AD-A052 072 NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF F/G 6/3

SINGLE HYDROPHONE TECHNIQUE FOR OBTAINING SPECTRAL SOURCE LEVEL--ETC(U)

DEC 77 R M BOSTIAN

UNCLASSIFIED NL

| OF |

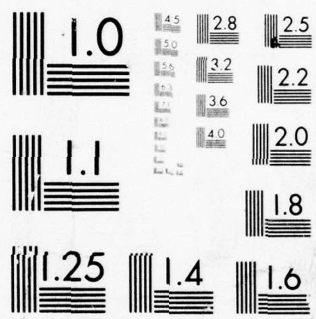
AD A052 072

END

DATE FILMED

5-78

DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0

NAVAL POSTGRADUATE SCHOOL

Monterey, California

Master's THESIS

SINGLE HYDROPHONE TECHNIQUE FOR OBTAINING SPECTRAL SOURCE LEVELS OF MARINE MAMMALS IN COASTAL WATERS

by

Richard M. Bostian

Dec[redacted]977

49 p.

Thesis Advisor: Herman Medwin

Approved for public release; distribution unlimited

251 450

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Single Hydrophone Technique for Obtaining Spectral Source Levels of Marine Mammals in Coastal Waters	5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; December 1977
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Richard Massey Bostian	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940	12. REPORT DATE December 1977
	13. NUMBER OF PAGES 50
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)

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

During the annual Gray Whale migration from the Aleutians to Baja California, the mammals travel in coastal waters, thereby presenting an opportunity for the study of their sound spectral and source levels and vocabulary. However, such measurements are distorted by surface and bottom reverberation. Using the theory of rough surface scattering, knowledge of the bottom impedance, and correlation techniques, it is possible to decompose the shallow water reverberation into the contributions from different paths. From this, the range, depth and the deverberated spectral source levels of the sounds of the mammal can be determined by use of only one hydrophone rather than the conventional three or four. The theory, deverberation programming, and experimental results are presented for a model of the whale's pulsed radiation in a laboratory model coastal environment.

Approved for public release; distribution unlimited

Single Hydrophone Technique for Obtaining Spectral Source Levels of Marine Mammals in Coastal Waters

by

Richard M. Bostian
Lieutenant Commander, United States Navy
B.S., University of Florida, 1967

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ENGINEERING ACOUSTICS

from the

NAVAL POSTGRADUATE SCHOOL
December 1977

Author: Richard M Bostian

Approved by: ______________________ Thesis Advisor

______________________ Second Reader

____
Chairman, Department of Physics and Chemistry

George J. Haltiner
Dean of Science and Engineering

ABSTRACT

During the annual Gray Whale migration from the Aleutians to Baja California, the mammals travel in coastal waters, thereby presenting an opportunity for the study of their sound spectral and source levels and vocabulary. However, such measurements are distorted by surface and bottom reverberation. Using the theory of rough surface scattering, knowledge of the bottom impedance, and correlation techniques, it is possible to decompose the shallow water reverberation into the contributions from different paths. From this, the range, depth and the deverberated spectral source levels of the sounds of the mammal can be determined by use of only one hydrophone rather than the conventional three or four. The theory, deverberation programming, and experimental results are presented for a model of the whale's pulsed radiation in a laboratory model coastal environment.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I wish to express my sincere appreciation to Dr. Herman Medwin for guidance, motivation and resources which he provided for this work. Special thanks are also due to Mrs. Jeanie Savage for her invaluable help in editing and modifying the computer programs used in this thesis. Discussions with Dr. William C. Cummings, formerly of NOSC (San Diego), and Dr. William A. Watkins of Woods Hole Oceanographic Institution (Massachusetts) have been very valuable. The financial assistance of the Office of Naval Research is appreciated.

I. INTRODUCTION

Once yearly, the Gray Whale, Eschrichtius glaucus, migrates southward from the Aleutians and passes very close to the California coast in shallow water. During this migration, the sounds in the water close to the whales can easily be recorded; but, they may not be the true sounds produced by the whale. The whale normally produces an intermittent, short duration signal which in shallow water is received at the hydrophone via direct, surface reflected and bottom reflected paths. Since the path lengths are different, the signals arrive at the hydrophone at different times and they interfere. To obtain the true sounds produced by the whales, this interference, which is called reverberation, must be eliminated.

The purpose of this thesis is to model in the laboratory the whale sounds in the shallow water and to develop a technique to determine the range, eliminate the surface and bottom reverberation and calculate the spectral source levels as a function of time.

II. THEORY

In a reverberant environment, the original signal can only be realized if the reverberation is removed. The method used to remove the reverberation, which is here called "deverberation," assumes that the whale is a point source, the geometrical spreading is spherical, the water is isovelocity and the water depth is constant.

Before the deverberation technique can be applied, the direct, surface reflected and bottom reflected path distances must be known. Normally, this information is obtained by knowing the source position and calculating the path distances from the known geometry. Determining the horizontal source position is difficult and requires at least two directional or three omnidirectional hydrophones, accurately fixed with respect to each other at all times. To determine the depth requires at least one additional hydrophone at a different depth. Generally, three or four hydrophones are deployed [Refs. 1 and 2]. With each additional hydrophone, the complexity of the system is increased since each hydrophone requires its own amplifying and recording system. In shallow water, however, taking advantage of the surface and bottom reflections, one can use a single hydrophone and gather all the information necessary for the application of the deverberation technique.

Consider the direct, surface scattered, and bottom scattered sounds received at only one hydrophone which is deployed in shallow water as depicted in figure 1. It will be shown that when the differences between the arrival times for the three paths are known the three path distances can be calculated. Using the surface specularly scattered path, R_s, it is seen that

$$R_S = R'_S + R''_S$$

$$R'^2_S = D^2 + S'^2$$

$$R''^2_S = H^2 + S''^2$$

where

$$S'^2 = \frac{D^2}{(H+D)^2}\left[R^2 - (H-D)^2\right]$$

$$S''^2 = \frac{H^2}{(H+D)^2}\left[R^2 - (H-D)^2\right]$$

Substituting and rearranging terms gives

(1) $$R'_S = \frac{D}{(H+D)}\left(R^2 + 4HD\right)^{\frac{1}{2}}$$

(2) $$R''_S = \frac{H}{(H+D)}\left(R^2 + 4HD\right)^{\frac{1}{2}}$$

Using τ_s, the time difference between the direct path arrival and the surface reflected path arrival and C, the mean speed of sound, gives

$$C\tau_S = R'_S + R''_S - R$$

and therefore

(3) $$C\tau_S = \left(R^2 + 4HD\right)^{\frac{1}{2}} - R$$

Similarly using the bottom specularly scattered path

$$R_B = R'_B + R''_B$$

$$R'^2_B = (Z-D)^2 + B'^2$$

$$R''^2_B = (Z-D)^2 + B''^2$$

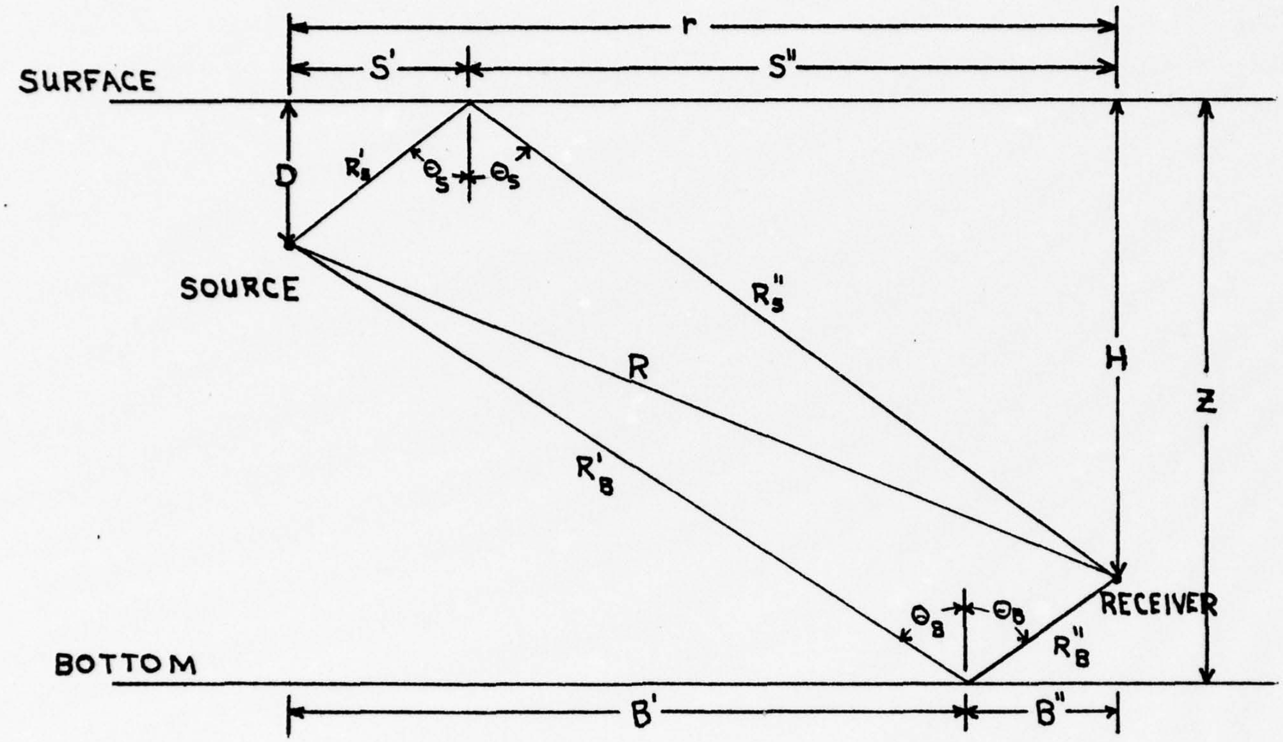


FIGURE 1: DEPLOYMENT OF ONE HYDROPHONE IN A SHALLOW WATER ENVIRONMENT

where

$$B' = \frac{(Z-D)}{(2Z-H-D)} \left[R^2 - (H-D)^2 \right]^{\frac{1}{2}}$$

$$B'' = \frac{(Z-H)}{(2Z-H-D)} \left[R^2 - (H-D)^2 \right]^{\frac{1}{2}}$$

Substituting and rearranging

$$(4) \quad R_B' = \frac{(Z-D)}{(2Z-H-D)} \left[R^2 + 4(Z^2 + HD - ZH - ZD) \right]^{\frac{1}{2}}$$

$$(5) \quad R_B'' = \frac{(Z-H)}{(2Z-H-D)} \left[R^2 + 4(Z^2 + HD - ZH - ZD) \right]^{\frac{1}{2}}$$

Using τ_B, the time difference between the direct path arrival and the bottom reflected path arrival gives

$$c\tau_B = R_B' + R_B'' - R$$

and, therefore,

$$(6) \quad c\tau_B = \left[R^2 + 4(Z^2 + HD - ZH - ZD) \right]^{\frac{1}{2}} - R$$

Solving equations (3) and (6) simultaneously for R, the range of the source from the receiver and D, the depth of the source, produces

$$(7) \quad D = \frac{\tau_s(c^2\tau_s\tau_B + 4Z^2 - 4ZH) - \tau_s(c\tau_B)^2}{4[H\tau_B + \tau_s(Z-H)]}$$

$$(8) \quad R = \frac{4HD - (c\tau_B)^2}{2c\tau_s}$$

The equation for the range is left as a function of the water depth to facilitate its being programmed. Now that the range and depth are known, the other two path distances can be calculated. From equations (1) and (2), the surface reflected path distance is

$$(9)\quad R_S = (R^2 + 4HD)^{\frac{1}{2}}$$

and from equations (4) and (5), the bottom reflected path distance is

$$(10)\quad R_B = [R^2 + 4(Z^2 + HD - ZH - ZD)]^{\frac{1}{2}}$$

Therefore, when the time arrival differences for the different paths are known, equations (8), (9), and (10) can be used to determine the first three path distances. The paths for multiple reflections can be calculated directly from the known geometry, assuming specular scatter.

For a transient signal, determination of the differential arrival times τ_S and τ_B can be realized by the use of an autocorrelation technique. The correlation function can be defined as [Ref. 3]

$$(11)\quad R(\tau) = E\{[v(t) - u][v(t+\tau) - u]\}$$

where v is the time average, u is the mean, and E is the expected value of the received signal. This process is programmed using a digital summation

$$(12)\quad R(\tau) = \frac{1}{n-\tau}\sum_{i=1}^{n-\tau}[v_i(t) - u][v_{i+\tau}(t) - u]$$

with n representing the total number of samples in the record. Performing the autocorrelation on the reverberant signal at the one hydrophone gives peaks at delay times corresponding to zero delay time, and the arrival delay times from the reflected signals. The peaks are realized only when the direct signal is delayed enough to correlate with the reflected signals. The computer program called AUTOPEAK performs the autocorrelation and then applies an envelope detection to determine the delay times for the peaks. These delay times are then used in equations (7), (8), (9), and (10) to determine the range, depth, and reflected path distances. Thereby, all the geometrical information necessary for the deverberation technique has been obtained.

Computer programs have been developed for deverberation in either the frequency or time domain.

The computer program designed to eliminate the reverberations in the frequency domain is called DEVERB. For the time being, assume only one frequency component, whose amplitude and frequency are functions of time. For a signal with time-varying frequency and amplitude, we can describe the pressure at the receiver, due only to the direct path signal as [Ref. 4]

$$(13)\quad P_D(t) = C(t)\, e^{j\omega(t)t}$$

Then, taking into account spherical divergence, the spatial phase shift, and specular scattering from a Gaussian rough surface, the pressure at the hydrophone due to the surface scattered signal can be written as

$$(14)\quad P_S(t') = P_S(t-\tau_s) = \frac{R}{R_S} e^{-\frac{g}{2}} C(t)^{j[\omega(t)t-(R_s-R)k(t)+\pi]}$$

$$t \geq \tau_s$$

and the pressure due to the once bottom reflected signal is

$$(15)\quad P_B(t'') = P_B(t-\tau_B) = \frac{R}{R_B}\mathcal{R}\, C(t)\, e^{j[\omega(t)t - (R_B - R)k(t) + \gamma]}$$
$$t \geq \tau_B$$

$\mathcal{R}$ and $e^{-g/2}$ are the frequency-dependent pressure amplitude reflection coefficients, and γ and π are the phase shifts due to the bottom and surface reflections, respectively. The surface reflection coefficient depends on the roughness of the surface. The exponent for specular scattering is [Ref. 5]

$$(16)\quad g^{\frac{1}{2}} = \frac{4\pi\sigma}{\lambda}\cos\theta_S$$

where σ is the R.M.S. wave height, λ is the wavelength of the sound signal and Θ_S is the angle of incidence. Equations (14) and (15) are derived from previously received direct path pressures, corrected for path differences and phase shifts and represent the pressures due to the single reflected paths. The coherent sum of equations (13), (14) and (15) is the pressure sensed by the receiver when these three components are present.

The signal processing is done digitally. Therefore, the continuous time dependence is replaced, whenever it occurs in the previous equations, by digital block numbers, indicated by the subscript index "k." Each block contains enough data samples of the incoming signal to give the desired spectral frequency resolution during a block duration which is small compared to the total duration of the time-varying signal. The frequency change of the signal within a block duration is assumed to be much smaller than the frequency resolution of the digital processing. The block duration is

$$T = \frac{\text{NUMBER OF SAMPLES IN THE BLOCK}}{\text{SAMPLING FREQUENCY}}$$

The time the block ends is related to the continuous time t by

(17) $t = TK; \; K = 1, 2, 3 \cdots$

The index "i" is used for the spectral frequency component of the complex wave being analyzed.

The equation for the source pressure at unit distance using the frequency deverberation correction is

$$(18)\quad D_{K,i}(1)\, e^{j(\beta_{K,i} - Rk_i)} = R\Big\{ C_{K,i}\, e^{j\phi_{K,i}} - \Big[\frac{R}{R_S} e^{-\frac{\beta}{2}} D_{N,i}\, e^{j(\alpha_{N,i} - (R_S - R)k_i - \pi)}\Big] 1(K-N) - \Big[\frac{R}{R_B} \mathcal{R}\, D_{M,i}\, e^{j(\alpha_{M,i} - (R_B - R)k_i - \gamma)}\Big] 1(K-M) - \Big[\frac{R}{R_{SB}} \mathcal{R} e^{-\frac{\beta}{2}} D_{L,i}\, e^{j(\alpha_{L,i} - (R_{SB} - R)k_i - \pi - \gamma)}\Big] 1(K-L) - \text{ETC.}\Big\}$$

where 1(K-N), 1(K-M) and 1(K-L) are unity factors with values

$1(K-N) = 1 \quad K \geq N$

$\quad\quad = 0 \quad$ otherwise

$1(K-M) = 1 \quad K \geq M$

$\quad\quad = 0 \quad$ otherwise

$1(K-L) = 1 \quad K \geq L$

$= 0 \quad$ otherwise

The pressure amplitude of the i^{th} frequency component in block K of the receiver reverberant signal is represented by $C_{K,i}$ and its phase represented by $\phi_{K,i}$. The deverberated pressure amplitude is represented by $D_{K,i}$ and its phase by $\beta_{K,i}$. The first term on the right hand side of the equation represents the received signal. The second term represents the correction due to a single specular scatter from the surface; the third represents a single bottom reflection correction, and the fourth represents the correction for a path which includes one surface and one bottom reflection. The equation can be expanded to include other multiple reflections.

The block indices are determined by

(19) $M = K - \frac{\tau_B}{T}$

(20) $N = K - \frac{\tau_S}{T}$

(21) $L = K - \frac{\tau_{SB}}{T}$

The output of the frequency deverberation program is a series of spectra of the consecutive blocks and its Fourier transform is a time plot of the deverberated signal.

The above procedure takes the signal from the time domain (the time series after A/D conversion) by Fourier transform to the frequency domain, where the known frequency dependent reflection coefficients are easily applied, and then back to the time domain to verify the effectiveness of the process.

When the reflection coefficients can be assumed to be frequency-independent, a simple point-by-point deverberation procedure can be applied in the time domain. The applicable temporal deverberation equation is

$$(22)\quad D_K = C_K + \langle e^{-\frac{g}{2}} \rangle \frac{R}{R_S} D_N - \mathcal{R} \frac{R}{R_B} D_M + \langle e^{-\frac{g}{2}} \rangle \mathcal{R} \frac{R}{R_{SB}} D_L + \text{ETC.}$$

C_K represents the pressure amplitude for the K^{th} sample. The other terms are similar to those in equation (18). For low roughness surfaces $g < 1$ and the use of $e^{-g/2}$ over the appropriate frequencies will be a good approximation which is essentially independent of frequency. One advantage of the temporal deverberation technique is the relative freedom from restrictions of block size; the block size is determined only by the desired frequency resolution and the rate of change of frequency of the transient source.

III. PROCEDURE

In order to model the Gray Whale's environment in Monterey Bay, a three meter cube "anechoic" water-filled tank was used. An artifical bottom made of hard rubber with an experimentally determined ρc of approximately 2.4×10^6 mks rayls was positioned one meter below the water surface. This type of bottom was chosen for its specific acoustic impedance since the bottom at the listening area in Monterey Bay has a ρc of approximately 3×10^6 mks rayls. The depth of the bottom and relative placement of the source and receiver were determined in order to obtain realistic modeled delay times between the reflected signals and the direct signal. A 1.8 cm diameter spherical source was used because of its small size and its ability to transmit a signal with minimum distortion; but it also limited the minimum frequency to about 10 kHz. An FM up-sweep of varying widths and sweep rates was used to model one of the sounds produced by the Gray Whale.

The equipment was connected as in figure 2 with the master unit pulse generator being used to synchronize the sampling frequency oscillator which determined the start and stop frequencies, sweep rate and pulse width. The pulse repetition rate was determined by the master unit pulse generator. The signal was then amplified and sent to the source. The reverberent signal was received by an LC-10 hydrophone, and then amplified again and sent to the analog-to-digital

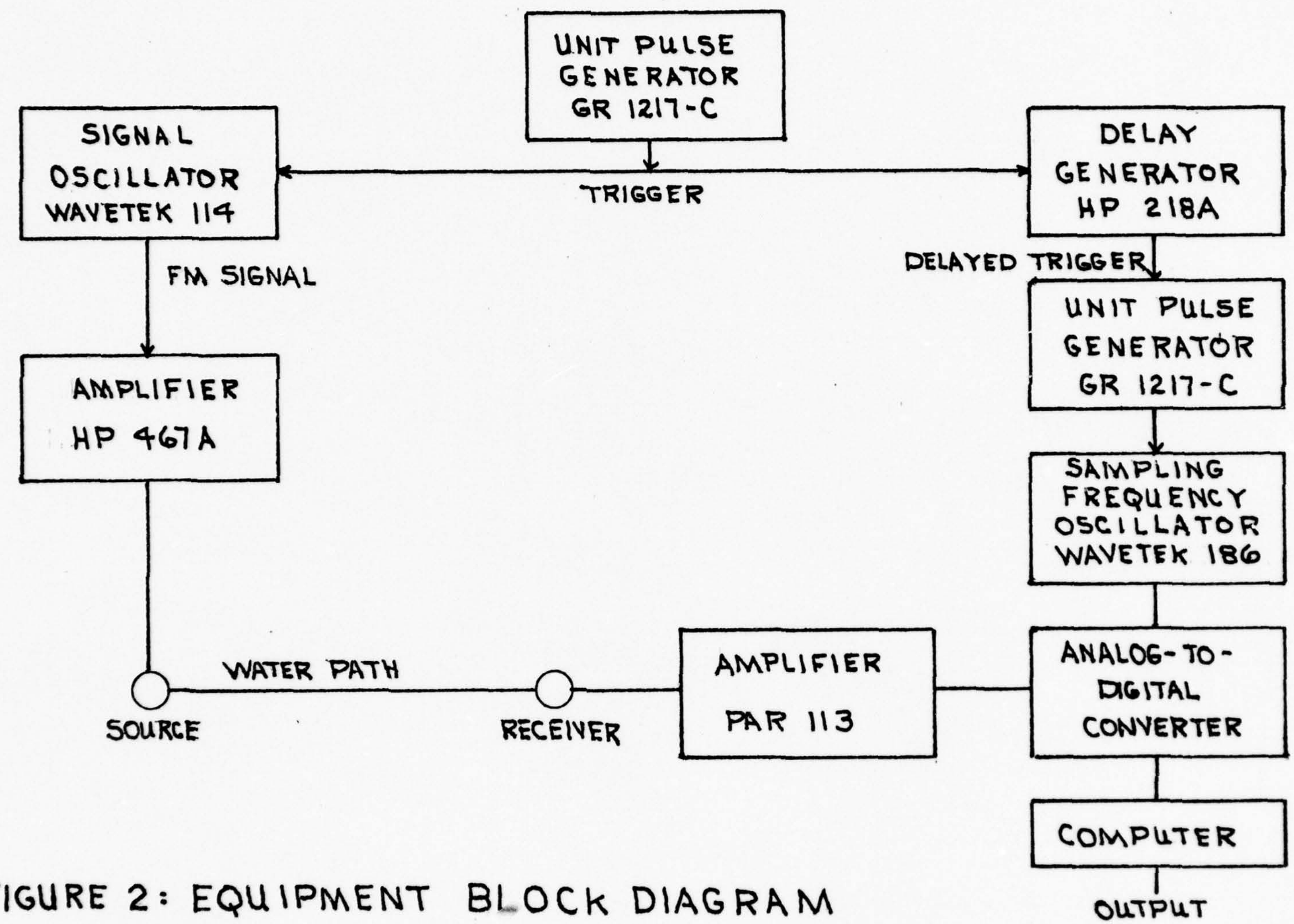


FIGURE 2: EQUIPMENT BLOCK DIAGRAM

converter. The A/D converter was triggered through the delay generator via the unit pulse generator and the sampling frequency oscillator. A sampling frequency of 320 kHz was used. The delay generator delays the trigger by the time required for the signal to be transmitted through the water and then this delayed pulse triggers the unit pulse generator. The unit pulse generator was used to determine the total on-time of the sampling frequency oscillator. The oscillator determined the samples per second taken by the converter during the oscillator's on-time. This complex equipment set-up was designed to allow the A/D converter to sample only during the time when the direct and reflected signals were present, thereby reducing the amount of computer time and storage required to process the data. Reflections from other surfaces in the tank were eliminated wherever possible in the sampling time by varying the pulse repetition rate of the master pulse generator or by varying the source and receiver placement. After the analog signal was changed by the converter to digital form, it was stored on cassette memory and then analyzed using the programs AUTOPEAK and DEVERB.

IV. DATA PROCESSING AND RESULTS

The autocorrelation plots of reverberation for two different sweep widths are seen in figure 3 with the scale factor for the delay time equal to 3.125 μ seconds. The plot of the 90 kHz sweep width shows a steeper slope of the envelope, thus a more clearly defined peak than the one with only a 10 kHz sweep width. This indicates, as expected, that as the difference between the upper and lower frequencies decreases, the correlation peaks become harder to determine. In the limit of only one frequency being present, there would be no peaks in the envelope. Figure 4 shows the range error (computer determination compared to direct measurement) versus the ratio of the upper frequency to the lower one. The graph indicates that for a ratio above 1.2 to 1 the frequency sweep of the signal is sufficient to get accurate time differences for the reflections and thereby to determine the source range and depth from the autocorrelation processes.

The frequency deverberation program is designed to give a true spectrum of the signal by eliminating frequency dependent reverberations. Since the signal is time-varying, a small block of time is desirable to keep the change in the signal to a minimum during the block. The limiting factor to the minimum block size is the desired frequency resolution. For a spectrum to accurately represent an instant of time rather than being a spectrum averaged over a length of

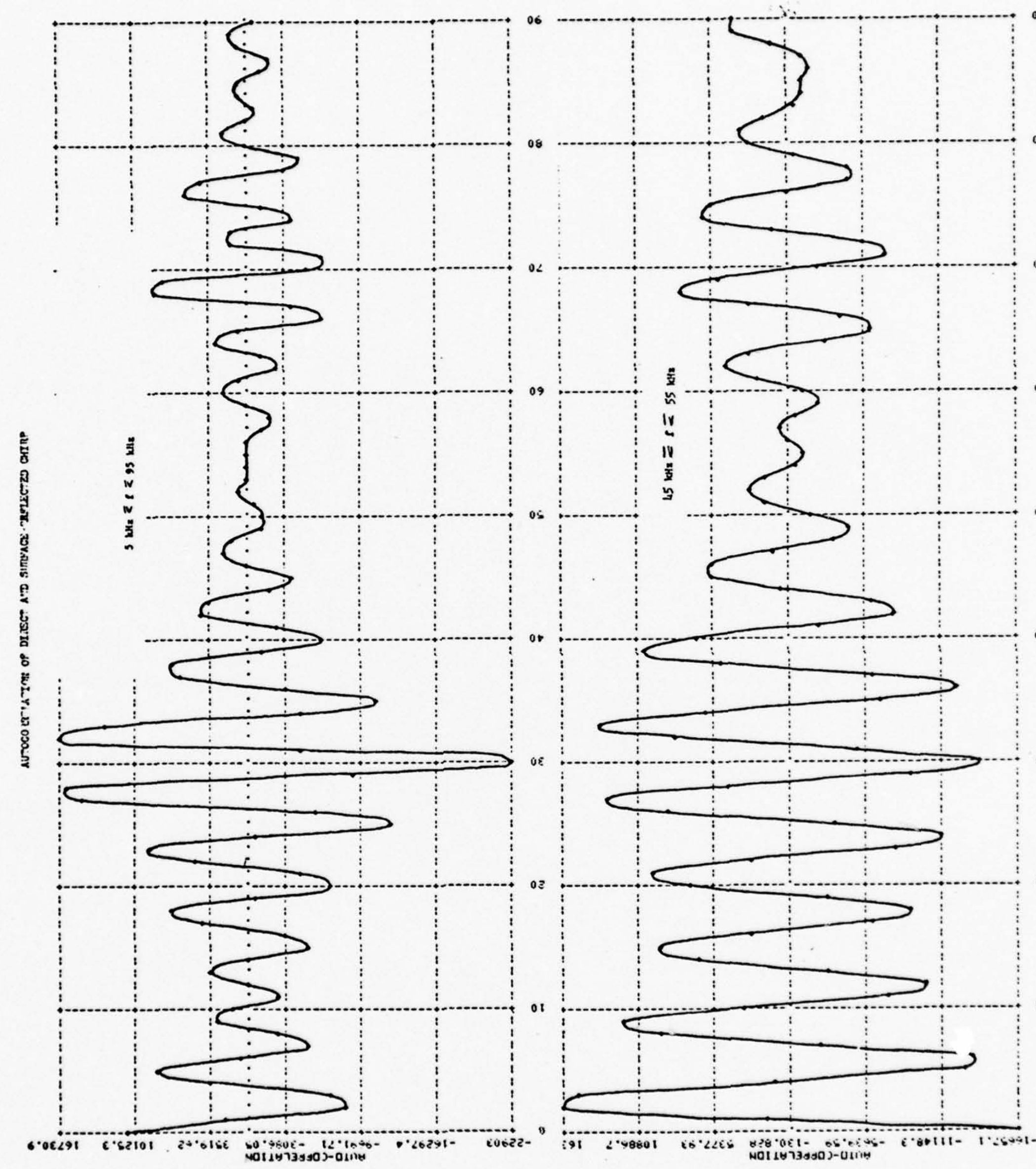


FIGURE 3.

AUTOCORRELATION OF DIRECT AND SURFACE REFLECTED CHIRP

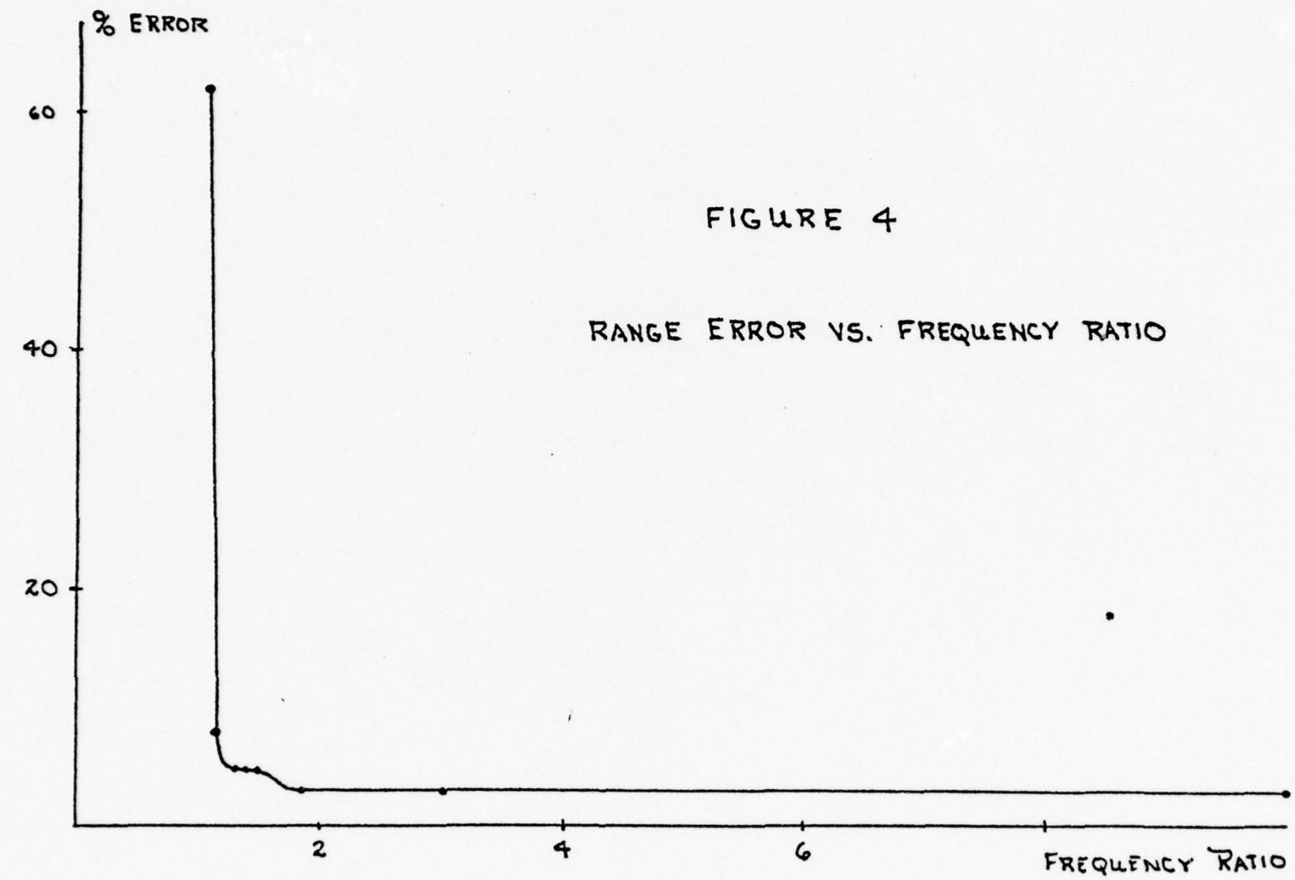


FIGURE 4

RANGE ERROR VS. FREQUENCY RATIO

time, the frequency change during the block time should be much less than the frequency resolution. In addition, where possible, the block duration should be equal to, or a sub-multiple of $|T_s - T_b|$ in order to permit equation (18) to be applied.

From figure 4, it is known that depth and range can be determined only when the frequency ratio of the chirp is greater than 1.2 to 1. Figure 5 shows the reverberant and deverberated signals for a chirp from 46 kHz to 54 kHz with a ratio of 1.17 to 1. The reverberant signal, top of figure 5, was divided into blocks, equal to $\tau_B - \tau_S$, and then transformed back into the time domain which is shown at the bottom of the figure. Some improvement can be seen but the expected slowly increasing frequency at the approximately constant amplitude has not been realized. It is believed that the inadequate deverberation is due to the fact that the frequency sweep during a block is approximately four-tenths of the frequency resolution. A slower sweep rate or a larger block duration would have cured this problem. However, a slower sweep rate for the model would have decreased the accuracy of the range determined by the autocorrelation; and a large block duration was precluded by the geometry which determined $|\tau_B - \tau_S|$. The temporal deverberation technique which is presented next did not suffer from these limitations.

The result of applying the temporal deverberation program to a signal with a sweep width from 5 kHz to 95 kHz can be seen in figure 6 with the reverberant signal on the top and

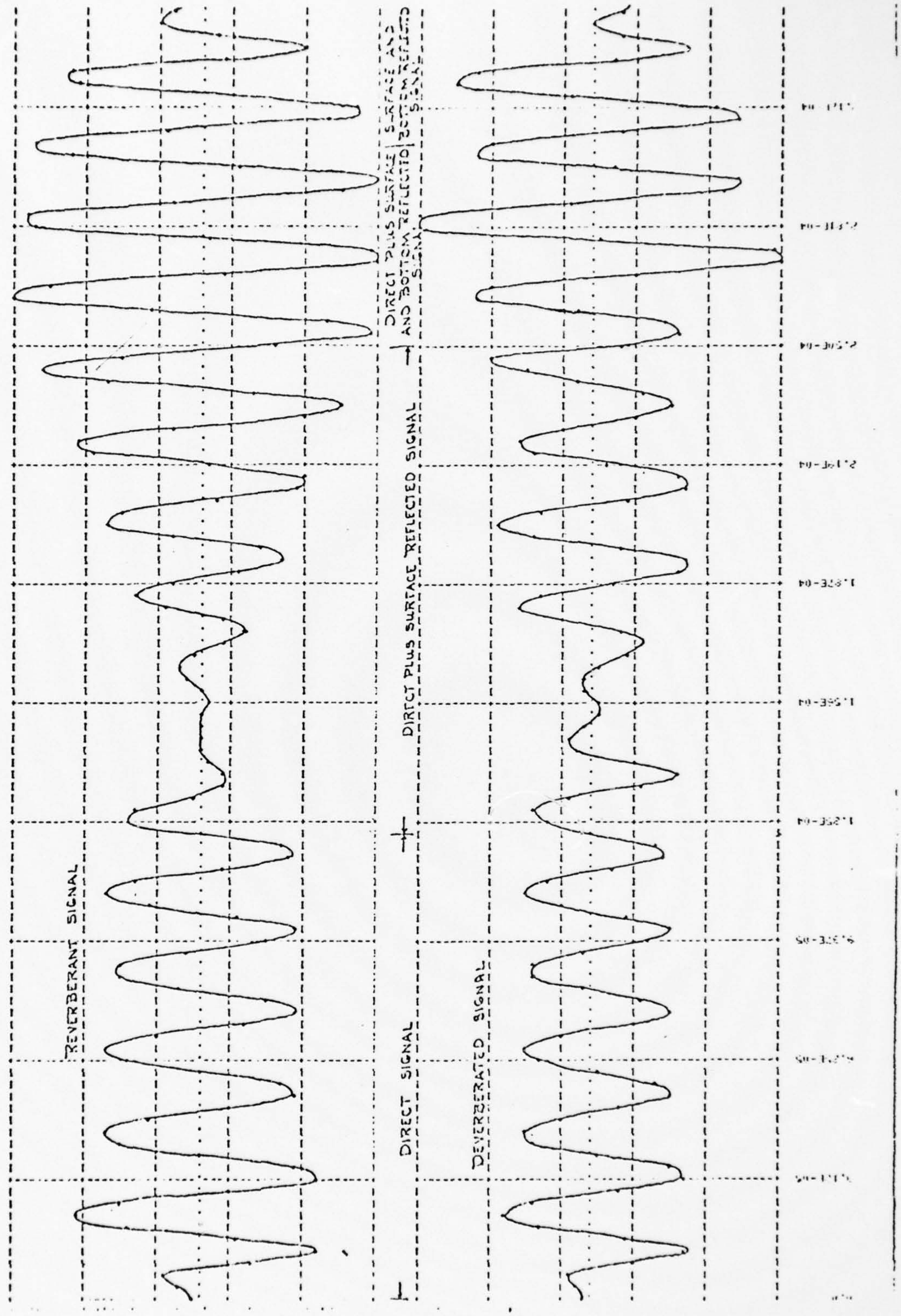


FIGURE 5.

FREQUENCY DEVERBERATION

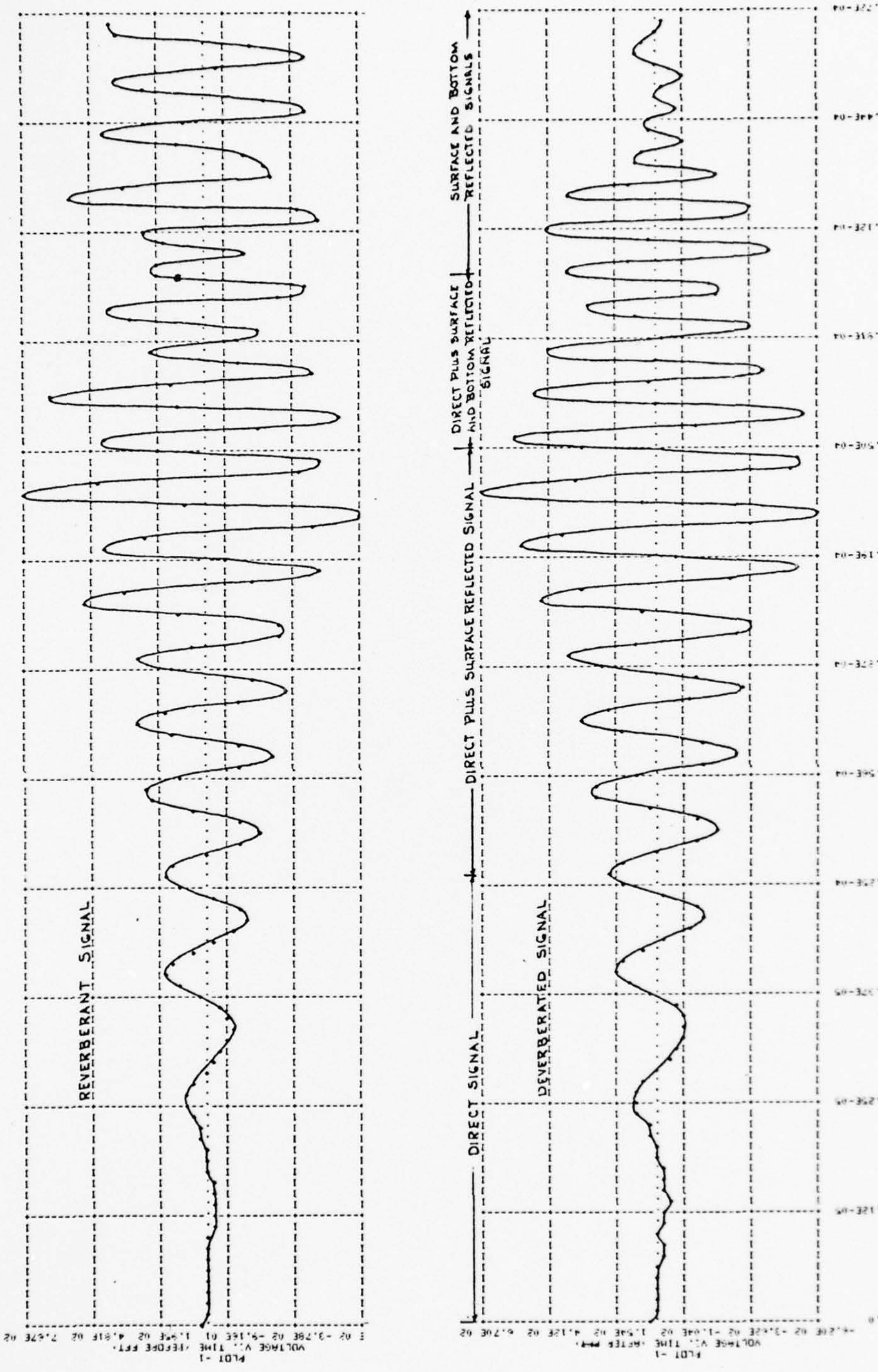


FIGURE 6

TEMPORAL DEVERBERATION

the deverberant signal below. Since the source was resonant at 64 kHz, the FM sweep grows in amplitude to 64 kHz and then decreases. The deverberated signal shows the frequency and amplitude modulation cleanly until the end of the direct signal. The reverberation after that time is due to scattering from the side walls of the tank. Possible reverberation due to the fourth and later terms on the right hand side of equation (22) were excluded by limiting the duration of sampling by the computer.

V. CONCLUSIONS

The work described in this thesis demonstrates the feasibility of obtaining a non-reverberant spectrum of a transient source in a reverberant environment. The technique includes calculation of the autocorrelation of the received signal to determine range and depth of the source and computer processing to correct for the surface and bottom reflections.

The autocorrelation function provides an accurate method for obtaining the range and depth of a source of transient sound in shallow water. The correlation technique can be performed for a chirp sound with ratio of upper to lower frequency of greater than 1.2 to 1. At least two reflections are required to obtain the depth and range of the source with respect to the receiver.

Frequency and time deverberation programs which use the position data from the autocorrelation have been developed to eliminate the reverberations and, thereby, to obtain corrected spectra or corrected time plots. Either technique can be used; however, because the output of the computer is a time series, it is natural to apply temporal deverberation. This becomes very simple if the surface and bottom reflection coefficients are independent of frequency.

COMPUTER PROGRAM AUTOPEAK

```
1 REM     DATE OF LAST CORRECTION:  10/31/77
2 REM
5 REM     *******************************************
10 REM    *    AUTOPEAK  --  9/29/77                 *
15 REM    *    JEANIE SAVAGE, PROGRAMMER             *
20 REM    *    SPECIFICATIONS BY RICK BOSTIAN        *
25 REM    *    THIS PROGRAM PERFORMS AN AUTO-        *
30 REM    *       CORRELATION ON SIGNAL DATA AND     *
35 REM    *       THEN PICKS OUT THE TWO PEAK        *
40 REM    *       VALUES.                            *
45 REM    *******************************************
50 REM
55 REM
100 DIM V$(48),V1$(6),Z$(1)
105 DIM A(1000),B(1000),Y(1000)
107 DIM Z1$(1),Z2$(1),S(1),D(1),Z4$(1)
108 DIM P(500,1),Z3$(1),M(50,2),R(1,2)
110 REM
112 Z4$="N"
113 Z$="Y"
115 REM          ***DRIVER ROUTINE***
120 REM          PERFORM INITIALIZATION PROCEDURE
125 GOSUB 500
127 IF Z4$="Y" GOTO 140
130 REM          READ DATA FROM TAPE
135 GOSUB 1000
140 REM          BUILD OTHER ARRAY
145 GOSUB 1300
150 REM          PERFORM AUTO-CORRELATION
155 GOSUB 1500
160 REM          DETERMINE INTERMEDIATE PEAK VALUES
165 GOSUB 2000
170 IF Z3$="N" GOTO 185
175 REM          PRINT INTERMEDIATE VALUES
180 GOSUB 2500
185 REM          DETERMINE TWO PEAK VALUES
190 GOSUB 3000
192 IF Z8=1 GOTO 150
205 REM          PRINT TIME DIFFERENCES
210 GOSUB 4000
215 REM          CALCULATE SOURCE DEPTH, S-R RANGE
220 GOSUB 4500
225 REM          PRINT DEPTH AND RANGE
230 GOSUB 5000
235 REM          CALCULATE REFLECTION PATH DISTANCES
240 GOSUB 5500
245 REM          PRINT DISTANCES
250 GOSUB 6000
255 PRINT "ARE YOU FINISHED? (Y OR N)--"
260 INPUT Z$
265 IF Z$="Y" THEN STOP
267 PRINT "SAME DATA? (Y OR N)--"
268 INPUT Z4$
270 PRINT "SAME PARAMETERS? (Y OR N)--"
275 INPUT Z$
280 IF Z$="N" GOTO 115
285 Z2$="Y"
290 GOTO 127
295 REM
500 REM          ***INITIALIZATION PROCEDURE***
503 IF Z$="N" GOTO 525
505 PRINT "AUTOPEAK"
510 PRINT
512 PRINT "SAMPLING FREQUENCY MUST BE FOUR TIMES THE"
513 PRINT "  GREATEST FREQUENCY OF INTEREST."
514 PRINT
515 PRINT "NUMBER OF POINTS PER BLOCK--"
520 INPUT N2
525 PRINT "MINIMUM TIME DIFFERENCE BETWEEN DIRECT PATH"
527 PRINT "  AND FIRST REFLECTED PATH RECEPTION (IN"
528 PRINT PRINT "  SAMPLES)--"
530 INPUT D8
```

```
535 PRINT "NUMBER OF POINTS TO BE USED FROM EACH BLOCK"
540 PRINT "    (NUMBER OF POINTS PLUS DELAY MUST BE LESS"
545 PRINT "    THAN NUMBER OF POINTS IN BLOCK)--"
550 INPUT N1
555 PRINT "MAXIMUM LAG--"
560 INPUT L1
562 PRINT  "PRINT PEAK VALUES? (Y OR N)--"
563 INPUT Z3$
565 IF Z$="N" THEN PRINT "CONTINUING WITH CALCULATIONS"
570 IF Z$="N" THEN RETURN
575 PRINT "IS THIS THE 1ST BLOCK OF A MULTIPLE RUN? (Y/N)--"
580 INPUT Z2$
585 PRINT "SAMPLING FREQUENCY--"
590 INPUT S1
595 PRINT "SPEED OF SOUND (M/SEC)--"
600 INPUT C
605 PRINT "DEPTH OF WATER (M)--"
610 INPUT H
615 PRINT "DEPTH OF RECEIVER (M)--"
620 INPUT R
625 RETURN
630 REM
1000 REM         ***ROUTINE TO READ DATA FROM TAPE***
1004 PRINT "SWITCH TO HIGH SPEED."
1005 IF Z2$="Y" THEN INPUT ON (2) V$
1010 K1=0
1015 FOR I=1 TO (N2/8)
1020 INPUT ON (2) V$
1025 FOR J=1 TO 48 STEP 6
1030 V1$="      "
1032 V1$(1,6)=""
1035 FOR K=0 TO 5
1040 IF V$(J+K,J+K)=" " GOTO 1050
1045 V1$=V1$+V$(J+K,J+K)
1050 NEXT K
1055 A(K1)=VAL(V1$)
1060 K1=K1+1
1065 NEXT J
1070 NEXT I
1080 PRINT "CONTINUE?"
1085 INPUT Z1$
1090 PRINT "CONTINUING WITH CALCULATIONS."
1095 RETURN
1100 REM
1300 REM         ***BUILD OTHER ARRAY***
1305 FOR I=0 TO N1-1
1310 B(I)=A(I+D8)
1315 NEXT I
1320 RETURN
1325 REM
1500 REM         ***CROSS-CORRELATION ROUTINE***
1505 A8=0
1510 B8=0
1515 FOR I=0 TO N1-1
1520 A8=A8+A(I)
1525 B8=B8+B(I)
1530 NEXT I
1535 A8=A8/N1
1540 B8=B8/N1
1545 FOR I=0 TO L1
1550 N9=N1-I
1555 S8=0
1560 FOR J=0 TO N9-1
1565 I9=J+I
1570 S8=S8+(A(J)-A8)*(B(I9)-B8)
1575 NEXT J
1580 Y(I)=S8/N9
1585 NEXT I
1590 RETURN
1595 REM
2000 REM         ***DETERMINE INTERMEDIATE PEAK VALUES***
2005 IF Y(1)>Y(0) THEN E=1
```

```
2010 IF Y(1)<Y(0) THEN E=-1
2015 P9=-1
2020 FOR I=2 TO L1
2025 IF E=(-1) GOTO 2060
2030 IF Y(I)>Y(I-1) GOTO 2085
2035 P9=P9+1
2040 P(P9,0)=I-1+D8
2045 P(P9,1)=Y(I-1)
2050 E=-E
2055 GOTO 2085
2060 IF Y(I)<Y(I-1) GOTO 2085
2065 P9=P9+1
2070 P(P9,0)=I-1+D8
2075 P(P9,1)=Y(I-1)
2080 E=-E
2085 NEXT I
2090 RETURN
2095 REM
2500 REM         ***PRINT INTERMEDIATE PEAK VALUES***
2505 PRINT
2510 PRINT "--PEAK VALUES--"
2515 PRINT
2520 FOR I=0 TO P9
2525 IF I/5=INT(I/5) GOTO 2540
2530 PRINT P(I,1),
2535 GOTO 2550
2540 PRINT P(I,1)
2545 PRINT
2550 NEXT I
2555 PRINT
2560 RETURN
2565 REM
3000 REM         ***DETERMINE TWO PEAK VALUES***
3003 Z8=0
3005 E=0
3010 K1=-1
3015 FOR I=0 TO P9-3
3020 IF E=1 GOTO 3040
3025 IF ABS(P(I,1))>ABS(P(I+1,1)) OR ABS(P(I,1))>
        ABS(P(I+2,1)) GOTO 3075
3030 IF ABS(P(I,1))> ABS(P(I+3,1)) GOTO 3075
3035 E=1
3040 IF ABS(P(I,1))<ABS(P(I+1,1) OR ABS(P(I,1))<
       ABS(P(I+2,1)) GOTO 3075
3045 IF ABS(P(I,1))<ABS(P(I+3,1)) GOTO 3075
3050 E=0
3055 K1=K1+1
3060 M(K1,0)=P(I,0)/S1
3065 M(K1,1)=P(I,1)
3070 M(K1,2)=I
3075 NEXT K
3076 IF K1>=0 GOTO 3081
3077 PRINT "CHANGE LAG -- MAXIMUM LAG="
3078 INPUT L1
3079 Z8=1
3080 RETURN
3081 REM    DETERMINE TWO HIGHEST PEAKS
3082 M8=M(0,1)
3085 I9=0
3090 FOR I=1,K1
3095 IF ABS(M(I,1))>ABS(M8) THEN I9=I
3100 IF ABS(M(I,1))> ABS(M8) THEN M8=M(I,1)
3105 NEXT I
3110 R(0,0)=M8
3115 R(0,1)=M(I9,0)
3120 R(0,2)=M(I9,2)
3125 I8=0
3130 IF I9=0 THEN I8=1
3135 M8=M(I8,1)
3137 IF K1=1 GOTO 3165
3140 FOR I=I8+1 TO K1
3145 IF I=I9 GOTO 3160
```

```
3150 IF ABS(M(I,1))>ABS(M8) THEN I8=I
3155 IF ABS(M(I,1))>ABS(M8) THEN M8=M(I,1)
3160 NEXT I
3165 R(1,0)=M8
3170 R(1,1)=M(I8,0)
3175 R(1,2)=M(I8,2)
3180 RETURN
3185 REM
4000 REM          ***PRINT TIME DIFFERENCES***
4005 PRINT
4010 PRINT
4015 PRINT "TIME DIFFERENCE BETWEEN DIRECT AND SURFACE
        REFLECTED PATHS"
4020 PRINT "-----------------------------------------
        ---------------"
4025 PRINT TAB(9);"PEAK VALUE"
4027 A3=R(0,1)*1000
4030 PRINT USING "       @@.@@@@@ MSEC",A3
4035 PRINT
4040 PRINT
4045 PRINT "TIME DIFFERENCE BETWEEN DIRECT AND BOTTOM
        REFLECTED PATHS"
4050 PRINT "-----------------------------------------
        ---------------"
4055 PRINT TAB(9);"PEAK VALUE"
4057 A3=R(1,1)*1000
4060 PRINT USING "       @@.@@@@@ MSEC",A3
4065 PRINT
4070 PRINT
4075 RETURN
4080 REM
4500 REM          ***CALCULATE SOURCE DEPTH AND S-R RANGE***
4505 FOR I=1 TO 1
4510 G0=R(0,I)*(C*R(1,I)) 2
4515 G1=C 2*R(0,I)*R(1,I)+4*H 2-4*H*R
4520 G2=R(0,I)*G1
4525 G3=4*(R(0,I)*(R-H)-R*R(1,I))
4530 S(I-1)=(G0-G2)/G3
4535 D(I-1)=2*R*S(I-1)/(C*R(0,I))-C*R(0,I)/2
4540 NEXT I
4545 RETURN
4550 REM
5000 REM          ***PRINT DEPTH AND DISTANCE VALUES***
5005 PRINT "RANGE OF SOURCE FROM RECEIVER AND SOURCE DEPTH
        IN METERS:"
5010 PRINT "---------------------------------------------
        ----------"
5015 PRINT TAB(9);"PEAK VALUES"
5020 PRINT TAB(5);"DEPTH";TAB(19);"RANGE"
5030 PRINT USING "   @@.@@@@@   ",S(0),D(0)
5035 PRINT
5040 PRINT
5045 PRINT
5050 RETURN
5055 REM
5500 REM          ***CALCULATE DISTANCES OF REFLECTED PATHS**
5505 REM     CALCULATE TRANSVERSE SOURCE-RECEIVER DISTANCE
5510 T=SQR(D(0) 2-(R-S(0)) 2)
5515 REM     CALCULATE DISTANCE OF SURFACE REFLECTION PATH
5520 X=S(0)*T/(R+S(0))
5525 Y=R*T/(R+S(0))
5530 U=SQR(X 2+S(0) 2)
5535 W=SQR(Y 2+R 2)
5540 D0=U+W
5545 REM     CALCULATE DISTANCE OF BOTTOM REFLECTION PATH
5550 A=(H-S(0))*T/(2*H-R-S(0))
5555 B=(H-R)*T/(2*H-R-S(0))
5560 E=SQR(A 2+(H-S(0)) 2)
5565 F=SQR(B 2+(H-R) 2)
5570 D1=E+F
5575 RETURN
5580 REM
```

```
6000 REM          ***PRINT PATH DISTANCES***
6005 PRINT "SURFACE REFLECTION PATH DISTANCE IN METERS:"
6010 PRINT "-------------------------------------------"
6015 PRINT USING "     @@@.@@@@@",D0
6020 PRINT
6025 PRINT
6030 PRINT "BOTTOM REFLECTION PATH DISTANCE IN METERS:"
6035 PRINT "------------------------------------------"
6040 PRINT USING "     @@@.@@@@@",D1
6045 PRINT
6050 PRINT
6055 RETURN
6060 REM
6065 END
```

```
C       ***********************************
C       *   DEVERB      10/21/77          *
C       *   JEANIE SAVAGE, PROGRAMMER     *
C       *   SPECIFICATIONS BY RICK BOSTIAN *
C       *   IN THIS PROGRAM DEVERBERATION *
C       *       IS PERFORMED IN THE FRE-  *
C       *       QUENCY DOMAIN.            *
C       *   LAST CORRECTION:  12/06/77    *
C       ***********************************
C
        INTEGER XGRID
        INTEGER*2 IZ2,IZ3,IZ4,IZ5,IZ6,YES,IZ7
        DATA YES/'Y '/
        DIMENSION ISTACK(20),A(1000),B(1000),IARY(1000)
        DIMENSION X(1000),Y(1000)
        DIMENSION FINAL(1000,2),IBEG(20),IEND(20),ICASE(20)
        COMMON SF,ISF,IBM,THETA,SIGMA,GAMMA,C,D,RCOEFF,N2,
      &ISIZE,DB,DS,NBLK,IZ3,IZ4
C
C             ***********************
C             INITIALIZATION ROUTINE
C             ***********************
C
        IREAD=0
        PRINT 500
 500     FORMAT('0','DEVERB')
        PRINT 510
 510     FORMAT('0','NUMBER OF POINTS PER SIGNAL (POWER OF 2)',
      &' (I5)--')
        READ 520, N2
 520     FORMAT(I5)
        PRINT 530
 530     FORMAT(' ','IS THIS THE FIRST SIGNAL OF A MULTIPLE',
      &' RUN? (Y/N)-')
        READ 540,IZ2
 540     FORMAT(A1)
        PRINT 550
 550     FORMAT(' ','SAMPLING FREQUENCY (F9.3)--')
        READ 560, SF
 560     FORMAT(F9.3)
        PRINT 570
 570     FORMAT(' ','DIRECT PATH DISTANCE (F9.5)--')
        READ 580, D
 580     FORMAT(F9.5)
        PRINT 590
 590     FORMAT(' ','SURFACE PATH DISTANCE IN METERS (F9.5)--')
        READ 580,DS
        PRINT 600
 600     FORMAT(' ','BOTTOM PATH DISTANCE IN METERS (F9.5)--')
        READ 580, DB
        PRINT 610
 610     FORMAT(' ','SURFACE REFLECTION TIME IN MSEC (F9.5)--')
        READ 580, TS
        TS=TS/1000
        PRINT 620
 620     FORMAT(' ','BOTTOM REFLECTION TIME IN MSEC (F9.5)--')
        READ 580, TB
        TB=TB/1000
        PRINT 630
 630     FORMAT(' ','BOTTOM REFLECTION COEFFICIENT (F9.5)--')
        READ 580, RCOEFF
        WRITE(6,711)
 711     FORMAT(' ','RMS WAVE HEIGHT (F9.5)--')
        READ 580, SIGMA
        PRINT 720
 720     FORMAT(' ','SURFACE ANGLE IF INCIDENCE (IN RADIANS)',
      &' (F9.5)--')
        READ 580, THETA
        PRINT 730
 730     FORMAT(' ','BOTTOM PHASE SHIFT (F9.5)--')
        READ 580, GAMMA
        PRINT 640
```

```
640    FORMAT(' ','SPEED OF SOUND (F9.3)--')
      READ 560, C
      PRINT 650
650    FORMAT(' ','FREQUENCY PLOT? (Y/N)--')
      READ 540,IZ3
      PRINT 655
655    FORMAT(' ','TIME PLOT BEFORE FFT? (Y/N)--')
      READ 540,IZ7
      IF(IZ7.NE.YES) GOTO 659
      PRINT 657
657    FORMAT(' ','NUMBER OF POINTS TO BE PLOTTED (I5)--')
      READ 520, INUM1
659    PRINT 660
660    FORMAT(' ','TIME PLOT AFTER FFT? (Y/N)--')
      READ 540,IZ4
      IF(IZ4.NE.YES) GOTO 672
      PRINT 657
      READ 520,INUM2
672    PRINT 675
675    FORMAT(' ','ALL BLOCKS? (Y/N)--')
      READ 540,IZ5
      IF(IZ5.EQ.YES) GOTO 1000
      KPTR=0
      PRINT 680
680    FORMAT(' ','SPECIFIC BLOCKS (I2) (INPUT 99 WHEN',
     &' FINISHED)--')
690    READ 700,ITEMP
700    FORMAT(I2)
      IF(ITEMP.EQ.99) GOTO 1000
      KPTR=KPTR+1
      ISTACK(KPTR)=ITEMP
      GOTO 690
C
C             **************
C             READ DATA TAPE
C             **************
C
1000   IF(IREAD.EQ.1) GOTO 2000
      PRINT 1005
1005   FORMAT(' ','READY TO READ DATA TAPE')
      IF(IZ2.NE.YES) GOTO 1020
      READ (5,1010) KEND
1010   FORMAT(8I6)
1020   DO 1030 I=1,N2,8
      ITEMP=I+7
      READ(5,1010) (IARY(J),J=I,ITEMP)
1030   CONTINUE
C
C             ******************************
C             DETERMINE DIVISION BOUNDARIES
C             ******************************
C
      PRINT 1501
1501   FORMAT(' ','CONTINUING WITH CALCULLATIONS')
1500   DO 1510 I=1,N2
      TIME=(I-1)/SF
      IF(TIME.LT.TS) GOTO 1510
      ISF=I
      GOTO 1520
1510   CONTINUE
      ISF=N2+1
      GOTO 1540
1520   DO 1530 I=1,N2
      TIME=(I-1)/SF
      IF(TIME.LT.TB) GOTO 1530
      IBM=I
      GOTO 1550
1530   CONTINUE
1540   IBM=N2+1
C
C      ***DETERMINE NUMBER OF POINTS IN EACH SECTOR***
1550   NPT1=ISF-1
```

```
      NPT2=IBM-ISF
      IF(IBM.LT.ISF) NPT2=ISF-IBM
      NPT3=N2-IBM+1
      IF(IBM.LT.ISF) NPT3=N2-ISF+1
C
C             ********************
C             DETERMINE BLOCK SIZE
C             ********************
C
C     ***FIND SMALLEST AND MIDDLE NUMBER OF POINTS IN SECTOR
 1700  ISMALL=MINO(NPT1,NPT2)
      ISMALL=MINO(NPT3,ISMALL)
      IF(ISMALL.EQ.NPT1) MIDDLE=MINO(NPT2,NPT3)
      IF(ISMALL.EQ.NPT2) MIDDLE=MINO(NPT1,NPT3)
      IF(ISMALL.EQ.NPT3) MIDDLE=MINO(NPT1,NPT2)
C
C     ***DETERMINE NUMBER OF POINTS PER BLOCK***
 2000 IF(IABS(MIDDLE/2-ISMALL).GT.(MIDDLE-ISMALL))
     &ISIZE=ISMALL
      IF(ISIZE.EQ.ISMALL) GOTO 2200
      DO 2010 K=1,30
      IF(ISMALL.EQ.MIDDLE/K) GOTO 2020
      IF((ISMALL.EQ.MIDDLE/K).LT.(ISMALL-MIDDLE/(K+1)))
     &GOTO 2020
 2010  CONTINUE
 2020  ISIZE=MIDDLE/K
C
C     ***DETERMINE NUMBER OF BLOCKS PER SECTOR***
 2200  NBLK1=NPT1/ISIZE
      NBLK2=NPT2/ISIZE
      NBLK3=NPT3/ISIZE
C
C     ***DETERMINE NUMBER OF POINTS SKIPPED EACH SECTOR***
 2300  NSKIP1=NPT1-NBLK1*ISIZE
      NSKIP2=NPT2-NBLK2*ISIZE
      NSKIP3=NPT3-NBLK3*ISIZE
C
C     ***PRINT SECTOR INFORMATION***
      WRITE(6,2405)  ISIZE
 2405  FORMAT('0','ISIZE=',I3)
      IT1=1
      IT2=2
      IT3=3
      WRITE(6,2410) IT1,NBLK1,NSKIP1
 2410 FORMAT('0','SECTOR ',I1,' CONTAINS ',I2,' BLOCKS,',
     $2X,I3,' POINTS SKIPPED')
      WRITE(6,2410) IT2,NBLK2,NSKIP2
      WRITE(6,2410) IT3,NBLK3,NSKIP3
C
C             ***********************************
C             BUILD STACK IF PROCESSING ALL BLOCKS
C             ***********************************
C
2500  IF(IZ5.NE.YES) GOTO 2700
      NUMBLK=N2/ISIZE
      DO 2510 I=1,NUMBLK
      ISTACK(I)=I
2510  CONTINUE
      KPTR=NUMBLK
C
C             *******************************
C             DETERMINE BEG & END POINTS & CASE
C             *******************************
C
2700  I6=0
C
C     ***BLOCKS IN 1ST SECTOR***
      ITEMP=NSKIP1+1
      DO 2710 I=1,NBLK1
      I6=I6+1
      IBEG(I6)=ITEMP
      IEND(I6)=ITEMP+ISIZE-1
```

```
      ICASE(I6)=1
      ITEMP=IEND(I6)+1
 2710  CONTINUE
C
C     ***BLOCKS IN 2ND SECTOR***
      ITEMP=ISF
      ITEMP1=NSKIP2
      DO 2720 I=1,NBLK2
      I6=I6+1
      IBEG(I6)=ITEMP
      IEND(I6)=ITEMP+ISIZE-1
      ITEMP=IEND(I6)+1
      ICASE(I6)=2
      IF(ISF.GT.IBM) ICASE(I6)=3
      IF(ITEMP2.EQ.0) GOTO 2720
      ITEMP=ITEMP+1
      ITEMP1=ITEMP1-1
2720  CONTINUE
C
C     ***BLOCKS IN 3RD SECTOR***
      DO 2730 I=1,NBLK3
      I6=I6+1
      IBEG(I6)=IBM+(I-1)*ISIZE
      IF(ISF.GT.IBM) IBEG(I6)=ISF+(I-1)*ISIZE
      IEND(I6)=IBEG(I6)+ISIZE-1
      ICASE(I6)=4
2730  CONTINUE
C
C             ***************************
C             PRINT TIME PLOT BEFORE FFT
C             ***************************
C
2900  IF(IZ7.NE.YES) GOTO 3000
      IF(INUM1.GT.1000) GOTO 2910
      NUM=INUM1
      KPLOTS=1
      GOTO 2920
2910  KPLOTS=INUM1/1000+1
      NUM=1000
2920  DO 2940 I=1,KPLOTS
      IF((I.EQ.KPLOTS).AND.(I.NE.1)) NUM=INUM1-(KPLOTS-1)
     &*1000
      DO 2930 J=1,NUM
      Y(J)=FLOAT(IARY((I-1)*1000+J))
      X(J)=((I-1)*1000+J-1)/SF
2930  CONTINUE
      XGRID=NUM/10
      IF(MOD(NUM,10).NE.0) XGRID=XGRID+1
      CALL PLOTDV(X,Y,NUM,XGRID,3,NBLK)
2940  CONTINUE
C
C             **************
C             PROCESS BLOCKS
C             **************
C
3000  DO 3110 I=1,KPTR
      NBLK=ISTACK(I)
C
C     ***DETERMINE BOUNDARIES AND CASE OF BLOCK***
      IBEGO=IBEG(NBLK)
      IENDO=IEND(NBLK)
      ICASEO=ICASE(NBLK)
C
C     ***PROCESS BLOCK***
      PRINT 3015, IBEGO,IENDO,ICASEO
 3015 FORMAT(///' ','IBEG=',I4,4X,'IEND=',I4,4X,'ICASE=',I1)
      K=1
      DO 3020 J=IBEGO,IENDO
      A(K)=FLOAT(IARY(J))
      B(K)=0.0
      K=K+1
3020  CONTINUE
```

```
      CALL SIGNAL(A,B,FINAL,IBEGO,ICASEO,ISIZE)
      IF((I.NE.1).OR.(NSKIP1.EQ.0)) GOTC 3040
      DC 3030 J=1,NSKIP1
      SLOPE=FINAL(IBEG(I)+1,1)-FINAL(IBEG(I),1)
      FINAL(J,1)=-SLOPE+FINAL(IBEG(I),1)
      SLOPE=FINAL(IBEG(I)+1,2)-FINAL(IBEG(I),2)
      FINAL(J,2)=-SLOPE+FINAL(IBEG(I),2)
 3030 CONTINUE
 3040 IF((I.GT.NBLK1+NBLK2).OR.(I.LE.NBLK1+1)) GOTO 3110
      IF(NSKIP2.EQ.0) GOTO 3110
      DO 3050 J=1,NSKIP2
      ITEMP=IEND(NBLK1+I)+1
      TEMP=FINAL(ITEMP+1,1)-FINAL(ITEMP-1,1)
      FINAL(ITEMP,1)=FINAL(ITEMP-1,1)+TEMP
      TEMP=FINAL(ITEMP+1,2)-FINAL(ITEMP-1,2)
      FINAL(ITEMP,2)=FINAL(ITEMP-1,2)+TEMP
 3050 CONTINUE
 3110 CONTINUE
C
C            *************************
C            PRINT TIME PLOT AFTER FFT
C            *************************
C
3300  IF(IZ4.NE.YES) GOTO 3500
      DO 3308 I=1,KPTR
      NBLK=ISTACK(I)
      N=ISIZE
      IBEGO=IBEG(NBLK)
      IENDO=IEND(NBLK)
      K=1
      DC 3304 J=IBEGO,IENDO
      A(K)=FINAL(J,1)
      B(K)=FINAL(J,2)
      K=K+1
 3304 CONTINUE
      CALL CFFT2(A,B,N,N,N,1)
      DO 3306 J=1,ISIZE
      FINAL(IBEGO+J-1,1)=A(J)
 3306 CONTINUE
 3308 CONTINUE
      IF(NSKIP1.EQ.0) GOTO 3313
      DC 3312 I=1,NSKIP1
      FINAL(I,1)=(FINAL(IBEG(1),1)/(NSKIP1+1))*I
 3312 CONTINUE
 3313 IF(NSKIP2.EQ.0) GOTO 3315
      DO 3314 I=1,NSKIP2
      ITEMP=(NBLK1+I)+1
      TEMP=FINAL(ITEMP+1,1)-FINAL(ITEMP-1,1)
      FINAL(ITEMP,1)=FINAL(ITEMP-1,1)+TEMP
 3314 CCNTINUE
 3315 IF(INUM2.GT.IEND(KPTR))INUM2=IEND(KPTR)
      IF(INUM2.GT.1000) GOTC 3310
      NUM=INUM2
      KPLOTS=1
      GOTO 3320
3310  KPLOTS=INUM2/1000+1
      NUM=1000
3320  DO 3340 I=1,KPLOTS
      IF((I.EQ.KPLOTS).AND.(I.NE.1)) NUM=INUM2-(KPLCTS-1)
     &*1000
      DC 3330 J=1,NUM
      Y(J)=FINAL((I-1)*1000+J,1)
      Y(J)=Y(J)/FLOAT(ISIZE)
      X(J)=((I-1)*1000+J-1)/SF
3330  CONTINUE
      XGRID=NUM/10
      IF(MOD(NUM,10).NE.0) XGRID=XGRID+1
      CALL PLOTDV(X,Y,NUM,XGRID,2,NBLK)
3340  CCNTINUE
C
C            **********
C            CONCLUSION
```

```
C          **********
C
3500   PRINT 3510
3510   FORMAT(' ','ARE YOU FINISHED? (Y/N)--')
       READ 540,IZ6
       IF(IZ6.EQ.YES) STOP
       IREAD=1
       GOTO 672
       DEBUG SUBCHK
       END
C
C
C                        ****SIGNAL SUBROUTINE****
C
       SUBROUTINE SIGNAL(A,B,FINAL,IBEGO,ICASEO,N)
       DIMENSION FINAL(1000,2)
       DIMENSION X(1000),Y(1000),A(1000),B(1000)
       INTEGER*2 YES,IZ3,IZ4
       COMMON SF,ISF,IBM,THETA,SIGMA,GAMMA,C,D,RCOEFF,N2,
      &ISIZE,DB,DS,NBLK,IZ3,IZ4
       INTEGER XGRID
       REAL KO
       DATA YES/'Y '/
C
C          ***********
C          PERFORM FFT
C          ***********
C
100    CALL CFFT2(A,B,N,N,N,-1)
C
C          *******************
C          PERFORM CORRECTIONS
C          *******************
C
500    DO 520 I=1,N
       FINAL(IBEGO+I-1,1)=A(I)
       FINAL(IBEGO+I-1,2)=B(I)
       FREQ=(I-1)*SF/N
       PI=3.14159
       KO=2.*PI*FREQ/C
       IF(ICASEO.EQ.1) GOTO 520
       IF(ICASEO.EQ.3) GOTO 510
C
C      ***CORRECTION FOR SURFACE REFLECTION***
       G=((4.*PI*SIGMA*FREQ/C)*COS(THETA))**2
       ITEMP=(IBEGO+I-1)-(ISF-1)
       TMAG=SQRT(FINAL(ITEMP,1)**2+FINAL(ITEMP,2)**2)
       TPHASE=ATAN2(FINAL(ITEMP,2),FINAL(ITEMP,1))
       S=D/DS*EXP(-G/2.)*TMAG
       SPHASE=TPHASE-(DS-D)*KO-PI
       FINAL(IBEGO+I-1,1)=FINAL(IBEGO+I-1,1)-S*COS(SPHASE)
       FINAL(IBEGO+I-1,2)=FINAL(IBEGO+I-1,2)-S*SIN(SPHASE)
       IF(ICASEO.EQ.2) GOTO 520
C
C      ***CORRECTION FOR BOTTOM REFLECTION***
 510   ITEMP=(IBEGO+I-1)-(IBM-1)
       TMAG=SQRT(FINAL(ITEMP,1)**2+FINAL(ITEMP,2)**2)
       TPHASE=ATAN2(FINAL(ITEMP,2),FINAL(ITEMP,1))
       S=RCOEFF*D/DB*TMAG
       SPHASE=TPHASE-(DB-D)*KO+GAMMA
       FINAL(IBEGO+I-1,1)=FINAL(IBEGO+I-1,1)-S*COS(SPHASE)
       FINAL(IBEGO+I-1,2)=FINAL(IBEGO+I-1,2)-S*SIN(SPHASE)
 520    CONTINUE
C
C          ***************************
C          PRINT FREQUENCY AND TIME PLOTS
C          ***************************
C
1000   IF(IZ3.NE.YES) GOTO 1020
C
C      ***FREQUENCY PLOT***
       K=0
```

```
      ITEMP=N/2
      DO 1010 I=1,ITEMP
      K=K+1
      Y(K)=FINAL(IBEGO+I-1,1)**2+FINAL(IBEGO+I-1,2)**2
      Y(K)=10*ALOG10(Y(K)*D)
      X(K)=(K-1)*SF/N
1010  CONTINUE
      NUM=K
      XGRID=NUM/10
      IF(MOD(NUM,10).NE.0) XGRID=XGRID+1
      CALL PLOTDV(X,Y,NUM,XGRID,1,NBLK)
 1020 RETURN
      DEBUG SUBCHK
      END
C
C
C
C                          ****PLOT SUBROUTINE****
C
      SUBROUTINE PLOTDV(X,Y,N,XGRID,M,NB)
      INTEGER D,XGRID,YGRID,AXIS
      DIMENSION Y(1000),C7(101),O(6),X(1000),KAXIS(51)
      DATA IDASH/1H-/,ISTAR/1H*/,IDOT/1H./
      DATA IBAR/1HI/,IPLUS/1H+/,IBLANK/1H /,IX/1HX/
      AXIS=51
      YGRID=6
      XGRID=XGRID+1
2120  N1=N-1
      Y6=Y(1)
      Y1=Y(1)
      DO 2200 I=1,N
      IF(Y6-Y(I).GE.0.0) GOTO 2180
      Y6=Y(I)
2180  IF(Y1-Y(I).LE.0.0) GOTO 2200
      Y1=Y(I)
2200  CONTINUE
      S=Y1*(AXIS-1)/(Y6-Y1)
      X1=X(1)
      X10=X(N)
      O(1)=Y1
      O(6)=Y6
      IIX=XGRID-1
      DO 2410 I=1,IIX
      C7(I)=X((I-1)*10+1)
2410  CONTINUE
      C7(XGRID)=C7(XGRID-1)+10*(X(2)-X(1))
      IF(N.EQ.(XGRID-1)*10) C7(XGRID)=X(N)
      IIY=YGRID-1
      DO 2440 I=2,IIY
      O(I)=(FLOAT(I-1)*(Y6-Y1)/FLOAT(YGRID-1))+Y1
2440  CONTINUE
      WRITE(6,2460)
2460  FORMAT(///,' ')
      IF(M.NE.1) GOTO 2485
      PRINT 2470,NB
2470  FORMAT('0',32X,'BLOCK',1X,I2)
2485  IF(M.EQ.1) PRINT 2486
2486  FORMAT(' ',27X,'DB''S VS. FREQUENCY')
      IF(M.EQ.2) PRINT 2488
2488  FORMAT(' ',23X,'VOLTAGE VS. TIME (AFTER FFT)')
      IF(M.EQ.3) PRINT 2487
2487  FORMAT(' ',23X,'VOLTAGE VS. TIME (BEFORE FFT)')
      WRITE(6,2500) (O(I),I=1,YGRID)
2500  FORMAT(9X,11(1PE10.2))
      S1=(X10-X1)/10.0*(XGRID-1)
      D=1
      L1=1
      L=1
      IZ=IFIX(-S+1.5)
      ITEMP=(XGRID-1)*10+1
      DO 2900 I1=1,ITEMP
      IF(N.LT.I1) GOTO 2510
```

```
      YTEMP=(Y(I1)*FLOAT(YGRID-1)*10.0/(Y6-Y1))-S
      IY=IFIX(YTEMP+1.5)
 2510 IF(L1.GT.L) GOTO 2760
      DO 2650 IP=1,AXIS
      KAXIS(IP)=IDASH
 2650 CONTINUE
      DO 2680 I=1,AXIS,10
      KAXIS(I)=IPLUS
 2680 CONTINUE
      IF(N.LT.I1) GOTO 2720
      IF((Y1.LE.0.0).AND.(0.0.LE.Y6)) KAXIS(IZ)=IDOT
      KAXIS(IY)=ISTAR
 2720 WRITE(6,2725) C7(D),(KAXIS(J),J=1,AXIS)
 2725 FORMAT(1PE13.2,2X,115A1)
      L1=L1+10
      D=D+1
      GOTO 2870
 2760 DO 2780 IP=1,AXIS
      KAXIS(IP)=IBLANK
 2780 CONTINUE
      DO 2810 I=1,AXIS,10
      KAXIS(I)=IBAR
 2810 CONTINUE
      IF(N.LT.I1) GOTO 2860
      IF((Y1.LE.0.0).AND.(0.0.LE.Y6)) KAXIS(IZ)=IDOT
      KAXIS(IY)=ISTAR
 2860 WRITE(6,2865) (KAXIS(J),J=1,AXIS)
 2865 FORMAT(15X,115A1)
 2870 L=L+1
      IF(L.GT.(XGRID-1)*10+2) GOTO 2910
 2900 CONTINUE
 2910 RETURN
      DEBUG SUBCHK
      END
C
C
C                       ***FOURIER TRANSFORM SUBROUTINE***
C
      SUBROUTINE CFFT2(A,B,NTOT,N,NSPAN,ISN)
```

```
C       ***********************************
C       *   TDEVERB          12/07/77     *
C       *   JEANIE SAVAGE, PROGRAMMER     *
C       *   SPECIFICATIONS BY RICK BOSTIAN *
C       *   IN THIS PROGRAM DEVERBERATION  *
C       *       IS PERFORMED IN THE TIME   *
C       *       DOMAIN.                   *
C       *   LAST CORRECTION:  12/08/77    *
C       ***********************************
C
        INTEGER XGRID
        INTEGER*2 IZ2,IZ3,IZ4,IZ5,IZ6,YES,IZ7
        DATA YES/'Y '/
        DIMENSION ISTACK(20),A(1000),B(1000),IARY(1000)
        DIMENSION X(1000),Y(1000)
        DIMENSION FINAL(1000),IBEG(20),IEND(20),ICASE(20)
C
C             **********************
C             INITIALIZATION ROUTINE
C             **********************
C
        IREAD=0
        PRINT 500
 500    FORMAT('0','DEVERB')
        PRINT 510
 510    FORMAT('0','NUMBER OF POINTS PER SIGNAL (POWER OF 2)',
       &'  (I5)--')
        READ 520, N2
 520    FORMAT(I5)
 540    FORMAT(A1)
        PRINT 550
 550    FORMAT(' ','SAMPLING FREQUENCY (F9.3)--')
        READ 560, SF
 560    FORMAT(F9.3)
        PRINT 570
 570    FORMAT(' ','DIRECT PATH DISTANCE (F9.5)--')
        READ 580, D
 580    FORMAT(F9.5)
        PRINT 590
 590    FORMAT(' ','SURFACE PATH DISTANCE IN METERS (F9.5)--')
        READ 580,DS
        PRINT 600
 600    FORMAT(' ','BOTTOM PATH DISTANCE IN METERS (F9.5)--')
        READ 580, DB
        PRINT 610
 610    FORMAT(' ','SURFACE REFLECTION TIME IN MSEC (F9.5)--')
        READ 580, TS
        TS=TS/1000
        PRINT 620
 620    FORMAT(' ','BOTTOM REFLECTION TIME IN MSEC (F9.5)--')
        READ 580, TB
        TB=TB/1000
        PRINT 630
 630    FORMAT(' ','BOTTOM REFLECTION COEFFICIENT (F9.5)--')
        READ 580, RCOEFF
        WRITE(6,711)
 711    FORMAT(' ','RMS WAVE HEIGHT (F9.5)--')
        READ 580, SIGMA
        PRINT 720
 720    FORMAT(' ','SURFACE ANGLE OF INCIDENCE (IN RADIANS)',
       &'  (F9.5)--')
        READ 580, THETA
        PRINT 730
 730    FORMAT(' ','BOTTOM PHASE SHIFT (F9.5)--')
        READ 580, GAMMA
        PRINT 640
 640    FORMAT(' ','SPEED OF SOUND (F9.3)--')
        READ 560, C
        PRINT 645
  645   FORMAT(' ','BLOCK SIZE (I5)--')
        READ 520,ISIZE
        PRINT 650
```

```
650    FORMAT(' ','FREQUENCY PLOT BEFORE CORRECTIONS?',
     &'  (Y/N)--')
      READ 540,IZ3
      PRINT 652
 652  FORMAT(' ','FREQUENCY PLOT AFTER CORRECTIONS? (Y/N)--')
      READ 540,IZ2
      PRINT 655
655    FORMAT(' ','TIME PLOT BEFORE CORRECTIONS? (Y/N)--')
      READ 540,IZ7
      IF(IZ7.NE.YES) GOTO 659
      PRINT 657
657    FORMAT(' ','NUMBER OF POINTS TO BE PLOTTED (I5)--')
      READ 520, INUM1
659    PRINT 660
660    FORMAT(' ','TIME PLOT AFTER CORRECTIONS? (Y/N)--')
      READ 540,IZ4
      IF(IZ4.NE.YES) GOTO 672
      PRINT 657
      READ 520,INUM2
672    PRINT 675
675    FORMAT(' ','ALL BLOCKS? (Y/N)--')
      READ 540,IZ5
      IF(IZ5.EQ.YES) GOTO 1000
      KPTR=0
      PRINT 680
680    FORMAT(' ','SPECIFIC BLOCKS (I2) (INPUT 99 WHEN FINIS
690    READ 700,ITEMP
700    FORMAT(I2)
      IF(ITEMP.EQ.99) GOTO 1000
      KPTR=KPTR+1
      ISTACK(KPTR)=ITEMP
      GOTO 690
C
C           **************
C           READ DATA TAPE
C           **************
C
 1000  IF(IREAD.EQ.1) GOTO 2500
      PRINT 1005
 1005  FORMAT(' ','READY TO READ DATA TAPE')
 1010  FORMAT(8I6)
 1020  DO 1030 I=1,N2,8
      ITEMP=I+7
      READ(5,1010) (IARY(J),J=I,ITEMP)
 1030  CONTINUE
C
C           ******************************
C           DETERMINE DIVISION BOUNDARIES
C           ******************************
C
      PRINT 1501
 1501  FORMAT(' ','CONTINUING WITH CALCULATIONS')
 1500  DO 1510 I=1,N2
      TIME=(I-1)/SF
      IF(TIME.LT.TS) GOTO 1510
      ISF=I
      GOTO 1520
 1510  CONTINUE
      ISF=N2+1
      GOTO 1540
 1520  DO 1530 I=1,N2
      TIME=(I-1)/SF
      IF(TIME.LT.TB) GOTO 1530
      IBM=I
      GOTO 2500
 1530  CONTINUE
 1540  IBM=N2+1
C
C
C           ************************************
C           BUILD STACK IF PROCESSING ALL BLOCKS
C           ************************************
```

```
C
2500  IF(IZ5.NE.YES) GOTO 2900
      NUMBLK=N2/ISIZE
      DO 2510 I=1,NUMBLK
      ISTACK(I)=I
2510  CONTINUE
      KPTR=NUMBLK
C
C             **********************************
C             PRINT TIME PLOT BEFORE CORRECTIONS
C             **********************************
C
2900  IF(IZ7.NE.YES) GOTO 3000
      IF(IZ5.NE.YES) GOTO 2950
C
C     ***TIME PLOT FOR ENTIRE SIGNAL***
      IF(INUM1.GT.1000) GOTO 2910
      NUM=INUM1
      KPLOTS=1
      GOTO 2920
2910  KPLOTS=INUM1/1000+1
      NUM=1000
2920  DO 2940 I=1,KPLOTS
      IF((I.EQ.KPLOTS).AND.(I.NE.1)) NUM=INUM1-(KPLOTS-1)*
     &1000
      DO 2930 J=1,NUM
      Y(J)=FLOAT(IARY((I-1)*1000+J))
      X(J)=((I-1)*1000+J-1)/SF
2930  CONTINUE
      XGRID=NUM/10
      IF(MOD(NUM,10).NE.0) XGRID=XGRID+1
      NBLK=-I
      CALL PLOTDV(X,Y,NUM,XGRID,3,NBLK)
2940  CONTINUE
      GOTO 3000
C
C     ***TIME PLOT FOR INDIVIDUAL BLOCKS***
 2950 DO 2970 I=1,KPTR
      NBLK=ISTACK(I)
      DO 2960 J=1,ISIZE
      Y(J)=FLOAT(IARY(NBLK-1)*ISIZE+J)
      X(J)=((NBLK-1)*ISIZE+J-1)/SF
 2960 CONTINUE
      XGRID=ISIZE/10
      IF(MOD(ISIZE,10).NE.0) XGRID=XGRID+1
      CALL PLOTDV(X,Y,ISIZE,XGRID,3,NBLK)
2970  CONTINUE
C
C
C
C             *******************
C             PERFORM CORRECTIONS
C             *******************
C
 3000 DO 3010 I=1,N2
      FREQ=(I-1)*SF/ISIZE
      PI=3.14159
      G=((4.*PI*SIGMA*FREQ/C)*COS(THETA))**2
      FINAL(I)=FLOAT(IARY(I))
      ITEMP=I-ISF+1
      ITEMP2=I-IBM+1
      IF(I.GE.ISF) FINAL(I)=FINAL(I)+EXP(-G/2.)*D*DS*
     &FINAL(ITEMP)
      IF(I.GE.IBM) FINAL(I)=FINAL(I)-RCOEFF*D*DB*
     &FINAL(ITEMP2)
 3010 CONTINUE
C
C            *********************************
C            PRINT TIME PLOT AFTER CORRECTIONS
C            *********************************
C
      IF(IZ4.NE.YES) GOTO 3400
```

```
      IF(IZ5.NE.YES) GOTO 3350
C
C     ***TIME PLOT FOR ENTIRE SIGNAL***
      IF(INUM2.GT.1000) GOTO 3310
      NUM=INUM2
      KPLOTS=1
      GOTO 3320
 3310 KPLOTS=INUM2/1000+1
      NUM=1000
 3320 DO 3340 I=1,KPLOTS
      IF((I.EQ.KPLOTS).AND.(I.NE.1)) NUM=INUM2-(KPLOTS-1)*
     &1000
      DO 3330 J=1,NUM
      Y(J)=FINAL((I-1)*1000+J)
      X(J)=((I-1)*1000+J-1)/SF
 3330 CONTINUE
      XGRID=NUM/10
      IF(MOD(NUM,10).NE.0) XGRID=XGRID+1
      NBLK=-I
      CALL PLOTDV(X,Y,NUM,XGRID,2,NBLK)
 3340 CONTINUE
      GOTO 3400
C
C     ***TIME PLOT FOR INDIVIDUAL BLOCKS***
 3350 DO 3370 I=1,KPTR
      NBLK=ISTACK(I)
      DO 3360 J=1,ISIZE
      Y(J)=FINAL((NBLK-1)*ISIZE+J)
      X(J)=((NBLK-1)*ISIZE+J-1)/SF
 3360 CONTINUE
      XGRID=ISIZE/10
      IF(MOD(ISIZE,10).NE.0) XGRID=XGRID+1
      CALL PLOTDV(X,Y,ISIZE,XGRID,2,NBLK)
 3370 CONTINUE
C
C
C             *********************
C             PRINT FREQUENCY PLOTS
C             *********************
C
 3400 IF(IZ3.NE.YES) GOTO 3450
C
C     ***FREQUENCY PLOT BEFORE CORRECTIONS***
      DO 3430 I=1,KPTR
      NBLK=ISTACK(I)
      DO 3410 J=1,ISIZE
      A(J)=FLOAT(IARY((NBLK-1)*ISIZE+J))
      B(J)=0.0
 3410 CONTINUE
      CALL CFFT2(A,B,ISIZE,ISIZE,ISIZE,-1)
      ITEMP=ISIZE/2
      DO 3420 J=1,ITEMP
      Y(J)=A(J)**2+B(J)**2
      Y(J)=10*ALOG10(Y(J))
      X(J)=(J-1)*SF/ISIZE
 3420 CONTINUE
      XGRID=ITEMP/10
      IF (MOD(ITEMP,10).NE.0) XGRID=XGRID+1
      CALL PLOTDV(X,Y,ITEMP,XGRID,1,NBLK)
 3430 CONTINUE
C
C     ***FREQUENCY PLOT AFTER CORRECTIONS***
 3450 IF(IZ2.NE.YES) GOTO 3500
      DO 3480 I=1,KPTR
      NBLK=ISTACK(I)
      DO 3460 J=1,ISIZE
      A(J)=FINAL((NBLK-1)*ISIZE+J)
      B(J)=0.0
 3460 CONTINUE
      CALL CFFT2(A,B,ISIZE,ISIZE,ISIZE,-1)
      ITEMP=ISIZE/2
      DO 3470 J=1,ITEMP
```

```
      Y(J)=A(J)**2+B(J)**2
      Y(J)=10.*ALOG10(Y(J))
      X(J)=(J-1)*SF/ISIZE
 3470 CONTINUE
      XGRID=ITEMP/10
      IF(MOD(ITEMP,10).NE.0) XGRID=XGRID+1
      CALL PLOTDV(X,Y,ITEMP,XGRID,4,NBLK)
 3480 CONTINUE
C
C            **********
C            CONCLUSION
C            **********
C
3500  PRINT 3510
3510  FORMAT(' ','ARE YOU FINISHED? (Y/N)--')
      READ 540,IZ6
      IF(IZ6.EQ.YES) STOP
      IREAD=1
      GOTO 672
      DEBUG SUBCHK
      END
C
C
C
C
C
C                          ****PLOT SUBROUTINE****
C
      SUBROUTINE PLOTDV(X,Y,N,XGRID,M,NB)
      INTEGER D,XGRID,YGRID,AXIS
      DIMENSION Y(1000),C7(101),O(6),X(1000),KAXIS(51)
      DATA IDASH/1H-/,ISTAR/1H*/,IDOT/1H./
      DATA IBAR/1HI/,IPLUS/1H+/,IBLANK/1H /,IX/1HX/
      AXIS=51
      YGRID=6
      XGRID=XGRID+1
2120  N1=N-1
      Y6=Y(1)
      Y1=Y(1)
      DO 2200 I=1,N
      IF(Y6-Y(I).GE.0.0) GOTO 2180
      Y6=Y(I)
2180  IF(Y1-Y(I).LE.0.0) GOTO 2200
      Y1=Y(I)
2200  CONTINUE
      S=Y1*(AXIS-1)/(Y6-Y1)
      X1=X(1)
      X10=X(N)
      O(1)=Y1
      O(6)=Y6
      IIX=XGRID-1
      DO 2410 I=1,IIX
      C7(I)=X((I-1)*10+1)
2410  CONTINUE
      C7(XGRID)=C7(XGRID-1)+10*(X(2)-X(1))
      IF(N.EQ.(XGRID-1)*10) C7(XGRID)=X(N)
      IIY=YGRID-1
      DO 2440 I=2,IIY
      O(I)=(FLOAT(I-1)*(Y6-Y1)/FLOAT(YGRID-1))+Y1
2440  CONTINUE
      WRITE(6,2460)
2460  FORMAT(///,' ')
      IF(NB.GT.0) GOTO 2466
      NB=-NB
      PRINT 2465, NB
 2465 FORMAT('0',32X,'PLOT',1X,I2)
      GOTO 2485
 2466 PRINT 2470,NB
2470  FORMAT('0',32X,'BLOCK',1X,I2)
2485  IF(M.EQ.1) PRINT 2486
2486  FORMAT(' ',17X,'DB''S VS. FREQUENCY (BEFORE CORRECTION
      IF(M.EQ.2) PRINT 2488
2488  FORMAT(' ',20X,'VOLTAGE VS. TIME (AFTER CORRECTIONS)')
```

```
      IF(M.EQ.3) PRINT 2487
2487  FORMAT(' ',20X,'VOLTAGE VS. TIME (BEFORE CORRECTIONS)')
      IF(M.EQ.4) PRINT 2489
 2489 FORMAT(' ',17X,'DB''S VS. FREQUENCY (AFTER ',
     &'CORRECTIONS)')
      WRITE(6,2500) (D(I),I=1,YGRID)
2500  FORMAT(9X,11(1PE10.2))
      S1=(X10-X1)/10.0*(XGRID-1)
      D=1
      L1=1
      L=1
      IZ=IFIX(-S+1.5)
      ITEMP=(XGRID-1)*10+1
      DO 2900 I1=1,ITEMP
      IF(N.LT.I1) GOTO 2510
      YTEMP=(Y(I1)*FLOAT(YGRID-1)*10.0/(Y6-Y1))-S
      IY=IFIX(YTEMP+1.5)
 2510 IF(L1.GT.L) GOTO 2760
      DO 2650 IP=1,AXIS
      KAXIS(IP)=IDASH
2650  CONTINUE
      DO 2680 I=1,AXIS,10
      KAXIS(I)=IPLUS
2680  CONTINUE
      IF(N.LT.I1) GOTO 2720
      IF((Y1.LE.0.0).AND.(0.0.LE.Y6)) KAXIS(IZ)=IDOT
      KAXIS(IY)=ISTAR
2720  WRITE(6,2725) C7(D),(KAXIS(J),J=1,AXIS)
2725  FORMAT(1PE13.2,2X,115A1)
      L1=L1+10
      D=D+1
      GOTO 2870
2760  DO 2780 IP=1,AXIS
      KAXIS(IP)=IBLANK
2780  CONTINUE
      DO 2810 I=1,AXIS,10
      KAXIS(I)=IBAR
2810  CONTINUE
      IF(N.LT.I1) GOTO 2860
      IF((Y1.LE.0.0).AND.(0.0.LE.Y6)) KAXIS(IZ)=IDOT
      KAXIS(IY)=ISTAR
2860  WRITE(6,2865) (KAXIS(J),J=1,AXIS)
2865  FORMAT(15X,115A1)
2870  L=L+1
      IF(L.GT.(XGRID-1)*10+2) GOTO 2910
2900  CONTINUE
2910  RETURN
      DEBUG SUBCHK
      END
C
C
C                    ***FOURIER TRANSFORM SUBROUTINE***
C
      SUBROUTINE CFFT2(A,B,NTOT,N,NSPAN,ISN)
```

LIST OF REFERENCES

1. Naval Undersea Warfare Center Technical Note 150, Requirements of a Method for Locating Underwater Bio-Acoustic Sources, pp. 3-8, by W. C. Cummings, July 1968.

2. Woods Hole Oceanographic Institution Report 71-60, Four Hydrophone Array for Acoustic Three-Dimensional Location, by W. A. Watkins and W. E. Schevill, pp. 5-8, October 1971.

3. Thomas, J. B., Statistical Communication Theory, pp. 85-92, John Wiley and Sons, Inc., New York, 1969.

4. Clay, C. S. and Medwin, H., Acoustical Oceanography, pp. 78-96, John Wiley and Sons, Inc., New York, 1977.

5. Clay, C. S., Medwin, H., and Wright, W. M., "Specularly Scattered Sound and the Probability Density Function of a Rough Surface," The Journal of the Acoustical Society of America, v. 53, n. 6, pp. 1677-1682, 1973.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Director of Defense Research and Engineering Office of the Secretary of Defense Washington, D.C. 20301 ATTN: Office, Assistant Director (Research)	1
2.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
3.	Office of Naval Research Arlington, Virginia 22217	
	ATTN: (Code 480)	3
	ATTN: (Code 460)	1
	ATTN: (Code 102-OS)	1
	ATTN· (Code 102IP)	6
4.	Commander Naval Oceanographic Office Washington, D.C. 20390	
	ATTN: (Code 1640)	1
	ATTN: (Code 70)	1
	NODC/NOAA Rockville, Maryland 20882	1
5.	Director Naval Research Laboratory Washington, D.C. 20375 ATTN: Library, Code 2620	6
6.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
7.	Department Chairman, Code 61 Department of Physics and Chemistry Naval Postgraduate School Monterey, California 93940	2
8.	Professor H. Medwin, Code 61Md Department of Physics and Chemistry Naval Postgraduate School Monterey, California 93940	7

		No. Copies
9.	Professor O. B. Wilson, Code 61W1 Department of Physics and Chemistry Naval Postgraduate School Monterey, California 93940	1
10.	LCDR Richard M. Bostian, USN 455 San Bernabe Drive Monterey, California 93940	1