UNCLASSIFIED

NAVAL UNDERWATER SYSTEMS CENTER
NEW LONDON LABORATORY
NEW LONDON, CONNECTICUT 06320

Technical Memorandum

ANALOG TO DIGITAL CONVERSION AND VERIFICATION PROGRAMS
FOR A VAX 11/780

Date: 21 November 1986

Prepared by: Patricia Maciejewski
Computer Scientist
Surface Ship Sonar

Distribution Statement A: Approved for Public Release,
Distribution Unlimited.
ABSTRACT

A procedure for digitizing data on a Vax 11/780 computer using VAX FORTRAN programs and VMS system service routines was developed. Brief descriptions and source listings of the programs are enclosed. In addition, several auxiliary programs used for verifying the procedure are also enclosed.

ADMINISTRATIVE INFORMATION

This work was supported under NUSC Project No. B69025, Broadband Bearing-Time Processing, P. D. Herstein, Principal Investigator. Funding was provided under Program Element No. 62711 through Naval Air Development Center, R. Fosko, Program Manager.

ACKNOWLEDGMENT

Appreciation is gratefully extended to Dave Potter, Code 3332, who provided much of the original analog to digital code, answered many questions and reviewed this document.
INTRODUCTION

Digitizing analog tapes is a necessary part of many data analysis projects. A FORTRAN program for digitizing data on a Vax 11/780 computer using VMS system service routines is described as well as several additional verification programs.

HARDWARE

For each experiment the hardware setup will vary slightly. Appendix A shows the hardware our experiment used. This included a Vax 11/780 computer with one LPA11-K, 12 bit, 16 channel analog to digital (A/D) converter in dedicated mode. The digital tape drive was a TU78, 9 track, 6250 bpi drive. (Slower speed tape drives were not able to keep up with our particular application.)

The A/D converter was triggered by an external Schmidt trigger. For this particular application, we sampled the data at 2.5 kHz. If the internal clock is desired, the source code would have to be altered.

A final digitized data sample is a 16 bit word ($I^2$ data type), accurate to 12 bits and ranging from -5 to +5 volts. Real data are scaled by

$$\text{REAL\_DATA=}\frac{\text{FLOAT}(I^2\_\text{DATA})}{409.5} - 5.$$  

PROCESSING PROCEDURE

For the application described, 24 channels of analog data were digitized. Since the computer had one 16 channel A/D, it was necessary to process the data in two passes of 14 channels each. The second pass repeated 4 channels; these were used for alignment. Disk/tape constraints forced each run to be digitized in four parts, an end alignment was necessary. The method used for alignment is described in TM NO. 861118, "RECAT - REDUNDANT CHANNEL ALIGNMENT TECHNIQUE", by W. HAUCK. The procedure for digitizing one tape follows.

A. Log onto the computer.
B. Mount the analog tape and check all hardware settings.
C. Mount the digital tape.
D. ALLOCATE MFA0: (HIGH SPEED TAPE DRIVE)
E. Activate A2D.COM
1. Initialize the magnetic tape
2. Assign the appropriate output data disk
3. Enter the digitizing setup via "ENTERHEAD" program
4. Assign and mount the magnetic tape
5. Alter the user priority to 28 (VERY HIGH!!)
6. Perform the digitizing task via "A2D" program
7. Set the priority back to 4 (NORMAL)
8. Copy the recat files for alignment (both passes and ends) from tape to disk via "SAVRECAT" program
9. Copy the quick check files to disk via "READDT" program
10. Dismount the magtape and deallocate the drive
11. Label the tape

<table>
<thead>
<tr>
<th>TAPEID</th>
<th>RUN</th>
<th>PASS</th>
</tr>
</thead>
<tbody>
<tr>
<td>9 TRK</td>
<td>START</td>
<td>: :</td>
</tr>
<tr>
<td>6250</td>
<td>STOP</td>
<td>: :</td>
</tr>
<tr>
<td>FOR</td>
<td>BUFFERS</td>
<td></td>
</tr>
</tbody>
</table>

12. Run the quick check plot program, "QUPL" to verify the data looks okay.

F. Repeat B-E as needed
G. Logoff when done

This procedure is repeated for one run worth of data (4 parts, 2 passes of each). The "RECAT" program will calculate the alignment values. The data is then downsampled and or filtered and later merged, depending on your desires.

PROGRAM DESCRIPTIONS

A. ENTERHEAD.FOR - The operator enters all identification information when this program is run. That information is stored in a data file for retrieval by the actual digitizing program. This method of entering was selected because the probability of repeating a digitizing session is very high.

Routines called are LISTHEAD.FOR.

B. A2D.FOR - Performs the digitizing task. The identification information entered by ENTERHEAD is written to the first record of each tape. It is followed by records containing the digitized data. The digitized records are 14,000 bytes long consisting of 7000 samples. (500 samples/channel x 14 channels) Within each record the order of the samples is: sample 1, chan 1-14, sample 2, chan 1-14, ... sample 500, chan 1-14. The final record is
followed by two end of file markers to signify end of tape.

The record size chosen was based on the sample rate and the number of channels. Users should establish a reasonable record size for writing to tape.

Routines called are GETHEAD and LISTHEAD.

NOTE: When digitizing, the user must have a real time priority; 28 or 30 seems to be sufficient. Ask the system manager for the ALTPRI privilege.

C. SAVRECAT.FOR - Reads a tape made by A2D and extracts the channels chosen for alignment. The first 500 records and the last 50 records are written to disk. RECAT, as described in TM NO. 861118 uses these files for alignment procedures.

Routines called are LISTHEAD.

D. READT.FOR - Reads a digitized tape and separates all channels into separate files. Each record contains 500 samples of A/D channels 1-14.

Routines called are LISTHEAD.

E. QUPL.FOR - Plots the first 4 records of digitized tape, 1 channel/plot and 4 channels/page.

Routines called are PLCHAN and plotting routines from the library QPLOT or GRAFX.

F. DOWNSAMP.FOR - Reads a digitized tape, downsamples it, changes the record sizes if desired and saves it on disk. Alignment procedures, appending, throwing away and saving data are completed. The disk file consists of one header identification record followed by records containing 1024 words from A/D chan 1, 1024 words from A/D chan 2, ..., 1024 words from A/D chan 14.

Routines called are LISTHEAD.

G. MERGEMT.FOR - Reads the disk files created by "DOWNSAMP" for one run, passes 1 and 2 and merges each record. The final tape contains all 24 analog channels, redundant channels are discarded. The final tape format is:
One header identification record 14000 bytes long
Records (49152 bytes) of 1024 samples from channel 1
  1024 samples from channel 2
  
  1024 samples from channel 24

Routines called are LISTHEAD.

H. READMT.FOR - Reads a merged tape and writes the data out to
disk, one channel per file. Records consist of 1024 samples
each.

Routines called are ASSMT, RBLOCKMT, and LISTHEAD.

REFERENCES

[1] LPA11-K FORTRAN USER'S REFERENCE GUIDE, prepared by Digital

[2] LPA11-K LABORATORY PERIPHERAL ACCELERATOR USER'S GUIDE,
    prepared by Digital Equipment Corporation, April 1978.

    Digital Equipment Corporation, July 1985, Version 4.2

APPENDIX TABLE

APPENDIX A - HARDWARE SETUP
APPENDIX B - LISTING OF A2D.COM
APPENDIX C - LISTINGS OF ENTERHEAD.FOR AND LISTHEAD.FOR
APPENDIX D - LISTINGS OF A2D.FOR, GETHEAD.FOR AND LISTHEAD.FOR
APPENDIX E - LISTINGS OF SAVRECAT.FOR AND LISTHEAD.FOR
APPENDIX F - LISTINGS OF READDT.FOR AND LISTHEAD.FOR
APPENDIX G - LISTINGS OF QUPL.COM, QUPL.FOR AND PLCHAN.FOR
APPENDIX H - LISTINGS OF DOWNSAMP.FOR AND LISTHEAD.FOR
APPENDIX I - LISTINGS OF MERGEMT.FOR AND LISTHEAD.FOR
APPENDIX J - LISTINGS OF READMT.FOR, ASSMT.FOR, RBLOCKMT.FOR
    AND LISTHEAD.FOR
APPENDIX K - SAMPLE DIGITIZING SESSION
APPENDIX L - COMMON ERRORS AND POSSIBLE CAUSES
APPENDIX A

HARDWARE SETUP

A-1/A-2
Reverse Blank
APPENDIX B

LISTING OF A2D.COM
$! A2D.COM COMMAND STREAM FOR DIGITIZING 14 A/D CHANNELS.

$! SET VERIFY
$ IF P1 .EQS. "" THEN -
$ INQUIRE P1 "DO YOU WANT TO INITIALIZE THE TAPE? (Y/N)"
$ IF F$STRING(P1) .EQS. "N" THEN GOTO SETDISK $!
$ IF P2 .EQS. "" THEN
$ INQUIRE P2 "ENTER TAPE LABEL? (R__P__)"
$ ALLOCATE MFA0:
$ INITIALIZE/DENSITY=6250 MFA0: 'P2'
$!
$ SETDISK:
$ IF P3 .EQS. "" THEN INQUIRE P3 "IS DATA GOING TO DISK 0 OR 1?"
$ IF F$STRING(P3) .EQS. "0" THEN SET DEF DRB0:[A2D.DATA]
$ IF F$STRING(P3) .EQS. "1" THEN SET DEF DRB1:[A2D.DATA]
$!
$ SETUP:
$ IF P4 .EQS. "" THEN -
$ INQUIRE P4 "HAVE YOU ENTERED THE DIGITIZING SETUP? (Y/N)"
$ IF F$STRING(P4) .EQS. "Y" THEN GOTO MTAPE
$ ASSIGN/USER SYSS$COMMAND SYSS$INPUT
$ RUN DRB0:[A2D.DIGIT]ENTERHEAD !ENTER HEADER INFORMATION
$!
$ MTAPE:
$ IF P5 .EQS. "" THEN -
$ INQUIRE P5 "DID YOU ASSIGN AND MOUNT THE MAGTAPE? (Y/N)"
$ IF F$STRING(P5) .EQS. "Y" THEN GOTO DIGIT
$ ASSIGN MFA0: MAG TAPE
$ MOUNT/FOREN/DENS=6250 MFA0:
$!
$ DIGIT:
$ SET PROC/PRIV=PSWAPM !SET UP PRIVILEGES/PRIORITY
$ SET PROC/PRIV=ALTPRI
$ SET PROC/PRI=30
$ ASSIGN/USER SYSS$COMMAND SYSS$INPUT
$ RUN DRB0:[A2D.DIGIT]A2D !PERFORM DIGITIZING
$ SET PROC/PRI=4 !SET PRIORITY BACK DOWN
$!
$ IF P6 .EQS. "" THEN -
$ INQUIRE P6 "DO YOU WANT TO COPY THE RECAT FILES TO DISK? (Y/N)"
$ IF F$STRING(P6) .EQS. "N" THEN GOTO QUPL
$ ASSIGN/USER SYSS$COMMAND SYSS$INPUT
$ RUN DRB0:[A2D.DIGIT]SAVRECAT !COPY THE RECAT FILES
$!
$ QUPL:
$ INQUIRE P7 "DO YOU WANT TO COPY THE QUICK CHECK FILES TO DISK? (Y/N)"
$ IF F$STRING(P7) .EQS. "N" THEN GOTO MTDISM
$ SET DEF DRB1:[A2D.QUPL]
$ ASSIGN/USER SYSS$COMMAND SYSS$INPUT
$ RUN DRB0:[A2D.DIGIT]READT
$ WRITE SYSS$OUTPUT "YOU NEED TO RUN DRB0:QUPL.COM TO MAKE PLOTS"
$!
$ MTDISM:
$ IF P8 .EQS. "" THEN -
$ INQUIRE P8 "DO YOU WANT TO DISMOUNT THE TAPE? (Y/N)"
$ IF F$STRING(P8) .EQS. "Y" THEN DISMOUNT MFA0:
$!
TM No. 861214

_DRB0: [MACIE.TM]A2D.COM; 30

$ DONE:
$ WRITE SYS$OUTPUT "DIGITIZING PROCEDURE COMPLETE"
$ EXIT
APPENDIX C

LISTINGS OF ENTERHEAD.FOR LISTHEAD.FOR
HAS THE OPERATOR ENTER THE HEADER INFORMATION
AND SAVE IT IN A DATA FILE.

```c
PARAMETER NCHN=14  !NUMBER OF A/D CHANNELS
PARAMETER NBYTES=NCHN*500*2  !NUMBER OF BYTES PER RECORD
PARAMETER N12=NBYTES/2  !NUMBER OF I*2 WORDS
PARAMETER N14=NBYTES/4  !NUMBER OF I*4 WORDS OR FLOATING PT

CHARACTER*1 ANS
BYTE HEADB(NBYTES),ITITLE(20),FILNAM(11)
INTEGER*2 HEADI2(N12)
INTEGER*4 HEADI4(N14)
REAL HEADFP(N14)
EQUIVALENCE (HEADB(11),ITITLE)
EQUIVALENCE (HEADB,HEADI2,HEADI4,HEADFP)
DATA HEADI4/NI4*0/
```
C

C-----ask the operator for all header info

C

100 CONTINUE

PRINT 1
1 FORMAT(X/X,'ENTER ANALOG TAPE NUMBER: ',$)
READ *, HEADI2(1)
C
PRINT 13
13 FORMAT(X/X,'ENTER RUN IDENTIFICATION NUMBER: ',$)
READ *, HEADI2(28)
C
PRINT 2
2 FORMAT(X/X,'ENTER DATA TAPE PASS NUMBER (1 OR 2)? ',$)
READ *, HEADB(3)
C
PRINT 11
11 FORMAT(X/X,'ENTER NUMBER OF CHANNELS/HYDROPHONES: ',$)
READ *, HEADB(4)
HEADB(4)=14
NCHAN=HEADB(4)
C
PRINT 3
3 FORMAT(X/X,'ENTER DIGITIZING DATE: MONTH ',$)
READ *, HEADB(5)
PRINT 4
4 FORMAT(X/X,'DAY ',$)
READ *, HEADB(6)
PRINT 5
5 FORMAT(X/X,'YEAR ',$)
READ *, HEADB(7)
C
PRINT 6
6 FORMAT(X/X,'ENTER START TIME CODE: HOUR ',$)
READ *, HEADB(8)
PRINT 7
7 FORMAT(X/X,'MINUTES ',$)
READ *, HEADB(9)
PRINT 8
8 FORMAT(X/X,'SECONDS ',$)
READ *, HEADB(10)
C
DO I=11,30
   HEADB(I)=''
END DO
PRINT 9
9 FORMAT(X/X,'ENTER 20 CHARACTER EVENT DESCRIPTION: ',$)
READ 10, NC,(ITITLE(I),I=1,NC)
10 FORMAT(Q,20A1)
DO I=1,NC
   HEADB(10+I)=ITITLE(I)
END DO
C
DO I=1,NCHAN
   PRINT 12, I
12 FORMAT(X/X,'ENTER ANALOG CHANNEL FOR A/D CHANNEL ',$)
   READ *, HEADB(I+30)
END DO
HEAUDP(15) = 1./2500./4. ! SAMPLE RATE 2.5 KHZ AT 1/4 SPEED

C-----SAVE THE INPUTS IN A DATA FILE

ENCOD(10,14,FILNAM) HEAD12(28),HEADB(3)
14 FORMAT('R', I2, 'P', I1, 'H.DAT')
FILNAM(11) = 0
PRINT 16
PRINT 15, (FILNAM(I), I = 1, 10)
15 FORMAT(X,'DATA WILL BE SAVED IN FILE: ',10A1)
PRINT 16
CALL LISTHEAD(HEADB,HEAD12,HEAD14)
PRINT 16
16 FORMAT(X,80('-'))
PRINT 17
17 FORMAT(X/X,'DOES THIS DATA LOOK CORRECT? (Y/N) ',$)
READ 18, ANS
18 FORMAT(A1)
IF(ANS.EQ.'N') GO TO 100
OPEN(UNIT=10, NAME=FILNAM, STATUS='NEW', FORM='UNFORMATTED')
WRITE(10) HEADB
CLOSE(UNIT=10)

CALL EXIT
END
LISTHEAD.FOR

LISTS OUT THE HEADER INFORMATION FOR A

SPECIFIED TAPE.

---

C HEADB(1-2) HEADI2(1) ANALOG TAPE NUMBER
C HEADB(3) HEADI2(2) PASS NUMBER (1, 2 OR 3)
C HEADB(4) " NUMBER OF CHANNELS 14 OR 24
C HEADB(5) HEADI2(3) DIGITIZING MONTH
C HEADB(6) " DIGITIZING DAY
C HEADB(7) HEADI2(4) DIGITIZING YEAR
C HEADB(8) " EXPERIMENT TIME CODE START HOUR
C HEADB(9) HEADI2(5) EXPERIMENT TIME CODE START MINUTE
C HEADB(10) " EXPERIMENT TIME CODE START SECONDS
C HEADB(11-30) HEADI2(6-15) EVENT NAME (20 CHAR MAXIMUM)
C HEADB(31) HEADI2(16) ANALOG CHANNEL IN POSITION 1
C HEADB(32) " ANALOG CHANNEL IN POSITION 2
C HEADB(33) HEADI2(17) ANALOG CHANNEL IN POSITION 3
C HEADB(34) " ANALOG CHANNEL IN POSITION 4
C HEADB(35) HEADI2(18) ANALOG CHANNEL IN POSITION 5
C HEADB(36) " ANALOG CHANNEL IN POSITION 6
C HEADB(37) HEADI2(19) ANALOG CHANNEL IN POSITION 7
C HEADB(38) " ANALOG CHANNEL IN POSITION 8
C HEADB(39) HEADI2(20) ANALOG CHANNEL IN POSITION 9
C HEADB(40) " ANALOG CHANNEL IN POSITION 10
C HEADB(41) HEADI2(21) ANALOG CHANNEL IN POSITION 11
C HEADB(42) " ANALOG CHANNEL IN POSITION 12
C HEADB(43) HEADI2(22) ANALOG CHANNEL IN POSITION 13
C HEADB(44) " ANALOG CHANNEL IN POSITION 14
C HEADB(45) HEADI2(23) ANALOG CHANNEL IN POSITION 15
C HEADB(46) " ANALOG CHANNEL IN POSITION 16
C HEADB(47) HEADI2(24) ANALOG CHANNEL IN POSITION 17
C HEADB(48) " ANALOG CHANNEL IN POSITION 18
C HEADB(49) HEADI2(25) ANALOG CHANNEL IN POSITION 19
C HEADB(50) " ANALOG CHANNEL IN POSITION 20
C HEADB(51) HEADI2(26) ANALOG CHANNEL IN POSITION 21
C HEADB(52) " ANALOG CHANNEL IN POSITION 22
C HEADB(53) HEADI2(27) ANALOG CHANNEL IN POSITION 23
C HEADB(54) " ANALOG CHANNEL IN POSITION 24
C HEADB(55) HEADI2(28) RUN IDENTIFICATION NUMBER
C HEADB(56) " ANALOG CHANNEL IN POSITION 16
C HEADB(57-60) HEADFP(15) SAMPLE RATE (2.5 KHZ=.0004SEC)

---

SUBROUTINE LISTHEAD(HEADB,HEADI2,HEADI4)

---

BYTE HEADB(1)
INTEGER*2 HEADI2(1)
INTEGER*4 HEADI4(1)

---

PRINT OUT THE ALL HEADER INFO

1 PRINT 1, (HEADB(I),I=11,30)
   FORMAT(X,' *** EVENT DESCRIPTION: ',20A1,' ***')

2 PRINT 2, HEADI2(1)
   FORMAT(X/X,' ANALOG TAPE NUMBER: ',4X,I4)

3 PRINT 3, HEADI2(28)

C-6
FORMAT(X,' RUN IDENTIFICATION NUMBER: ',6X,I2)

PRINT 4, HEADB(3)
FORMAT(X,' PASS NUMBER: ',7X,I1)

PRINT 5, HEADB(8),HEADB(9),HEADB(10)
FORMAT(X,' START TIME CODE: ',I2,:,:I2,:,:I2)

PRINT 6, HEADB(5),HEADB(6),HEADB(7)
FORMAT(X,' DATE: ',I2,'/',I2,'/'I2)

NCHAN=HEADB(4)

PRINT 7
FORMAT(X/X,' POSITION : CHANNEL        POSITION : CHANNEL',
X/X, '---------------------

NHALF=NCHAN/2
DO I=1,NHALF
   K=I+NHALF
   PRINT 8, I,HEADB(I+30),K,HEADB(K+30)
8 FORMAT(X,4X,I2,3X,': ',3X,I2,2X,5X,3X,I2,3X,': ',3X,I2)
END DO

RETURN
END
APPENDIX D

LISTINGS OF
A2D.FOR
GETHEAD.FOR
LISTHEAD.FOR

D-1/D-2
Reverse Blank
A2D.FOR ANALOG TO DIGITAL DIGITIZING PROGRAM

C

THIS PROGRAM IS CALLED VIA THE A2D COMMAND ROUTINE.

EACH TAPE TO BE DIGITIZED WILL REQUIRE TWO PASSES AT 1/4 SPEED.

THE FIRST PASS WILL DIGITIZE 14 DATA CHANNELS 1-12, AND 14-15

THE SECOND PASS WILL DIGITIZE 14 DATA CHANNELS, 16-25 AND

REPEAT CHANNELS 2, 4, 10 AND 14. THE REPEATS WILL BE USED FOR

TAPE ALIGNMENT. MAKE SURE THE HARDWARE IS SET UP ACCORDINGLY.

TO DIGITIZE:

1) LOG ON TO VAX
2) SET UP THE DIGITIZING HARDWARE
3) MOUNT A TAPE WITH A WRITE RING ON MFA0:
4) EXECUTE THE A2D.COM COMMAND FILE, THIS WILL:
A. INITIALIZE THE TAPE
B. SET THE OUTPUT DIRECTORY/DISK
C. ASK FOR HEADER INFORMATION
D. ASSIGN AND MOUNT THE TAPE
E. SETUP THE HIGH SPEED TAPE DRIVE
F. ASSIGN THE NECESSARY PRIVILEGES
G. UP THE PROCESS PRIORITY TO 28
H. EXECUTE THE DIGITIZING PROGRAM, ***"A2D"***
I. LOWER THE PROCESS PRIORITY BACK TO 4
J. COPY THE RECAT DATA TO DISK
K. COPY THE QUICK CHECK PLOT DATA TO DISK
L. DISMOUNT THE TAPE
5) CHECK THE QUICK CHECK PLOTS
6) REPEAT 2-5

PARAMETER NADC=14 !NUMBER OF A/D CHANNELS USED
PARAMETER NBSIZE=500 !NUMBER OF WORDS IN ONE BUFFER/CHANNEL
PARAMETER NRECSIZE=NADC*NBSIZE*2

BYTE HEADB(NRECSIZE)
INTEGER*2 HEADI2(NRECSIZE/2)
INTEGER*2 BUFFER(NADC,NBSIZE,3),IOSB(4),IOSBM(4)
INTEGER*4 HEADI4(NRECSIZE/4)
INTEGER*4 IBUF(50),DWEIl,SYS$ADJWSL,SYS$LCKPAG,INADR(2),IRETADR(2)
1,SY$SQIQW,IOSB4(2),WRINORET,WEOFNORET,RUNNO
DIMENSION HEADFP(NRECSIZE/4)
EQUIVALENCE (HEADB,HEADI2,HEADI4,HEADFP)

CHARACTER*1 IGO

COMMON /COMM1/ IBUF !COMMONS FOR LONGWORD ALLIGNMENT
COMMON /COMM2/ BUFFER

EQUIVALENCE (IOSB,IBUF),(IOSBM,IOSB4)

C-----SET UP I/O CODES FOR NO RETRY DUE TO REAL-TIME PROCESSING

EXTERNAL IOS$ WRITELBLK,IOS$ RewIND,IOS$ WRITEOF
EXTERNAL IOS$ SKIPFILE,IOS$ INHRETRY
WRINORET=%LOC(IOS$ WRITELBLK)+%LOC(IOS$ INHRETRY) ! '8020'X
WEOFNORET=%LOC(IOS$ WRITEOF)+%LOC(IOS$ INHRETRY) ! '8028'X

C-----GET PROGRAM SAMPLING SPECIFICATIONS
C

PRINT *, 'HOW MANY BUFFERS DO YOU WANT TO FILL?'
READ *, NBUF

C

NBUF=450 ! NUMBER OF BUFFERS TO FILL
ICHN=0 ! START WITH CHANNEL 0
NCHN=NADC ! NUMBER OF CHANNELS
LBUN=NADC*NBSIZE ! SIZE OF EACH BUFFER IN WORDS (I*2)
LBUN2=LBUN*2 ! SIZE OF EACH BUFFER IN BYTES
IPRSET=-1 ! DOES NOT MATTER, NOT USED (-1 TO -32768)
MODE=512 ! SAMPLING OPTION - EXTERNAL SCHMIDT TRIGGER
IRATE=-1 ! CLOCK RATE DIRECT COUPLED SCHMIDT TRIGGER 1
DWELL=1 ! CLOCK OVERFLOWS BETWEEN SAMPLES
IEF=0 ! DEFAULT TO EVENT FLAG 22
LDELAY=0 ! IGNORED IN DEDICATED MODE
ITYP=2 ! DEDICATED A/D MODE
NUM=0 ! FOR LOGICAL NAME LPA1150
INC=1 ! CHANNEL INCREMENT

C

C-----GET THE HEADER INFORMATION FOR THIS TAPE
C

PRINT *, 'ENTER RUN IDENTIFICATION NUMBER'
READ *, RUNNO
PRINT *, 'ENTER PASS NUMBER (1 OR 2)'
READ *, IPASS
CALL GETHEAD(RUNNO,IPASS,LBUF2,HEADB)
CALL LISTHEAD(HEADB,HEADI2,HEADI4)

C

C-----ASSIGN THE TAPE TO LOGICAL NAME MAG_TAPE AND REWIND
C

CALL SYSSASSIGN('MAG_TAPE',ICHAN,)
IRETCD=SYSSQIOW(,%VAL(ICHAN),IOS_REWIND,IOSBMT,,)
IF(IOSBMT(1).EQ.'IA4'X) STOP 'TAPE DRIVE IS OFFLINE'
IF(.NOT.IRETCD) THEN
WRITE(6,150) 7,(IOSBMT(1),I=1,4)
STOP 'MAG_TAPE REWIND ERROR'
ENDIF

C

C-----SKIP FILES IF REQUESTED
C

1 PRINT *, 'HOW MANY FILES DO YOU WANT TO SKIP?'
READ *, NSKIP
IF(NSKIP.LT.0) THEN
PRINT *, 'YOU MUST ONLY GO IN THE FORWARD DIRECTION'
PRINT *, 'THE NUMBER OF FILES MUST BE POSITIVE'
GOTO 1
END IF
IRETCD=SYSSQIOW(,%VAL(ICHAN),IOS.SkipFILE,IOSB,,)
IF (.NOT.IRETCD) STOP 'SKIP FILE ERROR'
IF(IOSB(2).NE.NSKIP) THEN
PRINT *, 'PROBLEM SKIPPING ',NSKIP,' FILES'
PRINT *, IOSB
STOP 'SKIPPING FILE ERROR'
END IF

C

C-----DUMP THE HEADER TO TAPE
C

NREC=1 ! RECORD 1 IS HEADER

D-4
IRETCODE=SYS$QIOW(%VAL(ICHAN),%VAL(WRINORET),IOSBMT
1,,HEADB,%VAL(LBUF2),,,,)  
C WRITE(6,150) 7,(IOSBMT(I),I=1,4)  
IF(IOSBMT(1).EQ."IA4'X) STOP 'TAPE DRIVE IS OFFLINE'  
IF(.NOT.IRETCODE) STOP 'ERROR MAGTAPE DUMPING HEADER'  
C------ADJUST WORKING SET SIZE AND LOCK BUFFER AREA PAGES INTO MEMORY  
C IRETCODE=SYS$ADJWSL(%VAL(200),ILIMIT)  
IF(IRETCODE.NE.1) THEN  
WRITE(6,123) IRETCODE,ILIMIT  
123 FORMAT(' RETCODE,NEW LIMIT:',2I8)  
STOP 'ERROR ADJUSTING WORKING SET SIZE'  
ENDIF  
C INADR(1)=%LOC(BUFFER(1,1,1))  
INADR(2)=%LOC(BUFFER(NADC,NBSIZE,3))  
IRETCODE=SYS$LCKPAG(INADR, IRETADR,  
124 FORMAT(' INADR,IRETADR,IRETCODE FOR BUFFER',/,4012,Z6,I6)  
STOP 'ERROR LOCKING PAGES INTO MEMORY'  
ENDIF  
C------LOAD MICROCODE FOR MULTIREQUEST MODE  
C CALL LPA$LOADMC(ITYPE,NUM,IND,IERROR)  
IF(IND.EQ.1) GO TO 20  
C LIKELY ERROR IS THAT LOADER IS NOT RUNNING. TO RUN LOADER  
C @[SYSMGR]LPA11STRT  
C WRITE(6,10) IND  
10 FORMAT(' ERROR LOADING MICROCODE, VAX/VMS ERROR CODE=',Z12,' (HEX)')  
STOP  
C------DETERMINE SAMPLING RATES AND START THE CLOCK  
C 20 CALL LPA$CLOCKA(IRATE,IPRSET,IND,NUM)  
IF(IND.EQ.1) GO TO 40  
WRITE(6,30) IND  
30 FORMAT(' ERROR STARTING CLOCK, VAX/VMS ERROR CODE=',Z12,' (HEX)')  
STOP  
C------SET IBUF ARRAY FOR SWEEPS, DEFINE THE 3 INPUT BUFFERS  
C 40 CALL LPA$SETIBF(IBUF,IND,,BUFFER(1,1,1),BUFFER(1,1,2)
1,BUFFER(1,1,3))  
IF(IND.EQ.1) GO TO 50  
STOP ' ERROR SETTING UP IBUF ARRAY FOR SWEEPS, LPA$SETIBF'  
C------RELEASE ALL THE BUFFERS  
C 50 CALL LPA$RLSBUF(IBUF,IND,0,1,2)  
IF(IND.EQ.1) GO TO 60  
STOP ' ERROR RELEASING BUFFER, LPA$RLSBUF'  
C------SET CHANNEL INFORMATION FOR SWEEPS, IFLAG RESERVED
C 60 CALL LPA$SETADC(IBUF,IFLAG,ICHN,NCHN,INC,IND) 
  IF(IND.EQ.1) GO TO 70 
  STOP ' ERROR SETTING CHANNEL INFORMATION, LPA$SETADC' 
C
C----GET READY TO DIGITIZE
C
70 WRITE(6,80) HEADB(8),HEADB(9),HEADB(10) 
80 FORMAT(' START ANALOG TAPE, HIT RETURN AT ',12,';I2,':'',12,';I2,$) 
1001 READ 1001, IGO 
1001 FORMAT(A1) 
C
C----START THE A/D SWEEPS
C
CALL LPA$ADSWP(IBUF,LBUF,NBUF,MODE,DWELL,,LDELAY,ICHN,NCHN,IND) 
  IF(IND.EQ.1) GO TO 100 
  WRITE(6,90) IND 
  90 FORMAT(' ERROR STARTING SWEEP, VAX/VMS ERROR CODE=',Z12,' (HEX)') 
  STOP 
C
C-----WAIT FOR NEXT BUFFER TO COMPLETE 
C
100 WRITE(6,110) 7 ! TURN ON SAMPLING PULSES AT THE TONE 
110 FORMAT(1X,A) 
C
120 IBUFIN=LPA$IWTBUF(IBUF,) 
  IF(IBUFIN.GE.0) GO TO 140 
  WRITE(6,130) 7,7,7,IBUFIN,NREC,IOSB,7,7,7 
130 FORMAT(IX,3Al,'ERROR FROM LPA$IWTBUF - IBUFIN=',15,‘REC’,15 /,‘ISOB (HEX)’,4Zl2,3A1) 
  STOP 'IBUFIN < 0' 
C
C-----DUMP THIS BUFFER TO TAPE
C
140 CONTINUE 
  NREC=NREC+1 
  !INCREMENT RECORD NUMBER 
  PRINT *, NREC 
  IRETCODE=SYS$QIOW(,%VAL(ICHAN),%VAL(WRINORET),IOSBMT 
  1,,BUFFER(1,1,1+IBUFIN),%VAL(LBUF2),,,) 
  IF(.NOT.IRETCODE) STOP 'MAG TAPE ERROR' 
  IF(IOSBMT(1).NE.1) THEN 
    WRITE(6,150) 7,(IOSBMT(I),I=1,4) 
  150 FORMAT(X,A1,'PROBLEM WRITING TO TAPE , IOSB ',4(Z6)) 
    WRITE(6,160) NREC,IBUFIN 
  160 FORMAT(X,'RECORD ',15,' BUFFER NUMBER ',12) 
  ENDIF 
  IF(NREC.GT.NBUF) GO TO 180 
C
C-----RELEASE THIS BUFFER FOR A/D TO REFILL, GO GET NEXT ONE
C
CALL LPA$RLSBUF(IBUF,IND,IBUFIN) 
  IF(IND.NE.1) STOP 'LPA$RLSBUF' 
  GO TO 120 
C 
C-----COME HERE AT END OF A CUT AND STOP IN-PROGRESS SWEEP
C
180 CALL LPA$STPSWP(IBUF,1,IND) 
  IF(IND.EQ.1) GO TO 200 
  STOP 
  200 FORMAT(A1) 
  STOP
WRITE(6,190) IND
190 FORMAT(' ERROR STOPPING SWEEP, VAX/VMS ERROR CODE=',Z12,' (HEX)')
C
C----DIGITIZING COMPLETE, PUT TWO END OF FILES ON MAGTAPE
C
200 WRITE(6,210)7,7,7,7
210 FORMAT(' END OF RUN',4A1)
    IRETCODE=SYSSQIOW(%VAL(ICHAN),IO$_WRITEOF,IOSBMT,.......)
    IRETCODE=SYSSQIOW(%VAL(ICHAN),IO$_WRITEOF,IOSBMT,.......)
C
CALL EXIT
END
GETHEAD.FOR GETS OUT THE HEADER INFORMATION FOR A SPECIFIED TAPE.

- **GETHEAD.FOR**
  -ヘッドフォーマット
- **頭E：**
  -ディジタルヘッダー
- **頭E：**
  -デジタルヘッダー

**C**

```
SUBROUTINE GETHEAD(RUNNO,IPASS,NBYTES,HEADB)
```

- **ヘッドフォーマット**
  -ヘッドインフォメーション
- **頭E：**
  -ディジタルヘッダー
- **頭E：**
  -デジタルヘッダー

**C**

```
BYTE HEADB(1),FILNAM(11)
INTEGER RUNNO
```

- **ヘッドフォーマット**
  -ヘッドインフォメーション
- **頭E：**
  -ディジタルヘッダー
- **頭E：**
  -デジタルヘッダー

**C**

```
--OPEN THE FILE AND READ IN THE HEADER
```

- **ヘッドフォーマット**
  -ヘッドインフォメーション
- **頭E：**
  -ディジタルヘッダー
- **頭E：**
  -デジタルヘッダー

**C**

```
ENCODE(10,4,FILNAM) RUNNO,IPASS
4 FORMAT( 'R',12,'P',II,'H.DAT')
FILNAM(11)=O
PRINT 5, (FILNAM(),I=1,10)
5 FORMAT(X/X,'HEADER FILE: ',10A1//40(''-'),/) 
OPEN(UNIT=10,NAME=FILNAM,STATUS='OLD',FORM='UNFORMATTED',READONLY)
READ(10) (HEADB(I),I=1,NBYTES)
CLOSE(UNIT=10)
```
DB0: [MACIE.TM] GETHEAD.FOR; 5
RETURN
END
TM No. 861214

DRB0:[MACIE.TM]LISTHEAD.FOR;11

C LISTHEAD.FOR LISTS OUT THE HEADER INFORMATION FOR A SPECIFIED TAPE.

| C | HEADB(1-2) | HEADI2(1) | ANALOG TAPE NUMBER |
| C | HEADB(3)   | HEADI2(2) | PASS NUMBER (1, 2 OR 3) |
| C | HEADB(4)   |           | NUMBER OF CHANNELS 14 OR 24 |
| C | HEADB(5)   | HEADI2(3) | DIGITIZING MONTH |
| C | HEADB(6)   |           | DIGITIZING DAY |
| C | HEADB(7)   | HEADI2(4) | DIGITIZING YEAR |
| C | HEADB(8)   |           | EXPERIMENT TIME CODE START HOUR |
| C | HEADB(9)   | HEADI2(5) | EXPERIMENT TIME CODE START MINUTE |
| C | HEADB(10)  |           | EXPERIMENT TIME CODE START SECONDS |
| C | HEADB(11-30)| HEADI2(6-15) | EVENT NAME (20 CHAR MAXIMUM) |
| C | HEADB(31)  | HEADI2(16) | ANALOG CHANNEL IN POSITION 1 |
| C | HEADB(32)  |           | ANALOG CHANNEL IN POSITION 2 |
| C | HEADB(33)  | HEADI2(17) | ANALOG CHANNEL IN POSITION 3 |
| C | HEADB(34)  |           | ANALOG CHANNEL IN POSITION 4 |
| C | HEADB(35)  | HEADI2(18) | ANALOG CHANNEL IN POSITION 5 |
| C | HEADB(36)  |           | ANALOG CHANNEL IN POSITION 6 |
| C | HEADB(37)  | HEADI2(19) | ANALOG CHANNEL IN POSITION 7 |
| C | HEADB(38)  |           | ANALOG CHANNEL IN POSITION 8 |
| C | HEADB(39)  | HEADI2(20) | ANALOG CHANNEL IN POSITION 9 |
| C | HEADB(40)  |           | ANALOG CHANNEL IN POSITION 10 |
| C | HEADB(41)  | HEADI2(21) | ANALOG CHANNEL IN POSITION 11 |
| C | HEADB(42)  |           | ANALOG CHANNEL IN POSITION 12 |
| C | HEADB(43)  | HEADI2(22) | ANALOG CHANNEL IN POSITION 13 |
| C | HEADB(44)  |           | ANALOG CHANNEL IN POSITION 14 |
| C | HEADB(45)  | HEADI2(23) | ANALOG CHANNEL IN POSITION 15 |
| C | HEADB(46)  |           | ANALOG CHANNEL IN POSITION 16 |
| C | HEADB(47)  | HEADI2(24) | ANALOG CHANNEL IN POSITION 17 |
| C | HEADB(48)  |           | ANALOG CHANNEL IN POSITION 18 |
| C | HEADB(49)  | HEADI2(25) | ANALOG CHANNEL IN POSITION 19 |
| C | HEADB(50)  |           | ANALOG CHANNEL IN POSITION 20 |
| C | HEADB(51)  | HEADI2(26) | ANALOG CHANNEL IN POSITION 21 |
| C | HEADB(52)  |           | ANALOG CHANNEL IN POSITION 22 |
| C | HEADB(53)  | HEADI2(27) | ANALOG CHANNEL IN POSITION 23 |
| C | HEADB(54)  |           | ANALOG CHANNEL IN POSITION 24 |
| C | HEADB(55)  | HEADI2(28) | RUN IDENTIFICATION NUMBER |
| C | HEADB(56)  |           | |
| C | HEADB(57-60)| HEADFP(15) | SAMPLE RATE (2.5 KHZ = .0004SEC) |

SUBROUTINE LISTHEAD(HEADB, HEADI2, HEADI4)

BYTE HEADB(1)
INTEGER*2 HEADI2(1)
INTEGER*4 HEADI4(1)

C PRINT OUT THE ALL HEADER INFO

PRINT 1, (HEADB(I), I=11, 30)
1 FORMAT(X,'*** EVENT DESCRIPTION: ',2CA1,'***')

PRINT 2, HEADI2(1)
2 FORMAT(X/X,' ANALOG TAPE NUMBER: ',4X,I4)

PRINT 3, HEADI2(28)

D-10
FORMAT(X,' RUN IDENTIFICATION NUMBER: ',6X,I2)

PRINT 4, HEADB(3)
FORMAT(X,' PASS NUMBER: ',7X,I1)

PRINT 5, HEADB(8),HEADB(9),HEADB(10)
FORMAT(X,' START TIME CODE: ',I2,':' ,I2,':' ,I2)

PRINT 6, HEADB(5),HEADB(6),HEADB(7)
FORMAT(X,' DATE: ',I2,'/',I2,'/',I2).

NCHAN=HEADB(4)

PRINT 7
FORMAT(X/X,' POSITION : CHANNEL          POSITION : CHANNEL',
       X/X,'------------------          ------------------')

NHALF=NCHAN/2
DO I=1,NHALF
    K=I+NHALF
    PRINT 8, I,HEADB(I+30),K,HEADB(K+30)
FORMAT(X,4X,I2,3X,' : ',3X,I2,2X,5X,3X,I2,3X,' : ',3X,I2)
END DO

RETURN
END
APPENDIX E

LISTINGS OF
SAVRECAT.FOR
LISTHEAD.FOR
SAVRECAT.FOR reads a foreign mag tape made by the A2D program. It copies and separates the redundant channels (2, 4, 10, 13) beginning and ending records to a disk file. File names: R_P_C_B.DAT 500 buffers

---------------------------------------------------------------

PARAMETER NA2DCH=14 ! number of channels
PARAMETER NSIZE=500 ! number of words/channel in a record
PARAMETER NDIM=NA2DCH*NSIZE*2

BYTE FILNAM(14), HEADB(NDIM)
INTEGER*2 HEADI2(NDIM/2)
INTEGER*2 BUF1(NA2DCH,NSIZE), BUF2(NSIZE,NA2DCH), IOSB(4)
INTEGER*4 ICHAN, IRETCODE, SYS$QIOW, RUNNO, PASS
INTEGER*4 HEADI4(NDIM/4)
DIMENSION IRCH(4)
EXTERNAL 10$ READLBLK, IO$ REWIND, 10$ SKIPFILE, IO$ SKIPRECORD
EQUIVALENCE (THEADB, HEADI2-, HEADI4-)
DATA IRCH/2, 4, 10, 13/ ! redundant channels

NCHAN=NA2DCH ! number of channels
NWORDS=NCHAN*NSIZE ! number of I*2 words/block
NBYTES=NWORDS*2 ! number of bytes/block

----- Find out what run we are processing -----

PRINT *, 'ENTER RUN IDENTIFICATION NUMBER'
READ *, RUNNO
PRINT *, 'ENTER PASS NUMBER'
READ *, PASS
IGROUP=RUNNO-(RUNNO/10*10)
PRINT *, 'PROCESSING GROUP ', IGROUP

----- Assign the 9 track tape drive (6250) and make sure it is rewound -----

CALL SYSS$ASSIGN('MAG_TAPE', ICHAN, )
IRETCODE=SYSS$QIOW(,%VAL(ICHAN), IO$ REWIND, IOSB, , , , , , )
IF (.NOT.IRETCODE) STOP 'REWIND ERROR'

----- Skip files if requested -----

1 PRINT *, 'HOW MANY FILES DO YOU WANT TO SKIP?'
READ *, NSKIP
IF(NSKIP.LT.0) THEN
   PRINT *, 'YOU MUST ONLY GO IN THE FORWARD DIRECTION'
   PRINT *, 'THE NUMBER OF FILES MUST BE POSITIVE'
   GOTO 1
END IF
IRETCODE=SYSS$QIOW(,%VAL(ICHAN), IO$ SKIPFILE, IOSB, , %VAL(NSKIP), , , , )
IF (.NOT.IRETCODE) STOP 'SKIP FILE ERROR'
IF(IOSB(2).NE.NSKIP) THEN
   PRINT *, 'PROBLEM SKIPPING ', NSKIP, ' FILES'
   PRINT *, 'SKIPPING FILE ERROR'
ENDIF

----- Read in the header and print out -----

E-3
IRETCODE=SYS$QIOW(,%VAL(ICHAN),IO$_READBLK,IOSB,,
1    HEADB,%VAL(NBYTES),,,)
    IF (.NOT.IRETCODE) STOP 'READING HEADER ERROR'
    IF(IOSB(2).NE.NBYTES) STOP 'WRONG NUMBER OF BYTES READ FOR HEADER'
    CALL LISTHEAD(HEADB,HEADI2,HEADI4)
C
C------OPEN OUTPUT FILES, CHANNELS WILL SEPERATED INTO R__P_C__B.DAT
C
    DO K=1,4
        I=IRCH(K)
        IUNIT=I+10
        ENCODE(13,10,FILNAM) RUNNO,PASS,I
        DO 10 FORMAT( 'R' ,I2, 'P' ,I1, 'C' ,I2, 'B.DAT'
        FILNAM(14)=0
            OPEN(UNIT=IUNIT,STATUS='NEW' ,FILE=FILNAM,FORM='UNFORM4ATTED')
            PRINT 20, IUNIT,(FILNAM(J),J=1,13)
    20 FORMAT (2X,'OUTPUT UNIT= ',15,2X,'FILE',13A1,' 500 BUFFERS')
    END DO
C
C------READ THE DATA IN
C    NRECORDS=500 !WE WANT 500 BEGINNING RECORDS
    DO NR=1,NRECORDS
        IRETCODE=SYS$QIOW(,%VAL(ICHAN),IO$_READBLK,IOSB,,
1        BUF1,%VAL(NBYTES),,,)
        IF (.NOT.IRETCODE) STOP 'ERROR READING DATA RECORD'
        IF(IOSB(1).EQ.'0870'X) THEN
            PRINT *, 'AT EOF IN RECORD ',NR
        ENDIF
        IF(IOSB(2).NE.NBYTES) STOP 'WRONG NUMBER OF BYTES READ FOR DATA'
C
C------SEPERATE OUT EACH CHANNEL AND SAVE IT ON DISK
C
    DO K=1,4
        I=IRCH(K)
        DO J=1,NSIZE
            BUF2(J,I)=BUF1(I,J)
        END DO
        IUNIT=10-I !OUTPUT UNITS 14-26
        WRITE(IUNIT) (BUF2(J,I),J=1,NSIZE)
    END DO
C
    DO K=1,4
        IUNIT=IRCH(K)+10
        CLOSE(UNIT=IUNIT)
    END DO
C
C------SKIP RECORDS TO END OF TAPE (FOR GROUPS 1-3 ONLY, 4 TOTAL)
C
    IF(IGROUP.EQ.4) CALL EXIT
    IF(PASS.EQ.1) THEN
        NSKIP=9250 !WE RECORDED 9800 FOR PASS 1
    ELSE
        NSKIP=9370 !WE RECORDED 9920 FOR PASS 2
    ENDIF
E-4
IRETCODE=SYS$QIOW(,%VAL(ICHAN),IO$_SKIPRECORD,IOSB,,,%VAL(NSKIP),,,,)
IF (.NOT.IRETCODE) THEN
    PRINT 600, (IOSB(I),I=1,4)
760 FORMAT(2X,'IOSB ',4Z8.8)
    STOP 'SKIP RECORD ERROR'
ENDIF
IF(IOSB(2).NE.NSKIP) THEN
    PRINT *, 'PROBLEM SKIPPING ',NSKIP,' RECORDS'
    PRINT *, IOSB
    STOP 'SKIPPING RECORD ERROR'
ENDIF
C
C-----OPEN OUTPUT FILES, CHANNELS WILL SEPERATED INTO R__P_C__E.DAT

DO K=1,4
    I=IRCH(K)
    IUNIT=I+10
    ENCODE(13,30,FILNAM) RUNNO,PASS,I
    30 FORMAT('R',12,'P',I1,'C',12,'E.DAT')
    FILNAM(14)=0
    OPEN(UNIT=IUNIT,STATUS='NEW',FILE=FILNAM,FORM='UNFORMATTED')
    PRINT 40, IUNIT,(FILNAM(J),J=1,13)
40 FORMAT (2X,'OUTPUT UNIT= ',15,2X,'FILE ',13AI,'BUFFERS')
END DO
C

C-----READ THE DATA IN

DO NR=1,51 !WE WANT 50 BUFFERS, MAKE SURE AT EOF
    IRETCODE=SYS$QIOW(,%VAL(ICHAN),IO$_READBLK,IOSB,,, Buf1,%VAL(NBYTES)),,,)
IF (.NOT.IRETCODE) STOP 'ERROR READING DATA RECORD'
IF(IOSB(1).EQ.'0870'X) THEN
    PRINT *, 'AT EOF IN RECORD ',NR
    GO TO 2
ENDIF
IF(IOSB(2).NE.NBYTES) STOP 'WRONG NUMBER OF BYTES READ FOR DATA'
C
C-----SEPERATE OUT EACH CHANNEL AND SAVE IT ON DISK

DO K=1,4
    !SWAP ORDER OF DATA
    I=IRCH(K)
    DO J=1,NSIZE
        BUF2(J,I)=BUF1(I,J)
    END DO
    IUNIT=10+I !OUTPUT UNITS 14-26
    WRITE(IUNIT) (BUF2(J,I),J=1,NSIZE)
END DO
END DO
C
C-----CLOSE THE FILES

2 DO K=1,4
    IUNIT=IRCH(K)+10
    CLOSE(UNIT=IUNIT)
END DO
PRINT *, 'RECAT FILES HAVE BEEN SAVED'
call EXIT
TM No. 861214

_END_

DRBO:[MACIE.TM]SAVRECAT.FOR;3
LISTHEAD FOR LISTS OUT THE HEADER INFORMATION FOR A
SPECIFIED TAPE.

C

C HEADB(1-2) HEADI2(1) ANALOG TAPE NUMBER
C HEADB(3) HEADI2(2) PASS NUMBER (1, 2 OR 3)
C HEADB(4) " NUMBER OF CHANNELS 14 OR 24
C HEADB(5) HEADI2(3) DIGITIZING MONTH
C HEADB(6) " DIGITIZING DAY
C HEADB(7) HEADI2(4) DIGITIZING YEAR
C HEADB(8) " EXPERIMENT TIME CODE START HOUR
C HEADB(9) HEADI2(5) EXPERIMENT TIME CODE START MINUTE
C HEADB(10) " EXPERIMENT TIME CODE START SECONDS
C HEADB(11-30) HEADI2(6-15) EVENT NAME (20 CHAR MAXIMUM)
C HEADB(31) HEADI2(16) ANALOG CHANNEL IN POSITION 1
C HEADB(32) " ANALOG CHANNEL IN POSITION 2
C HEADB(33) HEADI2(17) ANALOG CHANNEL IN POSITION 3
C HEADB(34) " ANALOG CHANNEL IN POSITION 4
C HEADB(35) HEADI2(18) ANALOG CHANNEL IN POSITION 5
C HEADB(36) " ANALOG CHANNEL IN POSITION 6
C HEADB(37) HEADI2(19) ANALOG CHANNEL IN POSITION 7
C HEADB(38) " ANALOG CHANNEL IN POSITION 8
C HEADB(39) HEADI2(20) ANALOG CHANNEL IN POSITION 9
C HEADB(40) " ANALOG CHANNEL IN POSITION 10
C HEADB(41) HEADI2(21) ANALOG CHANNEL IN POSITION 11
C HEADB(42) " ANALOG CHANNEL IN POSITION 12
C HEADB(43) HEADI2(22) ANALOG CHANNEL IN POSITION 13
C HEADB(44) " ANALOG CHANNEL IN POSITION 14
C HEADB(45) HEADI2(23) ANALOG CHANNEL IN POSITION 15
C HEADB(46) " ANALOG CHANNEL IN POSITION 16
C HEADB(47) HEADI2(24) ANALOG CHANNEL IN POSITION 17
C HEADB(48) " ANALOG CHANNEL IN POSITION 18
C HEADB(49) HEADI2(25) ANALOG CHANNEL IN POSITION 19
C HEADB(50) " ANALOG CHANNEL IN POSITION 20
C HEADB(51) HEADI2(26) ANALOG CHANNEL IN POSITION 21
C HEADB(52) " ANALOG CHANNEL IN POSITION 22
C HEADB(53) HEADI2(27) ANALOG CHANNEL IN POSITION 23
C HEADB(54) " ANALOG CHANNEL IN POSITION 24
C HEADB(55) HEADI2(28) RUN IDENTIFICATION NUMBER
C HEADB(56) " "
C HEADB(57-60) HEADFP(15) SAMPLE RATE (2.5 KHZ=.0004SEC)

C---SUBROUTINE LISTHEAD(HEADB,HEADI2,HEADI4)

C

BYTE HEADB(1)
INTEGER*2 HEADI2(1)
INTEGER*4 HEADI4(1)
C
C---PRINT OUT THE ALL HEADER INFO
C
PRINT 1, (HEADB(I),I=1,30)
FORMAT(X,'*** EVENT DESCRIPTION: ',20A',' ***')
C
PRINT 2, HEADI2(1)
FORMAT(X/X,' ANALOG TAPE NUMBER: ',4X,14)
C
PRINT 3, HEADI2(28)
1  \text{FORMAT}(X, \,' \text{RUN IDENTIFICATION NUMBER: ',6X,I2)}
2  \text{PRINT 4, HEADB(3)}
3  \text{FORMAT}(X, \,' \text{PASS NUMBER: ',7X,I1)}
4  \text{PRINT 5, HEADB(8),HEADB(9),HEADB(10)}
5  \text{FORMAT}(X, \,' \text{START TIME CODE: ',I2,'\:'},I2,':\:'),I2)
6  \text{PRINT 6, HEADB(5),HEADB(6),HEADB(7)}
7  \text{FORMAT}(X, \,' \text{DATE: ',I2, '/',I2, '/')}\12\text{DATE: ',I2, '/',I2, '/'})
8  NCHAN=HEADB(4)
9  NHALF=NCHAN/2
10  DO I=1,NHALF
11    K=I+NHALF
12    \text{PRINT 8, I,HEADB(I+30),K,HEADB(K+30)}
13  END DO
14  \text{RETURN}
15  END
APPENDIX F

LISTINGS OF
READDT.FOR
LISTHEAD.FOR
C READDT.FOR READS A FOREIGN MAG TAPE MADE BY THE A2D PROGRAM
C COPIES AND SEPARATES THE DIFFERENT CHANNELS INTO
C DIFFERENT FILES. THIS VERSION IS FOR 14 CHANNELS.

C

PARAMETER NA2DCH=14 !NUMBER OF CHANNELS
PARAMETER NSIZE=500 !NUMBER OF WORDS/CHANNEL IN A RECORD
PARAMETER NDIM=NA2DCH*NSIZE*2

BYTE FILE(13),HEADD(NDIM)
INTEGER*2 HEAD12(NDIM/2)
INTEGER*2 BUF1(NA2DCH,NSIZE),BUF2(NSIZE,NA2DCH),IOSB(4)
INTEGER*4 ICHAN,IRETCODE,SYS$QIOW,RUNNO,PASS
INTEGER*4 HEAD14(NDIM/4)
EXTERNAL IO$_READBLK,IO$_REWIND,IO$_SKIPFILE

C

NCHAN=NA2DCH !NUMBER OF CHANNELS
NWORDS=NCHAN*NSIZE !NUMBER OF I*2 WORDS/BLOCK
NBYTES=NWORDS*2 !NUMBER OF BYTES/BLOCK

C

C ----- FIND OUT WHAT RUN WE ARE PROCESSING
C
PRINT *, 'ENTER RUN IDENTIFICATION NUMBER'
READ *,RUNNO
PRINT *, 'ENTER PASS NUMBER'
READ *,P ASS

C-----OPEN OUTPUT FILES, CHANNELS WILL SEPARATED INTO R__P_C__.DAT
C
DO I=1,NCHAN
  IUNIT=I+10
  ICHAN=I
  ENCODE(12,10,FILNAM) RUNNO,PASS,ICHAN
  FORMAT('R',12,'P',11,'C',I2,'.DAT')
  FILNAM(13)=0
  OPEN(UNIT=IUNIT,STATUS='NEW',FILE=FILNAM,FORM='UNFORMATTED')
  PRINT 20, IUNIT,(FILNAM(J),J=1,12)
  FORMAT (2X,'OUTPUT UNIT= ',I5,2X,'FILE ',I2A1)
END DO

C-----ASSIGN THE 9 TRACK TAPE DRIVE (6250) AND MAKE SURE IT IS REWOUND
C
CALL SYS$ASSIGN('MAG TAPE',ICHAN,)
IRETCODE=SYS$QIOW(,%VAL(ICHAN),IOS_RE WIND,IOSB,........)
IF (.NOT.IRETCODE) STOP 'REWIND ERROR'

C-----SKIP FILES IF REQUESTED
C
1 PRINT *, 'HOW MANY FILES DO YOU WANT TO SKIP?'
READ *,NSKIP
IF(NSKIP.LT.0) THEN
  PRINT *, 'YOU MUST ONLY GO IN THE FORWARD DIRECTION'
  PRINT *, 'THE NUMBER OF FILES MUST BE POSITIVE'
  GOTO 1
END IF
IRETCODE=SYS$QIOW(,%VAL(ICHAN),IOS_SKIPFILE,IOSB,%,VAL(NSKIP),......)
IF (.NOT.IRETCODE) STOP 'SKIP FILE ERROR'
IF(IOSB(2).NE.NSKIP) THEN
  PRINT *, 'PROBLEM SKIPPING ', NSKIP, ' FILES'
  PRINT *, IOSB
  STOP 'SKIPPING FILE ERROR'
END IF

C-----READ IN THE HEADER AND PRINT OUT
C
  IRETCODE=SYSSQIOW(,%VAL(ICHAN),IOS_READLBLK,IOSB,,
  1     HEADB,%VAL(NBYTES),,,)
  IF (.NOT.IRETCODE) STOP 'READING HEADER ERROR'
  IF(IOSB(2).NE.NBYTES) STOP 'WRONG NUMBER OF BYTES READ FOR HEADER'
  CALL LISTHEAD(HEADB,HEADI2,HEADI4)
C

C-----READ THE DATA IN
C
  PRINT *, 'HOW MANY RECORDS DO YOU WANT TO READ?'
  READ *, NRECORDS
C
  DO NR=1,NRECORDS
C
    IRETCODE=SYSSQIOW(,%VAL(ICHAN),IOS_READLBLK,IOSB,,
    1       BUF1,%VAL(NBYTES),,,)
    IF (.NOT.IRETCODE) STOP 'ERROR READING DATA RECORD'
    IF(IOSB(1).EQ.'0870'X) THEN
      PRINT *, 'AT EOF IN RECORD ', NR
    ENDIF
    IF(IOSB(2).NE.NBYTES) STOP 'WRONG NUMBER OF BYTES READ FOR DATA'
C
C-----SEPERATE OUT EACH CHANNEL AND SAVE IT ON DISK
C
  DO I=1,NCHAN !SWAP ORDER OF DATA
    DO J=1,NSIZE
      BUF2(J,I)=BUF1(I,J)
    END DO
  END DO
  DO I=1,NCHAN !OUTPUT UNITS 14-26
    IUNIT=104-I
    WRITE(IUNIT) (BUF2(J,,I),J=1,NSIZE)
  END DO
C
C-----CLOSE THE FILE
C
  DO I=1,NCHAN
    IUNIT=10*I
    CLOSE(IUNIT)
  END DO
C
CALL EXIT
END
LISTHEAD.FOR LISTS OUT THE HEADER INFORMATION FOR A
SPECIFIED TAPE.

C HEADB(1-2) HEADI2(1) ANALOG TAPE NUMBER
C HEADB(3) HEADI2(2) PASS NUMBER (1, 2 OR 3)
C HEADB(4) " NUMBER OF CHANNELS 14 OR 24
C HEADB(5) HEADI2(3) DIGITIZING MONTH
C HEADB(6) " DIGITIZING DAY
C HEADB(7) HEADI2(4) DIGITIZING YEAR
C HEADB(8) " EXPERIMENT TIME CODE START HOUR
C HEADB(9) HEADI2(5) EXPERIMENT TIME CODE START MINUTE
C HEADB(10) " EXPERIMENT TIME CODE START SECONDS
C HEADB(11-30) HEADI2(6-15) EVENT NAME (20 CHAR MAXIMUM)
C HEADB(31) HEADI2(16) ANALOG CHANNEL IN POSITION 1
C HEADB(32) " ANALOG CHANNEL IN POSITION 2
C HEADB(33) HEADI2(17) ANALOG CHANNEL IN POSITION 3
C HEADB(34) " ANALOG CHANNEL IN POSITION 4
C HEADB(35) HEADI2(18) ANALOG CHANNEL IN POSITION 5
C HEADB(36) " ANALOG CHANNEL IN POSITION 6
C HEADB(37) HEADI2(19) ANALOG CHANNEL IN POSITION 7
C HEADB(38) " ANALOG CHANNEL IN POSITION 8
C HEADB(39) HEADI2(20) ANALOG CHANNEL IN POSITION 9
C HEADB(40) " ANALOG CHANNEL IN POSITION 10
C HEADB(41) HEADI2(21) ANALOG CHANNEL IN POSITION 11
C HEADB(42) " ANALOG CHANNEL IN POSITION 12
C HEADB(43) HEADI2(22) ANALOG CHANNEL IN POSITION 13
C HEADB(44) " ANALOG CHANNEL IN POSITION 14
C HEADB(45) HEADI2(23) ANALOG CHANNEL IN POSITION 15
C HEADB(46) " ANALOG CHANNEL IN POSITION 16
C HEADB(47) HEADI2(24) ANALOG CHANNEL IN POSITION 17
C HEADB(48) " ANALOG CHANNEL IN POSITION 18
C HEADB(49) HEADI2(25) ANALOG CHANNEL IN POSITION 19
C HEADB(50) " ANALOG CHANNEL IN POSITION 20
C HEADB(51) HEADI2(26) ANALOG CHANNEL IN POSITION 21
C HEADB(52) " ANALOG CHANNEL IN POSITION 22
C HEADB(53) HEADI2(27) ANALOG CHANNEL IN POSITION 23
C HEADB(54) " ANALOG CHANNEL IN POSITION 24
C HEADB(55) HEADI2(28) RUN IDENTIFICATION NUMBR
C HEADB(56) " ANALOG CHANNEL IN POSITION 25
C HEADB(57-60) HEADFP(15) SAMPLE RATE (2.5 KHZ=.0004SEC)

SUBROUTINE LISTHEAD(HEADB, HEADI2, HEADI4)

BYTE HEADB(1)
INTEGER*2 HEADI2(1)
INTEGER*4 HEADI4(1)

PRINT OUT THE ALL HEADER INFO

PRINT 1, (HEADB(I), I=11,30)
FORMAT(X,'*** EVENT DESCRIPTION: ',20A1,'***')

PRINT 2, HEADI2(1)
FORMAT(X/X,' ANALOG TAPE NUMBER: ',4X,14)

PRINT 3, HEADI2(28)
TM No. 861214

_DRB0:[MACIE.TM]LISTHEAD FOR;11

3 FORMAT(X,' RUN IDENTIFICATION NUMBER: ',6X,I2)
4 PRINT 4, HEADB(3)
   FORMAT(X,' PASS NUMBER: ',7X,I1)
5 PRINT 5, HEADB(8),HEADB(9),HEADB(10)
   FORMAT(X,' START TIME CODE: ',I2,':',I2,':',I2)
6 PRINT 6, HEADB(5),HEADB(6),HEADB(7)
   FORMAT(X,' DATE: ',12,'/',12,'/12')

C

NCHAN=HEADB(4)

C

PRINT 7
7 FORMAT(X/X,' POSITION : CHANNEL POSITION : CHANNEL',
            X/X,'-'-----------------------------------')

NHALLF=NCHAN/2
DO I=1,NHALF
   K=I+NHALF
   PRINT 8, I,HEADB(I+30),K,HEADB(K+30)
8   FORMAT(X,4X,I2,3X,' ',3X,I2,2X,5X,3X,I2,3X,' ',3X,I2)
END DO

C

RETURN
END
APPENDIX G

LISTINGS OF
QGPL.COM
QGPL.FOR
PLCHAN.FOR
QUPL.COM  COMMAND STREAM FOR QUICK PLOT CHECK OF 1ST 4 RECORDS
OF ALL 14 A/D CHANNELS.

SET VERIFY

IF P1 .EQS. "" THEN INQUIRE P1 "RUN NUMBER? "

IF P2 .EQS. "" THEN INQUIRE P2 "PASS NUMBER? "

IF P3 .EQS. "" THEN -
INQUIRE P3 "ARE YOU RUNNING ON THE TEKTRONICS? (Y/N)"

SET DEF DRB1:[A2D.QUPL]
OPEN/WRITE QUPL QUPL.IN
WRITE QUPL P1
WRITE QUPL P2
WRITE QUPL "0.,.2"
WRITE QUPL "0.,.2,.05"

IF F$STRING(P3) .EQS. "N" THEN GOTO TEKT
WRITE QUPL 2
WRITE QUPL 2
WRITE QUPL 2
WRITE QUPL 2
GOTO MAKEPLOT

TEKT:
WRITE QUPL 1
WRITE QUPL 1
WRITE QUPL 1
WRITE QUPL 1

MAKEPLOT:
CLOSE QUPL
ASSIGN/USER_MODE QUPL.IN FOR$READ
RUN DRB0:[A2D.DIGIT]QUPL !SEND THE PLOT TO THE DISK
DELETE QUPL.IN;*

DONE:
WRITE SYS$OUTPUT "PLOTTING COMPLETE"
EXIT
QUPL.FOR performs a quicklook plot of digitized data that has been copied from tape to disk. It is plotted one channel/plot and 4 channels/page. Right now it plots the first 4 records of 500 pts. All channels are plotted. 1-4, 5-8, 9-12, 13+14

PARAMETER NDIM=4096,NSIZE=500
BYTE FILNAM(13)
INTEGER*2 BUFFER(NSIZE)
INTEGER CHANNO,RUNNO,PASSNO,ICHAN(4)
REAL X(NDIM),Y(NDIM)
COMMON ISTART,ISTOP,TMIN,TMAX,TDEL
NCHAN=14
NREC=4

C-----find out what run we are processing

PRINT *, 'ENTER RUN IDENTIFICATION NUMBER'
READ *,RUNNO
PRINT *, 'ENTER PASS NUMBER'
READ *, PASSNO
PRINT *, 'RUN ',RUNNO,', PASS ',PASSNO

C------get operators requested plot bounds

DELTAT=1./2500./4. !SAMPLE INTERVAL
NPTS=FLOAT(NREC)*NSIZE
TIMMAX=FLOAT(NPTS)*DELTAT
PRINT *, ''
PRINT *, 'DATA RANGE IS 0 THRU',TIMMAX,' (SEC), DELTA=',DELTAT,' (SEC)'
PRINT *, 'INPUT DESIRED DATA MIN AND MAX IN SECONDS'
READ *, DMIN,DMAX
PRINT *, ''
PRINT *, 'INPUT PLOT TIME AXIS MIN, MAX AND LABEL INCREMENT IN SECONDS'
READ *, TMIN,TMAX,TDEL
PRINT *, ''

C------make up time axis, get start and stop

DO 1 IS=1,NPTS
X(IS)=FLOAT(IS-1)*DELTAT
1 CONTINUE

DO 2 I=1,NPTS
IF(X(I).GE.TMIN) THEN
ISTART=I
GO TO 3
ENDIF
2 CONTINUE

DO 4 I=NPTS,1,-1
IF(X(I).LE.TMAX) THEN
ISTOP=I
GO TO 5
ENDIF
4 CONTINUE

G-4
CONTINUE

C----- OPEN THE FILES, NAMING CONVENTION R_P_C_.DAT
C
   IPLLOT = 1
   DO K = 1, NCHAN
      CHANNON = K
      IUNIT = 10
      ENCODE(12, 10, FILNAM) RUNNO, PASSNO, CHANNO
      FORMAT('R',I2, 'P', I1, 'C', I2, '_DAT')
      FILNAM(13) = 0
      OPEN(UNIT = IUNIT, STATUS = 'OLD', FILE = FILNAM, FORM = 'UNFORMATTED')
C
C----- READ IN THE DATA
C
   N = 0
   DO NREC = 1, 4
      READ(IUNIT, END = 999) (BUFFER(I), I = 1, NSIZE)
      DO I = 1, NSIZE
         N = N + 1
         A = BUFFER(I)
         Y(N) = A / 409.5 - 5
      END DO
   END DO
   CALL CLOSE(IUNIT)
C
C----- PLOT THE DATA
C
   ILAST = 0
   IF(K.EQ.14) ILAST = 1
   IF(IPLLOT.EQ.4) ILAST = 1
   CALL PCHAN(ILAST, IPLLOT, RUNNO, PASSNO, CHANNO, N, X, Y)
   IPLLOT = IPLLOT + 1
   IF(IPLLOT.EQ.5) IPLLOT = 1
   END DO
C
999  CALL EXIT
END
SUBROUTINE PLCHAN(ILAST, IPLOT, IRUN, IPASS, ICHAN, NPT, X, Y)

C ----- SET UP X AXIS
C
XMIN = TMIN  ! X AXIS MINIMUM
XMAX = TMAX  ! X AXIS MAXIMUM
XRANGE = XMAX - XMIN  ! X AXIS RANGE
NX = XRANGE / TDEL  ! X AXIS # INCREMENTS
XDEL = TDEL  ! X AXIS LABEL INCREMENT
XSPACE = 8000.  ! X AXIS SPACE
XMID = XSPACE / 2.  ! X AXIS MIDPOINT
XSTART = 1500.  ! X AXIS START POINT

C ----- SET UP THE Y AXIS
C
YMIN = -5.  ! Y AXIS MINIMUM
YMAX = 5.  ! Y AXIS MAXIMUM
YRANGE = YMAX - YMIN  ! Y AXIS RANGE
NY = 2  ! Y AXIS # INCREMENTS
YDEL = YRANGE / FLOAT(NY)  ! Y AXIS LABEL INCREMENTS
YSPACE = 1200.  ! Y AXIS SPACE
YMID = YSPACE / 2.  ! Y AXIS MIDPOINT
YSTART = 7600. - (YSPACE + 400.) * FLOAT(IPLOT)  ! Y AXIS START POINT

C ----- INITIALIZE THE PLOTTING
C
IF(IPLOT.EQ.1) THEN
  PRINT *, 'PLOT TO IPF FILE ONLY(1)/TO TERMINAL ONLY(2)/BOTH?(3)'
  READ *, IOPTION
  CALL BFIL(IOPTION)
  CALL BJOB ! INITIALIZES A JOB
  CALL BFRM(1) ! WANT 11X8.5
ENDIF

C ----- SET DATA SPACE AND GRID
C
CALL SWRKS(XSTART, YSTART, XSPACE, YSPACE, 0.)  ! SET UP WORK SPACE
CALL DRECG(NX, NY*2, 0, 0, -100., -100.)  ! MAKE RECTANGULAR GRID
CALL SDATP(XMIN, YMIN, XRANGE, YRANGE)  ! SET UP DATA SPACE

C ----- WRITE TITLE ON PLOT
C
IF(IPLOT.EQ.1) THEN
  CALL SSPC(300., 300.)  ! SET CHARACTER SIZE
  CALL SJST(0, 0)  ! LT, BOTTOM JUSTIFY
  TITLAB = '**EXPERIMENT TITLE**'//CHAR(0)
  CALL LSTRG(TITLAB, 950., 1600.)  ! WRITE TITLE
  TITLAB = 'RUN - PASS'//CHAR(0)
  CALL LSTRG(TITLAB, 1700., 1300.)  ! WRITE TITLE
  CALL LNUMI(IRUN, 2900., 1300.)  ! WRITE RUN ID NUMBER
ENDIF
CALL LNUMI(IPASS, 5900., 1300.) !WRITE PASS NUMBER
ENDIF

C-----WRITE CHANNEL NUMBER ON PLOT
C
CALL SSPC(300., 300.) !SET CHARACTER SIZE
CALL SJST(0, 0) !LT, BOTTOM JUSTIFY
TITLAB='CHAN'//CHAR(0)
CALL LSTRG(TITLAB, 8100., 650.) !WRITE CHANNEL
CALL SJST(1, 1) !CENTER JUSTIFY
CALL LNUMI(ICHAN, 8700., 400.) !WRITE CHANNEL NUMBER
C
C-----LABEL THE X AXIS
C
IF(ILAST.EQ.1) THEN
CALL SSPC(300., 300.) !SET CHARACTER SIZE
CALL SJST(1, 1) !CENTER JUSTIFY
XSPINC=XSPACE/FLOAT(NX) !Y SPACE INCREMENT
XVAL=0. !INITIAL X SPACE
VALUE=XMIN !INITIAL X VALUE
DO 6 I=1, NX+1
IVALUE=VALUE*1000 !WANT MILLESEC LABEL
CALL LNUMI(IVALUE, XVAL, -350.) !WRITE X LABEL
VALUE=VALUE*XDEL !GET NEXT X VALUE
XVAL=XVAL+XSPINC !GET NEXT X SPACE
6 CONTINUE
TITLAB='TIME 1/4 SPEED (MSEC)'/CH(A(0)
CALL LSTRG(TITLAB, XMID, -750.) !MAKE X AXIS LABEL
TITLAB='VOLTAGE (VOLTS) '/CH(A(0)
CALL SJST(1, 1) !CENTER JUSTIFY
CALL SROT(0., 90.) !ROTATE 90
YVAL=(7600.-YSTART)/2. !FIGURE OUT CENTER
CALL LSTRG(TITLAB, -1250., YVAL) !MAKE Y AXIS LABEL
CALL SROT(0., 0.) !ROTATE BACK
ENDIF
C
C-----LABEL THE Y AXIS
C
CALL SSPC(300., 300.) !SET CHARACTER SIZE
CALL SJST(2, 1) !RT, CENTER JUSTIFY
YSPINC=YSPACE/FLOAT(NY) !Y SPACE INCREMENT
YVAL=0. !INITIAL Y SPACE
VALUE=YMIN !INITIAL Y VALUE
DO 7 I=1, NY+1
ID=VALUE !GET INTEGER LABEL
CALL LNUMI(ID, -100., YVAL) !WRITE Y INTEGER LABEL
VALUE=VALUE+YDEL !GET NEXT Y VALUE
YVAL=YVAL+YSPINC !GET NEXT Y SPACE
7 CONTINUE
C
C-----PLOT THE DATA
C
CALL SWRKS(XSTART, YSTART, XSPACE, YSPACE, 0.) !SET UP WORK SPACE
CALL SDATP(XMIN, YMIN, XRANGE, YRANGE) !SET UP DATA SPACE
N=ISTOP-ISTART+1
CALL DRECP(N, X(ISTART), Y(ISTART), 0, 0, 0) !PLOT DATA POINTS
C
C-----CLOSE PLOTTING IF DONE
C

IF(ILAST.EQ.1) THEN
    CALL EFRM
    CALL EJOB
    CALL EFIL
ENDIF

C

RETURN
END
APPENDIX H

LISTINGS OF
DOWNSAMP.FOR
LISTHEAD.FOR

H-1/H-2
Reverse Blank
DOWNsamp.for reads a foreign mag tape made by the A2D program.
It then down samples and writes all channels to
a data file for future merging. All left over data
for a group is saved and later appended to the
following group. For pass 2, the garbage data due
to alignment is thrown away.

parameter na2dch=14 !number of channels
parameter nisize=500 !number of words/channel in an input record
parameter nosize=1024 !number of words/channel in an output record

parameter ndim=na2dch*nisize*2

byte filnam(10),headb(ndim)
integer*2 iosb(4),head12(ndim/2)
integer*2 buf1(na2dch,nisize),buf2(nosize,na2dch)
integer*4 ichan,iretlcode,sys$qiow,runno,pass,head14(ndim/4)
dimension headfp(ndim/4)
external ios$_readblk,ios$_rewind,ios$_skipfile,ios$_skiprecord
equivalence (headb,head12,head14,headfp)

nchan=na2dch !number of channels
nwords=nchan*nisize !number of i*2 words/block
nbytes=nwords*2 !number of bytes/block
nout=nosize !samples in output buffer/chan

c----find out what we are processing

print *, 'enter run identification number'
read *, runno

print *, 'enter pass number'
read *, pass
igroup=runno-(runno/10*10)

1 print *, 'enter down sample increment, save every?'
read *, ipick
if(ipick.lt.0) go to 1

2 print *, 'enter number of records to look at?'
read *, nrecmax
if(nrecmax.lt.0) go to 2
nsampmax=nrecmax*nisize

3 print *, 'enter recat start sample number?'
read *, irecat
if(irecat.lt.1) go to 3
istartrec=irecat/nisize
istartsamp=irecat-istartrec*nisize

4 print *, 'enter maximum number of samples?'
read *, nsampmax

5 print *, 'enter recat start sample number?'
read *, irecat
if(irecat.lt.1) go to 5

---check on pass and group, setup accordingly

if(pas$eq.1) then
if(igroup.eq.1) then!
output buffer counter
F

NSAMPLES = 1

ELSE
IOLDRUN = RUNNO - 1
ENCOD(9, 10, FILNAM) IOLDRUN, PASS

10

! GET THE DATA TO APPEND

FORMAT ( 'R', I2, 'P', I1, ' .EXT')
FILNAM(10) = 0
OPEN(UNIT = 10, STATUS = 'OLD', FILE = FILNAM, FORM = 'UNFORMATTED')
READ(10) NEXTST, NSIZE, LASTIPICK
READ(10) NLEFT, ((BUF2(I, J), I = 1, NLEFT), J = 1, NA2DCH)
CLOSE(UNIT = 10)

! START SAMPLE
N = NLEFT

NSAMPLES = NLEFT * IPICK + IPICK

IF (NSIZE .NE. NOSIZE) STOP 'OUTPUT BUFFER MISMATCH'
IF (LASTIPICK .NE. IPICK) STOP 'DOWNSAMPLE INCREMENT MISMATCH'
ENDIF
ENDIF
C

IF (PASS .EQ. 2) THEN

C

! PASS 2

C

IF (IGROUP .EQ. 1) THEN

N = 0

NSAMPLES = 1

ELSE
IOLDRUN = RUNNO - 1
ENCOD(9, 10, FILNAM) IOLDRUN, PASS

FILNAM(10) = 0
OPEN(UNIT = 10, STATUS = 'OLD', FILE = FILNAM, FORM = 'UNFORMATTED')
READ(10) NEXTST, NSIZE, LASTIPICK
READ(10) NLEFT, ((BUF2(I, J), I = 1, NLEFT), J = 1, NA2DCH)
CLOSE(UNIT = 10)

! START SAMPLE
N = NLEFT

NSAMPLES = NLEFT * IPICK + IPICK

IF (NSIZE .NE. NOSIZE) STOP 'OUTPUT BUFFER MISMATCH'
IF (LASTIPICK .NE. IPICK) STOP 'DOWNSAMPLE INCREMENT MISMATCH'
ENDIF
ENDIF
C

NSAMPMAX = NSAMPMAX + NSAMPLES - 1
PRINT *, 'LEFT OVER SAMPLES ', NSAMPLES
PRINT *, 'NUMBER OF RECORDS TO PROCESS ', NRECMAX
PRINT *, 'FINAL MAXIMUM NUMBER OF SAMPLES ', NSAMPMAX
C

C----- ASSIGN THE 9 TRACK TAPE DRIVE (6250) AND MAKE SURE IT IS REWOUND
C

CALL SYSSASSIGN('MAG_TAPE', ICHAN, ,)
IRETCODE = SYSSQIOW(,%VAL(ICHAN), IOS_REWIND, IOSB, , , , , , , , ,)
IF (.NOT. IRETCODE) STOP 'REWIND ERROR'
C

C----- SKIP FILES IF REQUESTED
C

PRINT *, 'HOW MANY FILES DO YOU WANT TO SKIP?'
READ *, NSKIP
IF (NSKIP .LT. 0) THEN
PRINT *, 'YOU MUST ONLY GO IN THE FORWARD DIRECTION'
PRINT *, 'THE NUMBER OF FILES MUST BE POSITIVE'
GOTO 4

H-4
_DRBO:[MACIE.TM]DOWNSAMP.FOR;58

END IF
IRETCODE=SYS$QIOW(,%VAL(ICHAN),IOS$_SKIPFILE,IOSB,,,%VAL(NSKIP),,,,)
IF (.NOT.IRETCODE) STOP 'SKIP FILE ERROR'
IF(IOSB(2).NE.NSKIP) THEN
  PRINT 20, (IOSB(I),I=1,4)
  STOP 'SKIPPING FILE ERROR'
END IF
C
C-----READ IN THE HEADER AND PRINT OUT
C
IRETCODE=SYS$QIOW(,%VAL(ICHAN),IOS$_READLBLK,IOSB,,,HEADB,%VAL(NBYTES),,,)
1
IF (.NOT.IRETCODE) STOP 'READING HEADER ERROR'
IF(IOSB(2).NE.NBYTES) THEN
  PRINT 20, (IOSB(I),I=1,4)
  STOP 'WRONG NUMBER OF BYTES READ FOR HEADER'
ENDIF
CALL LISTHEAD(HEADB,HEADI2,HEADI4)
HEADFP(15)=HEADFP(15)*FLOAT(IPICK) !UPDATE SAMPLE INTERVAL
C
C------SKIP THE APPROPRIATE NUMBER OF RECORDS
C
NSKIP=ISTARTREC
IF(NSKIP.GT.0) THEN
  IRETCODE=SYS$QIOW(,%VAL(ICHAN),IOS$_SKIPRECORD,IOSB,,,HEADB,%VAL(NSKIP),,,)
1
  IF (.NOT.IRETCODE) STOP 'SKIP RECORD ERROR'
  IF(IOSB(2).NE.NSKIP) THEN
    PRINT 20, (IOSB(I),I=1,4)
    STOP 'SKIPPING RECORD ERROR'
  END IF
END IF
NRECIN=NSKIP !NUMBER OF INPUT RECORDS (TOTAL)
NRECOUT=0 !NUMBER OF OUTPUT RECORDS
ISAVE=ISTARTSAMP !NEXT POINT IN BUFFER TO SAVE
IF(ISAVE.GT.NISIZE) STOP 'STARTING POINT > RECORD SIZE'
C
C-----OPEN THE OUTPUT FILE, R__P_.DAT
C
ENCE(9,30,FILNAM) RUNNO,PASS
30 FORMAT( 'R' ,12, 'P' ,11,'.DAT')
FILNAM(IO)=0
OPEN(UNIT=11,STATtS='NEW',FILE=FILNAM,FORM='UNFORMATTED')
PRINT 40, (FILNAM(J),J=1,9)
:0 FORMAT (2X,'OUTPUT FILE: ',9Al)
WRITE(11) (HEADB(I),I=1,NBYTES)
C
C------READ THE DATA IN A RECORD AT A TIME, LOOK FOR THE EOF
C
5 CONTINUE
IRETCODE=SYS$QIOW(,%VAL(ICHAN),IOS$_READLBLK,IOSB,,,BUFI,%VAL(NBYTES),,,)
1
IF (.NOT.IRETCODE) STOP 'ERROR READING DATA RECORD'
IF(IOSB(1).EQ.'0870-X') THEN
  PRINT *, 'AT EOF'
  NSAMPLELEFT=IPICK*N+IPICK-ISAVE
  GO TO 7
ENDIF
IF (IOSB(2).NE.NBYTES) THEN
  PRINT 20, (IOSB(I),I=1,4)
  PRINT *, 'PROBLEM SKIPPING ',NSKIIP, ' RECORDS'
  STOP 'WRONG NUMBER OF BYTES READ FOR DATA'
ENDIF
NRECIN=NRECIN+1

C-----EXTRACT THE DATA POINTS I WANT TO SAVE AND DUMP FULL BUFFERS
C
6  CONTINUE
  N=N+1
  DO J=1,NA2DCH
    BUF2(N,J)=BUF1(J,ISAVE)
  END DO
  IF(N.EQ.NOUT) THEN
    WRITE(11) ((BUF2(I,J),I=1,NOUT),J=1,NA2DCH)
    NRECOUT=NRECOUT+1
    N=0
  ENDIF
  ISAVE=ISAVE+IPICK
  NSAMPLES=NSAMPLES+IPICK
  IF (NSAMPLES.GT.NSAMPMAX) THEN
    ISAVE=NSAMPLES-NSAMPMAX
    NSAMPLEFT=N*IPICK+(IPICK-ISAVE)
    NSAMPLES=NSAMPMAX
    GO TO 7
  ENDIF
  IF(ISAVE.LE.NISIZE) GO TO 6
  ISAVE=ISAVE-NISIZE
  GO TO 5

C-----AT EOF FILE, CLOSE FILES
C
7  CONTINUE
  CLOSE(UNIT=10)
  CLOSE(UNIT=11)
  NLEFT=N

C-----SAVE THE EXTRA DATA FOR APPENDING TO THE FRONT OF THE NEXT GROUP
C
ENC()E(9,50,FILNAM) RUNNO,PASS
50  FORMAT('R',I2,'P',I1,'.EXT')
FILNAM(10)=0
OPEN(UNIT=10,STATUS='NEW',FILE=FILNAM,FORM='UNFORMATTED')
WRITE(10) ISAVE,NOUT,IPICK !NEXT START, SIZE, INC
WRITE(10) NLEFT,((BUF2(I,J),I=1,NLEFT),J=1,NA2DCH)
CLOSE(UNIT=10)

C-----PRINT THE BOOK KEEPING INFORMATION
C
PRINT *, 'READ IN ',NRECIN, ' RECORDS (SKIPPED ',NSKIIP,')'
PRINT *, 'WROTE OUT ',NRECOUT, ' RECORDS'
PRINT *, 'BUFFER SIZES ARE: IN= ',INOUT,' OUT= ',NOUT
PRINT *, 'NUMBER OF SAMPLES LOOKED AT: ',NSAMPLES
PRINT 60, (FILNAM(I),I=1,9)
60  FORMAT(2X/2X'EXTRA FILE: ',9A7)
PRINT *, 'NEXT START OFFSET IS: ',ISAVE
PRINT *, 'NUMBER OF LEFT OVER POINTS TO BE APPENDED: ', NLEFT
PRINT *, '
PRINT *, 'NUMBER OF LEFT OVER SAMPLES: ', NSAMPLEFT
PRINT *, '
PRINT *, '***DOWNSAMPLE PROGRAM COMPLETE ***'
CALL EXIT
END
LISTHEAD.FOR  LISTS OUT THE HEADER INFORMATION FOR A
SPECIFIED TAPE.

C HEADB(1-2)  HEADI2(1) ANALOG TAPE NUMBER
C HEADB(3)  HEADI2(2) PASS NUMBER (1, 2 OR 3)
C HEADB(4)  " NUMBER OF CHANNELS 14 OR 24
C HEADB(5)  HEADI2(3) DIGITIZING MONTH
C HEADB(6)  " DIGITIZING DAY
C HEADB(7)  HEADI2(4) DIGITIZING YEAR
C HEADB(8)  " EXPERIMENT TIME CODE START HOUR
C HEADB(9)  HEADI2(5) EXPERIMENT TIME CODE START MINUTE
C HEADB(10)  " EXPERIMENT TIME CODE START SECONDS
C HEADB(11-30) HEADI2(6-15) EVENT NAME (20 CHAR MAXIMUM)
C HEADB(31)  HEADI2(16) ANALOG CHANNEL IN POSITION 1
C HEADB(32)  " ANALOG CHANNEL IN POSITION 2
C HEADB(33)  HEADI2(17) ANALOG CHANNEL IN POSITION 3
C HEADB(34)  " ANALOG CHANNEL IN POSITION 4
C HEADB(35)  HEADI2(18) ANALOG CHANNEL IN POSITION 5
C HEADB(36)  " ANALOG CHANNEL IN POSITION 6
C HEADB(37)  HEADI2(19) ANALOG CHANNEL IN POSITION 7
C HEADB(38)  " ANALOG CHANNEL IN POSITION 8
C HEADB(39)  HEADI2(20) ANALOG CHANNEL IN POSITION 9
C HEADB(40)  " ANALOG CHANNEL IN POSITION 10
C HEADB(41)  HEADI2(21) ANALOG CHANNEL IN POSITION 11
C HEADB(42)  " ANALOG CHANNEL IN POSITION 12
C HEADB(43)  HEADI2(22) ANALOG CHANNEL IN POSITION 13
C HEADB(44)  " ANALOG CHANNEL IN POSITION 14
C HEADB(45)  HEADI2(23) ANALOG CHANNEL IN POSITION 15
C HEADB(46)  " ANALOG CHANNEL IN POSITION 16
C HEADB(47)  HEADI2(24) ANALOG CHANNEL IN POSITION 17
C HEADB(48)  " ANALOG CHANNEL IN POSITION 18
C HEADB(49)  HEADI2(25) ANALOG CHANNEL IN POSITION 19
C HEADB(50)  " ANALOG CHANNEL IN POSITION 20
C HEADB(51)  HEADI2(26) ANALOG CHANNEL IN POSITION 21
C HEADB(52)  " ANALOG CHANNEL IN POSITION 22
C HEADB(53)  HEADI2(27) ANALOG CHANNEL IN POSITION 23
C HEADB(54)  " ANALOG CHANNEL IN POSITION 24
C HEADB(55)  HEADI2(28) RUN IDENTIFICATION NUMBER
C HEADB(56)  " "
C HEADB(57-60) HEADFP(15) SAMPLE RATE (2.5 KHZ=.0004SEC)

SUBROUTINE LISTHEAD(HEADB,HEADI2,HEADI4)

BYTE HEADB(1)
INTEGER*2 HEADI2(1)
INTEGER*4 HEADI4(1)

C --- PRINT OUT THE ALL HEADER INFO
C
PRINT 1, (HEADB(1),I=11,30)
1 FORMAT(X,'*** EVENT DESCRIPTION: ',20A1,'***')
C
PRINT 2, HEADI2(1)
2 FORMAT(X/X,' ANALOG TAPE NUMBER: ',4X,14)
C
PRINT 3, HEADI2(28)
FORMAT(X,' RUN IDENTIFICATION NUMBER: ',6X,I2)
C
PRINT 4, HEADB(3)  
FORMAT(X,' PASS NUMBER: ',7X,I1)
C
PRINT 5, HEADB(8),HEADB(9),HEADB(10)
FORMAT(X,' START TIME CODE: ',I2,' : ',I2,' : ',I2)
C
PRINT 6, HEADB(5),HEADB(6),HEADB(7)
FORMAT(X,' DATE: ',I2,' / ',I2,' / ',I2)
C
NCHAN=HEADB(4)
C
PRINT 7
FORMAT(X/X,' POSITION : CHANNEL  POSITION : CHANNEL', 
X/X,' ------------------ ---------------------')
1
NHALF=NCHAN/2
DO I=1,NHALF
  K=I+NHALF
  PRINT 8, I,HEADB(I+30),K,HEADB(K+30)
8 FORMAT(X,4X,I2,3X,' : ',3X,I2,2X,5X,3X,I2,3X,' : ',3X,I2)
END DO
C
RETURN
END
APPENDIX I

MERGEMT.FOR
LISTHEAD.FOR
MERGEMT.FOR MERGES TOGETHER DIGITIZED DATA FROM PASSES 1 AND 2 AND
WRITES THESE TO TAPE. REDUNDANT CHANNELS ARE DISCARDED.
A FOREIGN MAGTAPE IS PRODUCED WITH ONE IDENTIFICATION
HEADER RECORD FOLLOWED BY DATA RECORDS, 1024*24 WORDS LONG.
THE DATA ORDER IS 1024 SAMPLES FOR EACH CHANNEL 1-24.
THE A/D TAPE CHANNEL FOR EACH POSITION IS FOUND IN
THE HEADER, THE ORDER IS 1,2,3,4,5,6,7,8,9,10,11,12,
14,15,16,17,18,19,20,21,22,23,24,25 (NO 13).

1) LOG ON TO VAX
2) MOUNT A TAPE WITH A WRITE RING ON MFA0:
3) EXECUTE THE MERGEMT.COM COMMAND FILE, THIS WILL:
   A. INITIALIZE THAT TAPE TO 6250 BPI (NAME R__)
   B. OPEN THE TWO DOWNSAMPLED FILES FOR PASSES 1 AND 2
   C. READ AND MERGE THE TWO HEADER RECORDS
   D. WRITE THE NEW HEADER RECORD TO TAPE
   E. FOR ALL DATA RECORDS: READ AND MERGE ALL DATA RECORDS
      AND WRITE THEM TO TAPE
   F. MAKE SURE BOTH FILES AT THE SAME POINT
   G. CLOSE ALL FILES
   H. WRITE EOF MARKINGS ON OUTPUT TAPE
   I. DISMOUNT THE TAPE

PARAMETER NADC=24 !NUMBER OF FINAL A/D CHANNELS USED
PARAMETER NBSIZE=1024 !NUMBER OF WORDS IN ONE BUFFER/CHANNEL
PARAMETER NDIM=14*500*2,NDIM2=NDIM/2,NDIM4=NDIM/4

BYTE HBPI(NDIM),HBP2(NDIM),HBM(NDIM),FILNAM(10),EVENTDESC(20)
INTEGER*2 IOSBMT(4)
INTEGER*2 HI2P1(NDIM2),HI2P2(NDIM2),HI2M(NDIM2)
INTEGER*2 BUFPI(NBSIZE,14),BUFPI(NBSIZE,14),BUFM(NBSIZE,24)
INTEGER*4 HI4P1(NDIM4),HI4P2(NDIM4),HI4M(NDIM4)
INTEGER*4 SYSSQIOW,WRINORET,WEOFNORET,RUNNO
DIMENSION HFPI(NDIM4),HFPI(NDIM4),HFPI(NDIM4)
EQUIVALENCE (HBPI,HI2P1,HI4P1,HI4P1), (HBP2,HI2P2,HI4P2,HI4P2)
EQUIVALENCE (HBM,HI2M,HI4M,HI4M)
EQUIVALENCE (BUFPI,BUFP1,BUFP2,BUFP2,BUFP2)
FORMAT('R',12,'P2.FIL')
FILNAM(10)=0
OPEN(UNIT=12,STATUS='OLD',FILE=FILNAM,FORM='UNFORMATTED')
READ(12) HBP2
CALL LISTHEAD(HBP2,H12P2,H14P2)

C-----FORM THE NEW HEADER
C
IF(H12P1(28).NE.H12P2(28)) STOP 'DIFFERENT RUNS'
IF(HBP1(3).NE.1) STOP 'FIRST FILE IS NOT PASS 1'
IF(HBP2(3).NE.2) STOP 'SECOND FILE IS NOT PASS 2'
IF(HBP1(61).NE.1) STOP 'FIRST FILE IS NOT FILTERED'
DO I=1,60
   H12M(I)=H12P1(I)
END DO
IF(H12P1(1).NE.H12P2(1)) THEN
   PRINT *, 'TAPE NUMBERS DO NOT MATCH, ENTER TAPE NUMBER:'
   READ *, H12M(1)
ENDIF
HBM(3)=3 !MERGED CASE IS PASS 3
HBM(4)=24 !ALL 24 CHANNELS
PRINT *, 'ENTER MERGING DATE: MONTH,DAY'
READ *, HBM(5),HBM(6)
ENCODE(20,30,EVENTDESC) RUNNO
FORM(2X,'RUN',12)
DO I=1,20
   HBM(I+10)=EVENTDESC(I)
END DO
N=45
DO I=1,14
   IF(I.EQ.2.OR.I.EQ.4.OR.I.EQ.10.OR.I.EQ.13) GO TO 1
   HBM(N)=HBP2(1+30)
   N=N+1
1 CONTINUE
END DO
CALL LISTHEAD(HBM,H12M,H14M)

C-----ASSIGN THE TAPE TO LOGICAL NAME MAG_TAPE AND REWIND
C
CALL SYS$ASSIGN('MAG_TAPE',ICHAN,)
IRETCODE=SYS$QICW(),%VAL(ICHAN),IO$_REWIND,IOSBMT,++++,
IF(.NOT.IRETCODE) THEN
   IF( IOSBMT(1).EQ.'1A4'X) STOP 'TAPE DRIVE IS OFFLINE'
   PRINT 40, (IOSBMT(I),:=1,4)
40 FORMAT(2X,'IOSB: ','4(X,Z4.4))
   STOP 'MAG_TAPE REWIND ERROR'
ENDIF

C-----SKIP FILES IF REQUESTED
C
PRINT *, 'HOW MANY FILES DO YOU WANT TO SKIP?'
READ *, NSKIP
IF(NSKIP.LT.0) THEN
   PRINT *, 'YOU MUST ONLY GO IN THE FORWARD DIRECTION'
   PRINT *, 'THE NUMBER OF FILES MUST BE POSITIVE'
   GOTO 2
END IF
IRETCODE=SYS$QIOW(,%VAL(ICHAN),IOS_SKIPFILE,IOSBMT,,,%VAL(NSKIP),,,)
IF (.NOT.IRETCODE) STOP 'SKIP FILE ERROR'
IF(IOSBMT(2).NE.NSKIP) THEN
  PRINT 40, (IOSBMT(I),I=1,4)
  PRINT *, 'PROBLEM SKIPPING ',NSKIP, ' FILES'
  STOP 'SKIPPING FILE ERROR'
END IF

C-----DUMP THE HEADER TO TAPE

NBYTES=NDIM
IRETCODE=SYS$QIOW(,%VAL(ICHAN),IOS_WRITEBLK,IOSBMT,
  1 ,,HBM,%VAL(NBYTES),,,,)
IF(.NOT.IRETCODE) STOP 'ERROR MAGTAPE DUMPING HEADER'
NREC=1 !RECORD 1 IS HEADER
PRINT *, 'DUMPED HEADER IN ',NREC

C-----READ IN DATA FOR EACH PASS, SHIFT DOWN PASS 2 NEW CHANNELS

NBYTES=NPT*24*2 !TOTAL #BYTES PER RECORD
READ(11,END=4) ((BUFP1(I,J),I=1,NPT),J=1,14)
READ(12,END=7) ((BUFP2(I,J),I=1,NPT),J=1,14)
N=15
DO J=1,14
  IF(J.NE.2.AND.J.NE.4.AND.J.NE.10.AND.J.NE.13) THEN
    DO I=1,NPT
      BUFM(I,N)=BUFP2(I,J)
    END DO
    N=N+1
  ENDIF
END DO

C-----DUMP THIS BUFFER TO TAPE

IRETCODE=SYS$QIOW(,%VAL(ICHAN),IOS_WRITEBLK,IOSBMT,
  1 ,,BUFM,%VAL(NBYTES),,,,)
IF(.NOT.IRETCODE) THEN
  PRINT *, IRETCODE
  STOP 'MAG TAPE WRITE ERROR'
ENDIF
IF(IOSBMT(1).NE.1) THEN
  PRINT 40, (IOSBMT(I),I=1,4)
END IF
NREC=NREC+1 !INCREMENT RECORD NUMBER
PRINT *, 'DUMPED RECORD ',NREC,NBYTES
IF(IOSBMT(1).EQ.'0878'X) PRINT *, 'SENSED EOT MARKER AT ',NREC
GO TO 3

C-----AT EOF CHECK THE OTHER FILE

CONTINUE
PRINT *, 'AT EOF FOR PASS 1 FILE'
NLEFT=0
READ(12,END=6) ((BUFP2(I,J),I=1,NPT),J=1,14)
NLEFT=NLEFT+1
GO TO 5
CONTINUE
PRINT *, 'AT EOF FOR PASS 2 FILE'
IF(NLEFT.NE.0) PRINT *, 'READ ', NLEFT, ' MORE RECORDS'
GO TO 11
C
7 CONTINUE
PRINT *, 'AT EOF FOR PASS 2 FILE'
NLEFT=1
8 READ(11,END=9) ((BUFP2(I,J), I=1,NPT), J=1,14)
NLEFT=NLEFT+1
GO TO 8
9 CONTINUE
PRINT *, 'AT EOF FOR PASS 1 FILE'
IF(NLEFT.NE.0) PRINT *, 'READ ', NLEFT, ' MORE RECORDS'
C
C-----WRITE TWO EOF MARKERS ON TAPE
C
11 CONTINUE
IRETCODE=SYSSQIOW(,%VAL(ICHAN), IO$ _WRITEOF, IOSBMT, , , , , , , , )
IRETCODE=SYSSQIOW(,%VAL(ICHAN), IO$ _WRITEOF, IOSBMT, , , , , , , )
C
CALL EXIT
END
LISTHEAD.FOR LISTS OUT THE HEADER INFORMATION FOR A SPECIFIED TAPE.

C
HEADB(1-2) HEADI2(1) ANALOG TAPE NUMBER
HEADB(3) HEADI2(2) PASS NUMBER (1, 2 OR 3)
HEADB(4) " NUMBER OF CHANNELS 14 OR 24
HEADB(5) HEADI2(3) DIGITIZING MONTH
HEADB(6) " DIGITIZING DAY
HEADB(7) HEADI2(4) DIGITIZING YEAR
HEADB(8) " EXPERIMENT TIME CODE START HOUR
HEADB(9) HEADI2(5) EXPERIMENT TIME CODE START MINUTE
HEADB(10) " EXPERIMENT TIME CODE START SECONDS
HEADB(11-30) HEADI2(6-15) EVENT NAME (20 CHAR MAXIMUM)
HEADB(31) HEADI2(16) ANALOG CHANNEL IN POSITION 1
HEADB(32) " ANALOG CHANNEL IN POSITION 2
HEADB(33) HEADI2(17) ANALOG CHANNEL IN POSITION 3
HEADB(34) " ANALOG CHANNEL IN POSITION 4
HEADB(35) HEADI2(18) ANALOG CHANNEL IN POSITION 5
HEADB(36) " ANALOG CHANNEL IN POSITION 6
HEADB(37) HEADI2(19) ANALOG CHANNEL IN POSITION 7
HEADB(38) " ANALOG CHANNEL IN POSITION 8
HEADB(39) HEADI2(20) ANALOG CHANNEL IN POSITION 9
HEADB(40) " ANALOG CHANNEL IN POSITION 10
HEADB(41) HEADI2(21) ANALOG CHANNEL IN POSITION 11
HEADB(42) " ANALOG CHANNEL IN POSITION 12
HEADB(43) HEADI2(22) ANALOG CHANNEL IN POSITION 13
HEADB(44) " ANALOG CHANNEL IN POSITION 14
HEADB(45) HEADI2(23) ANALOG CHANNEL IN POSITION 15
HEADB(46) " ANALOG CHANNEL IN POSITION 16
HEADB(47) HEADI2(24) ANALOG CHANNEL IN POSITION 17
HEADB(48) " ANALOG CHANNEL IN POSITION 18
HEADB(49) HEADI2(25) ANALOG CHANNEL IN POSITION 19
HEADB(50) " ANALOG CHANNEL IN POSITION 20
HEADB(51) HEADI2(26) ANALOG CHANNEL IN POSITION 21
HEADB(52) " ANALOG CHANNEL IN POSITION 22
HEADB(53) HEADI2(27) ANALOG CHANNEL IN POSITION 23
HEADB(54) " ANALOG CHANNEL IN POSITION 24
HEADB(55) HEADI2(28) RUN IDENTIFICATION NUMBER
HEADB(56) " "
HEADB(57-60) HEADFP(15) SAMPLE RATE (2.5 KHZ=.0004SEC)

SUBROUTINE LISTHEAD(HEADB,HEADI2,HEADI4)

BYTE HEADB(1)
INTEGER*2 HEADI2(1)
INTEGER*4 HEADI4(1)

PRINT OUT THE ALL HEADER INFO

PRINT 1, (HEADB(I),I=11,30)
FORMAT(X,'*** EVENT DESCRIPTION: ',20A1,' ***')

PRINT 2, HEADI2(1)
FORMAT(X/X,' ANALOG TAPE NUMBER: ',4X,14)

PRINT 3, HEADI2(28)
FORMAT(X,'RUN IDENTIFICATION NUMBER: ',6X,I2)

PRINT 4, HEADB(3)
FORMAT(X,'PASS NUMBER: ',7X,I1)

PRINT 5, HEADB(8),HEADB(9),HEADB(10)
FORMAT(X,'START TIME CODE: ',I2,' : ',I2,' : ',I2)

PRINT 6, HEADB(5),HEADB(6),HEADB(7)
FORMAT(X,'DATE: ',I2,' / ',I2,' / ',I2)

NCHAN=HEADB(4)

PRINT 7
FORMAT(X/X,'POSITION : CHANNEL  POSITION : CHANNEL',X/X,'----------------  ----------------')

NHALF=NCHAN/2
DO I=1,NHALF
  K=I+NHALF
  PRINT 8, I,HEADB(I+30),K,HEADB(K+30)
END DO
FORMAT(X,4X,I2,3X,' : ',3X,I2,2X,5X,3X,I2,3X,' : ',3X,I2)
RETURN
END
APPENDIX J

READMT.FOR
ASSMT.FOR
RBLOCKMT.FOR
LISTHEAD.FOR
READMT.FOR  READS A FOREIGN MERGED DIGITIZED MAGNETIC TAPE.
THE FIRST RECORD CONTAINS HEADER IDENTIFICATION.
THE DATA IS GROUPED IN RECORDS OF 1024 PTS EACH
FOR ALL 24 CHANNELS (1-25, NO 13). THE FINAL DATA
IS WRITTEN TO DISK IN SEPARATE FILES FOR EACH CHANNEL.

PARAMETER NA2DCH=24 !NUMBER OF CHANNELS
PARAMETER NSIZE=1024 !NUMBER OF WORDS/CHANNEL IN A RECORD
PARAMETER NDIM=14*500*2

BYTE FILNAM(13),HEADB(NDIM),BUFFER(32768)
INTEGER*2 HEADI2(NDIM/2),I2BUF(NSIZE,24),IOSB(4)
INTEGER*4 ICHAN,CHANNO,RUNNO,HEADI4(NDIM/4)
DIMENSION IFLG(24),ISAVE(24)

EQUIVALENCE (HEADB,HEADI2,HEADI4),(BUFFER,I2BUF)
EQUIVALENCE (HEADI4(15),SAMP_INT)

NCHAN=NA2DCH !NUMBER OF CHANNELS
NWORDS=NCHAN*NSIZE !NUMBER OF I*2 WORDS/BLOCK
NBYTES=NWORDS*2 !NUMBER OF BYTES/BLOCK

C-----FIND OUT WHAT WE ARE RETREIVEING

PRINT *, 'ENTER RUN IDENTIFICATION NUMBER'
READ *, RUNNO
PRINT *, 'HOW MANY CHANNELS DO YOU WANT TO SAVE? (-1=ALL)'
READ *, NSAVE
IF(NSAVE.EQ.-1) THEN
  NSAVE=24
  N=1
  DO 1=1,24
    ISAVE(I)=N
    N=N+1
    IF(N.EQ.13) N=N+1
  END DO
ELSE
  PRINT *, 'ENTER ANALOG CHANNEL NUMBERS'
  READ *, (ISAVE(I),I=1,NSAVE)
ENDIF
PRINT *, 'HOW MANY RECORDS DO YOU WANT TO READ?'
READ *, NRECORDS

C-----ASSIGN THE MAGTAPE

IFILE=1
CALL ASSMT(ICHAN,IFILE)

C-----READ IN THE HEADER AND PRINT OUT

CALL RBLOCKMT(ICHAN,NPTS,HEADI2,ISTAT)
IF(ISTAT.EQ.2) STOP 'AT EOF'
IF(ISTAT.EQ.1) STOP 'TAPE READ ERROR'
IF(NPTS.NE.NDIM/2) STOP 'WRONG NUMBER OF BYTES READ FOR HEADER'
CALL LISTHEAD(HEADB,HEADI2,HEADI4)

C-----OPEN OUTPUT FILES, CHANNELS WILL SEPERATED INTO R__C__.DAT
DO I=1,NCHAN
   CHANNO=HEADB(30+I)
   IUNIT=10+I
   IFLAG(I)=0
   DO J=1,NSAVE
      IF(ISAVE(J).EQ.CHANNO) IFLAG(I)=1
   END DO
   IF(IFLAG(I).EQ.1)
      ENCODE(12,10,FILNAM) RUNNO,CHANNO
      FORMAT('R',I2,'C',I2,'.DAT')
      FILNAM(13)=Q
      OPEN(UNIT-IUNIT,STATUS='NEW',FILE=FILNAM,FORM='UNFORMATTED')
      PRINT 20, IUNIT,(FILNAM(J),J=1,12)
   20   FORMAT(2X,'OUTPUT UNIT= ',15,2X,'FILE '12A1)
   WRITE(IUNIT) RUNNO,CHANNO,NWORDS,NRECORDS,SAMP_INT
   END IF
END DO

C-----READ THE DATA IN

DO NR=1,NRECORDS
   CALL RBLOCKMT(ICHAN,NPTS,BUFFER,ISTAT)
   IF(ISTAT.EQ.2) STOP 'AT EOF'
   IF(ISTAT.EQ.1) STOP 'TAPE READ ERROR'
   IF(NPTS.NE.NWORDS) STOP 'WRONG NUMBER OF I*2 READ IN'
   DO I=1,NCHAN
      IF(IFLAG(I).EQ.1)
         IUNIT=10+I
         WRITE(IUNIT) (12BUF(J,1),J=1,NSIZE)
   END IF
   END DO
END DO

C-----CLOSE THE FILE

DO I=1,NCHAN
   IUNIT=10+I
   CLOSE(IUNIT)
END DO

CALL EXIT
END
ASSMT.FOR ASSIGNS A CHANNEL FOR THE MAGNETIC TAPE DRIVE.
IT MUST BE EXTERNALLY ASSIGNED TO THE LOGICAL NAME MAG_TAPE. THE TAPE IS REWOUND AND THE REQUESTED NUMBER OF FILES IS SKIPPED.

PARAMETERS:
ICHAN = MAGTAPE INPUT CHANNEL NUMBER
IFILE = MAGTAPE INPUT FILE NUMBER

SUBROUTINE ASSMT(ICHAN, IFILE)

EXTERNAL IO$REWIND, IO$SKIPFILE
INTEGER*2 IOSB(4), EOF, EOT
INTEGER*4 SYS$ASSIGNSYS$QIOW, ICHAN, IRET

EOF='0870'X !END OF FILE
EOT='0878'X !END OF TAPE
NORMAL='0001'X !NORMAL

----ASSIGN THE 9 TRACK TAPE DRIVE "MAG_TAPE" AND ASSIGN THE CHANNEL
IRET=SYS$ASSIGN('MAG_TAPE', ICHAN, ,)
IF (.NOT. IRET) STOP 'ASSIGN ERROR'

----REWIND THE TAPE
IRET=SYS$QIOW(,%VAL(ICHAN), IO$REWIND, IOSB, , , , , )
IF (.NOT. IRET) STOP 'REWIND ERROR'
IF (IOSB(1).NE.NORMAL) STOP 'REWIND ERROR'
NFILE=1 !PRESENT FILE NUMBER

----SKIP THE REQUESTED NUMBER OF FILES
NSKIP=IFILE-1 !NUMBER OF FILES TO SKIP
IF(NSKIP.NE.0) THEN
   IF(NSKIP.LT.0) NSKIP=NSKIP-1
   IRET=SYS$QIOW(,%VAL(ICHAN), IO$SKIPFILE, IOSB, , , , %VAL(NSKIP), , , , )
   IF (.NOT. IRET) STOP 'SKIPFILE ERROR'
   IF (IOSB(1).NE.NORMAL.AND.IOSB(1).NE.EOF) STOP 'SKIPFILE ERROR'
   IF (IOSB(2).NE.ABS(NSKIP)) STOP 'SKIPFILE ERROR'
   NFILE=NFILE+NSKIP !NEW FILE NUMBER
ENDIF

ENDIF

IFILE=NFILE !PRESENT MAGTAPE INPUT FILE NUMBER

RETURN
END
RBLOCKMT.FOR  READS A BLOCK OF DATA FROM A MAGNETIC TAPE.

PARAMETERS:
ICHAN = CHANNEL ASSIGNED TO THE MAG TAPE UNIT
NWORDS = NUMBER OF 1*2 DATA POINTS READ
I2BUF = BUFFER TO RECEIVE DATA
ISTAT = RETURN STATUS (0=OK, 1=ERROR, 2=EOF)

SUBROUTINE RBLOCKMT(ICHAN,NWORDS,I2BUF,ISTAT)

EXTERNAL IOS$ READLBLK
INTEGER*4 SYS$QIOW, ICHAN, IRET

INTEGER*2 I2BUF(1), IOSB(4), NORMAL, EOF, EOT, BUFFER(32768)

EOF='0870'X ! END OF FILE
EOT='0878'X ! END OF TAPE
NORMAL='0001'X ! NORMAL
ISTAT=0 ! ASSUME OK

C ---- READ IN ONE BLOCK OF DATA

IRET=SYS$QIOW(,%VAL(ICHAN),IO$ READLBLK,IOSB,,, 1
BUFFER,%VAL(65535),,,,)

IF(.NOT.IRET) THEN
  PRINT *, 'IRETURN CODE = ',IRET
  PRINT 100, (IOSB(I), I=1,4)
  STOP 'READING DATA ERROR'
ENDIF

NWORDS=0
CALL MVBITS(IOSB(2),0,16,NWORDS,0)
NWORDS=NWORDS/2

IF(IOSB(1).NE.NORMAL) THEN
  IF(IOSB(1).EQ.EOF) THEN
    ISTAT=2
    GO TO 200
  ENDIF
  IF(IOSB(1).EQ.EOT) THEN
    PRINT *, 'EOT REACHED'
    GO TO 200
  ENDIF
  ISTAT=1
  PRINT *, 'PROBLEM WITH MAGTAPE'
  PRINT 100, (IOSB(I), I=1,4)
  100 FORMAT(2X,4(Z4.4,2X))
  GO TO 200
ENDIF

ISTAT=1
PRINT *, 'PROBLEM WITH MAGTAPE'
PRINT 100, (IOSB(I), I=1,4)
100 FORMAT(2X,4(Z4.4,2X))
GO TO 200
ENDIF

CONTINUE
IF(NWORDS.NE.0) THEN
  DO I=1,NWORDS
    I2BUF(I)=BUFFER(I)
  END DO
ENDIF
RETURN
END
LISTHEAD.FOR  LISTS OUT THE HEADER INFORMATION FOR A SPECIFIED TAPE.

C

C HEADB(1-2)  HEADI2(1)  ANALOG TAPE NUMBER
C HEADB(3)  HEADI2(2)  PASS NUMBER (1,2 OR 3)
C HEADB(4)  "  NUMBER OF CHANNELS 14 OR 24
C HEADB(5)  HEADI2(3)  DIGITIZING MONTH
C HEADB(6)  "  DIGITIZING DAY
C HEADB(7)  HEADI2(4)  DIGITIZING YEAR
C HEADB(8)  "  EXPERIMENT TIME CODE START HOUR
C HEADB(9)  HEADI2(5)  EXPERIMENT TIME CODE START MINUTE
C HEADB(10)  "  EXPERIMENT TIME CODE START SECONDS
C HEADB(11-30)  HEADI2(6-15)  EVENT NAME (20 CHAR MAXIMUM)
C HEADB(31)  HEADI2(16)  ANALOG CHANNEL IN POSITION 1
C HEADB(32)  "  ANALOG CHANNEL IN POSITION 2
C HEADB(33)  HEADI2(17)  ANALOG CHANNEL IN POSITION 3
C HEADB(34)  "  ANALOG CHANNEL IN POSITION 4
C HEADB(35)  HEADI2(18)  ANALOG CHANNEL IN POSITION 5
C HEADB(36)  "  ANALOG CHANNEL IN POSITION 6
C HEADB(37)  HEADI2(19)  ANALOG CHANNEL IN POSITION 7
C HEADB(38)  "  ANALOG CHANNEL IN POSITION 8
C HEADB(39)  HEADI2(20)  ANALOG CHANNEL IN POSITION 9
C HEADB(40)  "  ANALOG CHANNEL IN POSITION 10
C HEADB(41)  HEADI2(21)  ANALOG CHANNEL IN POSITION 11
C HEADB(42)  "  ANALOG CHANNEL IN POSITION 12
C HEADB(43)  HEADI2(22)  ANALOG CHANNEL IN POSITION 13
C HEADB(44)  "  ANALOG CHANNEL IN POSITION 14
C HEADB(45)  HEADI2(23)  ANALOG CHANNEL IN POSITION 15
C HEADB(46)  "  ANALOG CHANNEL IN POSITION 16
C HEADB(47)  HEADI2(24)  ANALOG CHANNEL IN POSITION 17
C HEADB(48)  "  ANALOG CHANNEL IN POSITION 18
C HEADB(49)  HEADI2(25)  ANALOG CHANNEL IN POSITION 19
C HEADB(50)  "  ANALOG CHANNEL IN POSITION 20
C HEADB(51)  HEADI2(26)  ANALOG CHANNEL IN POSITION 21
C HEADB(52)  "  ANALOG CHANNEL IN POSITION 22
C HEADB(53)  HEADI2(27)  ANALOG CHANNEL IN POSITION 23
C HEADB(54)  "  ANALOG CHANNEL IN POSITION 24
C HEADB(55)  HEADI2(28)  RUN IDENTIFICATION NUMBER
C HEADB(56)  "  
C HEADB(57-60)  HEADFP(15)  SAMPLE RATE (2.5 KHZ=.0004SEC)

SUBROUTINE LISTHEAD(HEADB,HEADI2,HEADI4)

BYTE HEADB(1)
INTEGER*2 HEADI2(1)
INTEGER*4 HEADI4(1)

C -------PRINT OUT THE ALL HEADER INFO
C
PRINT 1, (HEADB(I),I=11,30)
1 FORMAT(X,'*** EVENT DESCRIPTION: ',20A1,'***')
C
PRINT 2, HEADI2(1)
2 FORMAT(X'X',' ANALOG TAPE NUMBER: ','4X,14)
C
PRINT 3, HEADI2(28)
 FORMAT(X,' RUN IDENTIFICATION NUMBER: ',6X,I2)
C
PRINT 4, HEADB(3)
4 FORMAT(X,' PASS NUMBER: ',7X,I1)
C
PRINT 5, HEADB(8),HEADB(9),HEADB(10)
5 FORMAT(X,' START TIME CODE: ',I2,:,:,I2,:,:,I2)
C
PRINT 6, HEADB(5),HEADB(6),HEADB(7)
6 FORMAT(X,' DATE: ',I2,'/',I2,'/',I2)
C
NCHAN=HEADB(4)
C
PRINT 7
7 FORMAT(X/X,' POSITION : CHANNEL          POSITION : CHANNEL',
      X/X,'-----------------------
      X/X,'-----------------------')
1
NH Alf=NCHAN/2
DO I=1,NHALF
   K=I+NHALF
   PRINT 8, I,HEADB(I+30),K,HEADB(K+30)
8 FORMAT(X,4X,I2,3X,:,:,3X,I2,2X,5X,3X,I2,3X,:,:,3X,I2)
END DO
C
RETURN
END
APPENDIX K

SAMPLE DIGITIZING SESSION
APPENDIX K - SAMPLE DIGITIZING SESSION

Username: 
Password: 
Welcome to VAX/VMS version V4.1 on node V331
Last interactive login on Wednesday, 18-JUN-1986 09:20
Last non-interactive login on Monday, 9-JUN-1986 13:19

$ ALLOC MFA0: 
%MCL-I-ALLOC, _MFA0: allocated

$ A2D
DO YOU WANT TO INITIALIZE THE TAPE? (Y/N) Y
ENTER TAPE LABEL? (R_P): Rk.i1pi
%MCL-I-ALLOC, _MFA0: allocated
IS DATA GOING TO DISK 0 OR 1?: 1
HAVE YOU ENTERED THE DIGITIZING SETUP? (Y/N) N

ENTER ANALOG TAPE NUMBER: 5

ENTER RUN IDENTIFICATION NUMBER: 61

ENTER DATA TAPE PASS NUMBER (1 OR 2)? 1

ENTER DIGITIZING DATE: MONTH 6
DAY 14
YEAR 56

ENTER START TIME CODE: HOUR 6
MINUTES 4L
SECONDS 36

ENTER 20 CHARACTER EVENT DESCRIPTION: Fun 41 Phys L

ENTER ANALOG CHANNEL FOR A/D CHANNEL 1 : L
ENTER ANALOG CHANNEL FOR A/D CHANNEL 2 : 2
ENTER ANALOG CHANNEL FOR A/D CHANNEL 3 : 3
ENTER ANALOG CHANNEL FOR A/D CHANNEL 4 : 4
ENTER ANALOG CHANNEL FOR A/D CHANNEL 5 : 5
ENTER ANALOG CHANNEL FOR A/D CHANNEL 6 : 6
ENTER ANALOG CHANNEL FOR A/D CHANNEL 7 : 7
ENTER ANALOG CHANNEL FOR A/D CHANNEL 8 : 8
ENTER ANALOG CHANNEL FOR A/D CHANNEL 9 : 9
ENTER ANALOG CHANNEL FOR A/D CHANNEL 10 : 10
ENTER ANALOG CHANNEL FOR A/D CHANNEL 11 : 11
ENTER ANALOG CHANNEL FOR A/D CHANNEL 12 : 12
ENTER ANALOG CHANNEL FOR A/D CHANNEL 13 : 13
ENTER ANALOG CHANNEL FOR A/D CHANNEL 14 : 14

DATA WILL BE SAVED IN FILE: R61P1H.DAT

*** EVENT DESCRIPTION: RUN 61 PASS 1 ***

ANALOG TAPE NUMBER: 5
RUN IDENTIFICATION NUMBER: 61
PASS NUMBER: 1
START TIME CODE: 6:41:30
DATE: 6/19/86

<table>
<thead>
<tr>
<th>POSITION</th>
<th>CHANNEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>POSITION</th>
<th>CHANNEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
</tr>
</tbody>
</table>

DOES THIS DATA LOOK CORRECT? (Y/N) Y
DID YOU ASSIGN AND MOUNT THE MAGTAPE? (Y/N) Y
%DCL-I-SUPERSEDE, previous value of MAG_TAPE has been superseded
%MOUNT-I-MOUNTED, R61P1 mounted on MFA0:
HOW MANY BUFFERS DO YOU WANT TO FILL? 1
ENTER RUN IDENTIFICATION NUMBER 61
ENTER PASS NUMBER (1 OR 2) 1

HEADER FILE: R61P1H.DAT

*** EVENT DESCRIPTION: RUN 61 PASS 1 ***

ANALOG TAPE NUMBER: 5
RUN IDENTIFICATION NUMBER: 61
PASS NUMBER: 1
START TIME CODE: 6:41:30
DATE: 6/19/86

<table>
<thead>
<tr>
<th>POSITION</th>
<th>CHANNEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
</tr>
</tbody>
</table>

K-4
HOW MANY FILES DO YOU WANT TO SKIP?

START ANALOG TAPE, HIT RETURN AT 6:41:30

END OF RUN

DO YOU WANT TO COPY THE RECAT FILES TO DISK? (Y/N) Y

ENTER RUN IDENTIFICATION NUMBER

ENTER PASS NUMBER

PROCESSING GROUP 1

HOW MANY FILES DO YOU WANT TO SKIP?

*** EVENT DESCRIPTION: RUN 61 PASS 1 ***

ANALOG TAPE NUMBER: 5
RUN IDENTIFICATION NUMBER: 61
PASS NUMBER: 1
START TIME CODE: 6:41:30
DATE: 6/19/86

OUTPUT UNIT = 12 FILE R61PIC 2B.DAT 500 BUFFERS
OUTPUT UNIT = 14 FILE R61PIC 4B.DAT 500 BUFFERS
OUTPUT UNIT = 20 FILE R61PIC10B.DAT 500 BUFFERS
OUTPUT UNIT = 23 FILE R61PIC13B.DAT 500 BUFFERS
OUTPUT UNIT = 12 FILE R61PIC 2E.DAT 50 BUFFERS
OUTPUT UNIT = 14 FILE R61PIC 4E.DAT 50 BUFFERS
OUTPUT UNIT = 20 FILE R61PIC10E.DAT 50 BUFFERS
OUTPUT UNIT = 23 FILE R61PIC13E.DAT 50 BUFFERS

AT EOF IN RECORD 51
RECAT FILES HAVE BEEN SAVED
DO YOU WANT TO COPY QUICK CHECK FILES TO DISK? (Y/N):
EVENT DESCRIPTION: RUN 61 PASS 1

ANALOG TAPE NUMBER: 5
RUN IDENTIFICATION NUMBER: 61
PASS NUMBER: 1
START TIME CODE: 6:41:30
DATE: 6/19/86

POSITION : CHANNEL
1 : 1
2 : 2
3 : 3
4 : 4
5 : 5
6 : 6
7 : 7

HOW MANY RECORDS DO YOU WANT TO READ?
5

YOU NEED TO RUN DRBO:QUPL.COM TO MAKE PLOTS
DO YOU WANT TO DISMOUNT THE TAPE? (Y/N): Y

DIGITIZING PROCEDURE COMPLETE

NOW GO TO A TEKTRONIX TERMINAL AND PERFORM THE FOLLOWING

Username:
Password:
Welcome to VAX/VMS version V4.1 on node V33:
Last interactive login on Wednesday, 18-JUN-1986 12:33
Last non-interactive login on Monday, 9-JUN-1986 13:19

$ QUPL
RUN NUMBER? : 61
PASS NUMBER? : 1
ARE YOU RUNNING ON THE TEKTRONICS? (Y/N): Y

PLOTTING COMPLETE
$ DEL R61P1*.DAT;
$ LO

K-6
APPENDIX L

COMMON ERRORS
POSSIBLE REASONS
It is very common to get tape/system errors when digitizing data. The digitizing program will print out error messages in the form:

PROBLEM WRITING TO TAPE, IOSB 8C 0 5C0 0
RECORD 5854 BUFFER NUMBER 2
ERROR FROM LPA$IWTBUF - IBUFNO -1 REC 5856
IOSB (HEX) 334 0 A3C0 EFD8

The first "IOSB" code, 8C is the error condition you should be concerned with. Here are a few common errors and actions to be taken. Please refer to the Digital Equipment reference manuals for more detail.

<table>
<thead>
<tr>
<th>CODE</th>
<th>ERROR CODE</th>
<th>MEANING</th>
<th>ACTION TO BE TAKEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>334</td>
<td>SS$_DEVREQERR</td>
<td>The program could not perform it's function in a timely manner. When you needed control someone else had it.</td>
<td>Try you run over. If the system is very busy and 334 keeps occurring in a short period of time, QUIT, GO HOME!!! TRY AGAIN LATER.</td>
</tr>
<tr>
<td>8C</td>
<td>SS$_DRVERR</td>
<td>Magtape drive error. This program inhibits all tape correction retries.</td>
<td>Set tape aside and mark with record number. Use a new tape. RERUN. The tape is good for other applications.</td>
</tr>
<tr>
<td>1A4</td>
<td>SS$_MEDOFL</td>
<td>Tape drive is offline</td>
<td>Go press the &quot;ON LINE&quot; button on digital drive in the computer room.</td>
</tr>
<tr>
<td>1F4</td>
<td>SS$_PARITY</td>
<td>Tape parity error</td>
<td>Set tape aside and mark. Use a new tape. RERUN. Tape may not be good.</td>
</tr>
</tbody>
</table>

Please note when digitizing it is critical that all things run in a smooth timely manner. Because all tape retries are inhibited and because you are usually sharing the system with other users, it is very common to have errors occur. Unfortunately, when digitizing the user cannot not tolerate any problems and a session must be restarted. Please expect this. In order to minimize problems, it is best to run when the system is not busy.
UNCLASSIFIED

ANALOG TO DIGITAL CONVERSION AND VERIFICATION PROGRAMS
FOR A VAX 11/780
P. Maciejewski
Environmental Acoustic Research and Analysis Branch
Surface Ship Sonar Department
TM No. 861214
Distribution Statement A: Approved for Public Release,
Distribution Unlimited
UNCLASSIFIED

DISTRIBUTION LIST

<table>
<thead>
<tr>
<th>External Distribution</th>
<th>No. of Copies</th>
</tr>
</thead>
<tbody>
<tr>
<td>NADC 5031 (B. Steinberg)</td>
<td>1</td>
</tr>
<tr>
<td>5033 (L. Allen)</td>
<td>1</td>
</tr>
<tr>
<td>5039 (R. Foska)</td>
<td>1</td>
</tr>
<tr>
<td>NOSC 732 (C. Persons)</td>
<td>1</td>
</tr>
<tr>
<td>NSWC U21 (M. Williams, E. Hein)</td>
<td>2</td>
</tr>
<tr>
<td>DIA</td>
<td>1</td>
</tr>
<tr>
<td>DTIC</td>
<td>2</td>
</tr>
</tbody>
</table>

External Copies = 11
END
4-87
DTIC