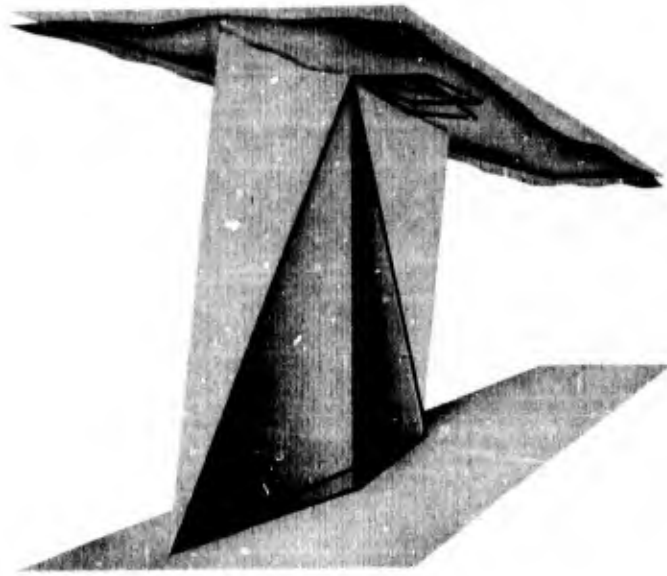


AD 713168

A FORTRAN PROGRAM FOR WAVEGUIDE PROPAGATION WHICH ALLOWS FOR BOTH VERTICAL AND HORIZONTAL DIPOLE EXCITATION

Interim Report No. 702

R.A. Pappert, W.F. Moler, L.R. Shockey



NAVAL ELECTRONICS LABORATORY CENTER
SAN DIEGO, CALIFORNIA 92152

15 June 1970

Prepared for DEFENSE ATOMIC SUPPORT AGENCY
on DASA Subtask HB 042

NELC Problem M402

This document has been approved
for public release and sale; its
distribution is unlimited.

DDC
RECEIVED
OCT 22 1970
RECEIVED

TABLE OF CONTENTS

Abstract.....ii
Introduction..... 1
Excitation Factors..... 3
Running the Program..... 9
References.....23
Appendix A: Program Listing.....24

FIGURES AND TABLES

Table 1: Sample Input for Horizontal Dipole.....16
Table 2: Sample Output for Horizontal Dipole.....18
Table 3: Sample Input for Vertical Dipole.....20
Table 4: Sample Output for Vertical Dipole.....22

ABSTRACT

This report presents an updated version of an earlier waveguide program written in the FORTRAN compiler language. The new program includes the following additions; (1) Provision for calculating excitation factors for all electric field components E_x , E_y , and E_z as a function of height within the guide; (2) Provision for generating the excitation factors for both vertical and horizontal electric dipole exciters.

I. INTRODUCTION:

This report is a continuation of a series (1, 2, 3, 4, 5, 6, 7, 8) which describes computer programs which have been developed for calculating ionospheric reflection coefficients and/or waveguide modal parameters. Reports 1 through 7 were written in the NELIAC computer compiler language for the CDC 1604, while 8 and the present program are written in FORTRAN 63 for the CDC machine.

As with earlier versions of the program, the modal equation is determined from the reflection matrix of the ionosphere as seen looking upward from a position in the guide and from the ground reflection matrix as seen looking downward from the same position in the guide. Solutions of the modal equation are obtained by application of Newton's method of iteration. The propagation parameters, attenuation rate, phase velocity, and excitation factor are determined for each solution of the modal equation. Full allowance is made for ionospheric inhomogeneity in the vertical direction, earth curvature in the direction of propagation, and the orientation and intensity of the geomagnetic field with respect to the path of propagation. As many as five charged particles of either sign and of arbitrary mass may be included as constituents.

The present program differs from that of reference 8 only to the extent that additions have been incorporated into the program described here which allow for greater flexibility in calculating the excitation factors (and height-gain functions). In previous versions of the program the excitation factor calculated has been for ground based vertical electric dipole excitation of the vertical component of the electric field, E_z , at the ground. In contrast, the present program can be used to generate excitation factors for all electric field components E_x , E_y , and E_z as a function of height within the guide. Furthermore, the excitation factors can be generated for both vertical and horizontal electric dipole exciters with allowance for arbitrary height of the exciters within the guide. Thus, problems relevant to elevated sources and receivers can be treated with the present program.

The physical input parameters for the waveguide program are:

1. The azimuth of propagation with respect to the horizontal component of the geomagnetic field.
2. The geomagnetic dip angle and field strength.
3. The radio frequencies.
4. Vertical profiles of up to five charged species (electron, positive and negative ions).
5. An exponential but otherwise arbitrary collision frequency distribution with height for each species of charged particles.
6. The ground conductivity and dielectric constant.
7. Transmitter and receiver altitudes (up to 25 each).
8. Horizontal dipole orientation relative to direction of propagation.

II. EXCITATION FACTORS:

In this section we summarize the excitation factor formulas which have been programmed. The formulas for the vertical dipole source are straightforward extensions of Budden's (reference 9) fundamental work, while the formulas for the horizontal dipole follow from the developments in reference 6.

VERTICAL DIPOLE EXCITATION

Field
Component

Excitation Factor

$$E_z \quad B_1 \frac{(1 + \bar{R}_0)^2 (1 - \bar{R}_1 R_1)}{\bar{R}_0} f_0(z_1) f_0(z)$$

$$E_x \quad \frac{B_1}{S} \frac{(1 + \bar{R}_0)^2 (1 - \bar{R}_1 R_1)}{\bar{R}_0} f_0(z_1) g(z)$$

$$E_y \quad -\frac{B_1}{S} \bar{R}_1 (1 + \bar{R}_0) (1 + \bar{R}_1) f_0(z_1) f_1(z)$$

HORIZONTAL DIPOLE EXCITATION

Field Component	Excitation Factor
E_z	$B_2 \left[\sin \psi \, {}_1R_{ }(1 + {}_1\bar{R}_1)(1 + {}_{ }\bar{R}_{ }) f_1(z_1) f_{ }(z) \right. \\ \left. + \cos \psi \frac{(1 + {}_{ }\bar{R}_{ })^2}{{}_{ }\bar{R}_{ }} (1 - {}_1\bar{R}_1 R_1) g(z_1) f_{ }(z) \right]$
E_x	$\frac{B_2}{S} \left[\sin \psi \, {}_1R_{ }(1 + {}_1\bar{R}_1)(1 + {}_{ }\bar{R}_{ }) f_1(z_1) g(z) \right. \\ \left. + \cos \psi \frac{(1 + {}_{ }\bar{R}_{ })^2}{{}_{ }\bar{R}_{ }} (1 - {}_1\bar{R}_1 R_1) g(z_1) g(z) \right]$
E_y	$-\frac{B_2}{S} \left[\sin \psi \frac{(1 + {}_1\bar{R}_1)^2}{{}_1\bar{R}_1} (1 - {}_{ }\bar{R}_{ } R_{ }) f_1(z_1) f_1(z) \right. \\ \left. + \cos \psi \, {}_{ }R_1(1 + {}_{ }\bar{R}_{ })(1 + {}_1\bar{R}_1) g(z_1) f_1(z) \right]$

The R and \bar{R} 's represent respectively elements of the reflection matrix looking into the ionosphere and towards the ground from the same level d within the guide. Consistent with the usual notation, the first subscript refers to the polarization of the incident wave while the second applies to the polarization of the reflected wave. B_1 and B_2 are given by

$$B_1 = \frac{S^{5/2}}{\frac{\partial F}{\partial \theta} \Big|_{\theta = \theta_n}} \quad ; \quad B_2 = -\frac{1 S^{3/2}}{\frac{\partial F}{\partial \theta} \Big|_{\theta = \theta_n}} \quad (1)$$

where S is the sine of the eigenangle, $i = \sqrt{-1}$ and the denominator is the derivative of the modal equation evaluated at the eigenangle, θ_n .

In the case of the horizontal dipole, ψ is the angle between the direction of the horizontal dipole and the direction of propagation.

Explicitly, $\psi = 0$ represents end fire while $\psi = \frac{\pi}{2}$ represents broadside launching. z_1 denotes the altitude of the transmitter while z is the altitude of the receiver. The height-gain functions f_{\parallel} , f_{\perp} and g are given by

$$f_{\parallel}(z) = \exp\left[\frac{z-d}{a}\right] \frac{F_1 h_1(q) + F_2 h_2(q)}{F_1 h_1(q_d) + F_2 h_2(q_d)} \quad (2)$$

$$f_{\perp}(z) = \frac{F_3 h_1(q) + F_4 h_2(q)}{F_3 h_1(q_d) + F_4 h_2(q_d)} \quad (3)$$

$$g = \frac{1}{ik} \frac{d}{dz}(f_{\parallel}) \quad (4)$$

$$F_1 = -\left\{ H_2(q_0) - i \frac{n_0^2}{N_g^2} \left(\frac{ak}{2}\right)^{1/3} (N_g^2 - S^2)^{1/2} h_2(q_0) \right\} \quad (5)$$

$$F_2 = H_1(q_0) - i \frac{n_0^2}{N_g^2} \left(\frac{ak}{2}\right)^{1/3} (N_g^2 - S^2)^{1/2} h_1(q_0) \quad (6)$$

$$F_3 = -\left\{ h_2'(q_0) - i \left(\frac{ak}{2}\right)^{1/3} (N_g^2 - S^2)^{1/2} h_2(q_0) \right\} \quad (7)$$

$$F_4 = h_1'(q_0) - i \left(\frac{ak}{2}\right)^{1/3} (N_g^2 - S^2)^{1/2} h_1(q_0) \quad (8)$$

$$q = \left(\frac{2}{ak}\right)^{-2/3} \left(C^2 - \frac{2}{a}(h-z)\right) \quad (9)$$

$$H_j(q) = h_j'(q) + \frac{1}{2} \left(\frac{2}{ak}\right)^{2/3} h_j(q); \quad j = 1, 2 \quad (10)$$

$$n^2 = 1 - \frac{2}{a}(h-z) \quad (11)$$

$$N_g^2 = \frac{\epsilon}{\epsilon_0} - i \frac{\sigma}{\omega \epsilon_0} \quad (12)$$

C = cosine of the angle of incidence at height h

k is the free space wave number

ϵ/ϵ_0 is the dielectric constant of the ground

σ is the ground conductivity

ω is the circular radio frequency

a is the earth's radius

The functions h_1 and h_2 are modified Hankel functions of order $1/3$ (which are linearly related to Airy functions) as defined by the Computation Laboratory at Cambridge, Massachusetts (reference 10) and the primes on these quantities denote derivatives with respect to the argument. Equation (11) is the modified refractive index which equals unity at height, h . The subscript, o , which appears on n^2 in Equations (5) and (6) signifies that Equation (11) is to be evaluated for $z = 0$. Similarly, the subscripts o and d which appear on q in Equations (2), (3) and (5) through (8) signify that Equation (9) is to be evaluated for $z=0$ and d respectively.

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 indeed justified, to use the flat earth analogues of Equations (2) thru (4). That is, to replace the height gains by

$$f_{||}(z) = \frac{\exp(ikCz) + {}_{||}\bar{R}_{||} \exp(-ikCz + 2ikCd)}{\exp(ikCd)(1 + {}_{||}\bar{R}_{||})} \quad (13)$$

$$f_{\perp}(z) = \frac{\exp(ikCz) + {}_{\perp}\bar{R}_{\perp} \exp(-ikCz + 2ikCd)}{\exp(ikCd)(1 + {}_{\perp}\bar{R}_{\perp})} \quad (14)$$

$$g = \frac{C [\exp(ikCz) - {}_{||}\bar{R}_{||} \exp(-ikCz + 2ikCd)]}{\exp(ikCd)(1 + {}_{||}\bar{R}_{||})} \quad (15)$$

When the absolute value of the imaginary part of the eigenangle exceeds 10° , the height-gain functions will be computed according to Equations (13), (14) and (15).

Our defining equations for the excitation factors are somewhat unconventional since the term excitation factor is generally reserved for the expressions defined by us when evaluated for $z_1 = z = 0$ (i.e. for a ground based transmitter and receiver) with the height variations relegated to height-gain terms. However, because of the inclusion of the height-gain terms in our definition of the excitation factors it becomes a simple matter to use them in mode sum calculations. Thus to utilize the excitation factors as defined here in a mode sum calculation with results expressed in the usual units of dB above a microvolt per meter per kilowatt radiated power and with geometric spreading allowed for by a $[\sin(\frac{\theta}{2})]^{-2}$ factor, where θ is the transmitter receiver distance, the factor B_2 should be multiplied by $(-i)$ and in addition the factors B_1 and B_2 should be multiplied by

$$\frac{0.03248 k}{\sqrt{f}}$$

with f the frequency in kHz and k expressed in inverse kilometers. It should be cautioned that the preceding statements apply for mode sum calculations in a guide which is homogeneous in the direction of propagation. In general, the excitation factors defined here cannot be used directly for WKB calculations.

The magnitudes of the quantities defined here as excitation factors are printed out under the label EXTRA MAG, while the arguments are printed out under the label EXTRA ANGLE. Each printout is accompanied by a legend which clearly indicates the applicable dipole exciter (horizontal or vertical) and field component (E_z , E_x , or E_y). The legend also gives transmitter and receiver altitudes and, in the case of horizontal dipole excitation, orientation.

For the purpose of normalizing in a consistent manner with previous works (Wait, pg. 221, reference 11) the quantities obtained by multiplying the excitation factors by $\frac{-ikh}{2}$ are printed out under the labels PHASE OF WAITS EXCITATION FACTOR and WAITS EXCITATION FACTOR IN DB. Here H (labeled REFLHT in the program) should be a height representative of the height where the bulk of the reflection occurs. Again the legend clearly indicates the applicable exciter and field component. Transmitter and receiver altitudes and orientation of the horizontal dipole are also given.

Also printed out are the height-gain functions shown below with these labels

<u>Function</u>	<u>Label</u>
$f_{ }(z_1)$	HGTVR
$g(z_1)$	HGTHE
$f_{\perp}(z_1)$	HGTHB
$f_{ }(z)$	HGRZ
$g(z)$	HGRX
$f_{\perp}(z)$	HGRY

As before z_1 and z denote the altitude of the transmitter and receiver respectively. When the program is run with the integration carried to zero (i.e. $d = 0$) the functions $f_{||}(z)$ and $f_{\perp}(z)$ will be unity at $z = 0$ so that they then agree with the usual definition of height-gain functions. However, g as defined by us here will not be unity at the ground but will differ from its customary definition by the factor

$$\frac{(1 + {}_oR_{||})_o}{(1 - {}_oR_{||})_o} C_n$$

where the subscripts o indicate that the reflection coefficients are to be evaluated at the ground and C_n is the cosine of the eigenangle θ_n .

III. RUNNING THE PROGRAM:

A. Options Available

The program allows for several input, output, or computational procedure options designed for specific user needs. They are:

1. Computation of either exact or approximate waveguide solutions.
2. Ionospheric constitution of from one to five particle species of arbitrary charge, mass, and collision frequency.
3. Calculation of excitation factors for both horizontal and vertical dipole radiation or for ground based vertical dipoles only.
4. Calculation of both horizontally and vertically polarized waveguide modes or either polarization only.
5. The charged particle-neutral species collision frequencies as a function of height can be input in tabular or in functional form.

The various options are demonstrated in Tables 1-4 and will be discussed in the following descriptions of the inputs and printouts of the program.

B. Input

1. Charged particle height-density profiles.

Tables 1 and 3 demonstrate the order and manner of profile input. Although only two species are shown three are implied. Space charge neutrality is assumed and by including a NEGIONS card the negative ion density is taken to be the number density difference between the positive ion and electron densities. If more than one positive or negative ion species is included the NEGIONS card can not be used.

Species and run identification cards as well as the nines (first eight columns) cards are positioned as shown in Tables 1 and 3. The charged particle densities (numbers cm^{-3}) are in columns 14-21, the height (kilometers) in columns 2-7, and the decimals are in columns 5 and 15. Only as many values of charged particle densities need be input as are needed to adequately describe the profiles; logarithmic interpolation is used to obtain the charged particle densities at intervening heights. All species must have number densities specified at identical height intervals.

2. Collision Frequency Profiles

The charged particle-neutral particle collision frequency as a function of height is input either directly (as in Table 3) or as coefficients for an exponential function (as in Table 1). If the direct method is used a SPECIES 1 COLLFREQ card follows the NINES card of the last species profile or the REGIONS card. The collision frequency is in columns 14-21, the height (kilometers) in columns 2-7, and the decimal points are in columns 5 and 15. Values must be input for the heights of the top and bottom of the species profiles. Also, the heights, for which values are input, must be the same for all species of charged particles. The collision frequency values are obtained at intermediate points by a logarithmic interpolation subroutine.

3. Name List Variables

The name list input variables may be placed anywhere on the card in the following format: The names, e.g. MAGFIELD, must have no spaces between the letters. The name is followed by one space, an = (equal) sign, another space, and the parameter value. If a parameter has more than one value they are listed on the same card or following cards with one or more spaces between values but without repeating the parameter name. An example of the format is COEFFNU in Table 1. The name list cards may be input in any order; the sequence of names, as follows, has no particular significance.

a. AZIMUTH is the direction of propagation in degrees measured clockwise from magnetic north. For example, west to east propagation corresponds to an azimuth of 90 degrees.

b. DIPANGLE is the conjugate of the usual dip angle, i.e. it is the angle between the total geomagnetic field vector and the zenith. The range is from 0 to 90 degrees in the northern hemisphere and from 90 to 180 degrees in the southern hemisphere.

c. MAGFIELD is the intensity of the geomagnetic field in Webers m^{-2} .

d. COEFFNU and EXPNU are coefficients for the functional form of collision frequency input. The form of the function is

$$v = \text{COEFFNU} \times \exp(\text{EXPNU} \times Z)$$

where v and COEFFNU are collisions per second, EXPNU is in inverse meters, and Z is in meters. Table 1 shows the input form for electrons, positive, and negative ions.

e. EPSILON and EPSILONO are the permittivities of the earth's surface and of free space respectively.

f. D is the height in kilometers at which the modal solution is obtained. It can be at or below the height of the lowest species profile input.

g. H is the height in kilometers at which the modified refractive index becomes unity. Because of the insensitivity of the solutions at the ground to H, it is convenient to set it at the same height as D.

h. REFLHT is a reasonable estimate of the height of wave reflection in kilometers and is used solely as a factor in obtaining WAITS EXCITATION FACTOR.

i. ALPHA is equal to $2/\text{earth's radius}$ in kilometers. If solutions are required for a "flat" earth case ALPHA should be set to a small non-zero number (e.g. 10^{-6} km^{-1}).

j. RELPREC sets the precision level (and computation speed) RELPREC ranges from 1.0 to 4.0 with 1.0 being fast but imprecise and 4.0 being slow but highly precise.

k. SIGMA is the ground conductivity in mhos/meter.

l. FREQKHZ is the radio frequency in kilohertz.

m. THETAINC controls the incrementing of the trial complex angles of incidence between succeeding iterations in search for solutions. THETAINC is in degrees and a usual increment is one degree.

n. MRATIO is the ion mass to electron mass ratio and is required only when ions species profiles are input.

o. CHARGE is only required when ions in addition to electrons are input. The electron and negative ion charges are -1.0 and positive ions +1.0.

p. TYPEITER is an optional command which is used to obtain vertically polarized modes only or horizontally polarized modes only. It is physically meaningful to apply this command for an isotropic or near isotropic ionosphere or for transverse propagation at the geomagnetic equator only. The command TYPEITER = 1 is used for vertically polarized modes and TYPEITER = 2 for horizontally polarized modes.

q. RPOLY = 1 is an optional command which allows the program to obtain approximate solutions by interpolating between calculated reflection coefficients for complex angles of incidence in a search for mode eigenangles. It is used to obtain solutions rapidly with some loss in accuracy. It is also used to obtain trial eigenangles for the exact method.

r. TLIST is a list of complex angles selected to bracket the expected range of modal eigenangles sought when using the approximate solution method. That is, TLIST is required when RPOLY = 1 is input but is otherwise deleted.

s. EIGEN is the list of trial eigenangles and is required for both the exact and approximate methods of obtaining solutions. Up to thirty trial angles may be input in the format of real part, space, imaginary part, space, real part etc. Table 3 lists the input for RPOLY = 1, a TLIST, and an appropriate EIGEN list.

t. PHI is the angle of propagation with respect to the axis of a horizontal dipole transmitting antenna. PHI = 0.0 for end launched radiation and PHI = $\pi/2$ for broadside launched radiation. As many as ten angles of propagation may be specified but in any case PHI = 0.0 and PHI = $\pi/2$ are mandatory for PHI₁ and PHI₂ in that order.

u. NRPHIS is the number of PHI's requested ranging from the mandatory 2 to 10.

v. ALT is the transmitter altitude in kilometers. As many as twenty-five transmitter heights may be specified.

w. NRALTS is the number of altitudes specified in the ALT list.

x. ALR is the receiver altitude in kilometers. As many as twenty-five receiver heights may be specified.

y. NRALR is the number of altitudes specified in the ALR list.

z. PRIPUN = -1 is an optional command that restricts the mode excitation factor computations to those for a ground based vertical transmitting antenna and ground based receiver. It is used with RPOLY to obtain trial eigenangles for input to the exact solution program or if only ground based equipment is of interest. If PRIPUN = -1 is used variable input cards PHI, NRPHIS, ALT, NRALTS, ALR, NRAIR may be deleted.

4. Start-Stop Commands

a. WAVEGUID is the command for the computer to start and obtain waveguide solutions. It follows the last of the name list variables but, as detailed in the next section, may be used repeatedly for linking several computer runs.

b. QUIT stops the computer and is the last input card.

5. Linking Several Computer Runs

By doing several different runs in one computer input one can sometimes save on turnaround time and cut down on card duplication of either inputs or profiles.

More than one data deck is run at a time by following the WAVEGUID card with a SPECIES 1 PROFILE card and so on through the profile or through the ion profiles as before. Doing this destroys the profiles which were originally in the computer but all the other inputs remain as they were. At the next WAVEGUID card the program starts on this "second" program. If any changes in input parameters were desired on this second run, they would be changed between the profile and the WAVEGUID card. For example, if $D = 0.0$ on the first profile and $D = 65.0$ on the second, the new profile would end with a nines card and then NEGIONS, if ions were included, then $D = 65.0$ and then WAVEGUID.

Changes in the parameters for the data deck are stated between WAVEGUID cards. All other inputs would remain unchanged. For example, if several mag fields are desired for a given profile and set of parameters, only mag field would be changed before calling WAVEGUID again. For example

```
MAGFIELD = 4.0E-05
```

```
WAVEGUID
```

```
MAGFIELD = 0.0
```

```
WAVEGUID
```

C. Computer Printouts

Tables 2 and 4 are samples of the printout options available. Table 2 includes the waveguide mode parameters for both horizontal and vertical dipole excitation. Table 4 is the printout for vertical dipole excitation only. It is called by the PRIPUN =-1 command.

Contents of Table 2

<u>Page</u>	<u>Line</u>	<u>Description</u>
1	1, 2	Real and imaginary parts of the eigenangle followed by the complex reflection coefficients.
1	3, 4	Magnitude and argument of the reflection coefficients.
1	5	Number of iterations performed to arrive at a solution.
1	6	Difference (real and imaginary parts) between the final two eigenangles.
1	7	Real and imaginary parts of the F function.
1	8	Phase velocity of the mode at ground level.
1	9	Phase velocity divided by the free space velocity.
1	10	Modal E field attenuation at the ground in dB/1000 km.
1	11, 12	Magnitude and angle of the mode mixing parameter.
1	13	Real and imaginary parts of the eigenangle at the ground.
2	1	Height of the transmitting (ALT) and the receiving antennas (ALR).
2	2	Direction of propagation with respect to the axis of a horizontal dipole transmitting antenna.
2	4, 5	Magnitude and phase of the mode excitation factors for the vertical E field.
2	6, 7	Phase and magnitude of the mode excitation factors for the vertical E field according to the J. Wait formulation.
2	8, 9, 10, 11	Same as above but for the E field in the direction of propagation.
2	12, 13, 14, 15	Excitation factors for the horizontal E field normal to the direction of propagation.

<u>Page</u>	<u>Line</u>	<u>Description</u>
2	16	Direction of propagation with respect to the axis of a horizontal dipole transmitting antenna.
2	17-28	Excitation factors, as in lines 4-15, but for the line 16 direction of propagation.
2	29-34	Height-gain functions as defined in Section II.

Contents of Table 4

<u>Line</u>	<u>Description</u>
1-13	Identical to the descriptions of line items on Table 2.
14, 15	Magnitude and phase of the excitation factor for vertical dipole radiation.
16, 17	Phase and amplitude of Wait's excitation factor.

T A B L E 1

SPECIES 1 PROFILE
SAMPLE ELECTRONS

120.00	1.85E 03
110.00	1.47E 03
105.00	1.39E 03
100.00	1.42E 03
95.00	1.66E 03
90.00	3.00E 03
85.00	6.51E 03
80.00	1.51E 04
75.00	3.03E 04
70.00	5.14E 04
65.00	4.22E 04
60.00	8.45E 03
55.00	1.35E 03
50.00	2.79E 02
45.00	3.38E 01
40.00	1.53E 00
35.00	1.33E 02
30.00	6.74E 04
25.00	1.93E 04
20.00	5.87E 05

99999999

SPECIES 2 PROFILE
SAMPLE IONS

120.00	1.85E 03
110.00	1.47E 03
105.00	1.39E 03
100.00	1.49E 03
95.00	1.72E 03
90.00	3.08E 03
85.00	6.76E 03
80.00	1.59E 04
75.00	3.28E 04
70.00	6.34E 04
65.00	9.26E 04
60.00	1.12E 05
55.00	1.10E 05
50.00	7.86E 04
45.00	3.47E 04
40.00	8.09E 03
35.00	3.40E 03
30.00	3.55E 03
25.00	3.68E 03
20.00	3.65E 03

99999999

REGIONS

T A B L E 1

(CONT.)

DIPANGLE = 40.0
 MAGFIELD = 4.4E-05
 EPSILON = 1.32B151E-10
 EPSILON = 8.85434E-12
 ALPHA = 3.14E-04
 RELPREC = 2.0
 COEFFNU = 4.303E11 1.076E10 1.076E10
 EXPNU = -1.622E-04 -1.622E-04 -1.622E-04
 THETAINC = 1.0
 REFLHT = 55.0
 D = 20.0
 H = 20.0
 FREQKHZ = 19.8
 AZIMUTH = 90.0
 SIGMA = 5.0
 MRATIO = 1.0 58000.0 58000.0
 CHARGE = -1.0 1.0 -1.0
 EIGEN = 85.730 -.35 84.900 -.270
 ALT = 0.0 4.0 8.0
 NRALTS = 3
 ALR = 10.0 20.0
 NRALR = 2
 PHI = 0.0.1.5.708
 NRPHIS = 2
 WAVEGUID
 QUIT

T A B L E 2

ITERATION TYPE 0		11R11 BAR		1R1 BAR		1R11		11R1		R11 BAR*11		R1 BAR*11	
MAX DELTA = 0		THETA		1R1 BAR		1R11		1R1		R11 BAR*11		R1 BAR*11	
REAL	85.730	9.16297E-01	6.31508E-01	7.24167E-01	7.08813E-01	-4.22691E-02	-4.22691E-02	7.06773E-01	7.06773E-01	6.60194E-01	6.60194E-01	4.00394E-01	4.00394E-01
IMAG	.035	4.63408E-01	8.01248E-01	-9.32680E-02	-2.65302E-01	4.37653E-01	4.37653E-01	4.3969	4.3969	2.50123E-01	2.50123E-01	.77212	.77212
MAG		1.02681	1.02020	.73015	.75684	.43969	.43969	95.51659	95.51659	19.48863	19.48863	31.23598	31.23598
ANGLE		26.82756	51.75640	352.66107	339.47958								
THETA	85.730	F MAG = 8.536E-04		DFDTHETA MAG = 1.154E 00									
THETA	85.731	F MAG = 8.536E-04		DFDTHETA MAG = 1.154E 00									
THETA		11R11 BAR	1R1 BAR	11R11	1R1	11R11	1R1	11R1	1R1	R11 BAR*11	R1 BAR*11	R1 BAR*11	R1 BAR*11
1 REAL	85.731	9.16297E-01	6.31508E-01	7.24167E-01	7.08813E-01	-4.22691E-02	-4.22691E-02	7.06773E-01	7.06773E-01	6.60194E-01	6.60194E-01	4.00394E-01	4.00394E-01
2 IMAG	.035	4.63408E-01	8.01248E-01	-9.32680E-02	-2.65302E-01	4.37653E-01	4.37653E-01	4.3969	4.3969	2.50123E-01	2.50123E-01	.77212	.77212
3 MAG		1.02681	1.02020	.73015	.75684	.43969	.43969	95.51659	95.51659	19.48863	19.48863	31.23598	31.23598
4 ANGLE		26.82756	51.75640	352.66107	339.47958								
5 ITERATIONS PERFORMED = 1													
6 DELTHETA		.0006212	.0004010										
7 F FUNCTION		.00085	.00012										
8 PHASE VELOCITY		= 2.98243E 05 KM PER SEC											
9 PHASE VELOCITY OVER C		= .99483											
10 ATTENUATION		= 1.67120E+01 DB											
11 POLARIZATION MAG		= 1.17141E 00											
12 POLARIZATION ANGLE		= 162.80839 DEG											
13 THETA PRIME		= 89.974 -5.03A DEG											

T A B L E 2 (CONT.)

1	ALY =	0 KM	ALR =	10.00000 KM		
2	HORIZONTAL DIPOLE	PHI =	0 RAD		VERTICAL DIPOLE	
4	EXTRA MAG =	3.14893E-06			EXTRA MAG =	6.79794E-03
5	EXTRA ANGLE =	.00390 RAD			EXTRA ANGLE =	1.58925 RAD
6	PHASE OF WAITS EXCITATION FACTOR =	5.51629 RAD			PHASE OF WAITS EXCITATION FACTOR =	.91846 RAD
7	WAITS EXCITATION FACTOR IN DB =	-8.46121E 01 DB			WAITS EXCITATION FACTOR IN DB =	-1.79277E 01 DB
8	EXTRA MAG =	1.07953E-07			EXTRA MAG =	2.33053E-04
9	EXTRA ANGLE =	5.51452 RAD			EXTRA ANGLE =	.91674 RAD
10	PHASE OF WAITS EXCITATION FACTOR =	3.94378 RAD			PHASE OF WAITS EXCITATION FACTOR =	4.72933 RAD
11	WAITS EXCITATION FACTOR IN DB =	-1.13911E 02 DB			WAITS EXCITATION FACTOR IN DB =	-4.72242E 01 DB
12	EXTRA MAG =	9.69338E-07			EXTRA MAG =	1.24133E-01
13	EXTRA ANGLE =	.88111 RAD			EXTRA ANGLE =	5.51654 RAD
14	PHASE OF WAITS EXCITATION FACTOR =	5.59350 RAD			PHASE OF WAITS EXCITATION FACTOR =	3.04975 RAD
15	WAITS EXCITATION FACTOR IN DB =	-9.57915E 01 DB			WAITS EXCITATION FACTOR IN DB =	7.30246E 00 DB
16	HORIZONTAL DIPOLE	PHI =	1.57080 RAD			
17	EXTRA MAG =	2.23483E-07			EXTRA MAG =	1.72232E-01
18	EXTRA ANGLE =	2.45260 RAD			EXTRA ANGLE =	4.71981E-01
19	PHASE OF WAITS EXCITATION FACTOR =	.88230 RAD			PHASE OF WAITS EXCITATION FACTOR =	4.54719E 01
20	WAITS EXCITATION FACTOR IN DB =	-1.07590E 02 DB			WAITS EXCITATION FACTOR IN DB =	3.15359E 02
21	EXTRA MAG =	7.64163E-09			EXTRA MAG =	5.03291E-01
22	EXTRA ANGLE =	.88924 RAD			EXTRA ANGLE =	2.70402E 02
23	PHASE OF WAITS EXCITATION FACTOR =	5.59268 RAD			PHASE OF WAITS EXCITATION FACTOR =	3.39342E-01
24	WAITS EXCITATION FACTOR IN DB =	-1.16959E 02 DB			WAITS EXCITATION FACTOR IN DB =	3.39342E-01
25	EXTRA MAG =	6.16622E-08			EXTRA MAG =	1.72232E-01
26	EXTRA ANGLE =	2.53419 RAD			EXTRA ANGLE =	4.71981E-01
27	PHASE OF WAITS EXCITATION FACTOR =	.96339 RAD			PHASE OF WAITS EXCITATION FACTOR =	4.54719E 01
28	WAITS EXCITATION FACTOR IN DB =	-1.18775E 02 DB			WAITS EXCITATION FACTOR IN DB =	3.15359E 02
29	WGTVR =	4.59330E-01			WGTVR =	4.71981E-01
30	WGTME =	2.12177E-04			WGTME =	4.54719E 01
31	WGTMR =	1.89908E-05			WGTMR =	3.15359E 02
32	WGRZ =	4.95847E-01			WGRZ =	5.03291E-01
33	WGRX =	1.69519E-02			WGRX =	2.70402E 02
34	WGRY =	1.72232E-01			WGRY =	3.39342E-01
35	EXTRA MAG =	6.16622E-08			EXTRA MAG =	1.72232E-01
36	EXTRA ANGLE =	2.53419 RAD			EXTRA ANGLE =	4.71981E-01
37	PHASE OF WAITS EXCITATION FACTOR =	.96339 RAD			PHASE OF WAITS EXCITATION FACTOR =	4.54719E 01
38	WAITS EXCITATION FACTOR IN DB =	-1.18775E 02 DB			WAITS EXCITATION FACTOR IN DB =	3.15359E 02

T A B L E 3

SPECIES 1 PROFILE
SAMPLE ELECTRONS

250.00	4.75E 05
200.00	2.60E 05
100.00	2.50E+05
90.00	2.25E+03
80.00	6.20E+02
70.00	1.20E+02
60.00	1.05E+02
50.00	4.27E-01
40.00	5.42E-02
30.00	4.03E-03
20.00	2.19E-04
10.00	1.47E-05
05.00	5.88E-06
00.00	3.07E-06

99999999

SPECIES 2 PROFILE
SAMPLE POSITIVE IONS

250.00	4.75E 05
200.00	2.60E 05
100.00	2.50E+05
90.00	2.27E+03
80.00	6.32E+02
70.00	1.73E+02
60.00	8.59E+02
50.00	1.54E+03
40.00	2.73E+03
30.00	3.55E+03
20.00	3.65E+03
10.00	3.72E+03
05.00	4.48E+03
00.00	5.50E+03

99999999

REGIONS

SPECIES 1 COLLFREQ	
250.00	1.90E 03
100.00	3.90E 04
000.00	4.30E 11

99999999

T A B L E 3

(CONT.)

SPECIES 2 COLLFREQ

250.00 4.75E 01
100.00 9.75E 02
000.00 1.07E 10

99999999

SPECIES 3 COLLFREQ

250.00 4.75E 01
100.00 9.75E 02
000.00 1.07E 10

99999999

AZIMUTH = 270.0

DIPANGLE = 65.0

MAGFIELD = 3.5E-05

EPSILON = 7.17201E-10

EPSILON = 8.85434E-12

D = 0.0

H = 0.0

ALPHA = 3.14E-04

RELPREC = 2.0

SIGMA = 4.64

REFLHT = 20.0

MRATIO = 1.0 58000.0 58000.0

CHARGE = -1.0 1.0 -1.0

THETAINC = 1.0

FREQKHZ = 18.0

RPOLY = 1

TLIST = 84.0 -1.5 85.0 -1.0 86.0 -.5

EIGEN = 85.230 -1.198

PRIPUN = -1

WAVEGUID

QUIT

References :

1. Smith, R., "Program for the full-wave calculations of reflections in an arbitrary ionosphere (U)," DASA Report 1615, 1965.
2. Gossard, E. E., et al, "A computer program for vlf mode constants in an earth-ionosphere waveguide of arbitrary electron density distribution," MIPR DASA No. 558-64, 1966.
3. Gossard, E. E., Y. Gough, V. Noonkester, R. Pappert, I. Rothmuller, R. Smith, and P. Winterbauer, "A computer program for vlf mode constants in an earth-ionosphere waveguide of arbitrary electron density distribution," Navy Electronics Laboratory Interim Report No. 671 for DASA, September 1966.
4. McKinley, R. R., "A rigorous waveguide program for a cylindrical waveguide," NEL Tech Memo 1315, October 1967.
5. Pappert, R. A. and C. H. Sheddy, "An accelerated waveguide program," Navy Electronics Laboratory Tech Memo 1040, January 1967.
6. Pappert, R. A., "On the problem of horizontal dipole excitation of the earth-ionosphere waveguide," NEL Interim Report 684 for DASA, April 1968.
7. Sheddy, C. H., Y. Gough, and R. A. Pappert, "An improved computer program for vlf mode constants in an earth-ionosphere waveguide of arbitrary electron density distribution," NEL Interim Report No. 682 for DASA, February 1968.
8. Sheddy, C. H., R. Pappert, Y. Gough, and W. Moler, "A fortran program for mode constants in an earth-ionosphere waveguide," NELC Interim Report No. 683 for DASA, May 1968.
9. Budden, G. G., "The influence of the earth's magnetic field on radio propagation by waveguide modes," Proc. Ray. Soc. (London) Ser. A265, No. 1323, 538-553. (1962).
10. Staff of the Computation Laboratory at Cambridge, Mass., "Tables of the modified Hankel functions of order one-third and of their derivatives," (Harvard Univ. Press, Cambridge, Mass) (1945).
11. Wait, J. R., "Electromagnetic waves in stratified media," p. 221 (Pergamon Press, New York, N. Y.) (1962).

APPENDIX A:
PROGRAM LISTING

```

PROGRAM INPJT
COMMON/INPUT/THETA,FREQ KHZ,AZIMUTH,DIP ANGLE,MAG FIELD,ALPHA,H,
$      D,REL PREC,TOP HT,LOWEST HT,Q MAX,DELHMIN,SWITCH,
$      IN EXTRA(85)
COMMON/SP INPUT/NR SPEC,COEFF NU(5),EXP NU(5),CHARGE(5),M RATIO(5)
COMMON/FLG INPUT/DEBUG,RC PRINT,POLY FLAG,INTEG UP,WF FLAG,
$      NEG ELECT,INDX REFR,INHOMG,J FLAG,GRAPHS,
$      FJ EXTRA(15)
COMMON/WG INPUT/TYPE ITER,MAX DELTA,EPSILON,EPSILON 0,SIGMA,
$      WGIN SKIP,
$      REF HT,R POLY,T LIST(11),EIGEN(30),THETA INC,
$      W3 OMIT(2),DTHETA,LUB,WG EXTRA(3)
COMMON/SOL TNS/NR SOL,SOLTN(25)
COMMON/I/SKIP1,HT LIST(150),LOG N LIST(150,5)
COMMON/COLL FREQ/CF SKIP,NU FLAG,C HT LIST(25),C LOG LIST(25,5)
COMMON/FLD INPUT/DIST,DEL DIST,ITITLE1(12),ITITLE2(12)
COMMON/PR INPT/MPRINT,HT0,FN,S RANGE,H0 REF,MIN,PR XHPA(19)
COMMON/A INPUT/NR A,B(9,9),DA(9)
COMMON/EXPR IN/NR V,EXP VAL(24),EXP JNC(24)
COMMON/FREQ IN/NR F,NR VPF,EXPER F(6)
COMMON/HU INPUT/ALT(25),NR ALTS,PHI(10),NR PHIS,HD FLAG,NR ALR,
$      ALR(25),HD SKIP(12)
$      ,PRIPUN
NAMELIST/DATA/THETA,FREQ KHZ,AZIMUTH,DIP ANGLE,MAG FIELD,ALPHA,H,
$      D,REL PREC,TOP HT,LOWEST HT,Q MAX,DELHMIN,SWITCH,
$      COEFF NU,EXP NU,CHARGE,M RATIO,SPECIES,
$      DEBUG,POLY FLAG,NEG ELECT,INDX REFR,INHOMG,
$      J FLAG,GRAPHS,
$      TYPE ITER,MAX DELTA,EPSILON,EPSILON 0,SIGMA,
$      REF HT,R POLY,T LIST,EIGEN,THETA INC,DTHETA,LUB,
$      NR SOL,SOLTN,
$      DIST,DEL DIST,
$      MPRINT,HT0,FN,S RANGE,H0 REF,MIN,
$      NR A,B1,B2,B3,B4,B5,B6,B7,B8,B9,DA,
$      EXP VAL,EXP JNC,
$      NR F,NR VPF,EXPER F,
$      ALT,NR ALTS,PHI,NR PHIS,HD FLAG,NR ALR,ALR
$      ,PRIPUN
EQUIVALENCE(B(1),B1),(B(10),B2),(B(19),B3),(B(28),B4),(B(37),B5),
$      (B(46),B6),(B(55),B7),(B(64),B8),(B(73),B9)

```

```

COMPLEX THETA
COMPLEX EIGEN,T LIST,DTHETA,LUB
COMPLEX SOLTN
REAL MAG FIELD,LOWEST HT,M RATIO
REAL MAX DELTA
REAL LOG N LIST
REAL NU
INTEGER DEBUG,RC PRINT,POLY FLAG,WF FLAG,GRAPHS
INTEGER BCD,SPECIES
INTEGER TYPE ITER,R POLY
INTEGER HD FLAG
INTEGER PRIPUN
DIMENSION BCD(10)

```

```
DATA(PRIPUN = 0)
```

DATA(MAG FI=LJ=1.0E99)
 DATA(ALPHA=0.0),(H=0.0)
 DATA(D=0.0)
 DATA(REL PR=C=3.5)
 DATA(LOWEST HT=0.0)
 DATA(Q MAX=1.0E99)
 DATA(DELMIN=0.03125)
 DATA(SWICH=1.0E 01)

 DATA(SPECIES=1),(NR SPEC=1)
 DATA(COEF NU=4.303E11,1.076E10,1.076E10)
 DATA(EXP NU=-1.622E-04,-1.622E-04,-1.622E-04)
 DATA(CHARGE=-1.0,1.0,-1.0)
 DATA(RATIO=1.0,58000.0,58000.0)
 DATA(NU FLAG=0)

 DATA(DEBUG=0)
 DATA(RC PRIVI=1)
 DATA(POLY FLAG=0)
 DATA(INTEG JP=0)
 DATA(XF FLAG=0)
 DATA(NEG ELECT=0)
 DATA(INDX REF=0)
 DATA(INHUMG=0)
 DATA(J FLAG=0)
 DATA(GRAPHS=0)

 DATA(TYPE IIR=0),(MAX DELTA=0.0)
 DATA(REFL HT=0.1)
 DATA(R POLY=0)
 DATA(T LIST=11((0.0,0.0)))
 DATA(EIGEN=30((0.0,0.0)))
 DATA(THETA INC=0.1)
 DATA(DTHETA=(0.05,0.01))
 DATA(LUB=(0.05,0.002))
 DATA(NR SOL=0)

 DATA(TITLE1=16FIELD STRENGTH -,10(BH))
 DATA(TITLE2=16RELATIVE PHASE -,10(BH))
 DATA(DIS=10000.0),(DEL DIST=0.0)

 DATA(PRINT=0)
 DATA(HT0=70.0)
 DATA(FN=3.0=03)
 DATA(S RANGE=0.5)
 DATA(H0 REF=65.0)
 DATA(MIN=20)
 DATA(NR A=2)
 DATA(B1=1.0,8(0.0))
 DATA(B2=0.0,1.0,7(0.0))
 DATA(B3=2(0.0),1.0,6(0.0))
 DATA(B4=3(0.0),1.0,5(0.0))
 DATA(B5=4(0.0),1.0,4(0.0))
 DATA(B6=5(0.0),1.0,3(0.0))
 DATA(B7=6(0.0),1.0,2(0.0))
 DATA(B8=7(0.0),1.0,0.0)

NOT REPRODUCIBLE

```

DATA(H9=8(0.0),1.0)
DATA(DA=0.2*0.2*0.2*1.0*0.1)
DATA(NR P=3)
DATA(NR VPF=3)

DATA(HD FLAG=0)

```

```

PRINT 100
100 FORMAT(1H1)

```

```

10 NW = INLIST(DATA,N)
IF(NW .EQ. /HPROFILE) GO TO 20
IF(NW .EQ. /HREVERSE) GO TO 40
IF(NW .EQ. /HREGIONS) GO TO 45
IF(NW .EQ. 8HCOLLFREQ) GO TO 40
IF(NW .EQ. 2HRC) GO TO 50
IF(NW .EQ. 8HINITIALA) GO TO 51
IF(NW .EQ. 2HAF) GO TO 55
IF(NW .EQ. 8HAVEGUID) GO TO 60
IF(NW .EQ. /HFITPROF) GO TO 70
IF(NW .EQ. 6HDESERT) GO TO 71
IF(NW .EQ. 2HID) GO TO 80
IF(NW .EQ. 4HQUIT) GO TO 99
GO TO 90

```

C ELECTRON AND ION DENSITY PROFILES

```

20 READ 201,(BCD(J),J=1,10)
PRINT 202,(BCD(J),J=1,10)
K = SPECIES
IF(K .GT. NR SPEC) NR SPEC = K
L = 1
21 READ 201,(BCD(J),J=1,10)
201 FORMAT(10A8)
PRINT 202,(BCD(J),J=1,10)
202 FORMAT(1H ,10A8)
IF(BCD(1) .EQ. 8H99999999) GO TO 22
DECODE(21,203,BCD) HT,EN
203 FORMAT(F7.2,5X,E9.2)
IF(K .NE. 1 .AND. HT .NE. HT LIST(L)) GO TO 90
HT LIST(L) = HT
IF(L .NE. 1 .AND. HT LIST(L) .GE. HT LIST(L-1)) GO TO 90
IF(EN .EQ. 0.0) EN = 1.0E-43
LOG N LIST(L,K) = LOGF(EN)
L = L+1
GO TO 21
22 TJP HT = HT LIST(1)
LOWEST HI = HT LIST(L-1)
NR PTS = L-1
INTEG UP = 0
GO TO 10

30 L HALF = NR PTS/2
D) 31 L=1,L HALF
M = NR PTS+1-
TEMP = HI LIST(L) $ HT LIST(L) = HT LIST(M) $ HI LIST(M) = TEMP

```

```

TEMP = LOG N LIST(L,1) & LOG N LIST(L,1) = LOG N LIST(M,1)
      LOG N LIST(M,1) = TEMP
31 CONTINUE
   DO 32 L=1,NR PTS
32 HT LIST(L) = -HT LIST(L)
   INTEG UP = 1-INTEG UP
   TEMP = TOP HT & TOP HI = -LOWEST HT & LOWEST HI = -TEMP
   GO TO 10

35 DO 36 L=1,NR PTS
   EN = EXP(LJG N LIST(L,2))-EXP(LJG N LIST(L,1))
   IF(EN .EQ. 0.0) EN = 1.0E-40
   LOG N LIST(L,3) = LOGF(EN)
36 PRINT 301,HT LIST(L),EN
301 FORMAT(1M,/,2,2X,E9.2)
   NR SPEC = J
   GO TO 10

C COLLISION FREQUENCY PROFILES
40 NF FLAG = 1
   K = SPECIES
   M = 1
41 READ 201,(HCD(J),J=1,10)
   PRINT 202,(BCD(J),J=1,10)
   IF(HCD(1) .EQ. 8H99999999) GO TO 10
   DECODE(21,203,BCD) C HT LIST(M),NU
   IF(NU .EQ. 0.0) NU = 1.0E-40
   C LOG LIST(M,K) = LOGF(NU)
   I = M+1
   GO TO 41

C RC
50 CALL ZERO CLK
   CALL INTEG
   CALL PRINTIME
   PRINT 100
   GO TO 10

51 CALL INITL A
   GO TO 10

C WAVEFIELDS
55 CALL ZERO CLK
   NF FLAG = 1
   CALL IM INTS
   CALL WAV FLJ
   NF FLAG = 0
   CALL PRINTIME
   GO TO 10

C WAVESJID
60 CALL WAVESJID
   GO TO 10

C PROFILE FITTING
70 POLY FLAG = 1

```

```
LOWEST HI = 0.0
NR V = NR V-F*NR F
RC PRINT = 0
CALL FIT PRF
RC PRINT = 1
GO TO 10

71 CONTINUE
RETURN

C 10
80 READ 801,(BCD(J),J=1,10)
801 FORMAT(10A8)
PRINT 802,(BCD(J),J=1,10)
802 FORMAT(1H ,10A8)
DO 81 J=1,10
ITITLE1(J+2) = BCD(J)
81 ITITLE2(J+2) = BCD(J)
GO TO 10

C ERROR EXIT
90 PRINT 900
900 FORMAT(1H0,10HERROR IN DATA DECK)

99 CONTINUE
IF(GRAPHS .EQ. 1) CALL PLOT(n,0,999)

END
```

NOT REPRODUCIBLE

```

FUNCTION CX FCT(ARG)

COMMON/OVRFLO/NR OVFL
COMPLEX CX FCT,ARG,I
REAL NEG
DATA(SAVE REAL=1.0E99),(SAVE IMAG=1.0E99)
DATA(I=(0.0,1.0))
DATA(NR OVFL=0)

ENTRY CPLX SQRT
ENTRY CX SQRT
ARG REAL = ARG & ARG IMAG = -I*ARG
RHO = SQRTF(ARG REAL*ARG REAL+ARG IMAG*ARG IMAG)
ABS REAL = ABSF(ARG REAL)
IF(RHO .LT. ABS REAL) RHO = ABS REAL
IF(ARG IMAG .GE. 0.0) 1,2
1 CX FCT = SQRTF((RHO+ARG REAL)*0.5)+I*SQRTF((RHO-ARG REAL)*0.5)
RETURN
2 CX FCT = SQRTF((RHO+ARG REAL)*0.5)-I*SQRTF((RHO-ARG REAL)*0.5)
RETURN

ENTRY CPLX LOGF
ENTRY CX LOGF
ARG REAL = ARG & ARG IMAG = -I*ARG
RHO = SQRTF(ARG REAL*ARG REAL+ARG IMAG*ARG IMAG)
IF(RHO .EQ. 0.0) GO TO 13
ABS REAL = ABSF(ARG REAL)
IF(RHO .LT. ABS REAL) RHO = ABS REAL
IF(ARG IMAG .GE. 0.0) 11,12
11 CX FCT = LOGF(RHO)+I*ACOSF(ARG REAL/RHO)
RETURN
12 CX FCT = LOGF(RHO)-I*ACOSF(ARG REAL/RHO)
RETURN
13 CX FCT = -200.0
RETURN

ENTRY CPLX EXPF
ENTRY CX EXPF
ARG REAL = ARG & ARG IMAG = -I*ARG
IF(ABSF(ARG REAL) .GT. 709.0 .OR. ABSF(ARG IMAG) .GT. 6.0E10)
& NR OVFL = NR OVFL+1
IF(NR OVFL .GT. 100) STOP
IF(ARG IMAG .EQ. 0.0) 21,22
21 CX FCT = EXPF(ARG REAL)
RETURN
22 IF(ARG REAL .EQ. 0.0) 23,24
23 CX FCT = COSF(ARG IMAG)+I*SINF(ARG IMAG)
RETURN
24 CX FCT = EXPF(ARG REAL)*(COSF(ARG IMAG)+I*SINF(ARG IMAG))
RETURN

```

```

ENTRY CPLX COSF
ENTRY CX COSF
ARG REAL = ARG & ARG IMAG = -I*ARG
IF(ARG IMAG .EQ. 0.0) 31,32
31 CX FCT = COSF(ARG REAL)
RETURN
32 IF(ARG REAL .EQ. SAVE REAL .AND. ARG IMAG .EQ. SAVE IMAG) GO TO 33
SAVE REAL = ARG REAL & SAVE IMAG = ARG IMAG
COS = COSF(ARG REAL)
SIN = SINF(ARG REAL)
POS = EXPF(ARG IMAG) & NEG = 1.0/POS
COSH = (POS+NEG)*0.5
SINH = (POS-NEG)*0.5
33 CX FCT = COS*COSH-I*SIN*SINH
RETURN

```

```

ENTRY CPLX SINF
ENTRY CX SINF
ARG REAL = ARG & ARG IMAG = -I*ARG
IF(ARG IMAG .EQ. 0.0) 41,42
41 CX FCT = SINF(ARG REAL)
RETURN
42 IF(ARG REAL .EQ. SAVE REAL .AND. ARG IMAG .EQ. SAVE IMAG) GO TO 43
SAVE REAL = ARG REAL & SAVE IMAG = ARG IMAG
COS = COSF(ARG REAL)
SIN = SINF(ARG REAL)
POS = EXPF(ARG IMAG) & NEG = 1.0/POS
COSH = (POS+NEG)*0.5
SINH = (POS-NEG)*0.5
43 CX FCT = SIN*COSH+I*COS*SINH
RETURN

```

```

ENTRY CONJ
ARG REAL = ARG & ARG IMAG = -I*ARG
CX FCT = ARG REAL-I*ARG IMAG
RETURN

```

```

END

```

```

SUBROUTINE MAG ANG(ARG,MAG,ANGLE)

COMPLEX ARG,I
REAL MAG
DATA(I=(0.0,1.0))
DATA(RAD TO DEG=57.29577951)

ENTRY MAG ANGLE
ARG REAL = ARG $ ARG IMAG = -I*ARG
MAG = SQRT(ARG REAL*ARG REAL+ARG IMAG*ARG IMAG)
IF(MAG.EQ.0.0) GO TO 10
COS = ARG REAL/MAG
IF(COS .GT. 1.0 .AND. COS .LT. 1.01) COS = 1.0
IF(COS .LT. -1.0 .AND. COS .GT. -1.01) COS = -1.0
> ANGLE = ACOS(COS)*RAD TO DEG
IF(ARG IMAG .LT. 0.0) ANGLE = 360.0-ANGLE
RETURN
10 COS = 0.0
GO TO >

END

```

SUBROUTINE WAVEGUID

```
COMMON/INPUT/THETA,IN OMIT(9H)
COMMON/FLG INPUT/FLG SKIP,RC PRINT,FLG OMIT(23)
COMMON/WG INPUT/WGIN SKIP(7),R POLY,WGIN OMIT(22),EIGEN(30),
$      WGIN NO(10)
COMPLEX THETA,EIGEN
INTEGER RC PRINT,R POLY
```

```
RC PRINT = 0
IF(R POLY .EQ. 1) CALL GEN R POLY
CALL INIT F_LDS
```

```
INDEX = 1
11 THETA = EIGEN(INDEX)
CALL ITERATE
CALL COMP PROC
CALL FIELDS
EIGEN RL = EIGEN(INDEX+1)
IF(EIGEN RL .EQ. 0.0) GO TO 21
INDEX = INDEX+1
GO TO 11
```

```
21 RC PRINT = 1
CALL PRNT F_LDS
RETURN
```

```
END
```

SUBROUTINE COMP PROC

```

COMMON/INPUT/THETA,FREQ KHZ,IN SKIP(3),ALPHA,H,D,IN OMIT(91)
COMMON/WG INPUT/WGIN SKIP(6),REFL HT,WGIN OMIT(93)
COMMON/F/F SKIP(2),DFDTHETA,R BAR11,R BAR22
COMMON/R/R SKIP(18),R11,R22,R12,R21
COMMON/RB CP/SW FAC(25),
$   SCRIT(25),SG FAC(25),ST FAC(25),SGFACR(25),STFACR(25)
COMMON/PROP DATA/WAITS EXC,KSM1,PD OMIT
COMMON/HD INPUT/ALT(25),NR ALTS,PHI(10),NR PHIS,HD FLAG,NR ALR,
$   ALR(25),HD SKIP(12)
$   ,PRIPUN
DIMENSION X(24)
COMPLEX I,S,SQRT S,
$   SH FAC,SCRIPT H,
$   DFDTHETA,THETA,CPLX SQRT,CPLX SIN F,S THETA P,C THETA P,
$   R11,R22,R12,R21,R BAR11,R BAR22,
$   EXTRA,WAITS EF,DENOM,POLAR,WAITS EXC,KSM1,
$   E I PI OV 4,STORE 1,STORE2,
$   XTRA1,XTRA2,SCRIT,SG FAC,ST FAC ,SGFACR,STFACR,
$   XTRAX,XTRAY
COMPLEX HGTVR,HGTHE,HGTHB,HGRX,HGRY,HGRZ,XCVX,XCVY,XCVZ,
$XHEX,XHEY,XHEZ,XHBX,XHBY,XHBZ
COMPLEX OHGTVR,OHGTHE,OHGTHB,OHGRX,OHGRY,OHGRZ,OXCVZ,OXCVX,
$   OXCVY,OXHEZ,OXHEX,OXHEY,OXHBZ,OXHBX,OXHBY
REAL K,LOGE10,IMAG
INTEGER HD FLAG
INTEGER PRIPUN
DATA(I)=(0,0,1,0)
DATA(DEG TO RAD=0.01745329252),(RAD TO DEG=57.29577951)
DATA(VEL LIGHT=2.997928E05)
DATA(PI=3.141592653)
DATA(LOGE10=2.302585093)
DATA(E I PI OV 4=(0.7071067812,0.7071067812))

IHGTVR = 5HHGTVR
IHGTHE = 5HHGTHE
IHGTHB = 5HHGTHB
IHGRX = 4HHGRX
IHGRY = 4HHGRY
IHGRZ = 4HHGRZ
IXCVX = 4HXCXVX
IXCVY = 4HXCXVY
IXCVZ = 4HXCXVZ
IXHEX = 4HXHEX
IXHEY = 4HXHEY
IXHEZ = 4HXHEZ
IXHBX = 4HXHBX
IXHBY = 4HXHBY
IXHBZ = 4HXHBZ
K = 2.0*PI*FREQ KHZ*1000.0/VEL LIGHT
CAP K = 1.0/(1.0-ALPHA*H/2.0)
S = CPLX SIN F(THETA*DEG TO RAD)
S THETA P = CAP K*S

```

```

STP REAL = S THETA P
STP IMAG = -I*S THETA P
PHASE VEL = VEL LIGHT/STP REAL
V OVER C = PHASE VEL/VEL LIGHT
ATTEN = -8.6858896*K*STP IMAG*1000.0
KSM1 = K*(S THETA P-1.0)
DENOM = R BAR11*R11-1.0
REAL = DENOM
IMAG = -I*DENOM
IF(REAL .EQ. 0.0 .AND. IMAG .EQ. 0.0) DEJON = 1.0E-150
POLAR = -R BAR11*R12/DENOM
CALL MAG ANGLE(POLAR,POLAR MAG,POLAR ANG)

C THETA P = CPLX SQRT(1.0-S THETA P**2)
CTP REAL = C THETA P
COS = SQRTF(CTP REAL**2-STP IMAG**2)
THETA P RL = ACOSF(COS)*RAD TO DEG
THETA P IM = LOGF((CTP REAL+STP IMAG)/COS)*RAD TO DEG

PRINT 100
100 FORMAT(1H0,/)
PRINT 101,PHASE VEL
101 FORMAT(1H ,2X,17HPHASE VELOCITY = ,E12.5,11H KM PER SEC)
PRINT 102,V OVER C
102 FORMAT(1H ,2X,24HPHASE VELOCITY OVER C = ,F9.5)
PRINT 103,ATTEN
103 FORMAT(1H0,2X,14HATTENUATION = ,E12.5,3H DB)
PRINT 116,POLAR MAG
116 FORMAT(1H0,2X,19HPOLARIZATION MAG = .E12.5)
PRINT 117,POLAR ANG
117 FORMAT(1H ,2X,21HPOLARIZATION ANGLE = ,F10.5,4H DEG)
PRINT 118,THETA P RL,THETA P IM
118 FORMAT(1H0,2X,14HTHETA PRIME = ,2F8.3,4H DEG)
IF(PRI PUN ,NE. -1) GO TO 317
M = 1
J = 1
SCRIPT H = EXPF(+ALPHA*(ALR(M)-D)/2.0)*SH FAC(M)
SQRT S = CPLX SQRT(S)
STORE 2 = (1.0 + R BAR 11)**2*(1.0-RBAR22*R22)*SCRIPT H*SCRIT(J)
$ /((DFDTHETA/DEG TO RAD*RBAR11)
EXTRA = STORE2*S*S*SQRT S
WAITS EF = -I*K*REFL HT/2.0*EXTRA
WAITS EXC = 0.06496*K/(2.0*SQRTF(FREQ KHZ))*EXTRA
CALL MAG ANGLE(EXTRA,EXTRA MAG,EXTRA ANG)
EXTRA ANG = EXTRA ANG*DEG TO RAD
CALL MAG ANGLE(WAITS EF,WAITS MAG,WAITS ANG)
WAITS ANG = WAITS ANG*DEG TO RAD
E SUB Z = 0.06496*WAITS MAG/(SQRTF(FREQ KHZ)*REFL HT)
WAIT EF DB = 20.0*LUGF(WAITS MAG)/LOGE10
E SUB Z DB = 20.0*LUGF(E SUB Z)/LOGE10
I02 = 1H
I0 = 2HVZ
XCVZ = EXTRA/(SCRIPTH*SCRIT(J))
HGTVR = SCRIT(J)
HGRZ = SCRIPTH

```

```

PRINT 104, EXTRA MAG
PRINT 105, EXTRA ANG
PRINT 112, WAITS ANG
PRINT 115, WAIT EF DB
RETURN
317 IF (PRIPUN, NE, 0, AND, PRIPUN, NE, 3) PUNCH 318, THETAPRL, THETAPII
318 FORMAT(1H, 2X, 14H THETA PRIME = , 2F8.3, 4H DEG)
20 DO 21 J = 1, NR ALTS, 1
DO 21 M = 1, NR ALK, 1

SCRIPT H = EXPF(+ALPHA*(ALR(M)-D)/2.0)*SM FAC(M)
SQRT S = CPLX SQRT(S)
STORE 2 = (1.0 + R BAR 11)**2*(1.0-RBAR22*R22)*SCRIPT H*SCRIT(J)
S / (DFUTHEA/DEG TO RAD*RBAR11)
EXTRA = STORE2*S*S*SQRT S
WAITS EF = -I*K*REFL HT/2.0*EXTRA
WAITS EXC = 0.06496*K/(2.0*SQRT(FREQ KHZ))*EXTRA
CALL MAG ANGLE(EXTRA, EXTRA MAG, EXTRA ANG)
EXTRA ANG = EXTRA ANG*DEG TO RAD
CALL MAG ANGLE(WAITS EF, WAITS MAG, WAITS ANG)
WAITS ANG = WAITS ANG*DEG TO RAD
E SUB Z = 0.06496*WAITS MAG/(SQRT(FREQ KHZ)*REFL HT)
WAIT EF DB = 20.0*LOGF(WAITS MAG)/LOGE10
E SUB Z DB = 20.0*LOGF(E SUB Z)/LOGE10
IF (PRIPUN, NE, 0, AND, PRIPUN, NE, 4) PUNCH 207, ALT(J), ALR(M)
PRINT 207, ALT(J), ALR(M)
207 FORMAT(1H1, 6H ALT = , 2X, F10.5, 3H KM; 5X, 6H ALR = , F10.5, 3H KM)
ID2 = 1H
ID = 2HVZ
XCVZ = EXTRA/(SCRIPT H*SCRIT(J))
HGTVR = SCRIT(J)
HGRZ = SCRIPT H
IF (PRIPUN, EQ, 1) PUNCH 503, EXTRA, ID, ID2
IF (PRIPUN, EQ, 2) PUNCH 503, WAITS EXC, ID, ID2
503 FORMAT(1X, C(E12.5, E12.5), 20X, A2, A1)
X(1) = EXTRA MAG
X(2) = EXTRA ANG
X(3) = WAITS ANG
X(4) = WAIT EF DB
STORE2 = STORE2/(S*SCRIPT H)*SGFACR(M)
EXTRA = STORE2*S*S*SQRT S
ID = 2HVX
XCVX = EXTRA/(SGFACR(M)*SCRIT(J))
HGRX = SGFACR(M)
IF (PRIPUN, EQ, 1) PUNCH 503, EXTRA, ID, ID2
WAITS EF = -I*K*REFL HT/2.0*EXTRA
WAITS EXC = 0.06496*K/(2.0*SQRT(FREQ KHZ))*EXTRA
IF (PRIPUN, EQ, 2) PUNCH 503, WAITS EXC, ID, ID2
CALL MAG ANGLE(EXTRA, EXTRA MAG, EXTRA ANG)
EXTRA ANG = EXTRA ANG*DEG TO RAD
CALL MAG ANGLE(WAITS EF, WAITS MAG, WAITS ANG)
WAITS ANG = WAITS ANG*DEG TO RAD
WAIT EF DB = 20.0*LOGF(WAITS MAG)/LOGE10
X(5) = EXTRA MAG
X(6) = EXTRA ANG
X(7) = WAITS ANG

```

```

X(8) = WAIT EF DB
STOKE2 = -1.0/S*R21*(1.0+RBAR11)*(1.0+RBAR22)*SCRIP(J)*STFACR(M)/
S   (DFDTHETA/DEG TO RAD)
EXTRA = STOKE2*S*S*SQRT S
ID = 2HVY
XCVY = EXTRA/(SCRIP(J)*STFACR(M))
HGRY = STFACR(M)
IF(PRIIPUN, EQ, 1) PUNCH 503, EXTRA, ID, ID2
WAITS EF = -I*K*REFL HT/2.0*EXTRA
WAITS EXC = 0.06496*K/(2.0*SORTF(FREQ KHZ))*EXTRA
IF(PRIIPUN, EQ, 2) PUNCH 503, WAITS EXC, ID, ID2
CALL MAG ANGLE(EXTRA, EXTRA MAG, EXTRA ANG)
EXTRA ANG = EXTRA ANG*DEG TO RAD
CALL MAG ANGLE(WAITS EF, WAITS MAG, WAITS ANG)
WAITS ANG = WAITS ANG*DEG TO RAD
WAIT EF DB = 20.0*LOGF(WAITS MAG)/LOGE10
X(9) = EXTRA MAG
X(10) = EXTRA ANG
X(11) = WAITS ANG
X(12) = WAIT EF DB
ID2 = 1HE
DO 21 L = 1, NR PHIS, 1
XTRA1 = -I*(SINF(PHI(L))*R12*(1.0+RBAR22)*(1.0+RBAR11)*ST FAC(J)
S   *SCRIPT H+COSF(PHI(L))*(1.0+RBAR11)**2*(1.0+RBAR22*R22)
S   * SG FAC(J)*SCRIP1 H/RBAR11)/(DFDTHETA/DEG TO RAD)*S*SQRT S
ID = 2HHz
IF(L, EQ, 1) XHEZ = XTRA1/(SCRIPTH*SGFAC(J))
XHBZ = XTRA1/(SCRIPTH*STFAC(J))
IF(L, EQ, 1) HGTHE = SGFAC(J)
HGTHB = STFAC(J)
IF(PRIIPUN, EQ, 1) PUNCH 503, XTRA1, ID, ID2
XTRA2 = -I*K*REFL HT/2.0*XTRA1
WAITS EXC = 0.06496*K/(2.0*SORTF(FREQ KHZ))*XTRA1
IF(PRIIPUN, EQ, 2) PUNCH 503, WAITS EXC, ID, ID2
CALL MAG ANGLE(XTRA1, XTRA1MAG, XTRA1ANG)
CALL MAG ANGLE(XTRA2, XTRA2MAG, XTRA2ANG)
XTRA1ANG = XTRA1ANG*DEG TO RAD
XTRA2ANG = XTRA2ANG*DEG TO RAD
XTRA2DB = 20.0*LOGF(XTRA2MAG)/LOGE10

X(13) = XTRA1 MAG
X(14) = XTRA1 ANG
X(15) = XTRA2 ANG
X(16) = XTRA2 DB
XTRAX = -1.0/S* I*(SINF(PHI(L))*R12*(1.0+RBAR22)*(1.0+RBAR11)*
S   STFACR(J)*SGFACR(M)+COSF(PHI(L))*(1.0+RBAR11)**2/RBAR11*
S   (1.0+RBAR22*R22)*SGFAC(J)*SGFACR(M))/(DFDTHETA/
S   DEG TO RAD)*S*SQRTS
ID = 2HMX
IF(L, EQ, 1) XHEX = XTRAX/(SGFAC(J)*SGFACR(M))
XHBX = XTRAX/(STFAC(J)*SGFACR(M))
IF(PRIIPUN, EQ, 1) PUNCH 503, XTRAX, ID, ID2
WAITS EF = -I*K*REFL HT/2.0*XTRAX
WAITS EXC = 0.06496*K/(2.0*SORTF(FREQ KHZ))*XTRAX
IF(PRIIPUN, EQ, 2) PUNCH 503, WAITS EXC, ID, ID2
CALL MAG ANG(XTRAX, EXTRA MAG, EXTRA ANG)

```

```

EXTRA ANG = EXTRA ANG*DEG TO RAD
CALL MAG ANGLE(WAITS EF,WAITS MAG,WAITS ANG)
WAITS ANG = WAITS ANG*DEG TO RAD
WAIT EF DB = 20.0*LOGF(WAITS MAG)/LOGE10
X(17)= EXTRA MAG
X(18)= EXTRA ANG
X(19)= WAITS ANG
X(20)= WAIT EF DB
XTRAY = 1.0/S* I*(SINF(PHI(L))*(1.0+RBAR22)**2/RBAR22*(1.0-RBAR11*
$      R11)*STFAC(J)*STFACR(M)+COSF(PHI(L))*R21*(1.0+RBAR11)*
$      (1.0+RBAR22)*SGFAC(J)*STFACR(M))/(DFDIHEIA/DEG TO RAD)
$      *S*SQRTS
ID = 2HHY
IF(L,EQ,1) XHEY = XTRAY/(SGFAC(J)*STFACR(M))
XHBY = XTRAY/(STFAC(J)*STFACR(M))
IF(PRIIPUN,EQ, 1) PUNCH 503,XTRAY, ID, ID2
WAITS EF = -I*K*REFL HT/2.0*XTRAY
WAITS EXC = 0.06496*K/(2.0*SQRTF(FREQ KHZ))*XTRAY
IF(PRIIPUN,EQ, 2) PUNCH 503,WAITS EXC, ID, ID2
CALL MAG ANG(XTRAY,EXTRA MAG,EXTRA ANG)
EXTRA ANG = EXTRA ANG*DEG TO RAD
CALL MAG ANGLE(WAITS EF,WAITS MAG,WAITS ANG)
WAITS ANG = WAITS ANG*DEG TO RAD
WAIT EF DB = 20.0*LOGF(WAITS MAG)/LOGE10
X(21)= EXTRA MAG
X(22)= EXTRA ANG
X(23)= WAITS ANG
X(24)= WAIT EF DB
IF(L,EQ,1) GO TO 72
IF(PRIIPUN,EQ,3) PUNCH 41,HGTVR, IHGTVR, HGTHE, IHGTHE, HGIHB, IHGTHB
IF(PRIIPUN,NE,4) GO TO 72
PUNCH 41,HGRZ, IHGRZ, HGRX, IHGRX, HG RY, IHGRY
PUNCH 4, XCVZ, IXCVZ, XCVX, IXCVX, XCVY, IXCVY,
$XHEZ, IXHEZ, XHEX, IXHEX, XHEY, IXHEY, XHBZ, IXHBZ, XHBX, IXHBX, XHBY, IXHBY
41 FORMAT(1X,C(E12.5,E12.5),10X,A5,2(/,1X,C(E12.5,E12.5),10X,A5))
4  FORMAT(1X,C(E12.5,E12.5),10X,A5,8(/,1X,C(E12.5,E12.5),10X,A5))
72 CONTINUE
IF(PRIIPUN,NE,0) GO TO 25
CALL MAG ANGLE(HGTVR,HGTVRM,HGTVRA)
CALL MAG ANGLE(HGTHE,HGTHEM,HGTHEA)
CALL MAG ANGLE(HGTHB,HGTHBM,HGTHBA)
CALL MAG ANGLE(HGRZ ,HG RZM,HG RZA)
CALL MAG ANGLE(HGRX ,HG RXM,HG RXA)
CALL MAG ANGLE(HGRY ,HG RYM,HG RYA)
CALL MAG ANGLE(XCVZ,XCVZM,XCVZA)
CALL MAG ANGLE(XCVX,XCVXM,XCVXA)
CALL MAG ANGLE(XCVY,XCVYM,XCVYA)
CALL MAG ANGLE(XHEZ,XHEZM,XHEZA)
CALL MAG ANGLE(XHEX,XHEXM,XHEXA)
CALL MAG ANGLE(XHEY,XHEYM,XHEYA)
CALL MAG ANGLE(XHBZ,XHBZM,XHBZA)
CALL MAG ANGLE(XHBX,XHBXM,XHBXA)
CALL MAG ANGLE(XHBY,XHBYM,XHBYA)
O-HGTVR = HGTVRM + I*HGTVRA
O-HGTHE = HGTHEM + I*HGTHEA
O-HGTHB = HGTHBM + I*HGTHBA

```

```

OHGRZ = HGRZM + I*HGRZA
OHGRX = HGRXM + I*HGRXA
OHGRY = HGRYM + I*HGRYA
OXCZV = XCVZM + I*XCVZA
OXCZVX = XCVXM + I*XCVXA
OXCZVY = XCVYM + I*XCVYA
OXHEZ = XHEZM + I*XHEZA
OXHEX = XHEXM + I*XHEXA
OXHEY = XHEYM + I*XHEYA
OXHBZ = XHBZM + I*XHBZA
OXHBX = XHBXM + I*XHBXA
OXHBY = XHBYM + I*XHBYA
PRINT 291
IF(L.EQ,1) PRINT 271
PRINT 203, PHI(L)
PRINT 272
PRINT 291
IF(L.EQ,2) PRINT 330
PRINT 280
IF(L.EQ,2) PRINT 320, IHGTVR, OHGTVR
PRINT 350
IF(L.EQ,2) PRINT 320, IHGTHE, OHGTHE
PRINT 350
IF(L.EQ,1) PRINT 204, X(1)
IF(L.EQ,2) PRINT 320, IHGTHB, OHGTHB
PRINT 104, X(13)
IF(L.EQ,1) PRINT 205, X(2)
IF(L.EQ,2) PRINT 320, IHGRZ , OHGRZ
PRINT 105, X(14)
IF(L.EQ,1) PRINT 212, X(3)
IF(L.EQ,2) PRINT 320, IHGRX , OHGRX
PRINT 112, X(15)
IF(L.EQ,1) PRINT 213, X(4)
IF(L.EQ,2) PRINT 320, IHGRY , OHGRY
PRINT 113, X(16)
PRINT 291
IF(L.EQ,2,AND,D.EQ,0.0)PRINT 572
PRINT 281
IF(L.EQ,2,AND,D.EQ,0.0)PRINT 320, IXCVZ, OXCZV
PRINT 350
IF(L.EQ,2,AND,D.EQ,0.0)PRINT 320, IXCVX, OXCZVX
PRINT 350
IF(L.EQ,1) PRINT 204, X(5)
IF(L.EQ,2,AND,D.EQ,0.0)PRINT 320, IXCVY, OXCZVY
PRINT 104, X(17)
IF(L.EQ,1) PRINT 205, X(6)
IF(L.EQ,2,AND,D.EQ,0.0)PRINT 320, IXHEZ, OXHEZ
PRINT 105, X(18)
IF(L.EQ,1) PRINT 212, X(7)
IF(L.EQ,2,AND,D.EQ,0.0)PRINT 320, IXHEX, OXHEX
PRINT 112, X(19)
IF(L.EQ,1) PRINT 213, X(8)
IF(L.EQ,2,AND,D.EQ,0.0)PRINT 320, IXHEY, OXHEY
PRINT 113, X(20)
IF(L.EQ,2,AND,D.EQ,0.0)PRINT 320, IXHBZ, OXHBZ
PRINT 350

```

```

IF(L.EQ,2,AND,D.EQ,0.0)PRINT 320,IXHBMX,OXHBMX
PRINT 350
IF(L.EQ,2,AND,D.EQ,0.0)PRINT 320,IXHBY,OXHBY
PRINT 282
PRINT 291
IF(L.EQ,1) PRINT 204, X(9)
PRINT 104,X(21)
IF(L.EQ,1) PRINT 205, X(10)
PRINT 105,X(22)
IF(L.EQ,1) PRINT 212, X(11)
PRINT 112,X(23)
IF(L.EQ,1) PRINT 215, X(12)
PRINT 113,X(24)
291 FORMAT(1H0)
271 FORMAT(1H+,50X,10X,15HVERTICAL DIPOLE)
203 FORMAT(1H+,25X,6HPHI = ,F10.5,4H RAD)
280 FORMAT(1H ,40X,7HE SUB Z)
281 FORMAT(1H ,40X,7HE SUB X)
282 FORMAT(1H ,40X,7HE SUB Y)
320 FORMAT(1H+,57X,A5,2H =,C(E12.5,E14.5))
330 FORMAT(1H+,60X,21HHEIGHT GAIN (MAG ANG))
350 FORMAT(1H )
572 FORMAT(1H+,60X,28HEXCITATION FACTORS (MAG ANG))
104 FORMAT(1H ,2X,12HEXTRA MAG = ,E12.5)
272 FORMAT(1H ,5X,17HHORIZONTAL DIPOLE)
105 FORMAT(1H ,2X,14HEXTRA ANGLE = ,F10.5,4H RAD)
112 FORMAT(1H ,2X,35HPHASE OF WAITS EXCITATION FACTOR = ,F10.5,4H RAD)
113 FORMAT(1H ,2X,32HWAITS EXCITATION FACTOR IN DB = ,E12.5,3H DB)
204 FORMAT(1H+,55X,2X,12HEXTRA MAG = ,E12.5)
205 FORMAT(1H+,55X,2X,14HEXTRA ANGLE = ,F10.5,4H RAD)
212 FORMAT(1H+,55X,2X,35HPHASE OF WAITS EXCITATION FACTOR = ,F10.5,
$      4H RAD)
213 FORMAT(1H+,55X,2X,32HWAITS EXCITATION FACTOR IN DB = ,E12.5,3H DB)
25 CONTINUE
   ID2 = 1HB
21 CONTINUE
   PRINT 200
200 FORMAT(1H0)
   RETURN

END

```

SUBROUTINE R OUT

```
COMMON/R/LOG R11,R11 ANG,LOG R22,R22 ANG,
$      LUG R12,R12 ANG,LOG R21,R21 ANG,
$      R OMIT(18)
COMMON/EN COLL/HT,EC OMIT(10)
COMMON/FIT PROF/V(4)
REAL LOG R11,LOG R22,LOG R12,LOG R21
DATA(RTD=57.29577951)
```

```
ENTRY R COLS
PRINT 100
100 FORMAT(1H0,8X,2HHT,10X,2H11R11,17X,3H1R1,16X,4H1R11,16X,4H11R1)
RETURN
```

```
ENTRY PRINT R
R11 MAG = EXPF(LOG R11)
R22 MAG = EXPF(LOG R22)
R12 MAG = EXPF(LOG R12)
R21 MAG = EXPF(LOG R21)
DEG11 = R11 ANG*RTD
DEG22 = R22 ANG*RTD
DEG12 = R12 ANG*RTD
DEG21 = R21 ANG*RTD
```

```
PRINT 200,HT,R11 MAG,DEG11,R22 MAG,DEG 22,R12 MAG,DEG12,
$      R21 MAG,DEG21
200 FORMAT(1H ,F10.2,4(F10.4,F10.2))
RETURN
```

```
ENTRY RAT DIF
RATIO = R12 MAG/R22 MAG
DIFF = DEG12-DEG22

PRINT 301,RATIO
301 FORMAT(1H0,19H1R11 MAG/1R1 MAG = ,E12.5)
PRINT 302,DIFF
302 FORMAT(1H ,23H1R11 ANGLE-1R1 ANGLE = ,F11.5)
PRINT 303
303 FORMAT(1H1)
RETURN
```

```
ENTRY V OUT
R22 MAG = EXPF(LOG R22)
R12 MAG = EXPF(LOG R12)
RATIO = R12 MAG/R22 MAG
V(1) = RATIO
DEG22 = R22 ANG*RTD
DEG12 = R12 ANG*RTD
DIFF = DEG12-DEG22
V(2) = DIFF
```

1969 289

V(3) = R22 MAG
R11 MAG = EXPF(LOG R11)
V(4) = R11 MAG
RETURN

END

SUBROUTINE AG OUTPUT

```

COMMON/INPUT/THETA,THETA IMG,IN OMIT(98)
COMMON/R/R SKIP(18),R(4)
COMMON/F/F,DFDTHETA,R BAR(2)
COMPLEX R BAR,R,ALL RS,1,F,DFDTHETA
REAL IMAG PART,MAG
DATA(1)=(0,0,1,0)
DIMENSION ALL RS(8),MAG(8),ANGLE(8),REAL PART(8),IMAG PART(8)

ENTRY PRINT RS
D) 11 J=1,2
11 ALL RS(J) = R BAR(J)
D) 12 J=1,4
12 ALL RS(J*2) = R(J)
   ALL RS(7) = R BAR(1)*H(1)
   ALL RS(8) = R BAR(2)*H(2)

D) 13 J=1,8
   CALL MAG ANGLE(ALL RS(J),MAG(J),ANGLE(J))
   REAL PART(J) = ALL RS(J)
13 IMAG PART(J) = -1*ALL RS(J)

PRINT 200
200 FORMAT(1H0,8X,5THETA,5X,9H1R11 BAR,6X,7H1R1 BAR,8X,5H11P11,
$      10X,3H1R1,9X,4H1R11,9X,4H11P1,2X,11H11 BAR*H11,
$      4X,9HR1 BAR*R1/)
PRINT 201,THETA,(REAL PART(J),J=1,8)
201 FORMAT(1H ,4HREAL,F9.3,1X,8E13.5)
PRINT 202,THETA IMG,(IMAG PART(J),J=1,8)
202 FORMAT(1H ,4HIMAG,F9.3,1X,8E13.5)
PRINT 203,(MAG(J),J=1,8)
203 FORMAT(1H ,5HMAG,11X,8F13.5)
PRINT 204,(ANGLE(J),J=1,8)
204 FORMAT(1H ,5HANGLE, 9X,8F13.5)
RETURN

ENTRY PRI THETA
CALL MAG ANGLE(F,F MAG,F ANGLE)
CALL MAG ANGLE(DFDTHETA,DFDT MAG,DFDT ANGLE)
PRINT 500,THETA,THETA IMG,F MAG,DFDT MAG
500 FORMAT(1H0,5THETA = ,2F10.3,10X,8HF MAG = ,E10.3,2X,
$      15HDFDTHETA MAG = ,E10.3)
RETURN

END

```

SUBROUTINE FIELDS

```

COMMON/INPUT/RE THETA,IM THETA,IN OMIT(98)
COMMON/PROP DATA/WAITS EXC,KSM1,NO SOL
COMMON/FLD INPUT/DIST,DEL DIST,ITITLE1(12),ITITLE2(12)
COMMON/FLG INPUT/FLG SKIP(9),GRAPHS,FG OMIT(15)
COMPLEX I,E,WAITS EXC,KSM1,CX EXPF
REAL MAG,IM T LIST,IM THETA
INTEGER GRAPHS
DIMENSION D(500),E(500),RE T LIST(10),IM T LIST(10),
$      E REAL(2),BUFFER(2000)
DATA(I*(0.0,1.0))
DATA(IABS=16HRHO IN MM )
DATA(IORD1=16FIELD DB/UV/M/KW)
DATA(IORD2=16PHASE IN DEGREES)
EQUIVALENCE(E,E REAL)

IF(DEL DIST .EQ. 0.0) RETURN
MM1 = M-1
IF(MM1 .EQ. 0) GO TO 11
IF(NO SOL .EQ. 1) GO TO 13
DO 10 L=1,MM1
IF(ABSF(RE THETA-RE T LIST(L)) .LT. 0.05 .AND. ABSF(IM THETA-
$      IM T LIST(L)) .LT. 0.005) 13,11
10 CONTINUE
11 RE T LIST(M) = RE THETA
IM T LIST(M) = IM THETA
M = M+1
DO 12 K=1,NR DIST
12 E(K) = E(K)+1.0/SQRTF(SINF(D(K)/6370.0))*WAITS EXC*
$      CX EXPF(-1*KSM1*D(K))
13 RETURN

ENTRY INT FLD
ENTRY INIT FLDS
IF(DEL DIST .EQ. 0.0) RETURN
M = 1

L = 1
D(1) = DEL DIST
E(1) = 0.0
IF(D(1) .GE. DIST) GO TO 22

L = 2
21 D(L) = D(L-1)+DEL DIST
E(L) = 0.0
IF(D(L) .GE. DIST) GO TO 22
L = L+1
GO TO 21

22 NR DIST = L
RETURN

```

```

ENTRY FLD OJT
ENTRY PRNT FLDS
IF(DELDIST .EQ. 0.0) RETURN
PRINT 300
300 FORMAT(1H1,7HD IN KM,5X,11HTOTAL FIELD,4X,9HREL PHASE,3X,
$          17HFIELD DB/10V/M/KW)
DO 30 L=1,NR DIST
CALL MAG ANG(E(L),MAG,ANGLE)
DB = 20.0*LJGF(MAG*1.0E06)/2 302589093
E(L) = DB+I*ANGLE
30 PRINT 301,D(L),MAG,ANGLE,DB
301 FORMAT(1H0,F8.0,5X,E11.4,5X,F8.2,5X,F8.2)

IF(GRAPH5 .EQ. 0) RETURN
DO 42 L=2,NR DIST
L2M1 = L*2-1
TEMP = E REAL(L2M1)
LP1 = L+1
DO 41 K=LP1,L2M1
KK = L2M1+(-P1-K)
41 E REAL(KK) = E REAL(KK-1)
42 E REAL(L) = TEMP

DO 43 L=1,NR DIST
LL = NR DIST+NR DIST+(1-L)
43 E REAL(LL+2) = E REAL(LL)

DO 44 L=1,NR DIST
44 D(L) = D(L)/1000.0

INTGR = D(NR DIST)
WIDE = 1.0*FLOATF(INTGR)
CALL CALCOMP(D,E REAL(1),NR DIST,0,ITITLE1,72,1,0.0,
$          0.0,1.0,0.0,0.0,0.0,0.0,0.0,
$          WIDE,10.0,IABS,9,IORD1,16,10.0,1,BUFFER,2000)
CALL CALCOMP(D,E REAL(NR DIST+3),NR DIST,0,ITITLE2,72,1,0.0,
$          0.0,1.0,0.0,0.0,0.0,0.0,0.0,
$          WIDE,10.0,IABS,9,IORD2,16,10.0,11,BUFFER,2000)

DO 47 L=1,NR DIST
47 D(L) = D(L)*1000.0
RETURN

END

```

SUBROUTINE ITERATE

```

COMMON/INPUT/THETA,IN OMIT(9#)
COMMON/WG INPUT/TYPE ITER,MAX DELTA,WGIN SKIP(88),IHEIA INC,
$      WGIN OMIT(2),DTHETA,LUB REAL,LUB IMAG,WGIN NO(3)
COMMON/R/R SKIP(18),R11,R22,R OMIT(4)
COMMON/F/F,DFDTHETA,R BAR11,R BAR22
COMMON/PROP DATA/PD SKIP(4),NO SOL
COMPLEX IHEIA,F0,F,DFDTHETA,DELTHETA,I,R11,R22,R BAR11,R BAR22,
$      EIGEN,DTHETA
REAL MAX DELTA,LUB REAL,LUB IMAG
INTEGER TYPE ITER,FIRST,FIXED INC
DATA(MAX ITER=20)
DATA(I=(0.0,1.0))

PRINT 101,TYPE ITER
101 FORMAT(1H1,14HITERATION TYPE ,I3)
PRINT 102,MAX DELTA
102 FORMAT(1H ,12HMAX DELTA = ,F5.2)
FIRST = 1
NR ITER = 0
NO SOL = 0

11 THETA = IHEIA-DTHETA
CALL COMPUTE F
F0 = F
THETA = IHEIA+DTHETA
CALL COMPUTE F
DFDTHETA = (F-F0)/DTHETA
DELTHETA = -F/DFDTHETA
FIXED INC = 1
GO TO 20

15 F0 = F
CALL COMPUTE F
DFDTHETA = (F-F0)/DELTHETA
DELTHETA = -F/DFDTHETA
FIXED INC = 0

20 IF(FIRST .EQ. 0) GO TO 21
CALL PRINT RS $ CALL PRT THETA $ FIRST = 0
21 DEL REAL = DELTHETA $ ABS REAL = ABSF(DEL REAL)
IF(ABS REAL .GT. THETA INC) DELTHETA = DELTHETA*THETA INC/ABS REAL
DEL IMAG = -I*DELTHETA $ ABS IMAG = ABSF(DEL IMAG)
IF(ABS IMAG .GT. THETA INC) DELTHETA = DELTHETA*THETA INC/ABS IMAG
THETA = IHEIA+DELTHETA
CALL PRT THETA
NR ITER = NR ITER+1
IF(NR ITER .GT. MAX ITER) NO SOL = 1
IF(NR ITER .GT. MAX ITER) GO TO 30
DEL REAL = DELTHETA $ DEL IMAG = -I*DELTHETA
ABS REAL = ABSF(DEL REAL) $ ABS IMAG = ABSF(DEL IMAG)
IF(ABS REAL .LE. LUB REAL .AND. ABS IMAG .LE. LUB IMAG) GO TO 30
IF(ABS REAL .LE. MAX DELTA .AND. ABS IMAG .LE. MAX DELTA) 15,11

```

```
30 IF(FIXED INC .EQ. 1) GO TO 31
   THETA = THETA-DTHETA
   CALL COMPUTE F
   FO = F
   THETA = THETA+DTHETA
   CALL COMPUTE F
   DFDTHETA = (F-FO)/DTHETA
31 IF(TYPE ITER ,EQ. 1) DFDTHETA = (R BAR22*R22-1.0)*DFDTHETA
   IF(TYPE ITER ,EQ. 2) DFDTHETA = (R BAR11*R11-1.0)*DFDTHETA

   CALL PRINT 4S
   PRINT 501,NR ITER
501 FORMAT(1H0,23HITERATIONS PERFORMED = ,I2)
   PRINT 502,DELTHETA
502 FORMAT(1H0,11HDELTHETA = ,C(F11.7,F11.7))
   PRINT 503,F
503 FORMAT(1H0,13HF FUNCTION = .C(F13.5,F13.5))
   RETURN

END
```

SUBROUTINE F FCT

```

COMMON/WG INPUT/ITER,WG SKIP(6),APPROX,WG OMIT(92)
COMMON/R/R SKIP(18),R11,R22,R12,R21
COMMON/F/F,F SKIP(2),R BAR11,R BAR22
COMMON/INPUT/THETA,IN OMIT(92)
COMMON/SOLTVS/NR SOL,SOLTN(25)
COMPLEX R11,R22,R12,R21,F,R BAR11,R BAR22,THETA,SOLTN
INTEGER APPROX
DATA(DTR=0.01745329252)
DATA(NR SOL=0)

```

```

ENTRY COMPUTE F
CALL R BARS

```

```

IF(APPROX .EQ. 0) CALL INTEG
IF(APPROX .EQ. 1) CALL USE POLY

```

```

IF(ITER .EQ. 0) F = (R BAR11*R11-1.0)*(R BAR22*R22-1.0)
S      -R BAR11*R BAR22*R12*R21
IF(ITER .EQ. 1) F = R BAR11*R11-1.0
IF(ITER .EQ. 2) F = R BAR22*R22-1.0

```

```

IF(NR SOL .EQ. 0) RETURN
DO 21 J=1,NR SOL
21 F = F/((THETA-SOLTN(J))*DTR)

```

```

RETURN
END

```

SUBROUTINE R BARS

```

COMMON/INPUT/THETA,FREQ KHZ,IN SKIP(3),ALPHA,H,D,IN OMIT(91)
COMMON/WG INPUT/WG SKIP(2),EPSILON,EPSILON 0,SIGMA,WG OMIT(95)
COMMON/F/F SKIP(4),R BAR11,R BAR22
COMMON/RB CP/ SH FAC(25),
S      SCRIT(25),SG FAC(25),ST FAC(25),SGFACR(25),STFACR(25)
COMMON/HD INPUT/ALT(25),NR ALTS,PHI(10),NR PHIS,HD FLAG,NR ALR,
S      ALR(25),HD SKIP(10)
S      ,PRIPUN
COMPLEX THETA,SH FAC,I,NG SQ,C,S,SQ ROOT,RATIO RT,
S      PD,QD,H1 D,H2 D,H1 PRIME D,H2 PRIME D,CAP H1 D,CAP H2 D,
S      P0,Q0,H1 0,H2 0,H1 PRIME 0,H2 PRIME 0,CAP H1 0,CAP H2 0,
S      A1ST,A2ND,A1,A2,A3,A4,R BAR11,R BAR22,
S      C POWER,Z1,Z2,C TEMP,
S      CPLX SORT,CPLX EXPF,CPLX SIN, C, CPLX COSF,
S      PT,H1T,H2T,H1TPRIME,H2TPRIME,A3RD,A4TH,SI FAC,
S      SG FAC,SCRIT,SGFACR,STFACR
REAL K,LOG K OVR A,K OVER A OT,K OVER A TT,ND SQ,NU SQ
INTEGER HD FLAG
INTEGER PRIPUN
DATA(PI=3.141592653)
DATA(VEL LIGHT=2.997928E05)
DATA(I=(0.0,1.0))
DATA(DEG TO RAD=0.01745329252)
DATA(TEST TH IM=10.0)

```

```

OMEGA = 2.0*PI*FREQ KHZ*1000.0
K = OMEGA/VEL LIGHT
NG SQ = (EPSILON-I*SIGMA/OMEGA)/EPSILON 0

```

```

C = CPLX COSF(THETA*DEG TO RAD)
S = CPLX SIN(THETA*DEG TO RAD)

```

```

LOG K OVR A = LOGF(K/ALPHA)
K OVER A OT = EXPF(LOG K OVR A/3.0)
K OVER A TT = K OVER A OT**2
A OVER K OT = 1.0/K OVER A OT
A OVER K TT = A OVER K OT**2

```

```

ND SQ = 1.0-ALPHA*(H-D)
N0 SQ = 1.0-ALPHA*H
SQ ROOT = CPLX SORT(NG SQ-S**2)
RATIO RT = N0 SQ/NG SQ*SQ ROOT

```

```

PD = K OVER A TT*(1.0-ALPHA*(H-D)-S**2)
P0 = K OVER A TT*(1.0-ALPHA*H-S**2)
QD = PD
Q0 = P0

```

```

THETA IM = -I*THETA
IF(=THETA IM .GT. TEST TH IM) GO TO 50
CALL MD HANKEL(QD,H1 D,H2 D,H1 PRIME D,H2 PRIME D)
CALL MD HANKEL(Q0,H1 0,H2 0,H1 PRIME 0,H2 PRIME 0)

```

```

CAP H1 D = H1 PRIME D*A OVER K TT*H1 D/2.0
CAP H1 0 = H1 PRIME 0*A OVER K TT*H1 0/2.0
CAP H2 D = H2 PRIME D*A OVER K TT*H2 D/2.0
CAP H2 0 = H2 PRIME 0*A OVER K TT*H2 0/2.0

```

```

A1ST = CAP H2 0-I*RATIO RT*K OVER A OT*H2 0
A2ND = CAP H1 0-I*RATIO RT*K OVER A OT*H1 0
A1 = C*ND SJ*(-H1 D*A1ST+H2 D*A2ND)
A3RD = H2 PRIME 0 -I*K OVER A OT*SQ ROOT*H2 0
A4TH = H1 PRIME 0 -I*K OVER A OT*SQ ROOT*H1 0
DO 31 L = 1, NR ALR, 1
PT = K OVER A TT*(1.0-ALPHA*(H-ALR(L))-S**2)
CALL MD HANKEL (PT, H1T, H2T, H1T PRIME, H2T PRIME)
ST FACR(L) = (-A3RD*H1T + A4TH*H2T)/(-A3RD*H1D + A4TH*H2D)
SH FAC(L) = (-A1ST*H1T+A2ND*H2T)/(-A1ST*H1D+A2ND*H2D)
SG FACR(L) = (-I*A OVER K OT *(-A1ST*H1T PRIME + A2ND*H2T PRIME)
$ -I*ALPHA/(2.0*K)*(-A1ST*H1T+A2ND*H2T))*EXPF(=ALPHA
$ *(D-ALR(L))/2.0)/(-A1ST*H1D+A2ND*H2D)
DO 31 J = 1, NR ALTS, 1
PT = K OVER A TT*(1.0-ALPHA*(H-ALT(J))-S**2)
CALL MD HANKEL (PT, H1T, H2T, H1T PRIME, H2T PRIME)
ST FAC(J) = (-A3RD*H1T + A4TH*H2T)/(-A3RD*H1D + A4TH*H2D)
SG FAC(J) = (-I*A OVER K OT *(-A1ST*H1T PRIME + A2ND*H2T PRIME)
$ -I*ALPHA/(2.0*K)*(-A1ST*H1T+A2ND*H2T))*EXPF(=ALPHA
$ *(D-ALT(J))/2.0)/(-A1ST*H1D+A2ND*H2D)
SCRIT(J) = (-A1ST*H1T + A2ND*H2T)/(-A1ST*H1D + A2ND*H2D)
$ *EXPF(ALPHA/2.0*(ALT(J)-D))

```

```
31 CONTINUE
```

```

32 A1ST = -I*A OVER K UT*CAP H2 0-RATIO RT*H2 0
A2ND = -I*A OVER K UT*CAP H1 0-RATIO RT*H1 0
A2 = -CAP H1 D*A1ST+CAP H2 D*A2ND
R BAR11 = (A1-A2)/(A1+A2)

```

```

A1ST = I*A OVER K OT*H2 PRIME 0+SQ ROOT*H2 0
A2ND = I*A OVER K OT*H1 PRIME 0+SQ ROOT*H1 0
A3 = -H1 PRIME D*A1ST+H2 PRIME D*A2ND
A1ST = H2 PRIME 0-I*K OVER A OT*SQ ROOT*H2 0
A2ND = H1 PRIME 0-I*K OVER A OT*SQ ROOT*H1 0
A4 = C*(-H1 D*A1ST+H2 D*A2ND)
R BAR22 = (A3+A4)/(A4-A3)
IF(D, EQ, 0.0) GO TO 50
RETURN

```

```
C FLAT EARTH
```

```

50 C POWER = CPLX EXPF(-2.0*I*K*D*C)
DO 139 L = 1, NR ALR, 1
R BAR11 = (NG SQ*C-SQ ROOT)/(NG SQ*C+SQ ROOT)*C POWER
R BAR22 = (C-SQ ROOT)/(C+SQ ROOT)*C POWER
IF(-THETA IM .GT. TEST TH IM) GO TO 25
RETURN
25 SH FAC(L) = (CPLXEXPF(I*K*C*ALR(L))*RBAR11+CPLXEXPF(-I*K*C*
$ ALR(L)+2.0*I*K*C*D))/(CPLXEXPF(I*K*C*D)*(1.0+RBAR11))
ST FACR(L) = (CPLXEXPF(I*K*C*ALR(L))*RBAR22+CPLXEXPF(-I*K*C*

```

```

$          ALR(L)+2.0*I*K*C*D))/(CPLXEXP(I*K*C*D)*(1.0+RBAR22))
SG FACR(L) = (C*(CPLXEXP(I*K*C*ALR(L))-RBAR11*CPLXEXP(-I*K*C*
$          ALR(L)+2.0*I*K*C*D)))/(CPLXEXP(I*K*C*D)*(1.0+RBAR11))
DO 139 J = 1, NR ALTS, 1
ST FAC(J) = (CPLXEXP(I*K*C*ALT(J))+RBAR22*CPLXEXP(-I*K*C*
$          ALT(J)+2.0*I*K*C*D))/(CPLXEXP(I*K*C*D)*(1.0+RBAR22))
SG FAC(J) = (C*(CPLXEXP(I*K*C*ALT(J))-RBAR11*CPLXEXP(-I*K*C*
$          ALT(J)+2.0*I*K*C*D)))/(CPLXEXP(I*K*C*D)*(1.0+RBAR11))
SCHIT(J) = (CPLXEXP(I*K*C*ALT(J))+RBAR11*CPLXEXP(-I*K*C*ALT(J)
$          +2.0*I*K*C*D))/(CPLXEXP(I*K*C*D)*(1.0+RBAR11))
139 CONTINUE
RETURN

END

```

SUBROUTINE MD HANKEL(Z,H1,H2,H1 PRIME,H2 PRIME)

COMPLEX Z,I,H1,H2,H1 PRIME,H2 PRIME,ZCUBED,Z POWER,TERM1,TERM2,
 \$ TERM3,TERM4,TERM5,TERM6,SUM1,SUM2,SUM3,SUM4,SUM5,SUM6,
 \$ SUM7,SUM8,F,Q,F PRIME,G PRIME,ONE RT Z,CPLX SQRT,
 \$ Z FOURTH,SQRT Z CUH, EXP1,EXP2,EXP3,EXP4,EXP5,
 \$ CPLX EXPF,BETA,RT Z,GM2F,GMFP,I POWER,M I POWER,
 \$ Z M3HALVS,Z M3HAFS M

DIMENSION A(23),B(23),C(23),D(23),CAP(14)

DATA(A= 0.7304 3671 693, 11.0145 5723 097, 206.7637 1487 316,
 \$ 574.3436 5242 545, 870.2176 5519 008, 828.7787 1922 864,
 \$ 541.6854 3740 434, 257.9454 4638 302, 93.4584 9506 631,
 \$ 26.6263 5187 074, 6.1210 0043 0056, 1.1592 8038 4480,
 \$ 0.1840 1275 9441, 0.0248 5303 0964, 0.0028 8420 8010,
 \$ 0.0002 9133 4142, 0.0000 2582 7495, 0.0000 0202 5686,
 \$ 0.0000 0014 1557, 0.0000 0000 8870, 0.0000 0000 0501,
 \$ 0.0000 0000 0026, 0.0000 0000 0001)

DATA(B= 0.6782 9872 514, 11.3049 7875 240, 53.8332 3215 431,
 \$ 119.6294 0478 735, 143.3710 3177 865, 127.8091 9314 888,
 \$ 74.7422 1821 572, 42.3559 3862 152, 10.7853 1287 384,
 \$ 2.8532 5737 403, 0.6136 0373 6351, 0.1093 7678 98,
 \$ 0.0164 2293 9955, 0.0021 0550 5122, 0.0002 3316 7788,
 \$ 0.0000 2252 8289, 0.0000 0191 5671, 0.0000 0014 4470,
 \$ 0.0000 0000 9729, 0.0000 0000 0589, 0.0000 0000 0032,
 \$ 0.0000 0000 0002, 0.0000 0000 0000)

DATA(C= 0.4652 1835 846, 6.2029 1144 619, 25.8454 6435 915,
 \$ 22.2130 5931 140, 42.1584 0394 215, 48.7516 8936 639,
 \$ 27.0842 7187 022, 11.2150 1940 796, 3.5945 5750 255,
 \$ 0.9181 5006 451, 0.1912 8126 3439, 0.0331 2229 6699,
 \$ 0.0048 4244 1038, 0.0006 0568 3682, 0.0000 6555 0182,
 \$ 0.0000 0619 8599, 0.0000 0051 6550, 0.0000 0003 8220,
 \$ 0.0000 0000 2528, 0.0000 0000 0150, 0.0000 0000 0008,
 \$ 0.0000 0000 0000, 0.0000 0000 0000)

DATA(D= 0.6782 9872 514, 45.2199 1500 962, 376.8326 2508 015,
 \$ 1196.2940 4787 350, 1943.8234 1312 250, 2044.9470 9038 206,
 \$ 1420.1021 4609 865, 711.8306 4967 351, 269.6328 2184 603,
 \$ 79.9912 0647 290, 19.0217 1582 6880, 3.7188 1052 3339,
 \$ 0.5076 4877 8323, 0.0842 2020 4896, 0.0100 2621 4869,
 \$ 0.0010 3630 1278, 0.0000 9386 7869, 0.0000 0751 2435,
 \$ 0.0000 0053 5074, 0.0000 0003 4135, 0.0000 0000 1962,
 \$ 0.0000 0000 0102, 0.0000 0000 0005)

DATA(CAP=0.1041 6666 6666 6666 7.0.0835 5034 7222 2222 2,
 \$ 0.1282 2657 4556 3271 6.0.2918 4902 6464 1404 6,
 \$ 0.5816 2726 7443 7576 5.3.3214 0828 1862 768,
 \$ 14.9957 6298 6862 6.78.9230 1301 1587.474.4515 3886 8,
 \$ 3207.4900 91.2 4086 5496.19 8923.12.179 1902.0,
 \$ 1748 4377.0)

DATA(PI=3.1415 92653 58979)

```

DATA(I=(0,0,1,0))
DATA(ALPHA=0.853 667 218 838 951)

CALL MAG ANGLE(Z,ZMAG,PHI)

Z POWER = 1.0
Z CUBED = Z**3

IF(ZMAG .GT. 4.2) GO TO 50

IF(ZMAG .LT. 3.2) 21,22
21 N=12
GO TO 30

22 IF(ZMAG .LT. 4.1) 23,24
23 N=15
GO TO 30

24 N=23

30 SJM1 = 0.0
SJM2 = 0.0
SUM3 = 0.0
SUM4 = 0.0

DO 31 M=1,N
TERM1 = A(M)*Z POWER
TERM2 = B(M)*Z POWER
TERM3 = C(M)*Z POWER
TERM4 = D(M)*Z POWER
SJM1 = SUM1+TERM1
SJM2 = SUM2+TERM2
SUM3 = SUM3+TERM3
SUM4 = SUM4+TERM4
Z POWER = -Z CUBED/200.0+Z POWER
31 CONTINUE

F = SUM1
G = Z*SUM2
GM2F = G-2.0*F
F PRIME = -Z**02*SUM3
G PRIME = SJM4
GPMFP = G PRIME-2.0*F PRIME
RT THIRD = SQRTF(1.0/3.0)

H1 = G+I*RT THIRD*GM2F
H1 PRIME = G PRIME+I*RT THIRD*GPMFP

H2 = G-I*RT THIRD*GM2F
H2 PRIME = G PRIME-I*RT THIRD*GPMFP

RETURN

50 SUM5 = 1.0

```

```

SUM6 = 1.0
SUM7 = 0.0
SUM8 = 0.0
Z REAL = Z
Z IMAG = -I*Z
I POWER = I
M I POWER = -I
RT Z = CPLX SQRT(Z)
ONE RT Z = 1.0/RT Z
Z FOURTH = CPLX SQRT(RT Z)
BETA = ALPHA/Z FOURTH
SQRT Z CUB = RT Z**3
Z M3HALVS = 1.0/SQRT Z CUB
Z M3HAFS M = Z M3HALVS
EXP1 = CPLX EXPF(I*2.0/3.0*SQRT Z CUB)
EXP2 = EXP1*CPLX EXPF(-I*5.0/12.0*PI)
EXP3 = 1.0/EXP1*CPLX EXPF(I*5.0/12.0*PI)
EXP4 = EXP1*CPLX EXPF(I*11.0/12.0*PI)
EXP5 = 1.0/EXP1*CPLX EXPF(-I*11.0/12.0*PI)

D) 51 M=1,14
TERM5 = I POWER*CAP(M)*Z M3HAFS M
TERM6 = M I POWER*CAP(M)*Z M3HAFS M
SUM5 = SUM5+TERM5
SUM6 = SUM6+TERM6
EM = M
SUM7 = SUM7+(-1.5*EM*1.0/Z)*TERM5
SUM8 = SUM8+(-1.5*EM*1.0/Z)*TERM6
Z M3HAFS M = Z M3HAFS M*Z M3HALVS
I POWER = I POWER*I
M I POWER = M I POWER*(-I)
51 CONTINUE

IF(Z REAL .GE. 0.0 .OR. Z IMAG .GE. 0.0) 61,62
61 H1 = BETA*EXP2*SUM6
H1 PRIME = BETA*EXP2*(SUM6*(-0.25*1.0/Z+I*RT Z)+SUM8)
G) TO 70

62 H1 = BETA*(EXP2*SUM6+EXP5*SUM5)
H1 PRIME = BETA*(EXP2*(SUM6*(-0.25*1.0/Z+I*RT Z)+SUM6)+EXP5*(SUM5
S (-0.25*1.0/Z-I*RT Z)+SUM7))

70 IF(Z REAL .GE. 0.0 .OR. Z IMAG .LT. 0.0) 71,72
71 H2 = BETA*EXP3*SUM5
H2 PRIME = BETA*EXP3*(SUM5*(-0.25*1.0/Z-I*RT Z)+SUM7)
RETURN

72 H2 = BETA*(EXP3*SUM5+EXP4*SUM6)
H2 PRIME = BETA*(EXP3*(SUM5*(-0.25*1.0/Z-I*RT Z)+SUM7)+EXP4*(SUM6
S (-0.25*1.0/Z+I*RT Z)+SUM8))
RETURN
END

```

SUBROUTINE R POLYNOM

```

COMMON/INPUT/THETA,IN OMIT(9#)
COMMON/WG INPJT/WGIN SKIP(8),T LIST(11),WGIN OMIT(70)
COMMON/R/LOG R(4),R SKIP(10),R(4)
COMMON/INIT R/ADJ FLAG
COMPLEX THETA,LOG R,R,LOG R MTRX,T LIST,LAGR INIP,CPLX EXPF
INTEGER DIMENSN,ADJ FLAG,DEBNG
DIMENSION LOG R MTRX(10,4)

```

```

ENTRY GEN R POLY

```

```

PRINT 100
100 FORMAT(1H ,////////,29HR MATRIX INTERPOLATION POINTS)

```

```

M = 1

```

```

11 THETA = T LIST(M)

```

```

CALL INTEG

```

```

DO 12 N=1,4

```

```

12 LOG R MTRX(M,N) = LOG R(N)

```

```

IF(T LIST(M+1) .EQ. 0.0) GO TO 13

```

```

ADJ FLAG = 1

```

```

M = M+1 → GO TO 11

```

```

13 DIMENSN = M

```

```

ADJ FLAG = 0

```

```

PRINT 300

```

```

300 FORMAT(1H0,6X,6HT LIST,14X,5H1R11,17X,3H1R1,

```

```

$ 17X,4H1R11,19X,4H1R1)

```

```

DO 31 N=1,DIMENSN

```

```

31 PRINT 301,T LIST(M),(LOG R MTRX(M,N),N=1,4)

```

```

301 FORMAT(1H ,1X,C(F6.2,F6.2),4(2X,C(F10.4,F10.2)))

```

```

RETURN

```

```

ENTRY USE POLY

```

```

DO 51 N=1,4

```

```

LOG R(N) = LAGR INTP(T LIST,LOG R MTRX(1,N),THETA,DIMENSN)

```

```

51 R(N) = CPLX EXPF(LOG R(N))

```

```

RETURN

```

```

END

```

```
FUNCTION LAGR INTP(X,Y,X0,N)
COMPLEX X,Y,X0,LAGR INTP,SUM,PROD
DIMENSION X(N),Y(N)

SUM = 0.0
DO 12 J=1,N
PROD = 1.0
DO 11 I=1,N
IF(I, EQ. J) GO TO 11
PROD = PROD*(X0-X(I))/(X(J)-X(I))
11 CONTINUE
12 SUM = SUM+PROD*Y(J)

LAGR INTP = SUM

END
```

SUBROUTINE INTEG
C RUDDEN

```

COMMON/INPUT/IN SKIP(5),MAG FLD,IN NOT(2),D,PRECSN,
$      TOP HT,IN OMIT(2),DHMIN,IN NO(86)
COMMON/FLG INPUT/DEBUG,RC PRNT,FG SKIP(6),J FLAG,FG OMIT(16)
COMMON/R/LOG R(8),DLOGRDH(8),R OMIT(10)
COMMON/L/HT LIST(150),OMIT1(150,5)
COMMON/COLL FREQ/M,NU FLAG,C HT LIST(25),CF OMIT(125)
COMMON/INIT FLAG/INIT FLAG
COMMON/EN COLL/HT,EC OMIT(10)
COMMON/STEP/DELH
COMMON/OVRFL0/NR OVFL
REAL LOG R,LOG N LIST,LOGR 0,LOWEST HT,LOGE10,MAG FLD
INTEGER SV FLAG,DEBUG,RC PRNT
DIMENSION LOGR 0(8),DLOGRDH n(8),
$      DEL LOGR 0(8),DEL LOGR 1(8),DEL LOGR 2(8)
DATA(LOGE10=2.302585093)

IF(RC PRNT .EQ. 1 .OR. DEBUG .GE. 1) CALL ZERO CLK
IF(RC PRNT .EQ. 1 .OR. DEBUG .GE. 1) PRINT 100
100 FORMAT(1H0,15X,21H R-MATRIX INTEGRATION )
IF(RC PRNT .EQ. 1 .OR. DEBUG .GE. 1) CALL R COLS

FACTOR = EXPF(-PRECSN*LOGE10)
EMAX = 3.0*FACTOR
EMIN = 0.3*FACTOR
L = 1
M = 1
HT = TOP HT
INIT PREV = 0
SV DELH = DHMIN
CALL INIT T

11 CALL T MTRX
INIT FLAG = 0
CALL INITL 1
IF(INIT PREV .EQ. 1 .AND. INIT FLAG .EQ. 0) PRINT 101,H1
101 FORMAT(1H0,16HTOP HT RESET TO ,F7.2,/)
INIT PREV = INIT FLAG
DELH = DHMIN
IF(INIT FLAG .EQ. 1) DELH = 0.5
IF(INIT FLAG .EQ. 1) GO TO 20
IF(RC PRNT .EQ. 1 .OR. DEBUG .GE. 1) CALL PRINT R
CALL S MTRX
CALL R DERIV

C RUNGE KUTTA
20 SV FLAG = 0
IF(HT .EQ. HT LIST(L+1)) L = L+1
IF(HT-DELH .GE. HT LIST(L+1)) GO TO 21
DELH = HT-HT LIST(L+1)
SV FLAG = 1

```

```

21 IF(NU FLAG .EQ. 0) GO TO 22
   IF(HT .EQ. C HT LIST(M+1)) M = M+1
   IF(HT-DELH .GE. C HT LIST(M+1)) GO TO 22
   DELH = HT-C HT LIST(M+1)
   SV FLAG = 1
22 IF(HT-DELH .LT. D) DELH = HT-D

   IF(INIT FLAG .EQ. 0) GO TO 30
   HT = HT-DELH
   GO TO 11

30 DO 31 I=1,8
   LOGR 0(I) = LOG R(I)
31 DLOGRDH 0(I) = DLOGRDH(I)
C TRY AGAIN
33 IF(NR OVFL .GT. 20) RETURN
   DO 34 I=1,8
   DEL LOGR 0(I) = -DLOGRDH 0(I)*DELH
34 LOG R(I) = LOGR 0(I)+0.5*DEL LOGR 0(I)

   HT = HT-0.5*DELH
   CALL S MIX
   CALL R DERIV

   DO 35 I=1,8
   DEL LOGR 1(I) = -DLOGRDH(I)*DELH
35 LOG R(I) = LOGR 0(I)+0.5*DEL LOGR 1(I)

   CALL R DERIV

   DO 36 I=1,8
   DEL LOGR 2(I) = -DLOGRDH(I)*DELH
36 LOG R(I) = LOGR 0(I)+DEL LOGR 2(I)

   HT = HT-DELH+0.5*DELH
   CALL S MIX
   CALL R DERIV

   SUM SQ = 0.0
   DO 37 I=1,8
   DEL LOGR 4 = ((-DLOGRDH(I)*DELH+DEL LOGR 0(I))/2.0
S      +DEL LOGR 1(I)+DEL LOGR 2(I))/3.0
   LOG R(I) = LOGR 0(I)+DEL LOGR 4
37 SUM SQ = SUM SQ+(DEL LOGR 2(I)-DEL LOGR 4)**2
   RMS ERR = SQRT(SUM SQ/8.0)
   IF(RMS ERR .GT. EMAX .AND. DELH .GT. DHMIN) 38,40

38 HT = HT+DELH
   DELH = DELH/2.0
   IF(DELH .LT. DHMIN) DELH = DHMIN
   GO TO 33

40 CALL R DERIV
   IF(RC PRNT .EQ. 1 .OR. DEBUG .GE. 1) CALL PRINT R
   IF(RMS ERR .LT. EMIN) DELH = 2.0*DELH
   IF(SV FLAG .EQ. 1) DELH = SV DELH

```

1969 289

```
SV DELH = DELH
IF(HT .GT. 0) GO TO 20

IF(RC PRNT .EQ. 1) CALL RAT NIF
IF(RC PRNT .EQ. 1 .OR. DEBUG .GE. 1) CALL PRINTIME
CALL V OUT
RETURN

END
```

SUBROUTINE F MTRX

```

COMMON/INPUT/THETA,FREQ,AZIM,DIP,MAG FLD,ALPHA,H,IN UMIT(92)
COMMON/SP INPUT/NR SPEC,SP SKIP(10),CHARGE(5),M RATIO(5)
COMMON/FLG INPUT/FLG SKIP(5),NEG ELECT,INDX REFR,FLG UMIT(18)
COMMON/M MTX/M11,M21,M31,M12,M22,M32,M13,M23,M33
COMMON/CSK/C,S,WAVE NR
COMMON/EN COLL/HT,EN(5),COLL FREQ(5)
COMMON/T MTX/T11,T21,T31,T41,T12,T22,T32,T42,
$      T13,T23,T33,T43,T14,T24,T34,T44
COMMON/3/HT LIST(10),SAVE T(32,10)
COMPLEX I,THETA,C,S,CSQ,
$      ILY,IMY,INY,U,USQ,D,
$      M11,M21,M31,M12,M22,M32,M13,M23,M33,
$      T11,T21,T31,T41,T12,T22,T32,T42,
$      T13,T23,T33,T43,T14,T24,T34,T44,
$      CPLX COSF,CPLX SINF
REAL MAG FLD,M RATIO,
$      LSQYSQ,MSQYSQ,NSQYSQ,LMYSQ,LNYSQ,MNYSQ
DIMENSION Y(5),YSQ(5),T(32),
$      ILY(5),IMY(5),INY(5),LSQYSQ(5),MSQYSQ(5),NSQYSQ(5),
$      LMYSQ(5),LNYSQ(5),MNYSQ(5)
DATA(PI=3.141592653)
DATA(DTR=0.01745329252)
DATA(COEFF X=3.182357E03),(COEFF Y=1.758796E11)
DATA(I=(0.0,1.0))
DATA(T21=(0.0,0.0)),(T22=(0.0,0.0)),(T24=(0.0,0.0))
DATA(T13=(0.0,0.0)),(T33=(0.0,0.0)),(T43=(0.0,0.0))
DATA(T23=(1.0,0.0))
DATA(VEL LT=2.997928E05)
EQUIVALENCE(T11,T)

DO 11 NN=1,10
IF(HT .EQ. HT LIST(NN)) GO TO 20
11 CONTINUE

M11 = 0.0 $ M12 = 0.0 $ M13 = 0.0
M21 = 0.0 $ M22 = 0.0 $ M23 = 0.0
M31 = 0.0 $ M32 = 0.0 $ M33 = 0.0

CALL FIND EN NU

DO 12 K=1,NR SPEC
X = COEFF X*(1.0E06*EN(K))*CHARGE(K)**2/(OMEGA**2*M RATIO(K))
IF(NEG ELECT .EQ. 1) X = -X
IF(INDX REFR .EQ. 1) X = 1.0-(EN(K)*1.0E-06+1.0)**2
Z = COLL FREQ(K)/OMEGA
IF(NEG ELECT .EQ. 1) Z = -Z
U = 1.0-I*Z
USQ = U*U
D = -X/(U*(JSQ-YSQ(K)))
M11 = M11+(JSQ-LSQYSQ(K))*D
M21 = M21+(INY(K)*U-LMYSQ(K))*D
M31 = M31+(-IMY(K)*U-LNYSQ(K))*D

```

```

M12 = M12+(-INV(K)*U-LMYSQ(K))*D
M22 = M22+(JSQ-MSQYSQ(K))*D
M32 = M32+(ILY(K)*U-MNYSQ(K))*D
M13 = M13+(IMY(K)*U-LNYSQ(K))*D
M23 = M23+(-ILY(K)*U-MNYSQ(K))*D
12 M33 = M33+(JSQ-NSQYSQ(K))*D

```

```

CURV TERM = ALPHA*(H-HT)
M11 = M11-CURV TERM
M22 = M22-CURV TERM
M33 = M33-CURV TERM

```

```

D = 1.0/(1.0+M33)
T11 = -S*M31*D
T31 = M23*M31*D-M21
T41 = 1.0+M11-M13*M31*D
T12 = S*M32*D
T32 = CSQ+M22-M23*M32*D
T42 = M32*M13*D-M12
T14 = (CSQ+M33)*D
T34 = S*M23*D
T44 = -S*M13*D

```

```

DO 13 J=1,32
13 SAVE T(J,N) = T(J)
HT LIST(N) = HT
N = N+1
IF(N,GT,10) N = 1
RETURN

```

```

20 DO 21 J=1,32
21 T(J) = SAVE T(J,NN)
RETURN

```

ENTRY INIT T

```

OMEGA = 2.0*PI*FREQ*1000.0
WAVE NR = OMEGA/VEL LT
SIN DIP = SIN(DIP*DTR)
DIR COS L = SIN DIP*COS(AZIM*DTR)
DIR COS M = SIN DIP*SIN(AZIM*DTR)
DIR COS N = -COS(DIP*DTR)
C = CPLX COS(THETA*DTR)
CSQ = C**2
S = CPLX SIN(THETA*DTR)

```

```

DO 31 K=1,NR SPEC
Y(K) = COEFF Y*CHARGE(K)*MAG FLD/(OMEGA+M RATIO(K))
YSQ(K) = Y(K)**2
ILY(K) = I*DIR COS L*Y(K)
IMY(K) = I*DIR COS M*Y(K)
INV(K) = I*DIR COS N*Y(K)
LSQYSQ(K) = DIR COS L**2*YSQ(K)
MSQYSQ(K) = DIR COS M**2*YSQ(K)
NSQYSQ(K) = DIR COS N**2*YSQ(K)
LMYSQ(K) = DIR COS L*DIR COS M*YSQ(K)

```

```
LNYSQ(K) = DIR COS L*DIR COS N*YSQ(K)  
31 MNYSQ(K) = DIR COS M*DIR COS N*YSQ(K)
```

```
DO 32 J=1,10  
32 HT LIST(J) = 9000.0  
N = 1  
RETURN
```

```
END
```

```

SUBROUTINE EN NU

COMMON/FLG INPUT/FLG SKIP(2),POLY FLAG,FLG OMIT(22)
COMMON/SP INPUT/NR SPEC,COEFF NU(5),EXP NU(5),SP OMIT(10)
COMMON/1/L,HT LIST(150),LOG N LIST(150,5)
COMMON/EN COLL/HT,EN(5),COLL FREQ(5)
COMMON/COLL FREQ/M,NU FLAG,C HT LIST(25),C LOG LIST(25,5)
REAL LOG N LIST,LOG N,LOG COLL
INTEGER POLY FLAG

ENTRY FIND EN NU
DO 23 K=1,NR SPEC
IF(POLY FLAG .EQ. 0) 11,12
11 LOG N = LOG N LIST(L+1,K)+(HT-HT LIST(L+1))
$      / (HT LIST(L)-HT LIST(L+1))
$      * (LOG N LIST(L,K)-LOG N LIST(L+1,K))
EN(K) = EXPF(LOG N)
GO TO 20
12 CALL FIND EN

20 IF(NU FLAG .EQ. 0) 21,22
21 ABS HT = HT
COLL FREQ(K) = COEFF NU(K)*EXPF(EXP NU(K)*1000.0*ABS HT)
GO TO 23
22 LOG COLL = C LOG LIST(M+1,K)+(HT-C HT LIST(M+1))/(C HT LIST(M)
$      -C HT LIST(M+1))*(C LOG LIST(M,K)-C LOG LIST(M+1,K))
COLL FREQ(K) = EXPF(LOG COLL)
23 CONTINUE
RETURN

END

```

SUBROUTINE DIFF EQ

```

COMMON/INPUT/IN SKIP(5),MAG FLD,IN OMIT(94)
COMMON/R/LOG R11,LOG R22,LOG R12,LOG R21,
$     DLNR11DH,DLNR22DH,DLNR12DH,DLNR21DH,
$     R SKIP(2),R11,R22,R12,R21
COMMON/T MTRX/T11,T21,T31,T41,T12,T22,T32,T42,
$     T13,T23,T33,T43,T14,T24,T34,T44
COMMON/CSK/C,CK SKIP(2),K
COMPLEX I,K OVER 2I,C,OV C,
$     T11,T21,T31,T41,T12,T22,T32,T42,
$     T13,T23,T33,T43,T14,T24,T34,T44,
$     S11A,D11A,S11B,D11B,S12,D12,S21,D21,S22,D22,
$     B11,B22,B12,B21,R12,R21,C12,C21,
$     DERIV11,DERIV22,DERIV12,DERIV21,
$     R11,R22,R12,R21,LOG R11,LOG R22,LOG R12,LOG R21,
$     DLNR11DH,DLNR22DH,DLNR12DH,DLNR21DH,
$     CPLX EXPF
REAL MAG FLD,K
DATA(I=(0.0,1.0))

```

```

ENTRY S MTRX
K OVER 2I = K/(2.0*I)
OV C = 1.0/C

```

```
CALL T MTRX
```

```

S11A = T11+T44
D11A = T11-T44
S11B = T14*JV C+C*T41
D11B = T14*JV C-C*T41
S12 = T12*OV C+T42
D12 = T12*OV C-T42
S21 = T34*OV C+T31
D21 = T34*OV C-T31
S22 = C+I32*OV C
D22 = C-I32*OV C
RETURN

```

```

ENTRY R DERIV
R11 = CPLX EXPF(LOG R11)
R22 = CPLX EXPF(LOG R22)
R12 = CPLX EXPF(LOG R12)
R21 = CPLX EXPF(LOG R21)

```

```

RL LOG R12 = LOG R12
IF(RL LOG R12 .LT. -11.0) R12 = 0.0
RL LOG R21 = LOG R21
IF(RL LOG R21 .LT. -11.0) R21 = 0.0

```

```

B11 = R11*(D11A-D11B)
B22 = R22*D22
B12 = R12*D21

```

```

B21 = R21*S12
R12R21 = R12*R21
C12 = R12*S21
C21 = R21*D12

DERIV11 = B11+B12+B21-S11B-S11B+(R12R21*D22+C12+C21-D11A-D11B)/R11
DERIV22 = B12+B21+B22-S22-S22+(R12R21*(D11A-D11B)+B12+B21+D22)/R22
IF(MAG FLD .EQ. 0.0) GO TO 41
DERIV12 = B11+B12+B22+S11A-S11B-S22+(R11*S12+D12)*(R22+1.0)/R12
DERIV21 = B11+B21+B22-S11A-S11B-S22+(R11*D21+S21)*(R22+1.0)/R21
GO TO 42
41 DERIV12 = 0.0
   DERIV21 = 0.0

42 DLNR11DH = < OVER 21*DERIV11
   DLNR22DH = < OVER 21*DERIV22
   DLNR12DH = < OVER 21*DERIV12
   DLNR21DH = < OVER 21*DERIV21
RETURN

END

```

```

SUBROUTINE INITL R

COMMON/INPUT/IN SKIP(5),MAG FLD,IN OMIT(6),Q MAX,IN NO(87)
COMMON/FLG INPUT/DEBUG,FG OMIT(24)
COMMON/T MTX/T11,T21,T31,T41,T12,T22,T32,T42,
$      T13,T23,T33,T43,T14,T24,T34,T44
COMMON/R/LOG R11,LOG R22,LOG R12,LOG R21,R SKIP(10),
$      R11,R22,R12,R21
COMMON/INIT R/ADJ FLG
COMMON/CSK/C,CK OMIT(3)
COMMON/P MTX/P(4,2)
COMMON/INIT FLAG/INIT FLAG
COMPLEX C,P,OVR DET,C TEMP,DFT,
$      B3,B2,B1,B0,Q,
$      T11,T21,T31,T41,T12,T22,T32,T42,
$      T13,T23,T33,T43,T14,T24,T34,T44,
$      G12,G13,G14,G23,G24,G34,
$      D00,D11,D22,D12,D21,R11,R22,R12,R21,
$      LOG R11,LOG R22,LOG R12,LOG R21,
$      FN SQ,F ROOT,
$      CX SQRT,CX LOGF,CONJ
REAL MAG FLU
INTEGER DEBJG,ADJ FLG
DIMENSION DIFF(4),Q(4)
DIMENSION PHASE R(8),PREV PHAS(8)
DATA(PI=3.141592653)
DATA(ADJ FLG=0)
EQUIVALENCE(LOG R11,PHASE R)

IF(MAG FLD .EQ. 0.0) GO TO 5n

B3 = -(T11+T44)
B2 = T11*T44-T14*T41-T23*T32
B1 = -T23*(-T11*T32-T32*T44+T12*T31+T34*T42)
B0 = T23*(-T11*(T32*T44-T34*T42)
$      +T12*(T31*T44-T34*T41)
$      -T14*(T31*T42-T32*T41))

CALL QUANTC(B3,B2,B1,B0,Q,DEBUG)

Q MAXIM = 0.0
DO 12 N=1,4
CALL MAG ANG(Q(N),Q MAG,ANGLE Q)
IF(Q MAG .GT. Q MAXIM) Q MAXIM = Q MAG
IF(ANGLE Q .LT. 135.0) ANGLE Q = ANGLE Q+360.0
12 DIFF(N) = ABSF(ANGLE Q-315.0)
IF(Q MAXIM .GT. Q MAX) INIT FLAG = 1
IF(INIT FLAG .EQ. 1) RETURN

DO 15 M=2,4
DO 15 N=M,4
IF(DIFF(N) .GT. DIFF(M-1)) GO TO 15
TEMP = DIFF(N)
DIFF(N) = DIFF(M-1)

```

```

DIFF(M-1) = TEMP
C TEMP = Q(N)
Q(N) = Q(M-1)
Q(M-1) = C TEMP
15 CONTINUE

DO 21 N=1,2
DET = (T11-Q(N))*(T44-Q(N))-T14*T41
OVR DET = 1.0/DET

P(1,N) = (T12*T23*Q(N)-T23*(T12*T44-T14*T42))*OVR DET
P(2,N) = T23
P(3,N) = Q(N)
P(4,N) = (T23*T42*Q(N)+T23*(T12*T41-T11*T42))*OVR DET

POYNTG = P(1,N)*CONJ(P(4,N))-CONJ(P(3,N))*(-P(2,N))
IF(POYNTG .LT. 0.0) PRINT 201,Q(N),POYNTG
201 FORMAT(1H0,8HFOR Q = ,C(E11.3,E11.3),11HPOYNTG = ,E11.3)
21 CONTINUE

G12 = P(1,1)*P(2,2)-P(1,2)*P(2,1)
G13 = P(1,1)*P(3,2)-P(1,2)*P(3,1)
G14 = P(1,1)*P(4,2)-P(1,2)*P(4,1)
G23 = P(2,1)*P(3,2)-P(2,2)*P(3,1)
G24 = P(2,1)*P(4,2)-P(2,2)*P(4,1)
G34 = P(3,1)*P(4,2)-P(3,2)*P(4,1)

D00 = -G13+C*(G34-G12+C*G24)
D11 = G13+C*(G34+G12+C*G24)
D22 = G13+C*(-G34-G12+C*G24)
D12 = 2.0*C*G14
D21 = 2.0*C*G23

R11 = D11/D00
R22 = D22/D00
R12 = D12/D00
R21 = D21/D00

22 LOG R11 = CX LOGF(R11)
LOG R22 = CX LOGF(R22)
IF(MAG FLD .EQ. 0.0) GO TO 24
LOG R12 = CX LOGF(R12)
LOG R21 = CX LOGF(R21)
GO TO 30
23 LOG R12 = (-99.0,0.0)
LOG R21 = (-99.0,0.0)

30 IF(ADJ FLG .EQ. 0) GO TO 36
DO 35 N=2,8,2
33 IF(PHASE R(N)-PREV PHAS(N) .IE. PI) GO TO 34
PHASE R(N) = PHASE R(N)-2.0*PI
GO TO 33
34 IF(PREV PHAS(N)-PHASE R(N) .IE. PI) GO TO 35
PHASE R(N) = PHASE R(N)+2.0*PI
GO TO 34
35 CONTINUE

```

```
36 DO 37 N=2,8,2
37 PREV PHAS(N) = PHASE R(N)
   RETURN
```

```
C RS BY FRESNEL
```

```
50 FN SQ = I41
   F ROOT = CX SQRT(T14+I41)
   R11 = (FN SJ+C-F ROOT)/(FN SQ+C+F ROOT)
   R22 = (C-F ROOT)/(C+F ROOT)
   R12 = 0.0
   R21 = 0.0
   GO TO 22
```

```
END
```

```
SUBROUTINE QUARTC(FOUR B3,SIX B2,FOUR B1,B0,0,DEBUG)
```

```
COMPLEX B3,B2,B1,B0,0,
$      FOUR B3,SIX B2,FOUR B1,
$      B3 SQ,H,I,G,H PRIME,G PRIME,
$      SJ ROOT,P PLUS,P,LOG P,
$      CUBE RT0,CUBE RT1,CUBE RT2,OMEGA1,OMEGA2,
$      ROOT P,ROOT Q,ROOT R,
$      FNJCTION,CPLX I,
$      CPLX SQRT,CPLX LOGF,CPLX EXPF
INTEGER DEBUG,ERR FLAG
REAL MAG PLUS,MAG MINUS,MAG F
DIMENSION Q(4),P RI(2),FUNCTION(4)
DATA(OMEGA1=(-0.5,0.8660254038)),(OMEGA2=(-0.5,-0.8660254038))
DATA(CPLX I=(0.0+1.0))
DATA(TOL=1.0E-02)
EQUIVALENCE(P,P RI)
```

```
B3 = FOUR B3*0.25
B2 = SIX B2/6.0
B1 = FOUR B1*0.25
```

```
B3 SQ = B3**2
H = B2-B3 SJ
I = B0-4.0*B3*B1+3.0*B2**2
G = B1*B3*(-3.0*B2+2.0*B3 SQ)
H PRIME = -I/12.0
G PRIME = -3**2/4.0-H*(H**2+1.0*H PRIME)
```

```
SJ ROOT = CPLX SQRT(G PRIME**2+4.0*H PRIME**3)
P = (-G PRIME+SQ ROOT)*0.5
MAG PLUS = ABSF(P RI(1))+ABSF(P RI(2))
P PLUS = P
P = (-G PRIME-SQ ROOT)*0.5
MAG MINUS = ABSF(P RI(1))+ABSF(P RI(2))
IF(MAG PLUS .GT. MAG MINUS) P = P PLUS
LOG P = CPLX LOGF(P)
CUBE RT0 = CPLX EXPF(LOG P/3.0)
CUBE RT1 = OMEGA1*CUBE RT0
CUBE RT2 = OMEGA2*CUBE RT0
```

```
ROOT P = CPLX SORT(CUBE RT0-H PRIME/CUBE RT0-H)
ROOT Q = CPLX SORT(CUBE RT1-H PRIME/CUBE RT1-H)
ROOT R = CPLX SORT(CUBE RT2-H PRIME/CUBE RT2-H)
SIGN = -ROOT P*ROOT Q*ROOT R+2.0/G
IF(SIGN .LT. 0.0) ROOT R = -ROOT R
Q(1) = +ROOT P+ROOT Q+ROOT R-B3
Q(2) = +ROOT P-ROOT Q-ROOT R-B3
Q(3) = -ROOT P+ROOT Q-ROOT R-B3
Q(4) = -ROOT P-ROOT Q+ROOT R-B3
```

```
DO 32 N=1,4
Q4 = Q(N) $ AJS QR = ABSF(QR)
QI = -CPLX I*Q(N) $ ABS QI = ABSF(QI)
```

```

IF (ABS QR+TOL .LT. ABS QI) GO TO 31
QI = CPLX I*(((QR+4.0*B3)*QR+6.0*B2)*QR+4.0*B1)*QR+B0)
$      /(((4.0*QR+12.0*B3)*QR+12.0*B2)*QR+4.0*B1)
31 IF (ABS QI+TOL .LT. ABS QR) GO TO 32
QR = (((QI-CPLX I*4.0*B3)*QI-6.0*B2)*QI+CPLX I*4.0*B1)*QI+B0)
$      /(((CPLX I*4.0*QI+12.0*B3)*QI-CPLX I*12.0*B2)*QI-4.0*B1)
32 Q(N) = QR+CPLX I*QI

IF (DEBUG .EQ. 0) RETURN

ERR FLAG = 0
DO 51 N=1,4
IF (DEBUG .GE. 2) PRINT 501,N,Q(N)
501 FORMAT(1H+,5HROOT ,1,3H = ,C(E12.5,E13.5))
FUNCTION(N) = (((Q(N)+4.0*B3)*Q(N)+6.0*B2)*Q(N)+4.0*B1)*Q(N)+B0)
$      /30
IF (DEBUG .GE. 2) PRINT 502,N,FUNCTION(N)
502 FORMAT(1H ,40X,9HFUNCTION ,1,C(E11.3,F11.3))
CALL MAG ANGLE(FUNCTION(N),MAG F,ANGLE F)
IF (MAG F .LT. 0.10) GO TO 51
PRINT 503,N,MAG F
503 FORMAT(1H ,5HROOT ,1,23H IS NO GOOD MAG F = ,E10.3)
ERR FLAG = 1
51 CONTINUE
IF (ERR FLAG .EQ. 1) STOP
RETURN

END

```

```

SUBROUTINE INITL A
COMMON/INPUT/IN SKIP(10),TOP HT,IN OMIT(89)
COMMON/R/R SKIP(18),R11,R22,R12,R21
COMMON/EN COLL/HT,EC OMIT(10)
COMMON/CSK/C,CK OMIT(3)
COMMON/L/L,OMIT1(150,6)
COMPLEX R11,R22,R12,R21,DELTA,A11,A22,A12,A21,C

```

```

L = 1
HT = TOP HT
CALL INIT T
CALL S MIRX
CALL INITL R

```

```

DELTA = (R11-1.0)*(R22+1.0)-R12*R21
A11 = -(2.0/DELTA*(R22+1.0)+1.0)/C
A22 = -(2.0/DELTA*(R11-1.0)-1.0)*C
A12 = -2.0*R12/DELTA
A21 = -2.0*R21/DELTA

```

```

PRINT 100
100 FORMAT(1H1,27HCROMBIE SOLUTION - A MATRIX)
PRINT 101,A11
101 FORMAT(1H0,6HA11 = ,C(E15.8,F15.8))
PRINT 102,A22
102 FORMAT(1H0,6HA22 = ,C(E15.8,F15.8))
PRINT 103,A12
103 FORMAT(1H0,6HA12 = ,C(E15.8,F15.8))
PRINT 104,A21
104 FORMAT(1H0,6HA21 = ,C(E15.8,F15.8))

```

```

RETURN
END

```

SUBROUTINE DUMMY
C FOR BUDDEN WAVEGUIDE WITH INITIAL A

ENTRY ZEROCLK
ENTRY PRINTIME
ENTRY IH INTG
ENTRY WAV FLD
ENTRY FIT PRF
ENTRY FIND EN
RETURN

END

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D		
<i>(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)</i>		
1. ORIGINATING ACTIVITY (Corporate author) Naval Electronics Laboratory Center San Diego, California 92152		2a. REPORT SECURITY CLASSIFICATION Unclassified
2b. GROUP		
3. REPORT TITLE "A Fortran program for waveguide propagation which allows for both vertical and horizontal dipole excitation"		
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Research Report		
5. AUTHOR(S) (First name, middle initial, last name) Richard A. Pappert, Linda R. Shockey, William F. Moler		
6. REPORT DATE 15 June 1970	7a. TOTAL NO. OF PAGES 72	7b. NO. OF REFS 11
8a. CONTRACT OR GRANT NO. DASA MIPR 542-70	9a. ORIGINATOR'S REPORT NUMBER(S) Interim Report No. 702	
b. PROJECT NO. DASA Subtask HB 042	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
d.		
10. DISTRIBUTION STATEMENT Distribution of this document is unlimited.		
11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Defense Atomic Support Agency	
13. ABSTRACT This report presents an updated version of an earlier waveguide program written in the FORTRAN compiler language. The new program includes the following additions: (1) Provision for calculating excitation factors for all electric field components Ex, Ey, and Ez as a function of height within the guide. (2) The excitation factors can be generated for both vertical and horizontal electric dipole exciters.		

DD FORM 1473 (PAGE 1)
1 NOV 65
S/N 0101-807-6801

UNCLASSIFIED

Security Classification JOB 73

REF ADR-12	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Waveguide modes						
VLF propagation						
Fortran program						
Mode excitation						
Horizontal dipole						
Vertical dipole						