SSC-237

COMPUTER PROGRAMS FOR THE DIGITIZING AND USING OF LIBRARY TAPES OF SHIP STRESS AND ENVIRONMENT DATA

SHIP STRUCTURE COMMITTEE

1973

SHIP STRUCTURE COMMITTEE

AN INTERAGENCY ADVISORY
COMMITTEE DEDICATED TO IMPROVING
THE STRUCTURE OF SHIPS

MEMBER AGENCIES:

UNITED STATES COAST GUARD
NAVAL SHIP SYSTEMS COMMAND
MILITARY SEALIFT COMMAND
MARITIME ADMINISTRATION
AMERICAN BUREAU OF SHIPPING

ADDRESS CORRESPONDENCE TO:

SECRETARY
SHIP STRUCTURE COMMITTEE
U.S. COAST GUARD HEADQUARTERS
WASHINGTON, D.C. 20590

SR 187
18 JUL 1973

This report is the companion to SSC 236, a Method for Digitizing, Preparing and Using Library Tapes of Ship Stress and Environment Data, and contains the details of the conversion program which has been developed to increase the usefulness of full scale hull stress, ship motion and environmental information which has been obtained over the last several years.

W. F. REA, III
Rear Admiral, U. S. Coast Guard
Chairman, Ship Structure Committee

SSC-237

Final Technical Report

on

Project SR-187, "Ship Response Data Study"

PART II

COMPUTER PROGRAMS FOR THE DIGITIZING AND USING OF LIBRARY TAPES OF SHIP STRESS AND ENVIRONMENT DATA

by
Aldie E. Johnson, Jr.
James A. Flaherty
Isaac J. Walters

Teledyne Materials Research

under
Department of the Navy
Naval Ship Engineering Center
Contract No. N00024-69-C-5161

U.S. Coast Guard Headquarters
Washington, D.C.
1973

ABSTRACT

Details of computer programs and their operating instructions are given for the processing of logbook-type data and associated analogue stress signals into digital format. The logbook data is keypunched, edited and formatted for subsequent merging with the analogue signal which has been processed through an Analogue-to-Digital (A/D) converter. Accumulation of summary data during the processing is also output on to digital magnetic tape which is then available for use in statistical analyses. A program for retrieval of selected data from the digital magnetic tape is included.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SHIP STRUCTURE COMMITTEE

The SHIP STRUCTURE COMMITTEE is constituted to prosecute a research program to improve the hull structures of ships by an extension of knowledge pertaining to design, materials and methods of fabrication.

RADM W. F. Rea, III, USCG, Chairman
Chief, Office of Merchant Marine Safety
U.S. Coast Guard Headquarters

Capt. J. E. Rasmussen, USN
Head, Ship Systems Engineering
and Design Department
Naval Ship Engineering Center
Naval Ship Systems Command

Mr. E. S. Dillon
Deputy Asst. Administrator
for Operations
Maritime Administration

Mr. K. Morland
Vice President
American Bureau of Shipping

CAPT L. L. Jackson, USN
Maintenance and Repair Officer
Military Sealift Command

SHIP STRUCTURE SUBCOMMITTEE

The SHIP STRUCTURE SUBCOMMITTEE acts for the Ship Structure Committee on technical matters by providing technical coordination for the determination of goals and objectives of the program, and by evaluating and interpreting the results in terms of ship structural design, construction and operation.

NAVAL SHIP ENGINEERING CENTER

Mr. P. M. Palermo - Chairman
Mr. J. B. O'Brien - Contract Administrator
Mr. G. Sorkin - Member
Mr. C. H. Pohler - Member

U. S. COAST GUARD

LCDR C. S. Loosmore - Secretary
CAPT H. H. Bell - Member
CDR J. L. Coburn - Member
CDR W. M. Devlin - Member

MARITIME ADMINISTRATION

Mr. J. J. Nachtsheim - Member
Mr. F. Dashnaw - Member
Mr. A. Maillar - Member
Mr. R. F. Coombs - Member
Mr. F. Seibold - Member

MILITARY SEALIFT COMMAND

Mr. R. R. Askren - Member
Mr. T. W. Chapman - Member
CDR A. McPherson, USN - Member
Mr. A. B. Stavovy - Member

AMERICAN BUREAU OF SHIPPING

Mr. S. Stiansen - Member
Mr. I. L. Stern - Member

NATIONAL ACADEMY OF SCIENCES
Ship Research Committee

Mr. R. W. Rumke - Liaison
Prof. R. A. Yagle - Liaison

SOCIETY OF NAVAL ARCHITECTS & MARINE ENGINEERS

Mr. T. M. Buerman - Liaison

BRITISH NAVY STAFF

Dr. V. Flint - Liaison

WELDING RESEARCH COUNCIL

Mr. K. H. Koopman - Liaison

INTERNATIONAL SHIP STRUCTURE CONGRESS

Mr. J. Vasta - Liaison

I.. INTRODUCTION

Midship bending stress data from four dry-cargo ships, accumulated during an eleven-year period ending in 1970 under Ship Structure Committee Project SR-153, "Ship Response Statistics" were available as analogue signals recorded on magnetic tape. Associated with these data were logbooks which contain hand-entry data relative to pertinent ship, sea and weather information. Subsequent to the collection of data, better techniques became available for digital processing of data by high-speed computers, thus permitting easier and better access to the data for statistical purposes.

In Reference 1, the general method was described which prepared and digitized the analogue signals and combined them with the corresponding logbook information. Further, demonstration examples were given of the retrieval of various items of data and of presentation formats for use in statistical analyses. Although no statistical analyses were intended as part of this project, ample evidence was given of the possibilities available through the use of the detailed programs developed during the project.

This report presents the details of the several computer programs which were developed to handle, process, edit, compact, retrieve and display the data originally recorded in analogue form (on magnetic tape) and as hand-entry logbooks. Although the programs were specifically written to handle midship bending stress and related data, they can be used with minor modifications to handle a broad variety of analogue signals.

II. COMPUTER PROGRAMS

To aid the reader, each of the several programs developed has been described in a separate Appendix as shown on Table I. Thus, use of one or more of the programs for other applications should be made more convenient. However, a short description of the major programs is given here with the subroutine and peripheral programs given only brief reference.

The logbook data were prepared as punched card input. The analogue processing equipment utilized a perforated paper-tape input, thus a Logbook Pre-Processor program was used to edit the punched-card data and prepare the perforated paper tape. The paper tape also contained operating instructions for the computer. These paper tapes were then read into the processor computer and stored for subsequent use and merging with the digitized stress signals. In Appendix A, are given the program

TABLE I - LIST OF PROGRAMS

APPENDIX A	-	LOGBOOK PRE-PROCESSOR PROGRAM
APPENDIX B	-	LOGBOOK PAPER TAPE LOAD PROGRAM
APPENDIX C	-	DATA CONVERSION AND ANALYSIS PROGRAM
APPENDIX D	-	SUMMARY TAPE AND EDIT PROGRAM
APPENDIX E	-	FINAL SUMMARY TAPE PROGRAM
APPENDIX F	-	SUMMARY TAPE CORRECTION PROGRAM
APPENDIX G	-	SUMMARY TAPE LISTING PROGRAM
APPENDIX H	-	PARAMETRIC STUDIES PROGRAM
APPENDIX I	-	RELATIVE WIND DIRECTION CORRECTION SUBROUTINE

details for the Logbook Pre-Processor program and in Appendix B are given the details of the Logbook Paper-Tape Load program which reads and stores the logbook data in the computer.

The main processing program (called Data Conversion and Analysis) operated in a real-time environment through the Real-Time Programmable Clock within the computer. In Appendix C are given the details of the program, written in Assembler Language which permitted processing to be done at a rate increase factor up to 25 over the original recording rate (0.3 inch/second) without requiring starting and stopping of the analogue playback unit.

The practicality of processing and writing a complete digital data tape without errors (which would have necessitated considerable rerunning of much already completed and correct data) early indicated a requirement for the capability to edit and compact partially filled data tapes onto one essentially filled data tape. A Summary Tape and Edit program (Appendix D) provides this capability and permits certain editing options. For data retrieval to be used for statistical studies, the computer read time could be shortened appreciably by reading only the pertinent data and not reading the 12,000 pieces of raw data recorded for each interval. A Final Summary Tape program (Appendix E) was written which allowed the complete summary information for the 217 voyages to be recorded on two tapes.

The utilization of the data on the Final Summary Tapes required a program to read the tapes and perform the comparisons or to select certain data. The Parametric Studies program (Appendix H) permits the reading of the tapes and provides output of required information as punched cards, printout, or stored on magnetic tape for further processing.

The programs or subroutine given in the remaining Appendices were used to list data on the magnetic tape (Appendix G) and correct the relative wind computation (Appendices F and I).

III. RESULTS

Use of the several computer programs given in the Appendices permitted the processing of the analogue stress data and logbook information into digital form and allowed selective retrieval of data for use in statistical studies. As a point of reference, typical usage of the program can be used to determine approximate costs for additional or similar use. The typical sequence of activities to process 100 intervals (and associated logbook information) and output to a Final Summary Tape consists of the following.

The keypunched logbook data are loaded onto perforated paper tape (using a standard IBM 1130 computer with paper-tape punch). Approximately one hour of computer time is required to list, edit and prepare the paper tape (including verification printout). The paper tape is read into a PDP-8/I Computer and the data stored on DECtape using approximately 30 minutes of computer processor time. To digitize and process 100 intervals (originally recorded for approximately 32 minutes each at 0.3 inch/minute) at a speed-up factor of 25, requires approximately 2 1/4 hours of PDP-8/I computer time.

The preparation of a Final Summary Tape from several tapes is dependent on several factors. The generation of a tape (equivalent to the full-bridge data tape of approximately 7700 intervals from 15 data tapes) would require approximately 1 hour of IBM 360/65 computer time. It would require approximately 10 minutes to run such a Final Summary Tape through the IBM 360/65 to retrieve the data from the PARM program. However, judicious use of the program permits several studies to be run with each pass of the tape through the computer. For example, the first eight

of the demonstration examples given in Reference 1, were retrieved in one pass. Depending on the output specified, mechanical card sorting and preparation of computer plots are very dependent on equipment used and operator experience.

IV. CONCLUDING REMARKS

The details of the computer programs necessary to prepare and digitize analogue and logbook data obtained during eleven years of acquiring midship bending stress data from four dry-cargo vessels under Ship Structure Committee Project SR-153, "Ship Response Statistics" are given. In addition, the details are given of the program used to retrieve selected data and present the results in a form for statistical analysis.

While the programs were written for the specific application, only minor modifications would be required to permit utilization on a wider variety of logbook-type data and recorded analogue signals.

V. REFERENCES

1. Johnson, A. E. Jr., Flaherty, J. A., and Walters, I. J. A Method for Digitizing, Preparing and Using Library Tapes of Ship Stress and Environment Data, Ship Structure Committee Report SSC-236, 1973.

APPENDIX A

LOGBOOK PREPROCESSOR PROGRAM

INTRODUCTION

This program was written to take the logbook data, which has been punched on computer cards, and process it to output as a punched paper tape for subsequent merging with the analogue signal data to produce the digital magnetic tape of data.

Logbook data had been recorded in four slightly different formats during the data acquisition projects. This program accepts data in any of the four formats (as indicated on the header card) and converts to a standard format.

After completion of all logbook data preprocessing and during the production runs of data, it was determined that there was an error in converting certain wind direction data to the standard format Relative Wind. A subroutine was subsequently written (see Appendix I) to correct the Relative Wind data at a later point in the editing portion of the processing. While this subroutine could be incorporated readily into the Logbook Preprocessor program to provide the correct information on the punched paper tapes (and eliminate the need for correction during the edit process), it has not been incorporated into the program listing given herein.

GENERAL DESCRIPTION

The program has been code named VOYAGE and herein is referred to by that name rather than the longer and more descriptive title.

VOYAGE reads logbook data in the form of header information (6 header cards), interval information (2 cards for each interval) and voyage identification information (2 cards for each voyage). The format of the input cards is shown in Table A-I. The program prints the input cards, punches a paper tape from the information and prints an image of the punched output according to the options specified by the operator. The program allows for any combination of the operations specified above except printing of cards and printing the punched output simultaneously. The format of the output (punched paper tape) is given in Table A-II.

The program reads six header cards and performs the required operations. It then reads the numbers of voyage and interval cards specified in the 3rd header card, performing the required calculations and operations after each set of 2 cards is read. As these cards are read, a check is made on Column 78 to determine the presence of a voyage card. (Voyage Identification cards are identified by the letter V in Column 78.) If a voyage card is out of order or missing, an appropriate error message is typed.

There are four types of interval cards; formats A, B, C, and D. The format type is punched in the first interval card of each set. The program handles each type differently, providing conversions and information reordering where required. The interval punched output has only one form (format D). This form is arrived at

TABLE A-I - INPUT CARD FORMAT

Card No.	Item	Columns
Header 1	Library Tape Number	1-25
	Customer Name	26-55
	Contract Number	56-73
	Tape Label	74-77
	Card Number (H$\hat{1}$)	78-80
Header 2	Contract Number (short)	1-7
	Comments	8-72
	Tape Label	74-77
	Card Number (H$\hat{2}$)	78-80
Header 3	Voyage Number 1	1
	Number of Passes	2
	Number Intervals, Pass No. 1	3,4
	" " " No. 2	5,6
	" " " No. 3	7,8
	" " " No. 4	9,10
	Voyage Number 2	11
	Same as for Voyage No. 1	12-20
	Voyage Number 3	21
	Same as for Voyage No. 1	22-30
	Voyage Number 4	31
	Same as for Voyage No. 1	32-40
	Tape Label	74-77
	Card Number (H$\hat{3}$)	78-80
Header 4	Interval Deletions, Long Intervals or Rate	1-70 (5 c
	Tape Label	74-77 fi
	Card Number (H$\hat{4}$)	
Header 5	Interval Deletions, Long Intervals or Rate	1-70 (5 c fi
	Tape Label	74-77
	Card Number (H$\hat{5}$)	78-80

TABLE A-I - (Continued)

Header 6	Interval Deletions, Long Intervals or Rate	1-60 (5 cols. per interval)
	Tape Label	74-77
	Card Number (H$\hat{6}$)	78-80
Interval 1 (Formats A,B,C)		
	Data Format	1
	Tape Reference	2-13
	Interval No.	14-16
	Logbook Index Number	17-19
	Date	20-27
	Time	28-31
	Latitude	32-38
	Longitude	39-46
	Course	47-49
	Speed	50-53
	Engine RPM	54-57
	Wind Speed	58-59
	Wind Direction	60-61
	Beaufort Sea State	64-65
	Wave Direction	66-69
	Wave Height	70-71
	Wave Period	72-73
	Card Identification	74-80
Interval 2 (Formats A,B,C)	Wave Length	1-3
	Swell Height	4-5
	Swell Length	6-10
	Swell Direction	11-14
	Barometer	15-19
	Sea Temperature	20-21
	Air Temperature	22-24
	Weather	25-44
	Comments	45-73
	Card Identification	74-80
Interval 1 (Format D)	Data Format	1
	Tape Reference	2-13
	Interval Number	14-16
	Logbook Index No.	17-19
	Date	20-27
	Time	28-31
	Latitude	32-38
	Longitude	39-46

TABLE A-I - (Concluded)

	Course	47-49
	Speed	50-53
	Engine RPM	54-57
	Sea Temperature	58-59
	Air Temperature	60-62
	True Wind Speed	63-64
	True Wind Direction	65-68
	Weather	69-73
	Card Identification	74-80
Interval 2 (Format D)	Weather (continued)	1-15
	Beaufort Sea State	16-17
	True Wave Direction	18-21
	Wave Height	22-23
	Wave Period	24-25
	Wave Length	26-28
	Swell Height	29-31
	Swell Length	32-35
	True Swell Direction	36-39
	Barometer	40-44
	Comments	45-73
	Card Identification	74-80
Voyage 1	Ship Name	1-16
	Voyage Number	17-19
	Date Voyage Start	20-27
	Date Voyage End	28-35
	Location (first/lo)	36-67
	Route Code	68-72
	Tape Label	74-77
	Card Number (H$\hat{1}$)	78-80
Voyage 2		
	FSI Tape Reference	1-35
	Ship Calibration Factor	36-39
	Gage Location (Port/Stbd)	40-43
	Gage Location (Longitudinal)	44-52
	Draft, forward	53-57
	Draft, mid	58-62
	Draft, aft	63-67
	Tape Label	74-77
	Card Number (H$\hat{2}$)	78-80

TABLE A-II - Punched Tape Output
(128 Characters/Block)

Block	Item	Characters
Header 1	Library Tape Number	1-25
	Customer Name	26-55
	Contract Number	56-80
	Comments	81-128
Header 2	Comments (continued)	1-17
	Blank	18-19
	Number of Voyage	20
	Voyage 1	
	Number of passes	21
	Number Intervals, pass 1	22-23
	" " pass 2	24-25
	" " pass 3	26-27
	" " pass 4	28-29
	Voyage 2	
	Same as Voyage 1	30-38
	Voyage 3	
	Same as Voyage 1	39-47
	Voyage 4	
	Same as Voyage 1	48-56
	Deletions, Long Intervals, Halts (5 char. ea.)	57-128
Header 3	Deletions, Long Intervals, Halts (cont.)	1-128
Interval 1	Tape Reference	1-12
	Logbook Index Number	13-15
	Interval Number	16-18
	Date	19-26
	Time	27-30
	Latitude	31-37
	Longitude	38-45
	Course	46-48
	Speed	49-52

TABLE A-II - (Concluded)

	Engine RPM	53-56
	Beaufort Sea State	57-58
	Relative Wind Direction	59-62
	Relative Wind Velocity	63-64
	True Wind Velocity	65-66
	Relative Wave Direction	67-70
	Wave Height	71-72
	Wave Period	73-74
	Wave Length	75-77
	Relative Swell Direction	78-81
	Swell Height	82-84
	Swell Length	85-88
	Barometer	89-93
	Sea Temperature	94-95
	Air Temperature	96-98
	Weather	99-118
	Blank	119-128
Interval 2	Comments	1-29
Voyage 1	Ship Name	1-16
	Voyage Number	17-19
	Date Voyage Start	20-27
	Date Voyage End	28-35
	Route (from/to)	36-67
	Route Code	68-72
	FM Tape Reference	73-107
	Ship Calibration Factor	108-111
	Gage Location (port/stbd)	112-115
	Gage Location (longit.)	116-124
	Blank	124-128
Voyage 2	Draft, fwd.	1-5
	Draft, mid	6-10
	Draft, aft	11-15
	Blank	16-29

by reordering and converting the input as required.

If the operator requests that the cards be listed on the printer, the header cards will be listed first. Then the first card from each interval in a voyage will be listed, while the second card from each interval is stored on disk. When all interval cards for a voyage are read, the second card for each interval will be printed followed by the two voyage cards.

An "End of Job" message on the console typewriter signals the successful completion of the run.

A listing of the program is given in Table A-III and the flow chart is given in Figures A-1 and A-2.

SYSTEM REQUIREMENTS

VOYAGE requires a modified version of the PAPTZ routine. This modified version is program No. 1130-03.440.6. However, this "ASCII PAPTZ" was modified to provide for the specific requirements. These modifications consist of:

(1) Changing some entries in the conversion table to give the correct punched-paper-tape characters.

(2) Changing the program to prevent the output of an EOR (End of record) character following a 72 character group.

VOYAGE requires the 1130 Commercial Subroutines VIII GET, PUT, WHOLE and NZONE.

VOYAGE requires 100 sectors in the Fixed Area for its file, VFILE.

The VOYAGE program is made up of two "Links", that is, two core loads. The first link, VOY, types the printing and punching option messages. It also operates on the six header cards. The second link, VOY1, performs the remainder of the operations. Both links are stored in Core Image form, in order to reduce execution time.

OPERATING INSTRUCTIONS

1. Card Input

// JOB

// XEQ VOY

1st header card

2nd header card

3rd header card

4th header card

5th header card

6th header card

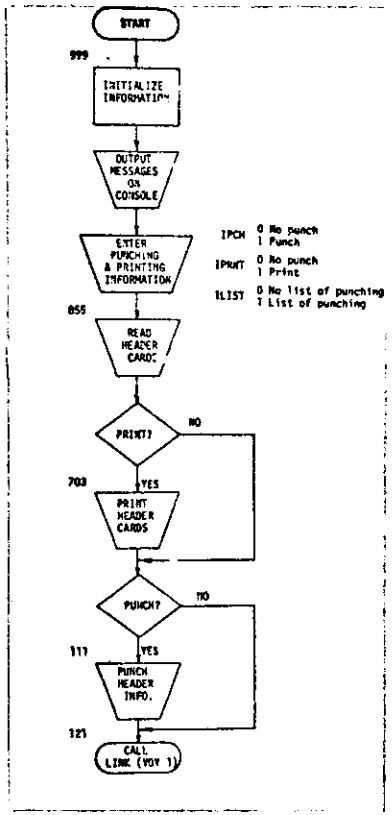


FIG. A-1 - LOGBOOK PRE-PROCESSOR COMPUTER PROGRAM FLOW CHART -- VOY (1st LINK)

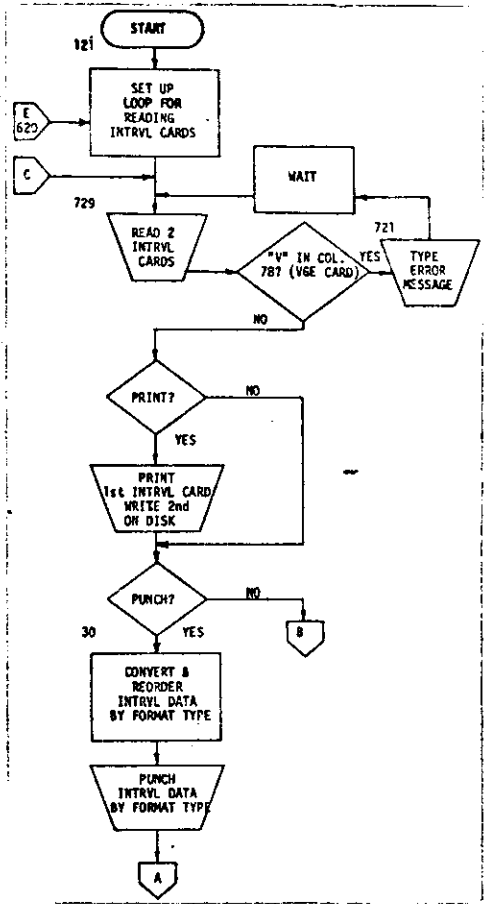


FIG. A-2 - LOGBOOK PREPROCESSOR COMPUTER PROGRAM FLOW CHART -- VOY1 (2nd LINK)

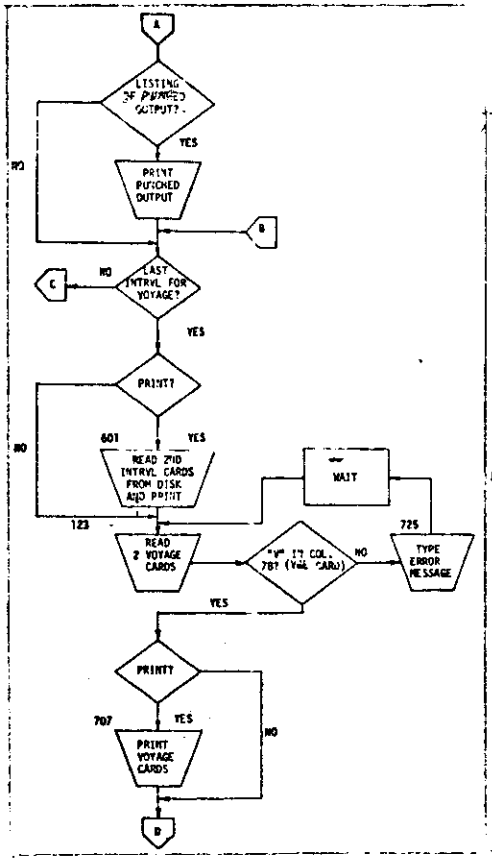


FIG. A-2-(Continued)

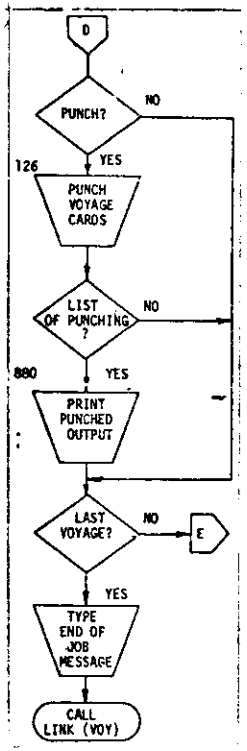


FIG. A-2-(Concluded)

TABLE A-III - PROGRAM LISTING

```
// JOB
// DUP
*DELETE             VOY
// FOR
*LIST ALL
*EXTENDED PRECISION
*ONE WORD INTEGERS
*IOCS(CARD,PAPER TAPE,TYPEWRITER,1132 PRINTER,KEYBOARD,DISK)
C                                                                    VYGE0005
C         VOYAGE PROGRAM                                             VYGE0010
C            *READS HEADER CARDS, INTERVAL CARDS AND VOYAGE          VYGE0012
C             IDENTIFICATION CARDS                                   VYGE0014
C            *PRINTS CARDS FOR CHECKING PURPOSES                     VYGE0018
C            *PUNCHES 125 CHARACTER BLOCKS ON PAPER TAPE FOR         VYGE0020
C             FURTHER ANALYSIS                                       VYGE0022
C                                                                    VYGE0024
      DEFINE FILE 10(300,8,U,IFILE)                                  VYGE0026
      DIMENSION INPT1(80),INPT2(80),          INPT4(80)              VYGE0028
      DIMENSION IVOYG(4,5),ITRW(2),ID(8)                             VYGE0030
      DIMENSION INPT5(80),INPT6(80)                                  VYGE0031
      COMMON IVOYG,ITRW,ID,IFILE,N,M,IASTR,ISTP,IPCH,ILIST,IPRNT     VYGE0032
      COMMON RAD,INVG                                                VYGE0034
  999 IFILE=1                                                        VYGE0036
      RAD=2*3.1416/360                                               VYGE0038
      ITRW(1)=16448                                                  VYGE0040
      ITRW(2)=16448                                                  VYGE0042
      M=4                                                            VYGE0044
      N=3                                                            VYGE0046
      IASTR=23616                                                    VYGE0110
      ISTP=19520                                                     VYGE0120
      WRITE(3,24)                                                    VYGE0130
   24 FORMAT(1H1)                                                    VYGE0140
C                                                                    VYGE0180
C     WRITE MESSAGE ON CONSOLE AND READ KEYBOARD TO DETERMINE        VYGE0190
C     IF PAPER TAPE IS TO BE PUNCHED                                 VYGE0200
C                                                                    VYGE0210
   11 WRITE(1,20)                                                    VYGE0220
   20 FORMAT(' VOYAGE PROGRAM-VYGE'/' ENTER 0 IF NO PUNCH,ENTER 1 IF PVYGE0230
     1UNCH')                                                         VYG 0240
      READ(6,21) IPCH                                                VYG 0250
      IF(IPCH) 850,853,850                                           VYGE0251
C     TYPE MESSAGE FOR LISTING PUNCHED OUTPUT                        VYGE0252
  850 WRITE(1,852)                                                   VYGE0253
  852 FORMAT(' ENTER 1 FOR LISTING OF PUNCHED OUTPUT,0 FOR NO LISTING') VYGE0254
      READ(6,21) ILIST                                               VYGE0255
      IF(ILIST) 854,853,854                                          VYGE0256
  854 IPRNT=0                                                        VYGE0257
      GO TO 855                                                      VYGE0258
   21 FORMAT(I1)                                                     VYG 0260
C        CONSOLE MESSAGE TO DETERMINE PRINTING OPTION                VYGE0261
  853 WRITE(1,701)                                                   VYGE0262
  701 FORMAT(' ENTER 0 IF NO PRINT,ENTER 1 IF PRINT')                VYGE0264
      READ(6,21) IPRNT                                               VYGE0266
C                                                                    VYG 0270
C     READ HEADER CARDS,PRINT HEADER CARDS AND PUNCH HEADER INFORMATION VYG 0280
C     IF REQUIRED                                                    VYG 0290
  855 READ(2,22) INPT1,INPT2,INVG,((IVOYG(I,J),J=1,5),I=1,4),ID,INPT4  VYGE0300
   22 FORMAT(80A1/80A1/I1,4(I1,4I2),35X,8A1/80A1)                    VYGE0310
      READ(2,65) INPT5,INPT6                                         VYGE0312
```

TABLE A-III - (Continued)

```
   65 FORMAT(80A1/80A1)
      IF(IPRNT) 703,702,703                                          VYGE0315
  703 WRITE(3,23) INPT1,INPT2                                        VYG 0320
   23 FORMAT(' ',25A1,5X,30A1,5X,18A1,26X,7A1/,' ',7A1,5X,65A1,31X,8A1) VYGE0330
      WRITE(3,27) INVG,((IVOYG(I,J),J=1,5),I=1,4),ID,INPT4           VYGE0340
   27 FORMAT(' ',I1,4(2X,I1,2X,I2,2X,I2,2X,I2,2X,I2),31X,8A1/' ',    VYGE0350
     1 14(5A1,2X),8X,10A1)                                           VYGE0355
      WRITE(3,32) INPT5,(INPT6(I),I=1,50),(INPT5(I),I=74,80)         VYGE0357
   32 FORMAT(' ',14(5A1,2X),8X,10A1/' ',12(5A1,2X),25X,7A1///)       VYGE0359
  702 IF(IPCH) 111,121,111                                           VYG 0360
C     PUNCH PAPER TAPE                                               VYG 0370
  111 WRITE(M,26)(INPT1(I),I=1,73),(INPT2(I),I=1,55),ISTP,(INPT2(I),  VYGE0380
     1 I=56,72),INVG,((IVOYG(I,J),J=1,5),I=1,4),(INPT4(I),I=1,70),   VYGE0390
     2 INPT5(1),INPT5(2),ISTP                                        VYGE0395
   26 FORMAT(72A1/57A1/17A1,2X,I1,4(I1,4I2),16A1/57A1)               VYGE0400
      WRITE(M,26) (INPT5(I),I=3,70),(INPT6(I),I=1,60),ISTP           VYGE0401
      IF(ILIST) 856,121,856                                          VYGE0402
  856 WRITE(N,29)(INPT1(I),I=1,73),(INPT2(I),I=1,55),ISTP,(INPT2(I),  VYGE0403
     1 I=56,72),INVG,((IVOYG(I,J),J=1,5),I=1,4),(INPT4(I),I=1,70),   VYGE0404
     2 INPT5(1),INPT5(2),ISTP                                        VYGE0405
   29 FORMAT(' ',72A1/' ',57A1/' ',17A1,2X,I1,4(I1,4I2),16A1/' ',57A1) VYGE0406
      WRITE(N,29) (INPT5(I),I=3,70),(INPT6(I),I=1,60),ISTP           VYGE0408
  121 CALL LINK(VOY1)                                                VYGE0409
      END                                                            VYGE0407
// DUP
*STORECI    WS  UA  VOY
*FILES(10,VFILE)
```

TABLE A-III - (Continued)

```
// JOB
// DUP
*DELETE              VOY1
// FOR
*EXTENDED PRECISION
*ONE WORD INTEGERS
*IOCS(CARD,PAPER TAPE,TYPEWRITER,1132 PRINTER,KEYBOARD,DISK)
      DEFINE FILE 10(300,81,U,IFILE)                                    VYGE0408
      DIMENSION INPT1(80),INPT2(80)                                     VYGE0409
      DIMENSION IVOYG(445),ITRW(2),ID(8)                                VYGE041C
      COMMON IVOYG,ITRW,ID,IFILE,N,M,IASTR,ISTP,IPCH,ILIST,IPRNT        VYGE0411
      COMMON RAD,INVG                                                   VYGE0412
C                                                                       VYG 0413
C     READ INTERVAL CARDS,PRINT INTERVAL CARDS AND PUNCH INTERVAL       VYG 0420
C     INFORMATION IF REQUESTED                                          VYG 0430
C                                                                       VYG 0440
C                                                                       VYGE0441
C     SET UP LOOP FOR READING INTERVAL CARDS                            VYGE0442
   24 FORMAT(1H1)                                                       VYGE0443
  121 ILOP=1                                                            VYGE0444
  630 IF(ILOP-INVGT 620,620,600                                         VYGE0445
  620 ININT=0                                                           VYGE0446
      DO 610 I1=2,5                                                     VYGE0447
  610 ININT=ININT+IVOYG(ILOP,I1)                                        VYGE0448
      DO 650 INIT=1,ININT                                               VYGE0450
  729 READ(2,28)(INPT1(I),I=1,49),SPEED,(INPT1(I),I=54,80),INPT2        VYGE0451
   28 FORMAT(49A1,F4.1,27A1/80A1)                                       VYGE0460
      IF (INPT1(78)+6848) 720,721,720                                   VYGE0465
  720 IF(INPT2(78)+6848) 722,721,722                                    VYGE0470
C        TEST FOR PRINT OPTION                                          VYGE0472
  722 IF(IPRNT) 705,141,705                                             VYGE0474
C     TEST FOR CARD FORMAT                                              VYGE0476
  705 IF(INPT1(1)+152961) 122,140,122                                   VYGE0480
C     PRINT D FORMAT CARDS                                              VYGE0483
  140 WRITE(3,50)(INPT1(I),I=1,49),SPEED,(INPT1(I),I=54,80)             VYGE0484
   50 FORMAT(' ',1A1,2X,12A1,2(2X,3A1),2X,8A1,2X,4A1,2X,7A1,2X,8A1,2X,  VYGE0485
     13A1,2X,F4.1,2X,4A1,2X,2A1,2X,3A1,2X,2A1,2X,4A1,2X,5A1,2X,7A1)     VYGE0486
      WRITE(10'IFILE) INPT1(I),INPT2                                    VYGE0487
      GO TO 141                                                         VYGE0490
  122 WRITE(3,31)(INPT1(I),I=1,49),SPEED,(INPT1(I),I=54,80)             VYGE0500
   31 FORMAT(' ',1A1,2X,12A1,2X,3A1,2X,3A1,2X,5A1,2X,4A1,2X,7A1,2X,     VYGE0510
     18A1,2X,3A1,2X,F5.1,2X,4A1,2X,2A1,2X,4A1,2X,2A1,2X,4A1,2X,2A1,2X,  VYG 0520
     22A1,2X,7A1)                                                       VYGE0530
      WRITE (10'IFILE) INPT1(I),INPT2                                   VYGE0540
  141 IF(IPCH) 30,650,30                                                VYGE0561
C                                                                       VYGE0562
C        BEGIN CALCULATIONS FOR PUNCHING INTERVAL CARDS                 VYGE0563
C                                                                       VYGE0568
C        TEST FOR FORMAT A                                              VYGE0569
   30 IF(INPT1(1)+16064)55,56,55                                        VYGE0570
   56 WRITE(M,33)(INPT1(I),I=2,49),SPEED,(INPT1(I),I=54,71),ITRW        VYGE0580
   33 FORMAT(12A1,T16,3A1,T19,3A1,T19,30A1,T49,F4.1,T53,4A1,T63,2A1,T59,VYGE0590
     14A1,T57,2A1,T67,6A1,T65,2A1)                                      VYGE05V0
      WRITE(M,34) INPT1(72),INPT1(73),(INPT2(I),I=1,44),ISTP            VYG 0600
   34 FORMAT(5A1,T10,7A1,T8,4A1,T17,30A1,T57,A1)                        VYG 0610
C                                                                       VYG 0620
C     SECOND BLOCK OF INTERVAL DATA TO BE PUNCHED                       VYG 0630
C                                                                       VYG 0640
      WRITE(M,35) (INPT2(I),I=45,73),ISTP                               VYG 0650
   35 FORMAT(30A1)                                                      VYGE0660
```

TABLE A-III - (Continued)

```
      IF(ILIST) 890,891,890                                             VYGE0661
  890 WRITE(N,17)(INPT1(I),I=2,49),SPEED,(INPT1(I),I=54,71),ITRW        VYGE0662
   17 FORMAT(' ',12A1,T17,3A1,T14,3A1,T20,30A1,T50,F4.1,T54,4A1,T64,2A1,VYGE0663
     1T60,4A1,T58,2A1,T68,6A1,T66,2A1)                                  VYGE0664
      WRITE(N,12) INPT1(72),INPT1(73),(INPT2(I),I=1,44),ISTP            VYGE0665
   12 FORMAT(' ',5A1,T11,7A1,T7,4A1,T18,30A1,T58,A1)                    VYGE0666
      WRITE(N,13) (INPT2(I),I=45,73),ISTP                               VYGE0668
   13 FORMAT(' ',30A1)                                                  VYGE0670
  891 ITRW(1)=16448                                                     VYGE0672
      ITRW(2)=16448                                                     VYGE0674
C     GO READ NEXT INTERVAL CARD                                        VYG 0676
  650 CONTINUE                                                          VYGE0678
      ILOP=ILOP+1                                                       VYGE0680
C        PRINT SECOND INTERVAL CARDS                                    VYGE0682
      IF(IPRNT) 123,123,601                                             VYGE0683
  601 WRITE(3,24)                                                       VYGE0684
      KEND=IFILE-1                                                      VYGE0685
      DO 660 I=1,KEND                                                   VYGE0686
      READ (10'I) INPT1(1),INPT2                                        VYGE0687
      IF(INPT1(1)+152961) 920,921,920                                   VYGE0688
  920 WRITE(3,32) INPT2                                                 VYGE0689
      GO TO 660                                                         VYGE0690
  921 WRITE(3,51) INPT2                                                 VYGE0691
  660 CONTINUE                                                          VYGE0692
      IFILE=1                                                           VYGE0693
   32 FORMAT(' ',3A1,4X,3A1,4X,4A1,4X,4A1,4X,5A1,4X,2A1,4X,3A1,4X,     VYG 0694
     1 20A1,4X,29A1,4X,7A1)                                             VYG 0695
   51 FORMAT(' ',15A1,2X,2A1,2X,4A1,2(2X,2A1),2X,3A1,2X,3A1,2X,4A1,2X,  VYGE0696
     14A1,2X,5A1,2X,29A1,2X,7A1)                                        VYGE0697
C                                                                       VYG 0698
C     READ VOYAGE IDENTIFICATION CARDS,PRINT VOYAGE IDENTIFICATION      VYG 0700
C     CARDS AND PUNCH IT INTO PAPER TAPE IF REQUIRED                    VYG 0710
C                                                                       VYG 0720
  123 READ(2,37) INPT1,INPT2                                            VYG 0730
      IF(INPT1(78)+6848) 725,724,725                                    VYGE0732
  724 IF(INPT2(78)+6848) 725,726,725                                    VYGE0734
   37 FORMAT(80A1)                                                      VYG 0740
  726 IF(IPRNT) 707,706,707                                             VYGE 745
  707 WRITE(3,38) INPT1                                                 VYG 0750
   38 FORMAT(///' ',16A1,5X,3A1,5X,8A1,5X,8A1,5X,3ZA1,5X,5A1,5X,8A1)    VYG 0760
      WRITE(3,39) INPT2                                                 VYG 0770
   39 FORMAT(' ',35A1,5X,4A1,5X,4A1,3X,9A1,3(5X,8A1),15A1)              VYG 0780
C     CHECK FOR PUNCHING                                                VYG 0790
  706 IF (IPCH) 126,10,126                                              VYG 0800
C     CONTINUE FOR PUNCHING                                             VYG 0810
  126 WRITE(M,40)(INPT1(I),I=1,72),(INPT2(I),I=1,52),ISTP,              VYGE0820
     1(INPT2(I),I=53,67),ISTP                                           VYG 0830
   40 FORMAT(72A1/52A1,4X,A1/15A1,14X,A1)                               VYGE0840
      IF(ILIST) 880,10,880                                              VYGE0841
  880 WRITE(N,14)(INPT1(I),I=1,72),(INPT2(I),I=1,52),ISTP,              VYGE0842
     1(INPT2(I),I=53,67),ISTP                                           VYGE0843
   14 FORMAT(' ',72A1/' ',52A1,4X,A1/' ',15A1,14X,A1)                   VYGE0844
   10 WRITE(3,24)                                                       VYGE0845
      GO TO 630                                                         VYGE0846
C                                                                       VYG 0850
C                                                                       VYG 0870
C        CHECK FOR FORMAT 'D'                                           VYGE0880
C        CHECK FOR BLANK FIELDS                                         VYGE0881
   55 IF(INPT1(1)+15296) 833,834,833                                    VYGE0882
```

TABLE A-III (Continued)

```
834  M1=67                                                          VYGE0883
     M2=38                                                          VYGE0884
     GO TO 831                                                      VYGE0885
835  M1=62                                                          VYGE0886
     M2=13                                                          VYGE0887
831  IF(INPT1(M1)-16448) 835,836,835                                VYGE0888
835  ISW1=0                                                         VYGE0889
     GO TO 837                                                      VYGE0890
836  ISW1=1                                                         VYGE0891
837  IF(INPT1(11)+15296) 838,839,838                                VYGE0892
839  IF(INPT2(20)-16448) 840,841,840                                VYGE0893
840  ISW2=0                                                         VYGE0894
     GO TO 842                                                      VYGE0895
841  ISW2=1                                                         VYGE0896
     GO TO 842                                                      VYGE0897
838  IF(INPT1(60)-16448) 840,841,840                                VYGE0898
842  IF(INPT2(M2)-16448) 843,844,843                                VYGE0899
843  ISW3=0                                                         VYGE0900
     GO TO 845                                                      VYGE0901
844  ISW3=1                                                         VYGE0902
845  IF(INPT1(11)+15296) 57,58,57                                   VYGE0903
 57  WNDR=GET(INPT1,60,62,1,0)                                      VYGE0904
     WVDR=GET(INPT1,66,69,1,0)                                      VYGE0910
     SWDR=GET(INPT2,11,13,1,0)                                      VYGE0920
C    CHECK FOR FORMAT 'C'                                           VYGE0930
     IF(INPT1(1)+15552) 59,60,59                                    VYGE0940
 60      L=60                                                       VYGE0950
         K=63                                                       VYGE0960
C    CONVERT RELATIVE WIND AND WAVE DIRECTION TO REL P AND S        VYGE0970
 69  IF(WNDR-180.) 61,62,63                                         VYGE0980
 61  INPT1(K)=-7616                                                 VYGE0990
     GO TO 64                                                       VYGE1000
 62  INPT1(K)=16448                                                 VYGE1110
     GO TO 64                                                       VYGE1120
 63  WNDR=360.-WNDR                                                 VYGE1130
     INPT1(K)=-10432                                                VYGE1140
 64  K=K-1                                                          VYGE1150
     CALL PUT(INPT1,L,K,WNDR,0.,0)                                  VYGE1160
     IF(K-62) 66,67,66                                              VYGE1170
 67  K=69                                                           VYGE1180
     L=66                                                           VYGE1190
     WNDR=WVDR                                                      VYGE1200
     GO TO 65                                                       VYGE1210
C    CONVERT RELATIVE SWELL DIRECTION TO RELTV P AND S              VYGE1220
 66  IF(SWDR-180.) 68,69,70                                         VYGE1230
 68  INPT2(14)=-7616                                                VYGE1240
     GO TO 71                                                       VYGE1250
 69  INPT2(14)=16448                                                VYGE1260
     GO TO 71                                                       VYGE1270
 70  INPT2(14)=-10432                                               VYGE1280
     SWDR=360.-SWDR                                                 VYGE1290
 71  CALL PUT(INPT2,11,13,SWDR,0.,0)                                VYGE1300
     IF(ISW1) 860,621,860                                           VYGE1310
860  DO 857 I=60,63                                                 VYGE1311
857  INPT1(I)=16448                                                 VYGE1312
621  IF(ISW2) 858,859,858                                           VYGE1313
858  DO 862 I=66,69                                                 VYGE1314
862  INPT1(I)=16448                                                 VYGE1315
859  IF(ISW3) 863,55,863                                            VYGE1316
                                                                    VYGE1317
```

TABLE A-III - (Continued)

```
863  DO 865 I=11,14                                                 VYGE1318
865  INPT2(I)=16448                                                 VYGE1319
     GO TO 56                                                       VYGE1320
C    CALCULATE RELATIVE FROM TRUE                                   VYGE1330
C                                                                   VYGE1340
C    FORMAT B                                                       VYGE1350
 59  WNDS=GET(INPT1,58,59,1,0)                                      VYGE1360
     CRSE=GET(INPT1,47,49,1,0)                                      VYGE1370
C    WAVE DIRECTION                                                 VYGE1380
 79  WVDR=WVDR-CRSE                                                 VYGE1390
     IF(WVDR) 74,75,75                                              VYGE1400
 74  WVDR=WVDR+360.                                                 VYGE1410
C    SWELL DIRECTION                                                VYGE1420
 75  SWDR=SWDR-CRSE                                                 VYGE1430
     IF(SWDR) 76,77,77                                              VYGE1440
 76  SWDR=SWDR+360.                                                 VYGE1450
C    WIND DIRECTION                                                 VYGE1460
C    CALCULATE RELATIVE WIND DIRECTION FROM TRUE WIND DIRECTION     VYGE1470
 77  CORS=CRSE                                                      VYGE1472
     CRSE=RAD*CRSE                                                  VYGE1474
     WNDR=RAD*WNDR                                                  VYGE1476
     SNSC=SPEED*COS(CRSE)                                           VYGE1480
     SEWC=SPEED*SIN(CRSE)                                           VYGE1490
C    WIND COMPONENTS                                                VYGE1500
     WNSC=WNDS*COS(WNDR)                                            VYGE1510
     WEWC=WNDS*SIN(WNDR)                                            VYGE1520
     RWNS=SNSC+WNSC                                                 VYGE1530
     RWEW=SEWC+WEWC                                                 VYGE1540
     THETA=RWEW/RWNS                                                VYGE1550
     THETA=ATAN(THETA)                                              VYGE1560
     THETA=THETA/RAD                                                VYGE1565
     IF(RWEW) 150,151,151                                           VYGE1570
150  THETA=THETA+180                                                VYGE1580
151  WNDR=THETA-CORS                                                VYGE1590
     IF(WNDR) 152,159,159                                           VYGE1600
152  WNDR=WNDR+360.                                                 VYGE1610
159  RWNDS=RWEW**2+RWNS**2                                          VYGE1620
     RWNDS=SQRT(RWNDS)                                              VYGE1630
C        CHECK FOR FORMAT 'B'                                       VYGE1635
     IF(INPT1(1)+15808) 153,154,153                                 VYGE1640
154  ITRW(1)=INPT1(58)                                              VYGE1650
     ITRW(2)=INPT1(59)                                              VYGE1660
     CALL PUT(INPT1,58,59,RWNDS,.5,0)                               VYGE1670
     GO TO 60                                                       VYGE1680
153  ITRW(1)=INPT1(63)                                              VYGE1690
     ITRW(2)=INPT1(64)                                              VYGE1700
     CALL PUT(INPT1,63,64,RWNDS,.5,0)                               VYGE1710
     GO TO 80                                                       VYGE1720
C                                                                   VYGE1730
C    FORMAT D RELATIVE FROM TRUE                                    VYGE1740
C                                                                   VYGE1750
 98  WNDR=GET(INPT1,65,67,1,0)                                      VYGE1760
     WVDR=GET(INPT2,18,20,1,0)                                      VYGE1770
     SWDR=GET(INPT2,36,38,1,0)                                      VYGE1780
     WNDS=GET(INPT1,63,64,1,0)                                      VYGE1790
     CRSE=GET(INPT1,47,49,1,0)                                      VYGE1800
     GO TO 79                                                       VYGE1810
C    FORMAT D CONVERT RELATIVE TO RELATIVE P AND S                  VYGE1820
 80  K=1                                                            VYGE1830
```

TABLE A-III - (Continued)

```
   85 IF(WNDR-180) 81,82,83                                          VYGE1840
   81 IMASK=-7616                                                    VYGE1850
      GO TO 84                                                       VYGE1860
   82 IMASK=16448                                                    VYGE1870
      GO TO 84                                                       VYGE1880
   83 WNDR=360.-WNDR                                                 VYGE1890
      IMASK=-10432                                                   VYGE1900
   84 GO TO (86,87,88),K                                             VYGE1910
   86 INPT1(68)=IMASK                                                VYGE1920
      CALL PUT(INPT1,65,67,WNDR,0.,0)                                VYGE1930
      WNDR=WVDR                                                      VYGE1940
      K=2                                                            VYGE1950
      GO TO 85                                                       VYGE1960
   87 INPT2(21)=IMASK                                                VYGE1970
      CALL PUT(INPT2,18,20,WNDR,0.,0)                                VYGE1980
      WNDR=SWDR                                                      VYGE1990
      K=3                                                            VYGE2000
      GO TO 85                                                       VYGE2010
   88 INPT2(39)=IMASK                                                VYGE2020
      CALL PUT(INPT2,36,38,WNDR,0.,0)                                VYGE2030
C     PUNCH 'D' FORMAT CARDS                                         VYGE2031
C        IF PROGRAM SWITCH IS ON, BLANK OUT APPROPRIATE FIELD        VYGE2032
C                                                                    VYGE2033
      IF(ISW1) 846,847,846                                           VYGE2034
  846 DO 848 I=65,68                                                 VYGE2035
  848 INPT1(I)=16448                                                 VYGE2036
  847 IF(ISW2) 849,623,849                                           VYGE2037
  849 DO 851 I=18,21                                                 VYGE2038
  851 INPT2(I)=16448                                                 VYGE2039
  623 IF(ISW3) 624,861,624                                           VYGE2040
  624 DO 626 I=36,39                                                 VYGE2041
  626 INPT2(I)=16448                                                 VYGE2042
  861 WRITE(M,91)(INPT1(I),I=2,49),SPEED,(INPT1(I),I=54,57),(INPT1(I),VYGE2043
     1I=63,68),(INPT2(I),I=16,23),ITRW                               VYGE2044
   91 FORMAT(12A1,T16,3A1,T19,3A1,T19,30A1,T49,F4.1,T53,4A1,T63,2A1,T59,VYGE2045
     14A1,T57,2A1,T67,8A1,T71,2A1,T65,2A1)                           VYGE2046
      WRITE(M,92) (INPT1(I),I=58,62),(INPT1(I),I=69,73),(INPT2(I),   VYGE2047
     1I=1,15),(INPT2(I),I=24,44),ISTP                                VYGE2048
   92 FORMAT(T22,5A1,T27,20A1,T1,5A1,T10,7A1,T6,4A1,T17,9A1,T57,A1)  VYGE2049
      WRITE(M,93)(INPT2(I),I=45,77),ISTP                             VYGE2050
      IF(ILIST) 870,94,870                                           VYGE2051
  870 WRITE(M,16)(INPT1(I),I=2,49),SPEED,(INPT1(I),I=54,57),(INPT1(I),VYGE2052
     1I=63,68),(INPT2(I),I=16,23),ITRW                               VYGE2053
   16 FORMAT(' ',12A1,T17,3A1,T19,3A1,T19,30A1,T50,F4.1,T54,4A1,T66,2A1,VYGE2054
     1T60,4A1,T58,2A1,T68,4A1,T72,2A1,T66,2A1)                       VYGE2055
      WRITE(M,15) (INPT1(I),I=58,62),(INPT1(I),I=69,73),(INPT2(I),   VYGE2056
     1I=1,15),(INPT2(I),I=24,44),ISTP                                VYGE2057
   15 FORMAT(' ',T23,9A1,T28,20A1,T2,5A1,T12,7A1,T7,4A1,T18,9A1,T58,A1) VYGE2058
      WRITE( N,13)(INPT2(I),I=45,73),ISTP                            VYGE2059
   94 ITRW(1)=16448                                                  VYGE2060
      ITRW(2)=16448                                                  VYGE2061
      GO TO 650                                                      VYGE2062
C                                                                    VYGE2063
C        ERROR ROUTINES                                              VYGE2064
C                                                                    VYGE2065
  721 WRITE(1,723)                                                   VYGE2066
  723 FORMAT (' VOYAGE CARD OUT OF ORDER')                           VYGE2068
      PAUSE                                                          VYGE2070
      GO TO 729                                                      VYGE2080
```

TABLE A-III - (Concluded)

```
  725 WRITE(1,727)                                                   VYGE2090
  727 FORMAT(' VOYAGE CARD MISSING')                                 VYGE2100
      PAUSE                                                          VYGE2110
      GO TO 123                                                      VYGE2120
  600 WRITE(M,970)                                                   VYGE2130
  970 FORMAT('//////////////////////////')                           VYGE2140
      WRITE(1,972)                                                   VYGE2150
  972 FORMAT(' END OF JOB')                                          VYGE2160
      CALL LINK(VOY)                                                 VYGE2170
      END                                                            VYGE2180
// DUP
*STORE CI   WS  UA  VOY1      1
*FILES(10,VFILE)
```

1st interval card

2nd interval card

. . .

. . .

. . .

1st interval card

2nd interval card

1st voyage identification card

2nd voyage identification card

and so on for all voyages.

The file locations for input cards are given in Table A-I.

For convenience, certain card input items have been number-coded for ease in data retrieval. The general routing terminations (see Voyage 1 card, Columns 68-72) are based on the following:

1 North America, East Coast
2 North America, West Coast
3 South America, East Coast
4 South America, West Coast
5 Northern Europe
6 Mediterranean
7 Africa
8 Persian Gulf
9 Orient
10 Australia

2. Console Messages

A. VOYAGE PROGRAM - VYGE

ENTER "0" IF NO PUNCH, ENTER "1" IF PUNCH

0 gives no punched output.
1 gives punched paper tape output.
Enter 0 or 1; then press EOF key.

B. ENTER 1 FOR LISTING OF PUNCHED OUTPUT, 0 FOR NO LISTING

If punched output is requested,

0 gives no listing of punched output.
1 gives image of punched output on printer.
Enter 0 or 1; then press EOF key.

C. ENTER 0 IF NO PRINT
ENTER 1 IF PRINT

If a listing of the punched output is not requested, this message will appear.

0 gives no listing.
1 gives an image of the input cards on the printer.
Enter 0 or 1; then press EOF key.

D. VOYAGE CARD OUT OF ORDER

A voyage identification card has been encountered before all interval cards have been read. Remove last two cards, read and correct, if possible. If not, restart program with corrected deck.

E. VOYAGE CARD MISSING

A valid voyage identification card does not follow the interval cards. (A voyage identification card is identified by a V in Column 78).

F. END OF JOB

All operations have been successfully completed.
Program returns to print option messages for another run.

3. Program Listing

The program listing is given in Table A-III.

APPENDIX B

LOGBOOK PAPER TAPE LOAD PROGRAM

The logbook punched paper tape (in ASCII format as outputted by the Preprocessor program--see Appendix A) is used as input for the PDP-8/I computer program "PAPT". The data on the punched paper tape is loaded on the PDP-8/I computer through the ASR-33 teletype. After being read into the computer, the data is converted to EBCDIC (Extended Binary Coded Decimal Interchange Code) to be consistent with the required magnetic tape format. After conversion, the data is stored on DECtape for subsequent merging with the digitized record during the data processing phase.

The punched paper tape has the data formatted in DECtape blocks which consist of 128_{10} characters/blocks. The first 3 blocks (i.e., Header) on the paper tape are utilized for the magnetic tape label and parameters needed for control of the data processing phase. The format for the Header blocks is given in Appendix A, Table A-II. The parameter used in the control of the data acquisition phase are the number of voyages to be written on each magnetic data tape, the number of original FM analogue tape passes for each voyage, the number of intervals in each pass of data, and a table giving the intervals which require special instructions; namely, the intervals that are long, to be deleted, or have program halts.

After the Header blocks the Interval Logbook data is read in. These consist of two blocks per interval. Four blocks are left blank after each logbook interval. These are utilized by the data processing program for storage of Interval Summary data (i.e., wave-induced peak-to-trough, RMS, and maximum peak-to-trough first-mode stresses, etc.).

The last two blocks in each voyage contain the Voyage Logbook data and are handled in the same manner as the Interval Logbook Data. Again, four DECtape blocks are left blank after each voyage for storage of Voyage Summary data.

The program which accomplishes the above is listed in Table B-I. Figure B-1 gives the flow chart of the program. The first 128 locations of the program contain parameters and the interrupt service routines. The interrupt system allows for overlap (multi-processing).

Locations 200_8 to 377_8 contain the DECtape handler routine. This is a standard routine supplied by Digital Equipment Corporation for reading from, and writing onto, the DECtape unit.

The program starts at location 0400_8, the interrupt enable is turned off and the counters and data storage buffers are cleared and initialized. The detailed procedure for loading the Paper-Tape Load Program (PAPT) and execution of it are given in Table B-II.

The Search subroutine searches for the Starting block on DECtape unit 3. After the starting block has been found, the teletype paper-tape feed switch is set to START, and the paper tape is read. The end of a block of data is distinguished by the special character (<) "less than". Use of the character eliminates the need to zero fill blocks of paper tape to get 128 characters. By using the de-limiter, the program assumes it is the end of a block of data. Since the teletype keyboard only reads 10 characters/sec, considerable unnecessary reading of zeroes is eliminated, thus speeding the reading of paper tapes.

As each character is read it is converted from ASCII to EBCDIC by the subroutine RECODE. After the Header information (3 blocks) is read in, the program uses the parameter given in the last two blocks of the paper-tape header to control the subsequent reading of the paper tape.

After the last voyage, logbook data is read in the program halts. The reading in of the paper tape is complete.

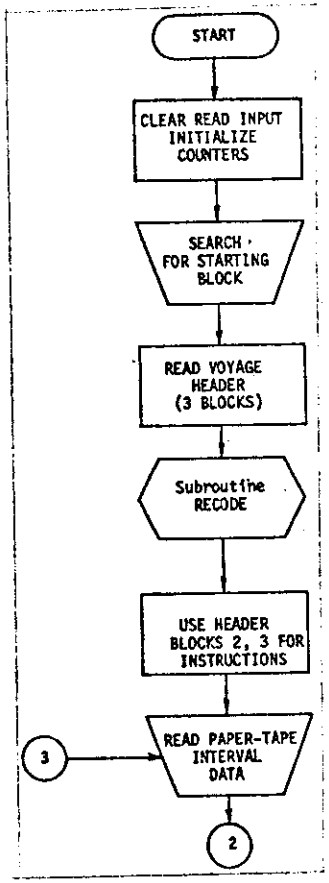


FIG. B-1 - FLOW CHART FOR PAPER TAPE LOAD

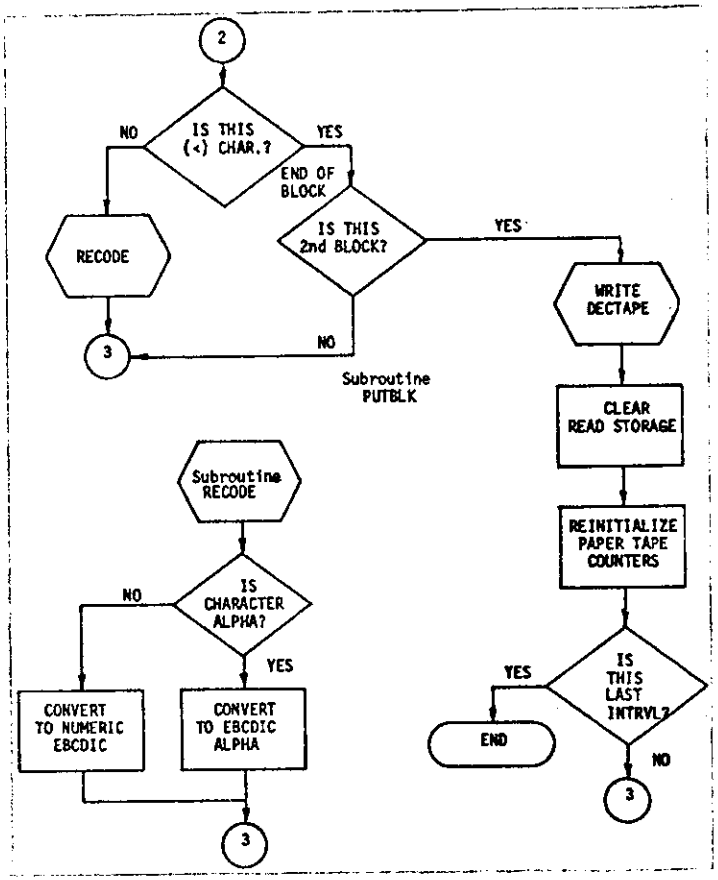


FIG. B-1 - (Continued)

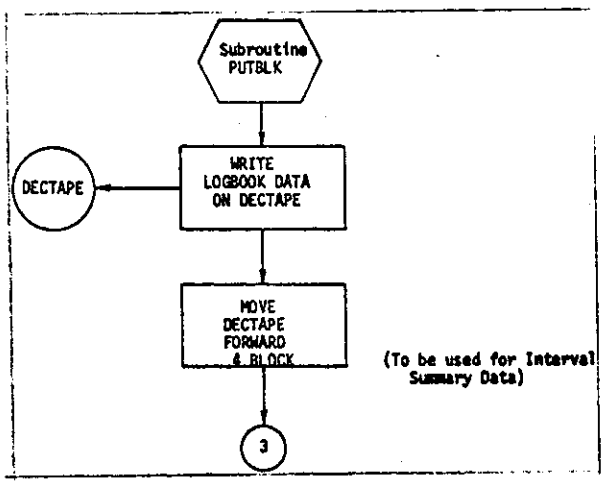


FIG. B-1 - (Concluded)

TABLE B-I - PROGRAM LISTING FOR LOGBOOK PAPER-TAPE LOAD PROGRAM

```
AC      0051    DTS1    0304    PITRLK  0466
ACA     0156    DTS2    0310    QUEST1  0743
ACB     0157    DTS3A   0323    RECODE  0600
ACC     0160    DTURN   0347    R128    0200
AGAIN   0434    DTURNX  0257    SEARCH  0104
AGAINA  0050    DWF     0230    SEROUT  0112
AGAINB  0161    DWAIT   0371    SETUP   0501
AGAIN1  0443    DWC     0264    SETUP1  0510
AGAIN3  0454    D2      0370    SETUP2  0511
AGAIN4  0457    D20     0270    SHL     7413
ALPHA   0617    D200    0276    SLASH   0741
ALPHAJ  0634    D30     0266    SLASH1  0742
ALPHAS  0645    D400    0273    SPACE2  0731
ALPHA9  0651    D614    0274    SPACE3  0732
ASR     7415    D7000   0333    TEST    0133
BLANK   0132    D7600   0335    W128    0206
BLOK    0130    FIRST   0144    W128Y   0175
BLOK1   0131    HOLD    0153    XR10    0010
BNO     0564    HOLD1   0154    XR11    0011
BSTART  0563    INT1    0142    XR12    0012
CLEAN   0746    LESS    0140    XR17    0017
CLEAN1  0754    LINK    0052    ZERO    0173
COMA    0733    LISN    0117
COM1    0734    LSH     7417
CORE    0127    MCOM    0054
COUNT   0147    MINUS2  0141
DASH    0735    MINUS3  0176
DASH1   0736    MQA     7501
DBLK    0272    MQL     7421
DCAA    0263    MUY     7405
DCINT   0267    M0017   0155
DCRET   0236    M100    0745
DECWRT  0546    M17     0730
DER     0351    M2222   0146
DGET    0232    M240    0723
DGR     0212    M272    0722
DINT    0252    M300    0724
DIS     0055    M301    0744
DNCR    0265    M312    0725
DOVE    0103    M323    0726
DRET    0227    M4      0136
DR127   0245    M4777   0174
DR128   0237    M7      0727
DR128C  0262    M7200   0134
DSERR   0277    M7400   0135
DSERR1  0116    M7400A  0137
DSTOP   0363    M7774   0150
DTBLK   0275    M7774A  0151
DTEM    0271    M7774B  0152
DTEMP   0232    NOBLKS  0145
DTEMX   0232    ONE     0143
DTERR   0100    OUT     0717
DTFIND  0357    PCREG   0053
DTLA    6766    PERID   0737
                PERID1  0740
```

```
                        /
                        /PAPER TAPE CONVERSION ROUTINE
                        / ASCII TO EBCDIC
                        /PROGRAM USED TO LOAD PAPER TAPE
                        / LOGBOOK INFORMATION
                        /ONTO DECTAPE FOR USE WITH
                        /A TO D SHIP PROGRAM
                        /
                        *0000
0000  0000              0000
0001  3451              DCA I AC
0002  7010              RAR
0003  3452              DCA I LINK
0004  1000              TAD 0000
0005  3453              DCA I PCREG
0006  7300              CLA CLL
0007  5020              JMP 0020
                        *0020
0020  2051              ISZ AC
0021  2052              ISZ LINK
0022  2053              ISZ PCREG
0023  6771              DTSF
0024  7410              SKP
0025  5454              JMP I MCOM
0026  6031              KSF
0027  7410              SKP
0030  5450              JMP I AGAINA
0031  5455              JMP I DIS
                        *0050
0050  0161  AGAINA,     AGAINB
0051  0000  AC,         0000
0052  0000  LINK,       0000
0053  0000  PCREG,      0000
0054  0000  MCOM,       0000
0055  0056  DIS,        .+1
0056  7300              CLA CLL
0057  6002              IOF
0060  7240              CLA CMA
0061  1051              TAD AC
0062  3051              DCA AC
0063  7240              CLA CMA
0064  1052              TAD LINK
0065  3052              DCA LINK
0066  7240              CLA CMA
0067  1053              TAD PCREG
0070  3053              DCA PCREG
0071  1453              TAD I PCREG
0072  3000              DCA 0000
0073  1452              TAD I LINK
0074  7104              CLL RAL
0075  1451              TAD I AC
0076  6001              ION
0077  5400              JMP I 0000
0100  0101  DTERR,      .+1
```

TABLE B-I - (Continued)

```
0101  7402          HLT
0102  5101          JMP .-1
0103  0371  DONE,   DWAIT
0104  0000  SEARCH, 0000
0105  7300          CLA CLL
0106  1130          TAD BLOK
0107  4516          JMS I DSERHI
0110  0112          SEROUT
0111  3000          3000                    /UNIT 3 FIELD 0
0112  7300  SEROUT, CLA CLL
0113  4503          JMS I DONE
0114  7300          CLA CLL
0115  5504          JMP I SEARCH
0116  0277  DSERHI, DSERH
0117  0000  LISN,   0000
0120  7300          CLA CLL
0121  6032          KCC
0122  6031          KSF
0123  5122          JMP .-1
0124  6036          KRB
0125  3153          DCA HOLD
0126  5517          JMP I LISN
0127  1777  CORE,   1777
            XR10=10
            XR12=12
0130  0011  BLOK,   0011
0131  0000  BLOK1,  0000
0132  0000  BLANK,  0000
0133  0000  TEST,   0000
0134  7200  M7200,  7200
0135  7377  M7400,  7377
0136  0006  M4,     0006
0137  0000  M7400A, 0000
0140  7504  LESS,   7504    /ASCII FOR LESS THAN
0141  7776  MINUS2, 7776
0142  0000  INT1,   0000
0143  7777  ONE,    7777
0144  0000  FIRST,  0000
0145  0000  NORLKS, 0
0146  2222  M2222,  2222
            XR11=0011
0147  0000  COUNT,  0000
0150  7774  M7774,  7774
0151  0000  M7774A, 0000
0152  0000  M7774B, 0000
0153  0000  HOLD,   0000
0154  0000  HOLD1,  0000
0155  0017  M0017,  0017
0156  6000  ACA,    6000
0157  6200  ACB,    6200
0160  6400  ACC,    6400
0161  7300  AGAINB, CLA CLL
0162  6032          KCC
0163  6034          KRS
0164  3412          DCA I XR12
```

TABLE B-I - (Continued)

```
0165  7300          CLA CLL
0166  1143          TAD ONE
0167  1173          TAD ZERO
0170  3173          DCA ZERO
0171  7300          CLA CLL
0172  5455          JMP I DIS
0173  0000  ZERO,   0000
            XR17=   0017
0174  4777  M4777,  4777
0175  0206  W128Y,  W128
0176  7775  MINUS3, 7775
                    MQL=7421
                    MUY=7405
                    MQA=7501
                    SHL=7413
                    ASR=7415
                    LSR=7417
                    *400
0400  6002          IOF
0401  7300          CLA CLL
0402  4777          JMS CLEAN
0403  7300          CLA CLL
0404  1156          TAD ACA
0405  3051          DCA AC
0406  1157          TAD ACB
0407  3052          DCA LINK
0410  1160          TAD ACC
0411  3053          DCA PCREG
0412  1127          TAD CORE
0413  3010          DCA XR10
0414  1130          TAD BLOK
0415  3131          DCA BLOK1
0416  1150          TAD M7774
0417  3151          DCA M7774A
0420  1150          TAD M7774
0421  3152          DCA M7774B
0422  4104          JMS SEARCH
0423  6002          IOF
0424  1176          TAD MINUS3
0425  3142          DCA INT1
0426  1176          TAD MINUS3
0427  3363          DCA BSTART
0430  1143          TAD ONE
0431  3144          DCA FIRST
0432  3173          DCA ZERO
0433  4117          JMS LISN
0434  7300  AGAIN,  CLA CLL
0435  4117          JMS LISN
0436  1132          TAD BLANK
0437  1153          TAD HOLD
0440  7650          SNA CLA
0441  5234          JMP AGAIN
0442  5257          JMP AGAIN4
0443  7300  AGAIN1, CLA CLL
0444  1173          TAD ZERO
```

TABLE B-I - (Continued)

```
0445  7650          SNA CLA
0446  5254          JMP AGAIN3
0447  1417          TAD I XR17
0450  3153          DCA HOLD
0451  2173          ISZ ZERO
0452  5257          JMP AGAIN4
0453  5257          JMP AGAIN4
0454  6002  AGAIN3, IOF
0455  7300          CLA CLL
0456  4117          JMS LISN
0457  1153  AGAIN4, TAD HOLD
0460  1140          TAD LESS
0461  7650          SNA CLA
0462  5266          JMP PUTBLK
0463  7300          CLA CLL
0464  4776          JMS RECODE
0465  5243          JMP AGAIN1
0466  2142  PUTBLK, ISZ INT1
0467  5243          JMP AGAIN1
0470  4346          JMS DECWRT
0471  1144          TAD FIRST
0472  7710          SPA CLA
0473  5301          JMP SETUP
0474  4777          JMS CLEAN
0475  2145          ISZ NORLKS
0476  5243          JMP AGAIN1
0477  7402          HLT
0500  5277          JMP .-1
0501  7300  SETUP,  CLA CLL
0502  3144          DCA FIRST
0503  1146          TAD M2222
0504  3011          DCA XR11
0505  1411          TAD I XR11
0506  0155          AND M0017
0507  3147          DCA COUNT
0510  2011  SETUP1, ISZ XR11
0511  7300  SETUP2, CLA CLL
0512  1411          TAD I XR11
0513  0155          AND M0017
0514  7421          MQL
0515  7405          MUY
0516  0012          0012
0517  7300          CLA CLL
0520  7501          MQA
0521  3154          DCA HOLD1
0522  1411          TAD I XR11
0523  0155          AND M0017
0524  1154          TAD HOLD1
0525  1147          TAD COUNT
0526  3147          DCA COUNT
0527  2151          ISZ M7774A
0530  5311          JMP SETUP2
0531  1150          TAD M7774
0532  3151          DCA M7774A
0533  2152          ISZ M7774B
```

TABLE B-I - (Continued)

```
0534  5310          JMP SETUP1
0535  7300          CLA CLL
0536  1147          TAD COUNT
0537  7041          CIA
0540  3145          DCA NOBLKS
0541  4777          JMS CLEAN
0542  2131          ISZ BLOK1
0543  1141          TAD MINUS2
0544  3363          DCA BSTART
0545  5243          JMP AGAIN1
0546  0000  DECWRT,  0000
0547  7300          CLA CLL
0550  3173          DCA ZERO
0551  1174          TAD M4777
0552  3017          DCA XR17
0553  1174          TAD M4777
0554  3012          DCA XR12
0555  3173          DCA ZERO
0556  1131          TAD BLOK1
0557  3364          DCA BNO
0560  4575          JMS I W128Y
0561  2000          2000
0562  3000          3000
0563  0000  BSTART,  0000
0564  0000  BNO,     0000
0565  7300          CLA CLL
0566  4503          JMS I DONE
0567  7300          CLA CLL
0570  1131          TAD BLOK1
0571  1136          TAD M4
0572  3131          DCA BLOK1
0573  1174          TAD M4777
0574  3017          DCA XR17
0575  5746          JMP I DECWRT
0576  0600
0577  0746

                    *200
                    /
                    /FOR DECTAPE ROUTINE
                    /
```

TABLE B-I - (Continued)

```
                    /TC01 SUB-ROUTINES      REV. 7/67
                DTXA=6764       /XOR AC TO STATUS A
                DTRB=6772       /READ STATUS B
                DTCA=6762       /CLEAR STATUS A
                DTRA=6761       /READ STATUS A
                DTLA=6766       /LOAD STATUS A (CLEAR AND XOR)
                DTLB=6774       /LOAD STATUS B
                DTSF=6771       /SKIP ON TC01 FLAGS
0200  0000  R128,  0                    /READ 128 WORDS
0201  4371          JMS DWAIT           /WAIT IF MOTION IS ON
0202  1200          TAD R128
0203  3206          DCA W128
0204  7201          CLA IAC             /SET TO WRITE
0205  5210          JMP DGR-2
0206  0000  W128,  0                    /WRITE 128 WORDS
0207  4371          JMS DWAIT           /WAIT IF MOTION IS ON
0210  1262          TAD DR128C          /DR128
0211  3227          DCA DRET            /READ WRITE RETURN AFTER SEARCH
0212  7240  DGR,    CLA CMA
0213  4232          JMS DGET
0214  3200          DCA R128            /FIRST CORE LOCATION-1 OF TRANSFER
0215  4232          JMS DGET
0216  3230          DCA DUF             /UNIT AND FIELD
0217  4232          JMS DGET
0220  3265          DCA DNCB            / -NUMBER OF BLOCKS TO BE TRANSFERRED
0221  1236          TAD DCRET
0222  3277          DCA DSERH
0223  3363          DCA DSTOP           /DON'T STOP TRANSPORT AFTER SEARCH
0224  4232          JMS DGET            /GET BLOCK NO.
0225  3271          DCA DTEM            /AND STORE
0226  5304          JMP DTS1            /INITIATE SEARCH
0227  0000  DRET,   0                   /DR128 IF WRITE,  OR DR128+1 IF READ
0230  0000  DUF,    0                   /UNIT AND FIELD
0231  5606          JMP I W128
                DTEMP,
                DTEMX,
0232  0000  DGET,   0                   /PICK UP ARGUMENTS
0233  1606          TAD I W128
0234  2206          ISZ W128
0235  5632          JMP I DGET
0236  0227  DCRET,  DRET
0237  1270  DR128,  TAD D28             /WRITE (NOT READ), (40-28)
0240  1266          TAD D30             /READ, CANCEL SEARCH (28+10)
0241  6764          DTXA                /SET FUNCTION
0242  1200          TAD R128            /1ST CORE LOC.-1 OF TRANSFER
0243  3663          DCA I DCAA          /TO 7755(CA)
0244  2054          ISZ MCOM            /POINT INTERRUPT RETURN TO DATA
```

TABLE B-I - (Continued)

```
                /PAGE 2
0245  6764  DR127,  DTXA            /SEND READ OR WRITE
0246  1335          TAD D7600       /SET WORD COUNT FOR 1 PAGE
0247  3664          DCA I DWC       /-128 TO 7754 (WC)
0250  5455          JMP I DIS       /EXIT
0251  5323          JMP DTS3A
0252  6772  DINT,   DTRB            /READ STATUS B
0253  7710          SPA CLA
0254  5351          JMP DER         /ERROR FLAG
0255  2265          ISZ DNCB        /COUNT BLOCKS
0256  5245          JMP DR127       /CONTINUE OPERATION
0257  1276  DTURNX, TAD D200       /COMPLEMENT MOTION AND DIRECTION
0260  1273          TAD D400
0261  5245          JMP DR127
0262  0237  DR128C, DR128
0263  7755  DCAA,   7755            /POINTER TO CURRENT ADDRESS
0264  7754  DWC,    7754            /POINTER TO WORD COUNT
0265  0000  DNCB,   0
0266  0030  D30,    30
0267  0251  DCINT,  DINT-1
0270  0020  D20,    20
0271  0000  DTEM,   0               /HOLDS REQUIRED BLK NO.
0272  0275  DBLK,   DTBLK
0273  0400  D400,   400             /CHANGE DIRECTION
0274  0614  D614,   614             /REVERSE, GO, SEARCH INTERRUPT ENABE
0275  0000  DTBLK,  0               /BLOCK NUMBER DEPOSITED HERE BY CONE
0276  0200  D200,   200             /CHANGE STOP/GO
0277  0000  DSERH,  0
0300  3271          DCA DTEM        /STORE BLOCK NO
0301  4371          JMS DWAIT
0302  1257          TAD DTURNX
0303  3363          DCA DSTOP       /STOP TRANSPORT AFTER SEARCH
0304  1272  DTS1,   TAD DBLK       /DTBLK TO 7755 (CA)
0305  3663          DCA I DCAA
0306  1267          TAD DCINT       /DINT-1
0307  3054          DCA MCOM        /INTERRUPT RETURN
0310  7201  DTS2,   CLA IAC
0311  1277          TAD DSERH       /DRET
0312  3232          DCA DTEMP
0313  1632          TAD I DTEMP
0314  0333          AND D7000       /PICK UP UNIT NUMBER
0315  1274          TAD D614        /SET TO SEARCH,NORMAL,REVERSE
0316  6766          DTLA            /LOAD STATUS A
0317  6774          DTLB            /FIELD 0
0320  2232          ISZ DTEMP
0321  6001          ION             /ENABLE INTERRUPT
0322  5632          JMP I DTEMP     /DUF+1, RETURN TO USER
```

TABLE B-I - (Continued)

```
                /PAGE 3
0323  6772  DTS3A,  DTRB            /READ STATUS B
0324  7006          RTL             /LOOK AT BIT 2
0325  7710          SPA CLA         /END ZONE?
0326  5257          JMP DTURNX      /YES (MOTION BIT=0), TURN
0327  6772          DTRB
0330  7710          SPA CLA
0331  5351          JMP DER         /ERROR FLAG BIT 0=1
0332  6761          DTRA
0333  7006  D7000,  RTL
0334  7006          RTL             /FOR-REV STATUS (BIT 3) IN LINK
0335  7600  D7600,  7600            /GROUP 2 CLA
0336  1275          TAD DTBLK
0337  7041          CMA IAC
0340  1271          TAD DTEM        /LINK COMP. IF REQUIRED BLK NO.
0341  7450          SNA             /IS BIGGER I.E. MUST GO FORWARD
0342  5357          JMP DTFIND      /FOUND BLOCK CHECK DIRECTION
0343  7041          CMA IAC
0344  7420          SNL
0345  7001          IAC             /GO 2 MORE BLOCKS BEFORE TURNING
0346  7420          SNL CLA
0347  1273  DTURN,  TAD D400        /TURN IF HERE
0350  5245          JMP DR127       /XOR TO A STATUS AND DISMIS
0351  6761  DER,    DTRA            /ERROR ROUTINE, READ STATUS A
0352  0276          AND D200        /STOP TAPE IF RUNNING,  I.E. SET BI
0353  1370          TAD D2          /DON'T CLEAR ERRORS     BIT 10=1
0354  6764          DTXA
0355  6772          DTRB            /ERROR STATUS B
0356  5500          JMP I DTERR
0357  7620  DTFIND, SNL CLA         /TEST DIRECTION
0360  5245          JMP DR127       /DONT TURN YET, STILL IN REVERSE
0361  1677          TAD I DSERH     /DRET, GET COMPLETION RETURN
0362  3277          DCA DSERH       /SINCE MOTION IS FORWARD
0363  0000  DSTOP,  0               /EITHER 0 (NOP) OR TAD D200 (STOP)
0364  6764          DTXA            /CLEAR FLAG
0365  1230          TAD DUF
0366  6774          DTLB            /SET MEMORY FIELD
0367  5677          JMP I DSERH     /EXIT TO COMPLETION RETURN
0370  0002  D2,     2
0371  0000  DWAIT,  0               /WAIT FOR NO MOTION
0372  6761          DTRA            /IOR STATUS A
0373  6761          DTRA            /AGAIN, IN CASE MOTION BIT
0374  0276          AND D200        /WAS 0 DUE TO END ZONE
0375  7640          SZA CLA
0376  5372          JMP .-4
0377  5771          JMP I DWAIT
      PAUSE   =600
              /
              /ASCII TO EBCDIC CONVERSION
              /
0600  0000  RECODE, 0000
0601  7300          CLA CLL
0602  1153          TAD HOLD
0603  1322          TAD M272        /IS IT ALPHA
```

TABLE B-I - (Continued)

```
0604  7500          SMA
0605  5217          JMP ALPHA
0606  7300          CLA CLL
0607  1153          TAD HOLD
0610  1323          TAD M260
0611  7510          SPA
0612  5251          JMP ALPHA9
0613  7300          CLA CLL
0614  1153          TAD HOLD
0615  1345          TAD M100        /NUMERIC
0616  5317          JMP OUT
0617  7300  ALPHA,  CLA CLL
0620  1153          TAD HOLD
0621  1344          TAD M301
0622  7510          SPA
0623  5251          JMP ALPHA9
0624  7300          CLA CLL
0625  1153          TAD HOLD
0626  1325          TAD M312
0627  7500          SMA
0630  5234          JMP ALPHAJ
0631  7300          CLA CLL
0632  1153          TAD HOLD
0633  5317          JMP OUT         /A THROUGH I
0634  7300  ALPHAJ, CLA CLL
0635  1153          TAD HOLD
0636  1326          TAD M323
0637  7500          SMA
0640  5245          JMP ALPHAS
0641  7300          CLA CLL
0642  1327          TAD M7
0643  1153          TAD HOLD
0644  5317          JMP OUT         /J THROUGH R
0645  7300  ALPHAS, CLA CLL
0646  1153          TAD HOLD
0647  1330          TAD M17
0650  5317          JMP OUT         /S THROUGH Z
0651  7300  ALPHA9, CLA CLL
0652  1153          TAD HOLD
0653  1333          TAD COMA
0654  7440          SZA
0655  5260          JMP .+3
0656  1334          TAD COM1
0657  5317          JMP OUT
0660  7300          CLA CLL
0661  1335          TAD DASH
0662  1153          TAD HOLD
0663  7440          SZA
0664  5267          JMP .+3
0665  1336          TAD DASH1
0666  5317          JMP OUT
0667  7300          CLA CLL
0670  1337          TAD PERID
0671  1153          TAD HOLD
0672  7440          SZA
```

TABLE B-I - (Concluded)

```
0673  5276          JMP .+3
0674  1340          TAD PERID1
0675  5317          JMP OUT
0676  7300          CLA CLL
0677  1341          TAD SLASH
0700  1153          TAD HOLD
0701  7440          SZA
0702  5305          JMP .+3
0703  1342          TAD SLASH1
0704  5317          JMP OUT
0705  7300          CLA CLL
0706  1331          TAD SPACE2
0707  1153          TAD HOLD
0710  7440          SZA
0711  5314          JMP .+3
0712  1332          TAD SPACE3
0713  5317          JMP OUT
0714  7300          CLA CLL
0715  1343          TAD QUEST1
0716  5317          JMP OUT
0717  3410  OUT,     DCA I XR10
0720  7300          CLA CLL
0721  5600          JMP I RECODE
0722  7506  M272,    7506
0723  7520  M260,    7520
0724  7477  M300,    7477
0725  7466  M312,    7466
0726  7455  M323,    7455
0727  0007  M7,      0007
0730  0017  M17,     0017
0731  7540  SPACE2,  7540
0732  0100  SPACE3,  0100
0733  7524  COMA,    7524
0734  0153  COM1,    0153
0735  7523  DASH,    7523
0736  0140  DASH1,   0140
0737  7522  PERID,   7522
0740  0113  PERID1,  0113
0741  7521  SLASH,   7521
0742  0141  SLASH1,  0141
0743  0157  QUEST1,  0157
0744  7477  M301,    7477
0745  0100  M100,    0100
0746  0000  CLEAN,   0000
0747  7300          CLA CLL
0750  1127          TAD CORE
0751  3010          DCA XR10
0752  1134          TAD M7200
0753  3137          DCA M7400A
0754  7300  CLEAN1,  CLA CLL
0755  3410          DCA I XR10
0756  2137          ISZ M7400A
0757  5354          JMP CLEAN1
0760  1135          TAD M7400
0761  3137          DCA M7400A
0762  1141          TAD MINUS2
0763  3142          DCA INT1
0764  1127          TAD CORE
0765  3010          DCA XR10
0766  5746          JMP I CLEAN
```

TABLE B-II - OPERATING PROCEDURE FOR PAPER-TAPE LOAD PROGRAM

The paper tape created on the IBM 1130 computer is loaded on the PDP-8/I Systems by use of program PAPT.

This program is loaded into core by the system monitor (underlined letters are the system monitor responses).

. load ↵

* IN-S: PAPT ↵

* OPT- 1 ↵

ST = ↵

†† CTRL/P

After typing CTRL/P, the computer HALTS. The following steps must be taken prior to starting the program:

A. DECTAPE

1. Put switch to local
2. Put switch to Write Lock
3. Press (↵) switch until DECTAPE system tape runs free
4. Put switch to OFF
5. Remove system tape
6. Place a Blank DECTAPE Reel on the DECTAPE.
7. Set selector switch to UNIT 3
8. Put switch to REMOTE
9. Set Write Enable Switch

B. Paper tape:

1. Place Paper Tape in Paper Tape Reader (Free)
2. Place Address $\emptyset 400_8$ in switch register
3. Press Load Address
4. Push Start
5. After DECTAPE Search is complete (Approx. 5 seconds)

   Program HALTS
6. Set paper tape switch to Start
7. Paper tape will now be loaded
8. When the paper tape has been completely read the computer will come to a logical HALT at address $\emptyset 477_8$ (the Memory Buffer register will contain 7402_8).

APPENDIX C

DATA CONVERSION AND ANALYSIS PROGRAM

INTRODUCTION

The processor program for digitizing and preparing digital library tapes of ship stresses and environmental data have been modified from that given in Reference 2. Early in the processing phase it was determined that some modifications to the program were needed in order to maintain good production schedule.

The type of data being digitized requires that the data conversion and analysis effort be operated in a Real-Time environment which is possible through the use of the Real-Time Programmable Clock. Programming has been done in assembler language (PAL-D, Programmable Assembler Language for DECtape) to take advantage of the shortened processing time, to work within the 8K word memory, and to enable the use of a single DECtape auxiliary storage unit. Processing of data was accomplished at a rate-increase factor of 25 over the recorded rate (of 0.3 inches/second) without requiring starting and stopping of the analogue playback unit.

The analogue signal comes into the A/D unit as three basic signals; the wave-induced signal, the first-mode signal, and the combined signal. Through the use of the analogue/digital multiplexer, each signal is digitized individually although the data processing is done simultaneously and continuously within the processor without the need to stop and start the playback of the analogue signal. The digitizing actually operates in terms of voltages and all processing within the computer is on the basis of voltages. This saves core storage since twice as many words would be necessary if stress units were stored. The specific program flow chart is shown in Figure C-1. All references to speeds and conversion times are in terms of real time i.e., (for a 30-minute interval). The speed-up factor of 25 is the maximum speed at which the data can be digitized and processed due to program and computer limitations. However, slower speeds can be used by changing one process instruction and will be discussed further on.

PROGRAM DESCRIPTION

The first 128 locations of the program contain parameters and the interrupt service routines. The interrupt system allows for overlap (multi-processing) and is the means by which the program is controlled.

The program interrupt can be explained as follows. When a large amount of computing is required, the program should activate an Input/Output (I/O) device (magnetic tape, clock, etc.) and then continue the main program, rather than wait for the slower I/O device to become ready to transfer data. The program interrupt facility, when enabled by the program, relieves the main program of the need for repeated flag checks by allowing the ready status of the I/O device flags to cause a program interrupt automatically. When the program interrupt occurs, program

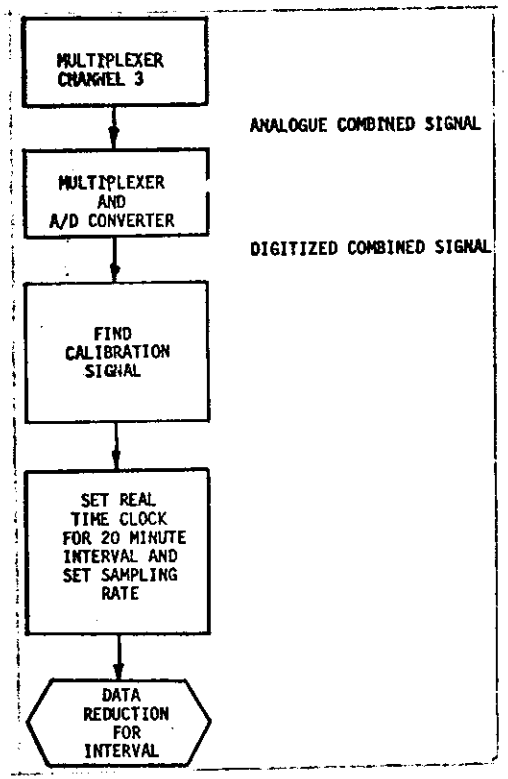


FIG.C-1 - FLOW CHART FOR DATA CONVERSION AND ANALYSIS PROGRAM.
(a) Start-Up Procedure for Interval Sampling

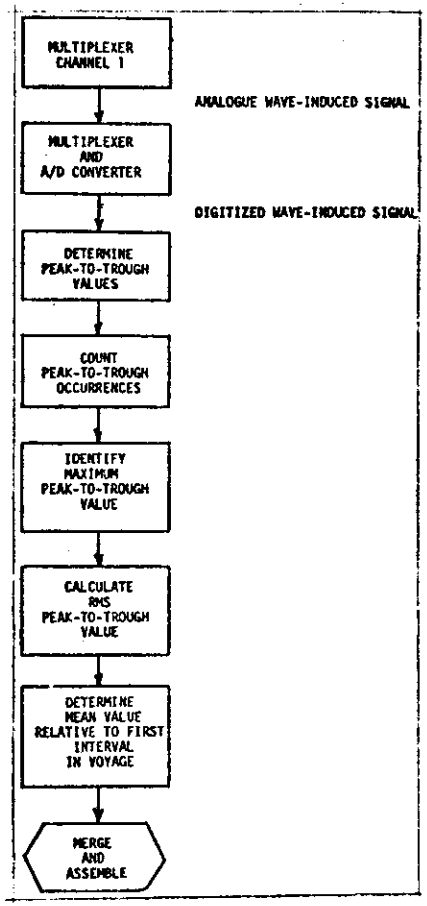


FIG. C-1 - (Continued)
(b) Wave-Induced Data Reduction for Interval

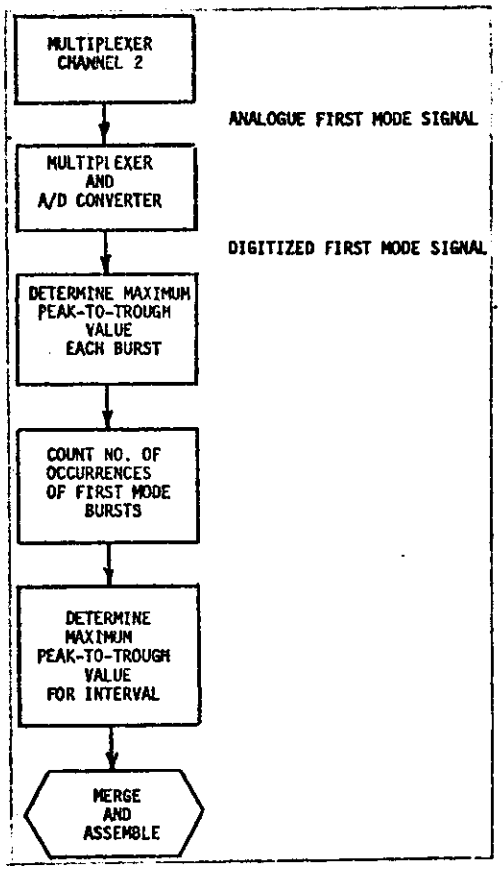


FIG. C-1 - (Continued)
(c) First-Mode Data Reduction for Interval

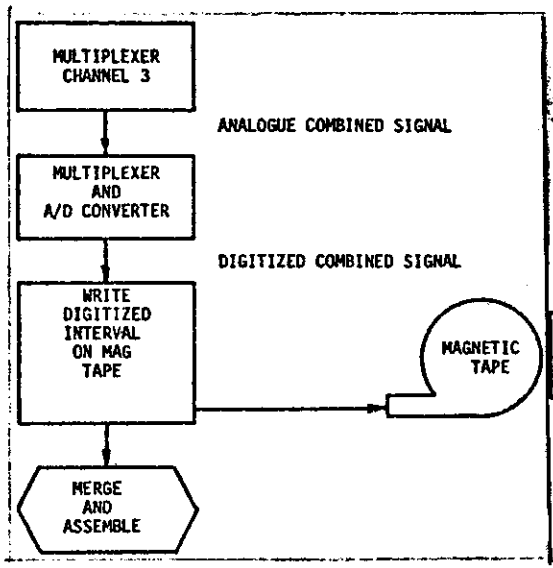


FIG. C-1 - (Continued)
(d) Combined Signal Data Handling For Interval

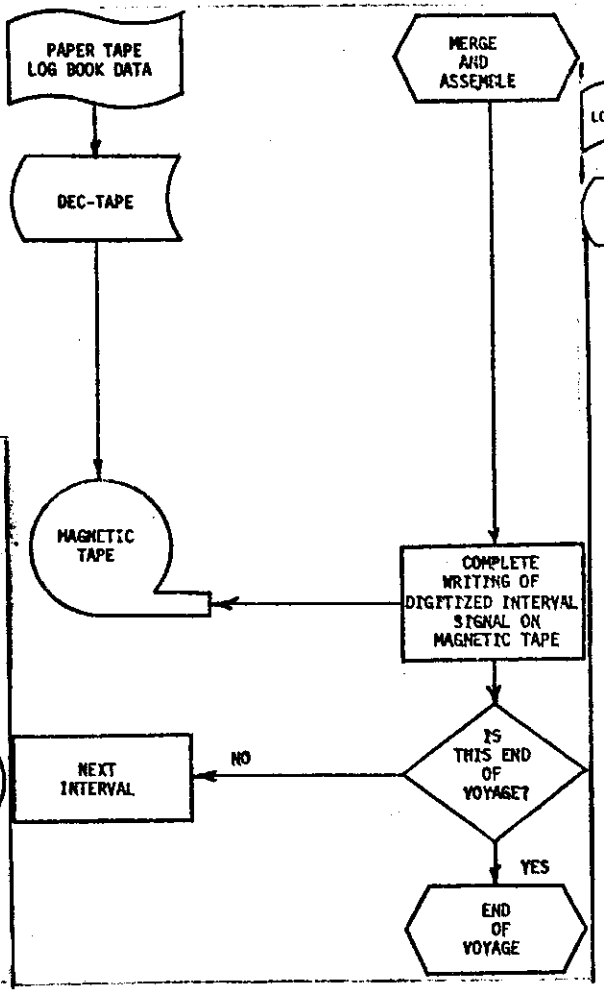


FIG. C-1 - (Continued)
(e) Interval Merge and Assemble Routine

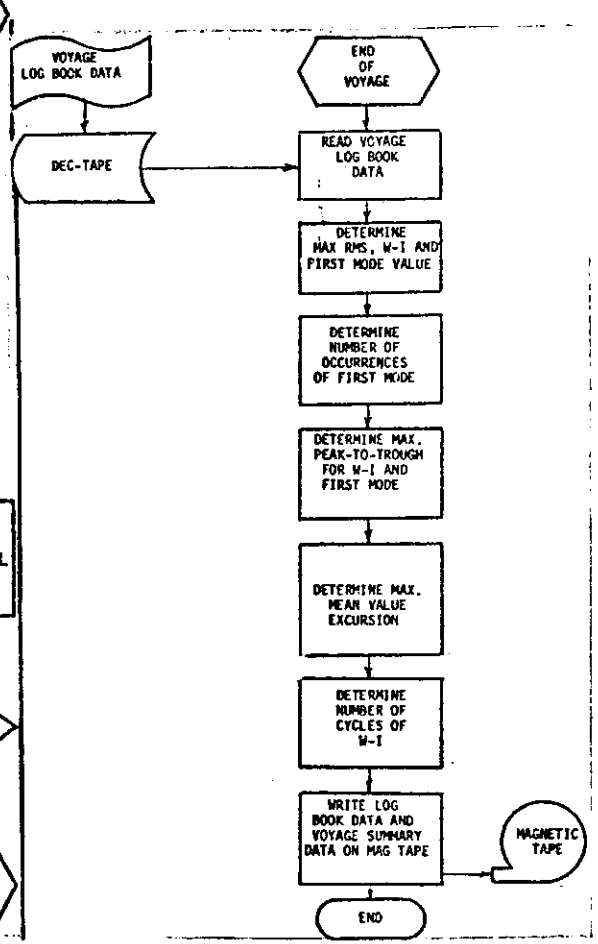


FIG. C-1 - (Concluded)
(f) End of Voyage Routine

control transfers to a subroutine that determines which device requested the interrupt and indicates an appropriate service routine.

When an interrupt occurs, the computer automatically branches to location 0000 and stores the last address location. Also the link, accumulator, and program counter registers (all parts of the Central Processor Unit registers) are stored off for later reinitialization. Also, the interrupt capability is turned off. Next, the interrupt bus is checked to see what device caused the interrupt. An interrupt can be caused by the Magnetic Tape, Real-Time Clock or DECtape. An interrupt caused by any other device will not be serviced since it should not have taken place.

After the interrupt is serviced, the program branches to the dismiss routine (DIS). This routine re-enables the interrupt, restores the link, accumulator and program counter registers and returns to the last address location prior to the interrupt. This allows for sampling at a fixed rate (by use of the Real-Time Clock) and the writing of the data onto magnetic tape while handling the arithmetic calculations needed to perform the total task.

Location 200_8 to 377_8 contain the DECtape handler and is the standard routine as supplied by Digital Equipment Corporation for reading from and writing onto the DECtape unit.

The program starts at location 0400_8 (BEGIN), the interrupt enable is turned off and the counters and data storage buffers are cleared and initialized.

The program types "DECTAPE ON 4? TOTAL NO. OF VOYAGES=" then waits for the user to respond. If the program system tape has not already been removed, it must be removed, and the proper logbook DECtape mounted before setting the selector switch to unit 4. The total number of voyages to be run (in decimal: ex. Ø1) is entered by the user. After the response, the program halts. The operator then pushes CONTINUE on the computer console. The program then searches the DECtape for the initial starting block number, positions the magnetic tape for writing, and writes the magnetic-tape label and then halts.

The header information which contains the number of intervals, number of voyages, is set up in a table which controls the running of the program. By use of the table, the program is told how many intervals there are in a voyage, which intervals are long intervals, which intervals are to be deleted (not digitized), how many passes of analog tape will be run, etc. This allows the program to run virtually unmanned, except for data checking, reel mounting, etc. By inputting in Halts into the table (for certain intervals) the operator may stop the tape at pre-planned locations. This option is especially valuable if the operator feels problems may arise at those locations due to analogue tape difficulties, etc.

A header to be used with part of the Interval Summary data is typed on the teletype. After this header is typed the program halts.

The computer and peripheral equipment (I/O devices) are now ready to handle the data conversion phase. After starting the FM analogue tape playback, the operator then pushes "CONTINUE" on the computer console and the program then branches to the Calibration (CALIB) routine.

The calibration subroutine was modified from that given in Reference 2 due to the problems which occurred once production was initiated. It was found that some of the early analogue tape did not contain a 1-minute zero calibration and some of the zero calibrations were extremely noisy or contained split signals. The program now searches for the "square-wave" calibration signal only. The combined signal (Multiplexer Channel 3) is digitized in the A/D converter and the program scans the digitized signal for the calibration signal.

When a signal is found which looks like the beginning of the calibration signal, the program tries to find 4 consecutive "square waves" within specified tolerances. (The 1-minute calibration signal is defined as a square wave, 3 seconds on, then 3 seconds off for 10 repetitions.) Due to variations in the pulse used to record the calibration signal, the actual signal may vary considerably in timing from the nominal.

The program accepts a pulse if it is on from 2.0 to 4.0 seconds and off from 2.0 to 4.0 seconds. The total on-off time allowed for each pulse is 4.5 to 6.5 seconds. In addition, when 4 consecutive signals are found, the total time for the 4 pulses must fall between 22 and 26 seconds. When this occurs, the program has found the calibration signal. This method of defining the calibration signal worked quite well during the production effort.

When the calibration signal has been found the program averages the middle ten points of both the peak and the trough for the 4 cycles (the digitizing rate is set at 10 samplings per second). The algebraic difference between the average peak and average trough readings is used to convert (voltage) to an equivalent stress. This procedure eliminates problems associated with transients or round-off created by filtering of the square wave. This, in effect, gives a value of stress for each 2.5 millivolts (i.e., the "least reading" of the A/D) of signal read.

After the first interval has been digitized, the program checks the parameter table prior to entering the calibration routine. If the next interval is a long interval (does not have a calibration signal) the program does not search for a calibration, instead it loafs for the remainder of the 30 minutes of the previous record and then starts digitizing the next interval.

During the production phase it was determined that some interval calibrations deviated sufficiently from the above description that difficulties occurred. As a consequence, only the first calibration in each pass was used to determine a stress/voltage conversion factor and this was used for all intervals in that pass.

In order to retain 4-place accuracy in the digitized data, all data is retained as voltages until it is written out on magnetic tape in stress units. Double precision is used on all arithmetic operations to retain the precision needed.

After the search for the calibration signal has been completed, the program calculates an effective mean value for the first 4 minutes of raw data (subroutine LMEAN). This value is used in defining an effective "zero crossing" in calculating peak-to-trough values. Three mean values are calculated, one for wave-induced (multiplexer channel 1) data, one for first-mode (multiplexer channel 2) data, and one for the combined signal(multiplexer channel 3). These mean-zero values

help to eliminate any zero shift due to the electronic equipment. After the calculation of the effective mean values, the program branches back to the BEGIN routine. All interrupt flags are checked and cleared. The interrupt registers are initialized for storing of accumulator, link, etc., and the interrupt bus is enabled. Real-Time Processing now begins.

Data are sampled every 100 ms (10 points per second, real time) and the programmable Real-Time Clock is set to cause an interrupt every 100 ms. This is done in the CLOCK subroutine. When the clock causes an interrupt, the computer program causes a jump to the clock-service routine (SERV). Also, a counter is incremented each time entrance is made. This allows for the 20 minutes (12,000 samples) of real-time data acquisition. The value "TLIM" in the clock routine defines the length of time between interrupts (TLIM/10000 sec.). By changing this value the rate of digitizing can be changed.

The SERV routine re-enables the clock, reads the three multiplexer points using the READ subroutine, converts the combined digital signal to an equivalent stress, loads and initializes the magnetic-tape buffer. The mean values found in the CALIB routine are algebraically added to the values read on Channel 1 (wave-induced), Channel 2 (first mode) and Channel 3 (combined signal) of the A/D converter, respectively. This process effectively shifts the average level of all data to electrical zero. The SERV routine then returns control to the DISMIS routine.

Upon entering the clock service routine (SERV) the parameter "WHERE" is checked. If it is negative, then data acquisition has exceeded data processing and the program automatically halts. This means that data is coming in faster than the arithmetic computation can handle it. The cause of this would be a "false" interrupt due to malfunction of the computer or due to the teletype or a nonprogram-defined I/O device causing an interrupt.

At the end of the data processing (i.e., calculation of peak-to-trough stresses, etc.) of each piece of data, the program goes to Subroutine NOGO. This routine is simply an endless loop. It essentially causes the computer to loaf. This uses up all spare time after each piece of data has been processed. Upon entry into this Subroutine, "WHERE" is set to zero (0). When the clock causes an interrupt and WHERE has been checked and found to be zero, the SERV routine now changes NOGO to a microsecond lag routine instead of an endless loaf routine. Upon dismissal from the SERV routine, the DISMIS routine which re-initializes the interrupt, branches to NOGO. Upon entry into NOGO, the sub-routine now causes an automatic branch to AROUND which does the data processing.

The AROUND routine modifies the NOGO routine to make it an endless routine again. The AROUND routine is composed of several subroutines which do the actual data processing.

The calculation of peak-to-trough values for wave-induced data (Subroutine IN) are based on the effective values computed in the SERV routine. There are two major problems associated with the calculation of peak-to-trough values using the computer program. The first problem is one of extreme accuracy, and is the definition of electronic zero. Since each count (2.5 milli-volts) read in the A/D converter is approximately equal to 7 psi, the computer is extremely sensitive to any type of electronic shift (zero frequency). This is different from the Probability Analyzer and the previous method of determining peak-to-trough

stresses. Due to the high degree of computer refinement and accuracy it was deemed necessary to define electronic zero as a banded zone instead of an absolute value. This "zero" zone was set at approximately $\pm$ 40 psi, which is the equivalent of $\pm$ 1 count on the Probability Analyzer. The value chosen for the zone may be changed by changing the value of "LIMSTO" in the program. The width of the zero zone should be based on the signal-to-noise ratio of the FM analogue tape.

In searching for a peak, the program checks the new digitized value and compares it with the previous value always saving the largest value. When the new value passes through "zero", the maximum positive value is stored and a switch is set. Now the program searches for the algebraically largest negative value (trough) in the same manner as it searched for the peak. When the wave-induced data again passes through "zero", the negative value is algebraically added to the positive value to give the peak-to-trough value. This value is then stored in a buffer for later use. Also, a counter (NUMPKT) is incremented which gives the number of peak-to-trough values found. This new peak-to-trough value is then compared with any previous peak-to-trough value and the maximum value is kept in MAXPKT.

The second problem is the minimum peak-to-trough value allowed. Prior to the storing of the peak-to-trough value it is compared with a minimum allowable (LIMIT) value. If it is below this value it is ignored and not used. The program can save a maximum of 500 peak-to-trough values which is well above the range found by examining previous Probability Analyzer results. After each peak-to-trough value has been found, it is squared and added to a counter for calculation of the RMS Peak-to-Trough value for the interval. This method is statistically superior to the Probability Analyzer method since each value is weighted equally.

The calculation of First-Mode Peak-to-Trough values is done in the MODAL subroutine. Again, the effective average value calculated for the first-mode data is used. A zero zone is used in the same manner as that described for wave-induced data.

Since interest is only in the apparent number of first-mode "bursts" and the maximum peak-to-trough value experienced in an interval, the logical method of analysis is slightly different than that used for the wave-induced data.

The zero band is not as critical as that used for the wave induced. However, the minimum peak-to-trough value chosen (BAND) becomes more important in the analysis. In the data reduction phase this was set at 600 psi. When this value is exceeded it initializes a switch which is used to define the start of a burst. When this occurs the program continues to calculate the peak-to-trough values searching for a maximum value and saving it. The end of a first-mode burst is defined as the time the peak-to-trough value goes below the minimum value (BAND). Each time a burst is found, the maximum peak-to-trough value for the present burst is kept and is used for the maximum peak-to-trough first cycle (MAXMOD) for the interval. This concludes the calculation of peak-to-trough values and the data processing of these values. The program then branches to NOGO to await another interrupt.

The use of the magnetic-tape handler and the merging of logbook data with the digitized data follows the first-mode determinations. The SERV routine converts the combined wave-induced and first-mode (multiplexer Channel 3) digitized signal to an equivalent stress value (in psi units), then loads and initializes the magnetic-tape buffer. The 9-track digital magnetic tape can write one character every 1.5 ms.

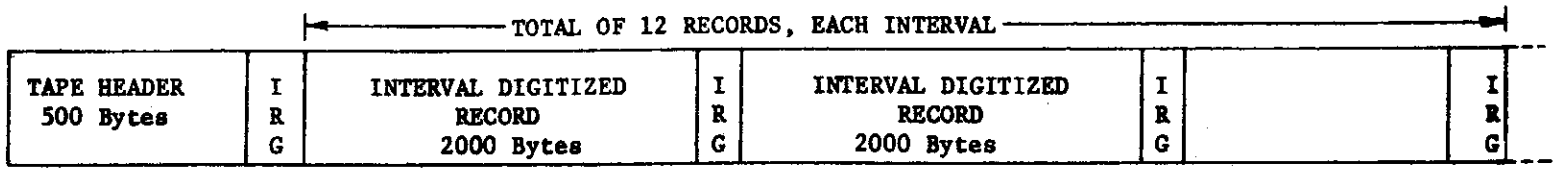
In order to write an IBM-compatible tape, capable of being read on a larger computer (i.e., IBM-360) using a language such as FORTRAN, certain rules must be followed.

Each digitized sample, when converted to stress, occupies two PDP-8/I words (24 bits). This data is converted in subroutine MGSET to a 16-bit, 2-byte word compatible with 9-track magnetic tape format (INTEGER*2). Under program interrupt, data is loaded into the tape-write buffer and the program returns to the user. When a magnetic-tape interrupt occurs, the data storage buffer is checked to see if more data needs to be written. Since data is written in blocks of 1000 (2000 bytes), the amount of time needed is less than the clock-interrupt pulse interval. However, at the end of each record (1000 digital points) an Inter-Record Gap (IRG) is generated. This takes approximately 80 ms. During this time, data are being converted and stored in the buffer. The clock SERV routine checks the condition of the magnetic-tape flag and if it is busy, continues on with the processing. If, however, the digital tape is not busy and there is data in the tape-write buffer, it reinitializes the tape-write routine (which will cause an interrupt when done) and continues on processing the data. This takes into account the times that each piece of data can be written faster than it is read and also, the times when data is coming in faster than it can be written on tape (due to an IRG, etc.).

At the end of the interval data-processing phase, the program branches to the ALDONE routine. The analogue tape continues to run and the program utilizes the remaining time (approximately 7 minutes real time) to complete writing the digitized record, and write the Interval Identification and Interval Summary data on the digital magnetic tape. This routine disables the interrupt system, writes the calibration factor (a shift factor which allows for double precision arithmetic while only retaining single-word data), number of peak-to-trough values, number of first-mode bursts, RMS stress for the wave-induced data, maximum wave-induced peak-to-trough stress, and maximum first-mode peak-to-trough stress, mean value of stress for the interval relative to the first interval of the pass, and the peak-to-trough values on DECtape. These values are written as voltages on DECtape to conserve storage and are converted to stress units when transferred to magnetic tape. Figure C-2 is a schematic of the format for the magnetic tape.

The logbook data is read from the DECtape and written on digital tape along with the Interval Summary data. After the Interval Summary data for each interval is written, Voyage Summary information such as number of wave-induced peak-to-trough values etc., for the voyage is updated on the DECtape for the voyage.

A check is made to see if this was the last interval in a pass based on information contained in the tape Header. If not, the program returns to BEGIN2 for processing another interval. If this is the end of a pass the program types out "END OF PASS X" and halts. The operator then mounts the next analogue record and pushes CONTINUE on the computer which will cause a branch to BEGIN2. If it is the end of the voyage, the program reads the voyage logbook data from DECtape and merges it with the Voyage Summary data (voyage wave-induced RMS, etc.) and writes this on digital tape. The program then types "END OF VOYAGE X" and halts. The operator will then mount a new voyage tape on the analogue device, push CONTINUE and the program branches to BEGIN1 which will reinitialize all counters for the new voyage. All voyages are treated alike within a digital magnetic tape.



TAPE HEADER
384 Bytes - Logbook Data
116 Bytes - Zero Fill

INTERVAL DATA
2000 Bytes — Data

INTERVAL I.D. AND SUMMARY
256 Bytes - Logbook Data
24 Bytes - Summary Information
2000 Bytes - Peak-to-trough (wave-induced) full word integers.

VOYAGE I.D. AND SUMMARY
376 Bytes - Logbook Data (120 blanks)
24 Bytes - Summary Information

FIG. C-2 - DATA TAPE OUTPUT FORMAT

When the last voyage to be written on the digital tape has been processed the computer writes an End-of-File Tape Mark (EOF), rewinds the digital magnetic tape, types out "END OF JOB" and halts. The operator removes the digital tape from the tape drive, removes the write ring, labels the tape and files it. The DECtape is removed from its unit, labelled and filed for later use. This completes the writing of a digital magnetic tape.

To continue processing of further voyages, all that is needed is a new digital magnetic tape, a DECtape with logbook information and the new analogue tape. The program is restarted by pressing CONTINUE on the computer console. It automatically branches back to location 400_8 for complete reinitialization of the program.

Useful during production processing was the diagnostic portion of the program which helped identify where problems occurred. This has been eliminated by the use of the teletype. During the period where the program is waiting after an interval calibration signal (or if a long interval, no calibration signal) has been found, the program prints out part of the previous interval summary information on the teletype. This gives the operator a visual display of the production sequence. This routine occupies part of the second 4K of core (see subroutine LIST). By checking the interval summary data as it is listed, the operator can identify any problems which occurred due to malfunction of the equipment and/or data. This saves valuable time in finding errors or problems when they occur, not after the total job has been processed.

Table C-I gives the program listing.

OPERATION PROCEDURE

To use the program, a DECtape which contains the logbook data with the necessary parameters must have been loaded via Program PAPT.

From the system device (DECtape UNIT 8), the program is loaded into memory by monitor. Actually, because of the size of the program, it has been written as connected programs and must be loaded in two passes. The procedure is as follows: (underlined items are the system monitor responses):

. LOAD ↵

* IN - S: LIST ↵

* OPT - 1

ST = ↵

↑↑

After typing CTRL/P, the computer HALTS. After the computer HALTS, Load Address 7600, and push START. This brings monitor back into core. When monitor has been initialized it will respond with an .. This allows the loading of the second set of programs.

. LOAD ↵

* IN SHIP ↵

* OPT - 1 ↵

ST = ↵

↑ ↑

The computer, once again, will HALT. The programs are now in memory. Place the address Ø4ØØ using the Switch Registers. Push LOAD ADDRESS, then push START.

The message

DECTAPE ON 4? TOTAL NO. OF VOYAGES= will be typed on the typewriter.

The operator must type the number of voyages in decimal (e.g., Ø4). After this is done the computer will HALT. Remove the System DECtape. Place the DECtape containing the logbook information on TCØ1 unit 4 (similar to the procedure used in PAPT).

Turn the A/D unit on and allow at least 15 minutes for it to warm up before running any data.

Place a digital magnetic tape on the magnetic-tape device. Make sure a ring (which allows the system to write on the tape) has been inserted on the back face. Instructions for threading the tape are given on the tape device.

After the tape has been threaded properly, turn the magnetic-tape drive ON. Push the LOAD FORWARD button in. Hold it in until the magnetic tape stops. The Aluminum foil (Beginning of Tape Mark) strip should be just beyond the READ/WRITE heads. Push REWIND. This will bring the Aluminum tape strip back before the READ/WRITE heads. Make sure the PAPER-TAPE SPACE PUNCH is OFF. Also, make sure the PAPER-TAPE READ has been set to FREE. The system is now ready to be used.

Press CONTINUE on the computer. The message

START OF VOYAGE 1

will be printed. Also, a Header label for the Interval Summary which appears at the end of each interval will be typed on the typewriter.

The program then HALTS.

By pushing CONTINUE, the computer will now enter the digitizing mode. Before Pushing CONTINUE, the FM tape should be started.

At the end of every interval an Interval Summary is typed. This serves as a check on the digitizing progress and a means to see that the proper merging of logbook and digital data is taking place.

At the end of every Pass, and/or Voyage, the computer will HALT. This allows the operator to mount a new FM tape or do whatever is necessary prior to starting a new pass.

By pushing CONTINUE, the program will reenter the digitizing phase.

At the end of the last Voyage, the Magnetic tape will be rewound and a message

END OF JOB

will be typed.

Before the program begins its search for the start of an interval, a check is made in the Header information for special instructions (long interval, delete or halt.

If a halt has been programmed in the program through instructions in the Header, the computer will stop. If the operator wished to skip over a number of invalid intervals, the analogue tape would be advanced to just before the next interval to be digitized and the START button pushed on the computer console.

Beside skipping over invalid portions of analogue tape, the HALT instruction must be used when the number of long and/or deletes exceeds the capacity of 40 instructions. The analogue tape must be stopped when the computer stops and the additional instructions entered manually into the computer via the computer console. Each instruction consists of two 12-bit words. The first word identifies voyage number and pass number (e.g., voyage 2, pass 3 would be 0023). The second word defines the three instructions (HALT = 1000, LONG = 2000, DELETE = 4000) plus the interval number in octal, (e.g., to delete interval 25 (octal 31) of voyage 2, pass 3, the two-word instruction would be 0023 4031). These instructions are placed in fields 7025_8 through 7144_8. Care must be taken to record the address in the program counter so that the program can be restarted at the proper location after adding the new instructions.

TABLE C-I - PROGRAM LISTING FOR DATA CONVERSION AND ANALYSIS PROGRAM

```
.PIP
*OPT-A

*OUT-S:S2
*
*IN-S:JAC2,S:TC02
*
*!!
*OPT-A

*OUT-S:S3
*
*IN-S:ALD1,S:RRD2,S:MODA,S:DVI6,S:MAG1
*
*
*
*
*!!!!!
*OPT-A

*OUT-S:S4
*
*IN-S:MAGA,S:CAL1,S:KAL1,S:KKB3,S:RMS
*
*
*
*
*!!!!!
*OPT-A

*OUT-S:S5
*
*IN-S:INT1,S:MLYP,S:VOY1,S:CON1,S:ADT2
*
*
*
*
*!!!!!
*OPT-A

*OUT-S:S6
*
*IN-S:KYB1,S:NEND
*
*!!
*OPT-
.PALJ
*OUT-S:SHIP
*
*IN-S:S2,S:S3,S:S4,S:S5,S:S6
*
*
*
*
*
*OPT-T
```

A	0100	CAM	7621	DIFF1	4367	F7775	5762	LMEAN	2426	MINTWO	2140
AC	0066	CAND1	4270	DINT	0252	GET1	5533	LMEAN1	2540	MODAL	1200
ACA	4142	CAND2	4204	DIS	0166	GET2	5335	LMEAN2	2541	MODAL1	1220
ACR	4143	CCEC	6136	DISMIS	2643	GET3	4714	LMEAN3	2452	MODE11	0056
ACC	4144	CCFF	6132	DIVTWO	0165	GET4	5007	LMEAN4	2543	MODE12	0072
ACCEPT	5306	CCLOK	2227	DMAG	1750	GOGO	0713	LMEAN5	2544	MODIN	1405
ACEPT1	5643	CCLOK1	2254	DMEAN	5400	GOGOM	1204	LMEAN6	2550	MODPOS	1257
ACEPT2	5655	CCLOK2	2211	DNCB	0265	HALT	4524	LMEAN7	2551	MOD1	1400
ADCC	6541	CECI	6137	DONE	0052	HALT1	4565	LNKDIV	0142	MOD2	1401
ADCV	6532	CEIL	6137	DRET	0227	HALT3	5125	LOAF	2412	MOD3	1402
ADRB	6534	CHAL1	0124	DR127	0245	HALT4	5115	LOAFA	2314	MOD4	1403
ADSC	6542	CHAN	0064	DR128	0237	HEADER	0175	LOAFBD	5136	MOD5	1404
ADSF	6531	CHAN3	2142	DR128C	0262	HEDCNT	4546	LOAFB1	5062	MOD6	1433
AGANB	5073	CHAR1	5600	DSERR	0277	HICORE	5720	LOAFER	2311	MPAS	5346
AGAN1A	5052	CHAR2	5347	DSERRH	2702	HIDIV	0141	LOAF1	2416	MPZ	4072
AGN	3024	CHAR3	4727	DSTOP	0363	HOLD	0065	LOAF2	2324	MP1	4074
ALDONE	2724	CHAR4	5016	DTBLK	0275	HOLD1	4544	LOCAF	5757	MP2	4052
ALDON1	3115	CHAR5	5674	DTEM	0271	HOLD2	4545	LOCTAP	0147	MQA	7501
ALDON2	3200	CHECKE	5077	DTEMP	0232	HOLD3	5144	LOCTP1	4365	MQL	7421
ALDON3	3231	CHNG	2231	DTENX	0232	H1	3161	LOC1	5142	MULPSI	4200
ALDON5	3241	CHNG3	2250	DTER	2665	H2	3162	LONGA	4605	MULT	4030
AOK	2224	CLEAN1	5430	DTERR	0130	IGS	6705	LONGER	5130	MULVAL	4264
AROUND	0700	CLENUP	3624	DTFIND	0357	IMC	6704	LONG1	4514	MUY	7405
ASR	7415	CLEN1	5440	DTLA	6766	IN	1000	LONG2	4600	MUOY	5544
A1	5761	CLEN2	5502	DTS1	0304	INMOD	1236	LONG5	4562	MY	5232
A2	1166	CLOCK	0600	DTS2	0310	INTCHK	5041	LONG6	4566	M0017	4543
B	0101	CLOK	2402	DTS3A	0323	INTNO	4622	LONG7	4455	M0400	1752
BACKFR	0532	COMBIN	2642	DTURN	0347	INTNOF	5760	LONG8	4540	M0777	5145
BAND	0122	CORADD	4307	DTURNX	0257	INTN1	5143	LOWVOL	2274	M08	2723
BAND1	1506	CORAD1	3274	DUBADD	1141	INT1	0536	LOWVO2	4233	M1	0144
BCN	2712	CORAD2	3277	DUF	0230	IRD	6707	LREAD	1616	M10A	4703
BEGIN	0401	CORAD3	4364	DVD	1563	IRG	0075	LSR	7417	M10AA	4724
BEGIN1	0431	COUNT	5231	DVI	7407	IRS	6701	LWRT	1604	M1000	4564
BEGIN2	0440	CRCA	6134	DV2	1522	IR1	0010	MAGCNT	0131	M1000A	5133
BEGIN3	0443	CRLF	5277	DWAIT	0371	ISR	6702	MAGIRG	1753	M1000B	5134
BLCK	0062	CSCF	6133	DWC	0264	IVR	6706	MAGIRZ	1754	M18A	4553
BLOK	0063	C1	5545	D2	0370	IWS	6703	MAGUP	1704	M18AA	4554
BNO	2713	C1000	2155	D20	0270	KEEP	1136	MAGUPE	0054	M18B	5131
BTHNEG	1037	C2	5673	D200	0276	KEEPER	1122	MAGUP1	1642	M18C	5132
BTHPOS	1012	C2000	0164	D30	0266	KEEPM	1336	MAG1	1743	M2	1755
BUFFPT	0074	DATAF	0176	D400	0273	K212	5226	MAG2	1730	M2A	4320
BUFP	4321	DAT1	5744	D614	0274	K215	5227	MASK	0106	M20A	2400
BUF1	2710	DBLK	0272	D7000	0333	L	0067	MAXMOD	0121	M20B	2401
CALIB	2000	DCAA	0263	D7600	0335	LABEL	4322	MAXPKT	0116	M2000	4563
CALIB2	2075	DCINT	0267	ENDED	0707	LABEL1	4627	MAX1	0153	M260	5546
CALIB3	2106	DCRET	0236	END1	4075	LARGER	1126	MAX2	0154	M3	2322
CALIB4	2022	DECRED	4304	END3A	4127	LENGTH	1354	MCOM	0053	M3A	2323
CALIB5	2033	DECTAP	2703	EOF	3751	LFR	1636	MEANMD	2545	M330	2536
CALIB6	2137	DEC1	0540	EOFEOF	5015	LFW	1634	MEANSG	3650	M330A	2537
CALIB7	2071	DELETE	4560	EOFTYP	5000	LIM	0152	MEAN1	0145	M340	2424
CALIB8	2017	DER	0351	FIRST	0102	LIMIT	1046	MEAN2	0146	M340A	2425
CALIB9	2010	DGET	0232	FOUR	0126	LIMSTO	0752	MGSET	1656	M3600	1641
CALSIG	2200	DGR	0212	FPKBUF	5756	LIMST1	1226	MGWRTE	1630	M3720	3301
CALVAL	4221	DIFF	4366	F08	4126	LISN	5317	MGWRT1	4214	M4	0137
				F4	0125	LKMULT	0143	MINFOR	2141	M4A	4550

TABLE C-I - (Continued)

M4AA	4551	OK	0643	SAVE3	3276	TESTM	1227	WRI	3400		
M4AAA	4552	OKOK	4255	SAVPOS	0107	TESTR	1751	WRPK	3430		
M40	4556	OK1	3054	SCA	7441	THOUS1	0150	WRPK1	3426		
M40A	4557	ONE	0060	SEARCH	2671	THOUS2	0151	WRTSUM	3651		
M4000	4561	OPPMOD	1263	SEROUT	2677	THOUS3	3756	WRT4	3513		
M476	4361	OPPSGN	1026	SERV	0636	THOUS4	3757	WRT5	3541		
M476A	4362	OPS	1104	SERVE	0051	TIME	2156	WRT7	3600		
M5A	5514	OUT	1165	SETUP1	4400	TLIM	0611	WRT8	3611		
M500	0537	OUTER	0676	SETUP2	4000	TRY1	4232	WR1	4343		
M6	5230	OUTERR	0677	SET1	0541	TWELVE	3755	WR2	0046		
M7	3157	OUTPT2	5730	SET2	4623	TWENT	0163	WR3	0047		
M7001	4625	OUT1	2721	SET3	4624	TYPE	5515	WR4	0050		
M7001A	5513	OUT2	1470	SHIFT	3752	T0017	0135	WR5	0167		
M7002	4626	OUT3	1424	SHIFT2	4244	T7400	0136	WR6	4357		
M7002A	5512	OUT4	1466	SHIFT3	4277	VA	4026	W128	0206		
M7025	5135	OUT5	1471	SHIFT4	4206	VALUE1	3750	W128Y	2722		
M7200	4370	PASHED	5327	SML	7413	VB	4027	XI	3164		
M7322	4547	PASNO	4621	SIGN	1561	VH	3746	XIFIX	3163		
M7400	1637	PCREG	0070	SIGNAL	2157	VISA	3032	XIPLUS	3165		
M7444	3551	PKBUF	0110	SIGNA2	2160	VISB	3033	YR17	0017		
M7444A	3552	PKBUFT	0112	SIGNA3	2161	VISOR1	2275	ZERO	0113		
M7600	1640	PKBUF1	0111	SIGNA4	2162	VISOR2	4234	Z1	3100		
M7610	3743	PLIER	0140	SIGNMT	0134	VIS1	3010	Z2	3110		
M7610A	3744	PLY1	0172	SLEEP	0123	VIS2	3011	Z4	3141		
M7772	3300	PLY3	0173	SLMEAN	2542	VIS5	3017	Z5	3151		
M7773	3760	PLY4	0174	SMA1	2310	VIS6	3020				
M7776	3166	POINT1	0155	SPAC	2307	VL	3747				
NEG	1554	POINT2	0156	SPA5	2304	VMAXPH	3642				
NEGMOD	1274	POS	1022	SPDIV	1507	VMAXPL	3643				
NEGSAV	1047	POSMOD	1247	SRD	3754	VMEAN1	3646				
NEGSGN	1125	POSONE	3273	START	5200	VMEAN2	3647				
NEG1	1065	PUTBCK	0535	START3	5427	VMODH	3644				
NEWDC1	5630	P1	2143	START5	5420	VMODL	3645				
NEWDEC	5621	P10	2154	STATUS	1635	VNOMDH	3636				
NEWPG2	2600	P2	2144	SUMPKH	0076	VNOMDL	3637				
NEWPG4	3000	P3	2145	SUMPKL	0077	VNOPKH	3634				
NEXT	5217	P4	2146	SUMPLL	1167	VNOPKL	3635				
NGSAVM	1303	P5	2147	SWT	0103	VOYEND	4705				
NMI	7411	P6	2150	SWT1	0104	VOYHED	5524				
NOBLKS	4311	P7	2151	SWT2	0105	VOYNO	4620				
NOBLK1	3275	P8	2152	S1	2546	VOY3	5671				
NOBLK2	4363	P9	2153	S2	2547	VPAS	4726				
NOGO	0743	RBLOK	0170	S3	2440	VSUMPH	3640				
NOGOIN	5146	RBLOK1	0171	S4	3160	VSUMPL	3641				
NOINT	3272	READ	0612	S5	2552	VYCNT	4542				
NOKEEP	1343	READ1	4312	S6	2555	VY1	5315				
NOPEAK	3052	REG14	0014	TAPBUF	0012	VY2	5316				
NOPUT	0674	REG15	0015	TAPE	0127	V1	3745				
NOTIN	4534	REWIND	1600	TAPLOC	0013	WAR	3753				
NOVOY	4555	RMSGN	1562	TAP1	0132	WAVE1	0055				
NUM	5672	R128	0200	TAP2	0133	WAVE2	0071				
NUMMOD	0120	R128Y	4317	TEN	0157	WHERE	0073				
NUMPK	0114	SAMP	0057	TENPTS	0162	WRAGN	3673				
NUMPKT	0115	SAMP1	0061	TEN1	0160	WRANG1	3672				
N19	0161	SAPMOD	0117	TEST	0734	WRA1	3720				

TABLE C-I - (Continued)

```
                                / NAVY CONTRACT NO.-NSSC-N00024-69-C-5161
                                /UPDATED VERSION FEB 15, 1971
                                /PROJECT 1149-SSC DIGITIZING SHIP FM
                                /ANALOG TAPES PROGRAM
                                / MULTIPLEXER CHANNEL ASSIGNMENT
                                /CHANNEL 1 WAVE INDUCED SIGNAL
                                /CHANNEL 2 1ST MODE SIGNAL
                                /CHANNEL 3 COMBINED SIGNAL WITH
                                / CALIBRATION SIGNAL
          *000
          / FIRST INSTRUCTION AFTER AN INTERRUPT
0000  0000      0000
0001  3466      DCA I AC
0002  7010      RAR
0003  3467      DCA I L
0004  1000      TAD 0
0005  3470      DCA I PCREG
0006  7300      CLA CLL
0007  5020      JMP 0020
                *0020
0020  2066      ISZ AC
0021  2067      ISZ L
0022  2070      ISZ PCREG
0023  6771      DTSF              /IS INTERRUPT CAUSED BY
0024  7410      SKP               /TC01 DECTAPE UNIT 4?
0025  5453      JMP I MCOM        /SERVICE DECTAPE
0026  6133      CSCF              /IS INTERRUPT CAUSED BY
0027  7410      SKP               /REAL TIME CLOCK?
0030  5451      JMP I SERVE       /SERVICE REAL TIME CLOCK
0031  6041      TSF
0032  7410      SKP
0033  7402      HLT
                /
                /INTERRUPT ROUTINE TO HANDLE MAG TAPE
                /
0034  6703      IWS               /SKIP ON WRITE DONE
0035  7410      SKP
0036  5454      JMP I MAGUPE
0037  6701      IRS               /SKIP ON READ DONE
0040  7410      SKP
0041  5566      JMP I DIS
0042  6705      IGS     /SKIP ON GAP DETECT
0043  7410      SKP
0044  5454      JMP I MAGUPE
0045  5566      JMP I DIS
0046  0000  WR2,     0000
0047  0000  WR3,     0000
0050  0000  WR4,     0000
0051  0636  SERVE,   SERV
0052  0371  DONE,    DWAIT
0053  0000  MCOM,    000     /DEFINED BY USER, INITIALIZED BY TAPE ROUTIN
0054  1704  MAGUPE,  MAGUP
0055  0000  WAVE1,   0000
0056  0000  MODE11,  0000
```

TABLE C-I - (Continued)

```
0057  7764  SAMP,   7764
0060  7777  ONE,    7777
0061  0000  SAMP1,  0000
0062  0014  BLCK,   0014
0063  0000  BLOK,   0000
0064  0001  CHAN,   0001
0065  0000  HOLD,   0
0066  0000  AC,     0000
0067  0000  L,      0000
0070  0000  PCREG,  0000
0071  0000  WAVE2,  0
0072  0000  MODE12, 0
0073  0000  WHERE,  0
0074  0000  BUFFPT, 0
0075  1000  IRG,    1000
0076  0000  SUMPKH, 0000    /DOUBLE PRECISION BUFFER FOR
0077  0000  SUMPKL, 0000    /SUM OF PEAK TO TROUGH VALUES
0100  0000  A,      0000
0101  0000  B,      0000
0102  0000  FIRST,  0000
0103  7776  SWT,    -2
0104  0000  SWT1,   0000
0105  0000  SWT2,   0000
0106  4000  MASK,   4000
0107  0000  SAVPOS, 0000
0110  6015  PKBUF,  6015
0111  5777  PKBUF1, 5777
0112  0000  PKBUFT, 0000    /PEAK TO TROUGH BUFFER STORAGE
0113  0000  ZERO,   0000
0114  7000  NUMPK,  7000
0115  0000  NUMPKT, 0000    /NO. OF OCCURENCES OF PEAK TO TROUGHS
0116  0000  MAXPKT, 0000    /MAX PEAK TO TROUGH VALUE
0117  0000  SAPMOD, 0000
0120  0000  NUMMOD, 0000    /NO OF 1ST MODE OCCURENCES
0121  0000  MAXMOD, 0000    /MAX 1ST MODE BURST VALUE
0122  0120  BAND,   0120    /MIN ALLOWABLE 1ST MODE BURST
0123  0002  SLEEP,  0000
0124  7776  CHAL1,  7776    /-2
0125  7773  F4,     7773
0126  0000  FOUR,   0000
0127  0000  TAPE,   0000
0130  2665  DTERR,  DTER
0131  0000  MAGCNT, 0000
            TAPBUF=0012
            TAPLOC=0013
                   REG14=0014
                   REG15=0015
0132  0000  TAP1,   0000    /FOR EBCDIC
0133  0000  TAP2,   0000
0134  4000  SIGNMT, 4000    /COMPATIBLE
0135  0017  T0017,  0017
0136  7400  T7400,  7400    /IBM 360
0137  0004  MA,     0004    /TAPE SETUP
                   /
                   /FOR EAE ELEMENT
```

TABLE C-I - (Continued)

```
0140  0000  PLIER,  0000
0141  0000  HIDIV,  0000
0142  1507  LNKDIV, SPDIV   /LOC OF DIVIDE ROUTINE
0143  4030  LKMULT, MULT    /LOC OF MULT ROUTINE
0144  0000  MI,     0
0145  0000  MEAN1,  0000
0146  0000  MEAN2,  0000
0147  7177  LOCTAP, 7177
0150  4704  THOUS1, 4704
0151  0000  THOUS2, 0000
0152  0007  LIM,    0007
0153  0000  MAX1,   0000
0154  0000  MAX2,   0000
0155  0000  POINT1, 0000
0156  0000  POINT2, 0000
0157  7634  TEN,    7634
0160  0000  TEN1,   0000
0161  7776  N19,    7776
0162  7766  TENPTS, 7766
0163  7754  TWENT,  7754
0164  2000  C2000,  2000
0165  0000  DIVTWO, 0000
0166  2643  DIS,    DISMIS
0167  3634  WR5,    VNOPKH
0170  0011  RBLOK,  0011
0171  0000  RBLOK1, 0000
0172  0000  PLY1,   0000
0173  0000  PLY3,   0000
0174  0000  PLY4,   0000
0175  0200  HEADER, 0200
0176  1000  DATAF,  1000
            *200
```

TABLE C-I - (Continued)

```
                        /TC01 SUB-ROUTINES      REV. 7/67
                DTXA=6764       /XOR AC TO STATUS A
                DTRB=6772       /READ STATUS B
                DTCA=6762       /CLEAR STATUS A
                DTRA=6761       /READ STATUS A
                DTLA=6766       /LOAD STATUS A (CLEAR AND XOR)
                DTLB=6774       /LOAD STATUS B
                DTSF=6771       /SKIP ON TC01 FLAGS
0200  0000  R128, 0             /READ 128 WORDS
0201  4371        JMS DWAIT     /WAIT IF MOTION IS ON
0202  1200        TAD R128
0203  3206        DCA W128
0204  7201        CLA IAC       /SET TO WRITE
0205  5210        JMP DGR-2
0206  0000  W128, 0             /WRITE 128 WORDS
0207  4371        JMS DWAIT     /WAIT IF MOTION IS ON
0210  1262        TAD DR128C    /DR128
0211  3227        DCA DRET      /READ WRITE RETURN AFTER SEARCH
0212  7240  DGR,  CLA CMA
0213  4232        JMS DGET
0214  3200        DCA R128      /FIRST CORE LOCATION-1 OF TRANSFER
0215  4232        JMS DGET
0216  3230        DCA DUF       /UNIT AND FIELD
0217  4232        JMS DGET
0220  3265        DCA DNCB      / #NUMBER OF BLOCKS TO BE TRANSFERRE
0221  1236        TAD DCRET
0222  3277        DCA DSERR
0223  3263        DCA DSTOP     /DON'T STOP TRANSPORT AFTER SEARCH
0224  4232        JMS DGET      /GET BLOCK NO.
0225  3271        DCA DTEM      /AND STORE
0226  5304        JMP DTS1      /INITIATE SEARCH
0227  0000  DRET, 0             /DR128 IF WRITE,  OR DR128+1 IF REAL
0230  0000  DUF,  0             /UNIT AND FIELD
0231  5606        JMP I W128
                DTEMP,
                DTEMX,
0232  0000  DGET, 0             /PICK UP ARGUMENTS
0233  1606        TAD I W128
0234  2206        ISZ W128
0235  5632        JMP I DGET
0236  0227  DCRET, DRET
0237  1270  DR128, TAD D20       /WRITE (NOT READ),(40-20)
0240  1266        TAD D30       /READ,CANCEL SEARCH (20+10)
0241  6764        DTXA          /SET FUNCTION
0242  1200        TAD R128      /1ST CORE LOC.-1 OF TRANSFER
0243  3663        DCA I DCAA    /TO 7755(CA)
0244  2053        ISZ MCOM      /POINT INTERRUPT RETURN TO DATA
```

TABLE C-I - (Continued)

```
0245  6764  DR127, DTXA         /SEND READ OR WRITE
0246  1335        TAD D7600     /SET WORD COUNT FOR 1 PAGE
0247  3664        DCA I DWC     /-128 TO 7754 (WC)
0250  5566        JMP I DIS     /EXIT
0251  5323        JMP DTS3A
0252  6772  DINT, DTRB
0253  7710        SPA CLA
0254  5351        JMP DER       /ERROR FLAG
0255  2265        ISZ DNCB      /COUNT BLOCKS
0256  5245        JMP DR127     /CONTINUE OPERATION
0257  1276  DTURNX, TAD D200     /COMPLEMENT MOTION AND DIRECTION
0260  1273        TAD D400
0261  5245        JMP DR127
0262  0237  DR128C, DR128
0263  7755  DCAA, 7755          /POINTER TO CURRENT ADDRESS
0264  7754  DWC,  7754          /POINTER TO WORD COUNT
0265  0000  DNCB, 0
0266  0030  D30,  30
0267  0251  DCINT, DINT-1
0270  0020  D20,  20
0271  0030  DTEM, 0             /HOLDS REQUIRED BLK NO.
0272  0275  DBLK, DTBLK
0273  0400  D400, 400           /CHANGE DIRECTION
0274  0614  D614, 614           /REVERSE, GO, SEARCH INTERRUPT ENABL
0275  0000  DTBLK, 0            /BLOCK NUMBER DEPOSITED HERE BY CONT
0276  0200  D200, 200           /CHANGE STOP/GO
0277  0000  DSERR, 0
0300  3271        DCA DTEM      /STORE BLOCK NO
0301  4371        JMS DWAIT
0302  1257        TAD DTURNX
0303  3263        DCA DSTOP     /STOP TRANSPORT AFTER SEARCH
0304  1272  DTS1, TAD DBLK      /DTBLK TO 7755 (CA)
0305  3663        DCA I DCAA
0306  1267        TAD DCINT     /DINT-1
0307  3053        DCA MCOM      /INTERRUPT RETURN
0310  7201  DTS2, CLA IAC
0311  1277        TAD DSERR     /DRET
0312  3232        DCA DTEMP
0313  1632        TAD I DTEMP
0314  0333        AND D7600     /PICK UP UNIT NUMBER
0315  1274        TAD D614      /SET TO SEARCH,NORMAL,REVERSE
0316  6766        DTLA          /LOAD STATUS A
0317  6774        DTLB          /FIELD 0
0320  2232        ISZ DTEMP
0321  6001        ION           /ENABLE INTERRUPT
0322  5632        JMP I DTEMP   /DUF+1, RETURN TO USER
```

TABLE C-I - (Continued)

```
0323  6772  DTS3A,  DTRB            /READ STATUS B
0324  7006          RTL             /LOOK AT BIT 2
0325  7710          SPA CLA         /END ZONE?
0326  5257          JMP DTURNX      /YES (MOTION BIT=0), TURN
0327  6772          DTRB
0330  7710          SPA CLA
0331  5351          JMP DER         /ERROR FLAG BIT 0=1
0332  6761          DTRA
0333  7006  D7000,  RTL
0334  7006          RTL             /FOR-REV STATUS (BIT 3) IN LINK
0335  7600  D7600,  7600            /GROUP 2 CLA
0336  1275          TAD DTBLK
0337  7041          CMA IAC
0340  1271          TAD DTEM        /LINK COMP. IF REQUIRED BLK NO.
0341  7450          SNA             /IS BIGGER I.E. MUST GO FORWARD
0342  5357          JMP DTFIND      /FOUND BLOCK CHECK DIRECTION
0343  7041          CMA IAC
0344  7420          SNL
0345  7001          IAC             /GO 2 MORE BLOCKS BEFORE TURNING
0346  7620          SNL CLA
0347  1273  DTURN,  TAD D400        /TURN IF HERE
0350  5245          JMP DR127       /XOR TO A STATUS AND DISMIS
0351  6761  DER,    DTRA            /ERROR ROUTINE, READ STATUS A
0352  0276          AND D200        /STOP TAPE IF RUNNING,  I.E. SET
0353  1370          TAD D2          /DON'T CLEAR ERRORS     BIT 10=1
0354  6764          DTXA
0355  6772          DTRB            /ERROR STATUS B
0356  5530          JMP I DTERR
0357  7620  DTFIND, SNL CLA         /TEST DIRECTION
0360  5245          JMP DR127       /DONT TURN YET, STILL IN REVERSE
0361  1677          TAD I DSERH     /DRET, GET COMPLETION RETURN
0362  3277          DCA DSERH       /SINCE MOTION IS FORWARD
0363  0000  DSTOP,  0               /EITHER 0 (NOP) OR TAD D200 (STOP)
0364  6764          DTXA            /CLEAR FLAG
0365  1230          TAD DUF
0366  6774          DTLB            /SET MEMORY FIELD
0367  5677          JMP I DSERH     /EXIT TO COMPLETION RETURN
0370  0002  D2,     2
0371  0000  DWAIT,  0               /WAIT FOR NO MOTION
0372  6761          DTRA            /IOR STATUS A
0373  6761          DTRA            /AGAIN, IN CASE MOTION BIT
0374  0276          AND D200        /WAS 0 DUE TO END ZONE
0375  7640          SZA CLA
0376  5372          JMP .-4
0377  5771          JMP I DWAIT
              PAUSE=400
                    CCFF=6132       /REAL
                    CCEC=6136       /TIME
                    CECI=6137       /CLOCK
                    CSCF=6133       /SYMBOL
                    CRCA=6134       /TABLE
                    CEIL=6137
                    ADCC=6541       /ANALOG
```

TABLE C-I - (Continued)

```
                    ADSC=6542       / TO
                    ADSF=6531       /DIGITAL
                    ADCV=6532       /SYMBOL
                    ADRB=6534       /TABLE
0400  6002          IOF
0401  7300  BEGIN,  CLA CLL
0402  1170          TAD RBLOK
0403  3171          DCA RBLOK1
0404  1062          TAD BLCK
0405  3063          DCA BLOK
0406  3340          DCA DEC1
0407  3777          DCA VOYNO
0410  1341          TAD SET1
0411  3776          DCA SET2
0412  4775          JMS START
0413  7402          HLT     /ALLOW TIME TO MOUNT REEL ON UNIT 4
0414  6042          TCF
0415  4774          JMS SEARCH
0416  6032          KCC
0417  6132          CCFF
0420  4773          JMS LWRT
0421  6703          IWS
0422  5221          JMP .-1         /LOAD FORWARD READY TO WRITE
0423  4772          JMS LABEL
0424  7300          CLA CLL
0425  3771          DCA VYCNT
0426  4770          JMS SETUP1
0427  1060          TAD ONE
0430  3340          DCA DEC1
0431  4767  BEGIN1, JMS SETUP2
0432  4766          JMS VOYHED
0433  6212          CIF+10
0434  4575          JMS I HEADER
0435  6201          CDF+00
0436  6202          CIF+00
0437  7402          HLT
0440  7300  BEGIN2, CLA CLL
0441  4765          JMS INTCHK
0442  4764          JMS CALIB
0443  7300  BEGIN3, CLA CLL
0444  1763          TAD ACA
0445  3066          DCA AC
0446  1762          TAD ACB
0447  3067          DCA L
0450  1761          TAD ACC
0451  3070          DCA PCREG
0452  1060          TAD ONE
0453  3102          DCA FIRST
0454  1103          TAD SWT
0455  3104          DCA SWT1
0456  1103          TAD SWT
0457  3105          DCA SWT2
0460  1060          TAD ONE
0461  3760          DCA COUNT
0462  3757          DCA MOD1
```

TABLE C-I - (Continued)

```
0463  3756          DCA MOD2
0464  3755          DCA MOD3
0465  3754          DCA MOD4
0466  3753          DCA MOD5
0467  3100          DCA A
0470  3101          DCA B
0471  3055          DCA WAVE1
0472  3071          DCA WAVE2
0473  3056          DCA MODE11
0474  3072          DCA MODE12
0475  3074          DCA BUFFPT
0476  3107          DCA SAVPOS
0477  3117          DCA SAPMOD
0500  3077          DCA SUMPKL
0501  3076          DCA SUMPKH
0502  3073          DCA WHERE
0503  3116          DCA MAXPKT
0504  3121          DCA MAXMOD
0505  3120          DCA NUMMOD
0506  3752          DCA NEGSGN
0507  1751          TAD OUTERR
0510  3750          DCA OUTER
0511  1147          TAD LOCTAP
0512  3012          DCA TAPBUF
0513  1147          TAD LOCTAP
0514  3013          DCA TAPLOC
0515  1113          TAD ZERO
0516  3131          DCA MAGCNT
0517  1747          TAD MAGIRG
0520  3746          DCA MAGIRZ
0521  1110          TAD PKBUF
0522  3112          DCA PKBUFT
0523  1057          TAD SAMP        /TOTAL SIZE OF SAMPLE=12032 PT.
0524  3061          DCA SAMP1
0525  1114          TAD NUMPK
0526  3115          DCA NUMPKT
0527  3745          DCA DMAG
0530  3744          DCA TESTR
0531  5743          JMP SERV
0532  1335  BACKER, TAD PUTBCK
0533  3750          DCA OUTER
0534  5742          JMP AROUND
0535  5566  PUTBCK, JMP I DIS
0536  0000  INT1,   0000
0537  7014  M500,   7014
0540  0000  DEC1,   0000
0541  0011  SET1,   0011
                    /
                    / END OF SUBROUTINE ALD
                    /
0542  0700
0543  0636
0544  1751
0545  1750
0546  1754
0547  1763
0550  0676
0551  0677
0552  1125
0553  1404
0554  1403
0555  1402
0556  1401
0557  1400
0560  5231
0561  4144
0562  4143
0563  4142
0564  2000
0565  5041
0566  5524
0567  4000
0570  4400
0571  4542
0572  4322
0573  1604
0574  2671
0575  5200
0576  4623
0577  4620
```

TABLE C-I - (Continued)

```
                    /
                    /SUBROUTINE CLOCK
                    /
              /REAL TIME CLOCK ENABLE ROUTINE
0600  0000  CLOCK,  0000
0601  6002          IOF
0602  6132          CCFF
0603  7300          CLA CLL
0604  1211          TAD TLIM
0605  6137          CEIL    /LOAD CLOCK COUNTER,ENABLE CLOCK
0606  7300          CLA CLL
0607  6001          ION
0610  5600          JMP I CLOCK     /CONNECT CLOCK TO INTERRUPT
0611  0050  TLIM,   0050    /DESIRED COUNT,NOT COMPLIMENT,.1 SECS
                    /10 POINTS PER SECONDS FOR 10 KC CLOCK
                    /FASTEST SPEED 25.0 *REAL TIME
                    /7 1/2 IPS CLOCK SET AT 0050
                    /USING 16 BIT WORD ,2 DATA POINTS PER 360 WO
                    /
                    /SUBROUTINE READ
                    /
              /ANALOG TO DIGITAL CONVERTER SUBROUTINE
0612  0000  READ,   0000
0613  6002          IOF
0614  7300          CLA CLL
0615  6541          ADCC    /CLEAR MULTIPLEXER CHAN ADDR REG
0616  1065          TAD HOLD        /SET ACCUMULATOR TO CHAN ASSIGN
0617  6542          ADSC    /SET UP MULTIPLEXER CHAN
0620  7000          NOP
0621  7000          NOP
0622  7000          NOP
0623  7000          NOP
0624  6532          ADCV            /CLEAR AD CONVERTER FLAG
                    /DIGITIZE VOLTAGE
0625  6531          ADSF            /SKIP IF AD FLAG IS 1
0626  5225          JMP .-1
0627  7300          CLA CLL
0630  6534          ADRB    /READ AD CONVERTER BUFFER INTO AC
0631  7000          NOP     /CIA FOR NEGATIVE CAL SIGNAL
0632  7000          NOP
0633  7000          NOP
0634  7100          CLL
0635  5612          JMP I READ
                    /
                    /CLOCK INTERRUPT SERVICE ROUTINE
                    /
0636  7300  SERV,   CLA CLL
0637  4200          JMS CLOCK
0640  2073          ISZ WHERE       /IF WHERE =-1,LOSS OF DATA
0641  5243          JMP OK
0642  7402          HLT
0643  7300  OK,     CLA CLL
0644  1060          TAD ONE
0645  3073          DCA WHERE
```

TABLE C-I - (Continued)

```
                        /
                        /SUBROUTINE CLOCK
                        /
                /REAL TIME CLOCK ENABLE ROUTINE
0600  0000  CLOCK,  0000
0601  6002          IOF
0602  6132          CCFF
0603  7300          CLA CLL
0604  1211          TAD TLIM
0605  6137          CEIL    /LOAD CLOCK COUNTER,ENABLE CLOCK
0606  7300          CLA CLL
0607  6001          ION
0610  5600          JMP I CLOCK     /CONNECT CLOCK TO INTERRUPT
0611  0050  TLIM,   0050    /DESIRED COUNT,NOT COMPLIMENT,.1 SECS
                        /10 POINTS PER SECONDS FOR 10 KC CLOCK
                        /FASTEST SPEED 25.6 *REAL TIME
                        /7 1/2 IPS CLOCK SET AT 0050
                        /USING 16 BIT WORD ,2 DATA POINTS PER 360 WD
                        /
                        /SUBROUTINE READ
                        /
                /ANALOG TO DIGITAL CONVERTER SUBROUTINE
0612  0000  READ,   0000
0613  6002          IOF
0614  7300          CLA CLL
0615  6541          ADCC    /CLEAR MULTIPLEXER CHAN ADDR REG
0616  1065          TAD HOLD        /SET ACCUMULATOR TO CHAN ASSIGN
0617  6542          ADSC    /SET UP MULTIPLEXER CHAN
0620  7000          NOP
0621  7000          NOP
0622  7000          NOP
0623  7000          NOP
0624  6531          ADCV            /CLEAR AD CONVERTER FLAG
                                      /DIGITIZE VOLTAGE
0625  6531          ADSF            /SKIP IF AD FLAG IS 1
0626  5225          JMP .-1
0627  7300          CLA CLL
0630  6534          ADRB    /READ AD CONVERTER BUFFER INTO AC
0631  7000          NOP     /CIA FOR NEGATIVE CAL SIGNAL
0632  7000          NOP
0633  7000          NOP
0634  7100          CLL
0635  5612          JMP I READ
                        /
                        /CLOCK INTERRUPT SERVICE ROUTINE
                        /
0636  7300  SERV,   CLA CLL
0637  4200          JMS CLOCK
0640  0073          ISZ WHERE       /IF WHERE =-1,LOSS OF DATA
0641  5243          JMP OK
0642  7402          HLT
0643  7300  OK,     CLA CLL
0644  1060          TAD ONE
0645  3073          DCA WHERE
```

TABLE C-I - (Continued)

```
0646  1064          TAD CHAN
0647  3065          DCA HOLD
0650  1060          TAD ONE
0651  3126          DCA FOUR
0652  1060          TAD ONE
0653  3123          DCA SLEEP
0654  4212          JMS READ
0655  1777          TAD SLMEAN
0656  3071          DCA WAVE2       /WAVE INDUCED
0657  2065          ISZ HOLD
0660  4212          JMS READ
0661  1776          TAD MEANMD
0662  3072          DCA MODE12
0663  2065          ISZ HOLD
0664  4212          JMS READ
0665  1775          TAD COMBIN
0666  3074          DCA BUFFPT
0667  7300          CLA CLL
0670  4774          JMS MULVAL
0671  4773          JMS MGSET
0672  4772          JMS MAGUP1
0673  6001          ION
0674  7300  NOPUT,  CLA CLL
0675  3073          DCA WHERE
0676  5771  OUTER,  JMP BACKER       /AFTER 1ST TIME THROUGH REPLACE
                                      /WITH   JMP I DIS
0677  5771  OUTERR, JMP BACKER
0700  7300  AROUND, CLA CLL         /SWITCH TO SAY WE HAVE
0701  1060          TAD ONE         /STARTED PEAK TO TROUGH
0702  3073          DCA WHERE       / EVALUATION
0703  1113          TAD ZERO
0704  3123          DCA SLEEP
0705  1125          TAD F4
0706  3126          DCA FOUR
                        /
                        /ROUTINE TO CALCULATE PEAK TO TROUGH VALUES
0707  7300  ENDED,  CLA CLL
0710  1102          TAD FIRST
0711  7510          SPA             /SKIP ON POS AC
0712  5334          JMP TEST
0713  7300  GOGO,   CLA CLL
0714  1055          TAD WAVE1
0715  3100          DCA A
0716  1071          TAD WAVE2
0717  3101          DCA B
0720  1071          TAD WAVE2
0721  7510          SPA
0722  7041          CIA
0723  1352          TAD LIMSTO
0724  7510          SPA
0725  5770          JMP MODAL
0726  4767          JMS IN
0727  7300          CLA CLL
0730  3101          DCA B
0731  1100          TAD A
```

TABLE C-I - (Continued)

```
0732  3055          DCA WAVE1
0733  5770          JMP MODAL
0734  7300  TEST,   CLA CLL
0735  1071          TAD WAVE2
0736  7510          SPA                 /SKIP ON POS AC
0737  5343          JMP NOGO
0740  7300          CLA CLL
0741  3102          DCA FIRST
0742  5313          JMP GOGO
0743  7300  NOGO,   CLA CLL
0744  3073          DCA WHERE
0745  2123          ISZ SLEEP
0746  5345          JMP .-1
0747  2126          ISZ FOUR
0750  5345          JMP .-3
0751  5300          JMP AROUND
0752  7775  LIMSTO, 7775
0767  1000
0770  1200
0771  0532
0772  1642
0773  1656
0774  4264
0775  2642
0776  2545
0777  2542

                    PAGE    /START OF NEW PAGE ******
1000  0000  IN,     0000
1001  7300          CLA CLL
1002  1100          TAD A
1003  0106          AND MASK            /MASK OUT ALL BUT BIT 0
1004  1101          TAD B
1005  7430          SZL                 /LINK=1 IMPLIES
1006  5237          JMP BTHNEG          /BOTH NEGATIVE
1007  7004          RAL                 /ROTATE BIT 0 INTO LINK
1010  7630          SZL CLA             /BIT 0=1 IMPLIES
1011  5226          JMP OPPSGN          /OPPOSITE SIGNS
1012  7300  BTHPOS, CLA CLL
1013  1100          TAD A
1014  7041          CIA     /COMPLIMENT AND INCREMENT AC
1015  1101          TAD B
1016  7510          SPA     /SKIP ON POSITIVE AC,B GR A
1017  7500          SMA     /SKIP ON MINUS AC, A GR B
1020  5222          JMP POS
1021  5365          JMP OUT
1022  7300  POS,    CLA CLL
1023  1101          TAD B
1024  3100          DCA A
1025  5365          JMP OUT
1026  7300  OPPSGN, CLA CLL
1027  2104          ISZ SWT1
1030  5232          JMP .+2
1031  5247          JMP NEGSAV
1032  1100          TAD A
1033  3107          DCA SAVPOS
```

TABLE C-I - (Continued)

```
1034  1101          TAD B
1035  3100          DCA A
1036  5365          JMP OUT
1037  7300  BTHNEG, CLA CLL
1040  1101          TAD B
1041  7041          CIA
1042  1100          TAD A
1043  7510          SPA     /SKIP ON POS AC
1044  5365          JMP OUT
1045  5222          JMP POS
1046  3773  LIMIT,  3773
1047  7300  NEGSAV, CLA CLL
1050  1100          TAD A
1051  7041          CIA
1052  1107          TAD SAVPOS
1053  7510          SPA
1054  5265          JMP NEG1
1055  1246          TAD LIMIT
1056  7710          SPA CLA
1057  5265          JMP NEG1
1060  1103          TAD SWT
1061  3104          DCA SWT1
1062  1107          TAD SAVPOS
1063  3100          DCA A
1064  5365          JMP OUT
1065  7300  NEG1,   CLA CLL
1066  1100          TAD A
1067  7041          CIA
1070  1107          TAD SAVPOS
1071  3107          DCA SAVPOS
1072  1107          TAD SAVPOS
1073  3512          DCA I PKBUFT
1074  1101          TAD B
1075  3100          DCA A
1076  2112          ISZ PKBUFT
1077  1103          TAD SWT
1100  3104          DCA SWT1
1101  2115          ISZ NUMPKT
1102  5304          JMP OPS
1103  7402          HLT
1104  7300  OPS,    CLA CLL
                    /CALCULATE MAX PEAK TO TROUGH VALUE
                          /END OF SUBROUTINE BHAE
1105  1325
```

TABLE C-I - (Continued)

```
                    TAD NEGSGN
1106  7440          SZA
1107  5326          JMP LARGER
1110  1107          TAD SAVPOS
1111  7710          SPA CLA
1112  5322          JMP KEEPER
1113  7300          CLA CLL
1114  1116          TAD MAXPKT
1115  7041          CIA
1116  1107          TAD SAVPOS
1117  7510          SPA
1120  5341          JMP DUBADD
1121  5336          JMP KEEP
1122  7240  KEEPER, STA
1123  3325          DCA NEGSGN
1124  5336          JMP KEEP
1125  0000  NEGSGN, 0000
1126  7300  LARGER, CLA CLL
1127  1107          TAD SAVPOS
1130  7500          SMA
1131  5341          JMP DUBADD
1132  7041          CIA
1133  1116          TAD MAXPKT
1134  7500          SMA
1135  5341          JMP DUBADD
1136  7300  KEEP,   CLA CLL
1137  1107          TAD SAVPOS
1140  3116          DCA MAXPKT
                    /DOUBLE PRECISION       ROUTINE FOR RMS CALCULATIONS
1141  7300  DUBADD, CLA CLL
1142  1107          TAD SAVPOS
1143  3347          DCA .+4
1144  1107          TAD SAVPOS
1145  7421          MQL
1146  7405          MUY
1147  0000          0000
1150  3366          DCA A2
1151  7300          CLA CLL
1152  7501          MQA
1153  1367          TAD SUMPLL
1154  3367          DCA SUMPLL
1155  7004          RAL
1156  1366          TAD A2
1157  1077          TAD SUMPKL
1160  3077          DCA SUMPKL
1161  7004          RAL
1162  1076          TAD SUMPKH
1163  3076          DCA SUMPKH
1164  3107          DCA SAVPOS
1165  5600  OUT,    JMP I IN
1166  0000  A2,     0000
1167  0000  SUMPLL, 0000
                    /
                    / 1ST MODE HULL VIBRATION SECTION
                    /
```

TABLE C-I - (Continued)

```
1200  7300          PAGE              /****START NEW PAGE
      MODAL,  CLA CLL
1201  1777          TAD COUNT
1202  7510          SPA
1203  5227          JMP TESTM
1204  7300  GOGOM,  CLA CLL
1205  1056          TAD MODE11
1206  3100          DCA A
1207  1072          TAD MODE12
1210  3101          DCA B
1211  1072          TAD MODE12
1212  7510          SPA
1213  7041          CIA
1214  1226          TAD LIMST1
1215  7710          SPA CLA
1216  5776          JMP NOGO
1217  5236          JMP INMOD
1220  7300  MODAL1, CLA CLL
1221  1100          TAD A
1222  3056          DCA MODE11
1223  3101          DCA B
1224  3073          DCA WHERE
1225  5776          JMP NOGO
1226  7770  LIMST1, 7770
1227  7300  TESTM,  CLA CLL
1230  1072          TAD MODE12
1231  7510          SPA
1232  5776          JMP NOGO
1233  7300          CLA CLL
1234  3777          DCA COUNT
1235  5204          JMP GOGOM
1236  7300  INMOD,  CLA CLL
1237  1100          TAD A
1240  0106          AND MASK
1241  1101          TAD B
1242  7430          SZL
1243  5274          JMP NEGMOD
1244  7004          RAL
1245  7630          SZL CLA
1246  5263          JMP OPPMOD
1247  7300  POSMOD, CLA CLL
1250  1100          TAD A
1251  7041          CIA
1252  1101          TAD B
1253  7510          SPA
1254  7500          SMA
1255  5257          JMP MODPOS
1256  5220          JMP MODAL1
1257  7300  MODPOS, CLA CLL
1260  1101          TAD B
1261  3100          DCA A
1262  5220          JMP MODAL1
1263  7300  OPPMOD, CLA CLL
1264  2105          ISZ SWT2
1265  5267          JMP .+2
```

TABLE C-I - (Continued)

```
1266  5303          JMP NGSAVM
1267  1100          TAD A
1270  3117          DCA SAPMOD
1271  1101          TAD B
1272  3100          DCA A
1273  5220          JMP MODAL1
1274  7300  NEGMOD, CLA CLL
1275  1101          TAD B
1276  7041          CIA
1277  1100          TAD A
1300  7510          SPA
1301  5220          JMP MODAL1
1302  5257          JMP MODPOS
1303  7300  NGSAVM, CLA CLL
1304  1100          TAD A
1305  7041          CIA
1306  1117          TAD SAPMOD
1307  3117          DCA SAPMOD
1310  1103          TAD SWT
1311  3105          DCA SWT2
1312  1101          TAD B
1313  3100          DCA A
1314  3101          DCA B
                    /
                    / BANDWIDTH CRITERIA DUE TO NOISE LEVEL
                    /
1315  1117          TAD SAPMOD
1316  7710          SPA CLA
1317  5775          JMP MODIN
1320  1774          TAD MOD4
1321  7710          SPA CLA
1322  5773          JMP OUT3
1323  1117          TAD SAPMOD
1324  7041          CIA
1325  1122          TAD BAND
1326  7500          SMA
1327  5772          JMP MOD6
1330  7300          CLA CLL
1331  1771          TAD MOD1
1332  7041          CIA
1333  1117          TAD SAPMOD
1334  7510          SPA
1335  5343          JMP NOKEEP
1336  7300  KEEPM,  CLA CLL
1337  1117          TAD SAPMOD
1340  3771          DCA MOD1
1341  3770          DCA MOD2
1342  5220          JMP MODAL1
1343  7300  NOKEEP, CLA CLL
1344  1770          TAD MOD2
1345  7440          SZA
1346  5772          JMP MOD6
1347  1060          TAD ONE
1350  3770          DCA MOD2
1351  1354          TAD LENGTH
```

TABLE C-I - (Continued)

```
1352  3767          DCA MOD3
1353  5220          JMP MODAL1
1354  7770  LENGTH, 7770
1367  1402
1370  1401
1371  1400
1372  1433
1373  1424
1374  1403
1375  1405
1376  0473
1377  5231
                    PAGE        /****START NEW PAGE
1400  0000  MOD1,   0000
1401  0000  MOD2,   0000    /0=BEGINNING OF BURST, -=END OF BURST
1402  0000  MOD3,   0000
1403  0000  MOD4,   0000
1404  0000  MOD5,   0000
1405  7300  MODIN,  CLA CLL
1406  1060          TAD ONE
1407  3203          DCA MOD4
1410  1060          TAD ONE
1411  3202          DCA MOD2
1412  1060          TAD ONE
1413  3204          DCA MOD5
1414  1117          TAD SAPMOD
1415  7041          CIA
1416  1200          TAD MOD1
1417  7700          SMA CLA
1420  5270          JMP OUT2
1421  1117          TAD SAPMOD
1422  3200          DCA MOD1
1423  5270          JMP OUT2
1424  7300  OUT3,   CLA CLL
1425  1117          TAD SAPMOD
1426  7041          CIA
1427  1122          TAD BAND
1430  7500          SMA
1431  5233          JMP MOD6
1432  5270          JMP OUT2
1433  7300  MOD6,   CLA CLL
1434  1201          TAD MOD2
1435  7700          SMA CLA
1436  5270          JMP OUT2
1437  7300          CLA CLL
1440  1117          TAD SAPMOD
1441  7041          CIA
1442  1306          TAD BAND1
1443  7700          SMA CLA
1444  5270          JMP OUT2
1445  2202          ISZ MOD3
1446  5270          JMP OUT2
1447  7300          CLA CLL
1450  3201          DCA MOD2
1451  3202          DCA MOD3
```

TABLE C-I - (Continued)

```
1452  7300          CLA CLL
1453  2120          ISZ NUMMOD
1454  1204          TAD MOD5
1455  7440          SZA
1456  5271          JMP OUT5
1457  1121          TAD MAXMOD
1460  7041          CIA
1461  1200          TAD MOD1
1462  7710          SPA CLA
1463  5266          JMP OUT4
1464  1200          TAD MOD1
1465  3121          DCA MAXMOD
1466  3200  OUT4,   DCA MOD1
1467  3203          DCA MOD4
1470  5777  OUT2,   JMP MODAL1
1471  7300  OUT5,   CLA CLL
1472  1121          TAD MAXMOD
1473  7700          SMA CLA
1474  5266          JMP OUT4
1475  7300          CLA CLL
1476  1200          TAD MOD1
1477  7041          CIA
1500  1121          TAD MAXMOD
1501  7700          SMA CLA
1502  5266          JMP OUT4
1503  1200          TAD MOD1
1504  3121          DCA MAXMOD
1505  5266          JMP OUT4
1506  0050  BAND1,  0050
```

TABLE C-I - (Continued)

```
                /DIGITAL 8-22-F
                /TWO'S COMPLEMENT, SINGLE PRECISION,
                /SIGNED DIVIDE SUBROUTINE WITH EAE.
                /RETURN QUOTIENT IN AC; REMAINDER IN DVD.
                /IF HIGH ORDER DIVIDEND IS EQUAL TO OR
                /GREATER THAN THE DIVISOR, DIVIDE OVERFLOW
                /WILL OCCUR AND THE LINK IS SET TO ONE.
1507  0000  SPDIV, 0
1510  7100        CLL
1511  7510        SPA     /TEST FOR NEGATIVE DIVIDEND.
1512  7060        CMA CML
1513  3363        DCA DVD /SAVE HIGH ORDER DIVIDEND.
1514  7420        SNL
1515  7040        CMA
1516  3362        DCA RMSGN       /7777 IF DIVIDEND POSITIVE.
1517  1707        TAD I SPDIV     /LOW ORDER DIVIDEND.
1520  7430        SZL
1521  5354        JMP NEG /GO TO COMPLEMENT DIVIDEND.
1522  7421  DV2,  MQL     /LOW ORDER DIVIDEND TO MQ.
1523  2307        ISZ SPDIV
1524  1707        TAD I SPDIV     /DIVISOR.
1525  2307        ISZ SPDIV       /SET UP FOR EXIT.
1526  7510        SPA     /TEST FOR NEGATIVE DIVISOR.
1527  7061        CMA CML IAC
1530  3336        DCA .+6 /STORE DIVISOR.
1531  7420        SNL
1532  7040        CMA
1533  3361        DCA SIGN        /7777 IF QUOTIENT POSITIVE.
1534  1363        TAD DVD /HIGH ORDER DIVIDEND.
1535  7407        DVI
1536  0000        0       /DIVISOR
1537  7430        SZL
1540  5707        JMP I SPDIV     /DIVIDE OVERFLOW --- LINK=1
1541  2362        ISZ RMSGN
1542  7041        CMA IAC
1543  3363        DCA DVD /REMAINDER WITH SIGN OF DIVIDEND.
1544  7120        STL
1545  7501        MQA     /QUOTIENT TO AC.
1546  7510        SPA
1547  5707        JMP I SPDIV     /OVERFLOW --- 12 BIT QUOTIENT.
1550  2361        ISZ SIGN
1551  7041        CMA IAC
1552  7100        CLL
1553  5707        JMP I SPDIV
1554  7061  NEG,  CMA CML IAC
1555  7430        SZL
1556  2363        ISZ DVD /2'S COMPLEMENT OF HIGH IF LOW=0.
1557  7120        STL     /RESTORE LINK TO 1.
1560  5322        JMP DV2
1561  0000  SIGN,  0       /7777 IF QUOTIENT POSITIVE.
1562  0000  RMSGN, 0       /7777 IF DIVIDEND POSITIVE.
1563  0000  DVD,   0       /REMAINDER.
1577  1200
```

TABLE C-I - (Continued)

```
                PAGE
                /PDP-8I INCREMENTAL MAG TAPE    FOR READ/WRITE TRANSPORT
                /
                /IOT DEFINITIONS
                /
                        IRS=6701                /SKIP ON READ DONE
                        ISR=6702                /READ STATUS REGISTER
                        IWS=6703                /SKIP ON WRITE DONE
                        IMC=6704                /MOVE COMMAND
                        IGS=6705                /SKIP ON GAP DETECT
                        IWR=6706                /WRITE AC INTO BUFFER
                        IRD=6707                /READ BUFFER INTO AC
                        /
                        /STATUS WORD BIT ASSIGNMENT
                        /AC BIT         FUNCTION
                        /0              TRANSPORT READY
                        /1              WRITE BUFFER BUSY
                        /2              TAPE NOT TENSIONED
                        /3              GAP IN PROCESS
                        /4              FILE PROTECT (WRITE RING)
                        /5              BOT
                        /6              EOT
                        /7              WRITE STATUS
                        /8              READ STATUS
                        /9              ECHO CHECK (WRITE ONLY)
                        /
                        /MOVE COMMAND BIT ASSIGNMENT
                        /COMMAND                BITS    0 1 2
                        /REWIND                         0 0 0
                        /WRITE IRG                      0 0 1
                        /WRITE EFG                      0 1 0
                        /SETUP TO WRITE                 0 1 1
                        /LOAD FORWARD READY READ        1 0 0
                        /LOAD FORWARD READY WRITE       1 0 1
                        /READ FORWARD                   1 1 0
                        /BACK SPACE RECORD              1 1 1
1600  0000  REWIND, 0000    /STORES RETURN ADDRESS
1601  6704        IMC     /START THE REWIND
1602  7300        CLA CLL
1603  5600        JMP I REWIND
                  /REWIND ROUTINE MUST ALWAYS BE FOLLOWED
                  /BY A LOAD FORWARD
                  /
1604  0000  LWRT,  0000    /START OF LOAD FORWARD TO WRITE
1605  7300        CLA CLL
1606  6702        ISR     /LOAD STATUS REGISTER CONTENTS INTO AC FOR
                          /CHECK
1607  0240        AND M7600       /AND 7600 WITH AC TO MASK UNCHECKED
                                  /BITS
1610  7440        SZA             /SKIP IF OKAY TO PROCEED
1611  7402        HLT             /HALT IF STATUS INCORRECT
1612  1234        TAD LFW         /TAD LOAD FORWARD TO WRITE INTO AC
1613  6704        IMC     /MOVE TAPE TO LOAD POINT
1614  7300        CLA CLL
1615  5604        JMP I LWRT      /RETURN TO MAIN PROGRAM
```

TABLE C-I - (Continued)

```
                        /SUBROUTINE DOES NOT WAITU UNTIL LOAD
                        /FORWARD IS COMPLETE, HOWEVER, THE LOAD
                        /FORWARD IS CONNECTED TO THE WRITE DONE FLAG
                        /AND WILL INHIBIT WRITING
                        /
                        /LOAD FORWARD TO READ
1616  0000  LREAD,      0000
1617  7300              CLA CLL
1620  6702              ISR     /LOAD CONTENTS OF STATUS REGISTER INTO
                                /AC FOR CHECK
1621  0237              AND M7400       /MASK UNCHECKED BITS
1622  7440              SZA             /SKIP IF OKAY TO PROCEED
1623  7402              HLT
1624  1236              TAD LFR         /TAD LOAD FORWARD TO READ
1625  6704              IMC             /INTO AC, MOVE TAPE TO LOAD PT
1626  7300              CLA CLL
1627  5616              JMP I LREAD
                        /
                        /MAG TAPE WRITESUBROUTINE
                        /
1630  0000  MGWRTE,     0000
1631  6706              TWK     /PLACE DATA WORD IN WRITE BUFFER AND
                                /AND GENERATE WRITE STEP
1632  7300              CLA CLL
1633  5630              JMP I MGWRTE    /SINGLE WORD RETURN
1634  5000  LFW,        5000
1635  0000  STATUS,     0000
1636  4000  LFR,        4000
1637  7400  M7400,      7400
1640  7600  M7600,      7600
1641  3600  M3600,      3600
1642  0000  MAGUP1,     0000
1643  7300              CLA CLL
1644  1351              TAD TESTR
1645  7440              SZA
1646  5642              JMP I MAGUP1
1647  7300              CLA CLL
1650  1350              TAD DMAG
1651  7440              SZA
1652  5642              JMP I MAGUP1
1653  1413              TAD I TAPLOC
1654  4030              JMS MGWRTE
1655  5642              JMP I MAGUP1
1656  0000
```

TABLE C-I - (Continued)

```
                MGSET,  0000
1657  7300              CLA CLL
1660  1132              TAD TAP1
1661  0135              AND T0017
1662  7006              RTL
1663  7006              RTL
1664  3132              DCA TAP1
1665  1133              TAD TAP2
1666  0136              AND T7400
1667  7012              RTR
1670  7012              RTR
1671  7012              RTR
1672  7012              RTR
1673  1132              TAD TAP1
1674  3412              DCA I TAPBUF    /BYTE 1
1675  1133              TAD TAP2
1676  3412              DCA I TAPBUF    /BYTE 2
1677  7300              CLA CLL
1700  1355              TAD M2
1701  1131              TAD MAGCNT
1702  3131              DCA MAGCNT
1703  5656              JMP I MGSET
1704  7300  MAGUP,      CLA CLL
1705  6002              IOF
1706  3350              DCA DMAG
1707  1351              TAD TESTR
1710  7710              SPA CLA
1711  5330              JMP MAG2
1712  1060              TAD ONE
1713  1131              TAD MAGCNT
1714  3131              DCA MAGCNT
1715  2354              ISZ MAGIRZ
1716  5330              JMP MAG2
1717  1075              TAD IRG
1720  4200              JMS REWIND
1721  1060              TAD ONE
1722  3351              DCA TESTR
1723  1353              TAD MAGIRG
1724  3354              DCA MAGIRZ
1725  2061              ISZ SAMP1
1726  5566              JMP I DIS
1727  5777              JMP ALLONE
1730  7300  MAG2,       CLA CLL
1731  3351              DCA TESTR
1732  1131              TAD MAGCNT
1733  7450              SNA
1734  5343              JMP MAG1
1735  7300              CLA CLL
1736  1413              TAD I TAPLOC
1737  4030              JMS MGWRTE
1740  1060              TAD ONE
1741  3350              DCA DMAG
1742  5566              JMP I DIS
1743  1147  MAG1,       TAD LOCTAP
1744  3013              DCA TAPLOC
```

TABLE C-I - (Continued)

```
1745  1147              TAD LOCTAP
1746  3012              DCA TAPBUF
1747  4566              JMS I DIS
1750  0000  DMAG,       0000
1751  0000  TESTR,      0000
1752  0400  M0400,      0400
1753  4060  MAGIRG,     4060
1754  0000  MAGIRZ,     0000
1755  0002  M2,         0002
```

TABLE C-I - (Continued)

```
                        /
                        /THIS ROUTINE LOOKS
                        /FOR CAL SIGNAL ONLY
                        /
1777  2724
                        PAGE
2000  0000  CALIB,     0000
2001  7300              CLA CLL
2002  1342              TAD CHAN3
2003  3065              DCA HOLD
2004  3350              DCA P6
2005  1777              TAD LONGER
2006  7710              SPA CLA
2007  5776              JMP LOAFER
2010  7300  CALIB9,     CLA CLL
2011  3352              DCA P8
2012  1060              TAD ONE
2013  3354              DCA P10
2014  7300              CLA CLL
2015  1341              TAD MINFOR
2016  3353              DCA P9
2017  1340  CALIB8,     TAD MINTWO
2020  3347              DCA P5
2021  3351              DCA P7
2022  7300  CALIB4,     CLA CLL
2023  4775              JMS READ
2024  3343              DCA P1
2025  4774              JMS CLOK
2026  4774              JMS CLOK
2027  4775              JMS READ
2030  3344              DCA P2
2031  2350              ISZ P6
2032  2351              ISZ P7
2033  7300  CALIB5,     CLA CLL
2034  4774              JMS CLOK
2035  4774              JMS CLOK
2036  4775              JMS READ
2037  3345              DCA P3
2040  2350              ISZ P6
2041  2351              ISZ P7
2042  1343              TAD P1
2043  7041              CIA
2044  1345              TAD P3
2045  7510              SPA
2046  7041              CIA
2047  3346              DCA P4
2050  7300              CLA CLL
2051  1344              TAD P2
2052  3343              DCA P1
2053  1345              TAD P3
2054  3344              DCA P2
2055  1346              TAD P4
2056  1355              TAD C1000
2057  7710              SPA CLA
2060  5271              JMP CALIB7
```

TABLE C-I - (Continued)

```
2061  1353              TAD P6
2062  1356              TAD TIME
2063  7710              SPA CLA
2064  5233              JMP CALIB5
2065  1777              TAD LONGER
2066  7700              SMA CLA
2067  5233              JMP CALIB5
2070  5773              JMP LMEAN
2071  7300  CALIB7,     CLA CLL
2072  2347              ISZ P5
2073  5275              JMP CALIB2
2074  5306              JMP CALIB3
2075  7300  CALIB2,     CLA CLL
2076  1354              TAD P10
2077  7710              SPA CLA
2100  5304              JMP .+4
2101  1351              TAD P7
2102  1352              TAD P8
2103  3352              DCA P8
2104  3351              DCA P7
2105  5222              JMP CALIB4
2106  7300  CALIB3,     CLA CLL
2107  1351              TAD P7
2110  1357              TAD SIGNAL
2111  7700              SMA CLA
2112  5210              JMP CALIB9
2113  1351              TAD P7
2114  1360              TAD SIGNA2
2115  7710              SPA CLA
2116  5210              JMP CALIB9
2117  1351              TAD P7
2120  1352              TAD P8
2121  3352              DCA P8
2122  3354              DCA P10
2123  2353              ISZ P9
2124  5217              JMP CALIB8
2125  7300              CLA CLL
2126  1352              TAD P8
2127  1361              TAD SIGNA3
2130  7700              SMA CLA
2131  5210              JMP CALIB9
2132  1352              TAD P8
2133  1362              TAD SIGNA4
2134  7710              SPA CLA
2135  5210              JMP CALIB9
2136  5772              JMP CALSIG
2137  5600  CALIB6,     JMP I CALIB
2140  7776  MINTWO,     7776
2141  7774  MINFOR,     7774
2142  0003  CHAN3,      0003
2143  0000  P1,         0000
2144  0000  P2,         0000
2145  0000  P3,         0000
2146  0000  P4,         0000
2147  0000  P5,         0000
```

TABLE C-I - (Continued)

```
2150  0000  P6,         0000
2151  0000  P7,         0000
2152  0000  P8,         0000
2153  0000  P9,         0000
2154  0000  P10,        0000
2155  2314  C1000,      2314
2156  4366  TIME,       4366
2157  7747  SIGNAL,     7747
2160  7772  SIGNA2,     7772
2161  7603  SIGNA3,     7603
2162  7661  SIGNA4,     7661
2172  2200
2173  2426
2174  2402
2175  0612
2176  2311
2177  5130
                        PAGE
2200  7300  CALSIG,     CLA CLL
2201  1777              TAD NOGOIN
2202  7710              SPA CLA
2203  5324              JMP LOAF2
2204  1776              TAD INT1
2205  7700              SMA CLA
2206  5304              JMP SPA5
2207  1162              TAD TENPTS
2210  3155              DCA POINT1
2211  4775  CCLOK2,     JMS CLOK
2212  2155              ISZ POINT1
2213  5211              JMP CCLOK2
2214  7300              CLA CLL
2215  1161              TAD N19
2216  3153              DCA MAX1
2217  3115              DCA NUMPKT
2220  3077              DCA SUMPKL
2221  3076              DCA SUMPKH
2222  1307              TAD SPAC
2223  3231              DCA CHNG
2224  7300  AOK,        CLA CLL
2225  1162              TAD TENPTS
2226  3154              DCA MAX2
2227  4775  CCLOK,      JMS CLOK
2230  4774              JMS READ
2231  7041  CHNG,       CIA
2232  3773              DCA SAVE3
2233  7300              CLA CLL
2234  1773              TAD SAVE3
2235  1077              TAD SUMPKL
2236  3077              DCA SUMPKL
2237  7004              RAL
2240  1076              TAD SUMPKH
2241  3076              DCA SUMPKH
2242  1772              TAD SAVE3
2243  7710              SPA CLA
2244  1060              TAD ONE
```

TABLE C-I - (Continued)

```
2245  1076              TAD SUMPKH
2246  3076              DCA SUMPKH
2247  2115              ISZ NUMPKT
2250  2154  CHNG3,      ISZ MAX2
2251  5222              JMP CCLOK
2252  1163              TAD TWENT
2253  3155              DCA POINT1
2254  4775  CCLOK1,     JMS CLOK
2255  2155              ISZ POINT1
2256  5254              JMP CCLOK1
2257  1310              TAD SMA1
2260  3231              DCA CHNG
2261  2153              ISZ MAX1
2262  5224              JMP AOK
2263  7300              CLA CLL
2264  7621              CAM
2265  1115              TAD NUMPKT
2266  7110              RAR CLL
2267  3275              DCA VISOR1
2270  1077              TAD SUMPPL
2271  3274              DCA LOWVOL
2272  1076              TAD SUMPKH
2273  4542              JMS I LNKDIV
2274  0000  LOWVOL,     0000
2275  0000  VISOR1,     0000
2276  7430              SZL
2277  7402              HLT
2300  7510              SPA
2301  7041              CIA
2302  3772              DCA VALUE1
2303  4771              JMS CALVAL
2304  7300  SPA5,       CLA CLL
2305  6132              CCFF
2306  5770              JMP LMEAN
2307  7000  SPAC,       NOP
2310  7041  SMA1,       CIA
2311  7300  LOAFER,     CLA CLL
2312  1322              TAD M3
2313  3323              DCA M3A
2314  7300  LOAFA,      CLA CLL
2315  4775              JMS CLOK
2316  2323              ISZ M3A
2317  5314              JMP LOAFA
2320  7300              CLA CLL
2321  5770              JMP LMEAN
2322  5520  M3,         5520
2323  0000  M3A,        0000
2324  7300  LOAF2,      CLA CLL
2325  1767              TAD M20A
2326  3766              DCA M20B
2327  1765              TAD M1000A
2330  3764              DCA M1000B
2331  4775              JMS CLOK
2332  2764              ISZ M1000B
2333  5331              JMP .+2
```

TABLE C-I - (Continued)

```
2334  2766          ISZ M20B
2335  5327          JMP .-6
2336  5763          JMP BEGIN2
2363  0440
2364  5134
2365  5133
2366  2401
2367  2400
2370  2426
2371  4221
2372  3750
2373  3276
2374  0612
2375  2402
2376  0536
2377  5146
                    PAGE
                    /
                    /END OF CALIB SUBROUTINE
                    /
2400  7764  M20A,   7764
2401  0000  M20B,   0000
```

TABLE C-I - (Continued)

```
                    /
                    /CONTINUATION OF CALIBRATION ROUTINE
                    /
                    /
                    /SUBROUTINE KAL1
                    /
2402  0000  CLOK,   0000
2403  7300          CLA CLL
2404  4777          JMS CLOCK
2405  6002          IOF
2406  6133          CSCF
2407  5206          JMP .-1
2410  7300          CLA CLL
2411  5602          JMP I CLOK
2412  0000  LOAF,   0000    /LOAF FOR 60 SECONDS
2413  7300          CLA CLL
2414  1224          TAD M340
2415  3225          DCA M340A
2416  7300  LOAF1,  CLA CLL
2417  4202          JMS CLOK
2420  2225          ISZ M340A
2421  5216          JMP LOAF1
2422  7300          CLA CLL
2423  5612          JMP I LOAF
2424  6650  M340,   6650
2425  0000  M340A,  0000
                    /
                    /SAMPLE FIRST 3 MINUTES AFTER CAL SIGNAL
                    /TO GET EFFECTIVE MEAN VALUES
                    /
2426  7300  LMEAN,  CLA CLL
2427  1776          TAD INT1
2430  7710          SPA CLA
2431  5352          JMP S5
2432  7300          CLA CLL
2433  6212          CIF+10
2434  4576          JMS I DATAF
2435  7300          CLA CLL
2436  6202          CIF+00
2437  6201          CDF+00
2440  1336  S3,     TAD M330        /2048 SAMPLES
2441  3337          DCA M330A
2442  3341          DCA LMEAN2
2443  3340          DCA LMEAN1
2444  3343          DCA LMEAN4
2445  3344          DCA LMEAN5
2446  3350          DCA LMEAN6
2447  3351          DCA LMEAN7
2450  3775          DCA LONGER
2451  4212          JMS LOAF
2452  7300  LMEAN3, CLA CLL
2453  1064          TAD CHAN
2454  3065          DCA HOLD
2455  4202          JMS CLOK
2456  4774          JMS READ
```

TABLE C-I - (Continued)

```
2457  3074          DCA BUFFPT
2460  1074          TAD BUFFPT
2461  1341          TAD LMEAN2
2462  3341          DCA LMEAN2
2463  7004          RAL
2464  1340          TAD LMEAN1
2465  3340          DCA LMEAN1
2466  1074          TAD BUFFPT
2467  7710          SPA CLA
2470  1060          TAD ONE
2471  1340          TAD LMEAN1
2472  3340          DCA LMEAN1
2473  2065          ISZ HOLD
2474  4774          JMS READ
2475  3074          DCA BUFFPT
2476  1074          TAD BUFFPT
2477  1344          TAD LMEAN5
2500  3344          DCA LMEAN5
2501  7004          RAL
2502  1343          TAD LMEAN4
2503  3343          DCA LMEAN4
2504  1074          TAD BUFFPT
2505  7710          SPA CLA
2506  1060          TAD ONE
2507  1343          TAD LMEAN4
2510  3343          DCA LMEAN4
2511  2065          ISZ HOLD
2512  4774          JMS READ
2513  3074          DCA BUFFPT
2514  1074          TAD BUFFPT
2515  1773          TAD COMBIN
2516  3074          DCA BUFFPT
2517  7300          CLA CLL
2520  1074          TAD BUFFPT
2521  1351          TAD LMEAN7
2522  3351          DCA LMEAN7
2523  7004          RAL
2524  1350          TAD LMEAN6
2525  3350          DCA LMEAN6
2526  1074          TAD BUFFPT
2527  7710          SPA CLA
2530  1060          TAD ONE
2531  1350          TAD LMEAN6
2532  3350          DCA LMEAN6
2533  2337          ISZ M330A
2534  5252          JMP LMEAN3
2535  5772          JMP NEWPS2
2536  3777  M330,   3777
2537  0000  M330A,  0000
2540  0000  LMEAN1, 0000
2541  0000  LMEAN2, 0000
2542  0000  SLMEAN, 0000
2543  0000  LMEAN4, 0000
2544  0000  LMEAN5, 0000
2545  0000  MEANND, 0000
```

TABLE C-I - (Continued)

```
2546  4370  S1,     4370    /LOAF FOR 3 MINUTES
2547  0000  S2,     0000
2550  0000  LMEAN6, 0000
2551  0000  LMEAN7, 0000
2552  7300  S5,     CLA CLL
2553  1346          TAD S1
2554  3347          DCA S2
2555  4202  S6,     JMS CLOK
2556  2347          ISZ S2
2557  5355          JMP S6
2560  7300          CLA CLL
2561  5240          JMP S3
2572  2600
2573  2642
2574  0612
2575  5130
2576  0536
2577  0600
                    PAGE    /***** NEW PAGE
2600  7300  NEWPG2, CLA CLL
2601  1777          TAD LMEAN2
2602  7421          MQL
2603  1776          TAD LMEAN1
2604  7415          ASR
2605  0012          0012    / DIVIDE BY 2048
2606  7701          MQA CLA
2607  7041          CIA
2610  3775          DCA SLMEAN
2611  7300          CLA CLL
2612  1774          TAD LMEAN5
2613  7421          MQL
2614  1773          TAD LMEAN4
2615  7415          ASR
2616  0012          0012
2617  7701          MQA CLA         /PEAK TO TROUGH 1ST MODE
2620  7041          CIA
2621  3772          DCA MEANMD
2622  7300          CLA CLL
2623  1771          TAD LMEAN6
2624  3145          DCA MEAN1
2625  1770          TAD LMEAN7
2626  3146          DCA MEAN2
2627  2767          ISZ INT1
2630  5766          JMP CALIB6
2631  1770          TAD LMEAN7
2632  7421          MQL
2633  1771          TAD LMEAN6
2634  7415          ASR
2635  0012          0012
2636  7701          MQA CLA         /COMBINED SIGNAL
2637  7041          CIA
2640  3242          DCA COMBIN
2641  5766          JMP CALIB6
2642  0000  COMBIN, 0000
                    /
```

TABLE C-I - (Continued)

```
                    /
                    /END OF SUBROUTINE KAL1
                    /
                    /START OF DISMISS ROUTINE
                    /
2643  7300  DISMIS, CLA CLL
2644  6002          IOF
2645  7240          CLA CMA
2646  1066          TAD AC
2647  3065          DCA AC
2650  7240          CLA CMA
2651  1067          TAD L
2652  3067          DCA L
2653  7240          CLA CMA
2654  1070          TAD PCREG

2655  3070          DCA PCREG
2656  1470          TAD I PCREG
2657  3000          DCA 0000
2660  1467          TAD I L
2661  7104          CLL RAL
2662  1466          TAD I AC
2663  6001          ION
2664  5400          JMP I 0000
                  /ERROR ROUTINE CAUSES HLT IF DECTAPE ERROR
                  /FLAG SET, NOT POSSIBLE TO RESTART BY CONTINUEING
2665  7300  DTER,   CLA CLL
2666  6772          DTRB
2667  7402          HLT
2670  5267          JMP .-1
                    /
                    /SUBROUTINE SEARCH
                    /
                    / INITIAL POSITIONING OF
                    /DECTAPE ON UNIT 4
                    /
2671  0000  SEARCH, 0000
2672  7300          CLA CLL
2673  1171          TAD RBLOK1
2674  4700          JMS I DSERR1
2675  2677          SEROUT
2676  4000          4000
2677  7300  SEROUT, CLA CLL
2700  4452          JMS I DONE
2701  5671          JMP I SEARCH
2702  0677  DSERR1, DSERR
                    /
                    /DECTAPE WRITE SUBROUTINE
                    /
2703  0000  DECTAP, 0000
2704  7300          CLA CLL
2705  1663          TAD BLOK
2706  3313          DCA BN0
2707  4722          JMS I W128Y
2710  0000  BUF1,   0000
2711  4000          4000
```

```
2712  7774  BCN,    7774
2713  0000  BN0,    0000
2714  4452          JMS I DONE
2715  7300          CLA CLL
2716  1663          TAD BLOK
2717  1393          TAD M88
2720  3663          DCA BLOK
2721  5703  OUT1,   JMP I DECTAP
2722  0206  W128Y,  W128
2723  0086  M88,    0086
```

TABLE C-I - (Continued)

```
                        /START NEW SUBROUTINE TAPE WRITE 1
2724  7300  ALDONE,  CLA CLL
2725  6002           IOF
2726  6132           CCFF
2727  6703           IVS
2730  5327           JMP .-1
2731  2765           ISZ NOINT
2732  1111           TAD PKBUF1
2733  3014           DCA REG14
2734  1764           TAD CAND1
2735  3763           DCA CAND2
2736  1764           TAD CAND1        /CAL FACTOR
2737  3414           DCA I REG14
2740  1762           TAD SHIFT3       /SHIFT CAL FACTOR
2741  3414           DCA I REG14
2742  1762           TAD SHIFT3
2743  3761           DCA SHIFT4
2744  1115           TAD NUMPKT
2745  1075           TAD IRG
2746  3760           DCA VIS2         /NO PEAK TO TROUGH FOR INTERVAL
2747  1760           TAD VIS2
2750  3414           DCA I REG14
2751  1760           TAD VIS2
2752  3757           DCA VIS6
2753  1120           TAD NUMMOD
2754  3414           DCA I REG14
2755  5755           JMP NEWPG4
2756  3000
2757  3020
2760  3011
2761  4206
2762  4277
2763  4204
2764  4270
2765  3272
2766  2137
2767  0536
2770  2551
2771  2550
2772  2545
2773  2543
2774  2544
2775  2542
2776  2540
2777  2541
                       PAGE             /***START NEW PAGE
3000  7300  NEWPG4,  CLA CLL
3001  1211           TAD VIS2
3002  7650           SNA CLA
3003  5252           JMP NOPEAK
3004  1077           TAD SUMPKL
3005  3210           DCA VIS1
3006  1076           TAD SUMPKH
3007  4542           JMS I LNKDIV
3010  0000  VIS1,    0000
```

TABLE C-I - (Continued)

```
3011  0000  VIS2,    0000
3012  3361           DCA H1           /RMS CALCULATION FOR INTERVAL
3013  1777           TAD SUMPLL
3014  3217           DCA VIS5
3015  1776           TAD DVD
3016  4542           JMS I LNKDIV
3017  0000  VIS5,    0000
3020  0030  VIS6,    0030
3021  3362           DCA H2
                     /
                     / SQUARE ROOT    ROUTINE
                     /
3022  1363           TAD XIFIX        /XIFIX=64 OCTAL
3023  3364           DCA XI
3024  1364  AGN,     TAD XI
3025  3233           DCA VISB
3026  1362           TAD H2
3027  3232           DCA VISA
3030  1361           TAD H1
3031  4542           JMS I LNKDIV
3032  0000  VISA,    0000
3033  0000  VISB,    0000
3034  1364           TAD XI
3035  7417           LSR
3036  0000           0000
3037  3365           DCA XIPLUS
3040  1364           TAD XI
3041  7041           CIA
3042  1365           TAD XIPLUS
3043  0366           AND M7776
3044  7650           SNA CLA          /STRIP LEAST SIGNIFICANT BIT
3045  5254           JMP OK1
3046  7300           CLA CLL
3047  1365           TAD XIPLUS
3050  3364           DCA XI
3051  5224           JMP AGN
3052  7300  NOPEAK,  CLA CLL
3053  3365           DCA XIPLUS
3054  7300  OK1,     CLA CLL
3055  1365           TAD XIPLUS
3056  3172           DCA PLY1
3057  4775           JMS MULPSI
3060  1173           TAD PLY3
3061  3414           DCA I REG14
3062  1174           TAD PLY4
3063  3414           DCA I REG14
3064  1174           TAD PLY4
3065  7417           LSR
3066  0000           0000
3067  3360           DCA S4
3070  1774           TAD VSUMPH
3071  7041           CIA
3072  1173           TAD PLY3
3073  7450           SNA
3074  5300           JMP Z1
```

TABLE C-I - (Continued)

```
3075  7700          SMA CLA
3076  5310          JMP Z2
3077  5315          JMP ALDON1
3100  7300  Z1,     CLA CLL
3101  1773          TAD VSUMPL
3102  7417          LSR
3103  0000          0000
3104  7041          CIA
3105  1360          TAD S4
3106  7510          SPA
3107  5315          JMP ALDON1
3110  7300  Z2,     CLA CLL
3111  1173          TAD PLY3
3112  3774          DCA VSUMPH
3113  1174          TAD PLY4
3114  3773          DCA VSUMPL
3115  7300  ALDON1, CLA CLL
3116  1116          TAD MAXPKT
3117  3172          DCA PLYI
3120  4775          JMS MULPSI
3121  1173          TAD PLY3        /MAX PEAK TO TROUGH
3122  3414          DCA I REG14     / VALUES
3123  1174          TAD PLY4
3124  3414          DCA I REG14
3125  1174          TAD PLY4
3126  7417          LSR
3127  0000          0000
3130  3360          DCA S4
3131  1772          TAD VMAXPH
3132  7041          CIA
3133  1173          TAD PLY3
3134  7450          SNA
3135  5341          JMP Z4
3136  7700          SMA CLA
3137  5351          JMP Z5
3140  5771          JMP ALDON2
3141  7300  Z4,     CLA CLL
3142  1770          TAD VMAXPL
3143  7417          LSR
3144  0000          0000
3145  7041          CIA
3146  1360          TAD S4
3147  7510          SPA
3150  5771          JMP ALDON2
3151  7300  Z5,     CLA CLL
3152  1173          TAD PLY3
3153  3772          DCA VMAXPH
3154  1174          TAD PLY4
3155  3770          DCA VMAXPL
3156  5771          JMP ALDON2
3157  0007  M7,     0007
3160  0000  S4,     0000
3161  0000  H1,     0000
3162  0000  H2,     0000
3163  0010  XIFIX,  0010
```

TABLE C-I - (Continued)

```
3164  0000  XI,     0000
3165  0000  XIPLUS, 0000
3166  7776  M7776,  7776
3170  3643
3171  3200
3172  3642
3173  3641
3174  3640
3175  4200
3176  1563
3177  1167
                    PAGE    /***START NEW PAGE
3200  7300  ALDON2, CLA CLL
3201  1121          TAD MAXMOD
3202  3172          DCA PLYI
3203  4777          JMS MULPSI      /CAL MAX 1ST MODE VALUES
3204  1173          TAD PLY3
3205  3414          DCA I REG14
3206  1174          TAD PLY4
3207  3414          DCA I REG14
3210  1776          TAD VMODH
3211  7041          CIA
3212  1173          TAD PLY3
3213  7510          SPA
3214  5231          JMP ALDON3
3215  7300          CLA CLL
3216  1775          TAD VMODL
3217  7041          CIA
3220  1174          TAD PLY4
3221  7510          SPA
3222  5231          JMP ALDON3
3223  7300          CLA CLL
3224  1173          TAD PLY3
3225  3776          DCA VMODH
3226  1174          TAD PLY4
3227  3775          DCA VMODL
3230  5231          JMP ALDON3
3231  7300  ALDON3, CLA CLL
3232  1774          TAD VISB
3233  1773          TAD VNOPKL
3234  3773          DCA VNOPKL
3235  7004          RAL
3236  1772          TAD VNOPKH
3237  3772          DCA VNOPKH
3240  4771          JMS DMEAN
3241  7300  ALDON5, CLA CLL
3242  1173          TAD PLY3
3243  3414          DCA I REG14
3244  1174          TAD PLY4
3245  3414          DCA I REG14
3246  1111          TAD PKBUF1      /WRITE INTERVAL SUMMARY DATA
3247  3770          DCA BUF1
3250  2770          ISZ BUF1        /(PEAK TO TROUGH) ON DECTAPE
3251  4767          JMS DECTAP      / USES 4 BLOCKS
3252  6002          IOF
```

TABLE C-I - (Continued)

```
3253  7300          CLA CLL
3254  1766          TAD M476
3255  3765          DCA M476A
3256  1274          TAD CORAD1
3257  3764          DCA CORADD      /LOG BOOK DATA READ, 2 BLOCKS
3260  1275          TAD NOBLK1
3261  3763          DCA NOBLKS
3262  1762          TAD M08
3263  1171          TAD RBLOK1
3264  3171          DCA RBLOK1
3265  1171          TAD RBLOK1
3266  4761          JMS DECRED
3267  1147          TAD LOCTAP
3270  3012          DCA TAPBUF
3271  5760          JMP WRI
3272  0000  NOINT,  0000
3273  0001  POSONE, 0001
3274  7200  CORAD1, 7200     /CORE BUFFER FOR LOGBOOK DATA
3275  7776  NOBLK1, 7776
3276  0000  SAVE3,  0000
3277  6000  CORAD2, 6000
3300  7775  M7772,  7775
3301  3720  M3720,  3720     /DECIMAL 2000
              /
              / START NEW SUBROUTINE HERE TAPE WRITE 2
```

TABLE C-I (Continued)

```
                /PART TWO OF MAGTAPE WRITE SUBROUTINE
                / FOR MERGING INTERVAL SUMMARY DATA
                /WITH LOG BOOK DATA
                /
3360  3400
3361  4394
3362  2723
3363  4311
3364  4307
3365  4562
3366  4361
3367  2703
3370  2710
3371  5420
3372  3634
3373  3635
3374  3611
3375  3645
3376  3644
3377  4200
                PAGE    /START NEW PAGE ****
3400  1412  WRI,  TAD I TAPBUF
3401  4777        JMS MGWRT1
3402  2776        ISZ M476A
3403  5200        JMP WRI
3404  1111        TAD PKBUF1
3405  3014        DCA MEG14
3406  1414        TAD I REG14
3407  3775        DCA CAND2       /MULTIPLIER
3410  1414        TAD I REG14
3411  3774        DCA SHIFT4
3412  1414        TAD I REG14
3413  3115        DCA NUMPKT
3414  1115        TAD NUMPKT
3415  3773        DCA VH
3416  1115        TAD NUMPKT
3417  7041        CIA
3420  3772        DCA VL
3421  1414        TAD I REG14
3422  3120        DCA NUMMOD
3423  1103        TAD SWT
3424  3104        DCA SWT1
3425  5230        JMP WRPK
3426  1120  WRPK1, TAD NUMMOD
3427  3773        DCA VH
3430  1113  WRPK,  TAD ZERO
3431  4777        JMS MGWRT1
3432  1113        TAD ZERO
3433  4777        JMS MGWRT1
3434  1773        TAD VH
3435  7417        LSR
3436  0007        0007
3437  4777        JMS MGWRT1
3440  1773        TAD VH
```

TABLE C-I - (Continued)

```
3441  4777          JMS MGWRT1
3442  2104          ISZ SWT1
3443  5226          JMP WRPK1
3444  1115          TAD NUMPKT
3445  3773          DCA VH
3446  1771          TAD M7772
3447  3050          DCA WR4
3450  4770          JMS WRANG1
3451  7300          CLA CLL
3452  1414          TAD I REG14
3453  3046          DCA WR2
3454  1414          TAD I REG14
3455  3047          DCA WR3
3456  1046          TAD WR2
3457  7710          SPA CLA
3460  1060          TAD ONE
3461  4777          JMS MGWRT1      /WRITE MEAN VALUE
3462  1046          TAD WR2
3463  7417          LSR
3464  0003          0003
3465  4777          JMS MGWRT1
3466  1047          TAD WR3
3467  7421          MQL
3470  1046          TAD WR2
3471  7413          SHL
3472  0003          0003
3473  4777          JMS MGWRT1
3474  1047          TAD WR3
3475  4777          JMS MGWRT1
3476  2014          ISZ REG14
3477  7300          CLA CLL
3500  1351          TAD M7444
3501  3352          DCA M7444A
3502  7300          CLA CLL
3503  4777          JMS MGWRT1
3504  2352          ISZ M7444A
3505  5302          JMP .-3
3506  7300          CLA CLL
3507  3767          DCA V1
3510  1115          TAD NUMPKT
3511  7650          SNA CLA
3512  5341          JMP WRT5
3513  1414  WRT4,   TAD I REG14
3514  3172          DCA PLY1
3515  4766          JMS MULPSI
3516  1113          TAD ZERO
3517  4777          JMS MGWRT1
3520  1173          TAD PLY3
3521  7417          LSR
3522  0003          0003
3523  4777          JMS MGWRT1
3524  1174          TAD PLY4
3525  7421          MQL
3526  1173          TAD PLY3
3527  7413          SHL
```

TABLE C-I - (Continued)

```
3530  0003          0003
3531  4777          JMS MGWRT1
3532  1174          TAD PLY4
3533  4777          JMS MGWRT1
3534  1137          TAD M4
3535  1767          TAD V1
3536  3767          DCA V1
3537  2772          ISZ VL
3540  5313          JMP WRT4
3541  7300  WRT5,   CLA CLL
3542  1120          TAD NUMMOD
3543  1765          TAD VNOMDL
3544  3765          DCA VNOMDL
3545  7004          RAL
3546  1764          TAD VNOMDH
3547  3764          DCA VNOMDH
3550  5763          JMP WRT7
3551  7610  M7444,  7610
3552  0000  M7444A, 0000
3563  3600
3564  3636
3565  3637
3566  4200
3567  3745
3570  3672
3571  3002
3572  3747
3573  3746
3574  4206
3575  4204
3576  4362
3577  4214

                    PAGE    /START NEW PAGE ******
3600  1345  WRT7,   TAD V1
3601  7041          CIA
3602  1777          TAD M3720       /2000 DECIMAL
3603  7450          SNA
3604  5211          JMP WRT8
3605  7300          CLA CLL
3606  4776          JMS MGWRT1
3607  2345          ISZ V1
3610  5200          JMP WRT7
3611  1075  WRT8,   TAD IRG
3612  4775          JMS REWIND
3613  6703          IWS
3614  5213          JMP .-1
                    / CLEAN UP INTERVAL BUFFER
                    /
3615  7300          CLA CLL
3616  4774          JMS HICORE
3617  7300          CLA CLL
3620  1111          TAD PKBUF1
3621  3014          DCA REG14
3622  1114          TAD NUMPK
3623  3112          DCA PKBUFT
```

TABLE C-I - (Continued)

```
3624  7300  CLENUP, CLA CLL
3625  3414          DCA I REG14
3626  2112          ISZ PKBUFT
3627  5224          JMP CLENUP
3630  7300          CLA CLL
3631  1123          TAD SWT
3632  3773          DCA VOY3
3633  5772          JMP CLEAN1
3634  0000  VNOPKH, 0000
3635  0000  VNOPKL, 0000
3636  0000  VNOMDH, 0000
3637  0000  VNOMDL, 0000
3640  0000  VSUMPH, 0000
3641  0000  VSUMPL, 0000
3642  0000  VMAXPH, 0000
3643  0000  VMAXPL, 0000
3644  0000  VMODH,  0000
3645  0000  VMODL,  0000
3646  0000  VMEAN1, 0000
3647  0000  VMEAN2, 0000
3650  0000  MEANSG, 0000
3651  7300  WRTSUM, CLA CLL
3652  4771          JMS LABEL1
3653  7300          CLA CLL
3654  1343          TAD M7610
3655  3344          DCA M7610A
3656  7300          CLA CLL
3657  4776          JMS MGWRT1
3660  2344          ISZ M7610A
3661  5256          JMP .-3
3662  7300          CLA CLL
3663  1167          TAD WR5
3664  1060          TAD ONE
3665  3014          DCA REG14
3666  1360          TAD M7773
3667  3050          DCA WR4
3670  4272          JMS WRANG1
3671  5320          JMP WRA1
3672  0000  WRANG1, 0000
3673  1414  WRAGN,  TAD I REG14
3674  3046          DCA WR2
3675  1414          TAD I REG14
3676  3047          DCA WR3
3677  7300          CLA CLL
3700  4776          JMS MGWRT1
3701  1046          TAD WR2
3702  7417          LSR
3703  0003          0003
3704  4776          JMS MGWRT1
3705  1047          TAD WR3
3706  7421          MQL
3707  1046          TAD WR2
3710  7413          SHL
3711  0003          0003
3712  4776          JMS MGWRT1
```

TABLE C-I - (Continued)

```
3713  1247          TAD VR3
3714  4776          JMS MGWRT1
3715  2050          ISZ VR4
3716  5273          JMP VRAGN
3717  5672          JMP I VRANG1
3720  7300  VRA1,    CLA CLL
3721  1246          TAD VMEAN1
3722  7710          SPA CLA
3723  1060          TAD ONE
3724  4776          JMS MGWRT1
3725  1246          TAD VMEAN1
3726  7417          LSR
3727  0003          0003
3730  4776          JMS MGWRT1
3731  1247          TAD VMEAN2
3732  7421          MQL
3733  1246          TAD VMEAN1
3734  7413          SHL
3735  0003          0003
3736  4776          JMS MGWRT1
3737  1247          TAD VMEAN2
3740  4776          JMS MGWRT1
3741  7300          CLA CLL
3742  5770          JMP END1
3743  7610  M7610,  7610
3744  0000  M7610A, 0000
3745  0000  V1,     0000
3746  0000  VH,     0000
3747  0000  VL,     0000
3750  0000  VALUE1, 0000
3751  2000  EOF,    2000
3752  0011  SHIFT,  0009
3753  3000  WAR,    3000
3754  6000  SRD,    6000
3755  7764  TWELVE, 7764
3756  0000  THOUS3, 0000
3757  0000  THOUS4, 0000
3760  7773  M7773,  7773
3770  4075
3771  4627
3772  5430
3773  5671
3774  5720
3775  1600
3776  4214
3777  3301

                    PAGE    /****START NEW PAGE
4000  0000  SETUP2, 0000
4001  7300          CLA CLL
4002  1060          TAD ONE
4003  3777          DCA INT1
4004  1060          TAD ONE
4005  3776          DCA START3
4006  3775          DCA COMBIN
4007  6211          CDF+10
```

TABLE C-I - (Continued)

```
4010  3626          DCA I VA
4011  3627          DCA I VB
4012  6201          CDF+00
4013  3774          DCA VNOPKH
4014  3773          DCA VNOPKL
4015  3772          DCA VSUMPH
4016  3771          DCA VSUMPL
4017  3770          DCA VMAXPH
4020  3767          DCA VMAXPL
4021  3766          DCA VMODH
4022  3765          DCA VMODL
4023  3764          DCA VNOMDL
4024  3763          DCA VNOMDH
4025  5600          JMP I SETUP2
4026  2054  VA,     2054
4027  2055  VB,     2055
```

TABLE C-I - (Continued)

```
                    /DIGITAL 8-21-F
                    /TWO'S COMPLEMENT, SINGLE PRECISION.
                    /SIGNED MULTIPLY ROUTINE WITH EAE.
                    /RETURN HIGH ORDER PRODUCT IN AC.
                    /LOW IN MP1.
                          MUY=7405
                          DVI=7407
                          NMI=7411
                          SHL=7413
                          MQA=7501
                          ASR=7415
                          LSR=7417
                          MQL=7421
                          SCA=7441
                          CAM=7621
4030  0000  MULT,    0
4031  7440          SZA     /TEST FOR ZERO MULTIPLIER.
4032  5235          JMP .+3
4033  3274          DCA MP1 /LOW ORDER PRODUCT.
4034  5272          JMP MPZ
4035  7100          CLL
4036  7510          SPA     /TEST FOR NEGATIVE MULTIPLIER.
4037  7061          CMA CML IAC
4040  7421          MQL     /MULTIPLIER TO MQ.
4041  1630          TAD I MULT
4042  7450          SNA     /TEST FOR ZERO MULTIPLICAND.
4043  5233          JMP MULT+3
4044  7510          SPA     /TEST FOR NEGATIVE MULTIPLICAND.
4045  7061          CMA CML IAC
4046  3252          DCA .+4 /STORE MULTIPLICAND.
4047  7004          RAL
4050  3274          DCA MP1 /SAVE LINK.
4051  7405          MUY
4052  0000  MP2,    0       /MULTIPLICAND.
4053  3252          DCA MP2 /SAVE HIGH ORDER PRODUCT.
4054  1274          TAD MP1
4055  7110          RAR CLL /RESTORE LINK.
4056  7501          MQA     /LOW ORDER PRODUCT TO AC.
4057  7420          SNL     /TEST SIGN OF PRODUCT.
4060  5270          JMP .+10
4061  7141          CLL CMA IAC     /COMPLEMENT PRODUCT.
4062  3274          DCA MP1 /LOW ORDER PRODUCT.
4063  1252          TAD MP2 /HIGH ORDER PRODUCT.
4064  7040          CMA
4065  7430          SZL
4066  7001          IAC
4067  5272          JMP MPZ
4070  3274          DCA MP1 /LOW ORDER PRODUCT
4071  1252          TAD MP2 /HIGH ORDER PRODUCT
4072  2230  MPZ,    ISZ MULT        /SET UP FOR EXIT.
4073  5630          JMP I MULT      /EXIT.
4074  0000  MP1,    0
```

TABLE C-I - (Continued)

```
                        /
                        /START VOY1 HERE
                        /
4075  1075  END1,    TAD IRG
4076  4762            JMS REWIND
4077  6703            IWS
4100  5277            JMP .-1
4101  1761            TAD NOVOY
4102  7041            CIA
4103  1760            TAD VOYNO
4104  7650            SNA CLA
4105  5327            JMP END3A
4106  7001            IAC
4107  3757            DCA PASNO
4110  7001            IAC
4111  3756            DCA INTNO
4112  4755            JMS VOYEND
4113  2760            ISZ VOYNO
4114  1754            TAD SET2
4115  1326            TAD F08
4116  3754            DCA SET2
4117  1754            TAD SET2
4120  3753            DCA SET3
4121  7001            IAC
4122  3752            DCA DEC1
4123  4751            JMS NEWDEC
4124  7402            HLT
4125  5750            JMP BEGIN1
4126  0010  F08, '    0010
4127  4755  END3A,    JMS VOYEND
4130  1747            TAD EOF
4131  4762            JMS REWIND
4132  6703            IWS
4133  5332            JMP .-1
4134  4762            JMS REWIND
4135  6703            IWS
4136  5335            JMP .-1
4137  4746            JMS EOFTYP      /TYPE END OF JOB
4140  7402            HLT
4141  5745            JMP BEGIN-1
4142  7150  ACA,      7150
4143  7160  ACB,      7160
4144  7170  ACC,      7170
4145  0000
4146  5000
4147  3751
4150  0431
4151  5621
4152  0540
4153  4624
4154  4623
4155  4705
4156  4622
4157  4621
4160  4620
```

TABLE C-I - (Continued)

```
4161  4555
4162  1600
4163  3636
4164  3637
4165  3645
4166  3644
4167  3643
4170  3642
4171  3641
4172  3640
4173  3635
4174  3634
4175  2642
4176  5427
4177  0536
                        PAGE            /START NEW PAGE ****
                        /
                        / MULTIPLY SUBROUTINE
                        / FOR POSITIVE NO ONLY
                        /
4200  0000  MULPSI,    0000
4201  1172              TAD PLY1
4202  7421              MQL
4203  7405              MUY
4204  0000  CAND2,     0000
4205  7417              LSR
4206  0000  SHIFT4,    0000
4207  3173              DCA PLY3
4210  7501              MQA
4211  3174              DCA PLY4
4212  7300              CLA CLL
4213  5600              JMP I MULPSI
4214  0000  MGWRT1,    0000
4215  4777              JMS MGWRTE
4216  6703              IWS
4217  5216              JMP .-1
4220  5614              JMP I MGWRT1
                        /
                        /FOR CALIBRATION ROUTINE
                        /
4221  0000  CALVAL,    0000
4222  7300              CLA CLL
4223  1113              TAD ZERO
4224  3244              DCA SHIFT2
4225  1776              TAD VALUE1
4226  3234              DCA VISOR2
4227  1151              TAD THOUS2
4230  3233              DCA LOWV02
4231  1150              TAD THOUS1
4232  4542  TRY1,      JMS I LNKDIV
4233  0000  LOWV02,    0000
4234  0000  VISOR2,    0000
4235  7420              SNL     /SKIP IF LINK NOT =0
4236  5255              JMP OKOK
4237  7300              CLA CLL
```

TABLE C-I - (Continued)

```
4240  1151          TAD THOUS2
4241  7421          MQL
4242  1150          TAD THOUS1
4243  7417          LSR
4244  0000  SHIFT2, 0000
4245  3775          DCA THOUS3
4246  7501          MQA
4247  3774          DCA THOUS4
4250  2244          ISZ SHIFT2
4251  1774          TAD THOUS4
4252  3233          DCA LOWD2
4253  1775          TAD THOUS3
4254  5232          JMP TRY1
4255  3270  OKOK,   DCA CAND1
4256  1244          TAD SHIFT2
4257  7041          CIA
4260  1773          TAD SHIFT
4261  3277          DCA SHIFT3
4262  7300          CLA CLL
4263  5621          JMP I CALVAL
                    /
                    /CAL MULTIPLICATION ROUTINE
                    /
4264  0000  MULVAL, 0000
4265  7300          CLA CLL
4266  1074          TAD BUFFPT
4267  4543          JMS I LKMULT
4270  0000  CAND1,  0000
4271  3132          DCA TAP1
4272  7300          CLA CLL
4273  1720          TAD I M2A
4274  7421          MQL
4275  1132          TAD TAP1
4276  7415          ASR
4277  0006  SHIFT3, 6000
4300  3132          DCA TAP1
4301  7501          MQA
4302  3133          DCA TAP2
4303  5664          JMP I MULVAL
                    /
                    / DECTAPE READ
                    /FOR LOGBOOK DATA
                    /
4304  0000  DECRED, 0000
4305  3512          DCA READ1
4306  4717          JMS I R128Y
4307  0000  CORADD, 0000          /STARTING CORE ADDRESS
4310  4000          4000          /FIELD 0 ,UNIT 4
4311  0030  NOBLKS, 0030          /NO. OF BLOCKS
4312  0000  READ1,  0000    /BLOCK NO TO START AT
4313  7300          CLA CLL
4314  4452          JMS I DONE
4315  6002          IOF
4316  5704          JMP I DECRED
4317  0200  R128Y,  R128
```

TABLE C-I - (Continued)

```
4320  4074  M2A,    MP1
4321  0030  BUFP,   0030
4322  0000  LABEL,  0000
4323  7300          CLA CLL
4324  1370          TAD M7200
4325  3352          DCA M476A
4326  1366          TAD DIFF
4327  3367          DCA DIFF1
4330  1364          TAD CORAD3
4331  3307          DCA CORADD
4332  1363          TAD NOBLK2
4333  3311          DCA NOBLKS
4334  1170          TAD RBLOK
4335  4304          JMS DECRED
4336  1772          TAD DEC1
4337  7710          SPA CLA
4340  5357          JMP WR6
4341  1365          TAD LOCTP1
4342  3012          DCA TAPBUF
4343  1412  WR1,    TAD I TAPBUF
4344  4214          JMS MGWRT1
4345  2362          ISZ M476A
4346  5343          JMP WR1
4347  7300          CLA CLL
4350  4214          JMS MGWRT1
4351  2367          ISZ DIFF1
4352  5347          JMP .-3
4353  1075          TAD IRG
4354  4771          JMS REWIND
4355  6703          IWS
4356  5355          JMP .-1
4357  2171  WR6,    ISZ RBLOK1
4360  5722          JMP I LABEL
4361  7400  M476,   7400
4362  0000  M476A,  0000
4363  7775  NOBLK2, 7775
4364  7000  CORAD3, 7000
4365  6777  LOCTP1, 6777
4366  7614  DIFF,   7614
4367  0000  DIFF1,  0000
4370  7200  M7200,  7200
```

TABLE C-I - (Continued)

```
                        /
                        / START CONT HERE
                        /
4371  1600
4372  0540
4373  3752
4374  3757
4375  3756
4376  3750
4377  1630
                        PAGE    /***START NEW PAGE
                        /
                        /READ DECTAPE ON NO VOY, PASSES AND INTERVALS
                        /TO BE DELETED
                        /
4400  0000  SETUP1,     0000
4401  7300              CLA CLL
4402  1346              TAD HEDCNT
4403  3017              DCA XR17
4404  1347              TAD M7322
4405  3012              DCA TAPBUF
4406  1412              TAD I TAPBUF
4407  7300              CLA CLL
4410  1777              TAD VY1
4411  0343              AND M0017
4412  7421              MQL
4413  7405              MUY
4414  0012              0012
4415  7701              MQA CLA
4416  3344              DCA HOLD1
4417  1776              TAD VY2
4420  0343              AND M0017
4421  1344              TAD HOLD1
4422  3355              DCA NOVOY
4423  1355              TAD NOVOY
4424  3417              DCA I XR17
4425  1350              TAD M4A
4426  3351              DCA M4AA
4427  1412              TAD I TAPBUF
4430  0343              AND M0017
4431  3417              DCA I XR17
4432  1350              TAD M4A
4433  3352              DCA M4AAA
4434  1412              TAD I TAPBUF
4435  0343              AND M0017
4436  7421              MQL
4437  7405              MUY
4440  0012              0012
4441  7701              MQA CLA
4442  3344              DCA HOLD1
4443  1412              TAD I TAPBUF
4444  0343              AND M0017
4445  1344              TAD HOLD1
4446  3417              DCA I XR17
4447  2352              ISZ M4AAA
```

TABLE C-I - (Continued)

```
4450  5234              JMP .-14
4451  2351              ISZ M4AA
4452  5227              JMP .-23
4453  1356              TAD M40
4454  3357              DCA M40A
4455  1412  LONG7,      TAD I TAPBUF
4456  0343              AND M0017
4457  1342              TAD VYCNT       /COMPUTED VOYAGE NO -1
4460  7413              SHL
4461  0002              0002            /VOY
4462  3344              DCA HOLD1
4463  1412              TAD I TAPBUF
4464  0343              AND M0017
4465  1344              TAD HOLD1
4466  3417              DCA I XR17
4467  1412              TAD I TAPBUF
4470  0343              AND M0017
4471  7421              MQL
4472  7405              MUY
4473  0012              0012
4474  7701              MQA CLA
4475  3344              DCA HOLD1
4476  1412              TAD I TAPBUF
4477  0343              AND M0017
4500  1344              TAD HOLD1
4501  3344              DCA HOLD1
4502  1412              TAD I TAPBUF
                        /
                        /IF INTERVAL WORD A
                        / -(4000) A DELETION
                        /2000, A LONG INTERVAL
                        / 0011 NOTHING
                        /
4503  3345              DCA HOLD2
4504  1345              TAD HOLD2
4505  1360              TAD DELETE
4506  7640              SZA CLA         /A DELETION
4507  5314              JMP LONG1
4510  1361              TAD M4000
4511  1344              TAD HOLD1
4512  3417              DCA I XR17
4513  5366              JMP LONG6
4514  1345  LONG1,      TAD HOLD2
4515  1362              TAD LONG5
4516  7640              SZA CLA
4517  5324              JMP HALT
4520  1363              TAD M2000
4521  1344              TAD HOLD1
4522  3417              DCA I XR17
4523  5366              JMP LONG6
4524  1345  HALT,       TAD HOLD2
4525  1365              TAD HALT1
4526  7640              SZA CLA
4527  5334              JMP NOTIN
4530  1364              TAD M1000
```

TABLE C-I - (Continued)

```
4531  1344          TAD HOLD1
4532  3417          DCA I XR17
4533  5366          JMP LONG6
4534  7300  NOTIN,  CLA CLL
4535  1344          TAD HOLD1
4536  3417          DCA I XR17
4537  5366          JMP LONG6
4540  7300  LONG5,  CLA CLL
4541  5600          JMP I SETUP1
4542  0000  VYCNT,  0000
4543  0017  M0017,  0017
4544  0000  HOLD1,  0000
4545  0000  HOLD2,  0000
4546  6777  HEDCNT, 6777
            XR17=0017
4547  7222  M7322,  7222
4550  7774  M4A,    7774
4551  0000  M4AA,   0000
4552  0000  M4AAA,  0000
4553  7756  M18A,   7756
4554  0000  M18AA,  0000
4555  0000  NOVOY,  0000
4556  7730  M40,    7730
4557  0000  M40A,   0000
4560  7474  DELETE, 7474
4561  4000  M4000,  4000
4562  7455  LONG5,  7455
4563  2000  M2000,  2000
4564  1000  M1000,  1000
4565  7470  HALT1,  7470
4566  2357  LONG6,  ISZ M40A
4567  5255          JMP LONG7
4570  1775          TAD DEC1
4571  7700          SMA CLA
4572  5774          JMP LONG2
4573  5340          JMP LONG8
4574  4600
4575  0540
4576  5316
4577  5315

                    PAGE
                    /
                    /ABOVE ROUTINE HANDLES 40 DELETIONS AND 0n LONG
                    / INTERVALS
                    /USES 98 DEC OR 142 OCTAL LOCATIONS 7000-7142
                    /
4600  1777  LONG8,  TAD DEC1
4601  7640          SZA CLA
4602  5205          JMP LONGA
4603  7001          IAC
4604  3220          DCA VOYNO
4605  7001  LONGA,  IAC
4606  3221          DCA PASNO       /PASS 1
4607  7001          IAC
4610  3222          DCA INTNO       /INTERVAL 1
```

```
4611  1223          TAD SET2
4612  3224          DCA SET3
4613  1225          TAD M7001
4614  3776          DCA M7001A
4615  1226          TAD M7002
4616  3775          DCA M7002A
4617  5774          JMP LONG5
4620  0000  VOYNO,  0000
4621  0000  PASNO,  0000
4622  0000  INTNO,  0000
4623  0011  SET2,   0011      /V1,P1
4624  0000  SET3,   0000
4625  7001  M7001,  7001
4626  7002  M7002,  7002
```

TABLE C-I - (Continued)

```
                                / START SUBROUTINE ADT
4627  0000  LABEL1,  0000
4630  7300          CLA CLL
4631  1111          TAD PKBUF1
4632  3014          DCA REG14
4633  1167          TAD VRS           /WRITE VOYAGE SUMMARY
4634  1060          TAD ONE           /DATA TO DECTAPE
4635  3010          DCA IR1
4636  1303          TAD M10A
4637  3304          DCA M10AA
4640  1410          TAD I IR1
4641  3414          DCA I REG14
4642  2304          ISZ M10AA
4643  5240          JMP .-3
4644  1773          TAD MEANSG
4645  3414          DCA I REG14
4646  1772          TAD VMEAN1
4647  3414          DCA I REG14
4650  1771          TAD VMEAN2
4651  3414          DCA I REG14
4652  7300          CLA CLL
4653  1111          TAD PKBUF1
4654  3770          DCA BUF1
4655  2770          ISZ BUF1
4656  4767          JMS DECTAP
4657  6002          IOF
4660  7300          CLA CLL
4661  1766          TAD M476
4662  3765          DCA M476A
4663  1764          TAD CORAD1
4664  3763          DCA CORADD        /READ VOYAGE LOGBOOK DATA
4665  1762          TAD NOBLK1
4666  3761          DCA NOBLKS
4667  1760          TAD M88
4670  1171          TAD RBLOK1
4671  3171          DCA RBLOK1
4672  1171          TAD RBLOK1
4673  4757          JMS DECRED
4674  1147          TAD LOCTAP
4675  3012          DCA TAPBUF
4676  1412          TAD I TAPBUF
4677  4756          JMS MGURT1
4700  2765          ISZ M476A
4701  5276          JMP .-3
4702  5627          JMP I LABEL1
4703  7770  M10A,   7768
4704  0000  M10AA,  0000
                    /
                    /END OF VOYAGE LABEL
                    /
4705  0000  VOYEND,  0000
4706  7300          CLA CLL
4707  4755          JMS CRLF
4710  1327          TAD CHAR3
4711  3010          DCA IR1
```

TABLE C-I - (Continued)

```
4712  1326          TAD VPAS
4713  3754          DCA C1
4714  1410  GET3,   TAD I IR1
4715  4753          JMS TYPE
4716  2754          ISZ C1
4717  5314          JMP GET3
4720  1220          TAD VOYNO
4721  1752          TAD M260
4722  4753          JMS TYPE
4723  4755          JMS CRLF
4724  2751          ISZ VYCNT
4725  5705          JMP I VOYEND
4726  7760  VPAS,   7760
4727  4727  CHAR3,  .
4730  0252          252
4731  0252          252       /*
4732  0305          305       /END
4733  0316          316
4734  0304          304
4735  0240          240
4736  0317          317       /OF
4737  0306          306
4740  0240          240
4741  0326          326       /VOYAGE
4742  0317          317
4743  0331          331
4744  0301          301
4745  0307          307
4746  0305          305
4747  0240          240
4751  4542
4752  5546
4753  5515
4754  5545
4755  5277
4756  4214
4757  4304
4760  2723
4761  4311
4762  3275
4763  4307
4764  3274
4765  4362
4766  4361
4767  2783
4770  2710
4771  3647
4772  3646
4773  3650
4774  4540
4775  5512
4776  5513
4777  0540
                    PAGE      /**START NEW PAGE
                    /
```

TABLE C-I - (Continued)

```
                    / END OF JOB LABEL
                    /
5000  0000  EOFTYP, 0000
5001  7300          CLA CLL
5002  4777          JMS CRLF
5003  1216          TAD CHAR4
5004  3010          DCA IR1
5005  1215          TAD EOFEOF
5006  3776          DCA C1
5007  1410  GET4,    TAD I IR1
5010  4775          JMS TYPE
5011  2776          ISZ C1
5012  5207          JMP GET4
5013  4777          JMS CRLF
5014  5600          JMP I EOFTYP
5015  7756  EOFEOF, 7756
5016  5016  CHAR4,  .
5017  0240          240
5020  0240          240
5021  0252          252
5022  0252          252       /*
5023  0240          240
5024  0305          305       /END
5025  0316          316
5026  0304          304
5027  0240          240
5030  0317          317       /OF
5031  0306          306
5032  0240          240
5033  0312          312       /JOB
5034  0317          317
5035  0302          302
5036  0240          240
5037  0252          252       /*
5040  0252          252
                    /
                    /INTERVAL DELETION CHECK
                    /
5041  0000  INTCHK, 0000
5042  7300          CLA CLL
5043  3346          DCA NOGOIN
5044  2343          ISZ INTN1
5045  1774          TAD M40
5046  3332          DCA M18C
5047  1335          TAD M7025       /CORE LOCATION FOR INT DELETE TABLE
5050  3017          DCA XR17
5051  3330          DCA LONGER
5052  1417  AGAN1A, TAD I XR17
5053  7041          CIA
5054  1773          TAD SET3
5055  7650          SNA CLA
5056  5262          JMP LOAFB1
5057  2017          ISZ XR17
5060  5273          JMP AGANB
5061  5252          JMP AGAN1A
```

TABLE C-I - (Continued)

```
5062  7300  LOAFB1, CLA CLL
5063  1417          TAD I XR17
5064  3344          DCA HOLD3
5065  1344          TAD HOLD3
5066  0345          AND M0777
5067  7041          CIA
5070  1343          TAD INTN1
5071  7650          SNA CLA
5072  5277          JMP CHECKE      /DELETE OR LONG INTERVAL
5073  7300  AGAND,  CLA CLL
5074  2332          ISZ M18C
5075  5252          JMP AGAN1A
5076  5641          JMP I INTCHK
5077  7300  CHECKE, CLA CLL
5100  7621          CAM
5101  1344          TAD HOLD3
5102  7413          SHL
5103  0000          0000
5104  7430          SZL
5105  5336          JMP LOAFBD
5106  7413          SHL
5107  0000          0000
5110  7420          SNL
5111  5325          JMP HALT3
5112  7300          CLA CLL
5113  1060          TAD ONE
5114  3330          DCA LONGER
5115  7300  HALT4,  CLA CLL
5116  7240          STA
5117  1017          TAD XR17
5120  3342          DCA LOC1
5121  3742          DCA I LOC1
5122  2342          ISZ LOC1
5123  3742          DCA I LOC1
5124  5641          JMP I INTCHK
5125  7402  HALT3,  HLT
5126  7300          CLA CLL
5127  5315          JMP HALT4
5130  0000  LONGER, 0000
5131  0000  M18B,   0000
5132  0000  M18C,   0000
5133  6030  M1000A, 6030
5134  0000  M1000B, 0000
5135  7624  M7025,  7024
5136  7300  LOAFBD, CLA CLL
5137  1060          TAD ONE
5140  3346          DCA NOGOIN
5141  5315          JMP HALT4
5142  0000  LOC1,   0200
5143  0022  INTN1,  0000
5144  0000  HOLD3,  0000
5145  0777  M0777,  0777
5146  0000  NOGOIN, 0000
                    /
                    /END OF SUBROUTINE ADT2
```

```
5173  4624
5174  4556
5175  5515
5176  5545
5177  5277
```

PAGE

TABLE C-I - (Continued)

```
5200  0000  START,  0000
5201  7300        CLA CLL
5202  1777        TAD ACA
5203  3066        DCA AC
5204  1776        TAD ACB
5205  3067        DCA L
5206  1775        TAD ACC
5207  3070        DCA PCREG
5210  3774        DCA COMBIN
5211  3773        DCA INTN1
5212  4277        JMS CRLF
5213  1233        TAD CHARAC
5214  3010        DCA IR1
5215  1230        TAD M6
5216  3231        DCA COUNT
5217  1410  NEXT, TAD I IR1
5220  4772        JMS TYPE
5221  2231        ISZ COUNT
5222  5217        JMP NEXT
5223  4277        JMS CRLF
5224  4306        JMS ACCEPT
5225  5600        JMP I START
                  IR1=10
5226  0212  K212, 212          /ASCII FOR LF
5227  0215  K215, 215          /ASCII FOR CR
5230  7735  M6,   7735
5231  0000  COUNT, 0000
5232  7447  MY,   7447          /ASCII Y-
5233  5233  CHARAC,.
5234  0304        304
5235  0305        305
5236  0303        303
5237  0324        324
5240  0301        301
5241  0320        320
5242  0305        305
5243  0240        240
5244  0317        317
5245  0316        316
5246  0240        240
5247  0264        264
5250  0277        277
5251  0240        240
5252  0324        324
5253  0317        317
5254  0324        324
5255  0301        301
5256  0314        314
5257  0240        240
5260  0316        316
5261  0317        317
5262  0256        256
5263  0240        240
5264  0317        317
```

TABLE C-I (Continued)

```
5265  0304        304
5266  0240        240
5267  0306        306
5270  0317        317
5271  0331        331
5272  0301        301
5273  0307        307
5274  0305        305
5275  0323        323
5276  0275        275
5277  0000  CRLF,  0        /CARRIAGE RETURN LINE FEED
5300  7300        CLA CLL
5301  1227        TAD K215
5302  4772        JMS TYPE
5303  1226        TAD K212
5304  4772        JMS TYPE
5305  5677        JMP I CRLF
                  /ROUTINE TO ACCEPT TWO CHARAC
5306  0000  ACCEPT, 0
5307  7300        CLA CLL
5310  4317        JMS LISN
5311  3315        DCA VY1
5312  4317        JMS LISN
5313  3316        DCA VY2
5314  5706        JMP I ACCEPT
5315  0000  VY1,   0000
5316  0000  VY2,   0000
5317  0000  LISN,  0        /INPUT READ SUBROUTINE
5320  6031        KSF
5321  5320        JMP .-1
5322  6036        KRB
5323  6046        TLS
5324  6041        TSF
5325  5324        JMP .-1
5326  5717        JMP I LISN
5327  0000  PASHED, 0000
5330  4277        JMS CRLF
5331  1347        TAD CHAR2
5332  3010        DCA IR1
5333  1346        TAD MPAS
5334  3771        DCA C1
5335  1410  GET2, TAD I IR1
5336  4772        JMS TYPE
5337  2771        ISZ C1
5340  5335        JMP GET2
5341  1770        TAD PASNO
5342  1767        TAD M260
5343  4772        JMS TYPE
5344  4277        JMS CRLF
5345  5727        JMP I PASHED
5346  7763  MPAS,  7763
5347  5347  CHAR2,  .
5350  0240        240
5351  0305        305      /END
5352  0316        316
```

TABLE C-I - (Continued)

```
5353  0304          304
5354  0240          240
5355  0317          317     /OF
5356  0306          306
5357  0240          240
5360  0320          320
5361  0301          301     /PASS
5362  0323          323
5363  0323          323
5364  0240          240
5367  5546
5370  4621
5371  5545
5372  5515
5373  5143
5374  2642
5375  4144
5376  4143
5377  4142
                    PAGE            /*****START NEW PAGE
                /ROUTINE DIVIDES MEAN VALUE BY 12000 PTS READ
                /
5400  0000  DMEAN,  0000
5401  7390          CLA CLL
5402  1146          TAD MEAN2
5403  7421          MQL
5404  1145          TAD MEAN1
5405  7415          ASR
5406  0012          0012
5407  7701          MQA CLA
5410  3074          DCA BUFFPT
5411  2227          ISZ START3
5412  5220          JMP START5
5413  3173          DCA PLY3
5414  3174          DCA PLY4
5415  3145          DCA MEAN1
5416  3146          DCA MEAN2
5417  5600          JMP I DMEAN
5420  7300  START5, CLA CLL
5421  4777          JMS MULVAL
5422  1132          TAD TAP1
5423  3173          DCA PLY3
5424  1133          TAD TAP2
5425  3174          DCA PLY4
5426  5600          JMP I DMEAN
5427  0000  START3, 0000
5430  7300  CLEAN1, CLA CLL
5431  1712          TAD I M7002A
5432  7041          CIA
5433  1776          TAD INTNO
5434  7650          SNA CLA
5435  5240          JMP CLEN1
5436  2776          ISZ INTNO
5437  5775          JMP BEGIN2
5440  7300  CLEN1,  CLA CLL
```

TABLE C-I - (Continued)

```
5441  6212          CIF+10
5442  4576          JMS I DATAF
5443  7300          CLA CLL
5444  6202          CIF+00
5445  6201          CDF+00
5446  7300          CLA CLL
5447  7001          IAC
5450  3776          DCA INTNO
5451  3774          DCA INTN1
5452  2312          ISZ M7002A
5453  4773          JMS PASHED
5454  1713          TAD I M7001A    /TYPE END OF PASS
5455  7041          CIA
5456  1772          TAD PASNO
5457  7650          SNA CLA
5460  5302          JMP CLEN2
5461  2772          ISZ PASNO
5462  1060          TAD ONE
5463  3771          DCA DEC1
5464  4770          JMS NEWDEC
5465  6212          CIF+10
5466  4575          JMS I HEADER
5467  7300          CLA CLL
5470  6201          CDF+00
5471  6202          CIF+00
5472  2767          ISZ SET3
5473  1060          TAD ONE
5474  3766          DCA INT1
5475  1060          TAD ONE
5476  3227          DCA START3
5477  3765          DCA COMBIN
5500  7402          HLT
5501  5775          JMP BEGIN2
5502  7300  CLEN2,  CLA CLL
5503  1313          TAD M7001A
5504  1314          TAD M5A
5505  3313          DCA M7001A
5506  7001          IAC
5507  1313          TAD M7001A
5510  3312          DCA M7002A
5511  5764          JMP WRTSUM
5512  0000  M7002A, 0000
5513  0000  M7001A, 0000
5514  0005  M5A,    0005
5515  0000  TYPE,   0000    /TYPE A CHARACTER
5516  6046          TLS
5517  6041          TSF
5520  5317          JMP .-1
5521  7300          CLA CLL
5522  6042          TCF
5523  5715          JMP I TYPE
5524  0000  VOYHED, 0000
5525  7300          CLA CLL
5526  4763          JMS CRLF
5527  1762          TAD CHAR1
```

TABLE C-I - (Continued)

```
5530  3010          DCA IR1
5531  1344          TAD MVOY
5532  3345          DCA C1
5533  1410  GET1,   TAD I IR1
5534  4315          JMS TYPE
5535  2345          ISZ C1
5536  5333          JMP GET1
5537  1761          TAD VOYNO
5540  1346          TAD M260
5541  4315          JMS TYPE
5542  4763          JMS CRLF
5543  5724          JMP I VOYHED
5544  7760  MVOY,   7760
5545  0000  C1,     0000
5546  0260  M260,   0260
5561  4620
5562  5600
5563  5277
5564  3651
5565  2642
5566  0536
5567  4624
5570  5621
5571  0540
5572  4621
5573  5327
5574  5143
5575  0440
5576  4622
5577  4264
                    PAGE
5600  5600  CHAR1,  .
5601  0323          323       /S
5602  0324          324       / TART
5603  0301          301
5604  0322          322
5605  0324          324
5606  0240          240
5607  0317          317       /0F
5610  0306          306
5611  0240          240
5612  0326          326       /VOYAGE
5613  0317          317
5614  0331          331
5615  0301          301
5616  0307          307
5617  0305          305
5620  0240          240
```

TABLE C-I - (Continued)

```
                    /
                    /MULTI-DECTAPE ENVIORMENT
                    /
5621  0000  NEWDEC,  0000
5622  7300          CLA CLL
5623  4777          JMS CRLF
5624  1274          TAD CHAR5
5625  3010          DCA IR1
5626  1272          TAD NUM
5627  3273          DCA C2
5630  1410  NEWDC1,  TAD I IR1
5631  4776          JMS TYPE
5632  2273          ISZ C2
5633  5230          JMP NEWDC1
5634  4243          JMS ACEPT1
5635  7300          CLA CLL
5636  4777          JMS CRLF
5637  7300          CLA CLL
5640  6042          TCF
5641  6032          KCC
5642  5621          JMP I NEWDEC
5643  0000  ACEPT1,  0
5644  7300          CLA CLL
5645  4775          JMS LISN
5646  1774          TAD MY
5647  7450          SNA
5650  5255          JMP ACEPT2
5651  7300          CLA CLL
5652  6032          KCC
5653  6042          TCF
5654  5643          JMP I ACEPT1
5655  7300  ACEPT2,  CLA CLL
5656  1170          TAD RBLOK
5657  3171          DCA RBLOK1
5660  1062          TAD BLCK
5661  3063          DCA BLOK
5662  6032          KCC
5663  6042          TCF
5664  4773          JMS SEARCH
5665  4772          JMS LABEL
5666  4771          JMS SETUP1
5667  7300          CLA CLL
5670  5643          JMP I ACEPT1
5671  0000  VOY3,    0000
5672  7755  NUM,     7755
5673  0000  C2,      0000
5674  5674  CHAR5,   .
5675  0315          315
5676  0317          317
5677  0325          325
5700  0316          316
5701  0324          324
5702  0240          240
5703  0316          316
5704  0305          305
```

TABLE C-I - (Continued)

```
5705  0327          327
5706  0240          240
5707  0304          304
5710  0305          305
5711  0303          303
5712  0324          324
5713  0301          301
5714  0320          320
5715  0305          305
5716  0277          277
5717  0240          240
5720  0000  HICORE,  0000
5721  7300          CLA CLL
5722  1356          TAD FPKBUF
5723  3014          DCA REG14
5724  1357          TAD LOCAF
5725  3010          DCA IR1
5726  1162          TAD TENPTS
5727  3155          DCA POINT1
5730  6201  OUTPT2,  CDF+00
5731  1414          TAD I REG14
5732  6211          CDF+10
5733  3410          DCA I IR1
5734  2155          ISZ POINT1
5735  5330          JMP OUTPT2
5736  1360          TAD INTNOF
5737  3014          DCA REG14
5740  1361          TAD A1
5741  3010          DCA IR1
5742  1362          TAD F7775
5743  3155          DCA POINT1
5744  6201  DAT1,    CDF+00
5745  1414          TAD I REG14
5746  6211          CDF+10
5747  3410          DCA I IR1
5750  2155          ISZ POINT1
5751  5344          JMP DAT1
5752  6202          CIF+00
5753  6201          CDF+00
5754  7300          CLA CLL
5755  5720          JMP I HICORE
5756  6001  FPKBUF,  6001
5757  0017  LOCAF,   0017
5760  7216  INTNOF,  7216
5761  0037  A1,      0037
5762  7775  F7775,   7775
5771  4400
5772  4322
5773  2671
5774  5232
5775  5317
5776  5515
5777  5277
```

TABLE C-I - (Cont.)

```
.PALJ
*OUT-S:LIST
*
*IN-S:LIS1,S:DIG1,S:DEC1,S:VOLM
*
*
*
*
*OPT-T


A1      1226
A2      1311
BLANK   1433
BLANK1  1440
BLANK2  1312
B1      0075
CAM     7621
CHNGF   0244
CRLF1   0222
C2      0246
C3      0247
DATA    1000
DATA1   1400
DATA2   1077
DATA3   1112
DATA4   1120
DATA5   1125
DATA6   1136
DATA7   1230
DATA9   1040
DAT1    1200
DAT2    1022
FOCTAL  0064
FPKBUF  0076
FSWT    0112
FX1     0010
FX2     0011
FX3     0012
F0100   0114
F0260   0113
F0360   0116
F1A     0065
F1B     0071
F10A    0066
F10B    0072
F100A   0067
F100B   0073
F1000A  0070
F1000B  0074
F2      0050
F2000A  0104
F240    0054
F260    0245
F4A     0105
```

TABLE C-I - (Continued)

F5	0107	TITLF	0400
F6	0110	TYPF	0231
F7	0056	TYPF1	0232
F7700	1227	T1	0057
F7766	0100	T2	0060
F7766A	0101	T3	0061
F7775	0102	T4	0062
F7775A	0103	T5	0063
F96	1062	UNITS	1425
GET5	0212	VA	2054
HEAD1	0200	VB	2055
HOLD1	2045	VYMEAN	2000
HOLD2	2046	VY1	2052
HOLD4	2047	VY2	2053
HUND	1411	ZOUT	2043
HUND1	1412	Z1	2026
H1A	0051	Z2	2012
H2A	0052		
H3	0053		
H4	0106		
INTNF1	0115		
INTNO	0040		
INTNOF	0055		
K212F	0241		
K215F	0242		
LOCAF	0077		
LSR	7417		
MQA	7501		
MQL	7421		
MSGN	0111		
MSGN1	1313		
MUY	7405		
NOP1	0243		
OUTPT2	1024		
PLY3	2050		
PLY4	2051		
PRINT	1446		
PRINT2	1475		
PRINT3	1511		
PRINT4	1535		
RMS	1047		
RMS1	1301		
SHL	7413		
TENS	1417		
TENS1	1420		
THOUS	1404		

TABLE C-I - (Continued)

```
                        FIELD 1
                        *200
0200  0000  HEAD1,     000
0201  6211              CDF+10
0202  7300              CLA CLL
0203  1243              TAD NOP1
0204  3232              DCA TYPF1
0205  1777              TAD TITLE
0206  3010              DCA FX1
0207  1246              TAD C2
0210  3247              DCA C3
0211  4222              JMS CRLF1
0212  1410  GET5,       TAD I FX1
0213  4231              JMS TYPF
0214  2247              ISZ C3
0215  5212              JMP GET5
0216  4222              JMS CRLF1
0217  6201              CDF+00
0220  6202              CIF+00
0221  5600              JMP I HEAD1
0222  0000  CRLF1,      0
0223  7300              CLA CLL
0224  1242              TAD K215F
0225  4231              JMS TYPF
0226  1241              TAD K212F
0227  4231              JMS TYPF
0230  5622              JMP I CRLF1
0231  0000  TYPF,       0
0232  7000  TYPF1,      NOP
0233  6046              TLS
0234  6041              TSF
0235  5234              JMP .-1
0236  7300              CLA CLL
0237  6042              TCF
0240  5631              JMP I TYPF
0241  0212  K212F,      212
0242  0215  K215F,      215
0243  7000  NOP1,       7000
0244  1245  CHNGF,      TAD F260
0245  0260  F260,       260
                        FX1=0010
0246  7561  C2,         7561
0247  0000  C3,         0000
0377  0400
                        PAGE
0400  0400  TITLE,      .
0401  0311              311          /INTERVAL
0402  0316              316
0403  0324              324
0404  0305              305
0405  0322              322
0406  0326              326
0407  0301              301
0410  0314              314
```

TABLE C-I - (Continued)

```
0411  0240              240
0412  0240              240
0413  0316              316          /NO.
0414  0317              317
0415  0256              256
0416  0240              240
0417  0320              320          /PEAK
0420  0305              305
0421  0301              301
0422  0313              313
0423  0240              240
0424  0324              324          /TO
0425  0317              317
0426  0240              240
0427  0324              324          /TROUGHS
0430  0322              322
0431  0317              317
0432  0325              325
0433  0307              307
0434  0310              310
0435  0323              323
0436  0240              240
0437  0240              240
0440  0240              240
0441  0240              240
0442  0240              240
0443  0322              322          /RMS
0444  0315              315
0445  0323              323
0446  0240              240
0447  0240              240
0450  0240              240
0451  0240              240
0452  0240              240
0453  0315              315          /MAX
0454  0301              301
0455  0330              330
0456  0240              240
0457  0320              320          /PEAK
0460  0305              305
0461  0301              301
0462  0313              313
0463  0240              240
0464  0324              324          /TO
0465  0317              317
0466  0240              240
0467  0324              324
0470  0322              322          /TROUGH
0471  0317              317
0472  0325              325
0473  0307              307
0474  0310              310
0475  0323              323
0476  0240              240
0477  0240              240
```

TABLE C-I - (Continued)

```
0500  0240              240
0501  0240              240
0502  0315              315          /MEAN
0503  0305              305
0504  0301              301
0505  0316              316
0506  0215              215          /CRLF
0507  0212              212
0510  0240              240
0511  0240              240
0512  0240              240
0513  0316              316          /NO.
0514  0317              317
0515  0256              256
0516  0240              240
0517  0240              240
0520  0240              240
0521  0240              240
0522  0240              240
0523  0240              240
0524  0327              327          /W.I.
0525  0256              256
0526  0311              311
0527  0256              256
0530  0240              240
0531  0240              240
0532  0240              240
0533  0240              240
0534  0240              240
0535  0261              261          /1ST
0536  0323              323
0537  0324              324
0540  0240              240
0541  0315              315          /MODE
0542  0317              317
0543  0304              304
0544  0305              305
0545  0240              240
0546  0240              240
0547  0240              240
0550  0240              240
0551  0240              240
0552  0327              327          /W.I.
0553  0256              256
0554  0311              311
0555  0256              256
0556  0240              240
0557  0240              240
0560  0240              240
0561  0240              240
0562  0327              327          /W.I.
0563  0256              256
0564  0311              311
0565  0256              256
0566  0240              240
```

TABLE C-I - (Continued)

```
0567  0240          240
0570  0240          240
0571  0240          240
0572  0240          240
0573  0240          240
0574  0261          261     /1ST
0575  0323          323
0576  0324          324
0577  0240          240
0600  0315          315     /MODE
0601  0317          317
0602  0304          304
0603  0305          305
0604  0240          240
0605  0240          240
0606  0240          240
0607  0240          240
0610  0240          240
0611  0326          326     /VALUE
0612  0301          301
0613  0314          314
0614  0325          325
0615  0305          305
0616  0215          215     /CRLF
0617  0212          212
```

TABLE C-I - (Continued)

```
                    /
                    /CONVERTS BINARY NUMBERS TO DECIMAL
                    /EQUIVALENT FOR DISPLAY ON TELETYPE
                    /
                    FIELD 1
                    PAGE
1000  0000  DATA,   0000
1001  6211          CDF+10
1002  7300          CLA CLL
1003  1050          TAD F2
1004  4777          JMS BLANK
1005  7300          CLA CLL
1006  3057          DCA T1
1007  3060          DCA T2
1010  3061          DCA T3
1011  3062          DCA T4
1012  3063          DCA T5
1013  7240          STA
1014  3112          DCA FSWT
1015  1055          TAD INTNOF
1016  3115          DCA INTNF1
1017  1102          TAD F7775
1020  3103          DCA F7775A
1021  5776          JMP DAT1
1022  7300  DAT2,    CLA CLL
1023  6211          CDF+10
1024  1077  OUTPT2,  TAD LOCAF
1025  3012          DCA FX3
1026  1412          TAD I FX3
1027  4775          JMS DATA1
1030  4774          JMS PRINT
1031  1110          TAD F6
1032  4777          JMS BLANK
1033  1412          TAD I FX3
1034  4775          JMS DATA1
1035  4774          JMS PRINT
1036  1102          TAD F7775
1037  3103          DCA F7775A
1040  3064  DATA9,   DCA FOCTAL
1041  1107          TAD F5
1042  4777          JMS BLANK
1043  1412          TAD I FX3
1044  3052          DCA H2A
1045  1412          TAD I FX3
1046  3053          DCA H3
1047  7300  BMS,     CLA CLL
1050  7621          CAM
1051  1052          TAD H2A
1052  7450          SNA
1053  5277          JMP DATA2       /HIGH ORDER =0
1054  7413          SHL
1055  0001          0001
1056  3057          DCA T1
1057  1052          TAD H2A
1060  7421          MQL
```

TABLE C-I - (Continued)

```
1061  7405          MUY
1062  0140  F96,     0140    /96 DEC
1063  7300          CLA CLL
1064  7501          MQA
1065  1053          TAD H3
1066  3053          DCA H3
1067  7420          SNL
1070  5277          JMP DATA2
1071  1262          TAD F96 /OVERFLOW ADD 96*4K
1072  1053          TAD H3
1073  3053          DCA H3
1074  1105          TAD F4A
1075  1057          TAD T1
1076  3057          DCA T1
1077  7300  DATA2,   CLA CLL
1100  1053          TAD H3
1101  7500          SMA
1102  5312          JMP DATA3       /UNDER 2048 DEC
1103  1104          TAD F2000A
1104  2057          ISZ T1
1105  2057          ISZ T1
1106  7500          SMA
1107  5312          JMP DATA3
1110  1070          TAD F1000A
1111  2057          ISZ T1
1112  3053  DATA3,   DCA H3
1113  7300          CLA CLL
1114  1053          TAD H3
1115  4775          JMS DATA1
1116  7300          CLA CLL
1117  1057          TAD T1
1120  1066  DATA4,   TAD F10A
1121  7510          SPA
1122  5325          JMP DATA5
1123  2063          ISZ T5
1124  5320          JMP DATA4
1125  1072  DATA5,   TAD F10B
1126  3057          DCA T1
1127  7300          CLA CLL
1130  4774          JMS PRINT
1131  2103          ISZ F7775A
1132  5240          JMP DATA9
1133  2112          ISZ FSWT
1134  5336          JMP DATA6
1135  5773          JMP DATA7
1136  7300  DATA6,   CLA CLL
1137  4772          JMS CRLF1
1140  6201          CDF+00
1141  6202          CIF+00
1142  5600          JMP I DATA
1172  0222
1173  1230
1174  1446
1175  1400
1176  1200
```

TABLE C-I - (Continued)

```
1177  1433
                      PAGE
1200  7300  DAT1,     CLA CLL
1201  1515            TAD I INTNF1
1202  3226            DCA A1
1203  1226            TAD A1
1204  1227            TAD F7700
1205  7640            SZA CLA
1206  5211            JMP .+3
1207  1054            TAD F240
1210  5217            JMP .+7
1211  1226            TAD A1
1212  1116            TAD F0360
1213  7510            SPA
1214  1114            TAD F0100
1215  1113            TAD F0260
1216  6211            CDF+10
1217  4777            JMS TYPF
1220  2115            ISZ INTNF1
1221  2103            ISZ F7775A
1222  5200            JMP DAT1
1223  1107            TAD F5
1224  4776            JMS BLANK
1225  5775            JMP DAT2
1226  0000  A1,       0000
1227  7700  F7700,    7700
1230  7300  DATA7,    CLA CLL
1231  3064            DCA FOCTAL
1232  7240            STA
1233  3103            DCA F7775A
1234  1107            TAD F5
1235  4776            JMS BLANK
1236  1412            TAD I FX3
1237  3052            DCA H2A
1240  1412            TAD I FX3
1241  3053            DCA H3
1242  1052            TAD H2A
1243  3774            DCA HOLD1
1244  1053            TAD H3
1245  3773            DCA HOLD2
1246  7621            CAM
1247  7300            CLA CLL
1250  3311            DCA A2
1251  1052            TAD H2A
1252  7700            SMA CLA
1253  1312            TAD BLANK2
1254  1313            TAD MSGN1
1255  4777            JMS TYPF
1256  1052            TAD H2A
1257  7700            SMA CLA
1260  5301            JMP RMS1
1261  1053            TAD H3
1262  7450            SNA
1263  2311            ISZ A2
1264  7041            CIA
```

TABLE C-I - (Continued)

```
1265  3053            DCA H3
1266  7001            IAC
1267  1052            TAD H2A
1270  7440            SZA
1271  5275            JMP .+4
1272  1311            TAD A2
1273  3052            DCA H2A
1274  5301            JMP RMS1
1275  7041            CIA
1276  1311            TAD A2
1277  3052            DCA H2A
1300  5301            JMP RMS1
1301  7300  RMS1,     CLA CLL
1302  1052            TAD H2A
1303  3772            DCA PLY3
1304  1053            TAD H3
1305  3771            DCA PLY4
1306  4770            JMS VYMEAN
1307  7300            CLA CLL
1310  5767            JMP RMS
1311  0000  A2,       0000
1312  7763  BLANK2,    7763
1313  0255  MSGN1,     255
1367  1047
1370  2000
1371  2051
1372  2050
1373  2046
1374  2045
1375  1022
1376  1433
1377  0231
```

TABLE C-I - (Continued)

```
                      PAGE
                      /
                      /HANDLES POSITIVE NUMBERS
                      / ONLY
                      /
1400  0000  DATA1,    0000
1401  3064            DCA FOCTAL
1402  7300            CLA CLL
1403  1064            TAD FOCTAL
1404  1070  THOUS,    TAD F1000A
1405  7510            SPA
1406  5211            JMP HUND
1407  2057            ISZ T1
1410  5204            JMP THOUS
1411  1074  HUND,     TAD F1000B
1412  1067  HUND1,    TAD F100A
1413  7510            SPA
1414  5217            JMP TENS
1415  2060            ISZ T2
1416  5212            JMP HUND1
1417  1073  TENS,     TAD F100B
1420  1066  TENS1,    TAD F10A
1421  7510            SPA
1422  5225            JMP UNITS
1423  2061            ISZ T3
1424  5220            JMP TENS1
1425  1072  UNITS,    TAD F10B
1426  1062            TAD T4
1427  3062            DCA T4
1430  6211            CDF+10
1431  7300            CLA CLL
1432  5600            JMP I DATA1
1433  0000  BLANK,    0000
1434  7041            CIA
1435  7450            SNA
1436  5633            JMP I BLANK
1437  3051            DCA H1A
1440  1054  BLANK1,   TAD F240
1441  4777            JMS TYPF
1442  2051            ISZ H1A
1443  5240            JMP BLANK1
1444  7300            CLA CLL
1445  5633            JMP I BLANK
1446  0000  PRINT,    0000
1447  7300            CLA CLL
1450  3075            DCA B1
1451  1063            TAD T5
1452  7440            SZA
1453  5275            JMP PRINT2
1454  2075            ISZ B1
1455  1057            TAD T1
1456  7440            SZA
1457  5275            JMP PRINT2
1460  2075            ISZ B1
1461  1060            TAD T2
```

TABLE C-I - (Continued)

```
1462  7440          SZA
1463  5275          JMP PRINT2
1464  2075          ISZ B1
1465  1061          TAD T3
1466  7440          SZA
1467  5275          JMP PRINT2
1470  2075          ISZ B1
1471  1062          TAD T4
1472  7440          SZA
1473  5275          JMP PRINT2
1474  2075          ISZ B1
1475  7300  PRINT2, CLA CLL
1476  7621          CAM
1477  1075          TAD B1
1500  4233          JMS BLANK
1501  1776          TAD CHNGF
1502  3775          DCA TYPF1
1503  1075          TAD B1
1504  7413          SHL
1505  0000          0000
1506  1335          TAD PRINT4
1507  3310          DCA .+1
1510  5310          JMP .
1511  1063  PRINT3, TAD T5
1512  4777          JMS TYPF
1513  1057          TAD T1
1514  4777          JMS TYPF
1515  1060          TAD T2
1516  4777          JMS TYPF
1517  1061          TAD T3
1520  4777          JMS TYPF
1521  1062          TAD T4
1522  4777          JMS TYPF
1523  7300          CLA CLL
1524  1774          TAD NOP1
1525  3775          DCA TYPF1
1526  3057          DCA T1
1527  3060          DCA T2
1530  3061          DCA T3
1531  3062          DCA T4
1532  3063          DCA T5
1533  7000          NOP
1534  5646          JMP I PRINT
1535  5311  PRINT4, JMP PRINT3
1574  0243
1575  0232
1576  0244
1577  0231

                    *0050
0050  0002  F2,     0002
0051  0000  H1A,    0000
0052  0000  H2A,    0000
0053  0000  H3,     0000
0054  0240  F240,   240
0055  0040  INTNOF, INTNO
```

TABLE C-I - (Continued)

```
0056  0007  F7,     0007
0057  0000  T1,     0000
0060  0000  T2,     0000
0061  0000  T3,     0000
0062  0000  T4,     0000
0063  0000  T5,     0000
0064  0000  FOCTAL, 0000
0065  7777  F1A,    7777
0066  7766  F10A,   7766
0067  7634  F100A,  7634
0070  6030  F1000A, 6030
0071  0001  F1B,    0001
0072  0012  F10B,   0012
0073  0144  F100B,  0144
0074  1750  F1000B, 1750
0075  0000  B1,     0000
0076  6001  FPKBUF, 6001
            FX2=0011
            FX3=0012
0077  0017  LOCAF,0017       /BUFFER FOR PEAK TO TROUGHS
0100  7766  F7766,  7766
0101  0000  F7766A, 0000
0102  7775  F7775,  7775
0103  0000  F7775A, 0000
0104  4060  F2000A, 4060
0105  0004  F4A,    0004
0106  0000  H4,     0000
0107  0005  F5,     0005
0110  0006  F6,     0006
0111  0255  MSGN,   255
0112  0000  FSWT,   0000
0113  0260  F0260,  0260
0114  0100  F0100,  0100
0115  0000  INTNF1, 0000
0116  7420  F0360,  7420
                    SHL=7413
                    CAM=7621
                    MQL=7421
                    MUY=7405
                    MQA=7501
                    INTNO=0040
```

TABLE C-I - (Concluded)

```
                    *2000
2000  0000  VYMEAN, 0000
2001  7300          CLA CLL
2002  3247          DCA HOLD4
2003  1254          TAD VA
2004  7041          CIA
2005  1250          TAD PLY3
2006  7510          SPA
2007  5243          JMP ZOUT
2010  7650          SNA CLA
2011  5226          JMP Z1
2012  7300  Z2,     CLA CLL
2013  1250          TAD PLY3
2014  3254          DCA VA
2015  1251          TAD PLY4
2016  3255          DCA VB
2017  6201          CDF+00
2020  1245          TAD HOLD1
2021  3652          DCA I VY1
2022  1246          TAD HOLD2
2023  3653          DCA I VY2
2024  6211          CDF+10
2025  5243          JMP ZOUT
2026  7300  Z1,     CLA CLL
2027  1251          TAD PLY4
2030  7417          LSR
2031  0000          0000
2032  3247          DCA HOLD4
2033  1255          TAD VB
2034  7417          LSR
2035  0000          0000
2036  7041          CIA
2037  1247          TAD HOLD4
2040  7710          SPA CLA
2041  5243          JMP ZOUT
2042  5212          JMP Z2
2043  7300  ZOUT,   CLA CLL
2044  5600          JMP I VYMEAN
2045  0000  HOLD1,  0000
2046  0000  HOLD2,  0000
2047  0000  HOLD4,  0000
2050  0000  PLY3,   0000
2051  0000  PLY4,   0000
2052  3646  VY1,    3646
2053  3647  VY2,    3647
2054  0000  VA,     0000
2055  0000  VB,     0000
                    LSR=7417
```

APPENDIX D

SUMMARY TAPE AND EDIT PROGRAM

INTRODUCTION

This program was written to provide an editing capability for the digital data tapes and to provide a capability for creating a full compacted data tape from up to four partially filled data tapes. The editing option permits a) deletion of an interval, b) recalculation of voyage summary data, c) correction of interval logbook data, and d) the addition of a voyage identification record at the beginning of each voyage (as well as at the end of the voyage as produced in the data processing).

Two versions of the program were written. Version I operates when intervals are identified by an interval number. Version II operates when intervals must be identified by both an interval number and a logbook index number. The latter case arises because the interval numbering system used during part of the data acquisition permits an interval number to be used more than once during a voyage. In Version II, the Delete control cards and the Logbook Data control cards refer to an interval by both numbers. In all other respects, the programs are similar.

The program listings are given in Tables D-I (for Version I) and D-II (for Version II). The flow chart is shown in Figure D-1.

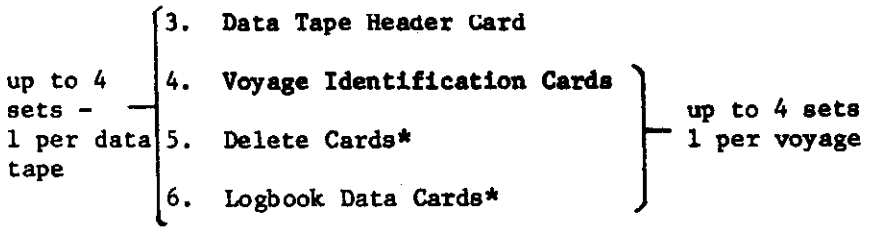
CARD INPUT

Types of card input

1. "C" coded control card
2. "H" coded header card for summary tape
3. "T" coded data tape header card
4. Voyage identification cards
5. "D" coded interval delete card
6. "L" coded logbook data cards

Order of card input

1. Control Card
2. Summary Tape Header Card



Control Card

Cols.	Description
1	"C"
5	Print Code — 0 - no print; 1 - print voyage summary record
10	Number of Data Tapes - up to 4

Summary Tape Header Card

Cols.	Description
1	"H"
2-10	Tape Identification
11-80	Description

Data Tape Header Card

Cols.		Description
1		"T"
2-27		Tape Identification
28-29		Number of Voyages
30-32	*	Voyage Number
33-35	*	Number of Intervals
36-37	*	Number of Delete Cards
37-38	*	Number of Sets (4/Set) of Logbook Data Cards
40-49		Same as *
50-59		Same as *
60-69		Same as *

Voyage Identification Cards

1. Two cards per voyage.

2. These are the same cards used to create paper tape (see Appendix A)--contains voyage logbook data.

3. Identified by a "V" in Column 78.

Delete Card (Version I)

Cols.	Description
1	"D"
2-4	Voyage Number
5-7	Interval Number
8-10	Interval Number
.	. .
.	. .
.	. .
31-33	Interval Number

10 intervals

Interval Logbook Data Cards, (Version I)

Cols.	Description
1	"L"
2	X = 1, 2, 3, or 4
3-5	Interval Number
6-77	Logbook Data 4th card only through Col. 45

Delete Card (Version II)

Cols.	Description
1	"D"
2-4	Voyage Number
5-7	Logbook Index Number
8-10	Interval Number
11-13	Logbook Index Number
14-16	Interval Number
.	. .
-64	Interval Number

10 intervals

Interval Logbook Data Cards (Version II)

Cols.	Description
1	"L"
2	X = 1, 2, 3, or 4
3-5	Logbook Index Number
6-8	Interval Number
9-80	Logbook Data 4th card only through Col. 48

TAPE INPUT/OUTPUT

The tape used as input are those tapes created using the program given in Appendix C (see Figure C-2 for tape format). The output tape is similar to the input with the exception that a Voyage Identification (logbook data only) has been inserted before the first interval in each voyage. This insertion does not include the Voyage Summary Data.

PRINTED OUTPUT (OPTION)

1. Listing of control cards

2. Listing of header record

3. INTERVAL XXX DELETED indicating one interval (XXX) has been deleted.

4. INTERVAL XXX CHANGED indicating that the logbook data for Interval XXX has been changed.

5. Listing of voyage summary record.

6. Error messages.

   a) CONTROL CARD MISSING
   "C" type control card is missing - correct and reenter

   b) SUMMARY TAPE HEADER CARD MISSING
   Self-explanatory--correct and reenter

   c) DATA TAPE HEADER CARD OUT OF SEQUENCE
   "T" type header card is missing or out of sequence--correct and reenter

   d) VOYAGE IDENTIFICATION CARD MISSING
   One of the 2 voyage identification cards for a voyage is missing or out of sequence--correct and reenter

   e) VOYAGE NO. ON DELETE CARD INCORRECT
   Voyage number on delete card does not match the appropriate voyage number on the data tape header card--correct and reenter

   f) INTERVAL LOGBOOK DATA CARD MISSING OR OUT OF ORDER
   According to count on data tape header card "L" type logbook data card is missing or out of sequence--
   correct and reenter

   g) TAPE IDENTIFICATION NOT THE SAME AS CARD IDENT
   Self-explanatory
   Correct and reenter

   h) VOYAGE SUMMARY RECORD FOR VOYAGE XXX MISSING
   Voyage number on voyage summary record does not match voyage number on data tape header card--correct card and reenter

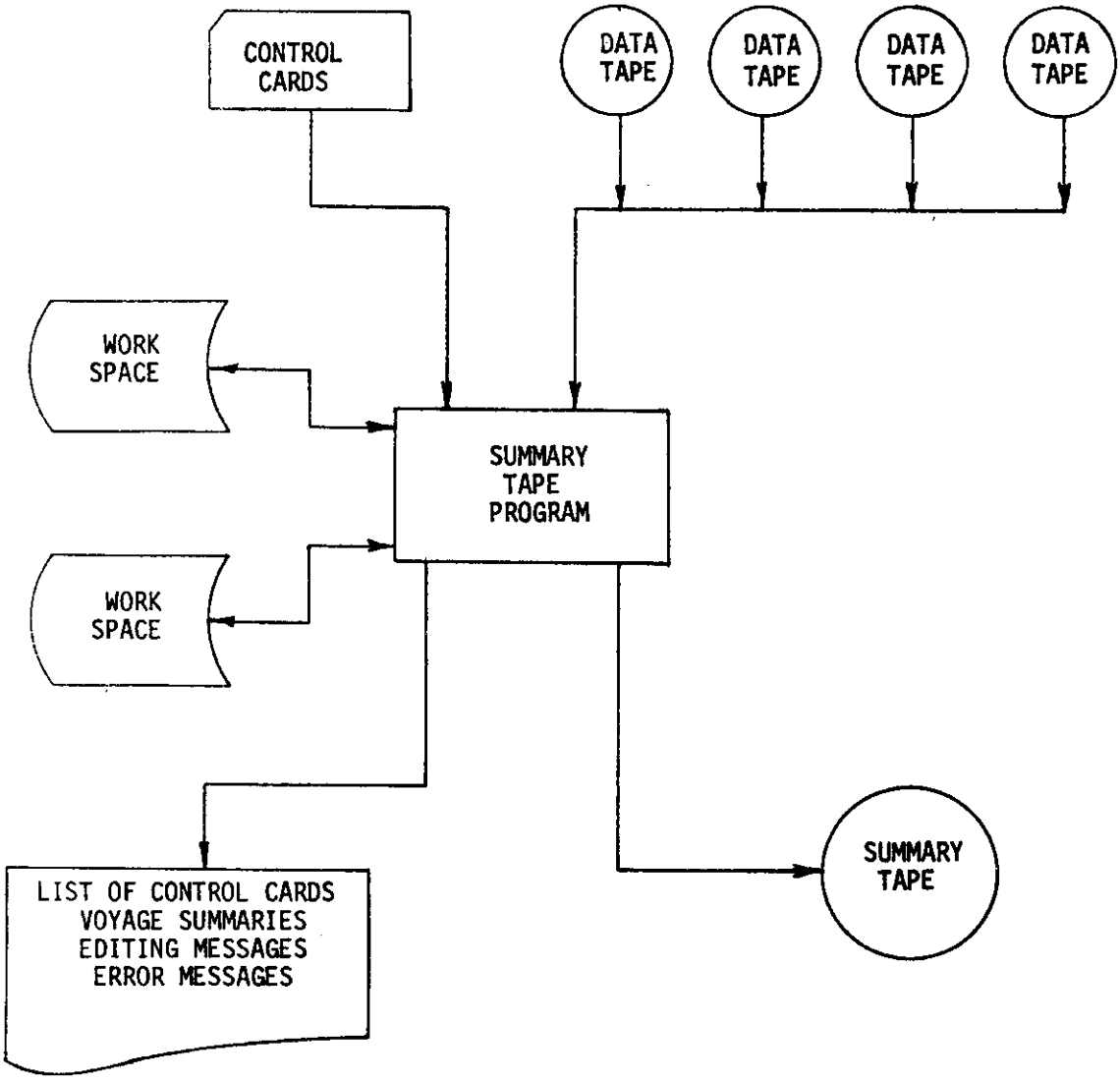


FIG. D-1 - FLOW CHART FOR SUMMARY TAPE AND EDIT PROGRAM

TABLE D-I - PROGRAM LISTING FOR SUMMARY TAPE AND EDIT PROGRAM (VERSION I)

```
//              MSGLEVEL=1
//STEP1 EXEC FORTGCLG,PARM.FORT='(NODECK,BCD,MAP),TIME.GO=(20,0)'
//FORT.SYSIN DD *
C                          VERSION I                                        SUMT0010
C                                                                           SUMT0020
C     THE SHIP SUMMARY TAPE PROGRAM CREATES A SUMMARY TAPE                  SUMT0030
C     OF VOYAGE AND INTERVAL INFORMATION FROM                               SUMT0040
C     A MAXIMUM OF 4 DATA TAPES CARD INPUT                                  SUMT0050
C     INDICATES THE TAPES TO BE READ , THE NO.                              SUMT0060
C     OF VOYAGES AND INTERVALS INVOLVED AND                                 SUMT0070
C     ANY INTERVALS TO BE DELETED OR                                        SUMT0080
C     CHANGED                                                               SUMT0090
C                                                                           SUMT0100
C                                                                           SUMT0110
      INTEGER*2 IDENT(40),IVOY1(36),IVOY2(34),IVOY3(3),                     SUMT0120
     1IFILL1(27),IFILL2(367),HEDR(250),ITDENT(13)                           SUMT0130
      DIMENSION IVOY(4),NINTV(4),NDEL(4),NLOG(4),IVOYN(16),INTVN(16),       SUMT0140
     1          INT(10),IYVOY1(9),INTY(9,10),IDATA(500),                    SUMT0150
     2IDATA(70),ILOG(18),ILOG1(18),ILOG2(18),ILOG3(10)                      SUMT0160
      DIMENSION MDATA(500),ITAPE(4)                                         SUMT0162
      DATA IC,IH,IT,IV,IB,IL /'C','H','T','V',' ','L'/,IFILL1,IFILL2,       SUMT0170
     1IVOYN,INTVN/27*' ',367*' ',16*' ',16*0/                               SUMT0180
      DATA MASK/Z00000FFF/,MASK1/ZFFFF0000/,MASK4/ZFFFFF00/
      DEFINE FILE 9 (200,50,U,IFILE)                                        SUMT0190
      DEFINE FILE 15(12,500,U,IDUM)                                         SUMT0194
      MASK3=0
      IFILE=1                                                               SUMT0200
      IEND=0                                                                SUMT0210
      ITNVOY=0                                                              SUMT0220
      IWRK=9                                                                SUMT0230
      IWRK2=15                                                              SUMT0235
      IOUT=6                                                                SUMT0240
      IN=5                                                                  SUMT0250
C     HEAD CONTROL CARD                                                     SUMT0260
      READ (IN,10) ICODE,IPCDE,NTAPS                                        SUMT0270
   10 FORMAT(A1,3X,I1,4X,I1)                                                SUMT0280
C     CHECK FOR CARD CODE 'C'                                               SUMT0290
      IF(ICODE-IC) 100,101,100                                              SUMT0300
  100 WRITE(IOUT,11)                                                        SUMT0310
   11 FORMAT(' CONTROL CARD MISSING')                                       SUMT0320
      STOP                                                                  SUMT0330
C     READ SUMMARY TAPE HEADER CARD                                         SUMT0340
  101 READ (IN,16) ICODE,IDENT                                              SUMT0350
      IF(ICODE-IH) 102,103,102                                              SUMT0360
  102 WRITE(IOUT,12)                                                        SUMT0370
   12 FORMAT(' SUMMARY TAPE HEADER CARD MISSING')                           SUMT0380
      STOP                                                                  SUMT0390
   16 FORMAT (A1,39A2,A1)                                                   SUMT0400
C     PRINT CONTROL CARD AND SUMMARY TAPE HEADER CARD                       SUMT0410
  103 WRITE(IOUT,15) IC,IPCDE,NTAPS,ICODE,IDENT                             SUMT0420
   15 FORMAT('1',A1,3X,I1,4X,I1,/,1X,A1,3X,40A2)                            SUMT0420
C     READ DATA TAPE CONTROL CARDS, CALCULATE FINAL HEADER                  SUMT0440
C     RECORD AND WRITE CARDS ONTO FILE                                      SUMT0450
      DO 106 JJ=1,NTAPS                                                     SUMT0460
C     READ DATA TAPE HEADER CARD                                            SUMT0470
      READ (IN,13) ICODE,ITDENT,NVOY, (IVOY(I),NINTV(I),
     1NDEL(I),NLOG(I),I=1,4)                                                SUMT0480
                                                                            SUMT0490
   13 FORMAT(A1,13A2,  I2,4(A3,I3,I2,I2))                                   SUMT0500
C     CHECK FOR CARD CODE 'T'                                               SUMT0510
      IF(ICODE-IT) 105,108,105                                              SUMT0520
```

TABLE D-I - (Continued)

```
   14 FORMAT(' DATA TAPE HEADER CARD OUT OF SEQUENCE')                      SUMT0530
  105 WRITE(IOUT,14)                                                        SUMT0540
      STOP                                                                  SUMT0550
  108 IK=1                                                                  SUMT0560
      ISTART=IEND+1                                                         SUMT0570
      IEND=ISTART+NVOY-1                                                    SUMT0580
      DO 104 JJ=ISTART,IEND                                                 SUMT0590
      IVOYN(JJ)=IVOY(IK)                                                    SUMT0600
      INTVN(JJ)=NINTV(IK)                                                   SUMT0610
  104 IK=IK+1                                                               SUMT0620
      ITNVOY=ITNVOY+NVOY                                                    SUMT0630
C     WRITE DATA TAPE HEADER CARD ON WORK FILE                              SUMT0640
      WRITE(IWRK'IFILE) ICODE,ITDENT,NVOY, (IVOY(I),NINTV(I),
     1NDEL(I),NLOG(I),I=1,4)                                                SUMT0660
C     PRINT DATA TAPE HEADER CARD                                           SUMT0670
      WRITE(IOUT,17) ICODE,ITDENT,NVOY, (IVOY(I),NINTV(I),
     1NDEL(I),NLOG(I),I=1,4)                                                SUMT0690
   17 FORMAT(//,1H ,A1,2X,13A2,2X,I2,2X,4(A3,2X,I3,2X,I2,2X,I2,2X)1        SUMT0700
      DO 123 JJ=1,NVOY                                                      SUMT0710
C     READ VOYAGE IDENTIFICATION CARDS                                      SUMT0720
      READ (IN,18)IVOY1,ICOD1,IVOY2,ICOD2                                   SUMT0730
   18 FORMAT(36A2,5X,A1/33A2,A1,10X,A1)                                     SUMT0740
C     CHECK FOR 'V' IN COLUMN 78                                            SUMT0750
      IF(ICOD1-IV) 109,110,109                                              SUMT0760
  110 IF(ICOD2-IV) 109,111,109                                              SUMT0770
  109 WRITE(IOUT,19)                                                        SUMT0780
   19 FORMAT(' VOYAGE IDENTIFICATION CARD MISSING')                         SUMT0790
      STOP                                                                  SUMT0800
C     WRITE VOYAGE CARDS ON DISK                                            SUMT0810
  111 WRITE(IWRK'IFILE) IVOY1,ICOD1                                         SUMT0820
      WRITE(IWRK'IFILE) IVOY2,ICOD2                                         SUMT0830
C        PRINT VOYAGE CARDS                                                 SUMT0832
      WRITE(IOUT,45) IVOY1,IVOY2                                            SUMT0834
   45 FORMAT(//1X,36A2/,1X,34A2//)                                          SUMT0836
C     READ DELETE CARDS                                                     SUMT0840
      IF(NDEL(JJ1)) 114,114,113                                             SUMT0850
  113 NDELL=NDEL(JJ1)                                                       SUMT0860
      DO 115 JJ2=1,NDELL                                                    SUMT0870
      READ(IN,20) ICODE,IYVOY,INT                                           SUMT0880
   20 FORMAT(A1,11A3)                                                       SUMT0890
      IF(IYVOY-IVOY(JJ1)) 116,117,116                                       SUMT0900
  116 WRITE(IOUT,21)                                                        SUMT0910
   21 FORMAT(' VOYAGE NO. ON DELETE CARD INCORRECT')                        SUMT0920
      STOP                                                                  SUMT0930
C     ADJUST COUNT FOR HEADER                                               SUMT0940
  117 KDEX=1                                                                SUMT0950
  122 IF(INT(KDEX)-10) 118,121,118                                          SUMT0960
  118 INTVN(ISTART+JJ1-1)=INTVN(ISTART+JJ1-1)-1                             SUMT0970
      IF(KDEX-10) 120,121,121                                               SUMT0980
  120 KDEX=KDEX+1                                                           SUMT0990
      GO TO 122                                                             SUMT1000
  121 WRITE(IWRK'IFILE) ICODE,IYVOY,INT                                     SUMT1010
C     PRINT DELETE CARD                                                     SUMT1020
      WRITE(IOUT,22) ICODE,IYVOY,INT                                        SUMT1030
   22 FORMAT(1X,A1,3X,A3,5X,10(A3,3X))                                      SUMT1040
  115 CONTINUE                                                              SUMT1050
C     READ INTERVAL LOG BOOK DATA CARDS                                     SUMT1060
  114 IF(NLOG(JJ1)) 123,123,124                                             SUMT1070
  124 NLOGG=NLOG(JJ1)                                                       SUMT1080
      DO 125 JJ2=1,NLOGG                                                    SUMT1090
```

TABLE D-I - (Continued)

```
      KZ=1                                                              SUNT1100
  130 READ (IN,23) ICODE,IX,IINT,ILOG                                   SUNT1110
   23 FORMAT(A1,I1,A3,18A4)                                             SUNT1120
      IF(ICODE-IL) 126,127,126                                          SUNT1130
  128 WRITE(IOUT,24)                                                    SUNT1140
   24 FORMAT(' INTERVAL LOG BOOK DATA CARD MISSING OR OUT OF ORDER')    SUNT1150
      STOP                                                              SUNT1160
  127 IF(IX-KZ) 126,128,126                                             SUNT1170
  128 WRITE(IWRK'IFILE) ICODE,IX,IINT,ILOG                              SUNT1180
      WRITE(IOUT,25) ICODE,IX,IINT,ILOG                                 SUNT1190
   25 FORMAT(' ',A1,3X,I1,3X,A3,5X,10A4)                                SUNT1200
      IF(KZ-4) 129,125,125                                              SUNT1210
  129 KZ=KZ+1                                                           SUNT1220
      GO TO 130                                                         SUNT1230
  125 CONTINUE                                                          SUNT1240
  123 CONTINUE                                                          SUNT1250
  106 CONTINUE                                                          SUNT1260
C     PRINT UPDATED HEADER INFORMATION                                  SUNT1270
      WRITE(IOUT,26) ITNVOY,(IVOYN(I),INTVN(I),I=1,16)                  SUNT1280
   26 FORMAT(' ',I6,3X,8(A3,2X,I4,3X)/10X,8(A3,2X,I4,3X))               SUNT1290
C     PART 2 CREATE SUMMARY TAPE FROM DATA TAPES                        SUNT1300
C                                                                       SUNT1310
      IEND=IFILE-1                                                      SUNT1320
      IFILE=1                                                           SUNT1330
      ITAPE(1)=10                                                       SUNT1340
      ITAPE(2)=11                                                       SUNT1350
      ITAPE(3)=12                                                       SUNT1360
      ITAPE(4)=13                                                       SUNT1370
      ISUMTP=14                                                         SUNT1380
C     WRITE HEADER RECORD ON SUMMARY TAPE                               SUNT1390
      WRITE(ISUMTP,40) IDENT,IFILL1,ITNVOY,(IVOYN(I),INTVN(I),I=1,16)   SUNT1400
     1,IFILL2                                                           SUNT1410
   40 FORMAT(40A2,27A2,A4,32A4,250A2,117A2)                             SUNT1415
      DO 135 JJ1=1,NTAPS                                                SUNT1420
      ITAP=ITAPE(JJ1)                                                   SUNT1430
C     READ HEADER RECORD FROM DATA TAPE                                 SUNT1440
      READ(ITAP,46) HEDR                                                SUNT1450
   46 FORMAT(256A2)                                                     SUNT1455
C     READ DATA TAPE HEADER CARD FROM WORK FILE                         SUNT1460
      READ(IWRK'IFILE) ICODE,ITDENT,NVOY, (IVOY(I),NINTV(I),NDEL(I),   SUNT1470
     1 NLOG(I),I=1,4)                                                   SUNT1480
C     CHECK TAPE IDENT AGAINST CARD IDENT                               SUNT1490
      DO 146 JZ=1,13                                                    SUNT1500
      IF(ITDENT(JZ)-HEDR(JZ)) 147,146,147                               SUNT1510
  147 WRITE(IOUT,29)                                                    SUNT1520
   29 FORMAT(' TAPE IDENTIFICATION NOT THE SAME AS CARD IDENT')         SUNT1530
      STOP                                                              SUNT1540
  146 CONTINUE                                                          SUNT1545
  148 K1=0                                                              SUNT1550
      K2=0                                                              SUNT1560
      DO 136 JJ2=1,NVOY                                                 SUNT1570
      READ (IWRK'IFILE) IVOY1,ICOD1                                     SUNT1580
      READ (IWRK'IFILE) IVOY2,ICOD2                                     SUNT1590
      NINTV= NLOG(JJ2)                                                  SUNT1600
      NDELM=NDEL(JJ2)                                                   SUNT1610
      NPKTT=0                                                           SUNT1615
      NPKTS=0                                                           SUNT1620
      IBM1=0                                                            SUNT1630
```

TABLE D-I - (Continued)

```
      MEANY=0                                                           SUNT1680
C     WRITE VOYAGE IDENTIFICATION ON SUMMARY TAPE                       SUNT1690
      WRITE(ISUMTP,41) IVOY1,(IVOY2(I),                                 SUNT1700
     1 I=1,26),(IFILL1(1),IFILL1(2),(IVOY2(I),I=27,34),                 SUNT1710
     2(IFILL2(I),I=1,116)                                               SUNT1720
   41 FORMAT(188A2)                                                     SUNT1725
      IF(NDELM) 137,137,138                                             SUNT1730
  138 DO 139 JJ3=1,NDELM                                                SUNT1740
  139 READ(IWRK'IFILE) ICODE,IYVOY1(JJ3),(INTY(JJ3,I),I=1,10)           SUNT1750
      IBE=1                                                             SUNT1760
      JBE=1                                                             SUNT1770
  137 IF(NINTV) 140,140,141                                             SUNT1780
  141 READ(IWRK'IFILE) ICODE,IX,IINT,ILOG                               SUNT1790
      READ(IWRK'IFILE) ICODE,IX1,IINT1,ILOG1                            SUNT1800
      READ(IWRK'IFILE) ICODE,IX2,IINT2,ILOG2                            SUNT1810
      READ(IWRK'IFILE)ICODE,IX3,IINT3,ILOG3                             SUNT1820
C                                                                       SUNT1830
C     BEGIN READING INTERVAL RECORD FROM DATA TAPE                      SUNT1840
C                                                                       SUNT1850
  148 NZAG=NINTV(JJ2)                                                   SUNT1860
      DO 149 JJ4=1,NZAG                                                 SUNT1870
      DO 150 JJ5=1,12                                                   SUNT1880
      READ(ITAP,42) IDATA                                               SUNT1890
  150 WRITE(IWRK2'JJ5) IDATA                                            SUNT1892
      READ(ITAP,43) IDATA,JDATA                                         SUNT1894
      ICOMP3>0                                                          SUNT1900
      ICOMP4>0                                                          SUNT1910
C                                                                       SUNT1920
C                                                                       SUNT1930
C     SUBPROGRAM BCD. NAME = AND= LANGUAGE=BAL                          SUNT1940
C     AVCO COMPUTER CENTER BOOLEAN SUBROUTINE                           SUNT1950
C      PURPOSE TO DO A LOGICAL AND ON TWO ARGUMENTS OF                  SUNT1960
C             LENGTH 2,4,OR 8 BYTES                                     SUNT1970
C        CALL AND(RESULT,ARG1,ARG2,NBYTES)                              SUNT1980
C     WHERE NBYTES =INTEGER*4 TAKES ON VALUES 2,4,OR 8 DEPENDING ON     SUNT1990
C          LENGTH OF ARG1,ARG2,RESULT.ALL MUST BE THE SAME LENGTH       SUNT2000
C                                                                       SUNT2010
      CALL AND(ICOMP1,MASK,IDATA(4),4)                                  SUNT2020
      CALL AND(ICOMP,MASK1,IDATA(5),4)                                  SUNT2030
C                                                                       SUNT2040
C                                                                       SUNT2050
C          LOGICAL SHIFT SUBROUTINE -AVCO COMPUTER CENTER               SUNT2060
C        LANGUAGE = BAL                                                 SUNT2070
C          PURPOSE TO SHIFT ANY PAIR OF FOUR-BYTE WORDS LEFT OR         SUNT2080
C   RIGHT N PLACES                                                      SUNT2090
C  USAGE CALL SHFT2V(WORD1,WORD2,WORD3,WORD4,N) WHERE WORD1 AND WORD2   SUNT2100
C     ARE ANY PAIR OF BYTE WORDS TO BE SHIFTED LEFT OR RIGHT            SUNT2110
C     WORD3 AND WORD4 ARE THE RESULTS OF THE SHIFT ON WORD1 AND WORD2   SUNT2120
C     TOGETHER.                                                         SUNT2130
C       N IS THE NUMBER OF BINARY PLACES SHIFTED EITHER TO THE LEFT     SUNT2140
C       AND SIGNED POSITIVE OR SHIFTED TO THE RIGHT AND SIGNED NEGATIVE.SUNT2150
C                                                                       SUNT2160
C                                                                       SUNT2170
      CALL SHFT2V(ICOMP1,ICOMP,ICOMP4,ICOMP3,-8)                        SUNT2180
      ICOMP=ICOMP3                                                      SUNT2190
      IF(NDELM.LE.0) GO TO 152                                          SUNT1960
      CALL AND(INTYB,MASK4,INTY(IBE,JBE),4)                             SUNT2200
      IF(INTYB.NE.ICOMP) GO TO 152                                      SUNT1980
      WRITE(IOUT,51) INTY(IBE,JBE)                                      SUNT2210
```

TABLE D-I - (Continued)

```
      JBE=JBE+1                                                          SUNT1990
      IF(JBE.GE.11)GO TO 172                                             SUNT2000
      IF(INTY(IBE,JBE).EQ.IB) GO TO 172                                  SUNT2010
      GO TO 149                                                          SUNT2030
  172 IBE =IBE+1                                                         SUNT2040
      JBE=1                                                              SUNT2045
      IF(IBE.LE.NDELM) GO TO 149                                         SUNT2050
      NDELM=0                                                            SUNT2060
      GO TO 149                                                          SUNT2070
C                                                                        SUNT2080
C     CHECK FOR PRESENCE OF LOG BOOK CARDS                               SUNT2090
  152 IF(NINTV.EQ.0) GO TO 153                                           SUNT2100
      CALL AND(INTYB,MASK4,IINT,4)                                       SUNT2120
      IF(ICOMP.NE.INTY(8) GO TO 153                                      SUNT2130
      WRITE(IOUT,52) IINT                                                SUNT2138
   52 FORMAT(' INTERVAL ',A3,' CHANGED')                                 SUNT2140
      DO 154 JT1=1,18                                                    SUNT2150
      IDATA(JT1)=ILOG(JT1)                                               SUNT2160
      IDATA(JT1+18)=ILOG1(JT1)                                           SUNT2170
  154 IDATA(JT1+36) = ILOG2(JT1)                                         SUNT2180
      DO 155 JT1=1,10                                                    SUNT2190
  155 IDATA(JT1+54)=ILOG3(JT1)                                           SUNT2200
      NINTV=NINTV-1                                                      SUNT2210
      IF(NINTV.EQ.0) GO TO 153                                           SUNT2230
      READ(IWRK'IF(LE) ICODE,IX,IINT,ILOG                                SUNT2240
      READ(IWRK'IF(LE) ICODE,IX1,IINT1,ILOG1                             SUNT2250
      READ(IWRK'IF(LE) ICODE,IX2,IINT2,ILOG2                             SUNT2260
      READ(IWRK'IF(LE) ICODE,IX3,IINT3,ILOG3                             SUNT2270
C                                                                        SUNT2280
C     WRITE INTERVAL RECORD ON SUMMARY TAPE                              SUNT2290
C                                                                        SUNT2300
  153 DO 175 KL=1,12                                                     SUNT2310
      READ(IWRK2'KL) MDATA                                               SUNT2320
  175 WRITE(ISUMTP,42) MDATA                                             SUNT2330
   42 FORMAT(250A4,256A4)                                                SUNT2340
      WRITE(ISUMTP,43) IDATA,JDATA                                       SUNT2350
   43 FORMAT(250A4,250A4,70A4)                                           SUNT2360
C     STORE INFORMATION FOR VOYAGE SUMMARY RECORD                        SUNT2370
C                                                                        SUNT2380
      NPKTT=NPKTT+IDATA(65)                                              SUNT2390
      NFMBS=NFMBS+IDATA(66)                                              SUNT2400
      IF(IRMS.GE.IDATA(67)) GO TO 158                                    SUNT2410
      IRMS=IDATA(67)                                                     SUNT2420
  158 IF(MPKTT.GE.IDATA(68)) GO TO 159                                   SUNT2430
      MPKTT=IDATA(68)                                                    SUNT2440
  159 IF(MPKTF.GE.IDATA(69)) GO TO 160                                   SUNT2450
      MPKTF=IDATA(69)                                                    SUNT2460
  160 IF(IABS(MEANV).GE.IABS(IDATA(70))) GO TO 149                       SUNT2470
      MEANV=IDATA(70)                                                    SUNT2480
  149 CONTINUE                                                           SUNT2490
C                                                                        SUNT2500
C     READ VOYAGE  SUMMARY RECORD                                        SUNT2510
C                                                                        SUNT2520
      READ(ITAP,44) (IDATA(I),I=1,100)                                   SUNT2530
      CALL AND(NCOMP,MASK4,IDATA(5),4)                                   SUNT2540
      CALL AND(NCOMP1,MASK4,IVOY(JJ2),4)                                 SUNT2550
      IF(NCOMP1.EQ.NCOMP) GO TO 161                                      SUNT2560
C     VOYAGE NO. ON SUMMARY RECORD DOES NOT AGREE WITH VOYAGE            SUNT2570
C     NUMBER IN TAPE HEADER CARD                                         SUNT2580
      WRITE(IOUT,30) IVOY(JJ2)                                           SUNT2590
```

TABLE D-I - (Concluded)

```
   30 FORMAT(' VOYAGE SUMMARY RECORD FOR VOYAGE ',A3,' MISSING')         SUNT2520
      STOP                                                               SUNT2530
  161 IDATA(95)=NPKTT                                                    SUNT2540
      IDATA(96)=NFMBS                                                    SUNT2550
      IDATA(97)=IRMS                                                     SUNT2560
      IDATA(98)=MPKTT                                                    SUNT2570
      IDATA(99)=MPKTF                                                    SUNT2580
      IDATA(100)=MEANV                                                   SUNT2590
C                                                                        SUNT2600
C     WRITE VOYAGE SUMMARY RECORD ON SUMMARY TAPE                        SUNT2610
C                                                                        SUNT2620
      WRITE(ISUMTP,44) (IDATA(I),I=1,100)                                SUNT2630
   44 FORMAT(100A4)                                                      SUNT2635
C                                                                        SUNT2640
C     CHECK FOR PRINT INDICATOR                                          SUNT2650
C                                                                        SUNT2660
      IF(IPCDE.EQ.0) GO TO 136                                           SUNT2670
      WRITE(IOUT,31) (IDATA(I),I=1,100)                                  SUNT2680
   31 FORMAT(3(1X,23A4/),1X,25A4/,1X,6(I10,2X))                          SUNT2690
  136 CONTINUE                                                           SUNT2700
  135 CONTINUE                                                           SUNT2710
C                                                                        SUNT2720
C     WHEN ALL DATA TAPES HAVE BEEN OPERATED ON AND                      SUNT2730
C     SUMMARY TAPE IS COMPLETE, WRITE EOF MARK ON                        SUNT2740
C     SUMMARY TAPE                                                       SUNT2750
      END FILE ISUMTP                                                    SUNT2760
      STOP                                                               SUNT2780
      END                                                                SUNT2790
/*
//GO.FT10F001 DD UNIT>TAPE9,LABEL>(,NL),VOLUME>SER>N059,
//             DISP>(OLD,KEEP),DCB>(RECFM>U,BLKSIZE>4000)
//GO.FT11F001 DD UNIT>TAPE9,LABEL>(,NL),VOLUME>SER>N060,
//             DISP>(OLD,KEEP),DCB>(RECFM>U,BLKSIZE>4000)
//GO.FT12F001 DD UNIT>TAPE9,LABEL>(,NL),VOLUME>SER>N061,
//             DISP>(OLD,KEEP),DCB>(RECFM>U,BLKSIZE>4000)
//GO.FT14F001 DD UNIT>TAPE9,DISP>(NEW,PASS),LABEL>(,NL),
//             VOLUME>SER>OUTPT,DCB>(RECFM>U,BLKSIZE>4000,DEN>2)
//GO.FT09F001 DD DISP>(NEW,PASS),UNIT>SYSDA,SPACE>(200,(200,10)),
//             DCB>(RECFM>F,BLKSIZE>200,DSORG>DA)
//GO.FT15F001 DD DISP>(NEW,PASS),UNIT>SYSDA,SPACE>(2000,(12,10)),
//             DCB>(RECFM>F,BLKSIZE>2000,DSORG>DA)
//GO.FT05F001 DD *
/*
```

TABLE D-II - PROGRAM LISTING FOR SUMMARY TAPE AND EDIT PROGRAM (VERSION II)

```
//YOBBCOMB JOB (S3;10006837,-----,101,---,IJWALT,C2444-------,7400],XX,X
//             MSGLEVEL=1
//STEP1 EXEC FORTGCLG,PARM.FORT=(NODECK,BCD,MAP),TIME.GO=(20,0)
//FORT.SYSIN DD *
C                         VERSION II                                 SUMT0010
C                                                                    SUMT0020
C      THE SHIP SUMMARY TAPE PROGRAM CREATES A SUMMARY TAPE          SUMT0030
C      OF VOYAGE AND INTERVAL INFORMATION FROM                       SUMT0040
C      A MAXIMUM OF 4 DATA TAPES CARD INPUT                          SUMT0050
C      INDICATES THE TAPES TO BE READ , THE NO.                      SUMT0060
C      OF VOYAGES AND INTERVALS INVOLVED AND                         SUMT0070
C      ANY INTERVALS TO BE DELETED OR                                SUMT0080
C      CHANGED                                                       SUMT0090
C                                                                    SUMT0100
C                                                                    SUMT0110
       INTEGER*2 IDENT(40),IVOY1(36),IVOY2(34),IVOY3(3),             SUMT0120
      1IFILL1(27),IFILL2(367),HEDR(250),ITDENT(13)                   SUMT0130
       DIMENSION IVOY(4),NINTV(4),NDEL(4),NLOG(4),IVOYN(16),INTVN(16), SUMT0140
      1              INT(10),IYVOY1(9),INTY(9,10),IDATA(500),        SUMT0150
      2JDATA(70),ILOG(18),ILOG1(18),ILOG2(18),ILOG3(10)              SUMT0160
       DIMENSION KDATA(500),ITAPE(4)                                 SUMT0162
       DIMENSION IDEX(10),IDEY(9,10)                                 SUMT0164
       DATA IC,IH,IT,IV,IB,IL /'C','H','T','V',' ','L'/,IFILL1,IFILL2, SUMT0170
      1IVOYN,INTVN/27*' ',367*' ',16*' ',16*0/                       SUMT0180
       DATA MASK/Z000000FF/,MASK1/ZFFFF0000/,MASK4/ZFFFFFF00/
       DATA MASK5/ZFFFFFF00/                                         SUMT0186
       DEFINE FILE 9 (200,50,U,IFILE)                                SUMT0190
       DEFINE FILE 15(12,500,U,IDUM)                                 SUMT0194
       MASK3=0
       IFILE=1                                                       SUMT0200
       IEND=0                                                        SUMT0210
       ITNVOY=0                                                      SUMT0220
       IWRK=9                                                        SUMT0230
       IWRK2=15                                                      SUMT0235
       IOUT=6                                                        SUMT0240
       IN=5                                                          SUMT0250
C      READ CONTROL CARD                                             SUMT0260
       READ (IN,10) ICODE,IPCDE,NTAPS                                SUMT0270
    10 FORMAT(A1,3X,I1,4X,I1)                                        SUMT0280
C      CHECK FOR CARD CODE 'C'                                       SUMT0290
       IF(ICODE-IC) 100,101,100                                      SUMT0300
   100 WRITE(IOUT,11)                                                SUMT0310
    11 FORMAT(' CONTROL CARD MISSING')                               SUMT0320
       STOP                                                          SUMT0330
C      READ SUMMARY TAPE HEADER CARD                                 SUMT0340
   101 READ (IN,16) ICODE,IDENT                                      SUMT0350
       IF(ICODE-IH) 102,103,102                                      SUMT0360
   102 WRITE(IOUT,12)                                                SUMT0370
    12 FORMAT(' SUMMARY TAPE HEADER CARD MISSING')                   SUMT0380
       STOP                                                          SUMT0390
    16 FORMAT (A1,39A2,A1)                                           SUMT0400
C      PRINT CONTROL CARD AND SUMMARY TAPE HEADER CARD               SUMT0410
   103 WRITE(IOUT,15) IC,IPCDE,NTAPS,ICODE,IDENT                     SUMT0420
    15 FORMAT('1',A1,3X,I1,4X,I1,/,1X,A1,3X,40A2)                    SUMT0420
C      READ DATA TAPE CONTROL CARDS, CALCULATE FINAL HEADER          SUMT0440
C      RECORD AND WRITE CARDS ONTO FILE                              SUMT0450
       DO 106 JJ=1,NTAPS                                             SUMT0460
C      READ DATA TAPE HEADER CARD                                    SUMT0470
       READ (IN,13) ICODE,ITDENT,NVOY, (IVOY(I),NINTV(I),            SUMT0480
      1NDEL(I),NLOG(I),I=1,4)                                        SUMT0490
```

TABLE D-II - (Continued)

```
    13 FORMAT(A1,13A2,  I2,4(A3,I3,I2,I2))                           SUMT0500
C      CHECK FOR CARD CODE 'T'                                       SUMT0510
       IF(ICODE-IT) 105,108,105                                      SUMT0520
    14 FORMAT(' DATA TAPE HEADER CARD OUT OF SEQUENCE')              SUMT0530
   105 WRITE(IOUT,14)                                                SUMT0540
       STOP                                                          SUMT0550
   108 IK=1                                                          SUMT0560
       ISTART=IEND+1                                                 SUMT0570
       IEND=ISTART+NVOY-1                                            SUMT0580
       DO 104 J1=ISTART,IEND                                         SUMT0590
       IVOYN(J1)=IVOY(IK)                                            SUMT0600
       INTVN(J1)=NINTV(IK)                                           SUMT0610
   104 IK=IK+1                                                       SUMT0620
       ITNVOY=ITNVOY+NVOY                                            SUMT0630
C      WRITE DATA TAPE HEADER CARD ON WORK FILE                      SUMT0640
       WRITE(IWRK'IFILE) ICODE,ITDENT,NVOY, (IVOY(I),NINTV(I),       SUMT0650
      1NDEL(I),NLOG(I),I=1,4)                                        SUMT0660
C      PRINT DATA TAPE HEADER CARD                                   SUMT0670
       WRITE(IOUT,17) ICODE,ITDENT,NVOY, (IVOY(I),NINTV(I),          SUMT0680
      1NDEL(I),NLOG(I),I=1,4)                                        SUMT0690
    17 FORMAT(//,1H ,A1,2X,13A2,2X,I2,2X,4(A3,2X,I3,2X,I2,2X,I2,2X)) SUMT0700
       DO 123 JJ1=1,NVOY                                             SUMT0710
C      READ VOYAGE IDENTIFICATION CARDS                              SUMT0720
       READ (IN,18) IVOY1,ICOD1,IVOY2,ICOD2                          SUMT0730
    18 FORMAT(36A2,5X,A1/33A2,A1,10X,A1)                             SUMT0740
C      CHECK FOR 'V' IN COLUMN 78                                    SUMT0750
       IF(ICOD1-IV) 109,110,109                                      SUMT0760
   110 IF(ICOD2-IV) 109,111,109                                      SUMT0770
   109 WRITE(IOUT,19)                                                SUMT0780
    19 FORMAT(' VOYAGE IDENTIFICATION CARD MISSING')                 SUMT0790
       STOP                                                          SUMT0800
C      WRITE VOYAGE CARDS ON DISK                                    SUMT0810
   111 WRITE(IWRK'IFILE) IVOY1,ICOD1                                 SUMT0820
       WRITE(IWRK'IFILE) IVOY2,ICOD2                                 SUMT0830
C         PRINT VOYAGE CARDS                                         SUMT0832
       WRITE(IOUT,45) IVOY1,IVOY2                                    SUMT0834
    45 FORMAT(//1X,36A2/,1X,34A2//)                                  SUMT0836
C      READ DELETE CARDS                                             SUMT0840
       IF(NDEL(JJ1)) 114,114,113                                     SUMT0850
   113 NDELL=NDEL(JJ1)                                               SUMT0860
       DO 115 JJ2=1,NDELL                                            SUMT0870
       READ(IN,20) ICODE,IYVOY,(IDEX(I),INT(I),I=1,10)               SUMT0880
    20 FORMAT(A1,21A3)                                               SUMT0890
       IF(IYVOY-IVOY(JJ1)) 116,117,116                               SUMT0900
   116 WRITE(IOUT,21)                                                SUMT0910
    21 FORMAT(' VOYAGE NO. ON DELETE CARD INCORRECT')                SUMT0920
       STOP                                                          SUMT0930
C      ADJUST COUNT FOR HEADER                                       SUMT0940
   117 KDEX=1                                                        SUMT0950
   122 IF(INT(KDEX)-IB) 118,121,118                                  SUMT0960
   118 INTVN(ISTART+JJ1-1)=INTVN(ISTART+JJ1-1)-1                     SUMT0970
       IF(KDEX-10) 120,121,121                                       SUMT0980
   120 KDEX=KDEX+1                                                   SUMT0990
       GO TO 122                                                     SUMT1000
   121 WRITE(IWRK'IFILE) ICODE,IYVOY,(IDEX(I),INT(I),I=1,10)         SUMT1010
C      PRINT DELETE CARD                                             SUMT1020
       WRITE(IOUT,22) ICODE,IYVOY,INT                                SUMT1030
    22 FORMAT(1X,A1,3X,A3,5X,10(A3,3X))                              SUMT1040
   115 CONTINUE                                                      SUMT1050
C      READ INTERVAL LOG BOOK DATA CARDS                             SUMT1060
```

TABLE D-II - (Continued)

```
  114 IF(NLOG(JJ1)) 123,123,124                                          SUMT1070
  124 NLOGG=NLOG(JJ1)                                                    SUMT1080
      DO 125 JJ2=1,NLOGG                                                 SUMT1090
      KZ=1                                                               SUMT1100
  130 READ(IN,23) ICODE,IX,NDEX,IINT,ILOG                                SUMT1110
   23 FORMAT(A1,I1,2A3,18A4)                                             SUMT1120
      IF(ICODE-IL) 126,127,126                                           SUMT1130
  126 WRITE(IOUT,24)                                                     SUMT1140
   24 FORMAT(' INTERVAL LOG BOOK DATA CARD MISSING OR OUT OF ORDER')     SUMT1150
      STOP                                                               SUMT1160
  127 IF(IX-KZ) 126,128,126                                              SUMT1170
  128 WRITE(IWRK*IFILE) ICODE,IX,NDEX,IINT,ILOG                          SUMT1180
      WRITE(IOUT,25) ICODE,IX,NDEX,IINT,ILOG                             SUMT1190
   25 FORMAT(' ',A1,3X,I1,3X,A3,3X,A3,5X,18A4)                           SUMT1200
      IF(KZ-4) 129,125,125                                               SUMT1210
  129 KZ=KZ+1                                                            SUMT1220
      GO TO 130                                                          SUMT1230
  125 CONTINUE                                                           SUMT1240
  123 CONTINUE                                                           SUMT1250
  106 CONTINUE                                                           SUMT1260
C     PRINT UPDATED HEADER INFORMATION                                   SUMT1270
      WRITE(IOUT,26) ITNVOY,(IVOYN(I),INTVN(I),I=1,16)                   SUMT1280
   26 FORMAT(' ',I6,3X,8(A3,2X,I4,3X)/10X,8(A3,2X,I4,3X))                SUMT1290
C     PART 2 CREATE SUMMARY TAPE FROM DATA TAPES                         SUMT1300
C                                                                        SUMT1310
      IEND=IFILE-1                                                       SUMT1320
      IFILE=1                                                            SUMT1330
      ITAPE(1)=10                                                        SUMT1340
      ITAPE(2)=11                                                        SUMT1350
      ITAPE(3)=12                                                        SUMT1360
      ITAPE(4)=13                                                        SUMT1370
      ISUMTP=14                                                          SUMT1380
C     WRITE HEADER RECORD ON SUMMARY TAPE                                SUMT1390
      WRITE(ISUMTP,40) IDENT,IFILL1,ITNVOY,(IVOYN(I),INTVN(I),I=1,16)    SUMT1400
     1,IFILL2                                                            SUMT1410
   40 FORMAT(40A2,27A2,A4,32A4,250A2,117A2)                              SUMT1415
      DO 135 JJ1=1,NTAPS                                                 SUMT1420
      ITAP=ITAPE(JJ1)                                                    SUMT1430
C     READ HEADER RECORD FROM DATA TAPE                                  SUMT1440
      READ(ITAP,46) HEDR                                                 SUMT1450
   46 FORMAT(250A2)                                                      SUMT1455
C     READ DATA TAPE HEADER CARD FROM WORK FILE                          SUMT1460
      READ(IWRK*IFILE) ICODE,ITDENT,NVOY, (IVOY(I),NINTV(I),NDEL(I),    SUMT1470
     1 NLOG(I),I=1,4)                                                    SUMT1480
C     CHECK TAPE IDENT AGAINST CARD IDENT                                SUMT1490
      DO 146 JZ=1,13                                                     SUMT1500
      IF(ITDENT(JZ)-HEDR(JZ)) 147,146,147                                SUMT1510
  147 WRITE(IOUT,29)                                                     SUMT1520
   29 FORMAT(' TAPE IDENTIFICATION NOT THE SAME AS CARD IDENT')          SUMT1530
      STOP                                                               SUMT1540
  146 CONTINUE                                                           SUMT1545
  148 K1=0                                                               SUMT1550
      K2=0                                                               SUMT1560
      DO 136 JJ2=1,NVOY                                                  SUMT1570
      READ (IWRK*IFILE) IVOY1,ICOD1                                      SUMT1580
      READ (IWRK*IFILE) IVOY2,ICOD2                                      SUMT1590
      NIMTV= NLOG(JJ2)                                                   SUMT1600
      NDELM=NDEL(JJ2)                                                    SUMT1610
      NPKTT=0                                                            SUMT1615
      NFMBS=0                                                            SUMT1620
```

TABLE D-II - (Continued)

```
      IRMS=0                                                             SUMT1630
      NPKTT=0                                                            SUMT1640
      NPKTF=0                                                            SUMT1670
      MEANV=0                                                            SUMT1680
C     WRITE VOYAGE IDENTIFICATION ON SUMMARY TAPE                        SUMT1690
      WRITE(ISUMTP,41) IVOY1,(IVOY2(I),                                  SUMT1700
     1 I=1,26),IFILL1(1),IFILL1(2),(IVOY2(I),I=27,36),                   SUMT1710
     2(IFILL2(I),I=1,116)                                                SUMT1720
   41 FORMAT(188A2)                                                      SUMT1725
      IF(NDELM) 137,137,138                                              SUMT1730
  138 DO 139 JJ3=1,NDELM                                                 SUMT1740
  139 READ(IWRK*IFILE) ICODE,IYVOY1(JJ3),(IDEY(JJ3,I),INTY(JJ3,I),I=1,10SUMT1750
     1)                                                                  SUMT1755
      IBE=1                                                              SUMT1760
      JBE=1                                                              SUMT1770
  137 IF(NIMTV) 140,140,141                                              SUMT1780
  141 READ(IWRK*IFILE) ICODE,IX,NDEX,IINT,ILOG                           SUMT1790
      READ(IWRK*IFILE) ICODE,IX1,LDEX,IINT1,ILOG1                        SUMT1800
      READ(IWRK*IFILE) ICODE,IX2,LDEX,IINT2,ILOG2                        SUMT1810
      READ(IWRK*IFILE) ICODE,IX3,LDEX,IINT3,ILOG3                        SUMT1820
C                                                                        SUMT1830
C     BEGIN READING INTERVAL RECORD FROM DATA TAPE                       SUMT1840
C                                                                        SUMT1850
  140 NZAG=NINTV(JJ2)                                                    SUMT1860
      DO 149 JJ4=1,NZAG                                                  SUMT1870
      DO 150 JJ5=1,12                                                    SUMT1880
      READ(ITAP,42) IDATA                                                SUMT1890
  150 WRITE(IWRK2*JJ5) IDATA                                             SUMT1892
      READ(ITAP,43) IDATA,JDATA                                          SUMT1894
      ICOMP3=0                                                           
      ICOMP4=0                                                           
C                                                                        
C                                                                        
C     SUBPROGRAM BCD. NAME - AND- LANGUAGE-BAL                           
C     AVCO COMPUTER CENTER BOOLEAN SUBROUTINE                            
C      PURPOSE TO DO A LOGICAL AND ON TWO ARGUMENTS OF                   
C           LENGTH 2,4,OR 8 BYTES                                        
C        CALL AND(RESULT,ARG1,ARG2,NBYTES)                               
C     WHERE NBYTES -INTEGER*4 TAKES ON VALUES 2,4,OR 8 DEPENDING ON      
C          LENGTH OF ARG1,ARG2,RESULT.ALL MUST BE THE SAME LENGTH        
C                                                                        
      CALL AND(ICOMP1,MASK,IDATA(4),4)                                   
      CALL AND(ICOMP,MASK1,IDATA(5),4)                                   
      CALL AND(ICOMP4,MASK5,IDATA(4),4)                                  
C                                                                        
C                                                                        
C          LOGICAL SHIFT SUBROUTINE -AVCO COMPUTER CENTER                
C         LANGUAGE - BAL                                                 
C          PURPOSE TO SHIFT ANY PAIR OF FOUR-BYTE WORDS LEFT OR          
C    RIGHT N PLACES                                                      
C   USAGE CALL SHFT2V(WORD1,WORD2,WORD3,WORD4,N) WHERE WORD1 AND WORD2   
C     ARE ANY PAIR OF BYTE WORDS TO BE SHIFTED LEFT OR RIGHT             
C     WORD3 AND WORD4 ARE THE RESULTS OF THE SHIFT ON WORD1 AND WORD2    
C     TOGETHER.                                                          
C       N IS THE NUMBER OF BINARY PLACES SHIFTED EITHER TO THE LEFT      
C      AND SIGNED POSITIVE OR SHIFTED TO THE RIGHT AND SIGNED NEGATIVE.  
C                                                                        
C                                                                        
      CALL SHFT2V(ICOMP1,ICOMP,ICOMP4,ICOMP3,-8)                         
      ICOMP=ICOMP3                                                       
```

TABLE D-II - (Continued)

```
      IF(NDELM.LE.0) GO TO 152                                          SUNT1960
      CALL AND(INTYB,MASK4,INTY(IBE,JBE),4)
      CALL AND(LNTYB,MASK5,IDEX(IBE,JBE),4)
      IF(LNTYB.NE.LCOMP) GO TO 152                                      SUNT1975
      IF(INTYB.NE.ICOMP) GO TO 152                                      SUNT1980
      WRITE(IOUT,51) INTY(IBE,JBE)
   51 FORMAT(' INTERVAL ',A3,' DELETED')
      JBE=JBE+1
      IF(JBE.GE.11)GO TO 172                                            SUNT1990
      IF(INTY(IBE,JBE).EQ.IB) GO TO 172                                 SUNT2000
      GO TO 149                                                         SUNT2010
  172 IBE =IBE+1                                                        SUNT2030
      JBE=1                                                             SUNT2040
      IF(IBE.LE.NDELM) GO TO 149                                        SUNT2045
      NDELM=0                                                           SUNT2050
      GO TO 149                                                         SUNT2060
C                                                                       SUNT2070
C     CHECK FOR PRESENCE OF LOG BOOK CARDS                              SUNT2080
  152 IF(NIMTV.EQ.0) GO TO 153                                          SUNT2090
      CALL AND(INTYB,MASK4,IINT,4)                                      SUNT2100
      CALL AND(LNTYB,MASK5,NDEX,4)                                      SUNT2105
      IF(LCOMP.NE.LNTYB) GO TO 153                                      SUNT2115
      IF(ICOMP.NE.INTYB) GO TO 153                                      SUNT2120
      WRITE(IOUT,52) IINT
   52 FORMAT(' INTERVAL ',A3,' CHANGED')
      DO 154 JT1=1,18                                                   SUNT2130
      IDATA(JT1)=ILOG(JT1)                                              SUNT2140
      IDATA(JT1+18)=ILOG1(JT1)                                          SUNT2150
  154 IDATA(JT1+36) = ILOG2(JT1)                                        SUNT2160
      DO 155 JT1=1,18                                                   SUNT2170
  155 IDATA(JT1+54)=ILOG3(JT1)                                          SUNT2180
      NIMTV=NIMTV-1                                                     SUNT2190
      IF(NIMTV.EQ.0) GO TO 153                                          SUNT2200
      READ(IWRK*IFILE) ICODE,IX,NDEX,IINT,ILOG                          SUNT2210
      READ(IWRK*IFILE) ICODE,IX1,LDEX,IINT1,ILOG1                       SUNT2220
      READ(IWRK*IFILE) ICODE,IX2,LDEX,IINT2,ILOG2                       SUNT2230
      READ(IWRK*IFILE) ICODE,IX3,LDEX,IINT3,ILOG3                       SUNT2240
C                                                                       SUNT2250
C     WRITE INTERVAL RECORD ON SUMMARY TAPE                             SUNT2260
C                                                                       SUNT2270
  153 DO 175 KL=1,12                                                    SUNT2260
      READ(IWRK2*KL) MDATA                                              SUNT2262
  175 WRITE(ISUMTP,42) MDATA                                            SUNT2264
   42 FORMAT(250A4,250A4)                                               SUNT2264
      WRITE(ISUMTP,43) IDATA,JDATA                                      SUNT2280
   43 FORMAT(250A4,250A4,70A4)                                          SUNT2285
C     STORE INFORMATION FOR VOYAGE SUMMARY RECORD                       SUNT2290
C                                                                       SUNT2300
      NPKTT=NPKTT+IDATA(65)                                             SUNT2310
      NFWBS=NFWBS+IDATA(66)                                             SUNT2320
      IF(IRMS.GE.IDATA(67)) GO TO 158                                   SUNT2330
      IRMS=IDATA(67)                                                    SUNT2340
  158 IF(MPKTT.GE.IDATA(68)) GO TO 159                                  SUNT2350
      MPKTT=IDATA(68)                                                   SUNT2360
  159 IF(MPKTF.GE.IDATA(69)) GO TO 160                                  SUNT2370
      MPKTF=IDATA(69)                                                   SUNT2380
  160 IF(IABS(MEANV).GE.IABS(IDATA(70))) GO TO 149                      SUNT2390
      MEANV=IDATA(70)                                                   SUNT2400
  149 CONTINUE                                                          SUNT2410
C                                                                       SUNT2420
```

TABLE D-II - (Continued)

```
C     READ VOYAGE  SUMMARY RECORD                                       SUNT2430
C                                                                       SUNT2440
      READ(ITAP,44) (IDATA(I),I=1,100)                                  SUNT2450
      CALL AND(NCOMP,MASK4,IDATA(5),4)                                  SUNT2460
      CALL AND(NCOMP1,MASK4,IVOY(JJ2),4)                                SUNT2470
      IF(NCOMP1.EQ.NCOMP) GO TO 161                                     SUNT2480
C     VOYAGE NO. ON SUMMARY RECORD DOES NOT AGREE WITH VOYAGE           SUNT2490
C     NUMBER IN TAPE HEADER CARD                                        SUNT2500
      WRITE(IOUT,39) IVOY(JJ2)                                          SUNT2510
   39 FORMAT(' VOYAGE SUMMARY RECORD FOR VOYAGE ',A3,' MISSING')        SUNT2520
      STOP                                                              SUNT2530
  161 IDATA(95)=NPKTT                                                   SUNT2540
      IDATA(96)=NFWBS                                                   SUNT2550
      IDATA(97)=IRMS                                                    SUNT2560
      IDATA(98)=MPKTT                                                   SUNT2570
      IDATA(99)=MPKTF                                                   SUNT2580
      IDATA(100)=MEANV                                                  SUNT2590
C                                                                       SUNT2600
C     WRITE VOYAGE SUMMARY RECORD ON SUMMARY TAPE                       SUNT2610
C                                                                       SUNT2620
      WRITE(ISUMTP,44) (IDATA(I),I=1,100)                               SUNT2630
   44 FORMAT(100A4)                                                     SUNT2635
C                                                                       SUNT2640
C     CHECK FOR PRINT INDICATOR                                         SUNT2650
C                                                                       SUNT2660
      IF(IPCDE.EQ.0) GO TO 136                                          SUNT2670
      WRITE(IOUT,31) (IDATA(I),I=1,100)                                 SUNT2680
   31 FORMAT(3(1X,23A4/),1X,25A4/,1X,6(I10,2X))                         SUNT2690
  136 CONTINUE                                                          SUNT2700
  135 CONTINUE                                                          SUNT2710
C                                                                       SUNT2720
C     WHEN ALL DATA TAPES HAVE BEEN OPERATED ON AND                     SUNT2730
C     SUMMARY TAPE IS COMPLETE, WRITE EOF MARK ON                       SUNT2740
C     SUMMARY TAPE                                                      SUNT2750
      END FILE ISUMTP                                                   SUNT2760
      STOP                                                              SUNT2780
      END                                                               SUNT2790
/*
//GO.F110F001 DD UNIT=TAPE9,LABEL=(,NL),VOLUME=SER=N052,
//            DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=4000)
//GO.F111F001 DD UNIT=TAPE9,LABEL=(,NL),VOLUME=SER=N053,
//            DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=4000)
//GO.F112F001 DD UNIT=TAPE9,LABEL=(,NL),VOLUME=SER=N054,
//            DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=4000)
//GO.F114F001 DD UNIT=TAPE9,DISP=(NEW,PASS),LABEL=(,NL),
//            VOLUME=SER=OUTPT,DCB=(RECFM=U,BLKSIZE=4000,DEN=2)
//GO.F109F001 DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(200,(200,100)),
//            DCB=(RECFM=F,BLKSIZE=200,DSORG=DA)
//GO.F115F001 DD DISP=(NEW,PASS),UNIT=SYSDA,SPACE=(2000,(12,10)),
//            DCB=(RECFM=F,BLKSIZE=2000,DSORG=DA)
//GO.F105F001 DD *
/*
```

TABLE D-II - (Continued)

```
SHFTZV START 0                                                          00026400
       STM   14,12,12(13)                                               00026410
       BALR  10,0                                                       00026420
       USING *,10                                                       00026430
       L     R4,16(R1)       POSITIONS TO BE SHIFTED                    00026440
       L     R4,0(R4)        **                                         00026450
       L     R3,4(R1)        GET 2ND ARG                                00026460
       L     R3,0(R3)        **                                         00026470
       L     R2,0(R1)        GET 1ST ARG                                00026480
       L     R2,0(R2)        **                                         00026490
       LTR   R5,R4           TEST SIGN                                  00026500
       BC    4,RIGHT         VALUE IS NEG. SHIFT RIGHT                  00026510
       STC   R4,SLA+3                                                   00026520
SLA    SLDL  R2,0            SHIFT LEFT DOUBLE (OVERLAY)                00026530
       BC    15,OUT                                                     00026540
RIGHT  LPR   R4,R4           SET SIGN POSITIVE                          00026550
       STC   R4,SRA+3                                                   00026560
SRA    SRDL  R2,0            SHIFT RIGHT DOUBLE                         00026570
OUT    L     R4,8(R1)        3RD ARG                                    00026580
       L     R5,12(R1)       4TH ARG                                    00026590
       ST    R2,0(R4)        1ST ARG IN 3RD                             00026600
       ST    R3,0(R5)        2ND ARG IN 4TH                             00026610
       LM    14,12,12(13)    RESTORE REGS                               00026630
       BALR  R15,R14         RETURN                                     00026640
R1     EQU   1                                                          00026650
R2     EQU   2                                                          00026660
R3     EQU   3                                                          00026670
R4     EQU   4                                                          00026680
R5     EQU   5                                                          00026690
R14    EQU   14                                                         00026700
R15    EQU   15                                                         00026710
       END                                                              00026730
```

```
SBLOGAND CSECT
*
*      SUBROUTINE TO TAKE THE LOGICAL AND OF TWO ARGUMENTS.
*      CALLING SEQUENCE    CALL ANDIRESULT,ARG1,ARG2,NBYTES)
*
       ENTRY AND
       USING AND,15
AND    SAVE  (14,12),,*
       L     R2,12(0,R1)
       L     R3,0(0,R2)
       S     R3,=F'4'
       LTR   R3,R3
       BP    BYTE8
```

TABLE D-II - (Concluded)

```
       BM    BYTE2
BYTE4  L     R2,4(0,R1)
       L     R3,0(0,R2)
       L     R2,4(0,R1)
       L     R5,0(0,R2)
       NR    R3,R5
       L     R2,0(0,R1)
       ST    R3,0(0,R2)
       B     RETURN
BYTE8  L     R2,4(0,R1)
       MVC   DOUBLE1,0(R2)
       L     R2,4(0,R1)
       MVC   DOUBLE2,0(R2)
       NC    DOUBLE1,DOUBLE2
       L     R2,0(0,R1)
       MVC   0(8,R2),DOUBLE1
       B     RETURN
BYTE2  L     R2,4(0,R1)
       MVC   HALF1,0(R2)
       L     R2,8(0,R1)
       MVC   HALF2,0(R2)
       NC    HALF1,HALF2
       L     R2,0(0,R1)
       MVC   0(2,R2),HALF1
RETURN RETURN (14,12),RC=0
DOUBLE1 DS   D
DOUBLE2 DS   D
HALF1  DS    H
HALF2  DS    H
R1     EQU   1
R2     EQU   2
R3     EQU   3
R4     EQU   4
R5     EQU   5
       END
*ANDV  ***  SUBROUTINE TO AND TWO FOUR-BYTE ARGUMENTS  ***
ANDVX  CSECT
       ENTRY ANDV
ANDV   STM   14,12,12(13)
       BALR  10,0
       USING *,10
       L     R2,0(R1)
       L     R2,0(R2)
       L     R3,4(R1)
       N     R2,0(R3)
       L     R3,4(R1)
       ST    R2,0(R3)
       LM    14,12,12(13)
       BALR  R15,R14
R1     EQU   1
R2     EQU   2
R3     EQU   3
R14    EQU   14
R15    EQU   15
*      ***  END SUBROUTINE ANDV  ***
       END
******END OF ROUTINE**********************************************
```

APPENDIX E

FINAL SUMMARY TAPE PROGRAM

INTRODUCTION

The edited and compacted data tapes (Appendix D) contain the digital records (12,000 points) necessary to reproduce the recorded signal for each interval. For use in studies such as those required under the parametric study phase of the contract, considerable efficiency in tape reading is accomplished by having a tape which contains only the information required for such studies. (Interval and Voyage Identification and Summaries). Elimination of the digital records permitted all the Interval and Voyage Identification and Summary data from the 25 edited and compacted data tapes to be written on two Final Summary Tapes (one each containing full-bridge data and half-bridge data). The Final Summary Tape program (FMST) was written to accomplish this. A maximum of 150 voyages is provided for in the header record.

The program listing is given in Table E-I and the flow chart is given in Figure E-1.

CARD INPUT

1. Control Card
2. Final Summary Tape Header Card
3. Header Cards for Input Data Tapes
   (one for each data tape)

Control Card

Col.	Description
1	"C"
5	Type Code — 1 - creation; 2 - addition
10	Number of Tapes - max. 4

Final Summary Tape Header Card

Col.	Description
1	"F"
2-80	Header information (description)

Header Card for Input Data Tapes*

Col.	Description
1	"H"
2-80	Header identification and description

*Also input to summary tape program

PRINTED OUTPUT

1. Control Card
2. Old Summary Tape Header Record of Addition Run
3. Final Summary Tape Header Card
4. Header Card for Each Data Tape
5. New Summary Tape Header Record
6. Old Voyage Summary
   for each voyage on the Old Summary Tape if addition run
7. New Voyage Summary
   for each voyage on the data tapes in the order of processing
8. Error Messages

Error Messages

1. CONTROL CARD MISSING
   "C" type control card missing or out of sequence--correct and reenter
2. CONTROL CARD COL 5 MUST BE 1 OR 2
   applies to "C" type control-card correct and reenter
3. HEADER CARD MISSING FOR FINAL SUMMARY TAPE
   "F" type header card is missing or out of sequence--correct and reenter
4. TAPE HEADER CARD MISSING

   "H" type header card is missing or out of sequence--correct and reenter
5. TAPE HEADER DOES NOT MATCH HEADER CARD XX_____X
   The first 79 bytes of the summary tape header does not match the data on the respective header card--correct and reenter XX _____X in the tape header

TAPE INPUT/OUTPUT

The tape input is that which results from the edited and compacted data tapes (see Appendix D).

The tape output results from deletion of 12 records (2000 bytes each) of Interval Data and the resultant compacting of the remaining data. The tape output is:

1. Tape Header (1400 bytes)
2. Inter-record Gap (IRG)
3. Voyage Identification (376 bytes)
4. IRG
5. Interval Summary (2280 bytes)
6. IRG
7. --Interval Summaries for remaining intervals
8. IRG
9. Voyage Summary (400 bytes)
   --Repeat 2-9 for each voyage.

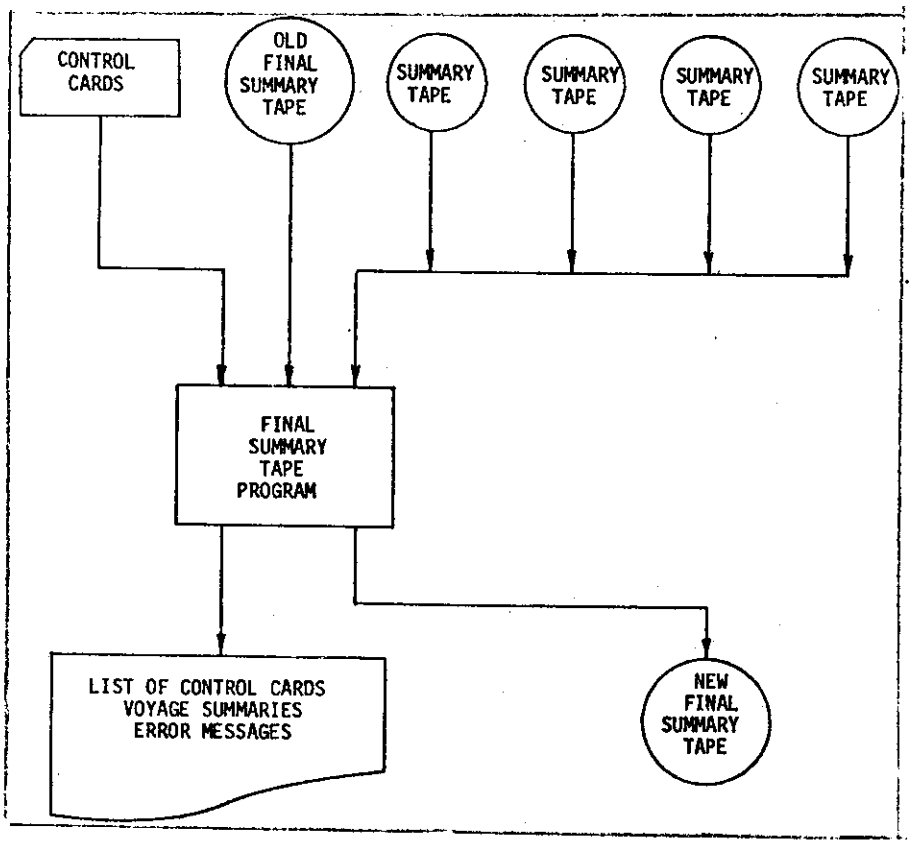


FIG. E-1 - FLOW CHART FOR FINAL SUMMARY TAPE PROGRAM

TABLE E-I - PROGRAM LISTING FOR FINAL SUMMARY TAPE PROGRAM

```
// JOB
//C86COMP1 JOB (S3110005842,-----,101,---,PETRVC,C1801-------,7400),XX,X
//              TYPRUN=HOLD,                                          X
//              MSGLEVEL=1
//STEP1 EXEC FORTGCLG,PARM,FORT='NODECK,MAP',TIME.GO=(20,0)
//FORT.SYSIN DD *
C                                                                     FSMT0010
C                                                                     FSMT0020
C       THE FINAL SUMMARY TAPE PROGRAM CREATES A FINAL SUMMARY        FSMT0030
C       TAPE OF VOYAGE AND INTERVAL INFORMATION FROM A MAXIMUM        FSMT0040
C       OF 4 DATA TAPES OF PACKED , EDITED DATA AND FROM              FSMT0050
C       ANOTHER SUMMARY TAPE (OUTPUT FROM THIS PROGRAM).              FSMT0060
C       THE DATA TAPE INPUT IS OUTPUT FROM THE SHIP                   FSMT0070
C       PACK AND EDIT PROGRAM CARD INPUT. CARD INPUT INDICATES HOW    FSMT0080
C       MANY AND WHICH TAPES ARE TO BE READ AND WHETHER               FSMT0090
C       IT IS TO BE A CREATION OR AN ADDITION RUN. THE                FSMT0100
C       DIGITIZED INTERVAL DATA IS OMITTED ON THE                     FSMT0110
C       FINAL SUMMARY TAPE.                                           FSMT0120
C                                                                     FSMT0130
C                                                                     FSMT0140
      DIMENSION IDATA2(201),JDATA(4,129),KDATA(570)                   FSMT0150
      DIMENSION JTAPE(4)
      INTEGER*2 IDENT(40),JDENT(40),IDATA1(67),IDATA3(31)             FSMT0160
      INTEGER*2 JFILL(27)
      INTEGER*4 PCODE
      INTEGER*4 OSUMTP
      DATA IC,I1,I2/'C','1','2'/,IDATA2/201*0/,IDATA3/31*0/,IFF,IH/'F',FSMT0170
     1 4H    /                                                        FSMT0180
      DATA IDATA1/67*' '/
      ICOUNT=2                                                        FSMT0190
      OSUMTP=9                                                        FSMT0200
      NSUMTP=14                                                       FSMT0210
      JTAPE(1)=10                                                     FSMT0220
      JTAPE(2)=11                                                     FSMT0230
      JTAPE(3)=12                                                     FSMT0240
      JTAPE(4)=13                                                     FSMT0250
      ICOUNT=2                                                        FSMT0260
      IN=5                                                            FSMT0270
      IOUT=6                                                          FSMT0280
C       READ CONTROL CARD                                             FSMT0290
      READ(IN,10) ICODE,PCODE,NTAPS                                   FSMT0300
   10 FORMAT(A1,3X,A1,4X,I1)                                          FSMT0310
      WRITE(IOUT,14) ICODE,PCODE,NTAPS                                FSMT0320
C       IF ADDITION RUN, READ HEADER RECORD FROM OLD SUMMARY TAPE     FSMT0330
      IF(ICODE.EQ.IC) GO TO 100                                       FSMT0340
      WRITE(IOUT,11)                                                  FSMT0350
   11 FORMAT(' CONTROL CARD MISSING')                                 FSMT0360
      STOP                                                            FSMT0370
  100 IF(PCODE.EQ.I1) GO TO 101                                       FSMT0380
      IF(PCODE.EQ.I2) GO TO 102                                       FSMT0390
      WRITE(IOUT,12)                                                  FSMT0400
   12 FORMAT(' CONTROL CARD COL. 5 MUST BE 1 OR 2 ')                  FSMT0410
      STOP                                                            FSMT0420
   14 FORMAT('1',A1,3X,A1,4X,I1)                                      FSMT0430
C       READ HEADER RECORD FROM OLD SUMMARY TAPE                      FSMT0435
  102 READ (OSUMTP,25) IDATA1,IDATA2,IDATA3                           FSMT0440
   25 FORMAT(67A2,201A4,31A2)                                         FSMT0450
C       PRINT HEADER RECORD                                           FSMT0460
      WRITE(IOUT,15) IDATA1,IDATA2,IDATA3                             FSMT0470
```

TABLE E-I - (Continued)

```
   15 FORMAT(' OLD SUMMARY TAPE HEADER'/,1X,33A2/,1X,34A2/,1X,14,2X,  FSMT0480
     1 10(/1X,10(A4,I4,2X))/1X,31A2////)                              FSMT0490
      IDEX=IDATA2(1)                                                  FSMT0500
      ICOUNT=IDEX+2                                                   FSMT0510
      ISAVE=IDATA2(1)                                                 FSMT0520
C       READ HEADER CARD FOR FINAL SUMMARY TAPE                       FSMT0530
  101 READ(IN,13) ICODE,(IDATA1(I),I=1,40)                            FSMT0540
      IF(ICODE.EQ.IFF) GO TO 107                                      FSMT0550
      WRITE(IOUT,19)                                                  FSMT0560
   19 FORMAT(' HEADER CARD MISSING FOR FINAL SUMMARY TAPE')           FSMT0570
      STOP                                                            FSMT0580
  107 WRITE(IOUT,23) ICODE,(IDATA1(I),I=1,40)                         FSMT0590
   23 FORMAT(' NEW TAPE HEADER CARD '/1X,A1,2X,40A2)                  FSMT0600
C       READ HEADER RECORD FROM EACH MINI SUMMARY TAPE                FSMT0610
      DO 103 I=1,NTAPS                                                FSMT0620
      READ(IN,13) ICODE,IDENT                                         FSMT0630
   13 FORMAT(A1,39A2,A1)                                              FSMT0640
      ITAPE=JTAPE(I)                                                  FSMT0650
      READ(ITAPE,26) JDENT,JFILL,(JDATA(I,KK),KK=1,129)               FSMT0660
   26 FORMAT(67A2,129A4)                                              FSMT0670
      WRITE(IOUT,16) ICODE,IDENT                                      FSMT0680
   16 FORMAT(1X,A1,2X,40A2)                                           FSMT0690
      IF(ICODE.EQ.IH) GO TO 104                                       FSMT0700
      WRITE(IOUT,17)                                                  FSMT0710
   17 FORMAT(' TAPE HEADER CARD MISSING')                             FSMT0720
      STOP                                                            FSMT0730
C       CHECK FOR MATCH BETWEEN CARD AND TAPE HEADER                  FSMT0740
  104 DO 105 J=1,40                                                   FSMT0750
      IF(IDENT(J).EQ.JDENT(J)) GO TO 105                              FSMT0760
      WRITE(IOUT,18) JDENT                                            FSMT0770
   18 FORMAT(' TAPE HEADER DOES NOT MATCH HEADER CARD '/,1X,40A2)     FSMT0780
      STOP                                                            FSMT0790
  105 CONTINUE                                                        FSMT0800
      IDEX=JDATA(1,1)                                                 FSMT0810
      IDATA2(1)=IDATA2(1)+IDEX                                        FSMT0820
      JDEX=1                                                          FSMT0830
      DO 106 J=1,IDEX                                                 FSMT0840
      IDATA2(ICOUNT)=JDATA(I,JDEX+1)                                  FSMT0850
      IDATA2(ICOUNT+1)=JDATA(1,JDEX+2)                                FSMT0860
      ICOUNT=ICOUNT+2                                                 FSMT0870
  106 JDEX=JDEX+2                                                     FSMT0880
  103 CONTINUE                                                        FSMT0890
C       PRINT NEW SUMMARY TAPE HEADER RECORD                          FSMT0900
      WRITE(IOUT,20) IDATA1,IDATA2,IDATA3                             FSMT0910
   20 FORMAT(' NEW SUMMARY TAPE HEADER'/,1X,33A2/,1X,34A2/,1X,14,2X,  FSMT0920
     1 10(/1X,10(A4,I4,2X))/1X,31A2////)                              FSMT0930
C       WRITE HEADER RECORD ON NEW SUMMARY TAPE                       FSMT0940
      WRITE(NSUMTP,25) IDATA1,IDATA2,IDATA3                           FSMT0950
      IF(PCODE.EQ.I1) GO TO 108                                       FSMT0960
C       READ OLD SUMMARY TAPE AND WRITE ONTO NEW SUMMARY TAPE         FSMT0970
      DO 109 I=1,ISAVE                                                FSMT0980
      READ(OSUMTP,27) (KDATA(J),J=1,94)                               FSMT0990
   27 FORMAT(94A4)                                                    FSMT1000
      WRITE(NSUMTP,27)(KDATA(J),J=1,94)                               FSMT1010
      IDEX=IDEX+2
      KDEX=IDATA2 (IDEX+1)
C       READ AND WRITE INTERVAL RECORDS                               FSMT0960
      DO 110 K=1,KDEX                                                 FSMT0970
      READ(OSUMTP,28) KDATA                                           FSMT0980
```

TABEL E-I - (Concluded)

```
   28 FORMAT(25CA4,25DA4,7CA4)                                          FSMT1015
  110 WRITE(NSUMTP,28) KDATA                                            FSMT1020
C         READ AND WRITE VOYAGE SUMMARY RECORD                          FSMT1030
      READ(OSUMTP,29) (KDATA(J),J=1,100)                                FSMT1040
   29 FORMAT(100A4)                                                     FSMT1045
      WRITE(NSUMTP,29) (KDATA(J),J=1,100)                               FSMT1050
  109 WRITE(IOUT,21)(KDATA(J),J=1,100)                                  FSMT1060
   21 FORMAT(' OLD VOYAGE SUMMARY',3(/1X,25A4)/,1X,19A4/,1X,6(I8,2X))   FSMT1070
C                                                                       FSMT1080
C         BEGIN COPYING MINI SUMMARY TAPES                              FSMT1090
C                                                                       FSMT1100
  108 DO 111 I=1,NTAPS                                                  FSMT1110
      ITAPE=JTAPE(I)                                                    FSMT1120
      KDEX=JDATA(I,1)                                                   FSMT1130
      DO 112 IJ=1,KDEX                                                  FSMT1140
C         READ AND WRITE VOYAGE IDENTIFICATION                          FSMT1150
      READ(ITAPE,29) (KDATA(J),J=1,94)                                  FSMT1060
      WRITE(NSUMTP,29)(KDATA(J),J=1,94)                                 FSMT1170
      MDEX=IJ*2                                                         FSMT1180
      LDEX=JDATA(I,MDEX+1)                                              FSMT1190
      DO 113 JK=1,LDEX                                                  FSMT1200
C         PASS DIGITIZED INTERVAL DATA                                  FSMT1210
      DO 114 KL=1,12                                                    FSMT1220
  114 READ(ITAPE,29) IDUM                                               FSMT1230
C     READ AND WRITE INTERVAL RECORDS                                   FSMT1240
      READ(ITAPE,28) KDATA                                              FSMT1250
  113 WRITE(NSUMTP,28) KDATA                                            FSMT1260
C         READ AND WRITE VOYAGE SUMMARY RECORD                          FSMT1270
      READ(ITAPE,29) (KDATA(J),J=1,100)                                 FSMT1280
      WRITE(NSUMTP,29)(KDATA(J),J=1,100)                                FSMT1290
  112 WRITE(IOUT,22)(KDATA(J),J=1,100)                                  FSMT1300
   22 FORMAT('-NEW VOYAGE SUMMARY',3(/1X,25A4),7,1X,19A4/,1X,6(I8,2X))  FSMT1310
  111 REWIND ITAPE                                                      FSMT1320
      END FILE NSUMTP                                                   FSMT1330
      STOP                                                              FSMT1340
      END                                                               FSMT1390
/*
//GO.FT10F001 DD UNIT=TAPE9,LABEL=(,NLT,VOLUME=SER=BEAR01,
//             DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=4000)
//GO.FT11F001 DD UNIT=TAPE9,LABEL=(,NL),VOLUME=SER=BEAR02,
//             DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=4000)
//GO.FT12F001 DD UNIT=TAPE9,LABEL=(,NL),VOLUME=SER=BEAR03,
//             DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=4000)
//GO.FT13F001 DD UNIT=TAPE9,LABEL=(,NL),VOLUME=SER=SCAN01,
//             DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=4000)
//GO.FT14F001 DD UNIT=TAPE9,DISP=(NEW,PASS),LABEL=(,NL),
//             VOLUME=SER=RESLT,DCB=(RECFM=U,BLKSIZE=4000,DEN=2)
//GO.FT05F001 DD *
C   1    4
F  INTERIM SUMMARY TAPE NUMBER ONE
HBEAR 01  DIGITAL DATA TAPE OF CALIFORNIA BEAR, CONTRACT NSSC-N00024-69-C-5161
HBEAR 02  DIGITAL DATA TAPE OF CALIFORNIA BEAR, CONTRACT NSSC-N00024-69-C-5161
HBEAR 03  DIGITAL DATA TAPE OF CALIFORNIA BEAR, CONTRACT NSSC-N00024-69-C-5161
HSCAN 01  DIGITAL DATA TAPE OF MORMACSCAN, CONTRACT NSSC-N00024-69-C-5161
/*
```

APPENDIX F

SUMMARY TAPE CORRECTION PROGRAM

INTRODUCTION

This program (CRCT) was written to incorporate the subroutine RELWND (see Appendix I) and thus provide capability for correction of the Final Summary Tapes. An error in programming of the conversion of (recorded) True Wind Direction to the required Relative Wind Direction (Appendix A) resulted in some Relative Wind Direction values to be 180° in error. Since the punched-paper tapes were created with the program error, and all the data were subsequently processed from these tapes, the correction has been made only on the Final Summary Tapes (see Appendix E).

The program listing is given in Table F-I and the flow chart in Figure F-1.

The program reads an uncorrected Final Summary Tape, reads each interval record, and creates a corrected Final Summary Tape. The description of the RELWND Subroutine, (Appendix I) gives the details for the correction process.

PROGRAM DETAILS

Input: (1) Final Summary Tape
(to be corrected)

Output: (1) Final Summary Tape
(corrected)

(2) Printed Output

A. SUMMARY TAPE HEADER RECORD

B. VOYAGE XXX
voyage number from each
voyage identification record

	1	2	3	4	5	6	7	8
C.	XXX	XXX	XXXX	XXXX	XX	XX	X	XXXX

(1) Interval number

(2) Course

(3) Original relative wind direction

(4) Speed

(5) True Wind Velocity

(6) Relative wind velocity

(7) Indica

(8) Corrected relative wind direction. This message appears for each interval record.

	1	2
D. BLANK	XXXX	XXXX

(1) true wind velocity

(2) course

This message appears if the true wind velocity, course or relative wind direction are blank in the interval record.

Subroutine:

RELWND - (App. I)

BCNV - Basic Conversion Subroutine available from IBM manual Form C27-6932-3, Appendix J as added by TNL N27-1313, July 23, 1969

The BCNV subroutine allows the FORTRAN or Assembler Language GSP user to change the format of data in storage when the data is in hexadecimal, integer, floating point, or EBCDIC E, F, I, or Z formats. The user-specified data is converted to the format requested in the CALL statement, and is placed in the output area designated. For information about the use of the various FORTRAN formats, see the publication IBM System/360 FORTRAN IV Language, Form C28-6515.

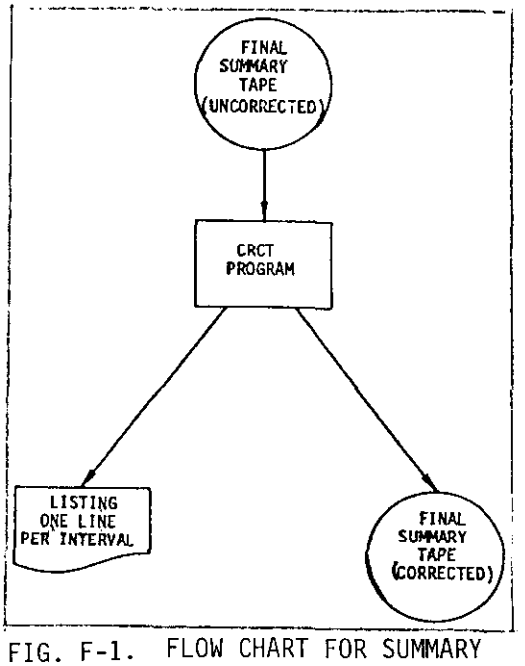


FIG. F-1. FLOW CHART FOR SUMMARY TAPE-CORRECTION PROGRAM

TABLE F-I - PROGRAM LISTING OF SUMMARY TAPE-CORRECTION PROGRAM

```
// JOB
//YGECRCTR JOB (S3110005842,-----,101,---,PETRVC,C1801-------,7400),W2,K
//         MSGLEVEL=1
//WIND EXEC FURTGCLG,PARM.FORT='NODECK,MAP',TIME.GO=42
//FORT.SYSIN DD *
C                                                                       CRCT0010
C     PROGRAM TO READ SUMMARY TAPE AND CORRECT RELATIVE WAVE DIRECTION  CRCT0020
C     IN EACH INTERVAL RECORD                                           CRCT0030
C         SUBROUTINES USED ARE                                          CRCT0040
C             RELWND                                                    CRCT0050
C             BCNV                                                      CRCT0060
      EXTERNAL ERROR                                                    CRCT0061
      INTEGER*4  OSUMTP                                                 CRCT0065
      INTEGER*2 KDIR,IDATA(1158),IVOYGE(188),NHEDR(134),LHEDR(31)       CRCT0070
      INTEGER*2 IRSLT2,KBLNK                                            CRCT0080
      DIMENSION IHEDR(301)                                              CRCT0090
      COMMON ISW1                                                       CRCT0091
      DATA IBLNK,KBLNK/' ',' '/                                         CRCT0100
      NSUMTP=11                                                         CRCT0110
      OSUMTP=10                                                         CRCT0120
      IOUT=6                                                            CRCT0130
      CALL ERRSET(215,101,0,0,ERROR)                                    CRCT0131
      ISW1=0                                                            CRCT0132
      REWIND NSUMTP                                                     CRCT0140
      REWIND OSUMTP                                                     CRCT0150
C         READ HEADER RECORD                                            CRCT0160
      READ(OSUMTP,24) NHEDR,IHEDR,LHEDR                                 CRCT0170
   24 FORMAT(134A1,Z200A4,101A4,31A2)                                   CRCT0180
      WRITE(IOUT,25) NHEDR,IHEDR                                        CRCT0190
   25 FORMAT('1',' SUMMARY TAPE HEADER RECORD'/' ',67A1/,               CRCT0200
     1 1X,67A1/,1X,14,2X,15(1X,10(A4,I4,2X)))                           CRCT0210
C         SET UP INDICIES FOR READING SUMMARY TAPE                      CRCT0220
      IDEX=IHEDR(1)                                                     CRCT0230
      ICOUNT=1                                                          CRCT0240
C         WRITE HEADER RECORD                                           CRCT0250
      WRITE(NSUMTP,24) NHEDR,IHEDR,LHEDR                                CRCT0260
C         BEGIN READING SUMMARY TAPE                                    CRCT0270
      DO 10 I=1,IDEX                                                    CRCT0280
      READ(OSUMTP,27) IVOYGE                                            CRCT0290
   27 FORMAT (188A2)                                                    CRCT0300
      WRITE(IOUT,37) IVOYGE(9),IVOYGE(10)                               CRCT0301
   37 FORMAT('1 VOYAGE  ',A2,A1)                                        CRCT0302
      WRITE(NSUMTP,27) IVOYGE                                           CRCT0310
      LDEX=I*2                                                          CRCT0320
      KDEX=IHEDR(LDEX+1)                                                CRCT0330
      DO 20 II=1,KDEX                                                   CRCT0340
      READ(OSUMTP,29)(IDATA(K),K=1,45),KCRSE,KSPEED,(IDATA(K),K=46,     CRCT0350
     151), KRWND,KDIR,KRWVEL,KTWVEL,(IDATA(K),K=52,57),                 CRCT0360
     2KRSWLL,(IDATA(K),K=58,1157)                                       CRCT0370
   29 FORMAT(45A1,A3,A4,6A1,A3,A1,A2,A2,5A2,A1,A4,A1,4(250A2),99A2)     CRCT0380
      LRWND=KRWND                                                       CRCT0381
      INDICA=0                                                          CRCT0390
      IF(KRWND.EQ.IBLNK) GO TO 45                                       CRCT0395
C         IF TRUE WIND VELOCITY IS BLANK, SKIP OVER                     CRCT0400
      IF(KTWVEL.EQ.IBLNK) GO TO 45                                      CRCT0410
C         IF COURSE IS BLANK, BLANK OUT RELATIVE VALUES                 CRCT0420
      IF(KCRSE.NE.IBLNK) GO TO 40                                       CRCT0430
   45 KRWND=IBLNK                                                       CRCT0440
      KDIR=KBLNK                                                        CRCT0450
```

TABLE F-I - (Concluded)

```
      KRWVEL=IBLNK                                                      CRCT0460
      IDATA(52)=KBLNK                                                   CRCT0470
      IDATA(53)=KBLNK                                                   CRCT0480
      KRSWLL=IBLNK                                                      CRCT0490
   45 IRSLT2=KDIR                                                       CRCT0491
      WRITE(IOUT,36) KTWVEL,KCRSE                                       CRCT0492
   36 FORMAT(' BLANK',2X,A4,2X,A4)                                      CRCT0493
      GO TO 41                                                          CRCT0500
   40 IF(KTWVEL.EQ.IBLNK) INDICA=1                                      CRCT0510
      CALL BCNV(KCRSE,ICRSE,203,3)                                      CRCT0520
      CALL BCNV(KSPEED,SPEED,202,4,1)                                   CRCT0530
      CALL BCNV(KRWND,IRWND,203,3)                                      CRCT0540
      CALL BCNV(KRWVEL,IRWVEL,203,2)                                    CRCT0550
      CALL BCNV(KTWVEL,ITWVEL,203,2)                                    CRCT0560
      IF(ISW1) 46,46,47                                                 CRCT0561
C         GET CORRECTED VALUE OF RELATIVE WIND DIRECTION                CRCT0570
   46 CALL RELWND(ICRSE,IRWND,KDIR,SPEED,ITWVEL,IRWVEL,INDICA,         CRCT0580
     1                IRSLT1,IRSLT2)                                   CRCT0590
      CALL BCNV(IRSLT1,KRWND,203,3)                                     CRCT0600
   41 WRITE(NSUMTP,29)(IDATA(K),K=1,45),KCRSE,KSPEED,(IDATA(K),K=46,51),CRCT0610
     1              KRWND,IRSLT2,KRWVEL,KTWVEL,(IDATA(K),K=52,         CRCT0620
     2              57),KRSWLL,(IDATA(K),K=58,1157)                    CRCT0630
      WRITE(IOUT,35) (IDATA(K),K=16,18),KCRSE,LRWND,KDIR,KSPEED,KTWVEL, CRCT0631
     1  KRWVEL,INDICA,KRWND,IRSLT2                                      CRCT0632
   35 FORMAT(' ',3A1,2X,A3,2X,A3,A1,2X,A4,2X,A2,2X,A2,2X,A1,2X,A3,A1)   CRCT0633
      ISW1=0                                                            CRCT0634
      GO TO 20                                                          CRCT0635
   47 KCRSE=IBLNK                                                       CRCT0636
      KTWVEL=IBLNK                                                      CRCT0637
      GO TO 40                                                          CRCT0638
   20 CONTINUE                                                          CRCT0640
C         READ AND WRITE VOYAGE SUMMARY RECORD                          CRCT0650
      READ(OSUMTP,30)(IDATA(K),K=1,200)                                 CRCT0660
   30 FORMAT(200A2)                                                     CRCT0670
      WRITE(NSUMTP,30)(IDATA(K),K=1,200)                                CRCT0680
   10 CONTINUE                                                          CRCT0690
      END FILE NSUMTP                                                   CRCT0700
      STOP                                                              CRCT0710
      END                                                               CRCT0720
```

APPENDIX G

SUMMARY TAPE LISTING PROGRAM

This program (LIST) was included to provide a capability to list the data which appears on the data tapes. An option is provided to suppress the printing of the 12,000 points of digitized record and print only Identification and Summary data for intervals and voyages.

A listing of the program is given in Table G-I.

TABLE G-I - PROGRAM LISTING OF SUMMARY TAPE-LISTING PROGRAM

```
//STEP4 EXEC FORTGCLG,PARM.FORT='NODECK,MAP'
//FORT.SYSIN DD *
      DIMENSION  IDENT(12000),DATA(12000),INT(4)
      DIMENSION IDATA(299)
      INTEGER *2 DATA
      REWIND 1
      N= 10
      DO 5 I=1,1000
      DATA(I)=0
    5 IDENT(I)=0
C         READ HEADER RECORD
      READ(1,101) IDATA
  101 FORMAT(67A2,2X,31A4,31A2)
      WRITE(6,1)
    1 FORMAT(1H1)
      WRITE(6,102) IDATA
  102 FORMAT(1H ,56A2/,1X,17A2/,1X,14/,10(/1X,10(A4,I4,2X))/1X,31A2///)
      INUM=IDATA(64)
      INUM=3
      DO 105 ME=1,INUM
      READ(1,106) (DATA(KK),KK=1,188)
  106 FORMAT(188A2)
      WRITE(6,107) (DATA(K),K=1,188)
  107 FORMAT(1H1,49A2,3(/,2X,47A2))
      JJ=ME*2+68
      KNUM=IDATA(JJ)
      DO 70 MI=1,KNUM
      IF(MI-N) 7,3,7
    3 WRITE(6, 1)
    7 DO 60 M=1,12
      READ(1,6)( DATA(L),L=1,1000)
      IF(MI-N)60,9,60
    9 J=0
      DO  50 I=1,50
      J=(I-1)*20+1
      K=J+19
      WRITE(6,12)( DATA(L),L=J,K)
      IF(I-25) 50,40,30
   30 IF(I- 50) 50,40,50
   40 WRITE(6, 1)
   50 CONTINUE
   60 CONTINUE
   12 FORMAT(1H ,1A16)
    4 WRITE(6, 1)
      READ(1,2)(IDENT(I),I=1,570)
      WRITE(6, 8)(IDENT(I),I=1, 64)
      WRITE(6,12)(IDENT(I),I= 65, 76)
      WRITE(6,12)(IDENT(I),I= 77,570)
   70 CONTINUE
      WRITE(6, 1)
      READ(1,2)(IDENT(I),I=1,100)
      WRITE(6, 8)(IDENT(I),I=1, 94)
      WRITE(6,12)(IDENT(I),I= 95,100)
  105 CONTINUE
    6 FORMAT(50(20A2))
    2 FORMAT(50(20A4))
    8 FORMAT(1H ,25A4)
      STOP
      END
/*
//GO.FT01F001 DD DSN=*.STEP1.GO.FT14F001,DISP=(OLD,KEEP),UNIT=TAPE9,
//             VOLUME=REF=*.STEP1.GO.FT14F001,
//             DCB=(RECFM=U,BLKSIZE=4000)
//GO.FT05F001 DD DUMMY
/*
```

APPENDIX H

PARAMETRIC-STUDIES PROGRAM

INTRODUCTION

The Parametric-Studies program (PARM) was written to provide the capability to read the Final Summary tapes and selectively extract data. The program has been written in as general a form as possible to provide the greatest options for flexibility. Output can be in the form of printed output, or stored on magnetic tape for subsequent analysis or dumping on cards. The program does not include the capability for analysis directly from magnetic tape output.

GENERAL DESCRIPTION

The program allows the user to specify criteria which each interval must meet. If these criteria are met, the user may specify a number of different data fields from the interval record to be provided as output of the program. For example, the user may request that for every interval which has a Beaufort Sea State greater than four, he should obtain as output, the Relative Wind Direction for that interval. In addition, for every interval which meets his criteria, he will always obtain the interval identification (tape reference number, interval number, and logbook index number) as part of the output.

For each run, the user may request up to four criteria for each interval and up to twelve fields of output (in addition to the criteria field which are always contained in the output). The user makes his specific requests on input data cards--one card for each criteria and one card for each output field. In addition, a run control card is required and a tape-header card is required to identify the Final Summary Tape to be used.

The system flow chart is given in Figure H-1 and the detailed flow charts are given in Figure H-2. The program listing is given in Table H-1.

VARIABLE NAMES

NCOMPS - number of compares to be made--number of criteria to be met

NPRNTS - number of fields to be punched/printed

ISTUDY - study code

Criteria Card

ITYPE(4) - type of field (I,A,F) to be compared

ICODE(4) - "hit or miss" code

LOCAT(4) - location of field to be compared

ISTART(4) - start of compare field

ILEN(4) - length of compare field

ICOND(4) - condition of compare

IVALU(4,12) - actual value to be compared

Print/Punch Card

NLOC(16) - location of print/punch field

INTV(16) - this interval or next

NSTART(16) - start of print/punch field

NLEN(16) - length of print/punch field

Interval Record

INTERV(28Ø) - this interval information

NINTRV(28Ø) - next interval information

IVOYGE(256) - voyage identification data

ICOMP(4) - storage of integer values for comparing

COMP(4) - storage of floating point values for comparing

PROGRAM LIMITATIONS

For summary numbers, have length of 1 and begin at 257,258...262

For character compare, only EQ allowed

For floating point compare, EQ not allowed

If large print field is used, must be on last print/punch card of group

On input of value on compare card, characters are left justified, integer and floating point numbers are right justified.

Integer is maximum of 6 digits including sign

Floating point is maximum of 10 digits including sign and decimal point

Character is maximum of 12 characters

INPUT

Data Cards

1. "C" type Run Control Card
2. "S" type Compare Card(s)
3. "P" type Print/Punch Card(s)
4. "F" type Final Summary Tape Identification Card

"C" type Run Control Card

Col.	Description
1	C
5	number of compares (1-4)
9-10	number of print punch cards (1-12)
15	Study Code

"S" type Compare Card

Col.	Description
1	S
2	type of field { I=integer, A=EBCDIC, F=floating pt.
3	code { Ø keep for a hit, 1 keep for a miss
4	location of field { Ø-interval, 1-voyage
5-7	start of field
8-9	length of field (in bytes)
10-11	type of compare condition EQ, LE, GE, GT, LT
12-23	value to be compared against (up to 12 Cols.) A type is left justified I,F type are right justified I is maximum of 6 characters including sign F is maximum of 10 characters including sign

One card required for each compare.

"P" type Print/Punch Card

Col.	Description
1	P
2	location of field { 0 interval, 1 voyage
3	{ Ø - this interval, 1 - next interval
4-6	start of field
7-8	length of field
	(maximum of 12 except 1 of length 2Ø)

One card required for each field to be printed/punched

"F" type Final Summary Tape Header Card

Col.	Description
1	F
2-8Ø	description

OUTPUT

Cards

Printed and on magnetic tape are card images

Col.	Description
1	study code
2	card in Series 1-4
3-14	Tape Reference Number
15-17	Logbook Index
18-20	Interval Number
21-32	Field No. 1
33-44	" " 2
45-56	" " 3
57-68	" " 4
69-80	" " 5

(1) first four fields on first card are the Sort fields (left blank if not used)

(2) maximum of 3 of these cards/interval

(3) plus a 4th card if required with a 20 col. print field
Cols. 21-40

Printed

1. NO. OF COMPARES=XX NO. OF FIELDS TO PRINT/PUNCH=XX
STUDY CODE=X
information from "C" type control card

2. COMPARE INFORMATION
(printed from cards)
compare control card listing

3. PRINT/PUNCH INFORMATION

(printed from cards)

print/punch control card listing

4. TAPE HEADER INFORMATION

(printed from cards)

printout of tape header control card

5. SUMMARY TAPE HEADER RECORD

(printed from cards)

printout of summary tape header record

6. LISTING OF MAGNETIC TAPE OUTPUT,
one to four lines for each interval which meets criteria, depending on number of field requested in print/punch cards.

5 of these per line

X	X	XXXXXXXXXXXX	XXXXXX	XXXXXXXXXXXX
Study Code	Card No.	Tape Ref. Number	Log-book Index / Intrv. No.	Print/punch output field

Interval Identification

and X X XXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXX

same as above except print/punch output field is 20 char. and only one per line--this appears only if a large print/punch field is requested or if the max. no. of fields are requested.

JOB SUCCESSFULLY COMPLETED

7. Error Messages

A. RUN CONTROL CARD MISSING

"C" type control card missing or out of sequence - correct and reenter

B. MAX. ALLOWABLE CRITERIA IS 4

more than 4 criteria are requested on Run Control Card - correct and reenter

C. MAX. ALLOWABLE PRINT/PUNCH REQUEST IS 12

more than 12 print/punch fields are requested on Run Control Card - correct and reenter

D. COMPARE CARD MISSING OR OUT OF SEQUENCE

self-explanatory - correct and reenter

E. PRINT/PUNCH CARD MISSING OR OUT OF SEQUENCE

self-explanatory - correct and reenter - must agree with number of requests on Run Control Card

F. HEADER CARD DESCRIPTION DOES NOT MATCH TAPE RECORD

description on "F" type header card does not match description in summary tape header record. Check tape and header card - correct and reenter

G. INTEGER COMPARE REQUIRED FOR SUMMARY NO.
if field compared against criteria is a summary number, the type of compare must be specified as integer - correct criteria card and reenter

H. EQ IS NOT ALLOWED AS A COMPARE CONDITION FOR AN F TYPE NUMBER

self explanatory - correct criteria card and reenter

I. ONLY EQ IS ALLOWED AS A COMPARE CONDITION FOR AN "A" TYPE COMPARE

self-explanatory - correct criteria card and reenter

J. PRINT FIELD LARGER THAN 12 MUST BE LAST PRINT/PUNCH CARD ENTERED

check the sequence of print/punch cards and reenter

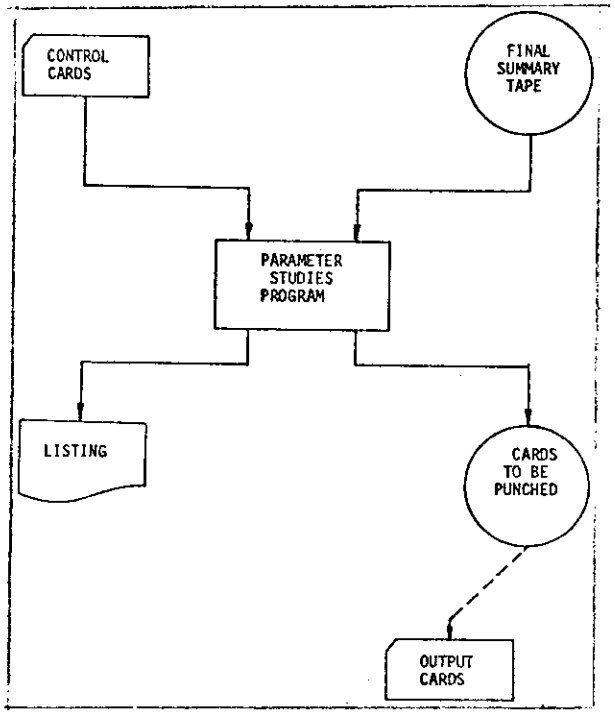


FIG. H-1. SYSTEM FLOW CHART FOR PARA-METRIC-STUDIES PROGRAM

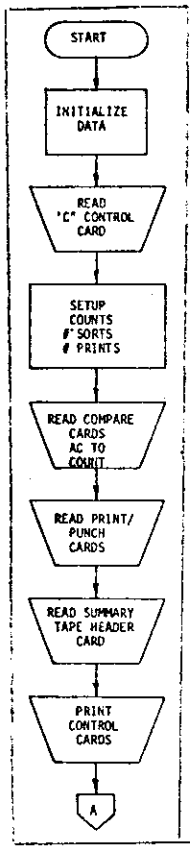


FIG. H-2. DETAIL FLOW CHART FOR PARAMETRIC-STUDIES PROGRAM

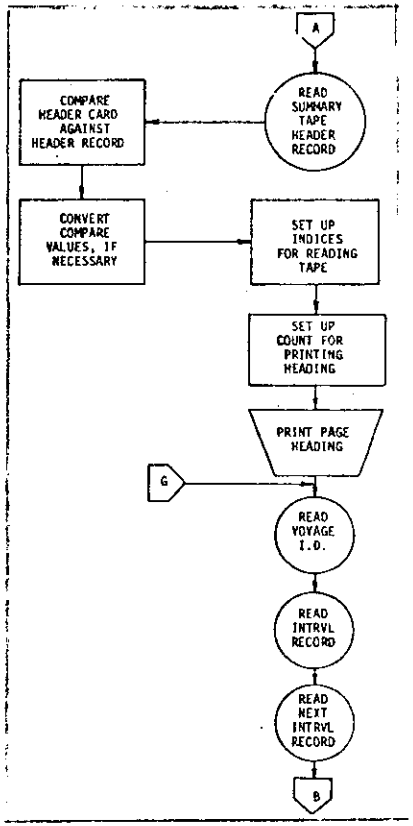


FIG. H-2 (Continued)

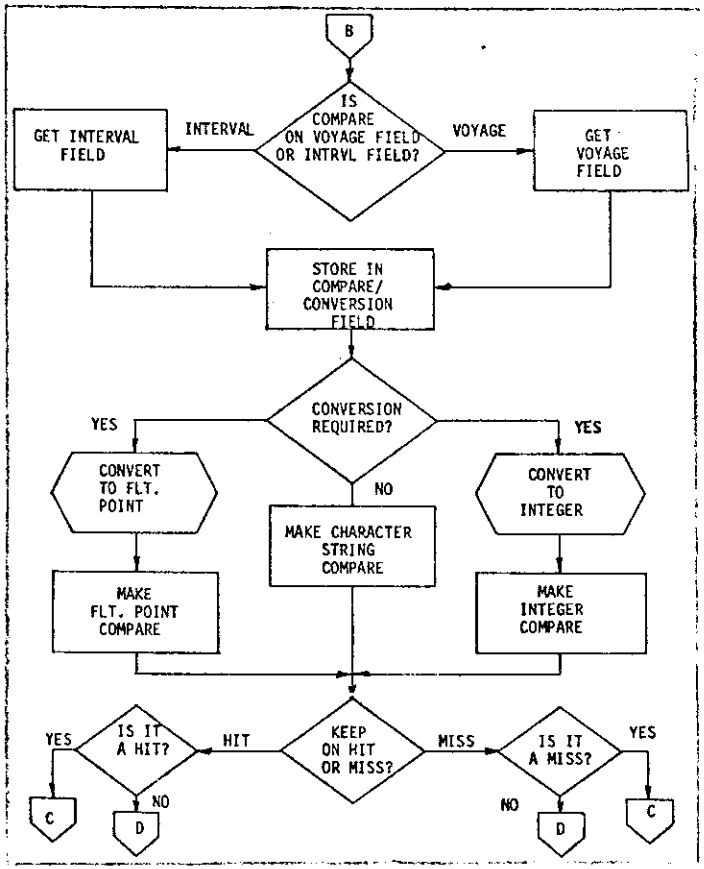


FIG. H-2 - (Continued)

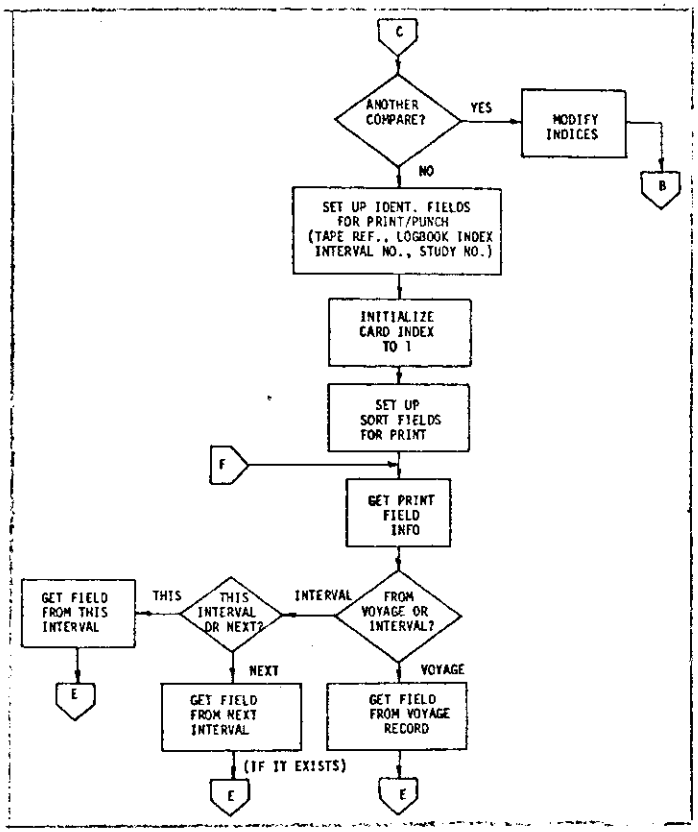


FIG. H-2 - (Continued)

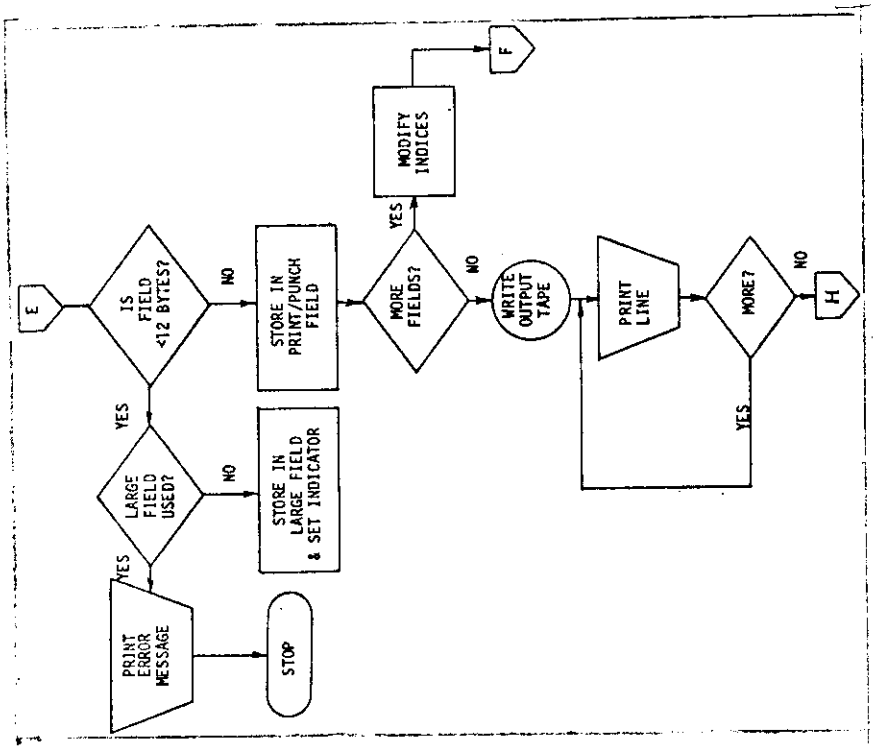


FIG. H-2 - (Continued)

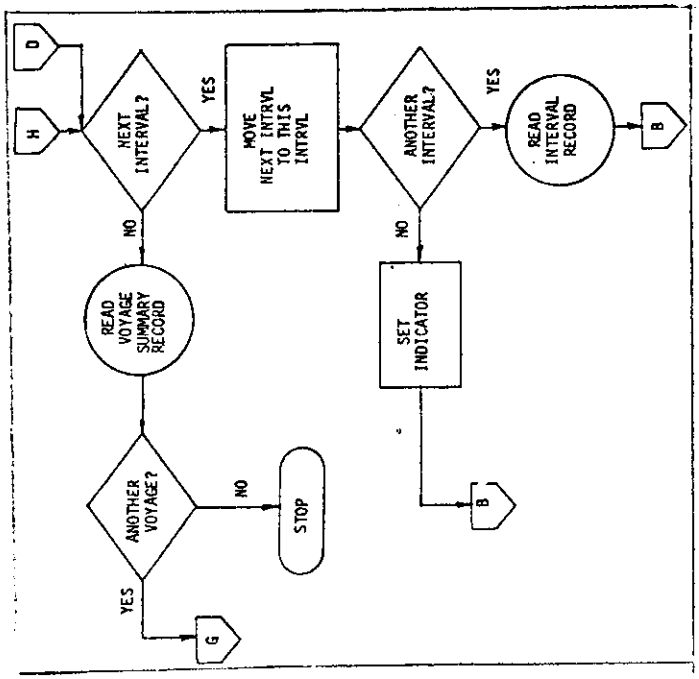


FIG. H-2 - (Concluded)

TABLE H-I - PROGRAM LISTING FOR PARAMETRIC STUDIES PROGRAM

```
//E24PARM  JOB (S3)10006964,-----,101,---,FLAMER,CJ198-------,7400),AX,A
//             MSGLEVEL>1
//ALNIST EXEC ASMFC,PARM.ASM>(XREF,LIST,NODECK,LOAD)
//ASM.SYSIN  DD  *
ALIGN    CSECT
*
*
*        ***** ROUTINE TO ALIGN DATA PRIOR TO CALLING
*              THE GSP CONVERT ROUTINE.  USE BY CODING
*
*              CALL ALIGN USING INARRAY, LENGTH, OUTARRAY
*
*              WHERE.....
*
*              INARRAY IS THE ADDRESS OF THE ARRAY CONTAINING
*                      THE DATA SEPRTED BY BLANKS.
*
*              LENGTH  IS AN INTEGER CONSTANT THAT
*                      IS THE NUMBER OF BYTES IN
*                      THE INPUT ARRAY.  THIS CONSTANT
*                      MUST BE AN EVEN NUMBER AND MUST
*                      NOT BE GREATER THAN 256 BYTES.
*
*              OUTARRAY IS THE ADDRESS OF AN ARRAY
*                      THAT IS TO CONTAIN THE ALIGNED DATA
*                      INTERSPERSED WITH BLANKS.
*
*
*
         SAVE  (14,12),T,*
         BALR  10,0
         USING *,10
         LA    12,SVAREA
         ST    12,8(13)
         ST    13,SVAREA<4
         LR    13,12
         LM    3,5,0(1)
*
*        R-3   > ADDRESS OF INPUT ARRAY
*        R-4   > ADDRESS OF CONSTANT CONTAINING ARRAY LENGTH.
*        R-5   > ADDRESS OF OUTPUT ARRAY
*
         MVC   TABLE(128),INIT
         SR    6,6
         L     7,0(4)
         D     6,>F'2'
         BCT   7,*<4
         EX    7,TRINST
         EX    7,MVINST
         L     13,SVAREA<4
         RETURN (14,12)
*
*
MVINST   MVC   0(0,5),TABLE
TRINST   TR    TABLE(0),0(3)
*
*
SVAREA   DS    18F
*
TABLE    DC    X'00020406080A0C0E'
```

TABLE H-I - (Cont.)

```
         DC    X'10121416181A1C1E'
         DC    X'20222426282A2C2E'
         DC    X'30323436383A3C3E'
         DC    X'40424446484A4C4E'
         DC    X'50525456585A5C5E'
         DC    X'60626466686A6C6E'
         DC    X'70727476787A7C7E'
         DC    X'80828486888A8C8E'
         DC    X'90929496989A9C9E'
         DC    X'A0A2A4A6A8AAACAE'
         DC    X'B0B2B4B6B8BABCBE'
         DC    X'C0C2C4C6C8CACCCE'
         DC    X'D0D2D4D6D8DADCDE'
         DC    X'E0E2E4E6E8EAECEE'
         DC    X'F0F2F4F6F8FAFCFE'
INIT     DC    X'00020406080A0C0E'
         DC    X'10121416181A1C1E'
         DC    X'20222426282A2C2E'
         DC    X'30323436383A3C3E'
         DC    X'40424446484A4C4E'
         DC    X'50525456585A5C5E'
         DC    X'60626466686A6C6E'
         DC    X'70727476787A7C7E'
         DC    X'80828486888A8C8E'
         DC    X'90929496989A9C9E'
         DC    X'A0A2A4A6A8AAACAE'
         DC    X'B0B2B4B6B8BABCBE'
         DC    X'C0C2C4C6C8CACCCE'
         DC    X'D0D2D4D6D8DADCDE'
         DC    X'E0E2E4E6E8EAECEE'
         DC    X'F0F2F4F6F8FAFCFE'
   LTORG
         END
/*
//STEP2 EXEC FORTGCLG
//FORT.SYSIN DD *
```

TABLE H-I - (Continued)

```
C          PARAMETRIC STUDIES PROGRAM                                    PARM0020
C                                                                        PARM0030
C     THE PARAMETRIC STUDIES PROGRAM ACCEPTS COMPARE CRITERIA            PARM0040
C     ON PUNCHED CARDS, EXAMINES EACH INTERVAL ON THE INPUT              PARM0050
C     SUMMARY TAPE ACCORDING TO THIS CRITERIS AND IF                     PARM0060
C     THE INTERVAL MEETS THESE CRITERIA PRODUCES                         PARM0070
C     CERTAIN OUTPUT DESIGNATED ON PRINT/PUNCH CONTROL CARDS             PARM0080
C                                                                        PARM0090
C               MAX. NO. OF CRITERIA IS 4                                PARM0100
C               MAX. NO. OF PRINT / PUNCH REQUEST IS 12                  PARM0110
C                                                                        PARM0120
C     THE OUTPUT IS IN THE FORM OF PRINTED OUTPUT AND                    PARM0130
C     A TAPE DATA SET                                                    PARM0140
C                                                                        PARM0150
C                                                                        PARM0160
      EXTERNAL ERROR                                                     PARM0165
      INTEGER*2  IVALU(12,4),NDESC(79),INTERV(256),NINTRV(256)           PARM0170
      INTEGER*2 IVOYGE(256),NHEDR(134),ITEMP1(3),ITEMP2(5)               PARM0180
      INTEGER*2 KBLANK,ITEMP4(6),ITEMP5(10),IDENT(15)                    PARM0190
      INTEGER*2 ITEMP3(4,12),ITEMP8(3),ITEMP9(4),ITEMPA(20)              PARM0200
      INTEGER*2 LPRINT(20),IPRINT(180),IFILL(40)                         PARM0210
      INTEGER*2 I5                                                       PARM0215
      DIMENSION ITYPE(4),ICODE(4),LOCAT(4),ISTART(4),ILEN(4),ICOND(4)    PARM0220
      DIMENSION NLOC(12),INTV(12),NSTART(12),NLEN(12),IHEDR(301)         PARM0230
      DIMENSION ICOMP(4),COMP(4),ISUM(6),NSUM(6),ISW(4)                  PARM0240
      DIMENSION KSRT(4)                                                  PARM0241
      COMMON ISWERR                                                      PARM0242
      DATA IC,IS,IP,IFF/'C','S','P','F'/,ICOMP,COMP/4*0,4*0.0/           PARM0250
      DATA IA/'A'/,IB/' '/,IEQ,ILE,IGE,IGT,ILT/'EQ','LE','GE',           PARM0260
     3 'GT','LT'/,IT/'T'/,KBLANK/'  '/                                   PARM0270
      DATA I1,I2,I3,I4/'1','2','3','4'/                                  PARM0280
      DATA IFILL/40*'  '/                                                PARM0290
      IKK = 0
      IPUN= 7                                                            PARM0295
      IN=5                                                               PARM0300
      IOUT=6                                                             PARM0310
      ISUMTP=9                                                           PARM0320
      IPNCH=10                                                           PARM0330
      CALL ERRSET(215,10,10,10,ERROR)
      ISWERR = 0                                                         PARM0241
      REWIND IPNCH                                                       PARM0340
      REWIND ISUMTP                                                      PARM0350
C          READ RUN CONTROL CARD                                         PARM0360
      READ(IN,10) NCODE,NCOMPS,NPRNTS,ISTUDY,ICARD                       PARM0370
   10 FORMAT(A1,3X,I1,3X,I2,4X,A1,4X,I1)                                 PARM0380
      IF( NCODE.EQ.IC) GO TO 101                                         PARM0390
      WRITE(IOUT,11)                                                     PARM0400
   11 FORMAT(' RUN CONTROL CARD MISSING')                                PARM0410
      STOP                                                               PARM0420
  101 IF(NPRNTS.GT.12) GO TO 199                                         PARM0430
      IF(NCOMPS.LE.4) GO TO 191                                          PARM0440
      WRITE(IOUT,42)                                                     PARM0450
   42 FORMAT(' MAX. ALLOWABLE CRITERIA IS 4')                            PARM0460
      STOP                                                               PARM0470
  199 WRITE(IOUT,43)                                                     PARM0480
   43 FORMAT(' MAX ALLOWABLE PRINT/PUNCH REQUEST IS 12')                 PARM0490
      STOP                                                               PARM0500
```

TABLE H-I - (Continued)

```
C        READ COMPARE CARDS                                              PARM0510
  191 DO 102 I=1,NCOMPS                                                  PARM0520
      READ(IN,12) NCODE,ITYPE(I),ICODE(I),LOCAT(I),ISTART(I),ILEN(I),    PARM0530
     1 ICOND(I),(IVALU(J,I),J=1,12),KSWT(I)                              PARM0540
   12 FORMAT(2A1,2I1,I3,I2,A2,12A1,I1)                                   PARM0550
      IKK = IKK + KSWT(I)                                                PARM0560
      IF(NCODE.EQ.ISS) GO TO 102                                         PARM0570
      WRITE(IOUT,13)                                                     PARM0580
   13 FORMAT(' COMPARE CARD MISSING OR OUT OF ORDER')                    PARM0590
      STOP                                                               PARM0600
  102 CONTINUE                                                           PARM0610
C        READ PRINT/PUNCH CARDS                                          PARM0620
      DO 103 I=1,NPRNTS                                                  PARM0630
      READ(IN,14) NCODE,NLOC(I),INTV(I),NSTART(I),NLEN(I)                PARM0640
   14 FORMAT(A1,2I1,I3,I2)                                               PARM0650
      IF(NCODE.EQ.IPP) GO TO 103                                         PARM0660
      WRITE(IOUT,15)                                                     PARM0670
   15 FORMAT(' PRINT/PUNCH CARD MISSING OR OUT OF ORDER')                PARM0680
      STOP                                                               PARM0690
  103 CONTINUE                                                           PARM0700
C        READ SUMMARY TAPE HEADER CARD                                   PARM0710
      READ(IN,16) NCODE,NDESC                                            PARM0720
   16 FORMAT(A1,79A1)                                                    PARM0730
      IF(NCODE.EQ.IFF) GO TO 104                                         PARM0740
      WRITE(IOUT,17)                                                     PARM0750
   17 FORMAT(' SUMMARY TAPE HEADER CARD MISSING')                        PARM0760
      STOP                                                               PARM0770
C        PRINT INFORMATION FROM CONTROL CARDS                            PARM0780
  104 WRITE(IOUT,18) NCOMPS,NPRNTS,ISTUDY                                PARM0790
   18 FORMAT('IND. OF COMPARES=',I2,2X,'NO.OF FIELDS TO PRINT/PUNCH=',   PARM0800
     1 I2,2X,'STUDY CODE=',A1)                                           PARM0810
      WRITE(IOUT,19)                                                     PARM0820
   19 FORMAT('///' COMPARE INFORMATION')                                 PARM0830
      DO 105 I=1,NCOMPS                                                  PARM0840
  105 WRITE(IOUT,20)ITYPE(I),ICODE(I),LOCAT(I),ISTART(I),ILEN(I),        PARM0850
     1 ICOND(I),(IVALU(J,I),J=1,12),KSWT(I)                              PARM0860
   20 FORMAT(' ',A1,2X,I1,2X,I1,2X,I3,2X,I2,2X,     A2,2X,12A1,2X,I1)   PARM0870
      WRITE(IOUT,21)                                                     PARM0880
   21 FORMAT('///' PRINT/PUNCH INFORMATION')                             PARM0890
      DO 106 I=1,NPRNTS                                                  PARM0900
  106 WRITE(IOUT,22) NLOC(I),INTV(I),NSTART(I),NLEN(I)                   PARM0910
   22 FORMAT(' ',I1,2X,I1,2X,I3,2X,I2)                                   PARM0920
      WRITE(IOUT,23) NDESC                                               PARM0930
   23 FORMAT('///' TAPE HEADER INFORMATION'/' ',79A1)                    PARM0940
C        READ SUMMARY TAPE HEADER RECORD AND PRINT                       PARM0950
      READ(ISUMTP,24) NHEDR,IHEDR                                        PARM0960
   24 FORMAT(134A1,A4,150(A6,A4))                                        PARM0970
      WRITE(IOUT,25) NHEDR,IHEDR                                         PARM0980
   25 FORMAT('///' SUMMARY TAPE HEADER RECORD'/' ',67A1/,1X,67A1/,       PARM0990
     1 1X,I4,2X,15(/1X,10(A4,I4,2X)))                                    PARM1000
C    COMPARE WITH HEADER CONTROL CARD                                    PARM1010
      DO 107 I=1,79                                                      PARM1020
      IF(NHEDR(I).EQ.NDESC(I)) GO TO 107                                 PARM1030
      WRITE(IOUT,26)                                                     PARM1040
   26 FORMAT('///' HEADER CARD DESCRIPTION DOES NOT MATCH TAPE RECORD')  PARM1050
      STOP                                                               PARM1060
  107 CONTINUE                                                           PARM1070
C        CONVERT VALUES FROM COMPARE CARDS IF NECESSARY                  PARM1080
C                                                                       PARM1090
```

TABLE H-I - (Continued)

```
      DO 108 I=1,NCOMPS                                                 PARM1090
      IF(ITYPE(I).EQ.IAI) GO TO 108                                     PARM1100
      IF(ITYPE(I).EQ.IFF) GO TO 109                                     PARM1110
C        CONVERT TO INTEGER FORMAT                                      PARM1120
      CALL ALIGN(IVALU(1,I),12,ITEMP1)                                  PARM1130
      CALL BCNV(ITEMP1,ISTO,209,6)                                      PARM1140
      ICOMP(I)=ISTO                                                     PARM1150
      GO TO 108                                                         PARM1160
C        CONVERT TO FLOATING POINT                                      PARM1170
  109 CALL ALIGN(IVALU(1,I),12,ITEMP2)                                  PARM1180
      CALL BCNV(ITEMP2,STORE,202,10,0)                                  PARM1190
      COMP(I)=STORE                                                     PARM1200
  108 CONTINUE                                                          PARM1210
C        SET UP INDECES FOR READING SUMMARY TAPE                        PARM1220
      IDEX=IHEDR(1)                                                     PARM1230
      ICOUNT=1                                                          PARM1240
      DO 130 I=1,NCOMPS                                                 PARM1250
      IF(ICOND(I).EQ.IEQ) ISW(I)=1                                      PARM1260
      IF (ICOND(I).EQ.ILE) ISW(I)=2                                     PARM1270
      IF (ICOND(I).EQ.IGE) ISW(I)=3                                     PARM1280
      IF (ICOND(I).EQ.ILT) ISW(I)=4                                     PARM1290
      IF (ICOND(I).EQ.IGT) ISW(I)=5                                     PARM1300
  130 CONTINUE                                                          PARM1310
C                                                                       PARM1320
C        BEGIN READING SUMMARY TAPE                                     PARM1330
C        READ VOYAGE RECORD                                             PARM1340
      INTCNT = 0
      NCOMPS = NCOMPS - IKK                                             PARM0381
      IF(ICARD.EQ.1) GO TO 2                                            PARM0382
      GO TO 4                                                           PARM0383
    2 LK= NCOMPS+NPRNTS                                                 PARM0384
      IK= 12*NCOMPS                                                     PARM0385
      JK =12*NPRNTS +48                                                 PARM0386
      IF (LK.LE.51) GO TO 4                                             PARM0387
      WRITE(IOUT,3)                                                     PARM0388
    3 FORMAT(' NO OF COMPS+ NO OF NPRNTS GREATER THAN 51')              PARM0389
      STOP                                                              PARM1350
    4 DO 112 I=1,IDEX                                                   PARM1360
      READ(ISUMTP,27) IVOYGE                                            PARM1370
   27 FORMAT(200A1,56A1)                                                PARM1380
      LDEX=I+2                                                          PARM1390
      KDEX=IHEDR(LDEX+1)                                                PARM1400
      NONE=0                                                            PARM1410
      READ(ISUMTP,28) INTERV,ISUM                                       PARM1420
   28 FORMAT(200A1,56A1,6A4)                                            PARM1430
      INTCNT = INTCNT + 1
      DO 113 II=1,KDEX                                                  PARM1440
C        READ NEXT INTERVAL IF THERE IS ONE                             PARM1450
C                                                                       PARM1460
      IF(II.EQ.KDEX) GO TO 114                                          PARM1470
      READ(ISUMTP,28) NINTRV,NSUM                                       PARM1480
      INTCNT = INTCNT + 1
      GO TO 115                                                         PARM1490
  114 NONE=1                                                            PARM1500
  115 KEEP=1
      IF( INTCNT - 3698 ) 389,113,389                                   PARM1510
C        BEGIN COMPARES                                                 PARM1520
  389 DO 116 KK = 1, NCOMPS                                             PARM1530
      LSUM=0                                                            PARM1540
```

TABLE H-I - (Continued)

```
C        DETERMINE WHETHER COMPARE FIELD IS IN VOYAGE RECORD OR          PARM1540
C          INTERVAL RECORD                                               PARM1550
      IZ=1                                                               PARM1560
      KDR=ISTART(KK)                                                     PARM1570
      LDR=ILEN(KK)                                                       PARM1580
      MDR=LDR+KDR-1                                                      PARM1590
      IF(LOCAT(KK1,ED,1) GO TO 117                                       PARM1600
C        GET FIELD FROM INTERVAL RECORD                                  PARM1610
      IF (KDR.GT.256) GO TO 118                                          PARM1620
      DO 119 KZ=KDR,MDR                                                  PARM1630
      ITEMP3(KK,IZ)=INTERV(KZ)                                           PARM1640
  119 IZ=IZ+1                                                            PARM1650
      GO TO 120                                                          PARM1660
C        COMPARE FIELD IS ONE OF SUMMARY NUMBERS                         PARM1670
  118 LSUM=1                                                             PARM1680
      KDR=KDR-256                                                        PARM1690
      GO TO 120                                                          PARM1700
C        GET FIELD FROM VOYAGE I.D. RECORD                               PARM1710
  117 IZ=1                                                               PARM1720
      DO 121 KZ=KDR,MDR                                                  PARM1730
      ITEMP3(KK,IZ)=IVOYGE(KZ)                                           PARM1740
  121 IZ=IZ+1                                                            PARM1750
C        MAKE CONVERSION IF REQUIRED                                     PARM1760
  120 IF(LSUM.EQ.1) GO TO 122                                            PARM1770
      IF(ITYPE(KK).EQ.1A) GO TO 123                                      PARM1780
      IF(ITYPE(KK).EQ.IFF) GO TO 124                                     PARM1790
C        CONVERT TO INTEGER FORMAT AND COMPARE                           PARM1800
      LEFT=8-LDR                                                         PARM1810
      IF(LEFT.EQ.0) GO TO 127                                            PARM1820
      DO 125 LONG=1,LEFT                                                 PARM1830
  125 ITEMP4(LONG)=1B                                                    PARM1840
  127 DO 126 LONG=1,LDR                                                  PARM1850
  126 ITEMP4(LONG+LEFT)=ITEMP3(KK,LONG)                                  PARM1860
      CALL ALIGN(ITEMP4,IZ,ITEMP1)                                       PARM1870
      CALL BCNV(ITEMP1,KCOMP,203,6)                                      PARM1880
      IF (ISWERR.EQ.1) GO TO 152                                         PARM1890
  141 ISW2=0                                                             PARM1900
      ISW1=ISW(KK)                                                       PARM1901
      IF(ISWERR.EQ.1) GO TO 152                                          PARM1910
C        MAKE COMPARES                                                   PARM1920
      IF(KSWT(KK).EQ.1) GO TO 300                                        PARM1930
      GO TO (131,132,133,134,135),ISW1                                   PARM1940
  131 IF(KCOMP.EQ.ICOMP(KK)) ISW2 = 1                                    PARM1950
      GO TO 136                                                          PARM1960
  132 IF(KCOMP.LE.ICOMP(KK)) ISW2 = 1                                    PARM1970
      GO TO 136                                                          PARM1980
  133 IF(KCOMP.GE.ICOMP(KK)) ISW2 = 1                                    PARM1990
      GO TO 136                                                          PARM2000
  134 IF(KCOMP.LT.ICOMP(KK)) ISW2 = 1                                    PARM2010
      GO TO 136                                                          PARM2020
  135 IF(KCOMP.GT.ICOMP(KK)) ISW2 = 1                                    PARM2030
      GO TO 136                                                          PARM2040
  300 IF(KCOMP.LE. ICOMP(KK)) GO TO 301                                  PARM2050
      IF(KCOMP.GE. ICOMP(KK+1)) GO TO 301                                PARM2060
      GO TO 136                                                          PARM2070
  301 ISW2 = 1                                                           PARM2080
      GO TO 136                                                          PARM2090
C        IF ONE OF SUMMARY NOS, BE SURE IS INTEGER COM                   PARM2100
  122 IF(ITYPE(KK).NE.1I) GO TO 140                                      PARM2110
```

TABLE H-I - (Continued)

```
      KCOMP=ISUM(KDR)                                                    PARM2120
      GO TO 141                                                          PARM2130
C        WRITE ERROR MESSAGE                                             PARM2140
  140 WRITE(IOUT,29)                                                     PARM2150
   29 FORMAT(' INTEGER COMPARE REQUIRED FOR SUMMARY NO.')                PARM2160
      STOP                                                               PARM2170
C                                                                        PARM2180
C        MAKE FLOATING POINT CONVERSION AND COMAPRE                      PARM2190
  124 DO 142 LONG=1,10                                                   PARM2200
  142 ITEMP5(LONG)=2B                                                    PARM2210
      LEFT=10-LDR                                                        PARM2220
      DO 143 LONG=1,LDR                                                  PARM2230
  143 ITEMP5(LONG+LEFT)=ITEMP3(KK,LONG)                                  PARM2240
      CALL ALIGN(ITEMP5,20,ITEMP2)                                       PARM2250
      CALL BCNV(ITEMP2,ZCOMP,202,10,0)                                   PARM2260
      ISW2=0                                                             PARM2270
      ISW1=ISW(KK)                                                       PARM2280
      IF(ISWERR.EQ.1) GO TO 152                                          PARM2290
C        MAKE FLOATING POINT COMPARES                                    PARM2300
      GO TO (144,145,146,147,148),ISW1                                   PARM2310
  144 WRITE(IOUT,30)                                                     PARM2320
   30 FORMAT(' EQ IS NOT ALLOWED AS A COMPARE CONDITION FOR AN',         PARM2330
     1' F TYPE NUMBER')                                                  PARM2340
      STOP                                                               PARM2350
  145 IF(ZCOMP.LE.COMP(KK)) ISW2 = 1                                     PARM2360
      GO TO 136                                                          PARM2370
  146 IF(ZCOMP.GE.COMP(KK)) ISW2 = 1                                     PARM2380
      GO TO 136                                                          PARM2390
  147 IF(ZCOMP.LT.COMP(KK)) ISW2 = 1                                     PARM2400
      GO TO 136                                                          PARM2410
  148 IF(ZCOMP.GT.COMP(KK)) ISW2 = 1                                     PARM2420
      GO TO 136                                                          PARM2430
C        MAKE CHARACTER COMPARE                                          PARM2440
  123 ISW1 = ISW(KK)                                                     PARM2450
      IF(ISW1.EQ.1) GO TO 149                                            PARM2460
      WRITE(IOUT,31)                                                     PARM2470
   31 FORMAT(' ONLY EQ IS ALLOWED AS A COMPARE CONDITION FOR A C',       PARM2480
     1' TYPE COMPARE')                                                   PARM2490
      STOP                                                               PARM2500
  149 ISW2=0                                                             PARM2510
      DO 150 LL=1,LDR                                                    PARM2520
      IF(IVALUE(LL+KKT).NE.ITEMP3(KK,LL)) GO TO 136                      PARM2530
  150 CONTINUE                                                           PARM2540
      ISW2=1                                                             PARM2550
C        DETERMINE WHETHER TO KEEP FOR A HIT OR A MISS                   PARM2560
  136 IF(ICODE(KK).EQ.0) GO TO 151                                       PARM2570
C        IF A HIT , THROW OUT, GO TO NEXT INTERVAL                       PARM2580
      IF(ISW2.EQ.1) GO TO 152                                            PARM2590
C        IF NOT A HIT , CONTINUE TO COMPARE                              PARM2600
      GO TO 116                                                          PARM2610
C        IF NOT A HIT , THROW OUT                                        PARM2620
  151 IF(ISW2.EQ.0) GO TO 152                                            PARM2630
C        IF A HIT CONTINUE TO COMAPRE                                    PARM2640
  116 CONTINUE                                                           PARM2650
C                                                                        PARM2660
C        IF INTERVAL HAS MET ALL CRITERIA , PREPARE OUTPUT RECORD        PARM2670
C                                                                        PARM2680
C                                                                        PARM2690
C        PREPARE IDENTIFICATION FIELDS                                   PARM2700
```

TABLE H-I - (Continued)

```
      DO 153 LL=1,19                                                    PARM2620
  153 IDENTIL(I)=INTERV(LL)                                             PARM2630
C          PREPARE COMPARE FIELDS FOR PRINTING                          PARM2640
      DO 174 LL=1,20                                                    PARM2650
  174 LPRINT(LL)=KBLANK                                                 PARM2660
      DO 154 LL=1,180                                                   PARM2670
  154 IPRINT(LL)=KBLANK                                                 PARM2680
      IZ=0                                                              PARM2690
      DO 155 KK=1,NCOMPS                                                PARM2700
      KDR=ISTART(KK)                                                    PARM2710
C          DETERMINE IF IT IS A SUMMARY NUMBER,IF YES,CONVERT           PARM2720
      IF(MDR.LE.256) GO TO 157                                          PARM2730
      KDR=MDR-256                                                       PARM2740
      CALL BCNVI(ISUM(KDR),ITEMP8,103,6)                                PARM2750
      CALL BCNVI(ISUM(KDR),ITEMP9,103,7)                                PARM2755
      IF(ISWERR.EQ.1) GO TO 152                                         PARM2756
C          PUT PRINT FIELD IN A FORMAT                                  PARM2760
      ITEMP3(KK,1) = ITEMP8(1)                                          PARM2770
      ITEMP3(KK,2) = ITEMP8(2)                                          PARM2780
      ITEMP3(KK,3) = ITEMP8(3)                                          PARM2790
      ITEMP3(KK,4) = ITEMP9(3)                                          PARM2800
      ITEMP3(KK,5) = ITEMP8(4)                                          PARM2810
      ITEMP3(KK,6) = ITEMP9(4)                                          PARM2820
      LDR=6                                                             PARM2830
      LEFT=6                                                            PARM2840
      GO TO 158                                                         PARM2850
  157 LDR=ILEN(KK)                                                      PARM2860
      LEFT=IZ-ILEN(KK)                                                  PARM2870
  158 IZZ=IZ+LEFT                                                       PARM2880
      DO 156 LK=1,LDR                                                   PARM2890
  156 IPRINT(IZZ+LK)=ITEMP3(KK,LK)                                      PARM2900
  155 IZ=IZ+12                                                          PARM2910
C                                                                       PARM2920
C          PREPARE PRINT FIELDS FOR PRINTING                            PARM2930
C                                                                       PARM2940
      IZ=48                                                             PARM2950
      LARGE=0                                                           PARM2960
      DO 160 KK=1,NPRNTS                                                PARM2970
      KDR=ASTART(KK)                                                    PARM2980
      LDR=ALEN(KK)                                                      PARM2990
      MDR=LDR+KDR-1                                                     PARM3000
      LEFT=12-LDR                                                       PARM3010
C          DETERMINE WHETHER PRINT/PUNCH FIELD IS IN VOYAGE             PARM3020
C          RECORD OR INTERVAL RECORD                                    PARM3030
      IF(NLOC(KK).EQ.1) GO TO 161                                       PARM3040
C          GET FIELD FROM INTERVAL,THIS OR NEXT                         PARM3050
      IF(INTV(KK).EQ.1) GO TO 162                                       PARM3060
C          GET FIELD FROM THIS INTERVAL                                 PARM3070
      ITF=1                                                             PARM3080
      IF(KDR.GT.256) GO TO 164                                          PARM3090
      DO 163 KL=KDR,MDR                                                 PARM3100
      ITEMPA(ITF)=INTERV(KL)                                            PARM3110
  163 ITF=ITF+1                                                         PARM3120
      GO TO 180                                                         PARM3130
C     GET SUM FIELD AND CONVERT TO A1 FORMAT                            PARM3140
  164 KDR=KDR-256                                                       PARM3150
      MSUM=ISUM(KDR)                                                    PARM3160
  165 CALL BCNVI(MSUM,ITEMP8,103,6)                                     PARM3170
      CALL BCNVI(MSUM,ITEMP9,103,7)                                     PARM3180
```

TABLE H-I - (Continued)

```
      IF(ISWERR.EQ.1) GO TO 152                                         PARM3181
      ITEMPA(1)=ITEMP8(1)                                               PARM3190
      ITEMPA(2)=ITEMP8(2)                                               PARM3200
      ITEMPA(3)=ITEMP8(3)                                               PARM3210
      ITEMPA(4)=ITEMP9(3)                                               PARM3220
      ITEMPA(5)=ITEMP8(3)                                               PARM3230
      ITEMPA(6)=ITEMP9(4)                                               PARM3240
      LDR=6                                                             PARM3250
      LEFT=6                                                            PARM3260
      GO TO 180                                                         PARM3270
C          GET FIELD FROM NEXT INTERVAL IF IT EXISTS                    PARM3280
C                                                                       PARM3290
  162 IF (NONE.NE.1) GO TO 166                                          PARM3300
      ITEMPA(1)=KBLANK                                                  PARM3310
      LEFT=11                                                           PARM3320
      LDR=1                                                             PARM3330
      GO TO 180                                                         PARM3340
C          GET FIELD FROM NEXT INTERVAL                                 PARM3350
  166 IF(KDR.LE.256) GO TO 167                                          PARM3360
      KDR=KDR-256                                                       PARM3370
      MSUM=ISUM(KDR)                                                    PARM3380
      GO TO 165                                                         PARM3390
  167 ITF=1                                                             PARM3400
      DO 168 KL=KDR,MDR                                                 PARM3410
      ITEMPA(ITF)=INTRV(KL)                                             PARM3420
  168 ITF=ITF+1                                                         PARM3430
      GO TO 180                                                         PARM3440
C          GET FIELD FROM VOYAGE RECORD                                 PARM3450
  161 ITF=1                                                             PARM3460
      DO 169 KL=KDR,MDR                                                 PARM3470
      ITEMPA(ITF)=VOYGE(KL)                                             PARM3480
  169 ITF=ITF+1                                                         PARM3490
C          PUT FIELD IN PROPER PRINTING POSITION                        PARM3500
C                                                                       PARM3510
  180 IF(LDR.LE.12) GO TO 170                                           PARM3520
C          USE LARGE PRINT FIELD                                        PARM3530
      IF(KK.EQ.NPRNTS) GO TO 171                                        PARM3540
      WRITE(IOUT,32)                                                    PARM3550
   32 FORMAT(' PRINT FIELD LARGER THAN 12 MUST BE LAST PRINT',          PARM3560
     1'/PUNCH CARD ENTERED')                                            PARM3570
      STOP                                                              PARM3580
  170 IF(KK.NE.1) GO TO 192                                             PARM3590
  171 LEFT=20-LDR                                                       PARM3600
      LARGE=1                                                           PARM3610
      DO 172 KL=1,LDR                                                   PARM3620
  172 LPRINT(LEFT+KL)=ITEMPA(KL)                                        PARM3630
      GO TO 160                                                         PARM3640
  192 IZZ=IZ+LEFT                                                       PARM3650
      DO 173 KL=1,LDR                                                   PARM3660
  173 IPRINT(IZZ+KL)=ITEMPA(KL)                                         PARM3670
  160 IZ=IZ+12                                                          PARM3680
C                                                                       PARM3690
C          PRINT RECORD AND WRITE OUT ON TAPE                           PARM3700
      LPRNTS=NPRNTS                                                     PARM3710
      IF(ISWERR.EQ.1) GO TO 152                                         PARM3711
      IF(ICARD.EQ.1) GO TO 5                                            PARM3711
      IF(LARGE.EQ.1) LPRNTS=LPRNTS-1                                    PARM3720
      IF(LPRNTS.LE.1) GO TO 183                                         PARM3730
      IF(LPRNTS.LE.6) GO TO 184                                         PARM3740
```

TABLE H-I - (Continued)

```
      GO TO 185
C           PRINT/PUNCH FIELD REQUESTED                               PARM3750
  183 IF(LARGE.EQ.1) GO TO 196                                        PARM3760
      WRITE(IPNCH,33) ISTUDY,I1,IDENT,(IPRINT(LL),LL=1,60)            PARM3770
   33 FORMAT(80A1)                                                    PARM3780
      GO TO 187                                                       PARM3790
  196 WRITE(IPNCH,25)ISTUDY,I1,IDENT,(IPRINT(LL),LL=1,60)             PARM3800
     1,ISTUDY,I4,IDENT,LPRINT,IFILL                                   PARM3810
      GO TO 187                                                       PARM3820
C         2 - 5 PRINT/PUNCH FIELDS REQUESTED                          PARM3830
    5 WRITE(IPUN   ,1) ISTUDY,I1,IDENT,(IPRINT(LL),LL=1,IK),          PARM3840
     1 (IPRINT(LL),LL=49,JK)                                          PARM3877
    1 FORMAT(80A1)                                                    PARM3878
  184 IF(LARGE.EQ.1) GO TO  188                                       PARM3879
      WRITE(IPNCH,35) ISTUDY,I1,IDENT,(IPRINT(LL),LL=1,60),           PARM3850
     1                ISTUDY,I2,IDENT,(IPRINT(LL),LL=61,120)          PARM3860
   35 FORMAT(160A1)                                                   PARM3870
      GO TO 157                                                       PARM3875
  158 WRITE(IPNCH,37) ISTUDY,I1,IDENT,(IPRINT(LL),LL=1,60),           PARM3876
     1 ISTUDY,I2,IDENT,(IPRINT(LL),LL=61,120),                        PARM3890
     1 ISTUDY,I4,IDENT,LPRINT,IFILL                                   PARM3900
   37 FORMAT(240A1)                                                   PARM3910
      GO TO 187                                                       PARM3920
C          7-11 PRINT /PUNCH FIELDS REQUESTED                         PARM3930
  185 IF(LARGE.EQ.1) GO TO 189                                        PARM3940
      WRITE(IPNCH,37) ISTUDY,I1,IDENT,(IPRINT(LL),LL=1,60)            PARM3950
     1,ISTUDY,I2,IDENT,(IPRINT(LL),LL=61,120),                        PARM3960
     2 ISTUDY,I3,IDENT,(IPRINT(LL),LL=121,180)                        PARM3970
      GO TO 187                                                       PARM3980
  189 WRITE(IPUN   ,1) ISTUDY,I1,IDENT,(IPRINT(LL),LL=001,060)        PARM3990
      WRITE(IPUN   ,1) ISTUDY,I2,IDENT,(IPRINT(LL),LL=061,120)
      WRITE(IPUN   ,1) ISTUDY,I3,IDENT,(IPRINT(LL),LL=121,180)
      WRITE(IPUN   ,1) ISTUDY,I4,IDENT,LPRINT,IFILL
      WRITE(IPNCH,39) ISTUDY,I1,IDENT,(IPRINT(LL),LL=1,60)
     1,ISTUDY,I2,IDENT,(IPRINT(LL),LL=61,120)                         PARM4000
     2,ISTUDY,I3,IDENT,(IPRINT(LL),LL=121,180),                       PARM4010
     3 ISTUDY,I4,IDENT,LPRINT,IFILL                                   PARM4020
   38 FORMAT(250A1,70A1)                                              PARM4030
C                                                                     PARM4040
C         PRINT                                                       PARM4050
C                                                                     PARM4060
  187 WRITE(IOUT,34)                                                  PARM4070
   34 FORMAT(//)                                                      PARM4080
      WRITE(IOUT,39) ISTUDY,I1,IDENT,(IPRINT(LL),LL=1,60)             PARM4090
   39 FORMAT(' ',A1,2X,A1,2X,18A1,5(2X,12A1))                         PARM4100
      IF(LPRNTS.LE.1) GO TO 190                                       PARM4110
      WRITE(IOUT,39) ISTUDY,I2,IDENT,(IPRINT(LL),LL=61,120)           PARM4120
      IF(LPRNTS.LE.6) GO TO 190                                       PARM4130
      WRITE(IOUT,39) ISTUDY,I3,IDENT,(IPRINT(LL),LL=121,180)          PARM4140
  190 IF(LARGE.NE.1) GO TO 152                                        PARM4150
      WRITE(IOUT,40) ISTUDY,I4,IDENT,LPRINT                           PARM4160
   40 FORMAT(' ',A1,2X,A1,2X,18A1,2X,20A1)                            PARM4170
C                                                                     PARM4180
C --------MOVE NEXT INTERVAL TO THIS INTERVAL-----                    PARM4190
  152 DO 181 KT=1,256                                                 PARM4200
  181 INTERV(KT)=NINTRV(KT)                                           PARM4210
      DO 182 KT=1,6                                                   PARM4220
  182 ISUM(KT)=NSUM(KT)                                               PARM4230
      ISWERR = 0                                                      PARM4240
                                                                      PARM0421
```

TABLE H-I - (Concluded)

```
  113 CONTINUE
C          READ VOYAGE SUMMARY RECORD                                 PARM4250
C                                                                     PARM4260
      READ(ISUMTP,2*) IDUM                                            PARM4270
  112 CONTINUE                                                        PARM4285
      END FILE IPNCH                                                  PARM4290
      WRITE(IOUT,41)                                                  PARM4300
   41 FORMAT(' JOB SUCCESSFULLY COMPLETED')                           PARM4310
      STOP                                                            PARM4320
      END                                                             PARM4330
      SUBROUTINE ERROR (IADDR,IERR,LOG)                               PARM4340
      INTEGER*4 IERR,IADDR                                            ERRS0010
      LOGICAL*1 LOG,LOGA                                              ERRS0020
      COMMON ISWERR                                                   ERRS0030
      DATA LOGA/'0'/                                                  ERRS0040
      ISWERR = 1                                                      ERRS0050
      IADDR = 1                                                       ERRS0060
      LOG = LOGA                                                      ERRS0070
      RETURN                                                          ERRS0080
      END                                                             ERRS0090
/*                                                                    ERRS0100
//LKED.SYSLIN DD
//         DD  DSNAME=*.ALNTST.ASM.SYSGO,DISP=(OLD,DELETE),
//             VOLUME=REF=*.ALNTST.ASM.SYSGO,
//             DCB=(,RECFM=FB,LRECL=80,BLKSIZE=3200)
//         DD *
 ENTRY MAIN
 ESD              IHCOSP04       BCNV          ADCON#
 ESD              ERRMON                                              IHCG0001
 TXT                         *   *    K 7                             IHCG0002
 TXT                                                                  IHCG0003
 TXT                 0 *                                   -         *IHCG0004
 TXT              *  2   M   6   -         *               *          IHCG0005
 TXT                0 0                                  *   0    *   IHCG0006
 TXT                              K        *              *           IHCG0007
 TXT    *                  *           *   0  K      0 6*             IHCG0008
 TXT                K               0       0     0           0    K  IHCG0009
 TXT               NIHC302I IBCOM - PARAM ERROR IN CALL TO BCNV  0  4 4IHCG0010
 TXT    8                                                             IHCG0011
 TXT                                                                  IHCG0012
 TXT                                          0123456789ABCDEF        IHCG0013
 RLD                                   M  K                           IHCG0014
 END                                                                  IHCG0015
/*                                                                    IHCG0016
//GO.FT10F001 DD UNIT=TAPE9,LABEL=(,NL),VOLUME=SER=RESLT,
//             DISP=(NEW,PASS),DCB=(RECFM=FB,LRECL=320,BLKSIZE=960)
//GO.FT09F001 DD UNIT=TAPE9,LABEL=(,NL),VOLUME=SER=HAFB,
//             DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=4000)
//GO.FT07F001 DD SYSOUT=B
//GO.FT05F001 DD *
C   3    3    C    1
SA0106805E001-05          0
S10102002LE             031
S10102002GE             100
P0005702
P0006502
P1006A08
FINAL SUMMARY TAPE (FULL BRIDGE), CONTRACT NSSC-N00024-69-C-5161
/*
```

APPENDIX I

RELATIVE WIND DIRECTION CORRECTION SUBROUTINE

INTRODUCTION

The original data entered in the logbooks permitted several options to report wind direction. The Logbook Preprocessor program (Appendix A) was written to accept the options and convert all to a single format namely, Relative Wind Direction (in degrees port or starboard). However, the programming handled incorrectly a small amount of data with the result that these Wind Directions were in error by 180°. A correct subroutine was written (RELWND) which could be incorporated in any of the existing programs to correct these data which were in error.

The RELWND subroutine, as listed in Table I-1, makes any necessary adjustment to the Relative Wind Direction data. Whether the existing value requires correction or not depends on several other given values (i.e., ship course, ship speed, true wind speed and relative wind speed). For those cases in which a correction is required, the corrected value of Relative Wind Direction is obtained by subtracting the original value from 180° and changing the port "P" designation to starboard "S" or vice versa. The last two arguments of the subroutine contain the correct Relative Wind Direction whether a correction was made or not.

The flow chart for the subroutine is shown in Figure I-1.

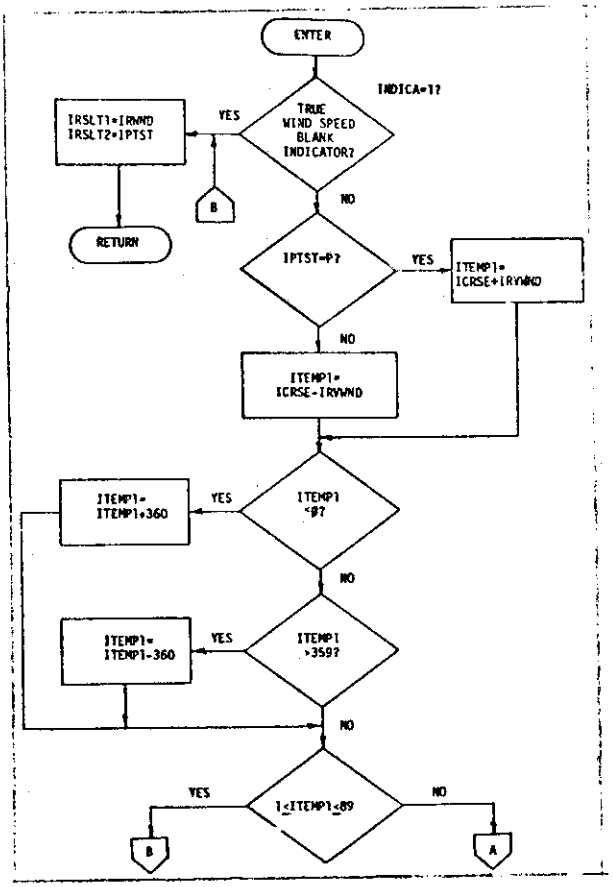


FIG. I-1. FLOW CHART FOR RELWND SUBROUTINE

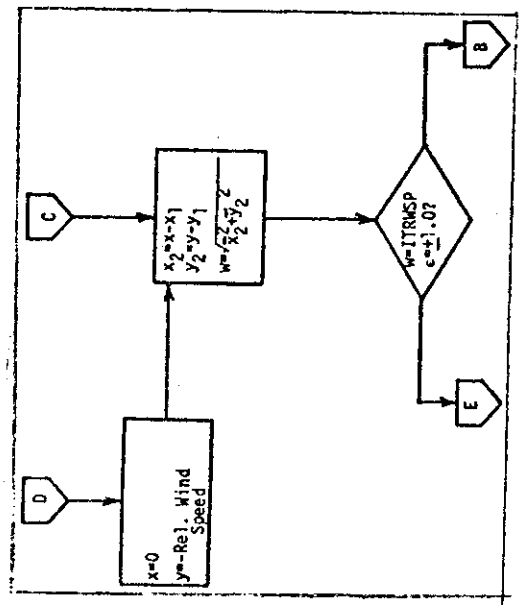


FIG. I-1. (Concluded)

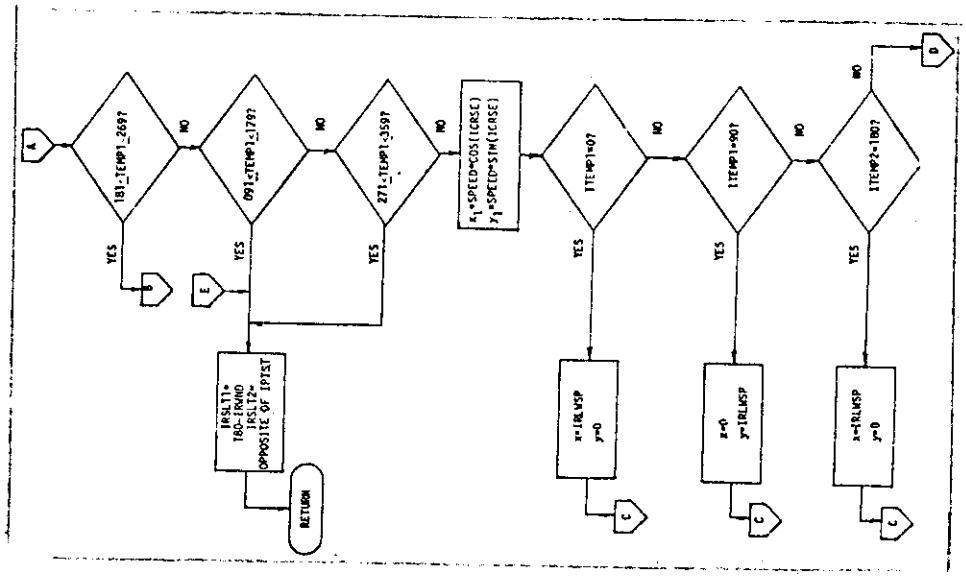


FIG. I-1. (Continued)

TABLE I-I - PROGRAM LISTING FOR RELATIVE WIND CORRECTION SUBROUTINE

```
// JOB
      SUBROUTINE RELWND(ICRSE,IRVWND,IPTST,SPEED,ITRWSP,IRLWSP,INDICA,
     1                  IRSLT1,IRSLT2)
C
C     SUBROUTINE TO CORRECT RELATIVE WIND DIRECTION
C                ICRSE     COURSE
C                IRVWND    RELATIVE WIND DIRECTION
C                IPTST     DIRECTION INDICATOR,P,S,OR BLANK
C                SPEED     SHIP SPEED
C                ITRWSP    TRUE WIND SPEED
C                IRLWSP    RELATIVE WIND SPEED
C                INDICA    INDICATES A BLANK TRUE WIND SPEED
C                IRSLT1    CORRECTED RELATIVE WIND SPEED
C                IRSLT2    CORRECTED DIRECTION INDICATOR
C
C
      INTEGER*2 KP,KS,IRSLT2,IPTST
      COMMON ISW1
      DATA KP,KS/'P','S'/
C        CHECK FOR BLANK TRUE WIND SPEED
      IF(INDICA.EQ.0) GO TO 101
  102 IRSLT1=IRVWND
      IRSLT2=IPTST
      RETURN
  101 IF(IPTST.EQ.KP) GO TO 103
      ITEMP1=ICRSE+IRVWND
      GO TO 104
  103 ITEMP1=ICRSE-IRVWND
  104 IF(ITEMP1.LT.0) GO TO 105
      IF(ITEMP1.GT.359) GO TO 106
      GO TO 107
  105 ITEMP1=ITEMP1+360
      GO TO 107
  106 ITEMP1=ITEMP1-360
  107 IF((ITEMP1.GE.1).AND.(ITEMP1.LE.89)) GO TO 102
      IF((ITEMP1.GE.181).AND.(ITEMP1.LE.269)) GO TO 102
      IF((ITEMP1.GE.91).AND.(ITEMP1.LE.179)) GO TO 108
      IF((ITEMP1.GE.271).AND.(ITEMP1.LE.359)) GO TO 108
      GO TO 109
  108 IRSLT1=180-IRVWND
      IF(IPTST.EQ.KP) GO TO 110
      IF(IPTST.EQ.KS) GO TO 111
      IRSLT2=IPTST
      RETURN
  110 IRSLT2=KS
      RETURN
  111 IRSLT2=KP
      RETURN
  109 CRSE=ICRSE*0.01745
      X1=SPEED*COS(CRSE)
      Y1=SPEED*SIN(CRSE)
      IF(ITEMP1.EQ.0) GO TO 112
      IF(ITEMP1.EQ.90) GO TO 113
      IF(ITEMP1.EQ.180) GO TO 114
      X=0
      Y=-IRLWSP
      GO TO 115
  112 X=IRLWSP
      Y=0
```

TABLE I-I - (Concluded)

```
      GO TO 115
  113 X=0
      Y=IRLWSP
      GO TO 115
  114 X=-IRLWSP
      Y=0
  115 X2=X-X1
      Y2=Y-Y1
      W=SQRT(X2**2+Y2**2)
      EPSI=ABS(W-ITRWSP)
      IF(EPSI.LE.1.0) GO TO 102
      GO TO 108
      END
      SUBROUTINE ERROR (IADDR,IERR,LOG)
      INTEGER*4 IERR,IADDR
      LOGICAL*1 LOG,LOGA
      COMMON  ISW1
      DATA LOGA/'D'/
      ISW1=1
      IADDR = 1
       LOG=LOGA
      RETURN
      END
/*
//LKED.SYSLIN DD
//        DD *
 ENTRY MAIN
 ESD          IHCGSP04       BCNV          ADCON*
 ESD          ERRMON
 TXT                 *    *    K /
 TXT                                                         -      *
 TXT             0 *                                         *
 TXT          -  2  M  6  -           *           *    0      *
 TXT            0 0           K        *           *      '
 TXT      -                      *     D  K      0 6.     '
 TXT   *              *                                    0  K
 TXT            K             0     0    0            0   4  *
 TXT            NIHC302I IBCOM - PARAM ERROR IN CALL TO BCNV
 TXT   6
 TXT                                           0123456789ABCDEF
 TXT                               M  K
 RLD
 END
/*
//GO.FT10F001 DD UNIT=TAPE9,LABEL=(,NL),VOLUME=SER=FULL,
//              DISP=(OLD,KEEP),DCB=(RECFM=U,BLKSIZE=4000)
//GO.FT11F001 DD UNIT=TAPE9,DISP=(NEW,PASS),LABEL=(,NL),
//              VOLUME=SER=CHBOY,DCB=(RECFM=U,BLKSIZE=4000,DEN=2)
//GO.FT05F001 DD *
/*
```

Unclassified

Security Classification

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)	2a. REPORT SECURITY CLASSIFICATION
TELEDYNE MATERIALS RESEARCH 303 Bear Hill Road Waltham, MA 02154	Unclassified 2b. GROUP

3. REPORT TITLE

COMPUTER PROGRAMS FOR THE DIGITIZING AND USING OF LIBRARY TAPES OF SHIP STRESS AND ENVIRONMENT DATA

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Final Report

5. AUTHOR(S) (First name, middle initial, last name)

Aldie E. Johnson, Jr.
James A. Flaherty
Isaac J. Walters

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
February 1973	106	1

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
N00024-69-C-5161 b. PROJECT NO. c. Ship Structure Committee Research Project SR-187-Part II d.	E-1149(d) 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) SSC-237

10. DISTRIBUTION STATEMENT

Distribution of this document is unlimited

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Ship Structure Committee Naval Ship Engineering Center

13. ABSTRACT

Processor programs are given to convert original analogue stress data and hand-entry ship, sea and weather environmental information to a final digital format. Included is a program to provide a capability to retrieve portions of the data for subsequent statistical analysis.

DD FORM 1473 REPLACES DD FORM 1473, 1 JAN 64, WHICH IS OBSOLETE FOR ARMY USE.

Unclassified

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
PROCESSOR PROGRAM						
LIBRARY TAPES						
DIGITIZING						

SHIP RESEARCH COMMITTEE
Maritime Transportation Research Board
National Academy of Sciences-National Research Council

The Ship Research Committee has technical cognizance of the inter agency Ship Structure Committee's research program:

PROF. R. A. YAGLE, Chairman, *Prof. of Naval Architecture, University of Michigan*

DR. H. N. ABRAMSON, *Tech. Vice President, Dept. of Mech. Sciences, Southwest Res. Inst.*

PROF. R. W. CLOUGH, *Prof. of Civil Engineering, University of California*

MR. E. L. CRISCUOLO, *Senior NDT Specialist, Naval Ordnance Laboratory*

PROF. J. E. GOLDBERG, *Prof. of Civil Engineering, Purdue University*

PROF. W. J. HALL, *Prof. of Civil Engineering, University of Illinois*

DR. S. R. HELLER, JR., *Chairman, Civil & Mech. Eng. Dept., The Catholic Univ. of Americ*

MR. G. E. KAMPSCHAEFER, JR., *Manager, Technical Services, ARMCO Steel Corporation*

MR. R. C. STRASSER, *Director of Research, Newport News Shipbuilding & Dry Dock Company*

MR. H. S. TOWNSEND, *Vice President, U. S. Salvage Association, Inc.*

DR. S. YUKAWA, *Consulting Engineer, General Electric Company*

MR. R. W. RUMKE, *Executive Secretary, Ship Research Committee*

Advisory Group I, "Ship Response and Load Criteria", prepared the project prospectus and evaluated the proposals for this project:

DR. H. N. ABRAMSON, CHAIRMAN, *Tech. Vice President, Dept. of Mech. Sciences, Southwest Research Institute*

DR. C. CHRYSSOSTOMIDIS, *Asst. Prof. of Naval Architecture, Mass. Inst. of Technology*

PROF. A. M. FREUDENTHAL, *College of Engineering, George Washington University*

DR. R. GLASFELD, *Naval Architect, General Dynamics Corporation*

MR. R. G. KLINE, *Assoc. Research Consultant, U. S. Steel Corporation*

DR. M. K. OCHI, *Research Scientist, Naval Ship Research & Development Center*

MR. R. C. STRASSER, *Director of Research, Newport News Shipbuilding & Dry Dock Company*

The SR-187 Project Advisory Committee provided the liaison technical guidance and reviewed the project reports with the investigator:

MR. W. H. BUCKLEY, Chairman, *Coordinator of Hydrofoil Struct. Res., Naval R & D Center*

PROF. A. M. FREUDENTHAL, *College of Engineering, George Washington University*

MR. R. G. KLINE, *Assoc. Research Consultant, U. S. Steel Corporation*

SHIP STRUCTURE COMMITTEE PUBLICATIONS

These documents are distributed by the National Technical Information Service, Springfield, Va. 22151. These documents have been announced in the Clearinghouse journal U.S. Government Research & Development Reports (USGRDR) under the indicated AD numbers.

SSC-224, *Feasibility Study of Glass Reinforced Plastic Cargo Ship* by R. J. Scott and J. H. Sommella. 1971. AD 735113.

SSC-225, *Structural Analysis of Longitudinally Framed Ships* by R. Nielsen, P. Y. Chang, and L. C. Deschamps. 1972. AD 752769.

SSC-226, *Tanker Longitudinal Strength Analysis - User's Manual and Computer Program* by R. Nielsen, P. Y. Chang, and L. C. Deschamps. 1972. AD 752770.

SSC-227, *Tanker Transverse Strength Analysis - User's Manual* by R. Nielsen P. Y. Chang, and L. C. Deschamps. 1972. AD 752771.

SSC-228, *Tanker Transverse Strength Analysis - Programmer's Manual* by R. Nielsen, P. Y. Chang, and L. C. Deschamps. 1972. AD 752742.

SSC-229, *Evaluation and Verification of Computer Calculations of Wave-Induced Ship Structural Loads* by P. Kaplan and A. I. Raff. 1972. AD 753220.

SSC-230, *Program SCORES -- Ship Structural Response in Waves* by A. I. Raff. 1972. AD 752468.

SSC-231, *Further Studies of Computer Simulation of Slamming and Other Wave-Induced Vibratory Structural Loadings on Ships in Waves* by P. Kaplan and T. P. Sargent. 1972. AD 752479.

SSC-232, *Study of the Factors which Affect the Adequacy of High-Strength, Low Alloy, Steel Weldments for Cargo Ship Hulls* by E. B. Norris, A. G. Pickett, R. D. Wylie. 1972. AD 752480.

SSC-233, *Correlation of Model and Full Scale Results in Predicting Wave Bending Moment Trends* by D. Hoffman, J. Williamson, E. V. Lewis. 1972. AD 753223.

SSC-234, *Evaluation of Methods for Extrapolation of Ship Bending Stress Data* by D. Hoffman, R. Van Hooff, E. V. Lewis. 1972. AD 753224.

SSC-235, *Effect of Temperature and Strain Upon Ship Steels* by R. L. Rothman and R. E. Monroe. 1973.

SSC-236, *A Method for Digitizing, Preparing and Using Library Tapes of Ship Stress and Environment Data* by A. E. Johnson, Jr., J. A. Flaherty, I. J. Walters. 1973.