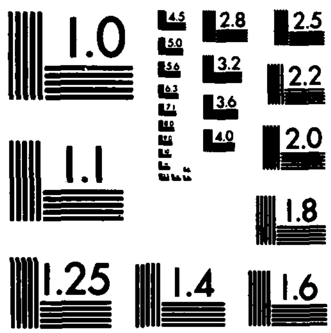
AD-A133 876 ELF/VLF (EXTREMELY LLW FREQUENCY/VERY LOW FREQUENCY) 1/2
LONG PATH PULSE PROG. (U) NAVAL OCEAN SYSTEMS CENTER
SAN DIEGO CA R A PAPPERT ET AL. AUG 83 NOSC/TR-891

UNCLASSIFIED F/G 20/14 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NOSC TR 891

Technical Report 891

[illegible] PULSE PROGRAM [illegible] OF ARBITRARY [illegible] ORIENTATION

R. A. Pappert
L. R. Hitney
J. A. Ferguson

[illegible] 1983
[illegible] report for period
[illegible] — August 1983

Prepared for
Defense Nuclear Agency

[illegible] distribution unlimited.

NOSC

NAVAL OCEAN SYSTEMS CENTER
San Diego, California 92152

DTIC

83 10 20 046

NAVAL OCEAN SYSTEMS CENTER SAN DIEGO, CA 92152

AN ACTIVITY OF THE NAVAL MATERIAL COMMAND

J.M. PATTON, CAPT, USN
Commander

R.M. HILLYER
Technical Director

ADMINISTRATIVE INFORMATION

The work reported here was performed by members of the Ocean & Atmospheric Sciences Division under program element RDDA, project MP20, during the period January 1983–August 1983. Washington sponsorship was provided by the Defense Nuclear Agency.

Released by
J.H. Richter, Head
Ocean & Atmospheric
Sciences Division

Under authority of
J.D. Hightower, Head
Environmental Sciences
Department

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NOSC Technical Report 891 (TR 891)	2. GOVT ACCESSION NO. AD-A133876	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) ELF/VLF LONG PATH PULSE PROGRAM FOR ANTENNAS OF ARBITRARY ELEVATION AND ORIENTATION		5. TYPE OF REPORT & PERIOD COVERED Interim January - August 1983 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) R.A. Pappert L.R. Hitney J.A. Ferguson		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Ocean Systems Center San Diego, CA 92152		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS RDDA, MP20
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Nuclear Agency Washington, DC 20305		12. REPORT DATE August 1983 13. NUMBER OF PAGES 109
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)

ELF (extremely low frequency)
VLF (very low frequency)
Antennas
Earth-ionosphere waveguide
Pulse propagation
Computer program

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

A computer program designed to handle pulse propagation problems when the propagation channel is the earth-ionosphere waveguide, and intended for use in the elf/vlf bands, is presented. The program is intended for use with laterally homogeneous channels. Allowance is made for transmit and receive antennas of arbitrary elevation and orientation. Mode data as a function of frequency from a waveguide program are required inputs to the present program. The mode data are interpolated by using cubic splines, and the requisite integrals are treated numerically by means of the fast Fourier transform.

CONTENTS

I. INTRODUCTION

This report describes and lists a computer program designed to handle pulse propagation problems when the propagating channel is the earth-ionosphere waveguide and is intended for use in the elf/vlf bands. The report is an extension of earlier work (ref 1), which was restricted to pulse propagation of the vertical electric field generated by a ground-based vertical eletric dipole source. The present extension makes allowance for calculating, at any height within the guide, all electric field components generated by electric dipole sources of arbitrary orientation and elevation. Inputs are mode data (i.e., eigenangles and excitation factors) as a function of frequency as determined, for example, by the waveguide program of reference 2. The mode data are interpolated by using cubic splines. The pulse shape integral (which is a Fourier transform) is calculated by employing, at the user's option, either a fast Fourier transform technique or a Filon technique. Normally, the fast Fourier transform (FFT) technique is used. Advantages and disadvantages of the FFT have been discussed by Seyler, Block, and Flynn (ref 3). Its major advantage is a savings in computational time, whereas a disadvantage may be that, strictly, only periodic pulse trains may be analyzed. Thus, when a nonperiodic pulse is considered, it must be treated as a periodic pulse train with period much greater than the pulse width in order to obtain adequate resolution. Another disadvantage of the FFT is that there is no measure of the accuracy of the integral evaluation. For this reason, a second integration routine based on the Filon method (ref 4) is included. The method is more direct but much slower than the FFT. In addition to the integral evaluation, output of the Filon integration contains an indication of the accuracy of the evaluation, and this is perhaps most useful for purposes of checking the FFT.

At present, the program is designed to handle only laterally homogeneous waveguides. It is likely that the subroutine "CHANEL" could be extended to allow for lateral inhomogeneity of the guide via WKB or mode conversion methods. Whereas the program of reference 1 was developed primarily as a tool for calculating slow-tail atmospheric wave forms (i.e., wave shapes in the elf band generated by lightning discharges), the present program is intended more as an aid to elf/vlf system designers. Thus, in addition to the slow-tail waveform capability, the program allows for the study of the distortion of square-wave and Gaussian pulse envelopes (as well as sequences of such pulses) and for the analysis of spread spectrum systems (ref 5, 6). However, it is stressed that alternative input waveforms can be accommodated by straightforward modification of the subroutine XMTR.

The mathematical problem at hand simply reduces to the calculation of a Fourier integral for which the integrand is made up of a transmitter spectrum, receiver spectrum, and channel spectrum, each of which is discussed in the following section. In section III, the program input is described and in section IV the program structure is outlined. Section V contains output description and sample results. The appendix contains a program listing.

II. SOURCE, RECEIVER, AND CHANNEL MODELS

OUTPUT WAVEFORM

In the following, x, y, z is a Cartesian coordinate system with x - z being the plane of propagation and z directed into the ionosphere with the ground at z = 0. In terms of the source, receiver, and channel spectrums, the output waveform, $G(x, z_R, t; z_T)$, at a great circle range x, altitude z_R, and time t generated by a source at x = y = 0, $z = z_T$ may be written as ($i = \sqrt{-1}$)

$$G(x, z_R, t; z_T) = \frac{1}{2\pi} \int_{-\infty}^{\infty} s(\omega)r(\omega)h(\omega, x, z_R; z_T)e^{i\omega t}d\omega$$

$$= \frac{1}{\pi} \text{Re} \int_{0}^{\infty} s(\omega)r(\omega)h(\omega, x, z_R; z_T)e^{i\omega t}d\omega$$

$$= 2 \text{ Re} \int_{0}^{\infty} S(f)R(f)H(f, x, z_R; z_T)e^{2\pi ift}df \qquad (1)$$

where

$$S(f) = s(\omega) = s(2\pi f) \text{ source (current moment) spectrum} \qquad (2)$$

$$R(f) = r(\omega) = r(2\pi f) \text{ receiver spectrum} \qquad (3)$$

$$H(f, x, z_R; z_T) = h(\omega, x, z_R; z_T)$$
$$= h(2\pi f, x, z_R; z_T) \text{ channel spectrum.} \qquad (4)$$

The second and third equalities in equation 1 follow from the requirement that G be a real quantity, so that

$$S(f) = S^*(-f), \; R(f) = R^*(-f), \; H(f, x, z_R; z_T)$$
$$= H^*(-f, x, z_R; z_T) \qquad (5)$$

where the asterisk denotes the complex conjugate. G can represent any of the electric field components, E_x, E_y, E_z, generated, as mentioned above, by an arbitrarily oriented electric dipole at x = y = 0, $z = z_T$. The receiver, source, and channel functions are described below.

RECEIVER

RECVR is a subroutine that can be easily modified or replaced to accommodate the individual user's needs. The particular RECVR subroutine contained in the program listing in the appendix assumes a receiver function of the form

$$R(f) = \left(\frac{if/f_1}{1 + if/f_1}\right)^P \left[\left(1 + i(f - f_2)/f_3\right)^{-Q} + \left(1 + i(f + f_2)/f_3\right)^{-Q}\right]; \; f_2 \neq 0$$

$$R(f) = \left(\frac{if/f_1}{1 + if/f_1}\right)^P \quad (1 + if/f_3)^{-Q} \; ; \; f_2 = 0 \qquad (6)$$

where the frequencies f_1, f_2, f_3 and the exponents P and Q are read into the program via namelist. This receiver function allows for modeling receivers representative of spread spectrum systems (ref 5) as well as those used in slow wave tail studies of atmospherics. Observe that the function satisfies the condition specified in equation 5.

TRANSMITTER

TRXMTR is a subroutine that, too, can be readily altered to meet specific needs of the user. In the present program, four source functions are available. They are called by setting IFLGTR = 1, 2, 3, or 4. The source functions are:

i)IFLGTR = 1

$$s(\omega) = u(\omega)v(\omega) \qquad (7)$$

where

$$u(\omega) = -\frac{e^{i\omega T/2}}{2}\left[\frac{1}{(\omega_o - \omega)}\left(e^{i(\omega_o-\omega)T} - 1\right) + \frac{1}{(\omega_o + \omega)}\left(e^{-i(\omega_o+\omega)T} - 1\right)\right] \qquad (8)$$

$$v(\omega) = \sum_{n=o}^{N} e^{-i\omega n(T+\delta t)}$$

$$= \frac{\exp\left(-i\omega(N + 1)(T + \delta t)/2\right)}{\exp\left(-i\omega(T + \delta t)/2\right)} \; \frac{\sin\left(\omega(N + 1)(T + \delta t)/2\right)}{\sin\left(\omega(T + \delta t)/2\right)} . \qquad (9)$$

The spectrum $s(\omega)$ corresponds to the time function

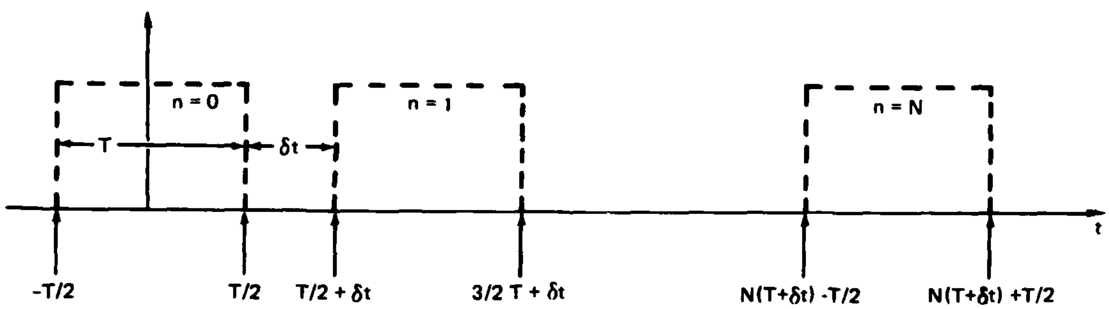
$$g(t) = \sum_{n=o}^{N} q_n(t) \qquad (10)$$

where

$$q_n(t) = \sin\left[\omega_o\left(t - n(T + \delta t) + T/2\right)\right] \; : \; n(T + \delta t) - T/2 \le t \le n(T + \delta t) + T/2$$

$$= 0 \; : \; \text{otherwise.}$$

Equation 10 represents a sine wave modulated by a series of N + 1 square wave envelopes as indicated below.



If equation 1 is used to define

$$a + ib = \int_o^\infty S(f)R(f)H(f,\, x,\, z_R;\, z_T)e^{2\pi ift}dt \; , \qquad (11)$$

the envelope $2\sqrt{a^2 + b^2}$ normalized to unity is the plotted output for IFLGTR = 1. Printed output in units of dB above a μV/m per kW is also available. This normalization assumes that equation 10 is multiplied by a current moment corresponding to a CW power output of 1 kW at f_o when placed vertically over a perfectly conducting half plane.

ii) IFLGTR = 2

$$s(\omega) = w_1(\omega)v_1(\omega) + w_2(\omega)v_2(\omega) \qquad (12)$$

where

$$w_1(\omega) = \frac{\sqrt{\pi}T}{2i} \exp\left(-(\omega_o - \omega)^2 T^2/4\right) \qquad (13)$$

$$w_2(\omega) = \frac{i\sqrt{\pi}T}{2} \exp\left(-(\omega_o + \omega)^2 T^2/4\right) \qquad (14)$$

$$v_1(\omega) = \sum_{n=o}^{N} e^{i(\omega_o - \omega)n\delta t}$$

$$= \frac{\exp\left(i(\omega_o - \omega)(N + 1)\delta t/2\right)}{\exp\left(i(\omega_o - \omega)\delta t/2\right)} \; \frac{\sin\left((\omega_o - \omega)(N + 1)\delta t/2\right)}{\sin\left((\omega_o - \omega)\delta t/2\right)} \qquad (15)$$

$$v_2(\omega) = \sum_{n=o}^{N} e^{-i(\omega_o + \omega)n\delta t}$$

$$= \frac{\exp\left(-i(\omega_o + \omega)(N + 1)\delta t/2\right)}{\exp\left(-i(\omega_o + \omega)\delta t/2\right)} \; \frac{\sin\left((\omega_o + \omega)(N + 1)\delta t/2\right)}{\sin\left((\omega_o + \omega)\delta t/2\right)} \; . \qquad (16)$$

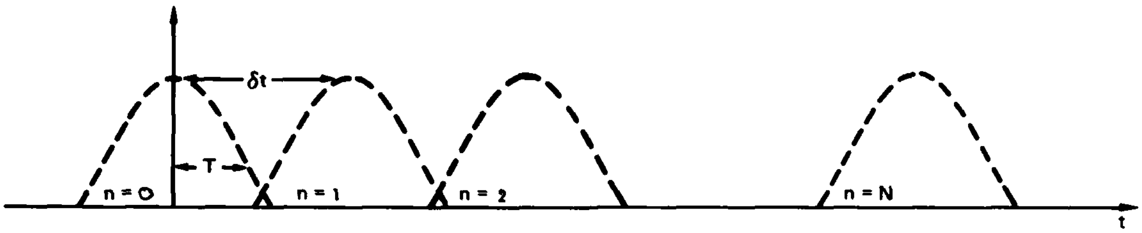
The spectrum in this instance corresponds to the time functions

$$g_1(t) = \sum_{n=o}^{N} q_{1n}(t) \qquad (17)$$

where

$$q_{1n}(t) = \exp\left(-(t - n\delta t)^2/T^2\right)\sin(\omega_o t) \; . \qquad (18)$$

Equation 18 represents a sine wave (carrier frequency f_o) modulated by a series of N + 1 Gaussian envelopes as indicated below.



In terms of the definitions for a and b given in equation 11, the envelope $2\sqrt{a^2 + b^2}$, normalized to unity, is the plotted output for IFLGTR = 2. Printed output in units of dB above a μV/m per kW is also available. This normalization assumes that equation 17 is multiplied by a current moment corresponding to a cw power output of 1 kW at f_o when placed vertically over a perfectly conducting half plane.

iii) IFLGTR = 3

Rothmuller (ref 5) and Kelly et al. (ref 6) have investigated the effect that the earth-ionosphere waveguide has on one type of vlf communication system. The system studied was characterized by a differential phase-encoded signal waveform composed by frequency shift keying (FSK) a carrier with a binary pseudorandom or pseudonoise (PN) sequence of pulses or chips. The FSK modulation index is 0.5, which is designated as minimum shift keying (MSK). For more detail concerning the basic waveform and terminology, the interested reader should see references 5 and 6. Here we note only that the PN sequence has a power spectrum (or source spectrum for our purposes) given by

$$S(f) = K \frac{8}{\pi^2 f_c} \left[\frac{\cos^2\left((f - f_o)2\pi/f_c\right)}{\left(1 - 16(f - f_o)^2/f_c^2\right)^2} + \frac{\cos^2\left((f + f_o)2\pi/f_c\right)}{\left(1 - 16(f + f_o)^2/f_c^2\right)^2} \right] \qquad (19)$$

where

f_o = carrier frequency

f_c = chip frequency

and K is a constant determined rather arbitrarily from the relation

$$\int_{-\infty}^{\infty} S(f)df = K \tag{20}$$

K is chosen so that, when used in conjunction with the channel function given subsequently, it would correspond to a vertical electric dipole current moment at frequency f_o, which would radiate 1 kW of power when placed over a perfectly conducting plane. This normalization gives

$$K = \frac{2.386 \times 10^8}{f_o} \quad A/m . \tag{21}$$

with f_o in Hz.

Output of the correlation receiver corresponding to the delay time τ is

$$2\text{Re}\left\{\int_0^{\infty} S(f)R(f)H(f, x, z_R;z_T)e^{2\pi ifz}\right\} = 2\ \text{Re}(a + ib)$$

$$= 2\sqrt{(a')^2 + (b')^2}\ \cos(2\pi f_o \tau + \phi) \tag{22}$$

where

$$\left.\begin{aligned} a' &= a \cos 2\pi f_o\tau + b \sin 2\pi f_o\tau \\ b' &= -a \sin 2\pi f_o\tau + b \cos 2\pi f_o\tau \\ \tan\phi &= b'/a' \end{aligned}\right\} \tag{23}$$

The envelope $2\sqrt{(a')^2 + (b')^2}$ expressed in dB above 1 μV/m/kW radiated (interpreted in the sense of equations 20 and 21) and the phase, ϕ, of the correlation vector as a function of the delay τ is the output corresponding to IFLGTR = 3.

iv) IFLGTR = 4

The principal motivation for the earlier work (ref 1) was to study the shape of slow wave tails associated with atmospheric discharges. This capability is retained in the present report. For this purpose, the particular source function contained in the subroutine TRXMTR is Williams' (ref 7) mean source description for a lightning discharge, which is given by

$$Idl(\omega) = v_o \sum_{n=1}^{4} \frac{A_n}{(\gamma_n + j\omega)^2}\left(1 - \frac{\exp\left[-\tau_\rho(\gamma_n + j\omega)\right]}{1 + \tau_v(\gamma_n + j\omega)}\right) \tag{24}$$

where

$$\left.\begin{array}{ll} A_1 = 16.8 \times 10^3\ A & \gamma_1 = 5.88 \times 10^5\ s^{-1} \\ A_2 = 15.35 \times 10^3\ A & \gamma_2 = 3.03 \times 10^4\ s^{-1} \\ A_3 = 10^3\ A & \gamma_3 = 2.0 \times 10^3\ s^{-1} \\ A_4 = 0.45 \times 10^3\ A & \gamma_4 = 1.47 \times 10^2\ s^{-1} \\ \tau_\rho = 43\ \mu s & \tau_v = 180\ \mu s \\ v_o = 3.5 \times 10^7\ m/s & \end{array}\right\} \tag{25}$$

The A_i's, γ_i's, τ_p, τ_v, and v_o are contained in DATA statements in TRXMTR. The units of amperes for the A_i expressions and m/s for v_o, coupled with the channel defined in the following subsection, yield a plotted wave form in units of μV/m. The printed output is in units of dB/μV/m.

CHANNEL-EXCITATION FACTORS AND HEIGHT GAINS

Summarized in this subsection are modal excitation and height gain formulas required as input for the mode sum evaluations, which allow for arbitrary elevation (within the guide) and orientation of the transmitter and receiver. The formulas have been given earlier (ref 8) and are included here for completeness. The excitation factor formulas are given in the 3 × 3 matrix below. The column headings apply to excitation of the electric field components E_z, E_y and E_x and the row headings apply to excitation by a vertical electric dipole (λ_v), horizontal broadside electric dipole (λ_B), and a horizontal end-on electric dipole (λ_E). The direction of z is taken positive into the ionosphere with x - z being the plane of propagation and y normal to the plane of propagation.

FIELD COMPONENT →	E_z	E_y	E_x	
EXCITER ↓				
λ_v	S^2T_1	$-ST_3$	ST_1	
λ_B	$-ST_3T_4$	T_2	$-T_3T_4$	(26)
λ_E	$-ST_1$	T_3	$-T_1$	

where

$$T_1 = \frac{(1 + {}_{\parallel}\bar{R}_{\parallel})^2 (1 - {}_{\perp}R_{\perp}\ {}_{\perp}\bar{R}_{\perp}) S^{1/2}}{\frac{\partial F}{\partial \theta}\ {}_{\parallel}\bar{R}_{\parallel}\ D_{11}} \tag{27}$$

$$T_2 = \frac{(1 + {}_{\perp}\bar{R}_{\perp})^2 (1 - {}_{\parallel}R_{\parallel}\ {}_{\parallel}\bar{R}_{\parallel}) S^{1/2}}{\frac{\partial F}{\partial \theta}\ {}_{\perp}\bar{R}_{\perp}\ D_{22}} \tag{28}$$

$$T_3 = \frac{(1 + {}_{\parallel}\bar{R}_{\parallel})(1 + {}_{\perp}\bar{R}_{\perp})\, {}_{\parallel}R_{\perp} S^{1/2}}{\frac{\partial F}{\partial \theta}\ D_{12}} \tag{29}$$

$$T_4 = \frac{{}_{\perp}R_{\parallel}}{{}_{\parallel}R_{\perp}} \tag{30}$$

The R and $\bar{R}$'s represent, repectively, elements of the reflection matrix looking into the ionosphere and toward the ground from the same level d within the guide. The first subscript refers to the polarization of the incident wave, while the second applies to the polarization of the reflected wave. S is the sine of an eigenangle, θ, and ∂F/∂θ is the derivative of the modal function which is evaluated at an eigenangle. The T and θ expressions are input from the waveguide program of reference 2.

The excitation factors must be supplemented with definitions of height gains. These, along with the definitions of the D_{ij} expressions, are

$$f_{\parallel}(z) = \exp\left(\frac{z - d}{a}\right)\left(F_1 h_1(q) + F_2 h_2(q)\right) \tag{31}$$

$$f_{\perp}(z) = F_3 h_1(q) + F_4 h_2(q) \tag{32}$$

$$g(z) = \frac{1}{ik}\frac{d}{dz} \; f_{\parallel}(z) \tag{33}$$

$$D_{\parallel} = f_{\parallel}^{\,2}(d) \qquad D_{12} = f_{\parallel}(d) f_{\perp}(d) \qquad D_{22} = f_{\perp}^{\,2}(d) \tag{34}$$

$$F_1 = -\left\{H_2(q_o) - i\,\frac{n_o^{\,2}}{N_g^{\,2}}\left(\frac{ak}{2}\right)^{1/3}\left(N_g^{\,2} - S^2\right)^{1/2} h_2(q_o)\right\} \tag{35}$$

$$F_2 = H_1(q_o) - i\,\frac{n_o^{\,2}}{N_g^{\,2}}\left(\frac{ak}{2}\right)^{1/3}\left(N_g^{\,2} - S^2\right)^{1/2} h_1(q_o) \tag{36}$$

$$F_3 = -\left\{h_2'(q_o) - i\left(\frac{ak}{2}\right)^{1/3}\left(N_g^{\,2} - S^2\right)^{1/2} h_2(q_o)\right\} \tag{37}$$

$$F_4 = h_1'(q_o) - i\left(\frac{ak}{2}\right)^{1/3}\left(N_g^{\,2} - S^2\right)^{1/2} h_2(q_o) \tag{38}$$

$$q = \left(\frac{2}{ak}\right)^{-2/3}\left(C^2 - \frac{2}{a}\,(h - z)\right) \tag{39}$$

$$H_j(q) = h_j'(q) + \frac{1}{2}\left(\frac{2}{ak}\right)^{2/3} h_j(q) \quad ; \quad j = 1,2 \tag{40}$$

$$n^2 = 1 - \frac{2}{a}(h - z) \tag{41}$$

$$N_g^{\;2} = \frac{\varepsilon}{\varepsilon_o} - i\,\frac{\sigma}{\omega\varepsilon_o} \tag{42}$$

where

C	=	cosine of the angle of incidence at height h
k	=	free space wave number
$\varepsilon/\varepsilon_o$	=	dielectric constant of the ground
σ	=	ground conductivity
ω	=	circular radio frequency
a	=	earth's radius.

The functions h_1 and h_2 are modified Hankel functions of order 1/3 (which are linearily related to Airy functions), as defined by the computation Laboratory at Cambridge, Massachusetts (ref 9). The primes on these quantities denote derivatives with respect to the argument. Equation 41 is the modified refractive index which equals unity at height, h. The subscript, o, which appears on n^2 in equations 35 and 36, signifies that equation 41 is to be evaluated for z = 0. Similarly, the subscript o that appears on q in equations 35 through 38 signify that equation 39 is to be evaluated for z = 0. It should be pointed out that $f_{\parallel}$ is the height gain for the vertical electric field E_z, $f_{\perp}$ the height gain for the horizontal electric field component (E_y) normal to the plane of propagation, and g the height gain for the horizontal electric field component (E_x), which is in the plane of propagation.

Because the imaginary part of the eigenangle in absolute value can become quite large when operating in the ELF range, it proves necessary to avoid overflow, and use of the flat earth analogues of equations 31 through 33 is justified. That is, to replace the height gains by

$$f_{\parallel}(z) = \exp(ikCz) + {}_{\parallel}\bar{R}_{\parallel}\, \exp(-ikCz + 2ikCd) \tag{43}$$

$$f_{\perp}(z) = \exp(ikCz) + {}_{\perp}\bar{R}_{\perp} \exp(-ikCz + 2ikCd) \tag{44}$$

$$g = C \left[\exp(ikCz) - {}_{\parallel}\bar{R}_{\parallel} \exp(-ikCz + 2ikCd)\right]. \tag{45}$$

When the absolute value of the imaginary part of the eigenangle exceeds 10°, the height gain functions will be computed by equations 43, 44, and 45.

The punched output of the waveguide program of reference 2 is transformed to correspond to d = 0, independent of the actual d used in the waveguide run. Therefore, in the present program d in the above formulas is set to zero.

CHANNEL-MODE SUM

In terms of the excitation factors and height gains defined in the previous section, the mode sum for the laterally homogeneous guide may be written as follows

$$E_j(x) = \frac{QM}{\left[\sin(x/2)\right]^{1/2}} \sum_n \left\{\lambda_v^n \cos(\gamma) f_{\parallel}^n(z_T) + \lambda_B^n \sin(\gamma)\sin(\phi) f_{\perp}^n(z_T) + \lambda_E^n \sin(\gamma)\cos(\phi) g^n(z_T)\right\} f_j^n(z_R) e^{-ik(S_n - 1)x} \tag{46}$$

The transmitter coordinates are $(0, 0, z_T)$ and the receiver coordinates are $(x, 0, z_R)$. The mode index is n and the index j takes on three values corresponding to the electric field component measured at the receiver.

$j = 1 \rightarrow z$ component $\rightarrow f_1 = f_{\parallel}$

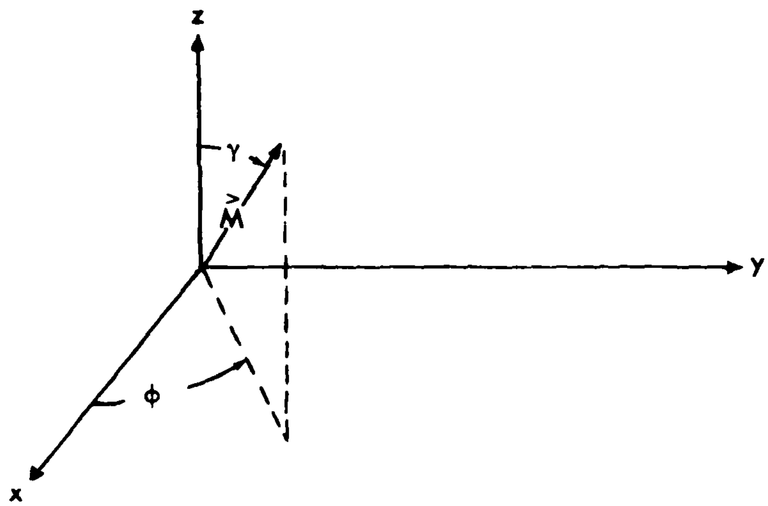
$j = 2 \rightarrow y$ component $\rightarrow f_2 = f_{\perp}$

$j = 3 \rightarrow x$ component $\rightarrow f_3 = g$.

M is the dipole moment in A/m and Q is

$$Q = 9.023 \times 10^{-8}(if)^{3/2} \tag{47}$$

with f in Hz. With this value for Q, the field is in units of μV/m. If M is chosen to correspond to 1 kW radiated power when the dipole source is placed vertically over a perfectly conducting plane, then E_j is expressed as μV/m/kW radiated. The angles γ and ϕ measure the orientation of the transmitter relative to the x, y, z coordinate system as shown below.



III. DESCRIPTION OF INPUT

All input to the pulse shape program is read in via the card reader. A listing of a sample input showing the data deck setup is given on pages 20 through 23.

There are two parts to the input. The first part is read in by means of an ASCII FORTRAN name list input format. The name list read in is initiated with a control card NAME in columns 1 through 4. The following variables and arrays may be specified in the name list input.

NFFT - 2**NFFT is the number of integration intervals in the range (FREQU-FREQL) when using the FFT for the Fourier evaluations. Default value is 11, which is the maximum allowed without changing dimensioning.

FREQU - upper frequency of integration in kilohertz. Default value is 100 kHz.

FREQL - lower frequency of integration in kilohertz. Default value is 0.0 kHz.

INTPRT - flag to control print interval for transmitter, receiver, channel, and product spectra as a function of frequency. For example, the first NPRNT (see below) spectra are printed and thereafter only those spectra for which the frequency index modulo INTPRT equals zero will be printed. Default value is 100.

NPRNT - flag to control print interval for transmitter, receiver, channel and product spectra as a function of frequency. The first NPRNT spectra are printed. Default value is 40.

TAUMAX - controls the latest time in seconds plotted on the output waveform curve. Also controls the maximum time for which the

Fourier integral is calculated when using the Filon routine. Default value is 0.002 s.

TAUO - controls the earliest time in seconds plotted on the output waveform curve. Also controls the minimum time for which the Fourier integral is calculated when using the Filon routine. Default value is -0.001 s.

NUMTAU - number of taus between and inclusive of TAUO and TAUMAX for which Filon evaluations are to be made. Default value is 41.

FREQO - carrier frequency in kHz. Default value is 23 kHz.

PULSEW - square-wave pulse width or Gaussian e^{-1} half width in μs. Default value is 600 μs.

FREQ1, FREQ2, FREQ3 - frequencies in kHz appearing in the receiver function given by equation 6. Default values are FREQ1 = 0.01 kHz, FREQ2 = 0.0 kHz, FREQ3 = 2.5 kHz.

P, Q - exponents appearing in the receiver function given by equation 6.Default values are P = 0.0, Q = 2.0.

RHOMIN - minimum range in km for which mode sum and pulse shape or correlation vector is to be evaluated. Default value is 1000 km.

RHOMAX - maximum range in km for which mode sum and pulse shape or correlation vector is to be evaluated. Default value is 1000 km.

DELRHO - incremental ranges in km between RHOMIN and RHOMAX for which mode sum and pulse shape or correlation vector is to be evaluated. Default value is 1000 km.

TALT - transmitter altitude in km. Default value is 0, which corresponds to ground-based transmitter.

RALT - receiver altitude in km. Default value is 0, which corresponds to ground-based receiver.

INCL - inclination of transmitter from positive z axis in degrees (angle γ in equation 44). Default value is 0°, which corresponds to a vertical antenna.

THETA - azimuth of transmitter measured counterclockwise from x axis in degrees. Default value is 0°, corresponding to an end on launch.

ICOMP - singles out electric field component calculated. ICOMP = 1 gives the vertical or E_z field. ICOMP = 2 gives the E_y field and ICOMP = 3 gives the E_x component. Default value is 1.

IFLGTR - selects input wave form. IFLGTR = 1 gives sequence of square wave pulses. IFLGTR = 2 gives sequence of Gaussian pulses. IFLGTR = 3 gives waveform composed by frequency shift keying (FSK) a carrier with a binary pseudorandom or pseudonoise (PN) sequence of pulses or chips. The FSK modulation index is 0.5. IFLGTR = 4 gives Williams' source for the slow wave tail calculation. Default value is 1.

INTFLG - selects integration scheme. INTFLG = 0 invokes FFT algorithm and is the normal operating mode. INTFLG ≠ 0 invokes the Filon method. Default value is 0.

CHIPFR - chip frequency (f_c in eq 19) in kHz to be used with IFLGTR = 3. Default value is 1. kHz.

NUMPLS - number of pulses in the square wave or Gaussian sequence. Default value is 1.

PULSED - time delay in μs between square wave or Gaussian pulses. Default value is 600. μs.

IPLOT - flag controlling plots of transmitter and receiver spectra. IPLOT = 0 gives no plots. IPLOT ≠ 0 gives both spectrum plots. Default value is 0.

IPLOT1 - flag controlling plots of channel and product spectra. IPLOT1 = 0 gives no plots. IPLOT1 ≠ 0 gives both spectrum plots. Default value is 0.

Following the namelist input, the second part of the input, consisting of run identification and mode constant cards, is read in. The read in is initiated by a control card with DATA in columns 1 through 4. Run identification is read in by using a 20A4 format. Mode data appearing on pages 20 through 23 consist of:

R - punched output from waveguide program, which is not used in present program.

F - frequency in kHz.

A - azimuth in degrees relative to geomagnetic north, for which waveguide program was run.

C - codip in degrees of geomagnetic field, for which waveguide program was run.

M - strength of geomagnetic field in Weber/m^2, for which waveguide program was run.

S - ground conductivity in Siemens/m.

E - dielectric constant of the ground.

T - punched output from waveguide program, which is not used in present program.

The card containing the above information is followed by two cards for each mode at frequency F. The first card contains the index 1 and the following mode input data:

TR1 - real part of the eigenangle in degrees.

TI1 - imaginary part of the eigenangle in degrees.

TMP1 - excitation factor coefficient T_1 given in equation 27.

TMP2 - excitation factor coefficient T_2 given in equation 28.

The second card contains the index 2 and the following mode input data:

TR1 - real part of the eigenangle in degrees (repeat of input of information on the first card).

TI1 - imaginary part of the eigenangle in degrees (repeat input of information on the first card).

TMP3 - excitation factor coefficient T_3 given in equation 29.

TMP4 - excitation factor coefficient T_4 given in equation 30.

After all data are read in for a given frequency, mode data for the next frequency are read in, etc. After data for all of the frequencies are read in, transfer of the program to its execution phase is initiated by a control card with STAR in columns 1 through 4. Upon completion of the program for the first complete set of data, a new complete set of data can be read in, processed, and executed, etc.

SAMPLE INPUT

```
NAME
 &DATUM
 INTPRT = 20
 TAUMAX = 3.E-3
 IFLGTR = 1,
 NPRNT = 20
 TAU0 = -1.E-3,
 NUMPLS = 2
 FREQ2=23.0,FREQ3=1.667,P=0.0,Q=3.0,
 IPLOT=1,
 IPLOT1=1,
 &END
DATA
BETA=0.5, HPRIME=87.0
R   .000 F 10.0000 A  96.848 C  26.610 M 4.680-005 S 4.640+000 E 81.0 T
1 89.54299 -1.600041 5.93769233-004-4.33237092-002-2.66095085-013-6.6910
2 89.54299 -1.600041 1.71712808-008 1.76637679-007 9.76119950-001-1.6857
1 83.15179  -.681192 1.40675269-003-4.30542375-004-1.30534744-010-1.6749
2 83.15179  -.681192 4.75598456-010-4.42039628-007 9.76321563-001-1.6914
1 76.02858  -.331871-1.81176863-003-5.79840308-002-4.57239026-012-1.6623
2 76.02858  -.331871 6.34841966-008 1.00232208-006 9.76755522-001-1.7083
1 71.53966  -.972492 3.57218043-003-9.53919356-004-6.43127843-010-5.4550
2 71.53966  -.972492-1.48832975-009-1.55049825-006 9.77606788-001-1.7225
1 64.67405  -.420081-5.74631611-003-5.44617977-002-5.96155712-011-1.2776
2 64.67405  -.420081 5.05197228-007 2.73988729-006 9.79136109-001-1.7517
1 60.43294 -1.736032 8.06905213-003-3.46847059-003-1.51790046-009-6.2679
2 60.43294 -1.736032-3.41322277-007-3.64204141-006 9.81437340-001-1.7637

R   .000 F 15.0000 A  96.848 C  26.610 M 4.680-005 S 4.640+000 E 81.0 T
1 89.87716 -4.447931 7.97650921-006-1.86808831-002-1.36770776-012-1.1737
2 89.87716 -4.447931 6.98923905-008 1.73850939-007 9.54479575-001-9.2540
1 89.42791 -2.627812 1.81225575-003-1.87533820-003-3.94927788-011-2.0386
2 89.42791 -2.627812-1.05341859-007-3.10618208-007 9.54131447-001-9.2843
1 82.12311  -.339621-1.10011261-003-4.10230709-002-3.93972420-012-5.8973
2 82.12311  -.339621 1.41023918-007 5.33180142-007 9.52227563-001-9.4214
1 78.92260  -.345582 2.13041180-003-1.65751675-003-2.50544481-010-4.5708
2 78.92260  -.345582-2.25597727-007-8.10605862-007 9.50611316-001-9.5459
1 74.36239  -.423671-2.86681467-003-3.58344899-002-3.07619798-011-4.3661
2 74.36239  -.423671 4.13103759-007 1.35826212-006 9.47571643-001-9.8035
1 71.51556  -.501782 4.20202594-003-3.25397411-003-6.03080086-010 2.5054
2 71.51556  -.501782-5.38389344-007-1.75630743-006 9.45343502-001-1.0014
1 66.95303  -.501101-5.60380652-003-3.21827605-002-1.31560195-010-1.6328
2 66.95303  -.501101 9.72961132-007 2.50720018-006 9.41323511-001-1.0420
1 64.15047  -.766432 7.26349233-003-6.12409366-003-1.05415335-009 1.3424
2 64.15047  -.766432-1.14101552-006-3.06261120-006 9.39071193-001-1.0742
1 59.43416  -.548001-8.34006036-003-2.83699052-002-3.93695111-010-3.9467
2 59.43416  -.548001 1.91310056-006 3.72124325-006 9.34739329-001-1.1304
1 56.38749 -1.181612 1.03418070-002-1.02234199-002-1.56078298-009 3.4352
2 56.38749 -1.181612-2.09625628-006-4.50653664-006 9.33384195-001-1.1776

R   .000 F 20.0000 A  96.848 C  26.610 M 4.680-005 S 4.640+000 E 81.0 T
1 89.95078 -5.509952-1.93003481-004-3.08029371-003-5.71833862-012-7.4570
2 89.95078 -5.509952 9.08982800-008 1.00341518-007 9.70444761-001-3.0651
1 89.72677 -4.972712 1.63144470-003-5.19614783-003-7.55528420-012-3.4144
```

```
2 89.72677 -4.972712-1.15447172-007-1.70629631-007 9.70000498-001-3.0777
1 85.57822  -.374531-9.40172227-004-3.21498699-002-2.93168319-012-3.6242
2 85.57822  -.374531 1.30888923-007 3.71334362-007 9.66683730-001-3.0583
1 83.08694  -.240632 2.02812662-003-2.06106337-003-1.15471073-010 5.1048
2 83.08694  -.240632-2.19895474-007-5.45562628-007 9.64558974-001-3.0479
1 79.12334  -.397391-2.51086711-003-2.78670846-002-2.04451764-011-2.5083
2 79.12334  -.397391 3.51979452-007 9.05249671-007 9.59300555-001-3.0774
1 77.11312  -.275102 3.30613877-003-3.18349389-003-3.01428101-010 2.1585
2 77.11312  -.275102-4.72247425-007-1.10789632-006 9.55758922-001-3.0714
1 73.44476  -.459411-4.59294498-003-2.38388672-002-8.37238039-011-9.7335
2 73.44476  -.459411 7.61633153-007 1.64506844-006 9.47922789-001-3.1771
1 71.63047  -.376812 5.62715752-003-5.50687622-003-5.23718499-010 9.5978
2 71.63047  -.376812-9.37261717-007-1.88752817-006 9.43363637-001-3.2154
1 67.93610  -.511771-7.17936538-003-1.97459643-002-2.53456083-010-2.5509
2 67.93610  -.511771 1.47960783-006 2.42963196-006 9.33020763-001-3.4446
1 66.14730  -.531512 8.40076327-003-9.21373558-003-7.26669051-010 2.5910
2 66.14730  -.531512-1.73095953-006-2.71428090-006 9.27492820-001-3.5861
1 62.41710  -.555411-9.07247630-003-1.51738158-002-6.00887652-010-4.8687
2 62.41710  -.555411 2.46706023-006 2.97335941-006 9.15089674-001-3.9828
1 60.47119  -.741961 1.05556111-002-1.39936338-002-8.50278722-010 5.0284
2 60.47119  -.741961-2.82933254-006-3.32145967-006 9.08521205-001-4.3171
1 56.76943  -.610202-9.88309338-003-1.11616928-002-1.13383616-009-7.4866
2 56.76943  -.610202 3.50012604-006 3.21396806-006 8.94910932-001-4.9218

R    .000 F 25.0000 A  96.848 C  26.610 M 4.680-005 S 4.640+000 E 81.0 T
1 89.95961 -6.296002 4.14965702-005-2.84423062-004-2.65959360-012-5.6880
2 89.95961 -6.296002 1.78132298-008 2.11544882-008 9.99736615-001-4.8076
1 89.75731 -5.702361 6.66626482-004-2.24545391-003-1.42592036-012-4.5403
2 89.75731 -5.702361-2.75980718-008-5.24136694-008 9.98948187-001-4.6248
1 88.74632  -.740711-1.14792565-003-2.27329135-002-3.50932488-012-3.4240
2 88.74632  -.740711 1.37327154-007 3.05298592-007 9.95857358-001-2.7427
1 86.21623  -.323792 2.80416722-003-3.36569911-003-5.43727906-011 1.7317
2 86.21623  -.323792-2.14213904-007-4.40146156-007 9.94571388-001-2.1050
1 82.17971  -.352921-2.85408430-003-2.34048092-002-1.54511952-011-1.7795
2 82.17971  -.352921 3.04256552-007 6.64677630-007 9.89881411-001-7.2338
1 80.61306  -.235442 3.28247802-003-3.42158275-003-1.68198547-010 1.6271
2 80.61306  -.235442-3.93487753-007-8.10396045-007 9.87260617-001 1.4355
1 77.39001  -.375861-4.55183111-003-1.89235786-002-6.26825926-011-6.8954
2 77.39001  -.375861 6.39205197-007 1.20104490-006 9.79809105-001 4.2323
1 76.07293  -.293152 5.18735842-003-5.37725317-003-2.98127936-010 6.9079
2 76.07293  -.293152-7.58520216-007-1.32923716-006 9.76186506-001 5.9032
1 72.92168  -.393581-6.79070043-003-1.46260866-002-1.92001999-010-1.8119
2 72.92168  -.393581 1.23595613-006 1.69775085-006 9.65308584-001 9.1862
1 71.71231  -.392682 7.51004531-003-8.79517768-003-3.95414586-010 1.8395
2 71.71231  -.392682-1.39636245-006-1.81699518-006 9.60419364-001 1.0552
1 68.56366  -.399602-8.11186887-003-9.74748202-003-4.66080674-010-3.3417
2 68.56366  -.399602 2.04390165-006 1.87442609-006 9.45522405-001 1.3631
1 67.33788  -.523181 8.86612176-003-1.34191969-002-3.99816361-010 3.3996
2 67.33788  -.523181-2.25887723-006-1.97178016-006 9.38771591-001 1.3818
1 64.22144  -.410722-7.88689498-003-5.51686942-003-8.83920132-010-4.6723
2 64.22144  -.410722 2.77423089-006 1.66314987-006 9.19652522-001 1.5219
1 62.85581  -.666111 8.66894366-003-1.77447477-002-3.15086311-010 4.7624
2 62.85581  -.666111-3.06723743-006-1.72537347-006 9.10496563-001 1.2858
1 59.81876  -.444912-7.08291441-003-2.62661430-003-1.40096686-009-5.7675
2 59.81876  -.444912 3.35079508-006 1.28801229-006 8.88060510-001 1.0387
1 58.20465  -.807741 7.90575973-003-2.10266041-002-1.87648097-010 5.8927
2 58.20465  -.807741-3.76355356-006-1.29681153-006 8.76700148-001 3.7316
```

```
R    .000 F 30.0000 A   96.848 C   26.610 M 4.680-005 S 4.640+000 E 81.0 T
1 89.95817 -6.801582 9.21416370-006-4.06682093-005-7.33488398-013-2.8557
2 89.95817 -6.801582 3.33612060-009 4.35358660-009 1.01745796+000-1.3958
1 89.77098 -6.172221 2.33388033-004-5.90389849-004-4.19298588-013-2.0551
2 89.77098 -6.172221-6.15653212-009-1.59281199-008 1.01659237+000-9.6001
1 89.86359 -3.077971-1.35654240-003-9.89436009-003-7.60637224-012-2.9350
2 89.86359 -3.077971 1.75959459-007 2.22063330-007 1.01431617+000 1.4027
1 89.29216 -1.802442 3.43013523-003-7.66506500-003-2.01165827-011 1.7334
2 89.29216 -1.802442-2.37264798-007-3.33145127-007 1.01360975+000 1.7748
1 84.46494  -.325201-3.52030885-003-2.00907697-002-1.34843656-011-1.4141
2 84.46494  -.325201 2.84209779-007 5.59975376-007 1.01064682+000 4.2818
1 83.12847  -.256792 3.77914089-003-4.37141588-003-9.63985307-011 1.3105
2 83.12847  -.256792-3.57363756-007-6.55233706-007 1.00918959+000 5.6036
1 80.14352  -.314231-4.87920851-003-1.52311159-002-5.40644682-011-5.3522
2 80.14352  -.314231 5.79850294-007 9.35443637-007 1.00445074+000 9.3897
1 79.07973  -.287462 5.28538768-003-6.02913019-003-1.77346710-010 5.3588
2 79.07973  -.287462-6.64902906-007-1.01968099-006 1.00235936+000 1.1205
1 76.30567  -.303281-6.47681038-003-1.05080334-002-1.66328987-010-1.3305
2 76.30567  -.303281 1.08696990-006 1.20814860-006 9.95344214-001 1.6763
1 75.35715  -.372092 6.98046980-003-9.46757186-003-2.19153317-010 1.3472
2 75.35715  -.372092-1.19382139-006-1.27433299-006 9.92226020-001 1.8781
1 72.65670  -.283522-6.66976604-003-5.72501693-003-3.90159866-010-2.1732
2 72.65670  -.283522 1.65506199-006 1.12109299-006 9.81998667-001 2.6303
1 71.70666  -.484141 7.19145511-003-1.37139540-002-1.82826587-010 2.2030
2 71.70666  -.484141-1.78910638-006-1.15557695-006 9.76869076-001 2.8041
1 69.08797  -.269702-5.52105560-003-2.30829942-003-6.97587266-010-2.6241
2 69.08797  -.269702 2.01539578-006 7.55881238-007 9.61933985-001 3.6731
1 68.03986  -.600941 6.02721120-003-1.69323110-002-9.76922293-011 2.6547
2 68.03986  -.600941-2.18522109-006-7.43441461-007 9.53540497-001 3.6948
1 65.52886  -.272232-4.25863202-003-2.97679551-004-1.05524151-009-2.8148
2 65.52886  -.272232 2.21227225-006 3.15480715-007 9.32598367-001 4.4015
1 64.31916  -.711151 4.72565473-003-1.89521380-002 1.70844382-012 2.8158
2 64.31916  -.711151-2.43150367-006-2.37438300-007 9.20637153-001 4.0183
1 61.92816  -.294362-3.26558537-003 9.93770678-004-1.46713304-009-2.9326
2 61.92816  -.294362 2.38145702-006-1.71861332-007 8.94900113-001 4.1165
1 60.51619  -.813321 3.65956590-003-2.04236524-002 1.19203678-010 2.8392
2 60.51619  -.813321-2.66933711-006 3.42168367-007 8.81666422-001 3.0688
1 58.24503  -.339362-2.46994873-003 2.02460142-003-1.95284630-009-3.0211
2 58.24503  -.339362 2.60870254-006-7.50150313-007 8.56219366-001 2.2998
1 56.59863  -.906611 2.72667289-003-2.18211829-002 2.76802057-010 2.7023
2 56.59863  -.906611-2.98982195-006 1.05839401-006 8.45654584-001 6.2300

R    .000 F 35.0000 A   96.848 C   26.610 M 4.680-005 S 4.640+000 E 81.0 T
1 89.95578 -7.154792 1.71951611-006-6.65459652-006-1.70822876-013-1.1727
2 89.95578 -7.154792 6.26939424-010 8.70561630-010 1.02251831+000-1.1474
1 89.78312 -6.518361 6.77289499-005-1.37311265-004-1.28066713-013-7.8820
2 89.78312 -6.518361-1.31226668-009-4.56250016-009 1.02154364+000-7.4143
1 89.93700 -4.242822 3.86689535-005-1.94379015-003-7.12259201-012-4.0835
2 89.93700 -4.242822 7.92866732-008 8.55532294-008 1.01944663+000 1.1845
1 89.62104 -3.394181 1.97066873-003-8.00630264-003-5.41413740-012-4.7556
2 89.62104 -3.394181-1.22145323-007-1.70501769-007 1.01870005+000 1.4662
1 86.43846  -.313841-3.94388248-003-1.61767765-002-1.34537301-011-1.0957
2 86.43846  -.313841 2.85340914-007 4.50308416-007 1.01656783+000 3.2084
1 85.15994  -.320462 4.36410488-003-6.50112930-003-5.23506863-011 1.0193
2 85.15994  -.320462-3.55639354-007-5.33603441-007 1.01554948+000 3.9114
1 82.24972  -.272241-5.03783813-003-1.21501258-002-4.91220250-011-4.0359
```

```
169        2 82.24972  -.272241 5.43721526-007 7.28336296-007 1.01222657+000 6.5597
170        1 81.31095  -.298492 5.24895213-003-7.24678044-003-1.05158253-010 4.0276
171        2 81.31095  -.298492-6.11081887-007-7.89565838-007 1.01078829+000 7.6739
172        1 78.80987  -.244612-5.64404484-003-7.40134291-003-1.44906575-010-9.1144
173        2 78.80987  -.244612 9.33826691-007 8.43385997-007 1.00627670+000 1.1729
174        1 77.97459  -.365431 5.98990329-003-1.03043913-002-1.21564057-010 9.2064
175        2 77.97459  -.365431-1.01311328-006-8.85114169-007 1.00417314+000 1.3088
176        1 75.62908  -.217262-4.96455305-003-3.48515713-003-3.13408421-010-1.3144
177        2 75.62908  -.217262 1.26447735-006 6.82723261-007 9.98098433-001 1.9012
178        1 74.77820  -.456311 5.33993304-003-1.34751227-002-8.85680166-011 1.3299
179        2 74.77820  -.456311-1.35897561-006-6.93880402-007 9.94550772-001 2.0513
180        1 72.56655  -.199072-3.68672452-003-1.12099861-003-5.27402795-010-1.4485
181        2 72.56655  -.199072 1.40992418-006 3.86078096-007 9.85836089-001 2.8270
182        1 71.62053  -.548311 4.04662930-003-1.54822026-002-3.42179783-011 1.4585
183        2 71.62053  -.548311-1.52351126-006-3.53674118-007 9.79646951-001 2.9281
184        1 69.54959  -.193242-2.58837634-003 1.79758459-004-7.71987724-010-1.4315
185        2 69.54959  -.193242 1.45006429-006 5.46745862-008 9.66708526-001 3.7370
186        1 68.46246  -.634701 2.90467023-003-1.66300139-002 2.58675477-011 1.4094
187        2 68.46246  -.634701-1.58884565-006 3.66019841-008 9.57122207-001 3.5885
188        1 66.53448  -.199332-1.73459633-003 9.74064271-004-1.04989406-009-1.3279
189        2 66.53448  -.199332 1.46168267-006-3.18594022-007 9.39793691-001 4.0768
190        1 65.27999  -.715771 1.97376651-003-1.74074376-002 9.42113323-011 1.2209
191        2 65.27999  -.715771-1.63377435-006 4.88440357-007 9.28504840-001 3.4118
192        1 63.49120  -.217322-9.96977105-004 1.55366775-003-1.36881209-009-1.1086
193        2 63.49120  -.217322 1.48137963-006-7.67130373-007 9.10345115-001 3.2940
194        1 62.05259  -.791751 1.10823596-003-1.80875934-002 1.79295710-010 8.2145
195        2 62.05259  -.791751-1.69562365-006 1.04387406-006 9.01143454-001 2.1334
196        1 60.39516  -.248472-2.39132431-004 2.07731486-003-1.74026334-009-6.7406
197        2 60.39516  -.248472 1.52686277-006-1.33272954-006 8.86461414-001 1.5787
198        1 58.75931  -.861241 1.39434616-004-1.88362205-002 2.94850606-010 2.9668
199        2 58.75931  -.861241-1.78866654-006 1.76271519-006 8.80726665-001 2.4958
200        1 57.22259  -.296182 6.81471960-004 2.65892252-003-2.18246249-009 1.6829
201        2 57.22259  -.296182 1.60517165-006-2.07563082-006 8.70145038-001-3.8564
202
203        END
204        START
```

IV. PROGRAM LAYOUT

This section describes only the basic features of the pulse shape program listed in the appendix. In particular, plot and label routines PLSPEC, BORDER, SYMBOL, CURVE, PLOT, WOPLOT, PLOT12, PLOT3 and PLLABL, are not described.

MAIN controls the program flow. Subroutines in the order of their call are described below:

SUBROUTINE INPUT

INPUT, called from MAIN, reads in NAMELIST and mode data.

SUBROUTINE HTGAIN(IOPT, FREQ, SIGMA, EPSR, ALPHA, NRMODE, TP, Z, HG)

HTGAIN, called from MAIN, evaluates the modal height gain functions that appear in equation 46. The arguments of HTGAIN are:

IOPT - option flag set for 1 in present program.

FREQ - frequency (kHz).

SIGMA - ground conductivity (s/m).

EPSR - real dielectric constant of the ground.

ALPHA - earth curvature constant (3.14×10^{-4} km^{-1}).

NRMODE - number of modes at any one of the input frequencies.

TP - complex ground eigenangle in degrees.

z - altitude at which height gain is evaluated (km).

HG - height gain.

SUBROUTINE MDHNKL(Z, H1, H2, H1PRME, H2PRME, THETA, IDBG)

This subroutine is called by HTGAIN. It evalutes the modified Hankel functions of order 1/3 and their derivatives according to the methods of reference 9. The arguments of MDHNKL are:

- Z - argument of modified Hankel functions of order 1/3 and their derivatives.
- H1, H2 - modified Hankel functions of order 1/3.
- H1PRME, H2PRME - derivatives of modified Hankel functions of order 1/3.
- THETA - not used in present program.
- IDBG - not used in present program.

SUBROUTINE FUNSPL(MD, LF)

FUNSPL is called from MAIN. Inputs to FUNSPL are a mode index, MD, which takes on values 1 through the maximum number of modes read in for a given frequency; and the index, LF, for the quantity that is to be approximated as a function of frequency by a cubic spline. LF takes on the integer values 1 through 4.

SUBROUTINE FUNCVF(MD, LF)

This subroutine called by FUNSPL selects for LF = 1(2), the real (imaginary) part of the excitation factor for fitting to a cubic spline. If LF = 3(4), the real (imaginary) part of the eigenangle is selected for fitting to a cubic spline.

SUBROUTINE SPLINE(X, Y, B, C, D, N)

This subroutine called from FUNSPL determines the coefficients B, C, D of a cubic spline interpolating the given curve [X(I), Y(I), I = 1, 2, . . . , N]. If X(I).LE.XX.LE.X(I+1) and H = XX - X(I), then the interpolated value at XX is F(XX) = Y(I) + B(I) * H + C(I) * H ** 2 + D(I) * H ** 3. The interpolated value is evaluated by using the function SPEVAL discussed subsequently.

SUBROUTINE TPLOT(FREQ, FL, FO, FC, DELTAF, PULSEW, PULSED, NUMPLS, IFLGTR, NRPT1, NF)

TPLOT, called from MAIN, sets up the arrays for plotting the transmitter spectrum. The arguments of TPLOT are as follows:

FREQ - input frequency in Hz.

FL - lowest frequency in Fourier integral evaluation.

FO - carrier frequency in Hz.

FC - chip frequency in Hz.

DELTAF - frequency interval in Hz, at which transmitter spectrum is calculated.

PULSEW - pulse width in μs when used with IFLGTR = 1; and 1/e pulse half width in μs when used with IFLGTR = 2.

PULSED - pulse delay in μs (used with IFLGTR = 1 or 2).

NUMPLS - number of pulses (used with IFLGTR = 1 or 2).

IFLGTR - transmitter flag. IFLGTR = 1 corresponds to sequence of square wave pulses. IFLGTR = 2 corresponds to sequence of Gaussian pulses. IFLGTR = 3 corresponds to a differential phase-encoded signal waveform composed by frequency shift keying (FSK) a carrier frequency with a pseudonoise sequence of pulses or chips. The FSK index is 0.5. IFLGTR = 4 corresponds to Williams' source for slow wave tail calculation.

NRPT1 - number of frequency points between FU and FL used in Fourier evaluation (FU is the highest frequency in Fourier integral evaluations).

NF - the number of frequencies read in.

SUBROUTINE TRXMTR(K, F, FO, FC, PULSEW, PULSED, NUMPLS, IFLGTR, LABELT, XMTR)

TRXMTR, called from TPLOT, evaluates the transmitter spectrum. The arguments of TRXMTR are:

K - integer index of frequencies for which transmitter spectrum is evaluated.

F - frequency in Hz.

LABELT - transmitter label.

XMTR - transmitter evaluation.

The remaining arguments are defined in TPLOT.

SUBROUTINE RPLOT(FREQ, FL, F1, F2, F3, DELTAF, P, Q, NRPT1, NF)

RPLOT, called from MAIN, sets up the arrays for plotting the receiver spectrum. The arguments of RPLOT are:

F1, F2, F3 - frequencies f_1, f_2 and f_3, respectively, in equation 6.

P, Q - exponents P and Q in equation 6.

All other arguments are the same as TPLOT. RPLOT calls RECVR and, like TPLOT, calls the controlling plot routine PLSPEC, which, in turn, calls BORDER, SYMBOL, CURVE, and PLOT.

SUBROUTINE RECVR(K, F, FO, F1, F2, F3, LABELR, P, Q, RCVR)

Called from RPLOT, this subroutine calculates the receiver spectrum according to equation 6.

K - integer index of frequency for which receiver spectrum is evaluated.

F - frequency in Hz.

FO - carrier frequency in Hz.

F1, F2,
F3 - frequencies f_1, f_2 and f_3 of equation 6.

LABELR - receiver label.

P, Q - exponents of equation 6.

RCVR - receiver evaluation.

SUBROUTINE CPPLOT(FREQ, FL, FO, FC, DELTAF, NRPT1, NF, BANDW, RHO)

CPPLOT, called from MAIN, sets up the arrays for plotting the channel spectra and the product spectra consisting of the transmitter, receiver, and channel. All arguments, except RHO, have been previously defined in TPLOT and RPLOT. The argument RHO is:

RHO - range (km).

CPPLOT calls CHANEL and, like TPLOT, calls the controlling plot routine PLSPEC, which, in turn, calls BORDER, SYMBOL, CURVE, and PLOT as well as a utility labeling subroutine called PLLABL.

SUBROUTINE CHANEL(F, RHO, CHNL)

CHANEL, called from CPPLOT, evaluates the channel spectra for frequency F (Hz) and range RHO (km) according to equation 46. The channel evaluation is placed in CHNL.

FUNCTION SPEVAL(XVAL, X, Y, B, C, D, N, INIT)

SPEVAL, called from CHANEL, evaluates the interpolating cubic spline for the data [X(I), Y(I), I = 1, . . ., N at XVAL]. INIT is an estimate of the interval where XVAL lies, X(INIT) .LE. XVAL .LE. X(INIT + 1), but need not be used. Set INIT = 0 if there is no estimate. On return, INIT will contain the interval number.

SUBROUTINE NLOGN(N, X, Y, SIGNT, A, B)

Called from MAIN when INTFLG .EQ. 0, NLOGN calculates integrals of the form (s = SIGNT).

$$\exp[-i2\pi sA\tau] \int_A^B \big(x(f)+iy(f)\big) \exp(i2\pi sf\tau)df$$

$$= \int_0^{B-A} \big(x(f + A)+iy(f + A)\big) \exp(i2\pi sf\tau)df \qquad (48)$$

by the fast Fourier transform technique of Cooley and Tukey (ref 10). This makes use of digital evaluations at the frequencies

$$f(L) = \frac{L - 1}{2^n}(B - A) \; ; \quad L = 1, 2, : ...., 2^n \qquad (49)$$

and the method yields evaluations for the times

$$\tau(K) = \frac{K - 1}{B - A} \; ; \quad K = 1, 2, ..., 2^n . \qquad (50)$$

Real and imaginary parts of the integral are then stored in X(K) and Y(K), respectively. The weight factors and endpoint corrections supplied in the earlier work (ref 1) were in error and have been abandoned in the present work. Though the consequent error in the output parameter range of interest (TAUO to TAUMAX) in reference 1 is quite small, it is strongly recommended that the present program be used instead of reference 1 even though the interest may be solely in vertical E at the ground produced by a ground-based transmitter.

The quantity s = SIGNT takes on the values ±1 and simply allows for plus or minus transforms as desired. It should be observed that although the region of significance of the integrand of equation 48 may be quite small, the integration limits A, B may of necessity be quite large in order to achieve a desired time resolution (see eq 50). N must be chosen to give small enough step sizes in the region where the integrand is significant. Specifically, step sizes must be small compared with distances (in frequency units) over which the integrand changes appreciably.

<u>SUBROUTINE FILON</u>(N, X, Y, TAU, FU, FL, SUM, SUMP)

Called from MAIN when INTFLG .NE. 0, FILON calculates integrals of the form

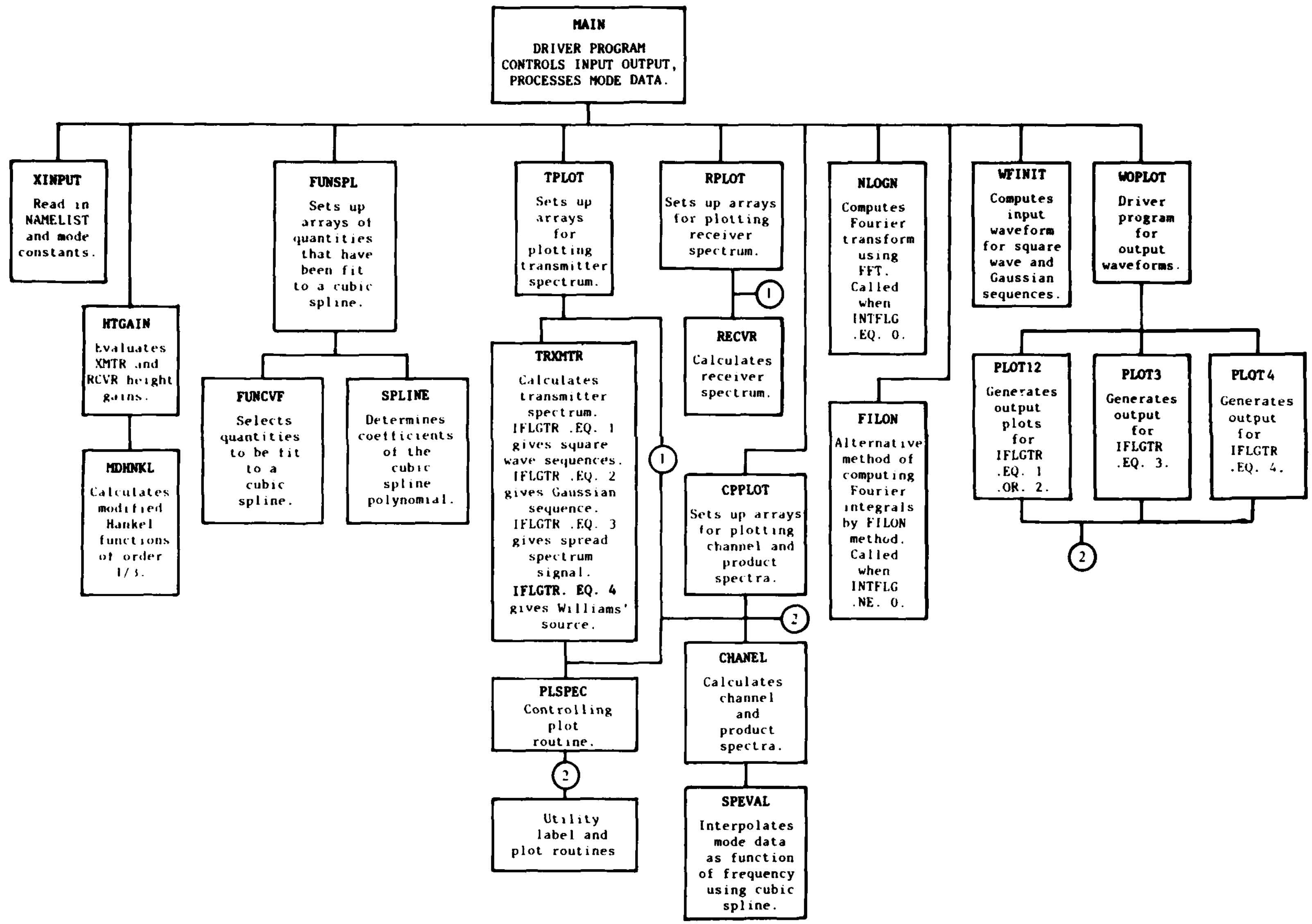
$$\int_{FL}^{FU} \left(x(f)+iy(f)\right) \exp(2\pi i \tau f)\,df \tag{51}$$

by Filon's method (ref 4), using $2^N + 1$ point evaluations in the range FU - FL. In particular, SUM represents the N point evaluation and SUMP represents the $2^{N-1} + 1$ point evaluation. The relative error |SUM-SUMP|/|SUM| is calculated and printed in MAIN. As discussed in the introduction, the evaluation is considerably slower than can be achieved by using NLOGN, but the method is used to give a measure of the error in the evaluation as noted above. Note that FL and FU are chosen on the basis that the integrand be sufficiently small at the end points, unlike the situation with NLOGN, where the limits are chosen on the basis of the time resolution required. Accordingly, the number of point evaluations (determined by N) using the Filon method can be appreciably smaller than the point evaluations using NLOGN. Despite this tradeoff, the NLOGN routine appears to be appreciably faster.

SUBROUTINE INITWF(IFLGTR, PULSEW, PULSED, NUMPLS, TAUO, TAUMAX, PLOTX3 PLOTY3)

Called from MAIN, this subroutine calculates the input waveforms for IFLGTR (eq 1 or 2). PULSEW, PULSED, NUMPLS and IFLGTR have been previously defined (see, e.g., TPLOT). TAUO and TAUMAX are the minima and maxima abscissa times in ms on the output waveform plots. PLOTX3 and PLOTY3 contain the output plot array data.

A chart showing the essential flow of the pulse program appears below.


MAIN
DRIVER PROGRAM
CONTROLS INPUT OUTPUT,
PROCESSES MODE DATA.
XINPUT
Read in NAMELIST and mode constants.
HTGAIN
Evaluates XMTR and RCVR height gains.
MDHNKL
Calculates modified Hankel functions of order 1/3.
FUNSPL
Sets up arrays of quantities that have been fit to a cubic spline.
FUNCVF
Selects quantities to be fit to a cubic spline.
SPLINE
Determines coefficients of the cubic spline polynomial.
TPLOT
Sets up arrays for plotting transmitter spectrum.
TRXMTR
Calculates transmitter spectrum. IFLGTR .EQ. 1 gives square wave sequences. IFLGTR .EQ. 2 gives Gaussian sequence. IFLGTR .EQ. 3 gives spread spectrum signal. IFLGTR. EQ. 4 gives Williams' source.
PLSPEC
Controlling plot routine.
Utility label and plot routines
RPLOT
Sets up arrays for plotting receiver spectrum.
RECVR
Calculates receiver spectrum.
CPPLOT
Sets up arrays for plotting channel and product spectra.
CHANEL
Calculates channel and product spectra.
SPEVAL
Interpolates mode data as function of frequency using cubic spline.
NLOGN
Computes Fourier transform using FFT. Called when INTFLG .EQ. 0.
FILON
Alternative method of computing Fourier integrals by FILON method. Called when INTFLG .NE. 0.
WFINIT
Computes input waveform for square wave and Gaussian sequences.
WOPLOT
Driver program for output waveforms.
PLOT12
Generates output plots for IFLGTR .EQ. 1 .OR. 2.
PLOT3
Generates output for IFLGTR .EQ. 3.
PLOT4
Generates output for IFLGTR .EQ. 4.
1
2

V. DESCRIPTION OF OUTPUT

The sample output shown below begins with the namelist output followed by the run identification. The mode data come next. For each frequency (given in increasing order), the number of modes, the real and imaginary parts of the eigenangle, and their modal equivalents of attenuation rate and phase velocity normalized to free-space velocity are listed.

The calculated output of the pulse shape program is presented following the sample output. The transmitter, receiver, channel, and product (XMTR * RCVR * CHNL) spectra are given as a function of frequency. Not all 2049 (i.e., $2^{NFFT} + 1$ with NFFT = 11) lines are listed. The printout is controlled by the namelist variables NPRNT and INTPRT.

Following the spectra output, output is presented pertaining to the envelope of the time signature of the output waveform, $G(x, z_R, t; z_T)$, given by equation 1. Specifically, the time TAU in seconds is given, along with the envelope in dB/μV/m/kW in the spirit of the normalization discussed in section II.

The signal envelope in absolute units for times less than -0.3 ms should be identically zero (or $-\infty$ on a dB scale). The fact that this is not so is believed to be associated with frequency truncation effects and/or discontinuities in the third and higher derivatives of the interpolated mode data. The principal outputs of the program are the plots. In the present instance, figures 1 through 6 are generated. These are

1. Transmitter spectrum versus frequency.
2. Receiver spectrum versus frequency.
3. Channel spectrum versus frequency.
4. Product (transmitter * receiver * channel) spectrum versus frequency.
5. Input and output envelopes normalized to unity.
6. Waveform output, including carrier frequency normalized to unity.

Shown in figure 5 is the maximum of the envelope (SIGNAL MAX) expressed in units of dB/μV/m/kW. The right-hand scale on the plot gives the envelope in dB relative to SIGNAL MAX.

Figures 1 through 6 just discussed apply to IFLGTR = 1 (i.e., square-wave envelopes). Figure 7 is the envelope output for IFLGTR = 2 (i.e., Gaussian envelope output). Figures 8 and 9 show output for IFLGTR = 3 (i.e., the correlator output for an MSK format). Figure 8 shows the amplitude of the correlator output and figure 9 the phase of the correlation vector given by equation 23. It can be observed from the labels that figures 8 and 9 apply to a different ionospheric profile (i.e., channel) than that used for figures 1 through 7.

Corresponding to IFLGTR = 4, figure 10 shows the transmitter spectrum for Williams' source, figure 11 shows a representative receiver spectrum used for slow tail measurements, figure 12 shows the channel spectrum, and figure 13 gives the product spectrum. Lastly, the slow wave tail output for the Williams' source is shown in figure 14.

SAMPLE OUTPUT

```
NAME
 $DATUM
NFFT =          11,FREQU = .10000000+003,FRECL = .00000000    ,INTPRT =          20,TAUMAX = .30000000-002,FREQ0 = .22000000+002,
PULSEW = .10000000+005,RHOMIN = .10000000+004,DELRHO = .10000000+004,RHOMAX = .10000000+004,TALT = .00000000    ,
RALT = .00000000    ,INCL = .00000000    ,THETA = .00000000    ,ICOMP =           1,IFLGTR =           1,INTFIG =           0,
NORNT =          20,TAU0 =-.10000000-002,NUMTAU =          41,CHIPFR = .10000000+001,NUMPLS =           2,PULSED = .60000000+003,
FREQ1 = .10000000-001,FREQ2 = .25000000+002,FREQ3 = .13670000+001,P = .00000000    ,Q = .30000000+001,IPLOT =           1,
,PLOT1 =           1
$END

DATA

BETA=0.5, HPRIME=67.0

START
```

NMF	FREQ KHZ	THETAR DEGREES	THETAI DEGREES	ATT DB	PHVCC
6	10.00000	89.54299	-1.60001	.40535	.99984
		83.15179	-.69119	2.58080	1.00711
		76.02858	-.33187	2.54585	1.03047
		71.53906	-.97249	9.78451	1.05110
		64.67405	-.42003	5.70052	1.10630
		60.43294	-1.73603	27.22175	1.14919
10	15.00000	89.57716	-4.44793	.45495	.99700
		89.42791	-2.62781	1.25032	.99900
		82.12511	-.33932	2.21823	1.00951
		78.92260	-.34558	3.19450	1.01347
		74.30239	-.42357	5.44279	1.03341
		71.51550	-.50178	7.52211	1.05436
		66.95303	-.50110	9.34961	1.08670
		64.15047	-.76643	15.92681	1.11108
		59.43416	-.54809	13.28158	1.16133
		56.38749	-1.18161	31.17653	1.20051
13	20.00000	89.25078	-5.50935	.30125	.99539
		89.72677	-4.97271	1.50879	.99826
		85.57822	-.37453	1.84443	1.00236
		83.08694	-.24063	1.84048	1.00731
		79.12334	-.39739	4.70506	1.01827
		77.11312	-.27510	3.80894	1.02983
		73.44476	-.45941	8.31849	1.04321
		71.63047	-.37681	7.54503	1.05367
		67.93610	-.51177	12.21029	1.07898
		66.14730	-.53151	13.[illegible]843	1.09334
		62.41710	-.55541	16.34246	1.12818
		60.47119	-.74195	23.23826	1.14919
		56.76943	-.61020	21.24930	1.19543
16	25.00000	89.95961	-6.29600	.35325	.99399
		89.75731	-5.70236	1.92175	.99506
		88.74632	-.74071	1.23732	1.00016
		86.21623	-.32379	1.64725	1.00217
		82.17971	-.35292	3.81440	1.00937
		80.61306	-.27544	3.05924	1.01359
		77.39001	-.37586	6.51787	1.02470
		76.07293	-.29315	5.60454	1.03027
		72.92168	-.39353	9.18132	1.04611
		71.71231	-.32268	9.78758	1.05317
		68.56356	-.39960	11.60042	1.07429
		67.33788	-.52318	16.01209	1.08352
		64.22144	-.41072	14.19827	1.11049
		62.85521	-.66611	24.14009	1.12359
		59.81876	-.44491	17.75598	1.15678
		58.20465	-.80774	33.80146	1.17644
20	30.00000	89.75817	-6.92139	.47743	.99300
		89.77098	-6.17272	2.35[illegible]18	.99123
		89.86259	-3.07797	.6[illegible]83	.99956
		89.23216	-1.30244	2.19281	.99578
		84.46494	-.32520	2.9[illegible]986	1.00457

		83.12947	-.25679	2.92851	1.00722
		80.14352	-.31425	5.12723	1.01497
		79.07973	-.29746	5.19061	1.01643
		76.30567	-.30308	6.84392	1.02924
		75.35715	-.37209	8.96590	1.03655
		72.65670	-.28352	8.05600	1.04762
		71.70666	-.48414	14.48506	1.05319
		69.08797	-.26970	9.17549	1.07050
		66.03926	-.60094	21.42119	1.07817
		65.52886	-.27223	10.74985	1.09268
		64.31916	-.71115	29.37626	1.10952
		61.92816	-.29436	13.20353	1.13331
		60.51619	-.81332	38.15711	1.14066
		58.24503	-.33936	17.02409	1.17603
		56.59863	-.90361	47.57449	1.19769
23	35.00000	89.95578	-7.15479	.61566	.99225
		89.78312	-6.51836	2.73976	.99357
		89.93700	-4.24282	.51927	.99727
		89.62104	-3.39418	2.49733	.99927
		86.43846	-.31384	2.16806	1.00192
		85.15994	-.32046	3.09685	1.00356
		82.24972	-.27224	4.09270	1.00921
		81.31095	-.29849	5.01464	1.01160
		78.80987	-.24461	5.27896	1.01937
		77.97459	-.36543	8.46073	1.02242
		75.62908	-.21726	5.99658	1.03229
		74.77820	-.45631	13.32323	1.03633
		72.56665	-.19907	6.63239	1.04314
		71.62053	-.54831	19.22621	1.05370
		69.54959	-.19324	7.50429	1.06726
		68.40246	-.63470	25.91188	1.07500
		66.53748	-.19033	8.82955	1.09015
		65.27999	-.71577	33.28718	1.10080
		63.49120	-.21732	10.78664	1.11748
		62.05259	-.79175	41.26530	1.13191
		60.39516	-.24847	13.65024	1.15014
		58.75931	-.86124	49.67367	1.16946
		57.22259	-.29618	17.83124	1.18936

RHO = 1000.

FREQ(HZ)	XMTR R	XMTR I	RCVR R	RCVR I	CHNL R	CHNL I	XMTR*RCVR*CHNL REAL	XMTR*RCVR*CHNL IMAG	K
.0000	9.9205-002	.0000	-1.6270-004	.0000	.0000	.0000	.0000	.0000	1
4.8828+001	9.8662-002	-8.0769-004	-1.6271-004	-4.5997-005	.0000	.0000	.0000	.0000	2
9.7656+001	9.6733-002	-2.3436-003	-1.6273-004	-9.1997-005	.0000	.0000	.0000	.0000	3
1.4648+002	9.2612-002	-4.9019-003	-1.6277-004	-1.3801-005	.0000	.0000	.0000	.0000	4
1.9531+002	8.5241-002	-8.5556-003	-1.6282-004	-1.8403-005	.0000	.0000	.0000	.0000	5
2.4414+002	7.3680-002	-1.2065-002	-1.6228-004	-2.3016-005	.0000	.0000	.0000	.0000	6
2.9297+002	5.7456-002	-1.4058-002	-1.6296-004	-2.7612-005	.0000	.0000	.0000	.0000	7
3.4180+002	3.6984-002	-1.2664-002	-1.6306-004	-3.2219-005	.0000	.0000	.0000	.0000	8
3.9063+002	1.3204-002	-6.0349-003	-1.6317-004	-3.6830-005	.0000	.0000	.0000	.0000	9
4.3945+002	-1.1436-002	6.9326-003	-1.6329-004	-4.1444-005	.0000	.0000	.0000	.0000	10
4.8828+002	-3.4476-002	2.6310-002	-1.6342-004	-4.6061-005	.0000	.0000	0000	.0000	11
5.3711+002	-5.2891-002	5.2062-002	-1.6357-004	-5.0682-005	.0000	.0000	.0000	.0000	12
5.8594+002	-6.4180-002	8.3067-002	-1.6374-004	-5.5309-005	.0000	.0000	.0000	.0000	13
6.3477+002	-6.6611-002	1.1455-001	-1.6392-004	-5.9938-005	.0000	.0000	.0000	.0000	14
6.8359+002	-5.9644-002	1.4402-001	-1.6412-004	-6.4573-005	.0000	.0000	.0000	.0000	15
7.3242+002	-4.4105-002	1.6792-001	-1.6433-004	-6.9214-005	.0000	.0000	.0000	.0000	16
7.8125+002	-2.2105-002	1.8320-001	-1.6455-004	-7.3860-005	.0000	.0000	.0000	.0000	17
8.3008+002	3.2036-003	1.8825-001	-1.6479-004	-7.8513-005	.0000	.0000	.0000	.0000	18
8.7891+002	2.8290-002	1.8216-001	-1.6504-004	-8.3173-005	.0000	.0000	.0000	.0000	19
9.2773+002	4.9505-002	1.6563-001	-1.6531-004	-8.7840-005	.0000	.0000	.0000	.0000	20
1.9043+003	-8.2067-002	1.0223-002	-1.7390-004	-1.8330-004	.0000	.0000	.0000	.0000	40
2.8809+003	-1.0690-002	-9.4937-003	-1.8913-004	-2.6515-004	.0000	.0000	.0000	.0000	60
3.8574+003	-5.1002-002	5.4545-002	-2.1211-004	-3.9727-004	.0000	.0000	.0000	.0000	30
4.8340+003	-7.3307-002	-2.1490-002	-2.4500-004	-5.2496-004	.0000	.0000	.0000	.0000	100
5.8105+003	-2.3779-002	-2.0419-001	-2.9044-004	-6.7143-004	.0000	.0000	.0000	.0000	120
6.7871+003	1.2975-001	2.3502-002	-3.5976-004	-8.4782-004	.0000	.0000	.0000	.0000	140
7.7637+003	7.1603-002	5.1570-002	-4.3976-004	-1.0620-003	.0000	.0000	.0000	.0000	160
8.7402+003	-7.2823-003	4.3532-003	-5.5809-004	-1.3265-003	.0000	.0000	.0100	.0000	180
9.7168+003	3.9900-002	1.5811-002	-7.2637-004	-1.6600-003	.0000	.0000	.0000	.0000	200
1.0693+004	-2.1086-002	2.3692-001	-9.6313-004	-2.1045-003	-3.9845-002	-6.0310-002	-2.6169-005	-9.0067-006	220
1.1670+004	-1.3591-001	-9.3191-002	-1.3217-003	-2.6862-003	-5.2228-002	-1.7950-003	4.5715-006	-2.5256-005	240
1.2646+004	-1.4959-001	-1.2705-001	-1.8657-003	-3.4630-003	-2.7002-002	2.5605-002	-1.4458-005	-2.6550-005	260
1.3623+004	1.1641-001	-2.5305-002	-2.7100-003	-4.5204-003	-5.0607-003	1.1165-002	9.2560-006	-2.7140-006	280
1.4600+004	-6.8970-003	-9.0729-004	-4.1004-003	-6.0513-003	-1.3951-002	1.1476-002	-8.2622-007	-3.6525-007	300
1.5576+004	1.0472-001	-2.3412-001	-6.4803-003	-8.2112-003	-6.9794-003	2.7702-002	-8.3298-007	-7.6443-005	320
1.6553+004	7.6310-002	1.5563-001	-1.0803-002	-1.1256-002	3.0180-002	1.8356-002	7.3743-005	-6.3062-005	340
1.7529+004	2.7953-001	4.0390-001	-1.8211-002	-1.5570-002	3.7550-002	-1.6067-002	-1.6005-004	-4.7260-004	360
1.8506+004	-4.5623-001	-1.1840-001	-3.6815-002	-2.0293-002	2.6049-002	-3.6396-002	8.7805-004	-1.7564-004	380
1.9482+004	-1.0039-001	4.7005-003	-7.6237-002	-1.6848-002	2.3192-004	-4.0358-002	8.7037-005	-4.0426-004	400
2.0459+004	-1.1771-001	1.4462-001	-1.6261-001	7.8146-002	1.4350-002	-1.8845-002	-2.3161-004	-7.7390-004	420
2.1436+004	2.0832-002	-2.4959-001	-2.4386-001	2.2901-001	1.3845-002	-5.7592-002	4.8093-003	-3.0610-003	440
2.2412+004	-1.2514+000	-4.5723+000	4.4101-001	7.1350-001	-2.2310-002	-5.2537-002	-2.1749-001	-7.2025-002	460
2.3389+004	-4.1110+000	-3.7550+000	7.1402-001	-5.9500-001	-3.8731-002	-4.4597-002	1.8708-001	2.3943-001	480
2.4365+004	-9.2977-001	-1.3498-001	-2.1704-001	-4.9771-001	[illegible]	[illegible]	-1.9245-003	-2.2872-002	500
2.5342+004	3.5296-001	-1.2718-001	-1.8716-001	-5.6418-002	-3.0470-002	-3.1806-003	2.4590-003	8.0694-005	520
2.6318+004	1.0733-002	1.1474-002	-8.9032-002	1.5711-002	-3.1460-002	-2.1243-003	3.3425-005	2.0945-005	540
2.7295+004	5.1655-002	-5.2451-001	-4.2150-002	2.1671-002	-3.6514-002	2.3066-003	-3.7584-004	-8.3319-004	560
2.8271+004	8.2054-002	3.0312-001	-2.1752-002	1.6665-002	-3.1782-002	2.7631-002	3.6295-004	-2.3193-005	580
2.9248+004	3.0445-001	1.6146-001	-1.2077-002	1.2154-002	9.2482-004	5.0957-002	-5.9765-005	-1.7318-004	600
3.0225+004	-2.6349-001	-3.3250-002	-7.1519-003	8.6655-003	1.5906-002	1.2743-002	6.7941-005	-1.1689-005	620
3.1201+004	-8.7557-003	1.5387-002	-4.4753-003	6.5440-003	1.1254-002	-1.0432-002	-2.0751-006	-7.6584-007	640
3.2178+004	-6.2772-002	1.0622-001	-2.9222-003	4.9257-003	-1.0563-003	-2.4088-003	-1.1308-006	1.4726-006	660
3.3154+004	5.3612-002	-8.2705-002	-1.9962-003	3.7809-003	1.2952-002	1.2714-003	2.6148-006	4.3781-006	680
3.4131+004	-1.4627-001	-1.6749-001	-1.4037-003	2.9561-003	2.6927-002	-2.0488-002	1.6177-005	-2.0160-005	700

3.5107+004	1.5423-001	1.1582-001	-1.0146-003	2.3503-003	.0000	.0000	.0000	.0000	720
3.6084+004	5.0616-002	-3.0609-002	-7.5104-004	1.9970-003	.0000	.0000	.0000	.0000	740
3.7061+004	-1.6508-002	1.0548-002	-5.0753-004	1.5520-003	.0000	.0000	.0000	.0000	760
3.8037+004	-2.4826-002	2.1333-003	-4.3558-004	1.2853-003	.0000	.0000	.0000	.0000	780
3.9014+004	3.3758-002	1.5303-001	-3.4141-004	1.0760-003	.0000	.0000	.0000	.0000	800
3.9990+004	-4.2561-002	-1.1719-001	-2.7074-004	9.0071-004	.0000	.0000	.0000	.0000	820
4.0967+004	-9.3110-002	-1.9171-003	-2.1744-004	7.7597-004	.0000	.0000	.0000	.0000	840
4.1943+004	8.0803-002	-5.5509-003	-1.7663-004	6.6724-004	.0000	.0000	.0000	.0000	860
4.2920+004	2.4970-003	5.3309-004	-1.4497-004	5.7787-004	.0000	.0000	.0000	.0000	880
4.3896+004	2.0474-002	-7.6917-002	-1.2011-004	5.0401-004	.0000	.0000	.0000	.0000	900
4.4873+004	-2.1673-002	6.4895-002	-1.0036-004	4.4223-004	.0000	.0000	.0000	.0000	920
4.5850+004	9.5716-002	5.4095-002	-8.4524-005	3.9021-004	.0000	.0000	.0000	.0000	940
4.6826+004	-8.5026-002	-4.3173-002	-7.1608-005	3.4611-004	.0000	.0000	.0000	.0000	960
4.7803+004	-1.0231-002	2.0872-002	-6.1223-005	3.0848-004	.0000	.0000	.0000	.0000	980
4.8779+004	-9.4881-003	9.9832-003	-5.2600-005	2.7616-004	.0000	.0000	.0000	.0000	1000
4.9756+004	2.7042-002	-1.2513-002	-4.5450-005	2.4826-004	.0000	.0000	.0000	.0000	1020
5.0732+004	-4.8807-002	-8.1914-002	-3.9481-005	2.2405-004	.0000	.0000	.0000	.0000	1040
5.1709+004	4.2072-002	7.2397-002	-3.4465-005	2.0292-004	.0000	.0000	.0000	.0000	1060
5.2685+004	4.3868-002	-2.0576-002	-3.0225-005	1.8441-004	.0000	.0000	.0000	.0000	1080
5.3662+004	-3.1910-002	9.6241-003	-2.6622-005	1.6812-004	.0000	.0000	.0000	.0000	1100
5.4639+004	-7.0073-003	-4.5142-003	-2.3544-005	1.5372-004	.0000	.0000	.0000	.0000	1120
5.5615+004	1.4155-003	6.4289-002	-2.0901-005	1.4095-004	.0000	.0000	.0000	.0000	1140
5.6592+004	6.9051-003	-5.8580-002	-1.8622-005	1.2958-004	.0000	.0000	.0000	.0000	1160
5.7568+004	-6.4599-002	-1.0033-002	-1.6613-005	1.1942-004	.0000	.0000	.0000	.0000	1180
5.8545+004	5.0029-002	1.4354-002	-1.4930-005	1.1032-004	.0000	.0000	.0000	.0000	1200
5.9521+004	-8.0597-004	-9.1837-003	-1.3431-005	1.0214-004	.0000	.0000	.0000	.0000	1220
6.0498+004	1.2328-002	-2.2853-002	-1.2116-005	9.4760-005	.0000	.0000	.0000	.0000	1240
6.1475+004	-2.5523-002	2.2149-002	-1.0931-005	8.8091-005	.0000	.0000	.0000	.0000	1260
6.2451+004	5.1127-002	4.4233-002	-9.9409-006	8.2047-005	.0000	.0000	.0000	.0000	1280
6.3428+004	-4.1823-002	-4.5600-002	-9.0382-006	7.6555-005	.0000	.0000	.0000	.0000	1300
6.4404+004	-1.7429-002	2.1295-002	-8.2368-006	7.1564-005	.0000	.0000	.0000	.0000	1320
6.5381+004	8.6790-003	-5.1521-003	-7.5233-006	6.6988-005	.0000	.0000	.0000	.0000	1340
6.6357+004	1.3087-002	1.8438-003	-6.8863-006	6.2812-005	.0000	.0000	.0000	.0000	1360
6.7334+004	-1.6716-002	-5.1957-002	-6.3161-006	5.8963-005	.0000	.0000	.0000	.0000	1380
6.8311+004	6.0287-003	5.1292-002	-5.8044-006	5.5437-005	.0000	.0000	.0000	.0000	1400
6.9287+004	4.2270-002	-9.4192-003	-5.3442-006	5.2232-005	.0000	.0000	.0000	.0000	1420
7.0264+004	-3.5240-002	-5.8986-004	-4.9292-006	4.9249-005	.0000	.0000	.0000	.0000	1440
7.1240+004	6.4004-004	1.0339-003	-4.5543-006	4.6491-005	.0000	.0000	.0000	.0000	1460
7.2217+004	-7.1060-003	2.9978-002	-4.2148-006	4.3946-005	.0000	.0000	.0000	.0000	1480
7.3193+004	1.9034-002	-2.9462-002	-3.9068-006	4.1586-005	.0000	.0000	.0000	.0000	1500
7.4170+004	-4.7016-002	-1.9131-002	-3.6263-006	3.9396-005	.0000	0000	.0000	.0000	1520
7.5146+004	3.8431-002	2.5974-002	-3.3718-006	3.7360-005	.0000	.0000	.0000	.0000	1540
7.6123+004	4.0375-003	-1.5323-002	-3.1391-006	3.5466-005	.0000	.0000	.0000	.0000	1560
7.7100+004	2.7456-003	-2.5550-003	-2.9265-006	3.3701-005	.0000	.0000	.0000	.0000	1580
7.8076+004	-1.8197-002	4.1258-003	-2.7318-006	3.2054-005	.0000	.0000	.0000	.0000	1600
7.9053+004	2.6735-002	3.8476-002	-2.5533-006	3.0516-005	.0000	.0000	.0000	.0000	1620
8.0029+004	-1.5923-002	-4.1878-002	-2.3893-006	2.9077-005	.0000	.0000	.0000	.0000	1640
8.1006+004	-2.4580-002	1.5559-002	-2.2385-006	2.7729-005	.0000	.0000	.0000	.0000	1660
8.1982+004	1.0155-002	-4.4686-003	-2.0996-006	2.6465-005	.0000	.0000	.0000	.0000	1680
8.2959+004	3.8164-003	2.6532-003	-1.9715-006	2.5279-005	.0000	.0000	.0000	.0000	1700
8.3935+004	-1.3368-003	-3.2551-002	-1.8531-006	2.4165-005	.0000	.0000	.0000	.0000	1720
8.4912+004	-1.0314-002	3.3365-002	-1.7436-006	2.3116-005	.0000	.0000	.0000	.0000	1740
8.5889+004	3.8336-002	2.1188-003	-1.6423-006	2.2125-005	.0000	.0000	.0000	.0000	1760
8.6865+004	-3.1995-002	-1.1647-002	-1.5493-006	2.1199-005	.0000	.0000	.0000	.0000	1780
8.7842+004	1.1678-003	7.9668-003	-1.4610-006	2.0322-005	.0000	.0000	.0000	.0000	1800
8.8818+004	-3.1054-003	1.0668-002	-1.3739-006	1.9493-005	.0000	.0000	.0000	.0000	1820
8.9795+004	1.8765-002	-1.1362-002	-1.3044-006	1.8710-005	.0000	.0000	.0000	.0000	1840

9.0771+004	-3.1664-002	-2.4627-002	-1.2341-006	1.7970-005	.0000	.0000	.0000	.0000	1860
9.1748+004	2.1993-002	3.1055-002	-1.1656-006	1.7259-005	.0000	.0000	.0000	.0000	1880
9.2725+004	1.1482-002	-1.6579-002	-1.1074-006	1.6605-005	.0000	.0000	.0000	.0000	1900
9.3701+004	-7.2497-003	3.8052-003	-1.0503-006	1.5975-005	.0000	.0000	.0000	.0000	1920
9.4678+004	-9.0778-003	-2.2057-003	-9.9660-007	1.5378-005	.0000	.0000	.0000	.0000	1940
9.5654+004	1.0333-002	3.0690-002	-9.4676-007	1.4810-005	.0000	.0000	.0000	.0000	1960
9.6631+004	9.2771-004	-3.3426-002	-8.9987-007	1.4271-005	.0000	.0000	.0000	.0000	1980
9.7607+004	-2.8028-003	8.3037-003	-8.5569-007	1.3759-005	.0000	.0000	.0000	.0000	2000
9.8584+004	2.3586-002	2.2911-003	-8.1461-007	1.3271-005	.0000	.0000	.0000	.0000	2020
9.9561+004	-1.1057-003	-1.8842-003	-7.7583-007	1.2807-005	.0000	.0000	.0000	.0000	2040

TAU(SEC)	ENVELOP DB/UV/M-KW								
-.10000-002	-.17204+002	-.45000-003	-.18110+002	.12000-003	.52544+002	.69000-003	.45899+002	.12600-002	.50833+002
-.99000-003	-.16633+002	-.44000-003	-.15727+002	.13000-003	.52787+002	.70000-003	.45231+002	.12700-002	.51179+002
-.98000-003	-.16237+002	-.43000-003	-.13435+002	.14000-003	.53013+002	.71000-003	.44641+002	.12800-002	.51506+002
-.97000-003	-.15927+002	-.42000-003	-.11782+002	.15000-003	.53223+002	.72000-003	.43976+002	.12900-002	.51812+002
-.96000-003	-.15698+002	-.41000-003	-.10851+002	.16000-003	.53416+002	.73000-003	.43297+002	.13000-002	.52101+002
-.95000-003	-.15607+002	-.40000-003	-.10556+002	.17000-003	.53598+002	.74000-003	.42574+002	.13100-002	.52371+002
-.94000-003	-.15728+002	-.39000-003	-.10736+002	.18000-003	.53764+002	.75000-003	.41837+002	.13200-002	.52624+002
-.93000-003	-.16097+002	-.38000-003	-.11186+002	.19000-003	.53918+002	.76000-003	.41075+002	.13300-002	.52861+002
-.92000-003	-.16664+002	-.37000-003	-.11830+002	.20000-003	.54059+002	.77000-003	.40291+002	.13400-002	.53081+002
-.91000-003	-.17308+002	-.36000-003	-.12791+002	.21000-003	.54190+002	.78000-003	.39487+002	.13500-002	.53286+002
-.90000-003	-.17962+002	-.35000-003	-.14529+002	.22000-003	.54306+002	.79000-003	.38665+002	.13600-002	.53476+002
-.89000-003	-.18713+002	-.34000-003	-.17783+002	.23000-003	.54414+002	.80000-003	.37830+002	.13700-002	.53652+002
-.88000-003	-.19737+002	-.33000-003	-.16081+002	.24000-003	.54512+002	.81000-003	.36986+002	.13800-002	.53815+002
-.87000-003	-.21148+002	-.32000-003	-.11411+002	.25000-003	.54601+002	.82000-003	.36142+002	.13900-002	.53964+002
-.86000-003	-.22752+002	-.31000-003	-.64062+001	.26000-003	.54682+002	.83000-003	.35301+002	.14000-002	.54102+002
-.85000-003	-.23538+002	-.30000-003	-.32107+001	.27000-003	.54756+002	.84000-003	.34471+002	.14100-002	.54229+002
-.84000-003	-.22329+002	-.29000-003	-.15501+001	.28000-003	.54821+002	.85000-003	.33665+002	.14200-002	.54341+002
-.83000-003	-.20100+002	-.28000-003	-.11231+001	.29000-003	.54884+002	.86000-003	.32894+002	.14300-002	.54444+002
-.82000-003	-.18131+002	-.27000-003	-.27826+001	.30000-003	.54939+002	.87000-003	.32168+002	.14400-002	.54545+002
-.81000-003	-.16741+002	-.26000-003	-.92767+001	.31000-003	.54988+002	.88000-003	.31497+002	.14500-002	.54632+002
-.80000-003	-.15854+002	-.25000-003	-.71419+001	.32000-003	.55030+002	.89000-003	.30891+002	.14600-002	.54711+002
-.79000-003	-.15296+002	-.24000-003	.35605+001	.33000-003	.55066+002	.90000-003	.30281+002	.14700-002	.54782+002
-.78000-003	-.14907+002	-.23000-003	.97233+001	.34000-003	.55092+002	.91000-003	.29739+002	.14800-002	.54847+002
-.77000-003	-.14600+002	-.22000-003	.14352+002	.35000-003	.55111+002	.92000-003	.29211+002	.14900-002	.54907+002
-.76000-003	-.14379+002	-.21000-003	.18195+002	.36000-003	.55120+002	.93000-003	.28732+002	.15000-002	.54960+002
-.75000-003	-.14306+002	-.20000-003	.21537+002	.37000-003	.55120+002	.94000-003	.28316+002	.15100-002	.55007+002
-.74000-003	-.14467+002	-.19000-003	.24503+002	.38000-003	.55105+002	.95000-003	.27976+002	.15200-002	.55047+002
-.73000-003	-.14897+002	-.18000-003	.27154+002	.39000-003	.550[illegible]3+002	.96000-003	.27759+002	.15300-002	.55080+002
-.72000-003	-.15529+002	-.17000-003	.29529+002	.40000-003	.55042+002	.97000-003	.27[illegible]0[illegible]+002	.15400-002	.55106+002
-.71000-003	-.16210+002	-.16000-003	.31655+002	.41000-003	.549[illegible]3+002	.98000-003	.278[illegible]+002	.15500-002	.55123+002
-.70000-003	-.16849+002	-.15000-003	.33552+002	.42000-003	.54905+002	.99000-003	.28331+002	.15600-002	.55132+002
-.69000-003	-.17564+002	-.14000-003	.35264+002	.43000-003	.54800+002	.10000-002	.29096+002	.15700-002	.55130+002
-.68000-003	-.18583+002	-.13000-003	.36798+002	.44000-003	.54686+002	.10100-002	.30174+002	.15800-002	.55117+002
-.67000-003	-.20036+002	-.12000-003	.38181+002	.45000-003	.54540+002	.10200-002	.31431+002	.15900-002	.55092+002
-.66000-003	-.21643+002	-.11000-003	.39434+002	.46000-003	.54389+002	.10300-002	.32827+002	.16000-002	.55048+002
-.65000-003	-.22129+002	-.10000-003	.40571+002	.47000-003	.54204+002	.10400-002	.34253+002	.16100-002	.549[illegible]+002
-.64000-003	-.20466+002	-.90000-004	.41611+002	.48000-003	.54007+002	.10500-002	.35640+002	.16200-002	.5490[illegible]+002
-.63000-003	-.18056+002	-.80000-004	.42559+002	.49000-003	.53792+002	.10600-002	.36978+002	.16300-002	.53[illegible]10+002
-.62000-003	-.16073+002	-.70000-004	.43429+002	.50000-003	.53557+002	.10700-002	.38322+002	.16400-002	.54[illegible]8[illegible]+002
-.61000-003	-.14727+002	-.60000-004	.44228+002	.51000-003	.53311+002	.10800-002	.39381+002	.16500-002	.54548+002
-.60000-003	-.13903+002	-.50000-004	.44967+002	.52000-003	.53[illegible]+002	.10900-002	.40453+002	.16600-002	.54387+002
-.59000-003	-.13406+002	-.40000-004	.45654+002	.53000-003	.527[illegible]+002	.11000-002	.41449+002	.16700-002	.542[illegible]7+002
-.58000-003	-.13068+002	-.30000-004	.46298+002	.54000-003	.52481+002	.11100-002	.42363+002	.16800-002	.54006+002
-.57000-003	-.12807+002	-.20000-004	.46904+002	.55000-003	.52176+002	.11200-002	.43221+002	.16900-002	.537[illegible]+002
-.56000-003	-.12642+002	-.10000-004	.47477+002	.56000-003	.51857+002	.11300-002	.44007+002	.17000-002	.53560+002
-.55000-003	-.12656+002	.00000	.48019+002	.57000-003	.51522+002	.11400-002	.44741+002	.17100-002	.53312+002
-.54000-003	-.12962+002	.10000-004	.48532+002	.58000-003	.51170+002	.11500-002	.45421+002	.17200-002	.53049+002
-.53000-003	-.13617+002	.20000-004	.49018+002	.59000-003	.50798+002	.11600-002	.46053+002	.17300-002	.52772+002
-.52000-003	-.14524+002	.30000-004	.49476+002	.60000-003	.50405+002	.11700-002	.46657+002	.17400-002	.52481+002
-.51000-003	-.15401+002	.40000-004	.49907+002	.61000-003	.49992+002	.11800-002	.47221+002	.17500-002	.52177+002
-.50000-003	-.16026+002	.50000-004	.50313+002	.62000-003	.49557+002	.11900-002	.47763+002	.17600-002	.51859+002
-.49000-003	-.16580+002	.60000-004	.50695+002	.63000-003	.49100+002	.12000-002	.48275+002	.17700-002	.51521+002
-.48000-003	-.17421+002	.70000-004	.51054+002	.64000-003	.48622+002	.12100-002	.48763+002	.17800-002	.51172+002
-.47000-003	-.18573+002	.80000-004	.51391+002	.65000-003	.48123+002	.12200-002	.49225+002	.17900-002	.50806+002
-.46000-003	-.19232+002	.90000-004	.51708+002	.66000-003	.47601+002	.12300-002	.49662+002	.18000-002	.50409+002
		.10000-003	.52005+002	.67000-003	.47057+002	.12400-002	.50075+002	.18100-002	.49996+002
		.11000-003	.52284+002	.68000-003	.46490+002	.12500-002	.50465+002	.18200-002	.49561+002

TAU(SEC)	ENVELOP DB/UV/M-KW				
.18200-002	.49105+002	.24000-002	.19059+002	.29700-002	-.17907+002
.18400-002	.48627+002	.24100-002	.19597+002	.29800-002	-.16873+002
.18500-002	.48123+002	.24200-002	.18124+002	.29900-002	-.15979+002
.18600-002	.47667+002	.24300-002	.17640+002	.30000-002	-.15265+002
.18700-002	.4[illegible]+002	.24400-002	.17145+002		
.18600-002	.4[illegible]+002	.24500-002	.16640+002		
.18900-002	.45506+002	.24600-002	.1613[illegible]+002		
.19000-002	.4[illegible]+002	.24700-002	.15631+002		
.19100-002	.44651+002	.24800-002	.15125+002		
.19200-002	.4338[illegible]	.24900-002	.14616+002		
.19300-002	.4329[illegible]	.25000-002	.14100+002		
.19400-002	.42385+002	.25100-002	.13572+002		
.19500-002	.41[illegible]+002	.25200-002	.13031+002		
.19600-002	.41[illegible]+002	.25300-002	.12480+002		
.19700-002	.[illegible]	.25400-002	.11925+002		
.19800-002	.3[illegible]+002	.25500-002	.11371+002		
.19900-002	.3[illegible]	.25600-002	.10822+002		
.20000-002	.3[illegible]	.25700-002	.10278+002		
.20100-002	.3[illegible]	.25800-002	.97342+001		
.20200-002	.[illegible]	.25900-002	.91652+001		
.20300-002	.[illegible]337+002	.26000-002	.8628[illegible]+001		
.20400-002	.3[illegible]	.26100-002	.80675+001		
.205[illegible]0-002	.[illegible]	.26200-002	.75063+001		
.20600-002	.32213+002	.26300-002	.69437+001		
.20700-002	.3[illegible]+002	.26400-002	.63[illegible]+001		
.20800-002	.314[illegible]+002	.26500-002	.58[illegible]		
.20900-002	.30[illegible]2[illegible]	.26600-002	.526[illegible]+001		
.21000-002	.30227+002	.26700-002	.46531+001		
.21100-002	.2[illegible]+002	.26800-002	.40875+001		
.21200-002	.2[illegible]05+002	.26900-002	.33275+001		
.21300-002	.28771+002	.27000-002	.26250+001		
.21400-002	.2[illegible]+002	.27100-002	.19127+001		
.21500-002	.2801[illegible]+002	.27200-002	.11958+001		
.21600-002	.27[illegible]	.27300-002	.48547+000		
.21700-002	.27[illegible]002	.27400-002	-.29977+000		
.21800-002	.2[illegible]+002	.27500-002	-.1127[illegible]+001		
.21900-002	.267[illegible]+002	.27600-002	-.20408+001		
.22000-002	.26[illegible]+002	.27700-002	-.30439+001		
.22100-002	.26[illegible]+002	.27800-002	-.41405+001		
.22200-002	.2[illegible]+002	.27900-002	-.53222+001		
.22300-002	.25[illegible]+002	.28000-002	-.655[illegible]+001		
.22400-002	.2524[illegible]+002	.28100-002	-.783[illegible]+001		
.22500-002	.2[illegible]+002	.28200-002	-.91627+001		
.22600-002	.2[illegible]+002	.28300-002	-.106[illegible]+002		
.22700-002	.2[illegible]+002	.28400-002	-.12[illegible]+002		
.22800-002	.2[illegible]+002	.28500-002	-.1392[illegible]+002		
.22900-002	.2[illegible]+002	.28600-002	-.15[illegible]+002		
.23000-002	.2[illegible]+002	.28700-002	.17407+002		
.23100-002	.228[illegible]+002	.28800-002	-.1820[illegible]+002		
.23200-002	.224[illegible]+002	.28900-002	-.1970[illegible]+002		
.23300-002	.2205[illegible]+002	.29000-002	-.2011[illegible]+002		
.23400-002	.2[illegible]+002	.29100-002	-.202[illegible]+002		
.23500-002	.21242+002	.29200-002	-.20434+002		
.23600-002	.20822+002	.29300-002	-.20304+002		
.23700-002	.20[illegible]2+002	.29400-002	-.202[illegible]+002		
.23800-002	.[illegible]+002	.29500-002	-.19827+002		
.23900-002	.1[illegible]511+002	.29600-002	-.18957+002		

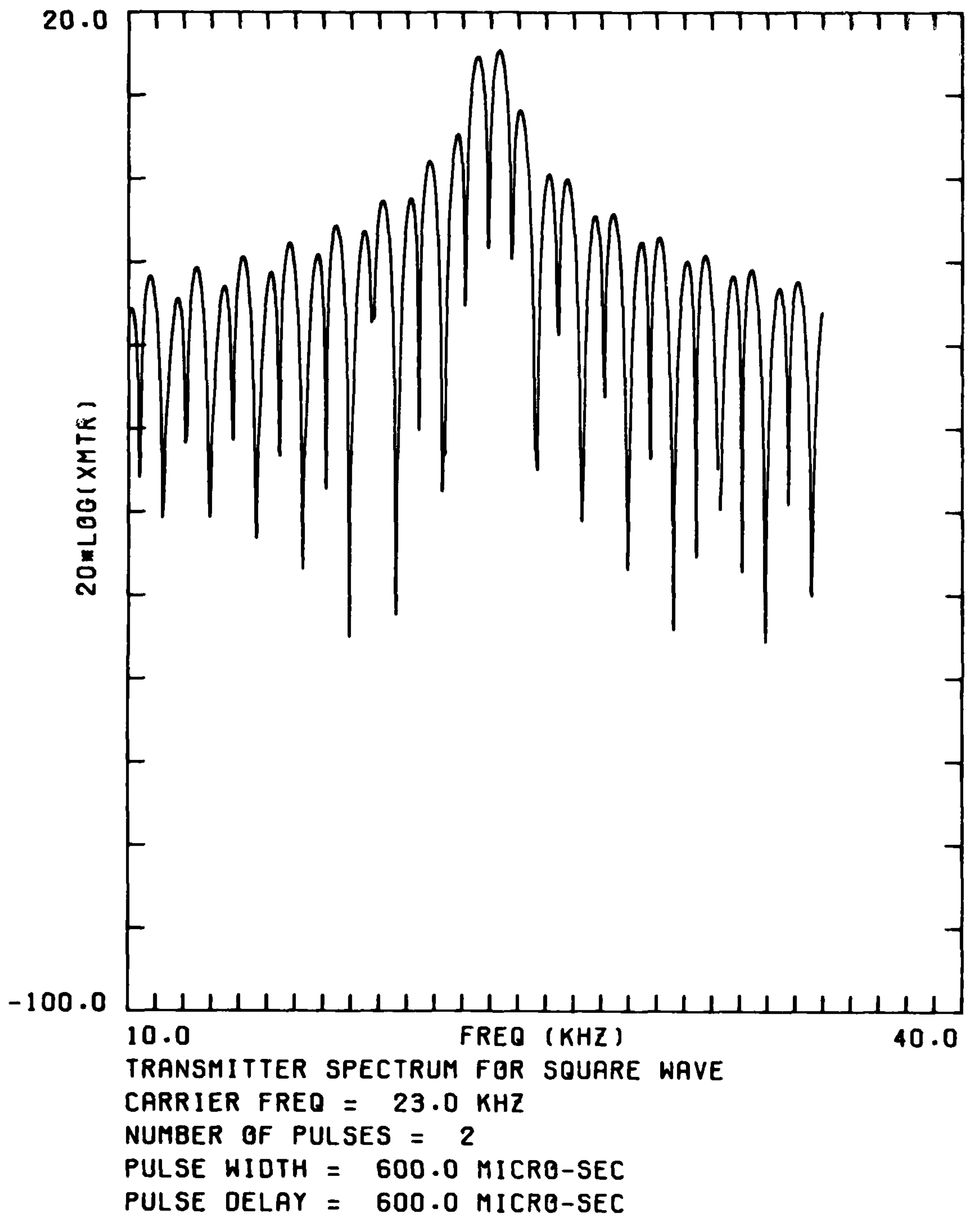


Figure 1. Transmitter spectrum versus frequency.

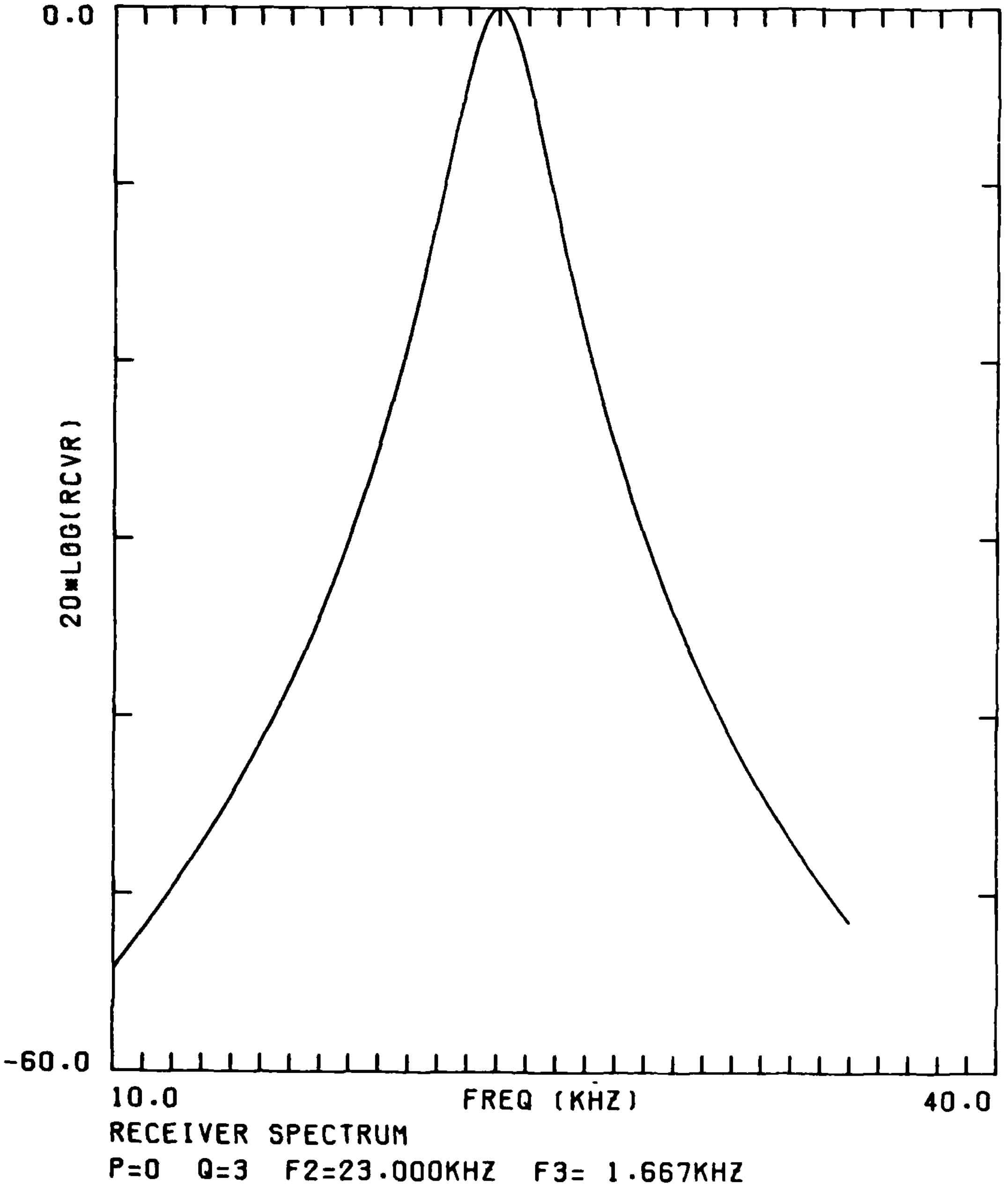


Figure 2. Receiver spectrum versus frequency.

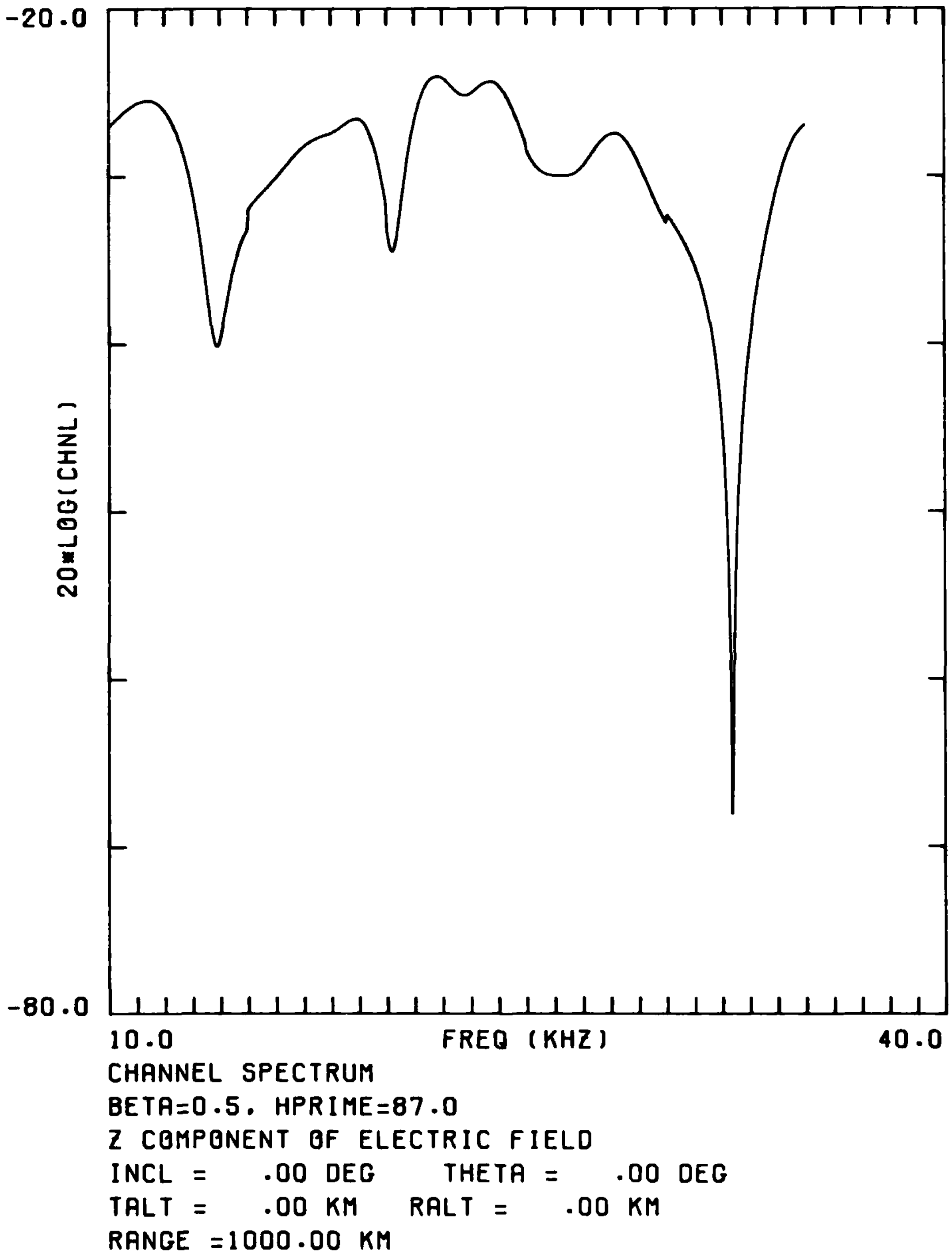


Figure 3. Channel spectrum versus frequency.

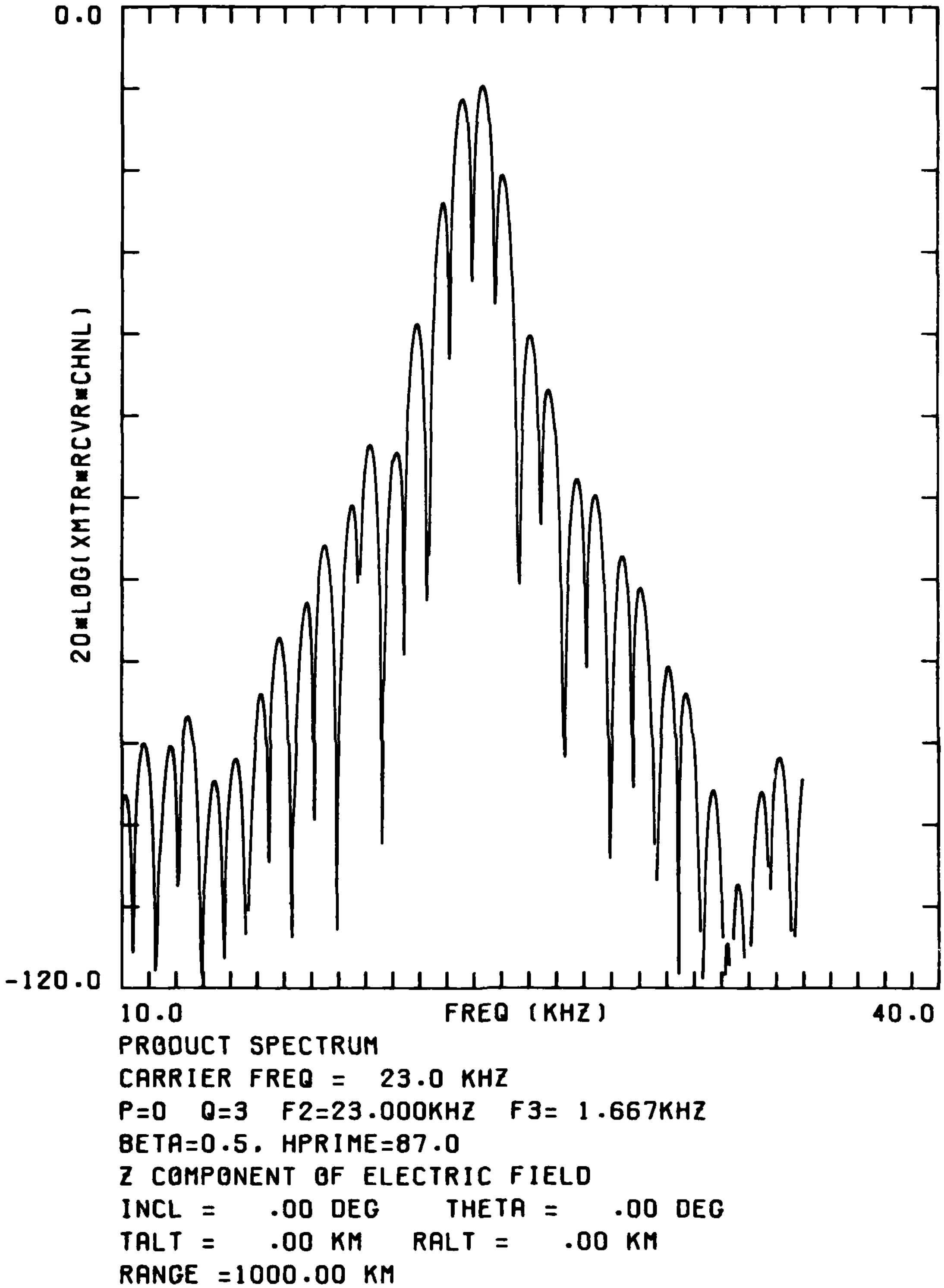
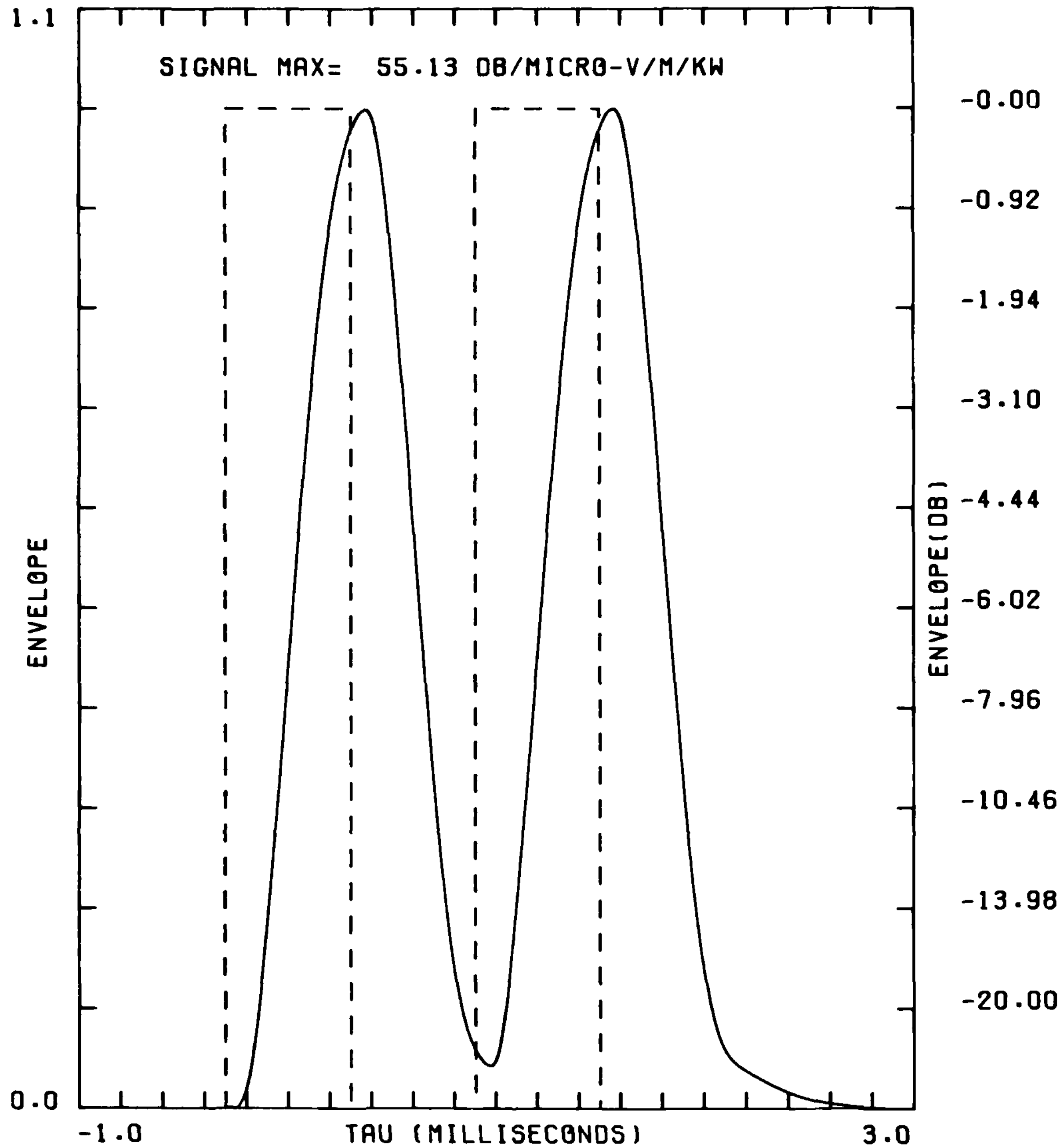
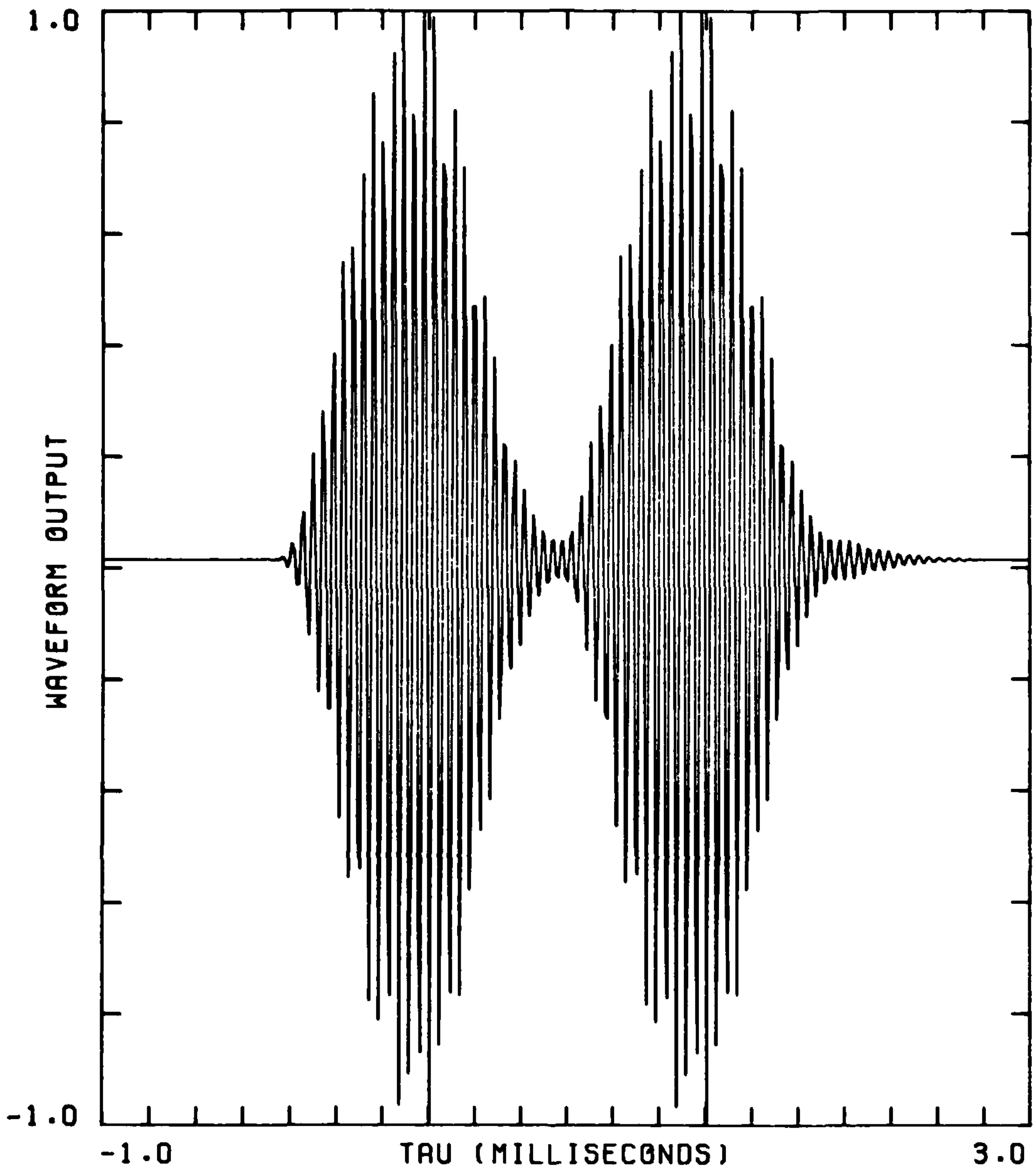


Figure 4. Product spectrum versus frequency.



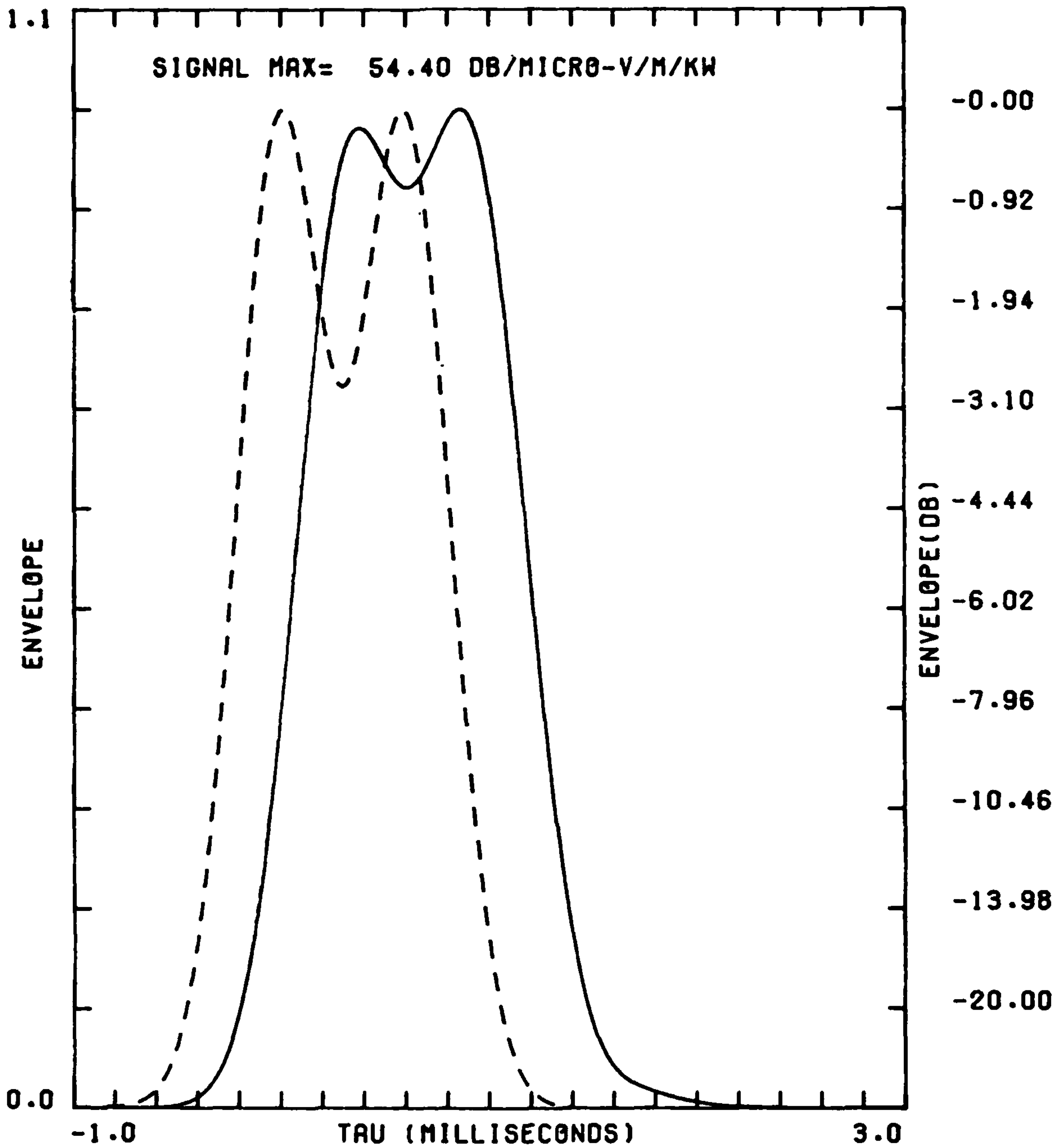
SQUARE WAVE
CARRIER FREQ = 23.0 KHZ
P=0 Q=3 F2=23.000KHZ F3= 1.667KHZ
BETA=0.5, HPRIME=87.0
Z COMPONENT OF ELECTRIC FIELD
INCL = .00 DEG THETA = .00 DEG
TALT = .00 KM RALT = .00 KM
RANGE =1000.00 KM
NUMBER OF PULSES = 2
PULSE WIDTH = 600.0 MICRO-SEC
PULSE DELAY = 600.0 MICRO-SEC

Figure 5. Input and output waveforms normalized to unity.



SQUARE WAVE
CARRIER FREQ = 23.0 KHZ
P=0 Q=3 F2=23.000KHZ F3= 1.667KHZ
BETA=0.5, HPRIME=87.0
Z COMPONENT OF ELECTRIC FIELD
INCL = .00 DEG THETA = .00 DEG
TALT = .00 KM RALT = .00 KM
RANGE =1000.00 KM
NUMBER OF PULSES = 2
PULSE WIDTH = 600.0 MICRO-SEC
PULSE DELAY = 600.0 MICRO-SEC

Figure 6. Waveform output, including carrier frequency.



GAUSSIAN
CARRIER FREQ = 23.0 KHZ
P=0 Q=3 F2=23.000KHZ F3= 1.667KHZ
BETA=0.5, HPRIME=87.0
Z COMPONENT OF ELECTRIC FIELD
INCL = .00 DEG THETA = .00 DEG
TALT = .00 KM RALT = .00 KM
RANGE =1000.00 KM
NUMBER OF PULSES = 2
1/E HALF WIDTH = 300.0 MICRO-SEC
PULSE DELAY = 600.0 MICRO-SEC

Figure 7. Envelope output for IFLGTR=2.

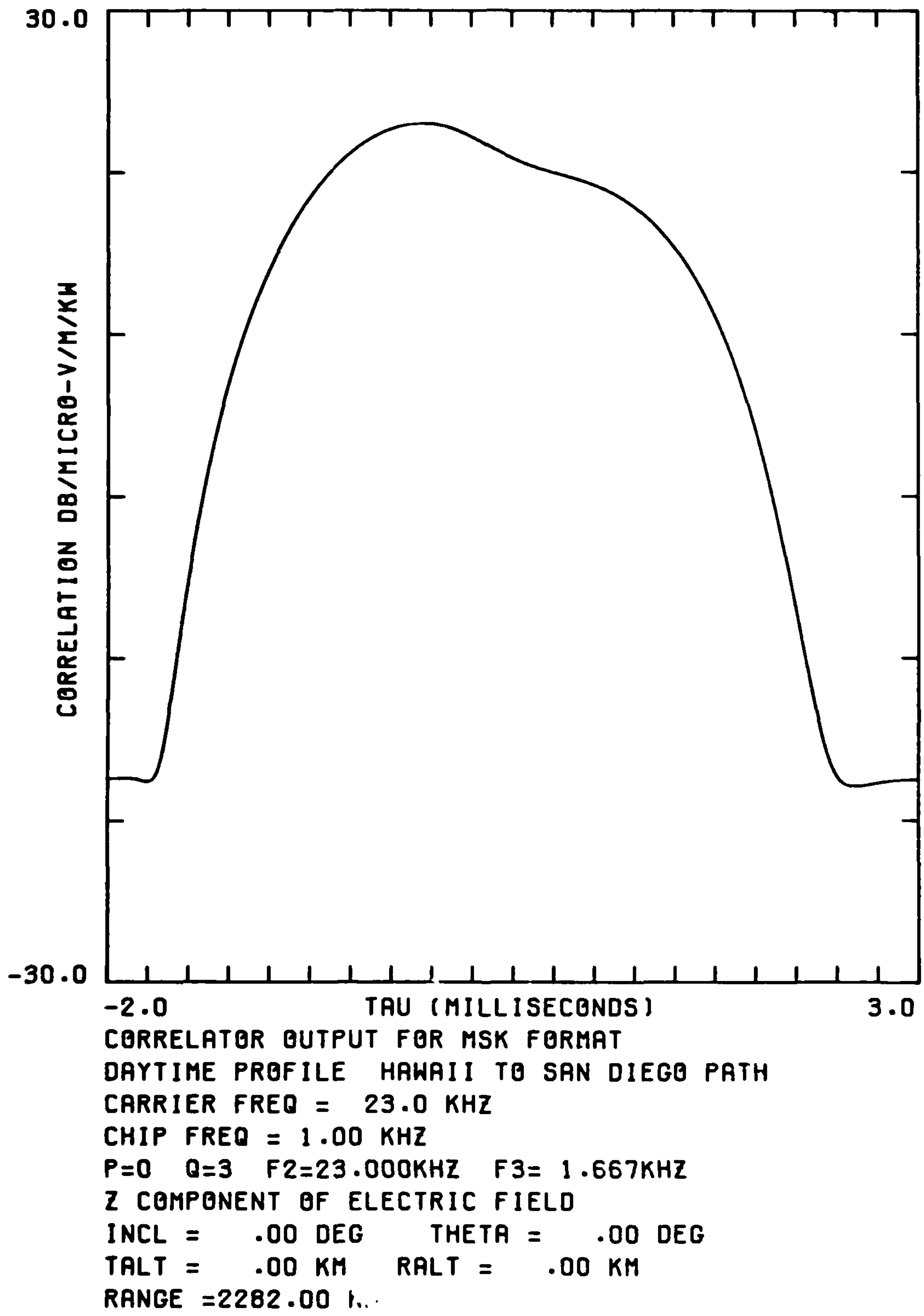
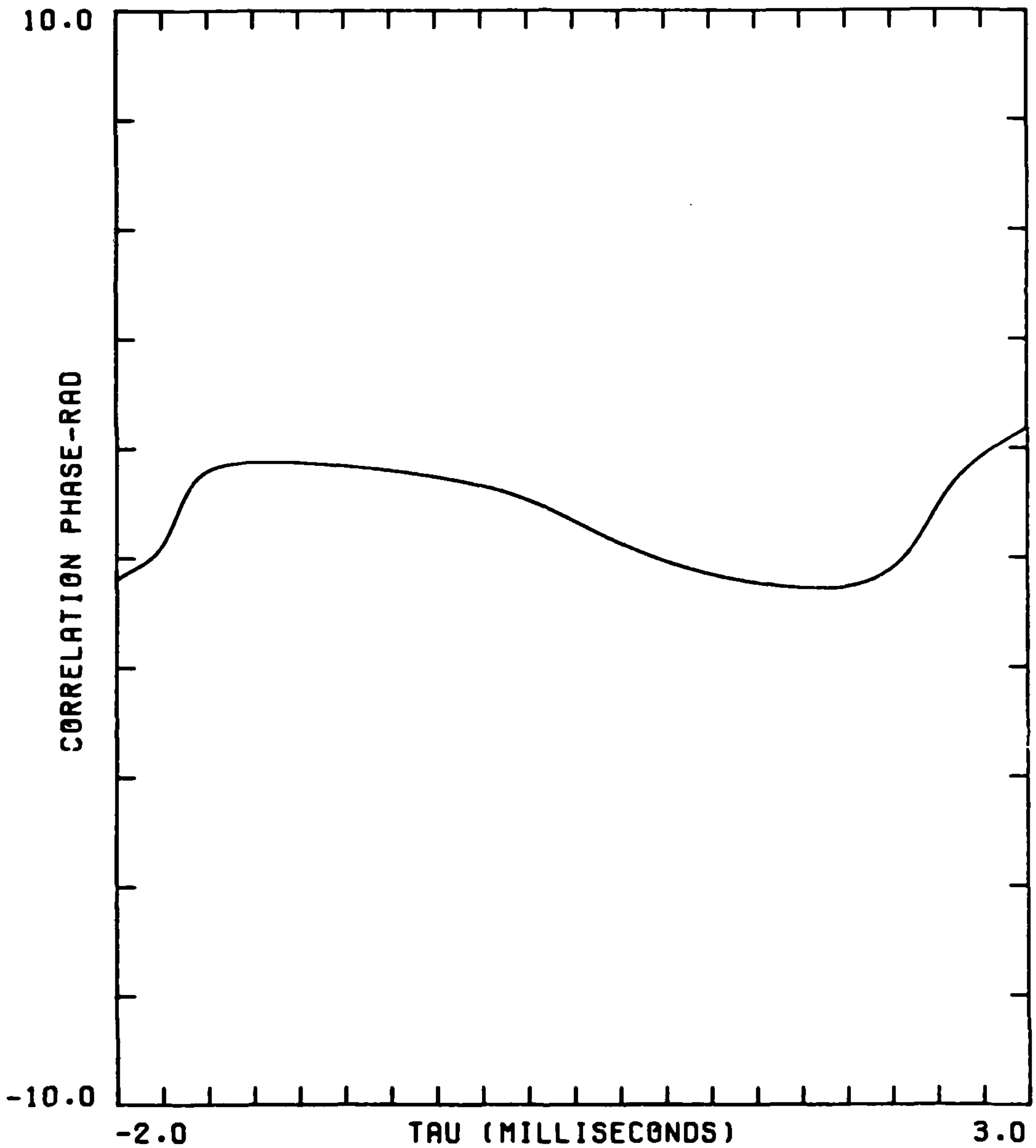


Figure 8. Envelope output for IFLGTR=3.



CORRELATOR PHASE
DAYTIME PROFILE HAWAII TO SAN DIEGO PATH
CARRIER FREQ = 23.0 KHZ
CHIP FREQ = 1.00 KHZ
P=0 Q=3 F2=23.000KHZ F3= 1.667KHZ
Z COMPONENT OF ELECTRIC FIELD
INCL = .00 DEG THETA = .00 DEG
TALT = .00 KM RALT = .00 KM
RANGE =2282.00 KM

Figure 9. Correlator phase output for IFLGTR=3.

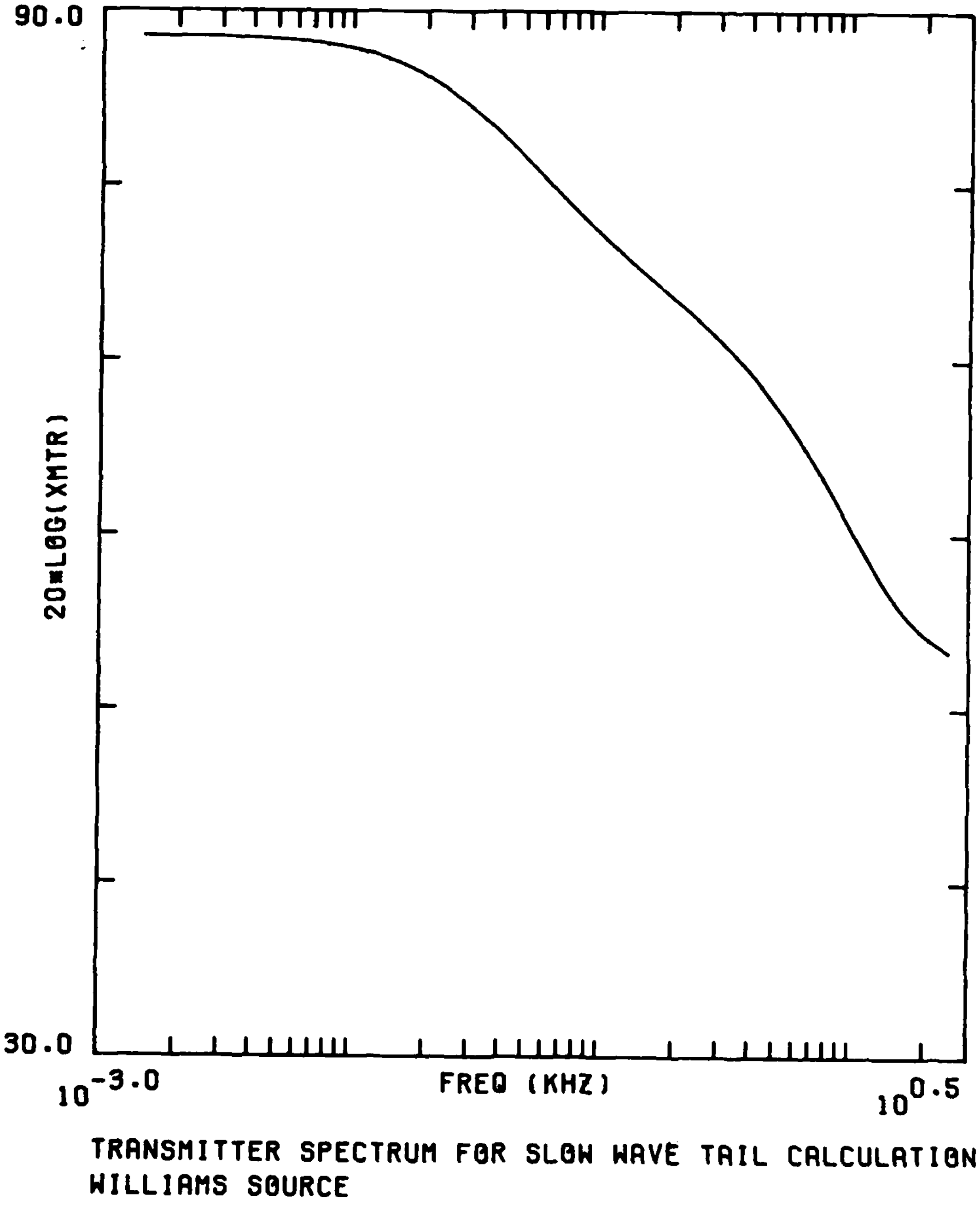


Figure 10. Transmitter spectrum for slow-wave-tail calculation Williams' source.

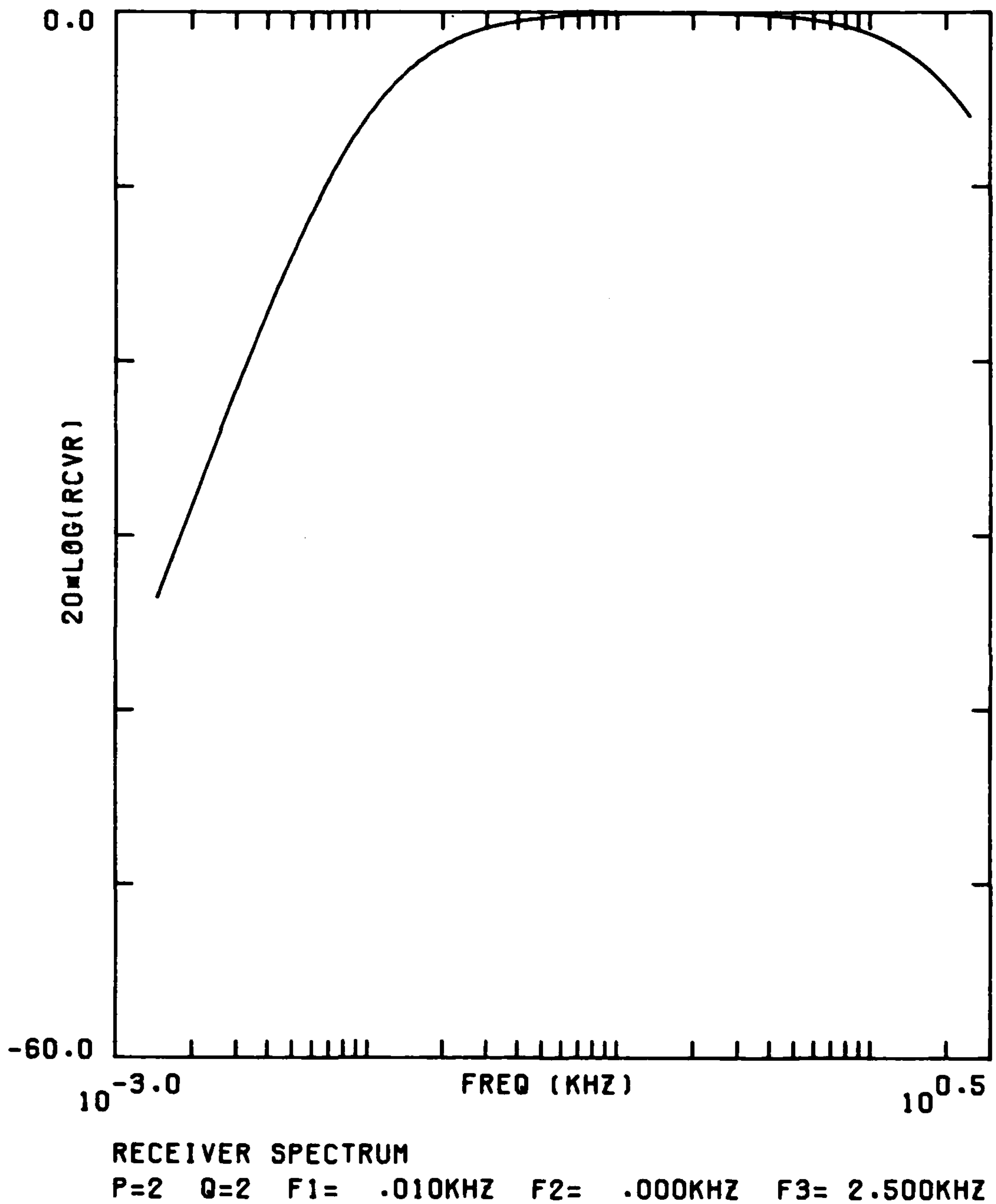


Figure 11. Receiver spectrum used for slow-tail measurements.

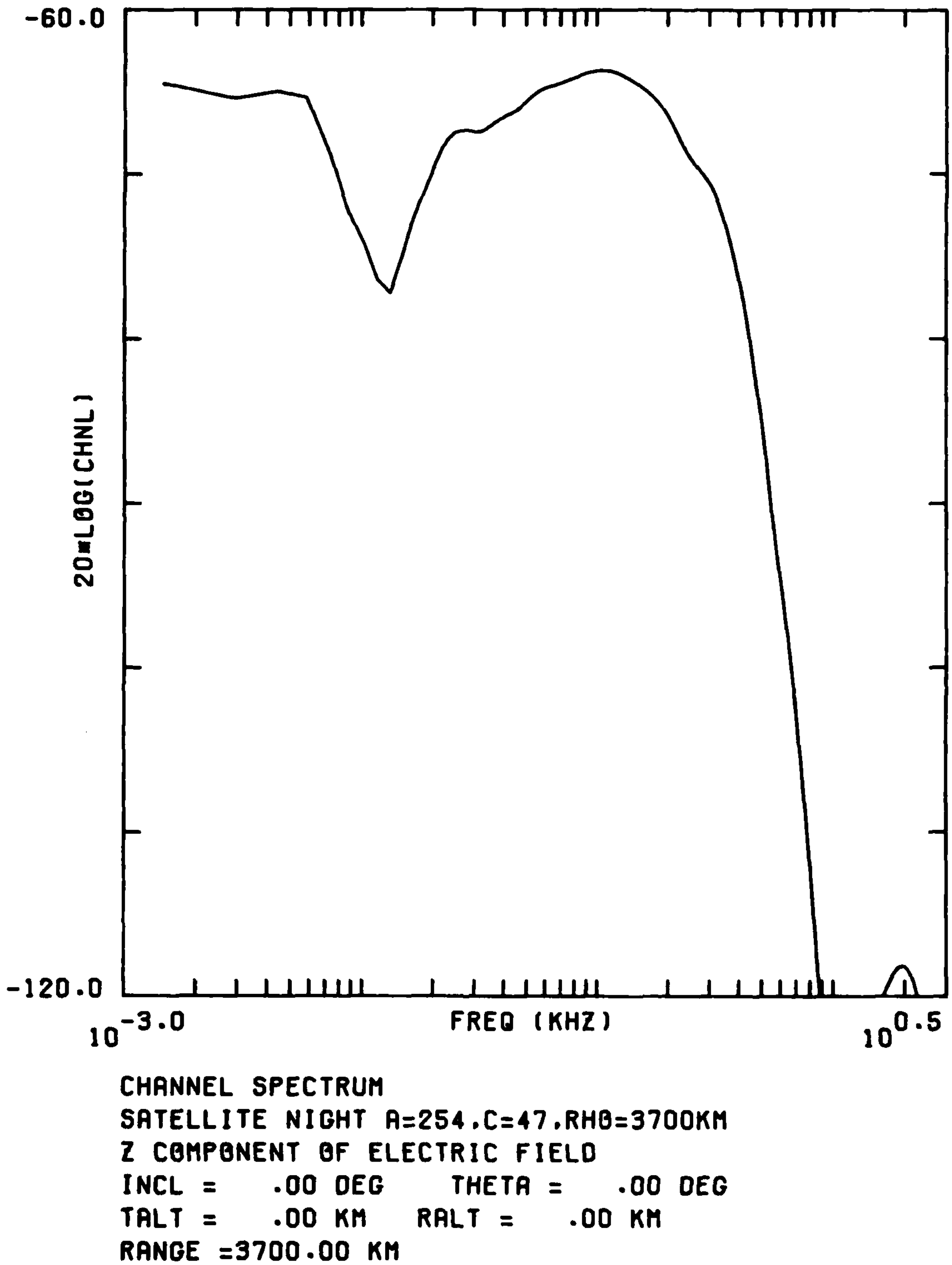


Figure 12. Channel spectrum.

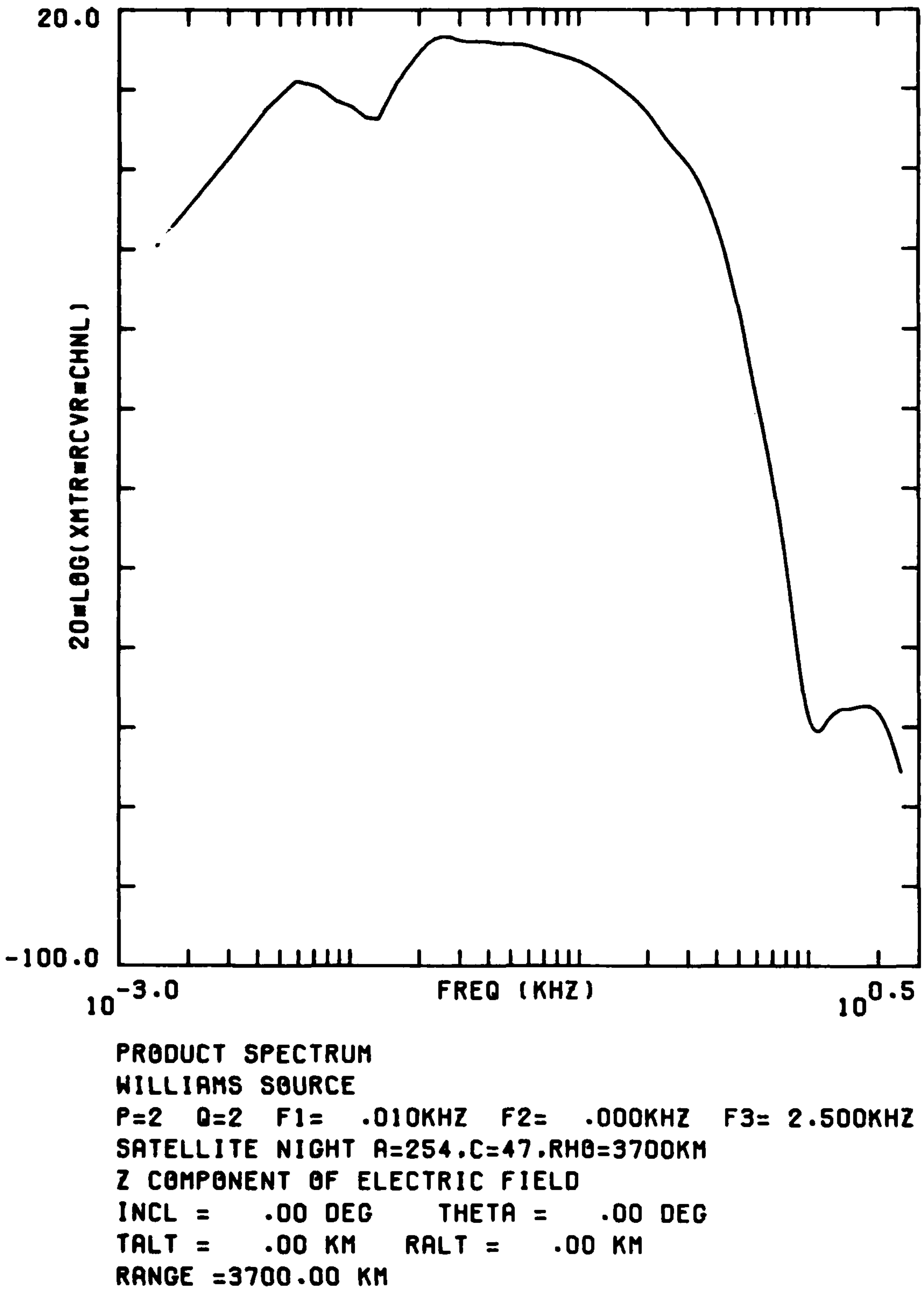


Figure 13. Product spectrum.

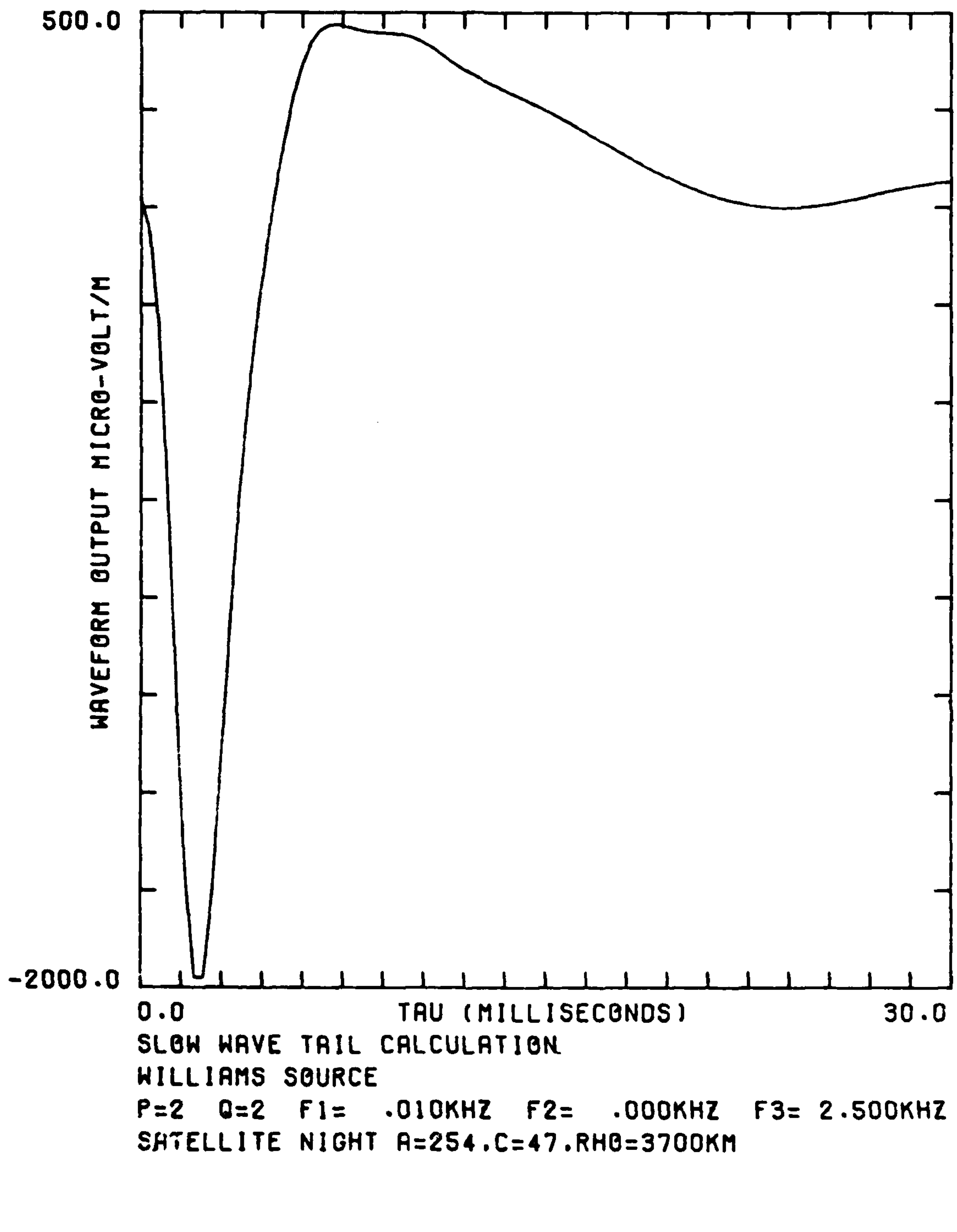


Figure 14. Slow-wave-tail output for the Williams' source.

REFERENCES

1. NOSC TR 444, A Computer Program for ELF/VLF Pulse propagation in a Laterally Homogeneous Earth-Ionosphere Waveguide, by R.A. Pappert and L.R. Shockey (prepared for Defense Nuclear Agency), August, 1979.

2. NELC Interim Report 77T, "MODESRCH," An Improved Computer Program for Obtaining ELF/VLF Mode Constants in an Earth-Ionosphere Waveguide, interim report 77T, by D.G. Morfitt and C.H. Shellman (prepared for Defense Nuclear Agency), October, 1976.

3. Seyler, Jr., C.E., S.C. Block, and R.W. Flynn, Pulse Propagation in a Magnetoplasma, 1. Longitudinal Propagation, J. Geophys Rev 77(22), p 4237-4241, 1972.

4. Scheif, F., Schaum's Outline Series of Theory and Problems of Numercial Analysis, McGraw-Hill, 1968.

5. NELC TR 1834, Effects of the VLF Propagation Channel on Spread-Spectrum Communication Systems, by I.J. Rothmuller, August, 1972.

6. NRL report 8521, Multipath VLF Propagation Effects on Correlation Receivers, by F.J. Kelley, J.P. Hauser, H.M. Beck, and F.J. Rhoads, September, 1981.

7. Galejs, J., Terrestrial Propagation of Long Electromagnetic Waves, Pergamon Press, 1972.

8. NELC Interim Report 713, WKB Mode Summing Program for VLF/ELF Antennas of Arbitrary Length, Shape, and Elevation, by R.A. Pappert and L.R. Shockey, interim report 713, June, 1971.

9. Computation Laboratory, Cambridge MA, Tables of the Modified Hankel Functions of Order One-Third and Their Derivatives, Harvard University Press, 1945.

10. Cooley, J.W., and J.W. Tukey, An Algorithm for the Machine Calculation of Complex Fourier Series, Math Comput 19, p 297-301, 1965.

APPENDIX-PROGRAM LISTING

```
C
      PARAMETER NMAX=2049
C
      INCLUDE SPECANS.COMMONSPECS,LIST
      COMMON/THREE/TP(NRMODE),RATIO(4,NRMODE),FRQ,KF,NMF
      COMMON/FOUR/NFFT,FREQU,FREQL,INTPRT,TAUMAX,FREQO,PULSEW,
     $      RHOMIN,DELRHO,RHOMAX,TALT,RALT,INCL,THETA,ICOMP,
     $      IFLGTR,INTFLG,NPRNT,TAUO,NUMTAU,CHIPFR,NUMPLS,PULSED,
     $      IPLOT,IPLOT1
      COMMON/FIVE/SIGMA,EPSR,NRPTS,NRPT1,NEWD
      COMMON/SIX/PLOTX(NMAX),PLOTY1(NMAX),PLOTY2(NMAX),NUMPTS
      COMMON/SEVEN/XMIN,XMAX,XTIC,SCALEX,XLNG,YLNG
      COMMON/NINE/X(NMAX),Y(NMAX)
      COMMON/TEN/XL,YL
      COMMON/COM11/PLOTX3(300),PLOTY3(300)
      COMMON/COM12/FREQ1,FREQ2,FREQ3,P,Q
C
      PARAMETER NRSAVE=NRMODE*10+3
      DIMENSION SAVEMC(NRSAVE)
      EQUIVALENCE (TP,SAVEMC)
C
      COMPLEX TMP1,TMP2,TMP3,XTRA,TP,STP,RATIO,
     $        EXC(3,NRMODE),HGT(3,NRMODE),HGR(3,NRMODE),FOFTAU,SUM,SUMP,
     $        IM/(0.0,1.0)/
      REAL INCL,INCLS
C
      DIMENSION XS(NMAX),YS(NMAX)
C
      DATA TWOPI/6.2831 85307 17958 64769 25287/
      DATA   DTR/0.0174 53292 51994 32957 69237/
      DATA ALPHA/3.14E-4/
      DATA XLNG/5./,YLNG/6./
C
C
9     CALL INPUT
C
C..PROCESS MODE DATA
      IF(ICOMP .EQ. ICOMPS .AND. TALT .EQ. TALTS .AND.
     $   RALT .EQ. RALTS .AND. INCL .EQ. INCLS .AND.
     $   THETA .EQ. THETAS .AND. NEWD .EQ. 0) GO TO 70
      INCLS=INCL
      THETAS=THETA
      SIN1=SIN(INCL*DTR)
      SIN2=SIN(THETA*DTR)
      COS1=COS(INCL*DTR)
      COS2=COS(THETA*DTR)
      NM=0
      DO 31 K=1,NRFREQ
      DO 31 M=1,NRMODE
31    MODE(M,K)=0
C
      PRINT 1040
      DO 49 K=1,NF
      READ(2) SAVEMC
```

```
C
      IF(NEWD .EQ. 0 .AND. TALT .EQ. TALTS) GO TO 33
      CALL HTGAIN(1,FRQ,SIGMA,EPSR,ALPHA,NMF,TP,TALT,HGT)
C
33    IF(NEWD .EQ. 0 .AND. RALT .EQ. RALTS) GO TO 34
      CALL HTGAIN(1,FRQ,SIGMA,EPSR,ALPHA,NMF,TP,RALT,HGR)
C
34    IF(ICOMP .EQ. ICOMPS) GO TO 40
      DO 38 M=1,NMF
      STP=CMPLX(STPR(M,K),STPI(M,K))
      IF(ICOMP-2) 35,36,37
35    EXC(1,M)=(STP*STP)*RATIO(1,M)
      EXC(2,M)=-STP*RATIO(3,M)*RATIO(4,M)
      EXC(3,M)=-STP*RATIO(1,M)
      GO TO 38
36    EXC(1,M)=-STP*RATIO(3,M)
      EXC(2,M)=RATIO(2,M)
      EXC(3,M)=RATIO(3,M)
      GO TO 38
37    EXC(1,M)=STP*RATIO(1,M)
      EXC(2,M)=-RATIO(3,M)*RATIO(4,M)
      EXC(3,M)=-RATIO(1,M)
38    CONTINUE
C
40    NM=MAX0(NM,NMF)
C..WN=2*PI*FREQ/C
      WN=20.95845*FRQ
      ACONST=-8.686*WN
C..REFERENCE EXCITATION TO REFLHT=70
      DO 42 M=1,NMF
      ATTEN=ACONST*STPI(M,K)
      VOVERC=1.0/STPR(M,K)
      TMP1 = EXC(1,M)*HGT(1,M)*HGR(ICOMP,M)
      TMP2 = EXC(2,M)*HGT(2,M)*HGR(ICOMP,M)
      TMP3 = EXC(3,M)*HGT(3,M)*HGR(ICOMP,M)
      XTRA=TMP1*COS1+(TMP2*SIN2+TMP3*COS2)*SIN1
C
C..SET UP ARRAYS FOR INTERPOLATION
      MODE(M,K)=M
      XTRAR(M,K)=REAL(XTRA)
      XTRAI(M,K)=AIMAG(XTRA)
C
      IF(M .GT. 1) THEN
      PRINT 1043,         TP(M),ATTEN,VOVERC
      ELSE
      PRINT 1041,NMF,FRQ,TP(M),ATTEN,VOVERC
      ENDIF
42    CONTINUE
C
C..END PROCESSING OF DATA FOR THIS FREQUENCY
49    CONTINUE
      ICOMPS=ICOMP
      TALTS=TALT
      RALTS=RALT
      NEWD=0
      REWIND 2
C
```

```
C..SET UP INTERPOLATION
      DO 56 M=1,NM
      KMODE = 0
      KMODE1 = 1
      DO 53 KF =1,NF
      IF(MODE(M,KF) .NE. 0) KMODE=KMODE+1
53    CONTINUE
      DO 54 KF=1,NF
      IF(MODE(M,KF) .NE. 0) MODE(M,KF)=KMODE
54    CONTINUE
      DO 55 KF=1,NF
      IF(MODE(M,KF) .EQ. 0) KMODE1=KMODE1+1
      KK(M) = KMODE1
      IF(MODE(M,KF) .NE. 0) GO TO 56
55    CONTINUE
56    CONTINUE
C
      DO 65 MD=1,NM
      DO 65 LF=1,4
      CALL FUNSPL (MD,LF)
65    CONTINUE
C
C..BEGIN
70    F0 = FREQ0*1000.0
      FL = FREQL*1000.0
      FU = FREQU*1000.0
      FC = CHIPFR*1000.
      F1 = FREQ1*1000.0
      F2 = FREQ2*1000.0
      F3 = FREQ3*1000.0
      DELTAF = (FU-FL)/NRPTS
      DELTAU = (TAUMAX-TAU0)/(FLOAT(NUMTAU)-1.)
C
      IF(IPLOT .EQ. 0)GO TO 90
      XMIN=AINT(FREQ( 1)/10000.)*10.
      XMAX=AINT(FREQ(NF)/10000.+.99)*10.
      IF(IFLGTR .EQ. 4) XMAX=FREQ3
      XTIC=1.
      SCALEX=(XMAX-XMIN)/XLNG
C
      IF (FREQ0 .EQ. FREQS .AND. PULSEW . EQ. PULSES .AND. CHIPFR .EQ.
     $ CHIPS .AND. PULSED .EQ. PULSDS . AND. NUMPLS .EQ. NMPLS)
     $ GO TO 80
C
C  TRANSMITTER SPECTRUM PLOT
      CALL TPLOT(FREQ,FL,F0,FC,DELTAF,NRPT1,NF)
C
   80 IF (FREQ0 .EQ. FREQS .AND. FREQ3  .EQ. FREQ3S .AND. Q .EQ.
     $  QS) GO TO 90
C
C  RECEIVER SPECTRUM PLOT
      CALL RPLOT(FREQ,FL,F1,F2,F3,DELTAF,P,Q,NRPT1,NF)
C
90    FREQS=FREQ0
      PULSES=PULSEW
      FREQ3S = FREQ3
      CHIPS = CHIPFR
```

```
      PULSDS = PULSED
      NMPLS = NUMPLS
      QS = Q
C
C..LOOP OVER RECEIVER DISTANCES
      RHO=RHOMIN
91    PRINT 1090,RHO
      PRINT 1091
      CALL CPPLOT(FREQ,FL,F0,FC,DELTAF,NRPT1,NF,F1,F2,F3,P,Q,RHO)
C
C..OUTPUT WAVEFORM
      IF(INTFLG .EQ. 0)THEN
      IF(IFLGTR .EQ. 1 .OR. IFLGTR .EQ. 2) PRINT 906
      IF(IFLGTR .EQ. 3) PRINT 905
      IF(IFLGTR .EQ. 4) PRINT 904
      MS = 1
      IF(TAU0 .LT. 0.)THEN
      DO 400 LL =1,NRPTS
      XS(LL) = X(LL)
  400 YS(LL) = Y(LL)
      CALL NLOGN(NFFT,XS,YS,-1.0,FL,FU)
      DO 401 LL = 1,NRPTS
      TAU = -(LL-1)/(FU-FL)
      IF(TAU .LT. TAU0)THEN
      MS = LL-1
      GO TO 402
      END IF
  401 CONTINUE
  402 CONTINUE
      IF(MS .LE. 1)THEN
      MS = 1
      GO TO 407
      END IF
      NUMPTS = 0
      DO 403 LL = 1,MS-1
      INDEX = MS-LL+1
      TAU=-(INDEX-1)/(FU-FL)
      FOFTAU = CMPLX(XS(INDEX),YS(INDEX))*CEXP(IM*TWOPI*TAU*(FL-F0))
      FOFTR = FOFTAU
      TWOFTR = 2.0*FOFTAU*CEXP(IM*TWOPI*TAU*F0)
      FOFTI = -IM*FOFTAU
      ENVLOP = 2.*CABS(FOFTAU)
      IF(ENVLOP .EQ. 0.)THEN
      CORR = -1000.
      PHZF = -1000.
      ELSE
      CORR = 20.*ALOG10(ENVLOP)
      PHZF = ATAN2(FOFTI,FOFTR)
      END IF
      NUMPTS = NUMPTS+1
      PLOTX(NUMPTS) = TAU*1.E3
      IF (IFLGTR .EQ. 3) THEN
      PLOTY1(NUMPTS) = CORR
      NS = 4
      IF(NUMPTS .GT. 1)THEN
      PHZC = ABS(PHZF-PHZF1)
      DO 408 NQ = 1,7
```

```
      IF(ABS(PHZF+(4-NQ)*TWOPI-PHZF1) .LT. PHZC)THEN
      PHZC = ABS(PHZF + (4-NQ)*TWOPI-PHZF1)
      NS = NQ
      END IF
  408 CONTINUE
      PHZF = PHZF + (4-NS)*TWOPI
      PHZF1 = PHZF
      ELSE
      PHZF1 = PHZF
      END IF
      PLOTY2(NUMPTS) = PHZF
      PRINT 907,TAU,CORR,PHZF
      ELSE
      PLOTY1(NUMPTS)=ENVLOP
      PLOTY2(NUMPTS) = TWOFTR
      PRINT 908,TAU,20.0*ALOG10(ENVLOP)
      END IF
  403 CONTINUE
      END IF
  407 CONTINUE
      CALL NLOGN(NFFT,X,Y,1.0,FL,FU)
      DO 404 LL = 1,NRPTS
      TAU = (LL-1)/(FU-FL)
      IF(TAU .GT. TAUMAX)THEN
      MSTOP = LL-1
      GO TO 405
      END IF
  404 CONTINUE
  405 CONTINUE
      NUMPTS = MS-1
      DO 406 LL = 1,MSTOP
      TAU = (LL-1)/(FU-FL)
      FOFTAU = CMPLX(X(LL),Y(LL))*CEXP(IM*TWOPI*TAU*(FL-FO))
      FOFTR = FOFTAU
      TWOFTR = 2.0*FOFTAU*CEXP(IM*TWOPI*TAU*FO)
      FOFTI = -IM*FOFTAU
      ENVLOP = 2.*CABS(FOFTAU)
      CORR = 20.*ALOG10(ENVLOP)
      PHZF = ATAN2(FOFTI,FOFTR)
      NUMPTS = NUMPTS+1
      PLOTX(NUMPTS) = TAU*1.E3
      IF (IFLGTR .EQ. 3) THEN
      PLOTY1(NUMPTS)=CORR
      NS = 4
      IF(NUMPTS .GT. 1)THEN
      PHZC = ABS(PHZF-PHZF1)
      DO 409 NQ = 1,7
      IF(ABS(PHZF+(4-NQ)*TWOPI-PHZF1) .LT. PHZC)THEN
      PHZC = ABS(PHZF + (4-NQ)*TWOPI-PHZF1)
      NS = NQ
      END IF
  409 CONTINUE
      PHZF = PHZF + (4-NS)*TWOPI
      PHZF1 = PHZF
      ELSE
      PHZF1 = PHZF
      END IF
```

```
      PLOTY2(NUMPTS) = PHZF
      PRINT 907,TAU,CORR,PHZF
      ELSE
      PLOTY1(NUMPTS)= ENVLOP
      PLOTY2(NUMPTS) = TWOFTR
      CALL INITWF(IFLGTR,PULSEW,PULSED,NUMPLS,TAU0,TAUMAX,PLOTX3,PLOTY3)
      PRINT 908,TAU,20.0*ALOG10(ENVLOP)
      END IF
  406 CONTINUE
      ELSE
      IF(IFLGTR .EQ. 1 .OR. IFLGTR .EQ. 2) PRINT 910
      IF(IFLGTR .EQ. 3) PRINT 909
      IF(IFLGTR .EQ. 4) PRINT 912
      TAU = TAU0
      NUMPTS = 0
      DO 310 JJ=1,NUMTAU
      CALL FILON(NFFT,X,Y,TAU,FU,FL,SUM,SUMP)
      ERROR = CABS(SUM-SUMP)/CABS(SUM)
      FOFTAU = SUM*CEXP(-IM*TAU*FO*TWOPI)
      FOFTR = FOFTAU
      FOFTI = -IM*FOFTAU
      ENVLOP = 2.*CABS(FOFTAU)
      IF(ENVLOP .EQ. 0.)THEN
      CORR = -1000.
      PHZF = -1000.
      ELSE
      CORR = 20.*ALOG10(ENVLOP)
      PHZF = ATAN2(FOFTI,FOFTR)
      END IF
      NUMPTS = NUMPTS+1
      PLOTX(NUMPTS) = TAU*1.E3
      IF (IFLGTR .EQ. 3) THEN
      PLOTY1(NUMPTS) = CORR
      NS = 4
      IF(NUMPTS .GT. 1)THEN
      PHZC = ABS(PHZF-PHZF1)
      DO 410 NQ = 1,7
      IF(ABS(PHZF+(4-NQ)*TWOPI-PHZF1) .LT. PHZC)THEN
      PHZC = ABS(PHZF + (4-NQ)*TWOPI-PHZF1)
      NS = NQ
      END IF
  410 CONTINUE
      PHZF = PHZF + (4-NS)*TWOPI
      PHZF1 = PHZF
      ELSE
      PHZF1 = PHZF
      END IF
      PLOTY2(NUMPTS) = PHZF
      PRINT 911,TAU,CORR,PHZF,ERROR
      ELSE
      PLOTY1(NUMPTS) = ENVLOP
      CALL INITWF(IFLGTR,PULSEW,PULSED,NUMPLS,TAU0,TAUMAX,PLOTX3,PLOTY3)
      PRINT 907,TAU,20.0*ALOG10(ENVLOP),ERROR
      END IF
      TAU = TAU+DELTAU
  310 CONTINUE
      END IF
```

```
C
      CALL WOPLOT(RHO)
      RHO = RHO + DELRHO
      IF(RHO .LT. RHOMAX+1.E-3) GO TO 91
      GO TO 9
C
C
  904 FORMAT('1',4X,'TAU(SEC)',3X,' OUTPUT',/,17X,'DB/UV/M')
  905 FORMAT('1',4X,'TAU(SEC)',2X,'CORRELATION',4X,'PHASE'/15X,'DB/UV/M
     $-KW',5X,'RAD')
  906 FORMAT('1',4X,'TAU(SEC)',3X,'ENVELOP',/,14X,'DB/UV/M-KW')
  907 FORMAT(' ',2E12.5,3X,E12.5)
  908 FORMAT(' ',2E12.5)
  909 FORMAT('1',4X,'TAU(SEC)',2X,'CORRELATION',4X,'PHASE',4X,'REL ERROR
     $'/15X,'DB/UV/M-KW',6X,'RAD')
  910 FORMAT('1',4X,'TAU(SEC)',3X,'ENVELOP',4X,'REL ERROR',/,15X,
     $  'DB/UV/M-KW')
  911 FORMAT(' ',4E12.5)
  912 FORMAT('1',4X,'TAU(SEC)',3X,' OUTPUT',4X,'REL ERROR',/,17X,
     $  'DB/UV/M')
 1040 FORMAT('1',28X,'NMF',4X,'FREQ',3X,'THETAR',4X,'THETAI',8X,
     $  'ATT',6X,'PHVOC'/36X,'KHZ',4X,'DEGREES',3X,'DEGREES',7X,'DB')
 1041 FORMAT(/28X,I5,5F10.5)
 1043 FORMAT( 41X     4F10.5)
 1090 FORMAT('1RHO = ',F6.0)
 1091 FORMAT(14X,'FREQ(HZ)',4X,'XMTR R',6X,'XMTR I',6X,'RCVR R',6X,
     $        'RCVR I',6X,'CHNL R',6X,'CHNL I',8X,'XMTR*RCVR*CHNL',
     $  10X,'K'/99X,'REAL',8X,'IMAG')
      END
```

```
      SUBROUTINE INPUT
C..THIS ROUTINE READS NAMELIST DATA AND MODE CONSTANT DATA
C
      PARAMETER NMAX=2049
C
      INCLUDE SPECANS.COMMONSPECS,LIST
      COMMON/THREE/TP(NRMODE),RATIO(4,NRMODE),FRO,KF,NMF
      COMMON/FOUR/NFFT,FREQU,FREQL,INTPRT,TAUMAX,FREQ0,PULSEW,
     $      RHOMIN,DELRHO,RHOMAX,TALT,RALT,INCL,THETA,ICOMP,
     $      IFLGTR,INTFLG,NPRNT,TAU0,NUMTAU,CHIPFR,NUMPLS,PULSED,
     $      IPLOT,IPLOT1
      COMMON/FIVE/SIGMA,EPSR,NRPTS,NRPT1,NEWD
      COMMON/EIGHT/LABELT,LABELR,LABELC
      COMMON/COM12/FREQ1,FREQ2,FREQ3,P,Q
C
      PARAMETER NRSAVE=NRMODE*10+3
      DIMENSION SAVEMC(NRSAVE)
      EQUIVALENCE (TP,SAVEMC)
C
      COMPLEX TMP1,TMP2,TMP3,TMP4,TP,STP,RATIO
      REAL INCL
      CHARACTER*50 LABELT,LABELR,LABELC
      CHARACTER*4 IOCNTL(2)
C
      NAMELIST/DATUM/NFFT,FREQU,FREQL,INTPRT,TAUMAX,FREQ0,PULSEW,
     $         RHOMIN,DELRHO,RHOMAX,TALT,RALT,INCL,THETA,ICOMP,IFLGTR,
     $         INTFLG,NPRNT,TAU0,NUMTAU,CHIPFR,NUMPLS,PULSED,
     $         FREQ1,FREQ2,FREQ3,P,Q,IPLOT,IPLOT1
C
      DATA   DTR/0.0174 53292 51994 32957 69237/
      DATA NFFT/11/,             INTPRT/100/,IPLOT/0/,IPLOT1/0/
      DATA FREQU/100.0/,FREQL/0.0/,TAUMAX/.002/
      DATA RHOMIN/1000.0/,RHOMAX/1000.0/,DELRHO/1000.0/
      DATA FREQ0/23./,PULSEW/600./,             IFLGTR/1/,INTFLG/0/,
     $ NPRNT/40/,TAU0/-0.001/,NUMTAU/41/,NUMPLS/1/,
     $ PULSED/600./,CHIPFR/1./
      DATA FREQ1/0.01/,FREQ2/0.0/,FREQ3/2.5/,P/0.0/,Q/2.0/
      DATA INCL,THETA,TALT,RALT/4*0./,ICOMP/1/
      DATA NEWD/-1/
C
C
      PRINT 1002
10    READ(5,1000,END=999) IOCNTL
      PRINT 1001,IOCNTL
      IF(IOCNTL(1) .EQ. 'NAME') GO TO 11
      IF(IOCNTL(1) .EQ. 'DATA') GO TO 20
      IF(IOCNTL(1) .EQ. 'STAR') GO TO 30
      GO TO 994
C
C..READ NAMELIST DATA
11    READ(5,DATUM)
      WRITE(6,DATUM)
C
      NRPTS=2**NFFT
```

```
      NRPT1=NRPTS+1
      IF(NRPT1 .GT. NMAX) GO TO 990
      GO TO 10
C
C..READ INPUT MODE CONSTANTS CARDS
20    READ 1005,LABELC
      PRINT 1006,LABELC
      KF=0
21    READ(5,1021,END=989) FRQ,AZM,CODP,BFLD,SGM,EPS
      IF(FRQ .EQ. 0.) GO TO 29
      IF(KF .EQ. NRFREQ) GO TO 991
      SIGMA=SGM
      EPSR=EPS
      KF=KF+1
      FREQ(KF)=FRQ*1000.
      M=0
23    READ(5,1023,END=989) INDX1,TR1,TI1,TMP1,TMP2
      IF(TR1 .EQ. 0.) GO TO 28
      READ(5,1023,END=989) INDX2,TR2,TI2,TMP3,TMP4
      IF(TR1 .NE. TR2 .OR. TI1 .NE. TI2) GO TO 992
      IF(M .EQ. NRMODE) GO TO 993
      M=M+1
      RATIO(2*INDX1-1,M)=TMP1
      RATIO(2*INDX1  ,M)=TMP2
      RATIO(2*INDX2-1,M)=TMP3
      RATIO(2*INDX2  ,M)=TMP4
      TP(M)=CMPLX(TR1,TI1)
      STP=CSIN(TP(M)*DTR)
      STPR(M,KF)=REAL(STP)
      STPI(M,KF)=AIMAG(STP)
      GO TO 23
C
C..END OF INPUT FOR FREQ(KF)
28    NMF=M
      WRITE(2) SAVEMC
      GO TO 21
C
C..END OF MODE CONSTANT INPUT
29    NF=KF
      NEWD=1
      REWIND 2
      GO TO 10
C
30    IF(NEWD .EQ. -1) GO TO 995
      RETURN
989   PRINT 1989
      GO TO 999
990   PRINT 1990
      GO TO 999
991   PRINT 1991
      GO TO 999
992   PRINT 1992
      GO TO 999
993   PRINT 1993
      GO TO 999
994   PRINT 1994
      GO TO 999
```

```
995    PRINT 1995
999    CALL PLOT(0.,0.,999)
       STOP
1000   FORMAT(20A4)
1001   FORMAT('0',20A4)
1002   FORMAT('1',20A4)
1005   FORMAT(A50)
1006   FORMAT('0',A50)
1021   FORMAT(8X,3(2X,E8.0),2(2X,E10.0),2X,E5.0)
1023   FORMAT(I1,2F9.0,1X,4E15.0)
1989   FORMAT('0ERROR:  PREMATURE END OF FREQUENCY DATA')
1990   FORMAT('0ERROR:  N IS TOO LARGE')
1991   FORMAT('0ERROR:  TOO MANY FREQUENCIES INPUT')
1992   FORMAT('0ERROR:  MODE CARDS OUT OF ORDER')
1993   FORMAT('0ERROR:  TOO MANY MODES INPUT')
1994   FORMAT('0ERROR:  UNRECOGNIZED CONTROL CARD')
1995   FORMAT('0ERROR:  NO MODE DATA')
       END
```

```
      SUBROUTINE HTGAIN(IOPT,FREQ,SIGMA,EPSR,ALPHA,NRMODE,TP,Z,HG)
      COMPLEX    TP(1),HG(3,1),HG0/(0.,1.45749544)/,
     $           C,SSQ,NGSQ,SQROOT,RATIO,A1,A2,A3,A4,EXPZ,
     $           MI/(0.,-1.)/,ONE/(1.,0.)/
      COMPLEX*16 TPM,P0,H10,H20,H1PRM0,H2PRM0,P1,H1Z,H2Z,H1PRMZ,H2PRMZ
      REAL K,KA13,KA23
      DATA DTR/1.745329252E-02/
C
      NGSQ=CMPLX(EPSR,-SIGMA/(5.5633459E-8*FREQ))
      K=2.0958426E-02*FREQ
      IF(ALPHA .EQ. 0.) GO TO 5
      AK=ALPHA/K
      AK13=EXP(ALOG(AK)/3.)
      AK23=AK13**2
      KA13=1./AK13
      KA23=KA13**2
      P1=KA23*ALPHA*Z
      EXPZ=EXP(.5*ALPHA*Z)
    5 DO 20 M=1,NRMODE
      SSQ=CSIN(TP(M)*DTR)**2
      CSQ=ONE-SSQ
      SQROOT=CSQRT(NGSQ-SSQ)
      IF(AIMAG(TP(M)) .LE. -10. .OR. ALPHA .EQ. 0.) GO TO 10
      TPM=TP(M)
      P0=KA23*(ONE-SSQ)
      CALL MDHNKL(P0   ,H10,H20,H1PRM0,H2PRM0,TPM,'HG 1')
      CALL MDHNKL(P0+P1,H1Z,H2Z,H1PRMZ,H2PRMZ,TPM,'HG 2')
      A1=H10   *H2Z   -H1Z   *H20
      A2=H1PRM0*H2Z   -H1Z   *H2PRM0
      A3=H10   *H2PRMZ-H1PRMZ*H20
      A4=H1PRM0*H2PRMZ-H1PRMZ*H2PRM0
      RATIO=SQROOT/NGSQ
      C=.5*AK23+KA13*MI*RATIO
      HG(1,M)=EXPZ*(C*A1+A2)
      HG(2,M)=KA13*MI*SQROOT*A1+A2
      HG(3,M)=.5*AK*MI*HG(1,M)+AK13*MI*EXPZ*(C*A3+A4)
      IF(IOPT .EQ. 1) GO TO 20
      HG(1,M)=HG(1,M)/HG0
      HG(2,M)=HG(2,M)/HG0
      HG(3,M)=HG(3,M)/(RATIO*HG0)
      GO TO 20
   10 C=CSQRT(ONE-SSQ)
      EXPZ=CEXP(CMPLX(0.,K*Z)*C)
      A1=(NGSQ*C-SQROOT)/(NGSQ*C+SQROOT)
      A2=(C-SQROOT)/(C+SQROOT)
      HG(1,M)=EXPZ+A1/EXPZ
      HG(2,M)=EXPZ+A2/EXPZ
      HG(3,M)=(EXPZ-A1/EXPZ)*C
      IF(IOPT .EQ. 1) GO TO 20
      HG(1,M)=HG(1,M)/(ONE+A1)
      HG(2,M)=HG(2,M)/(ONE+A2)
      HG(3,M)=HG(3,M)/((ONE-A1)*C)
   20 CONTINUE
      RETURN
```

END

55

```
      SUBROUTINE FUNSPL(MD,LF)
C
      INCLUDE SPECANS.COMMONSPECS.LIST
C
      CALL FUNCVF(MD,LF)
      CALL SPLINE(XX,YY,B,C,D,LM)
      DO 46 I-1,NF
      YC(LF,MD,I)=YY(I)
      BC(LF,MD,I)=B(I)
      CC(LF,MD,I)=C(I)
      DC(LF,MD,I)=D(I)
   46 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TPLOT(FREQ,FL,FO,FC,DELTAF,NRPT1,NF)
C..CALCULATE X AND Y COORDINATES FOR TRANSMITTER SPECTRUM PLOT
C
      COMPLEX XMTR
      REAL INCL
      CHARACTER*50 LABELT,LABELR,LABELC
      CHARACTER*24 LABEL1
      CHARACTER*20 LABEL2
      CHARACTER*40 PLTLBL
      CHARACTER*30 TLABEL(4)
      PARAMETER NMAX=2049
      COMMON/FOUR/NFFT,FREQU,FREQL,INTPRT,TAUMAX,FREQO,PULSEW,
     $      RHOMIN,DELRHO,RHOMAX,TALT,RALT,INCL,THETA,ICOMP,
     $      IFLGTR,INTFLG,NPRNT,TAUO,NUMTAU,CHIPFR,NUMPLS,PULSED,
     $      IPLOT,IPLOT1
      COMMON/SIX/PLOTX(NMAX),PLOTY1(NMAX),PLOTY2(NMAX),NUMPTS
      COMMON/EIGHT/LABELT,LABELR,LABELC
      DIMENSION FREQ(1)
      DATA TLABEL/'FOR SQUARE WAVE                 ',
     $            'FOR GAUSSIAN                    ',
     $            'FOR MSK SIGNAL FORMAT           ',
     $            'FOR SLOW WAVE TAIL CALCULATION'/
C
      NUMPTS = 0
      F=FL
      DO 71 K=1,NRPT1
      CALL TRXMTR(K,F,FO,FC,PULSEW,PULSED,NUMPLS,IFLGTR,LABELT,XMTR)
      IF(F .LT. FREQ(1) .OR. F .GT. FREQ(NF)) GO TO 71
      NUMPTS = NUMPTS+1
      PLOTX(NUMPTS) = F/1000.
      IF(IFLGTR .EQ. 4) PLOTX(NUMPTS)=ALOG10(PLOTX(NUMPTS))
      IF(CABS(XMTR) .EQ. 0.)THEN
      PLOTY1(NUMPTS) = -1000.
      ELSE
      PLOTY1(NUMPTS) = 20.*ALOG10(CABS(XMTR))
      END IF
71    F=F+DELTAF
C
      LABEL1='      20*LOG(XMTR)
      LABEL2='TRANSMITTER SPECTRUM'
      IF(IFLGTR .EQ. 4) THEN
      SCALEY=10.
      ELSE
      SCALEY=20.
      ENDIF
      CALL PLSPEC(PLOTX,PLOTY1,NUMPTS,LABELT,LABEL1,LABEL2,SCALEY)
      XL = 2.1
      YL = -0.4
      IF(IFLGTR .EQ. 4) YL=-0.6
      CALL SYMBOL(XL,YL,.1,TLABEL(IFLGTR),0.,30)
      XL = 0.0
      YL = -0.8
      IF(IFLGTR .EQ. 1 .OR. IFLGTR .EQ. 2) THEN
      ENCODE(40,900,PLTLBL) NUMPLS
```

```
900   FORMAT('NUMBER OF PULSES = ',I2)
      CALL SYMBOL(XL,YL,.1,PLTLBL,0.,40)
      YL = YL-0.2
      IF(IFLGTR .EQ. 1) THEN
      ENCODE(40,905,PLTLBL) PULSEW
905   FORMAT('PULSE WIDTH = ',F6.1,' MICRO-SEC')
      ELSE
      ENCODE(40,906,PLTLBL) PULSEW
906   FORMAT('1/E HALF WIDTH = ',F6.1,' MICRO-SEC')
      ENDIF
      CALL SYMBOL(XL,YL,.1,PLTLBL,0.,40)
      YL = YL-0.2
      ENCODE(40,910,PLTLBL) PULSED
910   FORMAT('PULSE DELAY = ',F6.1,' MICRO-SEC')
      CALL SYMBOL(XL,YL,.1,PLTLBL,0.,40)
      ENDIF
      IF(IFLGTR .EQ. 3) THEN
      ENCODE(40,915,PLTLBL) CHIPFR
915   FORMAT('CHIP FREQ =',F5.2,' KHZ')
      CALL SYMBOL(XL,YL,.1,PLTLBL,0.,40)
      ENDIF
      CALL PLOT(0.,0.,-4)
C
      RETURN
      END
```

```
      SUBROUTINE TRXMTR(K,F,FO,FC,PULSEW,PULSED,NUMPLS,IFLGTR,LABELT,
     $          XMTR)
      COMPLEX XMTR,IM/(0.,1.)/,GOMEGA,HOMEGA,FAC1,FAC2,H1,H2
     $,RATIO1,RATIO2,TERM
      CHARACTER*50 LABELT
      DIMENSION AA(4),GAMMA(4)
      DATA AA/-16.8E3,15.35E3,1.0E3,0.45E3/
      DATA GAMMA/5.88E5,3.03E4,2.0E3,1.47E2/
      DATA TAUP/43.0E-6/,TAUV/180.0E-6/,V0/3.5E7/
      DATA PI/3.1415926/
      DATA TWOPI/6.2831853/
C     IFLGTR = 1 FOR SQUARE WAVE INPUT
C     IFLGTR = 2 FOR GAUSSIAN INPUT
C     IFLGTR = 3 FOR POWER SPECTRUM OF FSK SIGNAL WITH
C     IFLGTR = 4 FOR SLOW WAVE TAIL CALCULATION WILLIAMS SOURCE
C     MODULATION INDEX 0.5
      GO TO (300,400,500,600) IFLGTR
C
  300 CONTINUE
      IF (K .GT. 1) GO TO 20
      OMEGAO = TWOPI*FO
      TW = PULSEW*1.E-6
      TD = PULSED*1.E-6
      FREQO = FO*1.E-3
      ENCODE(50,11,LABELT) FREQO
   11 FORMAT('CARRIER FREQ = ',F5.1,' KHZ')
C
   20 OMEGA = TWOPI*F
      IF (ABS((OMEGA-OMEGAO)*TW).GT. 1.E-4) GO TO 21
      GOMEGA = -0.5*IM*TW-(CEXP(-IM*2.*OMEGAO*TW)-1.)/(4.*OMEGAO)
      GO TO 23
   21 IF(ABS((OMEGA+OMEGAO)*TW) .GT. 1.E-4)GO TO 22
      GOMEGA = 0.5*IM*TW-(CEXP(IM*2.*OMEGAO*TW)-1.)/(4.*OMEGAO)
      GO TO 23
   22 GOMEGA = -0.5*(CEXP(IM*(OMEGAO-OMEGA)*TW)-1.)/(OMEGAO
     $     -OMEGA)-0.5*(CEXP(-IM*(OMEGAO+OMEGA)*TW)-1.)/(
     $     OMEGAO+OMEGA)
   23 CONTINUE
      DO 28 NN=1,NUMPLS
      IF(NN .EQ. 1)THEN
      HOMEGA=(1.,0.)
      RATIO1 = CEXP(-IM*OMEGA*(TW+TD))
      ELSE
      HOMEGA = HOMEGA+RATIO1
      RATIO1 = RATIO1*RATIO1
      END IF
   28 CONTINUE
      XMTR = GOMEGA*HOMEGA*CEXP(IM*OMEGA*TW/2.)*2.386E8/FO
      RETURN
C
  400 CONTINUE
      IF (K .GT. 1) GO TO 40
      OMEGAO = TWOPI*FO
      TW = PULSEW*1.E-6
```

```
      TD = PULSED*1.E-6
      FREQ0 = F0*1.E-3
      ENCODE(50,11,LABELT) FREQ0
C
   40 OMEGA = TWOPI*F
      EXP1 = ((OMEGA0-OMEGA)*TW)**2/4.
      EXP2 = ((OMEGA0+OMEGA)*TW)**2/4.
      IF(EXP1 .LT. 1.E20)THEN
      FAC1 = EXP(-EXP1)/(2.*IM)
      ELSE
      FAC1 = 0.
      END IF
      IF(EXP2 .LT. 1.E20)THEN
      FAC2 = IM*EXP(-EXP2)/2.
      ELSE
      FAC2 = 0.
      END IF
      DO 48 NN = 1,NUMPLS
      IF(NN .EQ. 1)THEN
      H1 = (1.,0.)
      H2 = (1.,0.)
      RATIO1 = CEXP(IM*(OMEGA0-OMEGA)*TD)
      RATIO2 = CEXP(-IM*(OMEGA0+OMEGA)*TD)
      ELSE
      H1 = H1+RATIO1
      H2 = H2+RATIO2
      RATIO1 = RATIO1*RATIO1
      RATIO2 = RATIO2*RATIO2
      END IF
   48 CONTINUE
      XMTR = (FAC1*H1+FAC2*H2)*4.229E8*TW/F0
      RETURN
C
  500 CONTINUE
      IF (K .GT. 1) GO TO 50
      PISQ = PI**2
      CONST = 8./(PISQ*FC)
      FREQ0 = F0*1.E-3
      FCHIP = FC*1.E-3
      ENCODE(50,11,LABELT) FREQ0
   50 IF (ABS((F-F0)/FC-.25) .GT. 1.E-4) GO TO 51
      EPS = (F-F0)/FC-.25
      XMTR = PISQ/(16.*(1.+2.*EPS)**2)
      XMTR = XMTR+(COS((F+F0)*TWOPI/FC))**2/(1.-16.*(F+F0)**2/FC**2)**2
      XMTR = CONST*XMTR
      GO TO 53
   51 IF (ABS((F-F0)/FC+.25) .GT. 1.E-4) GO TO 52
      EPS = (F-F0)/FC+.25
      XMTR = PISQ/(16.*(-1.+2.*EPS)**2)
      XMTR = XMTR+(COS((F+F0)*TWOPI/FC))**2/(1.-16.*(F+F0)**2/FC**2)**2
      XMTR = CONST*XMTR
      GO TO 53
   52 XMTR = (COS((F-F0)*TWOPI/FC))**2/(1.-16.*(F-F0)**2/FC**2)**2
      XMTR = XMTR+(COS((F+F0)*TWOPI/FC))**2/(1.-16.*(F+F0)**2/FC**2)**2
      XMTR = CONST*XMTR
   53 CONTINUE
      XMTR = 2.386E8*XMTR/F0
```

```
      RETURN
C
600   CONTINUE
      IF(K .GT. 1) GO TO 60
      ENCODE(50,12,LABELT)
12    FORMAT('WILLIAMS SOURCE')
60    CONTINUE
      XMTR = (0.0,0.0)
      OMEGA = TWOPI*F
      DO 30 I=1,4
      TERM = IM*OMEGA+GAMMA(I)
30    XMTR = XMTR+AA(I)/(TERM**2)*(1.0-CEXP(-TAUP*TERM)/(1.0+TAUV*TERM))
      XMTR = XMTR*V0
      RETURN
      END
```

```
      SUBROUTINE RPLOT(FREQ,FL,F1,F2,F3,DELTAF,P,Q,NRPT1,NF)
C..CALCULATE X AND Y COORDINATES FOR RECEIVER SPECTRUM PLOT
C
      REAL INCL
      COMPLEX RCVR
      CHARACTER*50 LABELT,LABELR,LABELC
      CHARACTER*24 LABEL1
      CHARACTER*20 LABEL2
      PARAMETER NMAX=2049
      COMMON/FOUR/NFFT,FREQU,FREQL,INIPRT,TAUMAX,FREQ0,PULSEW,
     $      RHOMIN,DELRHO,RHOMAX,TALT,RALT,INCL,THETA,ICOMP,
     $      IFLGTR,INTFLG,NPRNT,TAU0,NUMTAU,CHIPFR,NUMPLS,PULSED,
     $      IPLOT,IPLOT1
      COMMON/SIX/PLOTX(NMAX),PLOTY1(NMAX),PLOTY2(NMAX),NUMPTS
      COMMON/EIGHT/LABELT,LABELR,LABELC
      DIMENSION FREQ(1)
C
C
      NUMPTS = 0
      F=FL
      DO 81 K=1,NRPT1
      CALL RECVR(K,F,F1,F2,F3,LABELR,P,Q,RCVR)
      IF(F .LT. FREQ(1) .OR. F .GT. FREQ(NF)) GO TO 81
      NUMPTS = NUMPTS+1
      PLOTX(NUMPTS) = F/1000.
      IF(IFLGTR .EQ. 4) PLOTX(NUMPTS)=ALOG10(PLOTX(NUMPTS))
      PLOTY1(NUMPTS) = 20.0*ALOG10(CABS(RCVR))
81    F=F+DELTAF
C
      LABEL1='      20*LOG(RCVR)
      LABEL2='RECEIVER SPECTRUM
      SCALEY=10.
      CALL PLSPEC(PLOTX,PLOTY1,NUMPTS,LABELR,LABEL1,LABEL2,SCALEY)
      CALL PLOT(0.,0.,-4)
C
      RETURN
      END
```

```
      SUBROUTINE RECVR(K,F,F1,F2,F3,LABELR,P,Q,RCVR)
      COMPLEX RCVR,IM/(0.0,1.0)/
      CHARACTER*50 LABELR
      IF(K .GT. 1) GO TO 20
      FREQ1 = F1*1.0E-3
      FREQ2 = F2*1.0E-3
      FREQ3 = F3*1.0E-3
      IF(P .EQ. 0.0) THEN
      ENCODE(50,10,LABELR) INT(P),INT(Q),FREQ2,FREQ3
10    FORMAT('P=',I1,'  Q=',I1,'  F2=',F6.3,'KHZ  F3=',F6.3,'KHZ')
      ELSE
      ENCODE(50,11,LABELR) INT(P),INT(Q),FREQ1,FREQ2,FREQ3
11    FORMAT('P=',I1,'  Q=',I1,'  F1=',F6.3,'KHZ  F2=',F6.3,'KHZ  F3=',
     $  F6.3,'KHZ')
      ENDIF
20    CONTINUE
      IF(F .EQ. 0.0) THEN
         IF(P .EQ. 0.0) THEN
         RCVR = 1.0/(1.0+IM*(F-F2)/F3)**Q+1.0/(1.0+IM*(F+F2)/F3)**Q
         ELSE
         RCVR = (0.0,0.0)
         ENDIF
      ELSE
      RCVR = (IM*F/F1)**P/(1.0+IM*F/F1)**P*(1.0/(1.0+IM*(F-F2)/F3)**Q
     $  +1.0/(1.0+IM*(F+F2)/F3)**Q)
      ENDIF
      IF(F2 .EQ. 0.0) RCVR=RCVR/2.0
      RETURN
      END
```

```
      SUBROUTINE PLSPEC(PLOTX,PLOTY1,NUMPTS,LABEL,LABEL1,LABEL2,SCALEY)
C..DRAW BORDER,CURVE,X-LABEL,Y-LABEL, AND SPECTRUM LABELS FOR ALL PLOTS
      REAL INCL
      CHARACTER*50 LABEL
      CHARACTER*24 LABEL1
      CHARACTER*20 LABEL2
      DIMENSION FTIC(40)
      PARAMETER NMAX=2049
      LOGICAL UP(NMAX)
      COMMON/FOUR/NFFT,FREQU,FREQL,INTPRT,TAUMAX,FREQ0,PULSEW,
     $      RHOMIN,DELRHO,RHOMAX,TALT,RALT,INCL,THETA,ICOMP,
     $      IFLGTR,INTFLG,NPRNT,TAU0,NUMTAU,CHIPFR,NUMPLS,PULSED,
     $      IPLOT,IPLOT1
      COMMON/SEVEN/XMIN,XMAX,XTIC,SCALEX,XLNG,YLNG
      COMMON/TEN/XL,YL
      DIMENSION PLOTX(1),PLOTY1(1)
C
      YMAX=PLOTY1(1)
      DO 72 K=2,NUMPTS
      IF(YMAX .LT. PLOTY1(K)) YMAX=PLOTY1(K)
 72   CONTINUE
      IF(YMAX .GE. 0.0) THEN
      YMAX=AINT(YMAX/10.+.99)*10.
      ELSE
      YMAX = INT(YMAX/10.0)*10.0
      ENDIF
      YMIN=YMAX-SCALEY*YLNG
      YTIC=10.
      DO 73 K=1,NUMPTS
      UP(K)=.FALSE.
      IF(PLOTY1(K) .GE. YMIN) GO TO 73
      UP(K)=.TRUE.
 73   CONTINUE
      CALL PLOT(1.,3.,-3)
      IF(IFLGTR .EQ. 4) THEN
      XMIN=-3.0
      XMAX=ALOG10(3.0)
      SCALEX=(XMAX-XMIN)/XLNG
      CALL LOGTIC(XMIN,XMAX,FTIC,40,NRTIC)
      CALL BORDER(XLNG,XMIN,XMAX,FTIC,NRTIC,YLNG,YMIN,YMAX,YTIC,1)
      CALL SYMBOL(-.2,-.3,.1,'10',0.,2)
      CALL SYMBOL(4.5,-.3,.1,'10',0.,2)
      ELSE
      CALL BORDER(XLNG,XMIN,XMAX,XTIC,1, YLNG,YMIN,YMAX,YTIC,1)
      ENDIF
      CALL CURVE(PLOTX,PLOTY1,UP,NUMPTS,XMIN,YMIN,SCALEX,SCALEY,1)
      XL=-.2
      YL=.5*(YLNG-2.2)
      CALL SYMBOL(XL,YL,.1,LABEL1,90.,24)
      XL=.5*(XLNG-1.0)
      YL=-.2
      CALL SYMBOL(XL,YL,.1,'FREQ (KHZ)',0.,10)
      XL=0.
      YL=YL-.2
```

```
      IF(IFLGTR .EQ. 4) YL=YL-.2
      CALL SYMBOL(XL,YL,.1,LABEL2,0.,20)
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,LABEL ,0.,50)
C
      RETURN
      END
```

```
      SUBROUTINE CPPLOT(FREQ,FL,FO,FC,DELTAF,NRPT1,NF,F1,F2,F3,P,Q,RHO)
C..CALCULATE X AND Y COORDINATES FOR CHANNEL SPECTRUM PLOT
C  AND PRODUCT SPECTRUM PLOT
C
      COMPLEX XMTR,RCVR,CHNL,PROD
      REAL INCL
      CHARACTER*50 LABELT,LABELR,LABELC
      CHARACTER*40 PLTLBL
      CHARACTER*24 LABEL1
      CHARACTER*20 LABEL2
      PARAMETER NMAX=2049
      COMMON/FOUR/NFFT,FREQU,FREQL,INTPRT,TAUMAX,FREQ0,PULSEW,
     $      RHOMIN,DELRHO,RHOMAX,TALT,RALT,INCL,THETA,ICOMP,
     $      IFLGTR,INTFLG,NPRN1,TAU0,NUMTAU,CHIPFR,NUMPLS,PULSED,
     $      IPLOT,IPLOT1
      COMMON/SIX/PLOTX(NMAX),PLOTY1(NMAX),PLOTY2(NMAX),NUMPTS
      COMMON/EIGHT/LABELT,LABELR,LABELC
      COMMON/NINE/X(NMAX),Y(NMAX)
      COMMON/TEN/XL,YL
      DIMENSION FREQ(1)
C
      NUMPTS = 0
      F=FL
      DO 93 K=1,NRPT1
      CALL TRXMTR(K,F,F0,FC,PULSEW,PULSED,NUMPLS,IFLGTR,LABELT,XMTR)
      CALL RECVR(K,F,F1,F2,F3,LABELR,P,Q,RCVR)
      CALL CHANEL(F,RHO,CHNL)
      PROD=XMTR*RCVR*CHNL
      X(K)= REAL(PROD)
      Y(K)=AIMAG(PROD)
      IF(F .LT. FREQ(1) .OR. F .GT. FREQ(NF)) GO TO 92
      IF(IPLOT1 .NE. 0) THEN
      NUMPTS = NUMPTS+1
      PLOTX(NUMPTS) = F/1000.
      IF(IFLGTR .EQ. 4) PLOTX(NUMPTS)=ALOG10(PLOTX(NUMPTS))
      PLOTY1(NUMPTS) = 20.0*ALOG10(CABS(CHNL))
      AUX = CABS(CMPLX(X(K),Y(K)))
      IF(AUX .EQ. 0.)THEN
      PLOTY2(NUMPTS) = -1000.
      ELSE
      PLOTY2(NUMPTS) = 20.*ALOG10(AUX)
      END IF
      ENDIF
   92 IF(K .GT. NPRNT .AND. MOD(K,INTPRT) .NE. 0)GO TO 93
      PRINT 1092,F,XMTR,RCVR,CHNL,X(K),Y(K),K
93    F=F+DELTAF
C
      IF(IPLOT1 .EQ. 0) RETURN
C
      IF(RHO .EQ. 0.0) GO TO 110
C
C  CHANNEL SPECTRUM PLOT
      LABEL1='      20*LOG(CHNL)
      LABEL2='CHANNEL SPECTRUM
```

```
      SCALEY=10.
      CALL PLSPEC(PLOTX,PLOTY1,NUMPTS,LABELC,LABEL1,LABEL2,SCALEY)
      CALL PLLABL(RHO)
      CALL PLOT(0.,0.,-4)
C
C  PRODUCT SPECTRUM PLOT
110   SCALEY=20.
      LABEL1=' 20*LOG(XMTR*RCVR*CHNL)
      LABEL2='PRODUCT SPECTRUM     '
      CALL PLSPEC(PLOTX,PLOTY2,NUMPTS,LABELT,LABEL1,LABEL2,SCALEY)
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,LABELR,0.,50)
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,LABELC,0.,50)
      CALL PLLABL(RHO)
      IF(IFLGIP .EQ. 3) THEN
      ENCODE(40,915,PLTLBL) CHIPFR
915   FORMAT('CHIP FREQ =',F5.2,' KHZ')
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,PLTLBL,0.,40)
      ENDIF
      CALL PLOT(0.,0.,-4)
      RETURN
1092  FORMAT(11X,1P9E12.4,I6)
      END
```

```
      SUBROUTINE CHANEL(F,RHO,CHNL)
C
      INCLUDE SPECANS.COMMONSPECS,LIST
C
      COMPLEX IM/(0.0,1.0)/,CONST,MSUM,MIKRHO,STP,EXC,CHNL
      DIMENSION E(4)
C
      CHNL=(1.,0.)
      IF(RHO .EQ. 0.) GO TO 99
      CONST = 9.323E-8*(IM*F)**1.5/SQRT(SIN(RHO/6371.))
      MSUM = (0.0,0.0)
      DO 45 MD=1,NM
      LF=0
23    INIT=0
      LF=LF+1
      DO 25 I=1,NF
      IF(MODE(MD,I) .EQ. 0) GO TO 25
      JJ = I-KK(MD)+1
      MF = MODE(MD,I)
      XX(JJ)= FREQ(I)
      YY(JJ)= YC(LF,MD,JJ)
      B(JJ) = BC(LF,MD,JJ)
      C(JJ) = CC(LF,MD,JJ)
      D(JJ) = DC(LF,MD,JJ)
25    CONTINUE
      IF(F.GE.XX( 1)) GO TO 30
      GO TO 45
30    IF(F.LE.XX(MF)) GO TO 33
      GO TO 45
33    E(LF)=SPEVAL(F,XX,YY,B,C,D,MF,INIT)
      IF(LF.LT.4) GO TO 23
      EXC=CMPLX(E(1),E(2))
      STP=CMPLX(E(3),E(4))
      MIKRHO=CMPLX(0.,-20.958445E-6*F*RHO)
      MSUM=MSUM+EXC*CEXP(MIKRHO*(STP-(1.,0.)))
45    CONTINUE
      CHNL=CONST*MSUM
99    RETURN
      END
```

```
      FUNCTION  SPEVAL (XVAL. X, Y, B, C, D, N, INIT)
      DIMENSION X(1), Y(1), B(1), C(1), D(1)
C
C   SP EVAL EVALUATES THE INTERPOLATING CUBIC SPLINE
C   FOR THE DATA (X(I),Y(I)), I=1,...N AT X = XVAL.
C   IT IS ASSUMED THAT THE CUBIC POLYNOMIALS GIVEN
C   IN B(I), C(I), D(I) HAVE BEEN PREVIOUSLY
C   COMPUTED BY THE SUBROUTINE SPLINE OR PSPLIN.
C   INIT IS AN ESTIMATE OF THE INTERVAL WHERE XVAL
C   LIES, X(INIT) .LE. XVAL .LE. X(INIT+1), BUT NEED
C   NOT BE USED.  SET INIT=0 IF THERE IS NO ESTIMATE.
C   ON RETURN, INIT WILL CONTAIN THE INTERVAL NUMBER.
C
      FN = N - 1
      EPS = 1.0E-3 * (X(N) - X(1)) / FN
      IF (XVAL .LT. X(1)-EPS)  GO TO 900
      IF (XVAL .GT. X(N)+EPS)  GO TO 800
      IF (INIT .LE. 0)  GO TO 200
      IF (INIT .GE. N)  GO TO 200
C
      IF (XVAL .LT. X(INIT))  GO TO 150
  100 IF (XVAL .LT. X(INIT+1))  GO TO 300
      IF (INIT+1 .GE. N)  GO TO 300
      INIT = INIT + 1
      GO TO 100
  150 INIT = INIT - 1
      IF (INIT .LE. 0)  GO TO 200
      IF (XVAL .GE. X(INIT))  GO TO 300
      GO TO 150
C
  200 INIT = 1
      GO TO 100
C
  300 H = XVAL - X(INIT)
      SPEVAL = ((D(INIT)*H + C(INIT))*H + B(INIT))*H + Y(INIT)
      RETURN
  800 PRINT 900
      PRINT 901,  XVAL,X(1),X(N)
      RETURN
  900 FORMAT (' ERROR IN SP EVAL:  XVAL OUT OF INTERPOLATION RANGE')
  901 FORMAT (5X,' XVAL =',1PE12.5,'  X(1) =',1PE12.5,' X(N)=',1PE12.5/)
      END
```

```
      SUBROUTINE NLOGN (N,X,Y,SIGNT,A,B)
      DIMENSION X(1), Y(1), M(15)
      DATA  TWOPI/6.2831 85307 17958 64769 25287/
      DATA HALFPI/1.5707 96326 79489 66192 31322/
      M(N) = 1
      DO 1 I=N-1,1,-1
    1 M(I) = M(I+1)*2
      LX = M(1)*2
      FLX = LX
      V0 = SIGNT*TWOPI/FLX
      FLXI1 = (B-A)/FLX
      NBLOCK = 1
      LBLOCK = LX
      DO 6 L = 1,N
      LBHALF = LBLOCK/2
      KO = 0
      ISTART = 0
      DO 5 IBLOCK = 1, NBLOCK
      FK = KO
      V = V0*FK
      Z1 = COS(V)
      Z2 = SIN(V)
      IF (ABS(V + HALFPI)  .GE.  1.0E-6) GO TO 12
      Z2 = -1.0
   12 DO 2 I = 1, LBHALF
      J = ISTART + I
      K = J + LBHALF
      Q1 = X(K)*Z1 - Y(K)*Z2
      Q2 = Y(K)*Z1 + X(K)*Z2
      X(K) = X(J) - Q1
      Y(K) = Y(J) - Q2
      X(J) = X(J) + Q1
      Y(J) = Y(J) + Q2
    2 CONTINUE
      DO 3 I = 2, N
      II = I
      IF (AND(M(I),KO) .LE. 0) GO TO 4
      KO = KO - M(I)
    3 CONTINUE
    4 KO = KO + M(II)
      ISTART = ISTART + LBLOCK
    5 CONTINUE
      NBLOCK = NBLOCK*2
    6 LBLOCK = LBLOCK/2
      KO = 0
      DO 50 K = 1, LX
      K1 =KO + 1
      IF (K1 .LE. K) GO TO 55
      H1 = X(K1)
      H2 = Y(K1)
      X(K1) = X(K)
      Y(K1) = Y(K)
      X(K) = H1
      Y(K) = H2
```

```
55    DO 85 I = 1, N
      II = I
      IF (AND(M(I),KO) .LE. 0) GO TO 75
      KO = KO - M(I)
85    CONTINUE
75    KO = KO + M(II)
50    CONTINUE
      DO 100 K=1,LX
      X(K)=X(K)*FLXI1
      Y(K)=Y(K)*FLXI1
100   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INITWF(IFLGTR,PULSEW,PULSED,NUMPLS,TAUO,TAUMAX,PLOTX3,
     $PLOTY3)
      DIMENSION PLOTX3(300),PLOTY3(300)
      GO TO(10,20,30)IFLGTR
   10 TW = PULSEW*1.E-6
      TD = PULSED*1.E-6
      NUMPTS = 0
      T = -TW/2.
      NMAX = 4*NUMPLS
      DO 1 NN = 1,NMAX
      IF(NN .GE. 5)GO TO 2
      IF(NN .EQ. 1)GO TO 3
      IF(NN .EQ. 2)GO TO 4
      IF(NN .EQ. 3)GO TO 5
      GO TO 6
    3 NUMPTS = NUMPTS+1
      PLOTX3(NUMPTS) = T
      PLOTY3(NUMPTS) = 0.
      GO TO 1
    4 NUMPTS = NUMPTS+1
      PLOTX3(NUMPTS) = T*.999
      PLOTY3(NUMPTS) = 1.
      GO TO 1
    5 T = T+TW
      NUMPTS = NUMPTS+1
      PLOTX3(NUMPTS) = T*.999
      PLOTY3(NUMPTS) = 1.
      GO TO 1
    6 NUMPTS = NUMPTS+1
      PLOTX3(NUMPTS) = T
      PLOTY3(NUMPTS) = 0.
      GO TO 1
    2 IF(MOD(NN,4) .EQ. 1)GO TO 7
      IF(MOD(NN,4) .EQ. 2)GO TO 8
      IF(MOD(NN,4) .EQ. 3)GO TO 9
      GO TO 11
    7 T = T+TD
      NUMPTS = NUMPTS+1
      PLOTX3(NUMPTS) = T
      PLOTY3(NUMPTS) = 0.
      GO TO 1
    8 NUMPTS = NUMPTS+1
      PLOTX3(NUMPTS) = T*1.001
      PLOTY3(NUMPTS) = 1.
      GO TO 1
    9 T = T+TW
      NUMPTS = NUMPTS+1
      PLOTX3(NUMPTS) = T*.999
      PLOTY3(NUMPTS) = 1.
      GO TO 1
   11 NUMPTS = NUMPTS+1
      PLOTX3(NUMPTS) = T
      PLOTY3(NUMPTS) = 0.
    1 CONTINUE
```

```
      GO TO 30
   20 TW = PULSEW*1.E-6
      TD = PULSED*1.E-6
      NUMPTS = 0
      DELTAU = (TAUMAX-TAU0)/200.
      TAU = TAU0
      DO 21 NN = 1,201
      SUM = 0.
      DO 22 JJ = 1,NUMPLS
   22 SUM = SUM+EXP(-((TAU-(JJ-1)*TD)/TW)**2)
      NUMPTS = NUMPTS+1
      PLOTX3(NUMPTS) = TAU
      PLOTY3(NUMPTS) = SUM
   21 TAU = TAU+DELTAU
   30 RETURN
      END
```

```
      SUBROUTINE FILON(N,X,Y,TAU,FU,FL,SUM,SUMP)
      COMPLEX IM,SUM,SUMP,SUM1,SUM2,SUM3,SUM4,G,H
      DIMENSION X(1),Y(1)
      DATA IM/(0.,1.)/
      DATA TWOPI/6.2831853/
      NP = 2*N
      NP1 = NP+1
      SUM1 = (0.,0.)
      SUM2 = (0.,0.)
      SUM3 = (0.,0.)
      SUM4 = (0.,0.)
      DELF = (FU-FL)/NP
      F = FL
      DO 1 L=1,NP1
      IF (MOD(L,2) .EQ. 1) THEN
      G = X(L)+IM*Y(L)
      H = CEXP(TWOPI*IM*F*TAU)
      SUM1 = SUM1+G*H
      IF (MOD(L,4) .EQ. 1) THEN
      SUM3 = SUM3+G*H
      ELSE
      IF (MOD(L,4) .EQ. 3) THEN
      SUM4 = SUM4+G*H
      ELSE
      GO TO 2
      END IF
    2 CONTINUE
      END IF
      ELSE
      SUM2 = SUM2+(X(L)+IM*Y(L))*CEXP(TWOPI*IM*F*TAU)
      END IF
      F = F+DELF
    1 CONTINUE
      PHI1 = TWOPI*TAU*DELF
      PHI2 = 2.*PHI1
      C1 = COS(PHI1)
      C2 = COS(PHI2)
      S1 = SIN(PHI1)
      S2 = SIN(PHI2)
      IF(ABS(PHI1) .LT. 1.D-5)THEN
      BETA1 = 2./3.
      BETA2 = BETA1
      GAMMA1 = 4./3.
      GAMMA2 = GAMMA1
      ELSE
      BETA1 = 2.*(1.+C1**2-2.*S1*C1/PHI1)/PHI1**2
      BETA2 = 2.*(1.+C2**2-2.*S2*C2/PHI2)/PHI2**2
      GAMMA1 = 4.*(S1/PHI1-C1)/PHI1**2
      GAMMA2 = 4.*(S2/PHI2-C2)/PHI2**2
      END IF
      SUM = (BETA1*SUM1+GAMMA1*SUM2)*DELF
      SUMP = (BETA2*SUM3+GAMMA2*SUM4)*2.*DELF
      RETURN
      END
```

```
      SUBROUTINE WOPLOT(RHO)
C..WAVEFORM OUTPUT PLOTS
      REAL INCL
      COMMON/FOUR/NFFT,FREQU,FREQL,INTPRT,TAUMAX,FREQO,PULSEW,
     $      RHOMIN,DELRHO,RHOMAX,TALT,RALT,INCL,THETA,ICOMP,
     $      IFLGTR,INTFLG,NPRNT,TAU0,NUMTAU,CHIPER,NUMPLS,PULSED,
     $      IPLOT,IPLOT1
      COMMON/SEVEN/XMIN,XMAX,XTIC,SCALEX,XLNG,YLNG
C
      XMIN=TAU0*1000.0
      XMAX=TAUMAX*1000.
      XTIC = (XMAX-XMIN)/20.0
      SCALEX=(XMAX-XMIN)/XLNG
      IF(IFLGTR .EQ. 1 .OR. IFLGTR .EQ. 2) CALL PLOT12(RHO)
      IF(IFLGTR .EQ. 3) CALL PLOT3(RHO)
      IF(IFLGTR .EQ. 4) CALL PLOT4(RHO)
C
      RETURN
      END
```

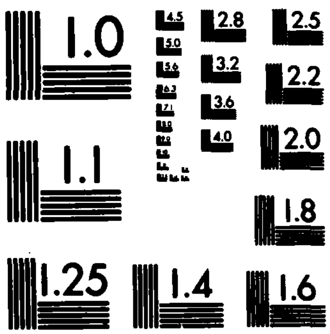
```
      SUBROUTINE PLOT12(RHO)
      PARAMETER NMAX=2049
      LOGICAL UP(NMAX)
      REAL INCL
      CHARACTER*40 PLTLBL
      CHARACTER*50 LABELT,LABELR,LABELC
      COMMON/FOUR/NFFT,FREQU,FREQL,INTPRT,TAUMAX,FREQ0,PULSEW,
     $      RHOMIN,DELRHO,RHOMAX,TALT,RALT,INCL,THETA,ICOMP,
     $      IFLGTR,INTFLG,NPRNT,TAU0,NUMTAU,CHIPFR,NUMPLS,PULSED,
     $      IPLOT,IPLOT1
      COMMON/STA/PLOTX(NMAX),PLOTY1(NMAX),PLOTY2(NMAX),NUMPTS
      COMMON/SEVEN/YMIN,XMAX,XTIC,SCALEX,XLNG,YLNG
      COMMON/EIGHT/LABELT,LABELR,LABELC
      COMMON/TEN/XL,YL
      COMMON/COM11/PLOTX3(300),PLOTY3(300)
C
      DO 142 K=1,NUMPTS
      UP(K)=.FALSE.
  142 CONTINUE
      DO 200 J=1,2
      CALL PLOT(1.,3.,-3)
      XL=-0.2
      YL=0.5*(YLNG-1.5)
      IF(J .EQ. 1) THEN
      YMAX=ABS(PLOTY1(1))
      DO 141 K=2,NUMPTS
      YK=ABS(PLOTY1(K))
      IF(YMAX .LT. YK) YMAX=YK
  141 CONTINUE
      YMIN = 0.
      SCALEY = YMAX/YLNG
      CALL BORDER(XLNG,XMIN,XMAX,XTIC,1,YLNG+YLNG/10.0,0.,1.1,.1,1)
      CALL CURVE(PLOTX,PLOTY1,UP,NUMPTS,XMIN,YMIN,SCALEX,SCALEY,1)
      ENCODE(40,900,PLTLBL) 20.0*LOG10(YMAX)
  900 FORMAT('SIGNAL MAX=',F7.2,' DB/MICRO-V/M/KW')
      CALL SYMBOL(0.5,6.2,.1,PLTLBL,0.,40)
      IF(IFLGTR .EQ. 1) THEN
      NPOINT = NUMPLS*4
      ELSE
      NPOINT = 201
      ENDIF
      DO 145 K=1,NPOINT
      PLOTX3(K) = PLOTX3(K)*1000.0
  145 CONTINUE
      YMAX = PLOTY3(1)
      DO 148 K=2,NPOINT
      IF(YMAX .LT. PLOTY3(K)) YMAX=PLOTY3(K)
  148 CONTINUE
      SCALEY = YMAX/YLNG
      CALL CURVE(PLOTX3,PLOTY3,UP,NPOINT,  XMIN,YMIN,SCALEX,SCALEY,4)
      CALL SYMBOL(XL,YL,.1,'    ENVELOPE   ',90.0,15)
      XL=5.2
      YL=2.6
      CALL SYMBOL(XL,YL,.1,'ENVELOPE(DB)',90.,12)
```

```
      XL=5.3
      YL=0.6
      X1 = 0.1
      DO 143 K=1,10
      X2=20.0*LOG10(X1)
      CALL NUMBER(XL,YL,.1,X2,0.,2)
      X1=X1+0.1
      YL=YL+0.6
143   CONTINUE
      ELSE
      YMIN=PLOTY2(1)
      YMAX=PLOTY2(1)
      DO 150 K=2,NUMPTS
      IF(YMIN .GT. PLOTY2(K)) YMIN=PLOTY2(K)
      IF(YMAX .LT. PLOTY2(K)) YMAX=PLOTY2(K)
150   CONTINUE
      SCALEY=(YMAX-YMIN)/YLNG
      CALL BORDER(XLNG,XMIN,XMAX,XTIC,1,YLNG,-1.0,1.0,.2,1)
      CALL CURVE(PLOTX,PLOTY2,UP,NUMPTS,XMIN,YMIN,SCALEX,SCALEY,1)
      CALL SYMBOL(XL,YL,.1,'WAVEFORM OUTPUT',90.0,15)
      ENDIF
      XL=.5*(XLNG-1.8)
      YL=-.2
      CALL SYMBOL(XL,YL,.1,'TAU (MILLISECONDS)',0.,18)
      XL=0.
      YL=YL-.2
      IF(IFLGTR .EQ. 1) THEN
      CALL SYMBOL(XL,YL,.1,'SQUARE WAVE',0.,11)
      ELSE
      CALL SYMBOL(XL,YL,.1,'GAUSSIAN',0.,8)
      ENDIF
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,LABELT,0.,50)
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,LABELR,0.,50)
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,LABELC,0.,50)
      CALL PLLABL(RHO)
      XL=0.
      YL=YL-.2
      ENCODE(40,901,PLTLBL) NUMPLS
901   FORMAT('NUMBER OF PULSES = ',I2)
      CALL SYMBOL(XL,YL,.1,PLTLBL,0.,40)
      YL=YL-.2
      IF(IFLGTR .EQ. 1) THEN
      ENCODE(40,905,PLTLBL) PULSEW
905   FORMAT('PULSE WIDTH = ',F6.1,' MICRO-SEC')
      ELSE
      ENCODE(40,906,PLTLBL) PULSEW
906   FORMAT('1/E HALF WIDTH = ',F6.1,' MICRO-SEC')
      ENDIF
      CALL SYMBOL(XL,YL,.1,PLTLBL,0.,40)
      YL=YL-.2
      ENCODE(40,910,PLTLBL) PULSED
910   FORMAT('PULSE DELAY = ',F6.1,' MICRO-SEC')
      CALL SYMBOL(XL,YL,.1,PLTLBL,0.,40)
      CALL PLOT(0.,0.,-4)
```

AD-A133 876 ELF/VLF (EXTREMELY LLW FREQUENCY/VERY LOW FREQUENCY) 2/2
LONG PATH PULSE PROG. (U) NAVAL OCEAN SYSTEMS CENTER
SAN DIEGO CA R A PAPPERT ET AL. AUG 83 NOSC/TR-891

UNCLASSIFIED F/G 20/14 NL

END
FILMED



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
200   CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE PLOT3(RHO)
      PARAMETER NMAX=2049
      LOGICAL UP(NMAX)
      REAL INCL
      CHARACTER*40 PLTLBL
      CHARACTER*50 LABELT,LABELR,LABELC
      COMMON/FOUR/NFFT,FREQU,FREQL,INTPRT,TAUMAX,FREQ0,PULSEW,
     $      RHOMIN,DELRHO,RHOMAX,TALT,RALT,INCL,THETA,ICOMP,
     $      IFLGIR,INTFLG,NPRNT,TAUG,NUMTAU,CHIPER,NUMPLS,PULSED,
     $      IPLOT,IPLOT1
      COMMON/SIX/PLOTX(NMAX),PLOTY1(NMAX),PLOTY2(NMAX),NUMPTS
      COMMON/SEVEN/XMIN,XMAX,XTIC,SCALEX,XLNG,YLNG
      COMMON/EIGHT/LABELT,LABELR,LABELC
      COMMON/TEN/XL,YL
C
      DO 200 J=1,2
      CALL PLOT(1.,3.,-3)
      XL=-0.2
      YL=0.5*(YLNG-2.7)
      IF(J .EQ. 1) THEN
      YMAX=PLOTY1(1)
      DO 140 K=2,NUMPTS
      IF(YMAX .LT. PLOTY1(K)) YMAX=PLOTY1(K)
  140 CONTINUE
      IF(YMAX .GE. 0.0) THEN
      YMAX=INT(YMAX/10.0+0.99)*10.0
      ELSE
      YMAX=INT(YMAX/10.0)*10.0
      ENDIF
      YMIN=YMAX-60.0
      SCALEY=10.0
      DO 142 K=1,NUMPTS
      UP(K)=.FALSE.
      IF(PLOTY1(K) .GE. YMIN .AND. PLOTY1(K) .LE. YMAX) GO TO 142
      UP(K)=.TRUE.
  142 CONTINUE
      CALL BORDER(XLNG,XMIN,XMAX,XTIC,1,YLNG,YMIN,YMAX,10.0,1)
      CALL CURVE(PLOTX,PLOTY1,UP,NUMPTS,XMIN,YMIN,SCALEX,SCALEY,1)
      CALL SYMBOL(XL,YL,.1,'CORRELATION DB/MICRO-V/M/KW',90.0,27)
      ELSE
      DO 145 K=1,NUMPTS
      UP(K)=.FALSE.
  145 CONTINUE
      YMIN=PLOTY2(1)
      YMAX=PLOTY2(1)
      DO 150 K=2,NUMPTS
      IF(YMIN .GT. PLOTY2(K))YMIN=PLOTY2(K)
      IF(YMAX .LT. PLOTY2(K))YMAX=PLOTY2(K)
  150 CONTINUE
      IF(YMAX .GT. 0.0) THEN
      YMAX=INT(YMAX/10.0+0.99)*10.0
      ELSE
      YMAX=INT(YMAX/10.0)*10.0
      ENDIF
```

```
      IF(YMIN .GE. 0.0) THEN
      YMIN=INT(YMIN/10.0)*10.0
      ELSE
      YMIN=INT(YMIN/10.0-0.99)*10.0
      ENDIF
      YTIC=(YMAX-YMIN)/10.0
      SCALEY=(YMAX-YMIN)/YLNG
      CALL BORDER(XLNG,XMIN,XMAX,XTIC,1,YLNG,YMIN,YMAX,YTIC,1)
      CALL CURVE(PLOTX,PLOTY2,UP,NUMPTS,XMIN,YMIN,SCALEX,SCALEY,1)
      CALL SYMBOL(XL,YL,.1,'    CORRELATION PHASE-RAD    ',90.0,27)
      ENDIF
      XL=.5*(XLNG-1.8)
      YL=-.2
      CALL SYMBOL(XL,YL,.1,'TAU (MILLISECONDS)',0.,18)
      XL=0.
      YL=YL-.2
      IF(J .EQ. 1) THEN
      CALL SYMBOL(XL,YL,.1,'CORRELATOR OUTPUT FOR MSK FORMAT',0.,32)
      ELSE
      CALL SYMBOL(XL,YL,.1,'CORRELATOR PHASE',0.,16)
      ENDIF
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,LABELC,0.,50)
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,LABELT,0.,50)
      ENCODE(40,915,PLTLBL) CHIPFR
915   FORMAT('CHIP FREQ =',F5.2,' KHZ')
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,PLTLBL,0.,40)
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,LABELR,0.,50)
      CALL PLLABL(RHO)
      CALL PLOT(0.,0.,-4)
200   CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE PLOT4(RHO)
      PARAMETER NMAX=2049
      LOGICAL UP(NMAX)
      CHARACTER*50 LABELT,LABELR,LABELC
      COMMON/SIX/PLOTX(NMAX),PLOTY1(NMAX),PLOTY2(NMAX),NUMPTS
      COMMON/SEVEN/XMIN,XMAX,XTIC,SCALEX,XLNG,YLNG
      COMMON/EIGHT/LABELT,LABELR,LABELC
C
      DO 142 K=1,NUMPTS
      UP(K)=.FALSE.
  142 CONTINUE
      CALL PLOT(1.,3.,-3)
      YMAX=500.0
      YMIN=-2000.0
      YTIC=(YMAX-YMIN)/10.0
      SCALEY=(YMAX-YMIN)/YLNG
      CALL BORDER(XLNG,XMIN,XMAX,XTIC,1,YLNG,YMIN,YMAX,YTIC,1)
      CALL CURVE(PLOTX,PLOTY2,UP,NUMPTS,XMIN,YMIN,SCALEX,SCALEY,1)
      XL=-0.2
      YL=0.5*(YLNG-2.8)
      CALL SYMBOL(XL,YL,.1,'WAVEFORM OUTPUT MICRO-VOLT/M',90.,28)
      XL=.5*(XLNG-1.8)
      YL=-.2
      CALL SYMBOL(XL,YL,.1,'TAU (MILLISECONDS)',0.,18)
      XL=0.
      YL=-.4
      CALL SYMBOL(XL,YL,.1,'SLOW WAVE TAIL CALCULATION'.0.,26)
      YL=-.6
      CALL SYMBOL(XL,YL,.1,LABELT,0.,50)
      YL=-.8
      CALL SYMBOL(XL,YL,.1,LABELR,0.,50)
      YL=-1.0
      CALL SYMBOL(XL,YL..1,LABELC,0.,50)
      CALL PLOT(0.,0.,-4)
C
      RETURN
      END
```

```
      SUBROUTINE PLLABL(RHO)
C..PUT LABELS ON PLOTS
C
      REAL INCL
      CHARACTER*1 LABEL(3)
      CHARACTER*40 PLTLBL
      COMMON/FOUR/NFFT,FREQU,FREQL,INTPRT,TAUMAX,FREQO,PULSEW,
     $      RHOMIN,DELRHO,RHOMAX,TALT,RALT,INCL,THETA,ICOMP.
     $      IFLGTR,INTFLG,NPRNT,TAUO,NUMTAU,CHIPFR,NUMPLS,PULSED,
     $      IPLOT,IPLOT1
      COMMON/TEN/XL,YL
      DATA LABEL/'Z','Y','X'/
C
      XL=0.
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,LABEL(ICOMP),0.,1)
      XL=XL+0.2
      CALL SYMBOL(XL,YL,.1,'COMPONENT OF ELECTRIC FIELD',0.,27)
      ENCODE(40,900,PLTLBL) INCL,THETA
  900 FORMAT('INCL =',F6.2,' DEG    THETA =',F6.2,' DEG')
      XL=0.
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,PLTLBL.0.,40)
      ENCODE(40,910,PLTLBL) TALT,RALT
  910 FORMAT('TALT =',F6.2,' KM   RALT =',F6.2,' KM')
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,PLTLBL,0.,40)
      ENCODE(40,920,PLTLBL) RHO
  920 FORMAT('RANGE =',F7.2,' KM')
      YL=YL-.2
      CALL SYMBOL(XL,YL,.1,PLTLBL,0.,40)
C
      RETURN
      END
```

```
      SUBROUTINE SPLINE (X, Y, B. C, D, N)
      DIMENSION X(1), Y(1), B(1), C(1), D(1)
C
C   SPLINE DETERMINES THE COEFFICIENTS B. C, D OF A CUBIC SPLINE
C   INTERPOLATING THE GIVEN CURVE X(I),Y(I), I=1,...,N.
C   IF X(I) .LE. XX .LE. X(I+1) AND H = XX - X(I),
C   THEN THE INTERPOLATED VALUE AT XX IS
C   F(XX) = Y(I) + B(I)*H + C(I)*H**2 + D(I)*H**3.
C   THE INTERPOLATED VALUE CAN BE EVALUATED WITH THE FUNCTION SP EVAL.
C   B,C,D, MUST HAVE LENGTH AT LEAST N.
C
      IF (N.GT.2)  GO TO 050
      C(1) = 0.0
      D(1) = 0.0
      B(1) = (Y(2) - Y(1)) / (X(2) - X(1))
      RETURN
  050 NN = N - 1
      TB = 0.
      DO 100  I = 1,NN
      IF (X(I+1).LE.X(I))  GO TO 800
      D(I) = X(I+1) - X(I)
      TA = (Y(I+1) - Y(I)) / D(I)
      C(I) = TA - TB
      TB = TA
  100 CONTINUE
      C(1) = 0.
      C(N) = 0.
      TA = 0.
      TB = 0.
      DO 200  I = 2,NN
      C(I) = C(I) - TA * C(I-1)
      B(I) = 2.0 * (D(I) + D(I-1)) - TA * TB
      TB = D(I)
      TA = TB / B(I)
  200 CONTINUE
      C(NN) = C(NN) / B(NN)
      IF (NN.LT.3)  GO TO 350
      DO 300  I = 3,NN
      J = NN + 2 - I
  300 C(J) = (C(J) - D(J) * C(J+1)) / B(J)
  350 DO 400  I = 1,NN
      B(I) = (Y(I+1) - Y(I)) / D(I)
     $ - (C(I) + C(I) + C(I+1)) * D(I)
      D(I) = (C(I+1) - C(I)) / D(I)
      C(I) = 3.0 * C(I)
  400 CONTINUE
      RETURN
  800 PRINT 900
      PRINT 901, I,X(I),X(I+1)
      RETURN
  901 FORMAT (1X,'    I =',I5,'  X(I) =',1PE12.5,'  X(I+1) =',1PE12.5 /)
  900 FORMAT (' ERROR IN SPLINE',/,
     $ ' X-COORDINATE VALUES ARE NOT IN INCREASING ORDER')
      END
```

```
      SUBROUTINE FUNCVF(MD,LF)
C
      INCLUDE SPECANS.COMMONSPECS,LIST
C
      GO TO (30,40,50,60),LF
30    DO 35 I=1,NF
      IF(MODE(MD,I) .EQ. 0) GO TO 35
      JJ = I-KK(MD)+1
      YY(JJ) = XTRAR(MD,I)
      XX(JJ) = FREQ(I)
      LM = MODE(MD,I)
35    CONTINUE
      GO TO 99
40    DO 45 I=1,NF
      IF(MODE(MD,I) .EQ. 0) GO TO 45
      JJ = I-KK(MD)+1
      YY(JJ) = XTRAI(MD,I)
      XX(JJ) = FREQ(I)
      LM = MODE(MD,I)
45    CONTINUE
      GO TO 99
50    DO 55 I=1,NF
      IF(MODE(MD,I) .EQ. 0) GO TO 55
      JJ = I-KK(MD)+1
      YY(JJ) = STPR(MD,I)
      XX(JJ) = FREQ(I)
      LM = MODE(MD,I)
55    CONTINUE
      GO TO 99
60    DO 65 I=1,NF
      IF(MODE(MD,I) .EQ. 0) GO TO 65
      JJ = I-KK(MD)+1
      YY(JJ) = STPI(MD,I)
      XX(JJ) = FREQ(I)
      LM = MODE(MD,I)
65    CONTINUE
99    RETURN
      END
```

```
      SUBROUTINE MDHNKL (Z,H1,H2,H1PRME,H2PRME,THETA,IDBG)
      IMPLICIT COMPLEX*16 (A-H,O-Z)
      COMPLEX*16 I,MPOWER,MTERM
      REAL*8 A,B,C,D,CAP,PART1,PART2,ZMAG
      CHARACTER*4 IDBG
      DIMENSION A(30), B(30), C(30), D(30), CAP(30), PART1(2), PART2(2)
      EQUIVALENCE (PART1,TERM4), (PART2,SUM4)
      DATA A      9.3043671832929948190D-01,  3.1014557230974314911D+01,
     $            2.0676371457316209807D+02,  5.7434365242545027449D+02,
     $            8.7021765519097617734D+02,  8.2377871922864397320D+02,
     $            5.4168543740477229651 2D+02, 2.5794541638302022111D+02,
     $            9.3458415096831187422 1D+01, 2.6526351870714065662D+01,
     $            6.1210094300901072794D+00,  1.1592803844803233472D+00,
     $            1.8401275944172116610D-01,  2.4833030963741048003D-02,
     $            2.8842090097260218300D-03,  2.9133414235656786138D-04,
     $            2.5827494893312753446D-05,  2.0256858779853140063D-06,
     $            1.4155736366074870724D-07,  8.8690900130004431240D-09,
     $            5.0110220345327953880D-10,  2.5658074934115685526D-11,
     $            1.1961905405011228460D-12,  5.0988092481207283185D-14,
     $            1.3949332939651771634D-15,  7.1886100863126305797D-17,
     $            2.3938095525516785112D-18,  7.3893010891224645255D-20,
     $            2.1194208514407528762D-21,  5.6653258632471341093D-23/
      DATA B      6.7329872514427588456D-01,  1.1304978752404599303D+01,
     $            5.3833232154337609794D+01,  1.1962940478735024375D+02,
     $            1.5337103177645415841D+02,  1.2760919314887846509D+02,
     $            7.4742218215718409431D+01,  3.2355938021523117060D+01,
     $            1.0785312673841029091D+01,  2.8532573710320209050D+00,
     $            6.1360373035047223945D-01,  1.0937978109821251964D-01,
     $            1.6422959954865654465D-02,  2.1055051223957133911D-03,
     $            2.3316778704072130571D-04,  2.2528288509308256561D-05,
     $            1.9156708045016374595D-06,  1.4446989475979618839D-07,
     $            9.7286124416697709730D-09,  5.8854279743918795891D-10,
     $            3.2160808603234313644D-11,  1.5952782045255116351D-12,
     $            7.2151886229105003770D-14,  2.9876557444763976717D-15,
     $            1.1308553061173507190D-16,  3.9889659603766916030D-18,
     $            1.2946934700099855413D-19,  3.8925199340546098220D-21,
     $            1.0920223704314870626D-22,  2.8527230581567958120D-24/
      DATA C      4.6521335944546147241 0D-01, 6.2029114461938628922D+00,
     $            2.5845164339145262520D+01,  5.2213059311404570392D+01,
     $            6.2158103442146299012D+01,  4.8751689366390821897D+01,
     $            2.7084771870217123240D+01,  1.1215019407057400909D+01,
     $            3.5943537502539449002 2D+00, 9.1915008450416091470D-01,
     $            1.9128126034392533510D-01,  3.3127298697437809740D-02,
     $            4.8424410374250431140D-03,  6.0563309242456458321D-04,
     $            6.5550182039527768990D-05,  6.1959087743950508612D-06,
     $            5.1651963796625507119D-07,  3.8220198188402150980D-08,
     $            2.5278100653705126277D-09,  1.5033906103098380141D-10,
     $            8.0822936042484409157D-12,  3.9473961437101054471D-13,
     $            1.7590821906016512675D-14,  7.1914214762263778920D-16,
     $            2.6957287823672589641D-17,  9.3358572949515401565D-19,
     $            2.9922019906395981315D-20,  8.9015675760511320701D-22,
     $            2.4647128505125033375D-23,  6.3656020935361057409D-25/
      DATA D      6.7829872514427588456D-01,  4.5219915009618392131D+01,
     $            3.7683262508015326776D+02,  1.1962940478735024344D+03,
```

```
     $           1.993823413122504054BD+03,  2.044947090382055437SD+03,
     $           1.420102146096649609OD+03,  7.118306496735085746 3D+02,
     $           2.696326218160259743 2D+02,  7.999120647289658511 1D+01,
     $           1.902171582688013920 4D+01,  3.712810523330225668 2D+00,
     $           6.076437783230288572D-01,  8.422920469558285356 4 1D-02,
     $           1.002621498855101614 3D-02,  1.036301278403205802 1D-03,
     $           9.386786942050023543 2D-05,  7.512434527457401796 0D-06,
     $           5.350736842918377330 0D-07,  3.413546225147230162 8 5D-08,
     $           1.961803324797293193 5D-09,  1.020978056896327447 2D-10,
     $           4.834176377350035257 9D-12,  2.091359021133478372 3D-13,
     $           8.299013734606602939 0D-15,  3.031614149646268564 1D-16,
     $           1.022811791378633117 0D-17,  3.196786345924779236 4D-19,
     $           9.282190319177640015 3D-21,  2.510393299980430030 9D-22/
      DATA CAP / 1.041666666666666663D-01,  8.355034722222222211 6D-02,
     $           1.282265745563271601 9D-01,  2.918490264641404631 5D-01,
     $           8.816272674437576487 4D-01,  3.321408281862767526 4D+00,
     $           1.493579298092554540 D+01,  7.892391301158651753 0D+01,
     $           4.744515388682043168 7D+02,  3.207490090890661900 4D+03,
     $           2.408654964087400480 5D+04,  1.963231191695097912 1D+05,
     $           1.791902007775343806 3D+06,  1.744837718003412102 3D+07,
     $           1.837073796793307297 6D+08,  2.067904032945155150 8D+09,
     $           2.482751937592358984 72D+10,  3.169945498173488731 5D+11,
     $           4.277112696515471558 2D+12,  6.037113241139256074 4D+13,
     $           9.148694223435330679 2D+14,  1.441352517000935010 1D+16,
     $           2.378884439517575794 2D+17,  4.104608160094692188 5D+18,
     $           7.390004941570485309 3D+19,  1.385922000460394314 1D+21,
     $           2.703082593027576162 3D+22,  5.474747861964557333 5D+23,
     $           1.149893701438633352 4D+25.  2.501418069275360396 9D+26/
      DATA I/(0.D0,1.D0)/
      DATA ONE/(1.D0,0.D0)/,TWO/(2.D0,0.D0)/,ZERO/(0.D0,0.D0)/
      DATA ROOT3/(1.732050807568880D0,0.D0)/
      DATA ALPHA/(8.536672188389951D-1,0.D0)/
      DATA CONST1/( 2.588190451025220D-01,-9.659258262890670D-01)/
      DATA CONST2/( 2.588190451025220D-01, 9.659258262890670D-01)/
      DATA CONST3/(-9.659258262890670D-01, 2.588190451025220D-01)/
      DATA CONST4/(-9.659258262890670D-01,-2.588190451025220D-01)/
C
      ZPOWER=ONE
      SUM3=ZERO
      SUM4=ZERO
      ZMAG=CDABS(Z)
      IF(ZMAG .GT. 6.1D0) GO TO 70
      SUM1=ZERO
      SUM2=ZERO
      ZTERM=-Z**3/(200.D0,0.D0)
      DO 50 M=1,30
      SUM1=SUM1+DCMPLX(A(M),0.D0)*ZPOWER
      SUM2=SUM2+DCMPLX(B(M),0.D0)*ZPOWER
      SUM3=SUM3+DCMPLX(C(M),0.D0)*ZPOWER
      TERM4=DCMPLX(D(M),0.D0)*ZPOWER
      SUM4=SUM4+TERM4
      IF(DABS(PART1(1)/PART2(1)) .LE. 1.D-17 .AND.
     $   DABS(PART1(2)/PART2(2)) .LE. 1.D-17) GO TO 60
   50 ZPOWER=ZPOWER*ZTERM
   60 GM2F=I*(Z*SUM2-TWO*SUM1)/ROOT3
      GPMFP=I*(SUM4+TWO*Z*Z*SUM3)/ROOT3
      H1=Z*SUM2+GM2F
```

```
      H2=H1-TWO*GM2F
      H1PRME=SUM4+GPMFP
      H2PRME=H1PRME-TWO*GPMFP
      GO TO 999
70    MPOWER=ONE
      SUM1=ONE
      SUM2=ONE
      RTZ=CDSQRT(Z)
      SQRTZB=RTZ*Z
      ZTERM=I/SQRTZB
      MTERM=-ZTERM
      DM=ZERO
      TERM3=ONE
      DO 80 M=1,30
      ZPOWER=ZPOWER*ZTERM
      MPOWER=MPOWER*MTERM
      DM=DM+ONE
      TERM1=DCMPLX(CAP(M),0.D0)*ZPOWER
      TERM2=DCMPLX(CAP(M),0.D0)*MPOWER
      IF(CDABS(TERM2/TERM3) .GE. 1.D0) GO TO 81
      SUM1=SUM1+TERM1
      SUM2=SUM2+TERM2
      SUM3=SUM3+DM*TERM1
      TERM4=DM*TERM2
      SUM4=SUM4+TERM4
      IF(DABS(PART1(1)/PART2(1)) .LE. 1.D-17 .AND.
     $   DABS(PART1(2)/PART2(2)) .LE. 1.D-17) GO TO 81
80    TERM3=TERM2
81    ZTERM=(-1.5D0,0.D0)/Z
      SUM3=SUM3*ZTERM
      SUM4=SUM4*ZTERM
      TERM1=((-0.25D0,0.D0)-I*SQRTZB)/Z
      TERM2=((-0.25D0,0.D0)+I*SQRTZB)/Z
      EXP1=CDEXP((0.D0,0.6666666666666666667D0)*SQRTZB)
      EXP2=CONST1*EXP1
      EXP3=CONST2/EXP1
      EXP4=CONST3*EXP1
      EXP5=CONST4/EXP1
      ZTERM=ALPHA/CDSQRT(RTZ)
      TERM4=Z
      IF(PART1(1) .GE. 0.D0 .OR. PART1(2) .GE. 0.D0) GO TO 90
      H1=ZTERM*(EXP2*SUM2+EXP5*SUM1)
      H1PRME=ZTERM*(EXP2*(SUM2*TERM2+SUM4)+EXP5*(SUM1*TERM1+SUM3))
      GO TO 110
90    H1=ZTERM*EXP2*SUM2
      H1PRME=ZTERM*EXP2*(SUM2*TERM2+SUM4)
110   IF(PART1(1) .GE. 0.D0 .OR. PART1(2) .LT. 0.D0) GO TO 120
      H2=ZTERM*(EXP3*SUM1+EXP4*SUM2)
      H2PRME=ZTERM*(EXP3*(SUM1*TERM1+SUM3)+EXP4*(SUM2*TERM2+SUM4))
      GO TO 999
120   H2=ZTERM*EXP3*SUM1
      H2PRME=ZTERM*EXP3*(SUM1*TERM1+SUM3)
C CALCULATE WRONSKIAN AS PARTIAL CHECK ON VALIDITY
999   SUM4=H1*H2PRME-H1PRME*H2
      IF(DABS(PART2(1)) .LE. 1.D-8 .AND.
     $   DABS(PART2(2)+1.457495441040461D0) .LE. 1.D-8) GO TO 1000
      PRINT 1001,SUM4,THETA,IDBG
```

```
 1000  RETURN
 1001  FORMAT(' ***** POSSIBLE ERROR IN MDHNKL:  W = ',1P2E15.6,
     $       ' FOR THETA = ',0P2F10.4,' AT ',A4)
       END
```

```
      SUBROUTINE BORDER(XLNG,XMIN,XMAX,XINC,NX,YLNG,YMIN,YMAX,YINC,NY)
      DIMENSION XINC(NX),YINC(NY)
      LOGICAL FY,FX
      FX=.FALSE.
      FY=.FALSE.
      IF(NX .EQ. 1) FX=.TRUE.
      IF(NY .EQ. 1) FY=.TRUE.
      XT=XLNG-.1
      YT=YLNG-.1
      XSCALE=XLNG/(XMAX-XMIN)
      YSCALE=YLNG/(YMAX-YMIN)
      YM=ABS(YMIN)
      YLN=-.4
      IF(YM .GE. 10.) YLN=YLN-.1
      IF(YM .GE. 100.) YLN=YLN-.1
      IF(YM .GE. 1000.) YLN=YLN-.1
      IF(YMIN .LT. 0.) YLN=YLN-.1
      YM=ABS(YMAX)
      YLM=-.4
      IF(YM .GE. 10.) YLM=YLM-.1
      IF(YM .GE. 100.) YLM=YLM-.1
      IF(YM .GE. 1000.) YLM=YLM-.1
      IF(YMAX .LT. 0.) YLM=YLM-.1
      XM=ABS(XMAX)
      XLM=-.3
      IF(XM .GE. 10.) XLM=XLM-.1
      IF(XM .GE. 100.) XLM=XLM-.1
      IF(XM .GE. 1000.) XLM=XLM-.1
      IF(XMAX .LT. 0.) XLM=XLM-.1
      IF(FX) DX=XINC(1)
      IF(FY) DY=YINC(1)
      IY=1
      YL=0.
      CALL NUMBER(YLN,0.,.1,YMIN,0.,1)
      CALL PLOT(0.,0.,3)
      IF(FY) GO TO 110
   10 YP=(YINC(IY)-YMIN)*YSCALE
      GO TO 111
  110 YL=YL+DY
      YP=YL*YSCALE
  111 IF(YP .LT.   0.) GO TO 99
      IF(YP .GE. YLNG) GO TO 11
      CALL PLOT(0.,YP,2)
      CALL PLOT(.1,YP,2)
      CALL PLOT(0.,YP,2)
      IF(FY) GO TO 110
      IY=IY+1
      IF(IY .LE. NY) GO TO 10
   11 CALL PLOT(0.,YLNG,2)
      CALL NUMBER(YLM,YLNG-.1,.1,YMAX,0.,1)
      CALL PLOT(0.,YLNG,3)
      IX=1
      XL=0.
      IF(FX) GO TO 112
```

```
12    XP=(XINC(IX)-XMIN)*XSCALE
      GO TO 120
112   XL=XL+DX
      XP=XL*XSCALE
120   IF(XP .LT.   0.) GO TO 99
      IF(XP .GE. XLNG) GO TO 13
      CALL PLOT(XP,YLNG,2)
      CALL PLOT(XP,YT,2)
      CALL PLOT(XP,YLNG,2)
      IF(FX) GO TO 112
      IX=IX+1
      IF(IX .LE. NX) GO TO 12
13    CALL PLOT(XLNG,YLNG,2)
      IF(FY) GO TO 130
113   IY=IY-1
      IF(IY .LE. 0) GO TO 15
      YP=(YINC(IY)-YMIN)*YSCALE
      GO TO 14
130   YL=YL-DY
      YP=YL*YSCALE
      IF(YP .LE. 0.) GO TO 15
14    CALL PLOT(XLNG,YP,2)
      CALL PLOT(XT,YP,2)
      CALL PLOT(XLNG,YP,2)
      IF(FY) GO TO 130
      GO TO 113
15    CALL PLOT(XLNG,0.,2)
      CALL NUMBER(XLNG+XLM,-.2,.1,XMAX,0.,1)
      CALL PLOT(XLNG,0.,3)
      IF(FX) GO TO 150
115   IX=IX-1
      IF(IX .LE. 0) GO TO 17
      XP=(XINC(IX)-XMIN)*XSCALE
      GO TO 16
150   XL=XL-DX
      XP=XL*XSCALE
      IF(XP .LE. 0.) GO TO 17
16    CALL PLOT(XP,0.,2)
      CALL PLOT(XP,.1,2)
      CALL PLOT(XP,0.,2)
      IF(FX) GO TO 150
      GO TO 115
17    CALL PLOT(0.,0.,2)
      CALL NUMBER(0.,-.2,.1,XMIN,0.,1)
      RETURN
99    PRINT 100,XLNG,XMIN,XMAX,XINC(1),NX,YLNG,YMIN,YMAX,YINC(1),NY
100   FORMAT('0***  ERROR IN BORDER:  XLNG, XMIN, XMAX, XINC(1), NX =',
     $       1P4E15.5,I5/24X,'YLNG, YMIN, YMAX, YINC(1), NY =',1P4E15.5,
     $       I5/'0***')
      CALL PLOT(0.,0.,999)
      STOP
      END
```

```
      SUBROUTINE CURVE(X,Y,UP,NRPTS,XMIN,YMIN,XINC,YINC,LINE)
C
C X,Y,UP MUST BE DIMENSIONED AT LEAST NRPTS
C XMIN,YMIN ARE X,Y ORIGIN IN USER UNITS
C XINC,YINC ARE X,Y SCALES IN USER UNITS PER INCH
C
C LINE=1:  SOLID
C      2:  LONG DASH
C      3:  MEDIUM DASH
C      4:  SHORT DASH
C      5:  DOTTED
C      6:  SHORT + LONG DASH
C      7:  SHORT + SHORT + LONG DASH
C
      LOGICAL UP,UP1,UP2
      DIMENSION IPEN(10),JOC(7),X(NRPTS),Y(NRPTS),UP(NRPTS)
      DATA IPEN/3,2,3,2,3,2,2,2,2,2/,JOC/18, 61, 56, 54, 52, 11, 36/
      DATA DELR/.1/
C
      IF(NRPTS .LE. 1) GO TO 99
C
      IF(LINE) 1,2,3
1     KK=MOD(LINE,7)+7
      GO TO 4
2     KK=0
      GO TO 4
3     KK=MOD(LINE,7)
4     KK=KK+1
      JO=JOC(KK)/10
      JC=JOC(KK)-10*JO
C
      J=1
      IP=2
      IF(KK .EQ. 6) IP=3
      DR=0.
      RHO1=0.
      RHO2=DELR
      PX1=(X(1)-XMIN)/XINC
      PY1=(Y(1)-YMIN)/YINC
      UP1=UP(1)
      IF(UP1) GO TO 10
C
C GO TO FIRST POSITION WITH PEN UP
      CALL PLOT(PX1,PY1,3)
C
10    DO 40 I=2,NRPTS
      PX2=(X(I)-XMIN)/XINC
      PY2=(Y(I)-YMIN)/YINC
      UP2=UP(I)
      IF(UP2) GO TO 22
      IF(UP1) GO TO 37
      IF(KK .EQ. 2) GO TO 38
      DELX=PX2-PX1
      DELY=PY2-PY1
```

```
      RHO=SQRT(DELX**2+DELY**2)
      RHO1=RHO1+RHO
      IF(RHO2 .GT. RHO1) GO TO 38
      DELX=DELX*DELR/RHO
      DELY=DELY*DELR/RHO
      DX 6=DELX*.1
      DY 6=DELY*.1
      IF(DR .EQ. 0.) GO TO 20
      DX=DELX*DR/DELR
      DY=DELY*DR/DELR
      PX1=PX1+DX
      PY1=PY1+DY
      GO TO 21
20    IF(RHO2 .GT. RHO1) GO TO 38
      PX1=PX1+DELX
      PY1=PY1+DELY
21    CALL PLOT(PX1,PY1,IP)
      IF(KK .EQ. 6) CALL PLOT(PX1+DX6,PY1+DY6,2)
      J=J+1
      IP=IPEN(JD+MOD(J,JC))
      RHO2=RHO2+DELR
      GO TO 20
22    DR=0.
      RHO1=0.
      RHO2=DELR
      GO TO 39
C PEN HAS BEEN UP, PREPARE TO LOWER PEN
37    CALL PLOT(PX2,PY2,3)
      GO TO 39
38    CALL PLOT(PX2,PY2,IP)
      DR=RHO2-RHO1
39    PX1=PX2
      PY1=PY2
      UP1=UP2
40    CONTINUE
99    RETURN
      END
```

```
COMMONSPECS PROC
        PARAMETER NRFREQ=6,NRMODE=23
        COMMON/ONE/FREQ(NRFREQ),
       $        XTRAR(NRMODE,NRFREQ),XTRAI(NRMODE,NRFREQ),
       $        STPP(NRMODE,NRFREQ),STPI(NRMODE,NRFREQ),
       $        MODE(NRMODE,NRFREQ),KK(NRMODE),NF,NM,LM
        COMMON/TWO/XX(NRFREQ),YY(NRFREQ),B(NRFREQ),C(NRFREQ),D(NRFREQ),
       $        YC(4,NRMODE,NRFREQ),BC(4,NRMODE,NRFREQ),
       $        CC(4,NRMODE,NRFREQ),DC(4,NRMODE,NRFREQ)
C
        END
```

DISTRIBUTION LIST

DEPARTMENT OF DEFENSE
DEPUTY UNDER SECRETARY OF DEFENSE
CMD, CONT, COMM & INTELL
DEPARTMENT OF DEFENSE
WASHINGTON, DC 20301

DIRECTOR
COMMAND CONTROL TECHNICAL CENTER
11440 ISAAC NEWTON SQUARE, N
RESTON, VA 22091
C-650

DIRECTOR
COMMAND CONTROL TECHNICAL CENTER
ROOM ME682, THE PENTAGON
WASHINGTON, DC 20301
C-312

DIRECTOR
DEFENSE ADVANCED RESEARCH PROJECT AGENCY
1440 WILSON BLVD
ARLINGTON, VA 22209
NUCLEAR MONITORING RSCH
STRATEGIC TECH OFFICE

DEFENSE COMMUNICATION ENGINEERING CENTER
1860 WIEHLE AVENUE
RESTON, VA 22090
CODE R220 (M HOROWITZ)
CODE R410
CODE R103

DIRECTOR
DEFENSE COMMUNICATIONS AGENCY
WASHINGTON, DC 20305
CODE 810
CODE 480
CODE 101B

DEFENSE COMMUNICATIONS AGENCY
WWMCCS SYSTEM ENGINEERING ORG
WASHINGTON, DC 20305
RL CRAWFORD

DEFENSE TECHNICAL INFORMATION CENTER
CAMERON STATION
ALEXANDRIA, VA 22314

DIRECTOR
DEFENSE INTELLIGENCE AGENCY
WASHINGTON, DC 20301
DIAST-5
DB-4C (EDWARD O'FARRELL)

DIRECTOR
DEFENSE NUCLEAR AGENCY
WASHINGTON, DC 20305
DDST
TITL TECH LIBRARY
RAAE
STVL

DIRECTOR
JOINT STRAT TGT PLANNING STAFF JCS
OFFUTT AFB
OMAHA, NB 68113
JPST

COMMANDER
FIELD COMMAND
DEFENSE NUCLEAR AGENCY
KIRTLAND AFB, NM 87115
FCPR

DIRECTOR INTERSERVICE NUCLEAR WEAPONS SCHOOL
KIRTLAND AFB, NM 87115
DOCUMENT CONTROL

CHIEF
LIVERMORE DIVISION FLD COMMAND DNA
LAWRENCE LIVERMORE LABORATORY
PO BOX 808
LIVERMORE, CA 94550
FCPRL

DIRECTOR
NATIONAL SECURITY AGENCY
FT GEORGE G MEADE, MD 20755
W65
OLIVER H BARTLETT W32
TECHNICAL LIBRARY
JOHN SKILLMAN R52

OJCS/J-3
THE PENTAGON
WASHINGTON, DC 20301
OPERATIONS (WWMCCS EVAL OFF, MR TOMA)

OJCS/J-5
THE PENTAGON
WASHINGTON, DC 20301
PLANS & POLICY (NUCLEAR DIVISION)

UNDER SECRY OF DEFENSE FOR RESEARCH AND ENGINEERING
DEPARTMENT OF DEFENSE
WASHINGTON, DC 20301
S&SS (OS)

DEPARTMENT OF THE ARMY
COMMANDER/DIRECTOR
ATMOSPHERIC SCIENCES LABORATORY
US ARMY ELECTRONICS COMMAND
WHITE SANDS MISSILE RANGE, NM 88002
DELAS-AE-M (FE NILES)

COMMANDER
HARRY DIAMOND LABORATORIES
2800 POWDER MILL RD
ADELPHI, MD 20783
DELHD-NP (FN WIMENITZ)
MILDRED H WEINER DRXDO-II

COMMANDER
US ARMY FOREIGN SCIENCE & TECH CENTER
220 7TH STREET, NE
CHALOTTESVILLE, VA 22901
R JONES
PA CROWLEY

COMMANDER
US ARMY NUCLEAR AGENCY
7500 BACKLICK ROAD
BUILDING 2073
SPRINGFIELD, VA 22150
MONA-WE (J BERBERET)

CHIEF
US ARMY RESEARCH OFFICE
PO BOX 12211
TRIANGEL PARK, NC 27709
DRXRD-ZC

DEPARTMENT OF THE NAVY
CHIEF OF NAVAL OPERATIONS
NAVY DEPARTMENT
WASHINGTON, DC 20350
OP 941
OP-604C3
OP 943
OP 981

CHIEF OF NAVAL RESEARCH
NAVY DEPARTMENT
ARLINGTON, VA 22217
CODE 402
CODE 420
CODE 421
CODE 461
CODE 464

COMMANDING OFFICER
NAVAL INTELLIGENCE SUPPORT CENTER
4301 SUITLAND RD BLDG 5
WASHINGTON, DC 20390

OFFICER-IN-CHARGE
WHITE OAK LABORATORY
NAVAL SURFACE WEAPONS CENTER
SILVER SPRING, MD 20910
CODE WA501 NAVY NUC PRGMS OFF
CODE WX21 TECH LIBRARY

COMMANDER
NAVAL TELECOMMUNICATIONS COMMAND
NAVTELCOM HEADQUARTERS
4401 MASSACHUSETTS AVE, NW
WASHINGTON, DC 20390
CODE 24C

COMMANDING OFFICER
NAVY UNDERWATER SOUND LABORATORY
FORT TRUMBULL
NEW LONDON, CT 06320
PETER BANNISTER
DA MILLER

DIRECTOR
STRATEGIC SYSTEMS PROJECT OFFICE
NAVY DEPARTMENT
WASHINGTON, DC 20376
NSP-2141

DEPARTMENT OF THE AIR FORCE
COMMANDER
ADC/DC
ENT AFB, CO 80912
DC (MR LONG)

COMMANDER
ADCOM/XPD
ENT AFB, CO 80912
XPQDQ
XP

AF GEOPHYSICS LABORATORY, AFSC
HASCOM AFB, MA 01731
CRU (S HOROWITZ)

AF WEAPONS LABORATORY, AFSC
KIRTLAND AFB, NM 87117
SUL (2)
DYC

COMMANDER
ROME AIR DEVELOPMENT CENTER, AFSC
GRIFFISS AFB, NY 13440
EMTLD DOC LIBRARY

COMMANDER
ROME AIR DEVELOPMENT CENTER, AFSC
HANSCOM AFB, MA 01731
EEP JOHN RASMUSSEN

COMMANDER IN CHIEF
STRATEGIC AIR COMMAND
OFFUTT AFB, NB 68113
NRT
XPFS

LAWRENCE LIVERMORE NATIONAL
LABORATORY
PO BOX 808
LIVERMORE, CA 94550
TECH INFO DEPT L-3

LOS ALAMOS NATIONAL SCIENTIFIC
LABORATORY
PO BOX 1663
LOS ALAMOS, NM 87545
DOC CON FOR T F TASCHEK
DOC CON FOR D R WESTERVELT
DOC CON FOR P W KEATON

SANDIA LABORATORY
LIVERMORE NATIONAL LABORATORY
PO BOX 969
LIVERMORE, CA 94550
DOC CON FOR B E MURPHEY
DOC CON FOR T B COOK ORG 8000

SANDIA NATIONAL LABORATORY
PO BOX 5800
ALBUQUERQUE, NM 87115
DOC CON FOR SPACE PROJ DIV
DOC CON FOR A D THORNBROUGH
DOC CON FOR 3141 SANDIA RPT COLL

OTHER GOVERNMENT

DEPARTMENT OF COMMERCE
NATIONAL BUREAU OF STANDARDS
WASHINGTON, DC 20234
RAYMOND T MOORE

DEPARTMENT OF COMMERCE
OFFICE OF TELECOMMUNICATIONS
INSTITUTE FOR TELECOM SCIENCE
BOULDER, CO 80302
WILLIAM F UTLAUT
L A BERRY
A GLENN JEAN

DEPARTMENT OF DEFENSE CONTRACTORS

AEROSPACE CORPORATION
PO BOX 92957
LOS ANGELES, CA 90009
IRVING M GARFUNKEL

ANALYTICAL SYSTEMS ENGINEERING CORP
5 OLD CONCORD RD
BURLINGTON, MA 01803
RADIO SCIENCES

THE BOEING COMPANY
PO BOX 3707
SEATTLE, WA 98124
GLENN A HALL
J F KENNEY

ESL, INC
495 JAVA DRIVE
SUNNYVALE, CA 94086
JAMES MARSHALL

GENERAL ELECTRIC COMPANY
SPACE DIVISION
VALLEY FORGE SPACE CENTER
GODDARD BLVD KING OF PRUSSIA
PO BOX 8555
PHILADELPHIA, PA 19101
SPACE SCIENCE LAB (MH BORTNER)

KAMAN TEMPO
816 STATE STREET
PO DRAWER QQ
SANTA BARBARA, CA 93102
B GAMBILL
DASIAC
WARREN S KNAPP

GTE SYLVANIA, INC
ELECTRONICS SYSTEMS GRP
EASTERN DIVISION
77 A STREET
NEEDHAM, MA 02194
MARSHAL CROSS

ITT RESEARCH INSTITUTE
10 WEST 35TH STREET
CHICAGO, IL 60616
TECHNICAL LIBRARY

UNIVERSITY OF ILLINOIS
DEPARTMENT OF ELECTRICAL ENGINEERING
URBANA, IL 61803
AERONOMY LABORATORY

JOHNS HOPKINS UNIVERSITY
APPLIED PHYSICS LABORATORY
JOHNS HOPKINS ROAD
LAUREL, MD 20810
J NEWLAND
PT KOMISKE

LOCKHEED MISSILE & SPACE CO, INC.
3251 HANOVER STREET
PALO ALTO, CA 94304
E E GAINES
W L IMHOF D/52-12
J B REAGAN D652-12
R G JOHNSON D/52-12

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LINCOLN LABORATORY
PO BOX 73
LEXINGTON, MA 02173
D M TOWLE

MISSION RESEARCH CORPORATION
735 STATE STREET
SANTA BARBARA, CA 93101
R HENDRICK
F FAJEN

MITRE CORPORATION
PO BOX 209
BEDFORD, MA 01730
G HARDING

PACIFIC-SIERRA RESEARCH CORP
1456 CLOVERFIELD BLVD
SANTA MONICA, CA 90404
E C FIELD, JR

PENNSYLVANIA STATE UNIVERSITY
IONOSPHERIC RESEARCH LABORATORY
318 ELECTRICAL ENGINEERING EAST
UNIVERSITY PARK, PA 16802
IONOSPHERIC RSCH LAB

R&D ASSOCIATES
PO BOX 9695
MARINA DEL REY, CA 90291
FORREST GILMORE
WILLIAM J KARZAS
PHYLLIS GREIFINGER
CARL GREIFINGER
H A ORY
BRYAN GABBARD
R P TURCO
SAUL ALTSCHULER

RAND CORPORATION
1700 MAIN STREET
SANTA MONICA, CA 90406
TECHNICAL LIBRARY
CULLEN CRAIN

SRI INTERNATIONAL
333 RAVENSWOOD AVENUE
MENLO PARK, CA 94025
DONALD NEILSON
GEORGE CARPENTER
W G CHETNUT
J R PETERSON
GARY PRICE

STANFORD UNIVERSITY
RADIO SCIENCE LABORATORY
STANFORD, CA 94305
R A HELLIWELL
FRASER SMITH
J KATSUFRAKIS

CALIFORNIA INSTITUTE OF TECHNOLOGY
JET PROPULSION LABORATORY
4800 OAK GROVE DRIVE
PASADENA, CA 91103
ERNEST K SMITH