UNCLASSIFIED

NRL Report 8147

Phase-Difference Processing Incorporating Time-Alignment and Time-Compression Techniques

WENDELL L. ANDERSON

Large Aperture Acoustics Branch
Acoustics Division

August 10, 1977

NAVAL RESEARCH LABORATORY
Washington, D.C.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Report 8147	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) PHASE-DIFFERENCE PROCESSING INCORPORATING TIME-ALIGNMENT AND TIME-COMPRESSION TECHNIQUES		5. TYPE OF REPORT & PERIOD COVERED Interim Report on one phase of a continuing NRL Problem 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Wendell L. Anderson		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, D.C. 20375		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NRL Problem S01-61 ARPA Order 3874
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Advanced Research Projects Agency Arlington, Va. 22209		12. REPORT DATE August 10, 1977 13. NUMBER OF PAGES 15
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Coherence analysis
Correlation processing
Phase difference

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes a system for measuring the instantaneous phase difference between two signals as a function of time. This system allows the user to apply various time-difference and doppler corrections to the signals in order to obtain minimum (in the least-squares sense) phase difference between the signals.

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

CONTENTS

PHASE-DIFFERENCE PROCESSING INCORPORATING TIME-ALIGNMENT AND TIME-COMPRESSION TECHNIQUES

INTRODUCTION

Coherent processing has proven to be a viable technique in the study of undersea acoustic signals. To be effective, such processing requires the phase coherence between the received signals to be high. That is, the minimum phase difference between the two signals (after corrections for differences in transit times and doppler) must be small in the least-squares sense. Moreover the time-difference and doppler corrections required to achieve the minimum phase difference provide estimates of these parameters for localization and dynamics determination.

This report will examine the problem of correcting for time alignment and time scale-factor between two signals in the measurement of phase difference and describe in general terms a system for implementing these functions.

SYSTEM DESCRIPTION

The basic technique for measuring the instantaneous phase difference between two signals is illustrated in Fig. 1. The two bandlimited signals are processed through delay networks in order to align the two signals in time register, after which one of the two signals is processed through a time compressor (or expander) to correct for the doppler difference between the two channels. The signals are then fed to an axis-crossover comparator (phase meter) and a duel-trace oscilloscope. The phase difference between the two signals is recorded as a function of time on a Brush stylus recorder.

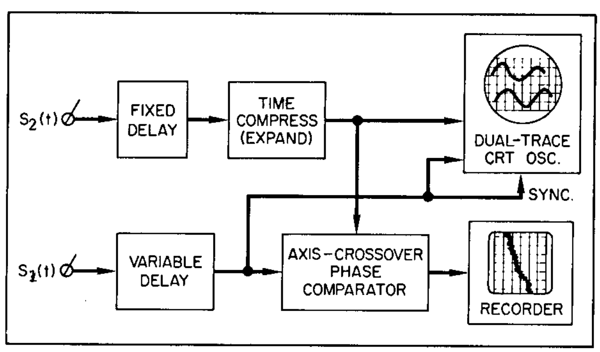

Fig. 1 — Phase-measurement technique

Manuscript submitted June 16, 1977

Digital System

The process was implemented in two steps. In the first step of the process, time delays and time compressions (or expansions) were introduced to the data channels by the use of a PDP-11/40 minicomputer. Each of the digital signals was stored in a contiguous file on the system disk, which allowed the data streams to be accessed by sample number. The time delay for each of the signals is obtained by specifying the starting sample number for each channel. The time alignment between the two signals is varied by changing the starting sample number for the first signal relative to the second. Time is compressed (or expanded) by periodically skipping (or duplicating) samples from the second channel. The new data streams are multiplexed into a single data stream and written onto magnetic tape for processing by the phase meter. A more detailed explanation of the software program to accomplish this first step is given in Appendix A.

Axis-Crossover Phase Meter

In the second step of the process the magnetic tape that is generated is used as the signal input for the system shown in Fig. 2. Each record on the magnetic tape is read into one of two buffers. To maintain a continuous flow of data through the system, one buffer is being filled while the other is being read by the demultiplexer. The two demultiplexed data channels are digital-to-analog converted and low-pass filtered. Each channel is then fed to the phase meter. At each positive axis-crossover of the channel 1 signal, a counter in the phase meter is reset, and at each positive axis-crossover of the channel 2 signal the contents of the counter is converted to a voltage which is fed to the strip-chart recorder. (Further details on the axis-crossover phase meter are given in Ref. 1.)

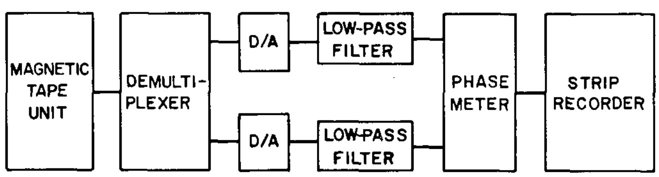

Fig. 2 — Phase-meter system

SYSTEM OPERATION

The system was tested using a signal with the frequency-modulation pattern illustrated in Fig. 3. (More details on the properties of this frequency function are reported in Ref. 2). In the examples given a signal with f_c = 33 Hz and Δf = 0.2 Hz were used. The signals were sampled at 264 Hz.

Time-Register Alignment

The first area of study was to determine the effect of time register (or time alignment) on the phase-difference pattern of identical signals. Figure 4 illustrates the phase-difference pattern for offsets of 0, 1, 2, 4, and 8 seconds. The 45-degree offset from the center of each plot is the result of the clocking from the demultiplexer. Each sample from channel

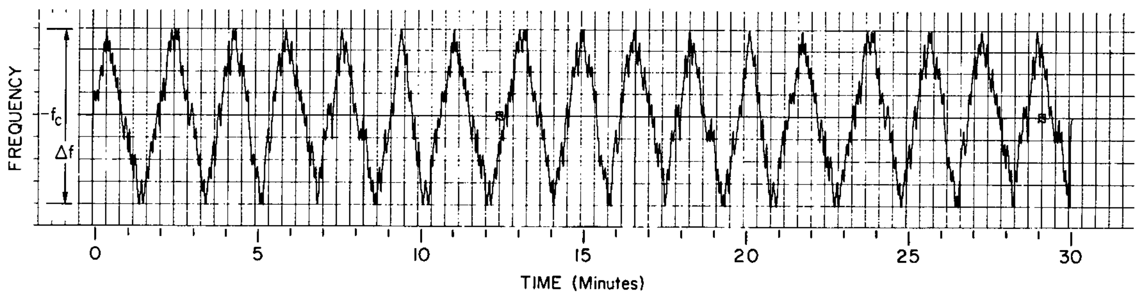

Fig. 3 — Source frequency function

2 is displaced 1/2 sample interval (1/8 wave period) after the corresponding sample for channel 1. In this example the phase-difference patterns approximate triangular wave forms, with peak-to-peak excursions directly proportional to the time-register difference.

Doppler Corrections

The system was also used to examine the effect of both time register and doppler alignment on a motional-source signal received at two widely spaced sensors. Since the locations of the source and sensors were known, it was easy to approximately time-align the signals. After the initial time alignment the phase-difference pattern appears as shown in Fig. 5a. The continuous phase migration (360-degree sawtooth wave) is a result of the doppler difference between the two received signals. Over the time interval of 30 minutes (475,200 samples) a cumulative phase difference of about 37-1/4 full periods (298 samples) is noted. To correct for doppler, it is thus necessary to drop a sample every 1595 samples from the second signal. The effect of this correction is shown in Fig. 5b. There is now no net phase movement of the phase difference over the 30-minute interval, but an irregular waveform similar to that shown in Fig. 4 may be noted. This suggests that some time-register difference still exists between the two signals. Figure 5c illustrates the final phase difference between the two signals after they have been properly time aligned. This final alignment indicates that the time difference between channel 2 and channel 1 was 1/2 second smaller than expected and that a doppler compression of 0.000627 (or 0.02069 Hz) existed between the two signals.

The high-frequency (45-degree-sawtooth) component of the phase-difference pattern seen in Figs. 5b and 5c is a result of the discrete manner of correcting for doppler difference. (The spikes in the Fig. 5 phase-difference plots are caused by tape dropouts during the reading of the magnetic tape and should be ignored.)

CONCLUSIONS

The system described in this report provides a simple and convenient method for measuring the instantaneous phase difference between two signals as a function of both time-register and time-scale-factor (doppler) difference. By examining the phase-difference plots, one can determine the time-register and doppler difference required for minimum phase

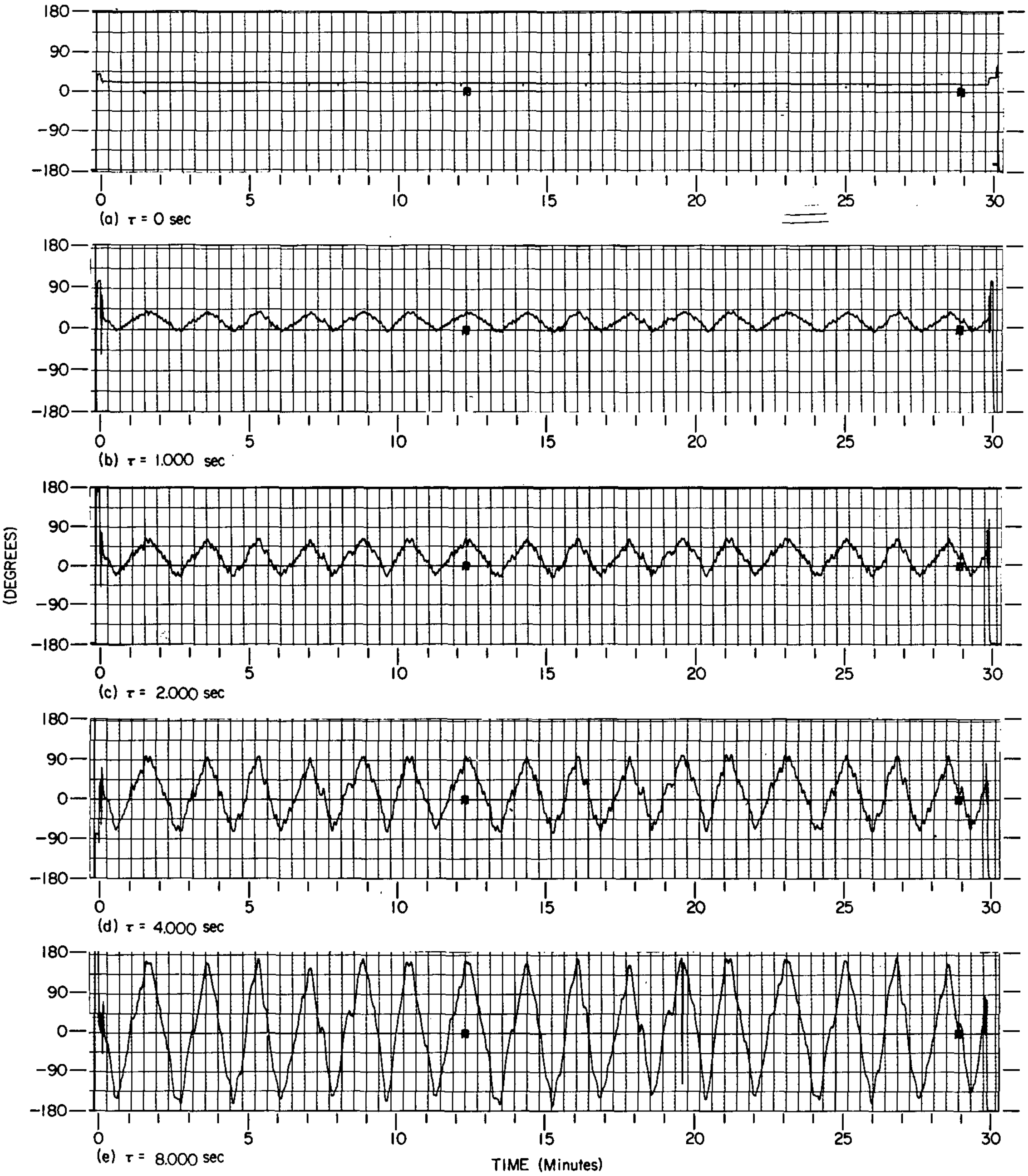

Fig. 4 — Phase-difference plots of the signal versus itself

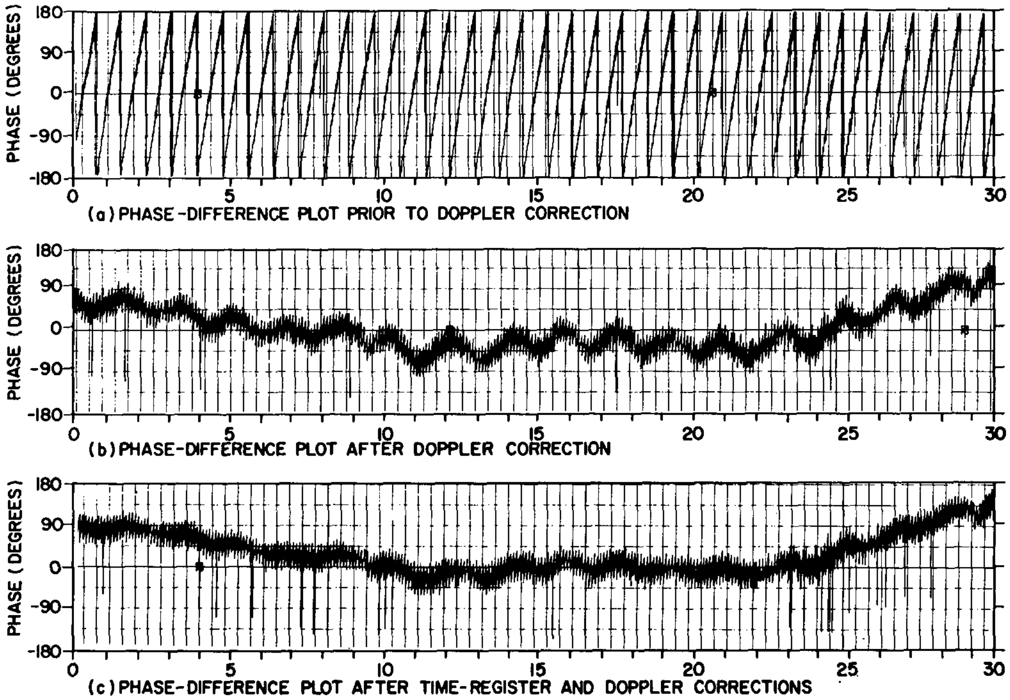

Fig. 5 — Phase-difference patterns of the signal received at separate sensors

difference. In so doing, estimates of the time-register and doppler shift are provided, and an estimate of the maximum phase correlation between the signal channels may readily be derived.

ACKNOWLEDGMENTS

The author is indebted to Dr. Albert A. Gerlach for his many helpful comments and discussions and to Mr. George Nacht, who built and operated the system described in Fig. 2.

The effort on the study was supported by the Defense Advanced Research Projects Agency under ARPA Order 3074.

REFERENCES

1. G. G. Nacht and H. L. Peterson, "A Hybrid Cumulative Phase Difference Meter," Report of NRL Progress, Feb. 1966, pp. 37-38.
2. A.A. Gerlach, "Random Frequency Function Generation," NRL Report 7697, Apr. 30, 1974.

Appendix A
PROGRAM LISTING

This appendix describes the assembly-language program which produced the phase-meter input tapes. The actual program is the first five of the six printout pages and runs under the DEC DOS/BATCH system. This program assumes that the data for the two channels have been stored as eight-bit bytes in the contiguous disk files STANO1.DAT and STANO2.DAT respectively. Each of these data files consists of 512-byte records. The program writes the multiplexed data onto a seven-track 556-BPI tape (556 binary digits per inch) with 2000 six-bit-byte records. (Each six-bit byte contains the six most significant bits of a sample.)

As indicated by the program contents, lines 1 through 29 initialize the program. The number of records in each input data file is entered first. This is followed by the number of samples that each data stream is to be delayed. These latter numbers are converted to a starting record number and a starting address within the record. This is accomplished by lines 39 through 43 and lines 53 through 58 (for channels 1 and 2 respectively), with the starting addresses stored in general-purpose registers R1 and R2.

Lines 60 and 61 of the program then examine bit 0 of the switch register to see if a doppler correction is to be made. If bit 0 is on (indicating a doppler correction), then the number of samples between address corrections is stored in DROP and DOPP, and the direction of the correction (retard or advance address by one) is stored in ANSWR.

The program now places the starting address of the output buffer in register R3 and sets the output-buffer counter register R4 to the size in words (one word is equivalent to two data samples) of the output buffer. The program then reads the first required data record for channels 1 and 2 into the buffers NPUT1 and NPUT2 respectively.

The program begins multiplexing the data channels into the output buffer. In line 84 a sample from channel 1 is moved from the channel 1 input buffer into the output buffer, and the two buffer pointers are incremented by 1. The process is repeated for channel 2 in line 85. The program then decrements the R4 counter by 1 and tests to see if the output buffer is full. If the buffer is full, the output buffer is written onto the magnetic tape and registers R3 (output-buffer pointer) and R4 (output-buffer counter) are reset.

Line 105 checks NOPOP to determine if a doppler correction is to be made. If so, the counter DROP is decremented by 1 and compared with zero. If DROP is zero, DROP is reset to DOPP and ANSWR is examined to determine whether a sample should be repeated or dropped. If a sample is to be repeated, the counter R2 is decremented by 1; and if a sample is to be dropped, R2 is incremented by 1 and the input buffer tested to determine if the channel-2 data have been exhausted. If the data have been exhausted, a new record is read into NPUT2.

This process continues until the program must access a record greater than the size of one of the input files (as specified by RECBL1 and RECBL2). Finally the most significant bit of the switch register is checked. If it is on, the program recycles to generate a new file of multiplexed data with a new time register and/or a new doppler correction. Otherwise control is returned to the DOS monitor.

An example of the entries to the program is shown in Fig. A1. (This program produced the data tape for the phase-difference plots shown in Fig. 5.) The underlined portions are the entries entered by the operator. In the example the data inputs are six-digit ASCII numbers, each less than the decimal number 65,536. In addition the entry indicating that samples were to be dropped from the second channel was a four-digit ASCII entry SUB2. (If samples were to have been added, the entry would have been ADD2.)

```
(all sense switches down)

$RU   DOPP12
LAST DATA RECORD FOR STATION 1                         001250
LAST DATA RECORD FOR STATION 2                         001250
NO. OF SAMPLES TO DROP FROM STATION-1 DATA             026400
NO. OF SAMPLES TO DROP FROM STATION-2 DATA             026400
(put sense switch 0 up)
NO. OF SAMPLES TO DROP FROM STATION-1 DATA             026400
NO. OF SAMPLES TO DROP FROM STATION-2 DATA             026400
NO. OF SAMPLES BETWEEN DOPPLER CORRECTIONS             001595
ADD OR SUB SAMPLES FROM CHANNEL 2?  SUB2
(put sense switch 15 up)
NO. OF SAMPLES TO DROP FROM STATION 1 DATA             026532
NO. OF SAMPLES TO DROP FROM STATION 2 DATA             026400
NO. OF SAMPLES BETWEEN DOPPLER CORRECTIONS             001595
ADD OR SUB SAMPLES FROM CHANNEL 2  SUB2
```

Fig. A1 — Example of a run of the multiplexing program

```
        SW=177570
; PROGRAM TO MULTIPLEX DATA
        .TITLE  DOPPLR
        .MCALL  .INIT,.RLSE,.WAIT,.OPEN,.CLOSE
        .MCALL  .EXIT,.RECRD,.D2BIN,.TRAN,.RADPK
        .GLOBL  OPUTI,OPUT,OPUTF,OPUTD
START:  .INIT   #DSKBL1
        .INIT   #DSKBL2
        .INIT   #KBBLK
; OBTAIN NUMBER OF LAST RECORD IN FIRST FILE
        .TRAN   #KBBLK,#TBLK1
        .WAIT   #KBBLK
        .TRAN   #KBBLK,#TRNBLK
        .WAIT   #KBBLK
        .D2BIN  #DECNO+1
        MOV     (SP)+,RECNO1
        DEC     RECNO1
; OBTAIN NUMBER OF LAST RECORD IN SECOND FILE
        .TRAN   #KBBLK,#TBLK2
        .WAIT   #KBBLK
        .TRAN   #KBBLK,#TRNBLK
        .WAIT   #KBBLK
        .D2BIN  #DECNO+1
        MOV     (SP)+,RECNO2
        DEC     RECNO2
; INITIALIZE FILES
BEGIN:  .OPEN   #DSKBL1,#FILNM1
        .OPEN   #DSKBL2,#FILNM2
        JSR     R5,OPUTI
; OBTAIN NUMBER OF SAMPLES TO SKIP FOR
; CHANNEL 1
        .TRAN   #KBBLK,#TBLK3
        .WAIT   #KBBLK
        .TRAN   #KBBLK,#TRNBLK
        .WAIT   #KBBLK
        .D2BIN  #DECNO+1
; STARTING RECORD NUMBER FOR FIRST FILE INTO RECBL1
; ADDRESS OF FIRST SAMPLE INTO R1
        MOV     (SP)+,R1
        CLR     R0
        DIV     #512.,R0
        ADD     R0,RECBL1+10
        ADD     #NPUT1,R1
; OBTAIN NUMBER OF SAMPLES TO SKIP FOR
; CHANNEL 2
        .TRAN   #KBBLK,#TBLK4
        .WAIT   #KBBLK
        .TRAN   #KBBLK,#TRNBLK
```

```
        .WAIT    #KBBLK
        .D2BIN   #DECNO+1
; STARTING RECORD NUMBER FOR SECOND FILE IN RECBL2
; ADDRESS OF FIRST SAMPLE INTO R3
        MOV      (SP)+,R3
        CLR      R2
        DIV      #512.,R2
        ADD      R2,RECBL2+10
        MOV      R3,R2
        ADD      #NPUT2,R2
;  TEST TO SEE IF DOPPLR CORRECTIONS ARE TO BE MADE
        MOV      @#SW,NODOP
        BIC      #177776,NODOP
        BEQ      INIT
; OBTAIN NUMBER OF SAMPLES BETWWEN DOPPLER CORRECTIONS
        .TRAN    #KBBLK,#TBLK5
        .WAIT    #KBBLK
        .TRAN    #KBBLK,#TRNBLK
        .WAIT    #KBBLK
        .D2BIN   #DECNO+1
        MOV      (SP)+,DOPP
        MOV      DOPP,DROP
; DROP OR ADD SAMPLES FROM CHANNEL 2?
        .TRAN    #KBBLK,#TBLK6
        .WAIT    #KBBLK
        .TRAN    #KBBLK,#TRNBL1
        .WAIT    #KBBLK
        .RADPK   #DECNO
        MOV      (SP)+,ANSWR
; INPUT DATA
INIT:   MOV      #OUT,R3
        MOV      #1000.,R4
        JSR      R5,RED1
        JSR      R5,RED2
; MULTIPLEX THE CHANNELS
LOOP:   MOVB     (R1)+,(R3)+        ;CHANNEL 1 SAMPLE INTO OUTPUT FILE
        MOVB     (R2)+,(R3)+        ;CHANNEL 2 SAMPLE INTO OUTPUT FILE
        DEC      R4
        BNE      CONT               ;OUTPUT BUFFER FULL?
        JSR      R5,OPUT            ;YES
        BR       .+4
        .WORD    OUT
        MOV      #1000.,R4
        MOV      #OUT,R3
        JSR      R5,OPUTD
        MOV      @#172520,R0
        BIC      #175777,R0
        BNE      DONE               ;ERROR IN WRITING?
CONT:   CMP      R1,#NPUT1+512.
        BNE      NXT                ;LAST SAMPLE IN CHANNEL 1 BUFFER?
        JSR      R5,RED1            ;YES
        MOV      #NPUT1,R1
```

```
NXT:     CMP    R2,#NPUT2+512.
         BNE    DOPPLR          ;LAST SAMPLE FOR CHANNEL 2 BUFFER?
         JSR    R5,RED2         ;YES
         MOV    #NPUT2,R2
DOPPLR:  TST    NODOP
         BEQ    LOOP            ;DOPPLER CORRECTIONS USED?
         DEC    DROP            ;YES
         BNE    LOOP            ;MAKE A DOPPLER CORRECTION NOW?
         MOV    DOPP,DROP       ;YES
         CMP    ANSWR,ADD
         BNE    TWO
         DEC    R2              ;ADD A SAMPLE
         JMP    LOOP
TWO:     INC    R2              ;DROP A SAMPLE
         CMP    R2,#NPUT2+512.
         BNE    LOOP            ;LAST SAMPLE FOR CHANNEL 2 BUFFER?
         JSR    R5,RED2         ;YES
         MOV    #NPUT2,R2
         JMP    LOOP
; READ DATA FOR CHANNEL 1
RED1:    CMP    RECBL1+10,RECNO1
         BGT    DONE            ;CHANNEL 1 DATA EXHAUSTED?
         .RECRD #DSKBL1,#RECBL1 ;NO
         .WAIT  #DSKBL1
         MOV    RECBL1,R0
         TSTB   RECBL1+1
         BNE    ERROR
         ADD    #1,RECBL1+10
         RTS    R5
; READ DATA FOR CHANNEL 2
RED2:    CMP    RECBL2+10,RECNO2
         BGT    DONE            ;CHANNEL 2 DATA EXHAUSTED?
         MOVB   NPUT2+512.,NPUT2-1      ;NO
         .RECRD #DSKBL2,#RECBL2
         .WAIT  #DSKBL2
         MOV    RECBL2,R0
         TSTB   RECBL2+1
         BNE    ERROR
         ADD    #1,RECBL2+10
         RTS    R5
; PROCESSING DONE
ERROR:   .TRAN  #KBBLK,#TBLK
         .WAIT  #KBBLK
DONE:    CLR    RECBL1+10
         CLR    RECBL2+10
         JSR    R5,OPUTF
         .CLOSE #DSKBL1
         .CLOSE #DSKBL2
         TST    @#SW            ;TEST BIT 15 OF SWITCH REGISTER
         BMI    FIN             ;IF ON DONE
```

```
            JMP     BEGIN       ;OTHERWISE GENERATE NEW MAG TAPE FILE
FIN:        .RLSE   #KBBLK
            .RLSE   #DSKBL1
            .RLSE   #DSKBL2
            .EXIT
; DISK LINK BLOCK
            .WORD   0
DSKBL1:     .WORD   0
            .RAD50  /DIS/
            .WORD   1
            .RAD50  /DK/
            .WORD   0
DSKBL2:     .WORD   0
            .RAD50  /DUS/
            .WORD   1
            .RAD50  /DK/
; KEYBOARD LINK BLOCK
            .WORD   0
KBBLK:      .WORD   0,0,1
            .RAD50  /KB/
; RECORD BLOCK FOR FIRST CHANNEL
RECBL1:     .WORD   4,NPUT1,512.,0,0
; RECORD BLOCK FOR SECOND CHANNEL
RECBL2:     .WORD   4,NPUT2,512.,0,0
; OUTPUT BLOCK FOR KEYBOARD
TBLK:       .WORD   0,MESG,4,2,0
TBLK1:      .WORD   0,MESG1,22.,2,0
TBLK2:      .WORD   0,MESG2,22.,2,0
TBLK3:      .WORD   0,MESG3,22.,2,0
TBLK4:      .WORD   0,MESG4,22.,2,0
TBLK5:      .WORD   0,MESG5,22.,2,0
TBLK6:      .WORD   0,MESG6,17.,2,0
MESG:       .ASCII  / ERROR/
            .BYTE   15,12
MESG1:      .ASCII  /LAST DATA RECORD TO READ FOR STATION 1      /
            .EVEN
MESG2:      .ASCII  /LAST DATA RECORD TO READ FOR STATION 2      /
            .EVEN
MESG3:      .ASCII  /NO. OF SAMPLES TO DROP FROM STATION 1 DATA /
           .EVEN
MESG4:      .ASCII  /NO. OF SAMPLES TO DROP FROM STATION 2 DATA /
            .EVEN
MESG5:      .ASCII  /NO. OF SAMPLES BETWEEN DOPPLER CORRECTIONS /
            .EVEN
MESG6:      .ASCII  /ADD OR SUB SAMPLES FROM CHANNEL 2? /
            .EVEN
ADD:        .RAD50  /ADD/
; TRAN BLOCK FOR KEYBOARD
TRNBLK:     .WORD   0,DECNO,4.,5,0
TRNBL1:     .WORD   0,DECNO,3,4,0
```

```
; FILENAME FOR FIRST CHANNEL
         .WORD   0,1
FILNM1:  .RAD50  /STAN01DAT/
         .WORD   0,233
; FILENAME FOR SECOND CHANNEL
         .WORD   0,1
FILNM2:  .RAD50  /STAN02DAT/
         .WORD   0,233
; STORAGE AREAS
ANSWR:   .WORD   0
DOPP:    .WORD   0
DROP:    .WORD   0
RECNO1:  .WORD   0
RECNO2:  .WORD   0
OUT:     .BLKW   1000.
DECNO:   .BLKW   4
NPUT1:   .BLKW   256.
         .WORD   0
NPUT2:   .BLKW   256.
NODOP:   .WORD   0
         .END    START
```

```

; OPUTI  - INITIALIZATION FOR OUTPUT
; OPUT   - OUTPUTS A RECORD TO 7 TRACK MAG TAPE (556 BPI)
; OPUTD  - CHECKS TO SEE IF OUTPUT COMPLETED
; OPUTF  - CLOSES FILE
         .TITLE  OUTPUT
         .MCALL  .INIT,.TRAN,.WAIT,.RLSE,.SPEC
OPUTI::  .INIT   #MAGBLK
         .SPEC   #MAGBLK,#SFBLK
         RTS     R5
OPUT::   MOV     R1,-(SP)
         MOV     R0,-(SP)
         MOV     2(R5),TRNBLK+2
         MOV     TRNBLK+2,R1
         MOV     #2000.,R0
LOOP:    ASRB    (R1)
         ASRB    (R1)+
         SOB     R0,LOOP
         .TRAN   #MAGBLK,#TRNBLK
         MOV     (SP)+,R0
         MOV     (SP)+,R1
         RTS     R5
OPUTD::  .WAIT   #MAGBLK
         RTS     R5
OPUTF::  .SPEC   #MAGBLK,#SFBLK1
         .RLSE   #MAGBLK
         RTS     R5
SFBLK:   .BYTE   6,3
         .WORD   0
         .BYTE   0,1
         .WORD   0
SFBLK1:  .BYTE   2,3
         .WORD   0,0,0
         .WORD   0
MAGBLK:  .WORD   0
         .RAD50  /MAG/
         .BYTE   1,1
         .RAD50  /MT/
TRNBLK:  .WORD   0,0,1000.,2,0
         .END
```