J.EQ.2)CALL PLOT(X,Y,1,-2)
11    CONTINUE
      CALL TINPUT(LA)
      CALL PLOT(X,Y,JJ,0)
      CALL TINPUT(LA)
14    CALL ASCALE(1,2,0)
      IMIN=IMIN-5
      CALL ASCALE(3,2,0)
      NCHAR=5
      CALL LABEL(IADE,1)
      IADE(1)=89
      NCHAR=5
      CALL LABEL(IADE,3)
      CALL FINITT(0,70)
      WRITE(7,1009)MON,DAY,IYR,K
1009  FORMAT(16X,'PROGRESSIVE VECTOR DIAGRAM',1X,I2,'-',I2,'-',I2,
     *1X,'WEEK NO.',1X,I1)
      WRITE(7,1010)(POSIT(NN),NN=1,13),LAT,ALAT,LONG,ALONG
1010  FORMAT(16X,13A2,1X,'(',I2,'-',F4.1,'N',1X,I3,'-',F4.1,'W)')
      WRITE(7,1011)IHITE,TAPE
1011  FORMAT(16X,'METER',1X,I3,1X,'FEET ABOVE BED.  TAPE
     * NUMBER ',4A2,'.')
      READ(5,999)JI
999   FORMAT(I5)
      TYPE 500
500   FORMAT('  ENTER BEGINNING DATE FOR NEXT WEEK OF DATA(3I2)')
      ACCEPT 501,MON,DAY,IYR
501   FORMAT(3I2)
13    CONTINUE
      STOP
      END
```

```
C       PROGRAM STEP7A
C       PLOTTING ROUTINE: PROGRESSIVE VECTOR DIAGRAM (FULL TAPE)
        DIMENSION IADE(5),POSIT(13)
        DIMENSION DATE(340),SPD(340),DIR(340),X(340),Y(340)
        CALL ASSIGN(3,'PLOT.DAT')
        COMMON/ADEO/IXY
        COMMON/PLOT/XMIN,XMAX,YMIN,YMAX,MINX,MAXX,MINY,MAXY,
       *IMIN,IMAX,JMIN,JMAX
        COMMON/TEKLET/NCHAR,XSIZE,YSIZE,SLOPE
        INTEGER TAPE(4),DAY,DDAY
        DATA IADE/88,40,75,77,41/
        TYPE 0001
0001    FORMAT('  HOW MANY AVERAGES ARE THERE EACH HOUR? (I2)')
        ACCEPT 1001,ITIMES
1001    FORMAT(I2)
        INT=60/ITIMES
        TYPE 0002
0002    FORMAT('  HOW MANY TOTAL WEEKS OF DATA? (I2)')
        ACCEPT 1002,IWEEK
1002    FORMAT(I2)
        TYPE 0003
0003    FORMAT('  WHAT IS THE DESIRED N/S PLOT RANGE IN KILOMETERS?
       *(I4,1X,I4)')
        ACCEPT 1003,IYMIN,IYMAX
1003    FORMAT(I4,1X,I4)
        TYPE 0004
0004    FORMAT('  WHAT IS THE DESIRED E/W PLOT RANGE IN KILOMETERS?
       * (I4,1X,I4)')
        ACCEPT 1004,IXMIN,IXMAX
1004    FORMAT(I4,1X,I4)
        TYPE 0005
0005    FORMAT(' DO YOU WANT TO USE THE X/Y PLOTTER (YES=1; NO=0)')
        ACCEPT 1005,IXY
1005    FORMAT(I1)
        TYPE 0007
0007    FORMAT('  TYPE POSITION DESCRIPTION (UP TO 26 CHARACTERS)')
        ACCEPT 1007,(POSIT(NN),NN=1,13)
1007    FORMAT(13A2)
        READ(3,1001)
        READ(3,1001)
        READ(3,0006)TAPE
0006    FORMAT(15X,4A2)
        READ(3,1001)
        READ(3,0008)LAT,ALAT,LONG,ALONG
0008    FORMAT(15X,I3,F5.1,2X,I3,F5.1)
        READ(3,1001)
        READ(3,0009)IHITE
0009    FORMAT(15X,I4)
        READ(3,1001)
        READ(3,0010)MON,DAY,IYR
0010    FORMAT(15X,I2,1X,I2,1X,I2)
        READ(3,0012)MMON,DDAY,IIYR
0012    FORMAT(15X,I2,1X,I2,1X,I2)
        READ(3,1001)
        XMIN=IXMIN
        XMAX=IXMAX
        YMIN=IYMIN
        YMAX=IYMAX
        MINX=220
        MAXX=800
```

```
      MINY=160
      MAXY=740
      X(1)=0.0
      Y(1)=0.0
      M=ITIMES*168
      PAUSE 'PAGE SCREEN AND TYPE <CR>'
      CALL INITT(120)
      CALL GRID(2,10,2,10,1.,1)
      DO 13 K=1,IWEEK
      DO 102 J=2,M+1
100   FORMAT(F10.2,2F10.1)
101   READ(3,100,END=102)DATE(J),SPD(J),DIR(J)
      SPD(J)=SPD(J)*1.0E-5
      JJ=J
102   CONTINUE
103   DO 11 J=2,JJ
      DT=60/ITIMES*60
      X(J)=SIN(DIR(J)/57.3)*SPD(J)*DT+X(J-1)
      Y(J)=COS(DIR(J)/57.3)*SPD(J)*DT+Y(J-1)
201   IF(J.EQ.2)CALL PLOT(X,Y,1,-2)
11    CONTINUE
      CALL TINPUT(LA)
      CALL PLOT(X,Y,JJ,0)
      X(1)=X(JJ)
      Y(1)=Y(JJ)
13    CONTINUE
      CALL TINPUT(LA)
14    CALL ASCALE(1,2,0)
      IMIN=IMIN-5
      CALL ASCALE(3,2,0)
      NCHAR=5
      CALL LABEL(IADE,1)
      IADE(1)=89
      NCHAR=5
      CALL LABEL(IADE,3)
      CALL FINITT(0,70)
      WRITE(7,1009)MON,DAY,IYR,MMON,DDAY,IIYR
1009  FORMAT(16X,'PROGRESSIVE VECTOR DIAGRAM',1X,I2,'-',I2,'-',I2,
     *1X,'TO',1X,I2,'-',I2,'-',I2)
      WRITE(7,1010)(POSIT(NN),NN=1,13),LAT,ALAT,LONG,ALONG
1010  FORMAT(16X,13A2,1X,'(',I2,'-',F4.1,'N',1X,I3,'-',F4.1,'W)')
      WRITE(7,1011)IHITE,TAPE
1011  FORMAT(16X,'METER',1X,I3,1X,'FEET ABOVE BED.  TAPE
     * NUMBER ',4A2,'.')
      READ(5,999)JI
999   FORMAT(I5)
      STOP
      END
```

```
C       PROGRAM STEP8
C       PLOTTING ROUTINE: HISTOGRAMS (SPEED AND DIRECTION)
C
        INTEGER DAY, DDAY, TAPE(4)
        DIMENSION DIR(73), SPD(251), POSIT(14)
        DIMENSION X(145), Y(145), U(501), V(501)
        DIMENSION IADE(20), JADE(17), KADE(12)
        COMMON/ADEO/IXY
        COMMON/PLOT/XMIN, XMAX, YMIN, YMAX, MINX, MAXX,
       *MINY, MAXY, IMIN, IMAX, JMIN, JMAX
        COMMON/TEKLET/NCHAR, XSIZE, YSIZE, SLOPE
        DATA IADE/68, 73, 82, 69, 67, 84, 73, 79, 78, 32, 40, 68, 69, 71
       *, 32, 84, 82, 85, 69, 41/
        DATA JADE/37, 32, 79, 70, 32, 79, 66, 83, 69, 82, 86, 65, 84,
       *73, 79, 78, 83/
        DATA KADE/83, 80, 69, 69, 68, 32, 40, 67, 77, 47, 83, 41/
        TYPE 0004
0004    FORMAT(' ENTER INPUT DEVICE AND FILENAME AFTER *. ', /, )
        CALL ASSIGN(3, IFILE, -10)
        TYPE 0001
0001    FORMAT('  TYPE POSITION DESCRIPTION (UP TO 26 CHARACTERS)')
        ACCEPT 1001, (POSIT(NN), NN=1, 14)
1001    FORMAT(14A2)
        TYPE 0002
0002    FORMAT('  DO YOU WANT TO USE THE X/Y PLOTTER? (YES=1; NO=0)')
        ACCEPT 1002,  IXY
1002    FORMAT (I1)
C
C       READ HEADER
C
        READ(3, 1002)
        READ(3, 1002)
2000    FORMAT(15X, 4A2)
        READ(3, 2000)TAPE
        READ(3, 1002)
        READ(3, 2001)LAT, ALAT, LONG, ALONG
2001    FORMAT(15X, I3, F5. 1, 2X, I3, F5. 1)
        READ(3, 1002)
        READ(3, 2002)IHITE
2002    FORMAT(15X, I4)
        READ(3, 1002)
        READ(3, 2003)MON, DAY, IYR
2003    FORMAT(15X, I2, 1X, I2, 1X, I2)
        READ(3, 2003)MMON, DDAY, IIYR
        READ(3, 1002)
        TOT=0
        DO 200 K=1, 72
        DIR(K)=0
200     CONTINUE
        DO 201 L=1, 250
        SPD(L)=0
201     CONTINUE
        DO 250 J=1, 28800
1003    FORMAT(1X, I3, 1X, 2I2, 2F10. 1)
        READ(3, 1003, END=251)KDATE, KHR, KMIN, SPDD, DIRR
        L=SPDD
        X=DIRR/5. 0
        IF(L. LT. 0. OR. L. GT. 250)GO TO 250
        IF(X. GT. 72)GO TO 250
        L=L+1
```

```
      K=K+1
      SPD(L)=SPD(L)+1.
      TOT=TOT+1.
      DIR(K)=DIR(K)+1.
250   CONTINUE
251   XMIN=0.0
      XMAX=360.
      YMIN=0.0
      YMAX=40.0
      MINX=220
      MAXX=800
      MINY=160
      MAXY=420
      PAUSE 'PAGE SCREEN AND TYPE <CR>'
      CALL INITT(120)
      CALL GRID(1,12,1,8,1.0,1)
      DO 253 KK=1,144,2
      K=KK/2+1
      X(KK)=5*(K-1)
      X(KK+1)=5*(K-1)+5
253   CONTINUE
      DO 255 K=1,144,2
      KK=K/2+1
      Y(K)=(DIR(KK)/TOT)*100.
      Y(K+1)=Y(K)
255   CONTINUE
      CALL TINPUT(LA)
      CALL PLOT(X,Y,144,0)
      CALL ASCALE(1,12,0)
      CALL ASCALE(3,8,0)
      NCHAR=20
      CALL LABEL(IADE,1)
      NCHAR=17
      CALL LABEL(JADE,3)
      CALL RROTAT(0.)
      MINY=480
      MAXY=740
      XMIN=0.0
      XMAX=250.
      YMAX=10.0
      CALL TINPUT(LA)
      CALL GRID(1,10,1,5,1.0,1)
      DO 254 LL=1,500,2
      L=LL/2+1
      U(LL)=L-1
      U(LL+1)=L
254   CONTINUE
      DO 256 L=1,500,2
      LL=L/2+1
      V(L)=(SPD(LL)/TOT)*100.
      V(L+1)=V(L)
256   CONTINUE
      CALL TINPUT(LA)
      CALL PLOT(U,V,500,0)
      CALL TINPUT(LA)
      CALL ASCALE(1,10,0)
      CALL ASCALE(3,5,0)
      NCHAR=12
      CALL LABEL(KADE,1)
      NCHAR=17
```

```
      CALL LABEL(JADE,3)
      CALL TINPUT(LA)
      CALL FINITT(0,70)
      WRITE(7,5000)MON,DAY,IYR,MMON,DDAY,IIYR
5000  FORMAT(17X,'STATISTICS: SPEED & DIRECTION',1X,I2,'-',
     *I2,'-',I2,1X,'TO',1X,I2,'-',I2,'-',I2)
      WRITE(7,5001)(POSIT(NN),NN=1,14),LAT,ALAT,LONG,ALONG
5001  FORMAT(17X,14A2,2X,'(',I2,'-',F4.1,'N',1X,I3,'-',
     *F4.1,'W)')
      WRITE(7,5002)IHITE,TAPE
5002  FORMAT(17X,'METER',1X,I3,1X,'FEET ABOVE BED.  TAPE
     * NUMBER ',4A2,'.')
      READ(5,999)JI
999   FORMAT(I5)
      STOP
      END
```