U.S. DEPARTMENT OF THE INTERIOR
GEOLOGICAL SURVEY

Fast evaluation of Hr and Hz field soundings near a rectangular loop source on a layered earth (Program HRZRECT)

by

Walter L. Anderson

Open-File Report 84-257

1984

DISCLAIMER

This program was written in FORTRAN-77 for a VAX-11/780 system*. Although program tests have been made, no guarantee (expressed or implied) is made by the author or the Geological Survey regarding program correctness, accuracy, or proper execution on all computer systems.

* Any use of trade names in this report is for descriptive purposes only and does not imply endorsement by the U.S. Geological Survey. This report is preliminary and has not been reviewed for conformity with U.S. Geological Survey editorial standards.

CONTENTS

ABSTRACT... 2
INTRODUCTION....................................... 3
SUMMARY OF CALCULATIONS............................ 4
PARAMETERS REQUIRED................................ 7
PROGRAM FILES...................................... 8
DETAILED PARAMETER DEFINITIONS..................... 8
EXAMPLES OF INPUT PARAMETERS....................... 10
VAX OPERATING INSTRUCTIONS......................... 10
ERROR MESSAGES..................................... 11
PRINTED OUTPUT..................................... 11
REFERENCES... 13

Appendix 1.-- Conversion to other systems.............. 14
Appendix 2.-- Test problem input/output................ 15
Appendix 3.-- Some sounding curve example plots......... 17
Appendix 4.-- Source code availability................. 32
Source code listing.................... 33

Figure 1.-- Loop geometry at z=0 (earth's surface)...... 4

ABSTRACT

A new technique is presented to rapidly compute parametric and geometric soundings for the radial and vertical magnetic fields inside or outside a rectangular loop source on a layered earth. A fast Hankel transform (FHT) algorithm (Anderson, 1982) was applied to rapidly obtain in one pass both field components. The FHT uses concepts of related and lagged convolutions (digital filtering), and when applied to the rectangular loop problem, reduces each field calculation to four elementary spline definite integrations. Several numerical comparisons with existing dipole and other rectangular loop solutions show at least 3-figure accuracy is achieved with the new method. Some numerical results, along with VAX execution times, are listed and plotted. A complete FORTRAN source listing is provided.

INTRODUCTION

Program HRZRECT rapidly computes frequency (parametric) and distance (geometric) sounding curves for radial and vertical magnetic field components (denoted respectively by Hr and Hz) over layered earth models. The fields can be computed at points inside or outside a rectangular loop source of arbitrary dimensions. The rectangular loop is assumed to be placed on the earth's surface and the layers are parallel to the surface. Displacement currents are neglected (quasi-static case) for all computations.

Well known methods exist for calculating the electromagnetic (EM) fields at any distance from a vertical magnetic dipole or horizontal loop source (e.g., Frischknecht, 1967; Wait, 1958; Wait, 1966). Linear digital filtering algorithms (e.g., Anderson, 1979) provide for rapid and accurate calculations for dipole sources. Kauahikaua (1978) presented a method for computing the magnetic field about a straight horizontal finite-length grounded wire source over a layered earth. Recently, Poddar (1983) developed the solution for Hz about a rectangular loop source of current on a multilayered earth. Poddar's solution used four separate double numerical integrations, and by superposition, obtained the total Hz field inside or outside the rectangular loop at arbitrary positions.

This report describes a new method to rapidly compute the Hr and Hz fields about a rectangular loop source in one pass, and does not require separate double integrations over each side of the rectangle. Recent advances in evaluating Hankel transforms (Anderson, 1982) lead naturally to this new solution, which extends Poddar's (1983) Hz solution to include Hr (or Hx and Hy) field components. Program HRZRECT is intended to provide a practical tool for studying the frequency response for cases where a dipole source cannot be assumed. In most field situations, it is easier to lay out a square or rectangular wire loop than a circular loop; consequently, this program should be more appropriate (and efficient) than an exclusively circular loop computation (e.g., Ryu and others, 1970).

Some tests were run using HRZRECT with small loop sizes and large spacings to simulate the dipole-dipole case. Both Hz and Hr field results agreed to about 3-place accuracy with existing dipole source results (Frischknecht, 1967). Tests using the same rectangular source and models as given by Poddar (1983, Fig. 2-3) were also run using HRZRECT, which included Hr as well as Hz fields; the results are listed and plotted (along with other models) in Appendix 3.

The remainder of this report contains 1) a summary of the basic computations, 2) a detailed description of the program parameters, and 3) the VAX operating instructions. Appendix 1 offers some suggestions in converting the VAX program to other computer systems; Appendix 2 lists a simple input/output test example; Appendix 3 provides several families of sounding curves computed by varying certain model parameters; and Appendix 4 lists the FORTRAN-77 source code.

SUMMARY OF CALCULATIONS

Figure 1 shows the coordinate system and geometry of the rectangular loop.

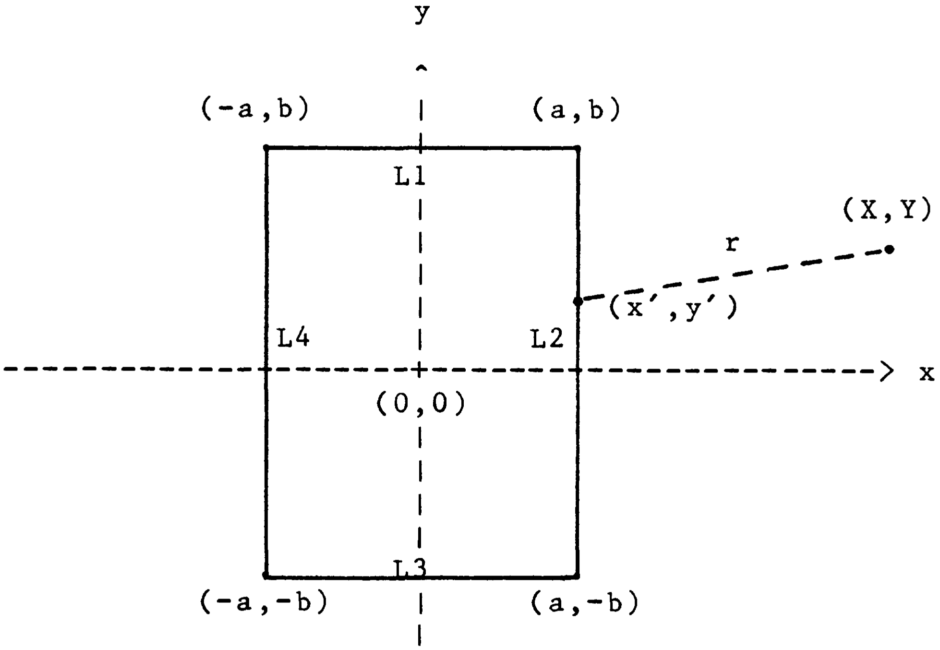


Fig.1.-- Loop geometry at z=0 (earth's surface).

The line segments [(-a,b):(a,b)], [(a,-b):(a,b)], [(-a,-b):(a,-b)], and [(-a,-b):(-a,b)] are denoted respectively as lines L1, L2, L3, and L4. The length of lines L1 and L3 is 2a, and lines L2 and L4 is 2b. The observation point is (X,Y), and (x',y') is any point on the rectangular loop source.

The magnetic field inside or outside a rectangular loop can be formally obtained by a suitable summation of the results from four separate finite grounded wires as defined in Kauahikaua (1978); however, the rectangular loop problem is simpler, because there are no currents injected into the earth at the ends of each wire segment. The formulas in Kauahikaua (1978) are written in a form such that the contribution from currents at the wire ends may be readily neglected. (The latter fact is used below in the Hr loop development.) Poddar (1983) derived his solution for a rectangular loop by starting with the electric field due to a magnetic dipole and then applying reciprocity. From Poddar (1983), the vertical magnetic field Hz at any point (X,Y) for a loop source with current I exp(iwt) is

$$Hz = I\ (H_{L1} + H_{L2} + H_{L3} + H_{L4})/2\pi \quad , \qquad (1)$$

where

$$\left.\begin{aligned}
H_{L1} &= -(b-Y) \int_{-a}^{a} (dx'/r) \int_{0}^{\infty} k(\lambda)\ J_1(\lambda r)\ d\lambda\ , \quad r^2 = R_1^2 \\
H_{L2} &= -(a-X) \int_{-b}^{b} (dy'/r) \int_{0}^{\infty} k(\lambda)\ J_1(\lambda r)\ d\lambda\ , \quad r^2 = R_2^2 \\
H_{L3} &= -(b+Y) \int_{-a}^{a} (dx'/r) \int_{0}^{\infty} k(\lambda)\ J_1(\lambda r)\ d\lambda\ , \quad r^2 = R_3^2 \\
H_{L4} &= -(a+X) \int_{-b}^{b} (dy'/r) \int_{0}^{\infty} k(\lambda)\ J_1(\lambda r)\ d\lambda\ , \quad r^2 = R_4^2
\end{aligned}\right\} \quad (2)$$

$$\left.\begin{aligned}
R_1^2 &= (x'-X)^2 + (b-Y)^2\ , \\
R_2^2 &= (a-X)^2 + (y'-Y)^2\ , \\
R_3^2 &= (x'-X)^2 + (b+Y)^2\ , \\
R_4^2 &= (a+X)^2 + (y'-Y)^2\ ,
\end{aligned}\right\} \quad (3)$$

and $k(\lambda)$ is a recursive complex kernel function (Poddar, 1983) containing the factor $\exp(-\lambda z)$, $z>0$. The Hankel transforms in (2) are not convergent if the observation point is on the surface ($z=0$). To overcome this problem, Poddar set $z=10^{-3}$ meters in $k(\lambda;z)$. This approach was not used in HRZRECT, because some advantage can be gained using $z=0$ and the fast converging formulas derived by Kauahikaua (1978), where the half-space response was removed from $k(\lambda)$ and a closed-form expression added outside the integrals. This modification to (2) becomes

$$\left.\begin{aligned}
H_{L1} &= -(b-Y) \int_{-a}^{a} (dx'/r)\ \{h_z^{(S)}(r)\}\ , \quad r^2 = R_1^2 \text{ in } (3) \\
H_{L2} &= -(a-X) \int_{-b}^{b} (dy'/r)\ \{h_z^{(S)}(r)\}\ , \quad r^2 = R_2^2 \text{ in } (3) \\
H_{L3} &= -(b+Y) \int_{-a}^{a} (dx'/r)\ \{h_z^{(S)}(r)\}\ , \quad r^2 = R_3^2 \text{ in } (3) \\
H_{L4} &= -(a+X) \int_{-b}^{b} (dy'/r)\ \{h_z^{(S)}(r)\}\ , \quad r^2 = R_4^2 \text{ in } (3),
\end{aligned}\right\} \quad (4)$$

where

$$h_z^{(S)}(r) = \frac{1}{\delta} \int_{0}^{\infty} f_4(\lambda)\ J_1(\lambda r)\ d\lambda - i\ \delta\ [h_z^{o}(B)]/(2r^4), \quad (5)$$

$$h_z^{o}(B) = 3-\{3+3B(1+i)+2iB^2\}\ \exp[-(1+i)B], \quad i = (-1)^{1/2},$$

$$B = r/\delta, \quad \delta = [2/(\sigma_1 u_0 w)]^{1/2}, \quad w = 2\pi f\ , \ f>0 \text{ Hertz},$$

$$\sigma_1 = \text{conductivity of layer 1 (mhos/meter)}, \ u_0 = 4\pi 10^{-7},$$

and

$f_4(g)$ is defined in Kauahikaua (1978), $g = \lambda\delta$.

(The recursive expressions used in $f_4(g)$ and all associated formulas and notations are explicitly listed in Kauahikaua, 1978, p. 1019-1021, and will not be repeated here; note that $f_4(g)$ contains all the parameters defining the layered earth model.)

Equation (5) is a continuous complex function defined for all r in $[r_{min}, r_{max}]$, where r_{min} and r_{max} are the respective minimum and maximum distances from (X,Y) to all points on the rectangular loop. The Hankel transform and other expressions in (5) are in general required over different subintervals of r for each definite integral in (4). If (5) is sufficiently discretized over all r in $[r_{min}, r_{max}]$, then a single predetermined spline interpolating function (denoted by superscript S) can be used instead of (5) for each definite integral. Thus the four double integrations in (2) are essentially replaced by four single fast spline integrations in (4). The Hankel transform evaluations in (5), coupled with a lagged convolution (or discretation) over r in $[r_{min}, r_{max}]$, is greatly facilitated by a new fast Hankel transform (FHT) algorithm developed by Anderson (1982). The FHT is called by subprogram HANKEL in Appendix 4, and is the main reason for the fast execution times illustrated by the examples given in Appendices 2 and 3.

The Hr radial field component is computed by analogy with Hz above, using the formula for H_y^{fin} in Kauahikaua (1978), but neglecting the term due to the wire ends. The Hr field at any point (X,Y) becomes,

$$Hr = Hx\ (X/R_0) + Hy\ (Y/R_0), \tag{6}$$

where

$$Hx = I\ (-h_{L2} + h_{L4})/2\pi,\quad Hy = I\ (-h_{L1} + h_{L3})/2\pi,\quad R_0^2 = X^2 + Y^2,$$

$$\left.\begin{aligned} h_{L1} &= -\int_{-a}^{a} (dx'/r)\ \{h_r^{(S)}(r)\}\ ,\ r^2 = R_1^2 \text{ in } (3) \\ h_{L2} &= -\int_{-b}^{b} (dy'/r)\ \{h_r^{(S)}(r)\}\ ,\ r^2 = R_2^2 \text{ in } (3) \\ h_{L3} &= -\int_{-a}^{a} (dx'/r)\ \{h_r^{(S)}(r)\}\ ,\ r^2 = R_3^2 \text{ in } (3) \\ h_{L4} &= -\int_{-b}^{b} (dy'/r)\ \{h_r^{(S)}(r)\}\ ,\ r^2 = R_4^2 \text{ in } (3), \end{aligned}\right\} \tag{7}$$

$$h_r^{(S)}(r) = -\{B \int_0^\infty f_4(\lambda)\ J_0(\lambda r)\ d\lambda + \tag{8}$$

$$+ \frac{1}{r}\ [\beta(I_0(\beta)K_1(\beta) - I_1(\beta)K_0(\beta)) - 2\ I_1(\beta)K_1(\beta)]\}$$

$$\beta = B(1+i)/2,\quad i = (-1)^{1/2},\quad B = r/\delta\ ,$$

and I_0, I_1, K_0, K_1 are modified Bessel functions of orders 0,1.

Equation (8) is replaced by a precomputed spline function in a similar way as done for $h_z^{(S)}(r)$ in (5). Bessel functions are needed initially in (8) to compute the spline coefficients, but they are not required while performing the four spline integrations in (7). Computation of all Hz and Hr Hankel transforms needed in (5) and (8) are rapidly obtained in one call to the FHT algorithm using related and lagged convolutions (Anderson, 1982). Thus both field components are obtained in nearly the same time as that required to evaluate a single component. Optionally, the Hx and Hy orthogonal components are computed directly from Hr.

The Hankel transforms in (5) and (8) are zero for a half-space model; this is one benefit of using the z=0 formulas from Kauahikaua (1978), instead of the z>0 case by Poddar (1983). The general expressions in (5) and (8) apply to either frequency (parametric) or distance (geometric) soundings, providing a unified mathematical treatment.

Examples of various models using HRZRECT are provided in Appendix 3. The results tabulated and plotted in Appendix 3 duplicate the Hz frequency and distance soundings illustrated in Poddar (1983). The same models were computed for Hr frequency and distance soundings, along with new Hz and Hr field soundings near the loop (inside and outside). For typical field cases and moderate accuracy parameters, the observation point should not be placed extremely close to the loop source; e.g., points where r<min(a,b)/10 should be avoided.

The output listings in Appendices 2 and 3 reflect both unnormalized fields and normalized mutual coupling ratios (Z/Z0). Z0 is defined as the free space field from a rectangular loop source of current and is given by Poddar (1982, p. 104). The section "PRINTED OUTPUT" describes all symbols used in the output listings.

PARAMETERS REQUIRED

Parameters required by program HRZRECT are read using FORTRAN NAMELIST input on the VAX/VMS system (version 3.5). The namelist name used is $PARMS. Default values are assumed whenever any parameter is omitted, except as noted otherwise. Preceding the $PARMS statement is an 80-character title.

The general input order read by program HRZRECT is as follows:

1. Title record (always required, maximum of 80-characters).
2. $PARMS --nondefault parameters--$END. All $PARMS allowed are described in the section DETAILED PARAMETER DEFINITIONS below. Note that $PARMS may begin in column 1, and records may be continued to succeeding records until the final $ or $END is encountered, where the "END" in $END is optional.
3. Optionally, subsequent problem sets using changed $PARMS may be given by repeating steps 1-2.

The above general input order is required whether the job is being run in time-sharing or batch modes (see VAX OPERATING INSTRUCTIONS below).

PROGRAM FILES

FOR001 to FOR004, FOR007, and FOR098-- Temporary work files used during execution; all work files are deleted on program end or error return to VMS.

FOR005-- Title and $PARMS input parameters.

FOR006-- Output on-line terminal file.

FOR012-- Output amplitude curves disk plot file (only written if IPLT>0).

FOR013-- Output phase curves disk plot file (only written if IPLT>0).

FOR016-- Output disk print-file (duplicate of on-line file FOR006).

DETAILED PARAMETER DEFINITIONS

$PARMS parameters (nondefault parameters must always be given):

M= Number of layers in the model (1≤M≤10; default M=1 for a homogeneous half-space).

SIG()= Array of M-layer conductivities (in mhos/m.), where SIG(1)>0 and SIG(I)≥0, for I=2,3,...,M.

D()= Array of M-1 layer thicknesses (in m.), where D(I)>0, for I=1,2,...,M-1. Array D() is ignored if M=1.

AX,BY= Coordinates (a,b) of corner of rectangular loop source in the first quadrant in Fig. 1. (Both AX and BY must be given >0.)

X,Y= Coordinates of the initial (or only) observation point. Note that X,Y must not lie on the rectangular source, and in general, X,Y should not be extremely close to any line segment L1, L2, L3, or L4 in Fig. 1; e.g., a guideline is to choose (X,Y) such that r>MIN(AX,BY)/10. Symmetry (e.g., X≥0, Y=0) should be used whenever possible.

DX,DY= Increments in X and (or) Y-directions, which when non-zero, selects "distance soundings" between (X,Y) to (XM,YM) in additive increments (DX,DY). When DX>0 or DY>0, then Xi is varied from X as X+i*DX to XM within each Y (with Yj varied outside each X-row from Y as Y+j*DY to YM). Generally, symmetry should be considered, and only unique points specified; e.g., use DX>0, DY=0 and vary X in just the first quadrant for fields that are symmetrical about the line X=0. Default is DX=0, DY=0, which computes fields at the single point (X,Y).

XM,YM= Maximum end-point coordinates in X and (or) Y-directions when selecting "distance soundings" (i.e., only applies when DX>0 or DY>0). Note that XM,YM must be specified in such a way that some (X+i*DX, Y+j*DY) would not cross the rectangular loop boundary in Fig. 1. Again, care should be taken to also avoid "near source" observation points. (Default XM=0, YM=0.)

EPS= Requested integration tolerance used to compute all Hankel transforms by related and lagged convolutions as described for subprogram HANKEL in Anderson (1982, p.352-353). Default EPS=0.1E-9, which is about the optimum request for a 32-bit word machine to give relative errors $\geq$0.1E-7.

EPS2= Requested relative error for all finite spline-interpolated integrations in (4) and (7) using adaptive Guassian quadrature subprogram ZSUBA1 (see code in Appendix 4). Default EPS2=0.1E-3, which gives about 3 or more figure accuracy in all Hr and Hz fields; the actual spline integration relative errors are listed for real and imaginary parts as ERR(Re) and ERR(Im)--see examples in Appendices 2 and 3, and also the comments on ERR(Re,Im) in the section PRINTED OUTPUT.

MXEVAL= Maximum function evaluations by spline-interpolation to allow in any adaptive Gaussian quadrature using subprogram ZSUBA1. (Default MXEVAL=500; MXEVAL should be increased if EPS2 is decreased from its default value, or if attempting near source problems.)

RFAC= An heuristic factor ($\geq$1.0) to use to expand the effective [RMIN,RMAX] range in order to avoid pertubations on the ends of each spline-defined function in (5) or (8). (Default RFAC=5, which should be more than adequate for typical field applications.)

F0,NF,FM= Initial, number/log-cycle, and final frequencies (in Hertz) to use to select a "frequency sounding". (Default NF=0 ignores this option.) NF>0 selects this option; and NF<0 flags a special option to indicate $|NF|\leq50$ frequencies are given in array FNF() for the frequency sounding.)

FNF()= Array of increasing frequencies (in Hertz) used only when NF<0. When NF<0, then a maximum of 50 increasing values >0 can be given in FNF(I) for I=1,2,...,|NF|.

B0,NB,BM= Initial, number/log-cycle, and final induction numbers B (see (5)) to use to select an "induction number sounding"--also called frequency sounding, because when B varies, F also varies, for a constant r-distance. (Default NB=0 ignores this option.) NB>0 selects this option; and NB<0 flags a special option to indicate $|NB|\leq50$ induction numbers are given in array BNB() for the induction sounding.)

BNB()= Array of increasing induction numbers used only when NB<0. When NB<0, then a maximum of 50 increasing values >0 can be given in BNB(I) for I=1,2,...,|NB|.

[For either induction number B-soundings, or frequency F-soundings, only a single observation point (X,Y) can be specified. Furthermore, for either case (B- or F-soundings), the opposite values are printed for additional information in the output files.]

INFO=0 (default) to ignore this option; i.e., INFO=0 implies only CPU integration times are printed in files FOR006 and FOR016.

INFO=1 to print additional information concerning the expanded [RMIN,RMAX] range for the given RFAC, and to give some other "debugging information" regarding the finite quadratures for lines L1, L2, L3, and L4 in Fig.1. (INFO=1 is not recommended nor needed for routine processing.)

IPLT=0 (default) to ignore this output option; i.e., IPLT=0 will suppress writing plot files FOR012 and FOR013.

IPLT>0 (see specific values below) will write disk output files FOR012 and FOR013 in a specific format used by a special USGS plot system, which is designed only for local or USGS plotting devices, and therefore, is not available for distribution. (For USGS users, see the author on the use of FOR012 and FOR013 plot files.)

IPLT=1 to output AMP(Z/Z0) on file FQR012 and PHASE(Z/Z0) on file FOR013.

IPLT=2 to output AMP(Z) on file FOR012 and PHASE(Z) = PHASE(Z/Z0) on file FOR013, where Z is the unnormalized selected field component.

ICOMP= Field component selection option:
ICOMP=0 for Hz only;
ICOMP=1 for Hr only;
ICOMP=2 (default) for Hz, Hr.
ICOMP=3 for Hz, Hr, Hx, and Hy.

$END [end of $PARMS parameters; the "END" in $END may be omitted, if desired.]

EXAMPLES OF INPUT PARAMETERS

```
Example Title
$PARMS M=2,SIG=.02,2,D=200,
AX=10,BY=10, X=100,Y=0,
B0=.1,NB=3,BM=10$
Next Model
$PARMS SIG(2)=1, D=400$END
```

(See Appendix 2 for a complete input/output example.)

VAX OPERATING INSTRUCTIONS

Assuming program HRZRECT and all associated subprograms have been compiled and linked using the VAX/VMS operating system, the following steps are general execution guidelines (note that many variations are possible using VMS in either time-sharing or batch modes):

1. Either assign (via $ASSIGN command) an input parameter

file name to the logical name FOR005, or let FOR005 default to the users terminal input (if logged-in on-line). The order of the parameters on FOR005 must be given exactly as defined in the section PARAMETERS REQUIRED above. To assign FOR005, use the DCL command:

```
$ASSIGN parameter_file_name FOR005
```

2. If IPLT>0 is selected, then specific output file names may be assigned to FOR012 and FOR013 (as in step 1); otherwise, the system will assume FOR012.DAT and FOR013.DAT as file names for FOR012 and FOR013, respectively. When IPLT=0 (default), this step may be ignored.

3. Program HRZRECT may be executed with the DCL command:

```
$RUN HRZRECT
```

On the USGS system, use the command:

```
$RUN [WANDERSON]HRZRECT
```

The above execution steps can also be submitted (via a $SUBMIT command) to be run in batch mode. For this reason, prompting messages and user responses have been excluded from program HRZRECT. VAX system-dependent commands and calls have been minimized in HRZRECT for ease of program conversion to other systems (see Appendix 1 for information on conversion problems).

Note that FOR016 is a duplicate (print) disk file (normally called FOR016.DAT, unless assigned otherwise), and file FOR006 is usually the on-line terminal print file (or LOG file if $SUBMIT was used).

ERROR MESSAGES

Most $PARMS syntactical errors are flagged and printed on files FOR006 and FOR016, and the job is aborted. If FOR005 was previously assigned to a disk parameter file, then correct the parameter file using any VAX editor and rerun the job (e.g., use $RUN or $SUBMIT). Other parameter errors (or omissions) are also flagged by program HRZRECT, and the job is terminated. The messages "ICK<0...AFTER ZSUBA1" or "NEVAL()>MXEVAL...AFTER ZSUBA1" may result if MXEVAL or EPS2 are too small when attempting near source problems.

PRINTED OUTPUT

Results are printed on files FOR006 and FOR016. Refer to Appendix 2 for a sample output listing of file FOR016 and corresponding input file FOR005. For each problem (title, $PARMS) set, a title line is printed and a complete NAMELIST write is given for all default and initial $PARMS values, as defined above. The next page repeats the title line, followed by several lines of results defined in the following table:

NAMES/TERMS PRINTED OUTPUT DEFINITIONS

F — Frequency (Hertz) corresponding to the induction number (B), where F is the given value if NF is not 0 (otherwise, F is computed from B).

B — Induction number (see (5)) corresponding to the frequency (F), where B is the given value if NB is not 0 (otherwise, B is computed from F).

FIELD — The selected field component name given by $PARMS ICOMP.

Re,Im — Real and imaginary parts of the unnormalized field component.

AMP — Amplitude of the unnormalized field component.

AMP Z/ZO — Amplitude of the mutual coupling ratio, where ZO is defined by Poddar (1982, p. 104).

PHZ Z/ZO — Phase (in degrees) of the mutual coupling ratio, where ZO is defined by Poddar (1982, p. 104). (The phase of the unnormalized field component is the same as PHZ Z/ZO for points outside and 180+PHZ(Z/ZO) inside the loop source.)

ERR(Re,Im) — Maximum relative errors achieved in the four complex adaptive Gaussian quadratures defined in (4) or (7). These values should be less than $PARMS EPS2. However, if (X,Y) is chosen too close to the rectangular loop source, then a small value in ERR(Re,Im) does not necessarily guarantee EPS2 accuracy in the field component; a test for "closeness" is not made nor attempted in this program.

X,Y — The observation point (given or generated as X within Y if distance soundings selected).

CPU TIME — The total integration CPU-time (in seconds) is always given for all Hankel and finite complex integrations in the selected soundings. If INFO>0, then additional "debugging" information is also provided (see $PARMS INFO>0 above).

REFERENCES

Anderson, W.L., 1982, Fast Hankel transforms using related and lagged convolutions: Assoc. for Computing Mach. Trans. on Math. Software, v. 8, no. 4, p. 344-368.

----------, 1979, Numerical integration of related Hankel transforms of orders 0 and 1 by adaptive digital filtering: Geophysics, v. 44, no. 7, p. 1287-1305.

Frischknecht, F.C., 1967, Fields about an oscillating magnetic dipole over a two-layer earth and application to ground and electromagnetic surveys: Quarterly of the Colo. School of Mines, v. 62, no. 1, 326 p.

Kauahikaua, J., 1978, Electromagnetic fields about a horizontal electric wire source of arbitrary length: Geophysics, v. 43, no. 5, p. 1019-1022.

Poddar, M., 1983, A rectangular loop source of current on multilayered earth: Geophysics, v. 48, no. 1, p. 107-109.

-------, 1982, A rectangular loop source of current on a two-layered earth: Geophysical Prospecting, v. 30, p. 101-114.

Ryu, J., Morrison, H.F., and Ward, S.H., 1970, Electromagnetic fields about a loop source of current: Geophysics, v. 35, no. 5, p. 862-896.

Wait, J.R., 1958, Induction by an oscillating dipole over a two-layer ground: Appl. Sci. Res., v. B-7, p. 73-80.

-------, 1966, Fields of a horizontal dipole over a stratified anisotropic half-space: IEEE Trans. on Antennas and Propagation, v. AP-14, no. 6, p. 790-792.

Appendix 1.-- Conversion to other systems

This program (and associated subprograms) was written in extended ANSI-standard FORTRAN-77 for the VAX-11/780 system. Conversion to systems without an ANSI-FORTRAN-77 compiler would necessitate extensive changes, particularly for all CHARACTER-type variables, IF-THEN-ELSE phrases, etc.

Changes for non-VAX systems might include some (or all) of the following FORTRAN-77 constructs and VAX concepts:

(1) Variables with more than 6-characters.
(2) Character strings delimited by single-quote characters (e.g., 'STRING'); also, character string concatination (e.g., 'STRING1'//'STRING2').
(3) Passing variable-length character strings in subroutine calls; e.g., CHARACTER*(*) passed length character arguments.
(4) Suppression of arithmetic or exponential underflow messages; note that a VAX-11 result is automatically set to 0.0 after any underflow--which is assumed for this program package. If the target system does not set underflows to 0.0, and suppress warning messages, then a suitable conversion procedure must be used for proper operation of this program package.
(5) VAX non-ANSI NAMELIST input and output statements.

Appendix 2.-- Test problem input/output listing

The following input file (FOR005) was used to run a test problem for program HRZRECT on a VAX system. The corresponding output file (FOR016) is given following FOR005.

FOR005

```
SAMPLE PROBLEM
$PARMS M=2,AX=150,BY=155,X=800,
SIG=.01,.2,D=200,INFO=1,
F0=1,NF=3,FM=10000$
```

FOR016

```
<HRZRECT>: SAMPLE PROBLEM

$PARMS
M       =          2,
SIG     =  9.9999998E-03,  0.2000000    , 8*0.0000000E+00,
D       =   200.0000    , 8*0.0000000E+00,
AX      =   150.0000    ,
BY      =   155.0000    ,
X       =   800.0000    ,
Y       =  0.0000000E+00,
DX      =  0.0000000E+00,
DY      =  0.0000000E+00,
XM      =  0.0000000E+00,
YM      =  0.0000000E+00,
EPS     =  1.0000000E-10,
EPS2    =  9.9999997E-05,
MXEVAL  =        500,
RFAC    =   5.000000    ,
FO      =   1.000000    ,
NF      =          3,
FM      =   10000.00    ,
FNF     = 50*0.0000000E+00,
BO      =  0.0000000E+00,
NB      =          0,
BM      =  0.0000000E+00,
BNB     = 50*0.0000000E+00,
INFO    =          1,
IPLT    =          0,
ICOMP   =          2
$END
```

```
<HRZRECT>:     SAMPLE PROBLEM                                                                                              Y=  0.00000E+00

  F           B          FIELD  Re           Im           AMP          AMP Z/ZO     PHZ Z/ZO     ERR (Re)     ERR (Im)      X

0.10000E+01 0.15895E+00 HZ=  0.16140E-04  0.12450E-05  0.16188E-04  0.10641E+01  0.44108E+01  0.11973E-04 -0.70289E-06  0.80000E+03
0.21544E+01 0.23331E+00 HZ=  0.17098E-04  0.15371E-05  0.17167E-04  0.11284E+01  0.51371E+01  0.12490E-04 -0.16581E-05  0.80000E+03
0.46416E+01 0.34245E+00 HZ=  0.18375E-04  0.11735E-05  0.18412E-04  0.12103E+01  0.36543E+01  0.13782E-04 -0.35135E-05  0.80000E+03
0.10000E+02 0.50265E+00 HZ=  0.19221E-04 -0.30767E-06  0.19223E-04  0.12636E+01 -0.91707E+00  0.17447E-04 -0.51446E-05  0.80000E+03
0.21544E+02 0.73780E+00 HZ=  0.18561E-04 -0.24240E-05  0.18719E-04  0.12304E+01 -0.74405E+01  0.24553E-04 -0.11320E-05  0.80000E+03
0.46416E+02 0.10829E+01 HZ=  0.16483E-04 -0.38879E-05  0.16935E-04  0.11132E+01 -0.13272E+02  0.30456E-04 -0.12919E-04  0.80000E+03
0.10000E+03 0.15895E+01 HZ=  0.14223E-04 -0.45516E-05  0.14934E-04  0.98165E+00 -0.17745E+02  0.30157E-04 -0.24445E-04  0.80000E+03
0.21544E+03 0.23331E+01 HZ=  0.12066E-04 -0.54584E-05  0.13243E-04  0.87051E+00 -0.24341E+02  0.29939E-04 -0.28664E-04  0.80000E+03
0.46416E+03 0.34245E+01 HZ=  0.89157E-05 -0.70904E-05  0.11391E-04  0.74878E+00 -0.38494E+02  0.29932E-04 -0.34227E-04  0.80000E+03
0.10000E+04 0.50265E+01 HZ=  0.33499E-05 -0.74376E-05  0.81572E-05  0.53619E+00 -0.65753E+02  0.30773E-04 -0.41745E-04  0.80000E+03
0.21544E+04 0.73780E+01 HZ= -0.40940E-06 -0.33633E-05  0.33882E-05  0.22271E+00 -0.96940E+02 -0.24543E-05 -0.29183E-04  0.80000E+03
0.46416E+04 0.10829E+02 HZ=  0.31068E-08 -0.12345E-05  0.12345E-05  0.81149E-01 -0.89856E+02 -0.28547E-04 -0.33532E-06  0.80000E+03
0.10000E+05 0.15895E+02 HZ=  0.20278E-09 -0.59395E-06  0.59395E-06  0.39041E-01 -0.89980E+02  0.35672E-04 -0.47562E-06  0.80000E+03


0.10000E+01 0.15895E+00 HR=  0.48466E-06  0.16529E-05  0.17225E-05  0.11323E+00  0.73658E+02  0.16039E-05  0.45436E-06  0.80000E+03
0.21544E+01 0.23331E+00 HR=  0.13500E-05  0.29164E-05  0.32137E-05  0.21125E+00  0.65161E+02  0.14487E-05  0.27782E-05  0.80000E+03
0.46416E+01 0.34245E+00 HR=  0.32351E-05  0.44522E-05  0.55034E-05  0.36175E+00  0.53997E+02  0.32573E-06  0.79441E-05  0.80000E+03
0.10000E+02 0.50265E+00 HR=  0.63602E-05  0.54567E-05  0.83802E-05  0.55085E+00  0.40628E+02  0.40480E-05  0.13865E-04  0.80000E+03
0.21544E+02 0.73780E+00 HR=  0.98815E-05  0.49686E-05  0.11060E-04  0.72702E+00  0.26694E+02  0.19919E-06  0.15620E-04  0.80000E+03
0.46416E+02 0.10829E+01 HR=  0.12283E-04  0.32893E-05  0.12716E-04  0.83585E+00  0.14991E+02  0.15085E-06  0.20186E-04  0.80000E+03
0.10000E+03 0.15895E+01 HR=  0.13348E-04  0.17221E-05  0.13459E-04  0.88469E+00  0.73515E+01  0.21837E-05 -0.92025E-04  0.80000E+03
0.21544E+03 0.23331E+01 HR=  0.13992E-04  0.50024E-06  0.14001E-04  0.92033E+00  0.20475E+01  0.54460E-05 -0.39133E-04  0.80000E+03
0.46416E+03 0.34245E+01 HR=  0.14565E-04 -0.14022E-05  0.14633E-04  0.96185E+00 -0.54987E+01  0.10584E-04 -0.19116E-04  0.80000E+03
0.10000E+04 0.50265E+01 HR=  0.13184E-04 -0.52011E-05  0.14172E-04  0.93159E+00 -0.21530E+02  0.21594E-04 -0.35487E-05  0.80000E+03
0.21544E+04 0.73780E+01 HR=  0.74755E-05 -0.65820E-05  0.99602E-05  0.65471E+00 -0.41363E+02  0.73945E-05 -0.26293E-04  0.80000E+03
0.46416E+04 0.10829E+02 HR=  0.44462E-05 -0.41950E-05  0.61128E-05  0.40181E+00 -0.43335E+02  0.14722E-04 -0.64542E-05  0.80000E+03
0.10000E+05 0.15895E+02 HR=  0.30203E-05 -0.29193E-05  0.42006E-05  0.27611E+00 -0.44026E+02  0.11000E-04 -0.10789E-04  0.80000E+03

** TOTAL INTEGRATION CPU TIME =  5.45      SEC.

   RMIN=  0.13000000E+03  RMAX=  0.48128086E+04  NR=   21
   LAST: NOFUN=   70  NEVAL(1:4)=    7    7    0    7

$$ TOTAL GLOBAL CPU TIME FOR ALL SOUNDINGS =        5.45      SEC.
```

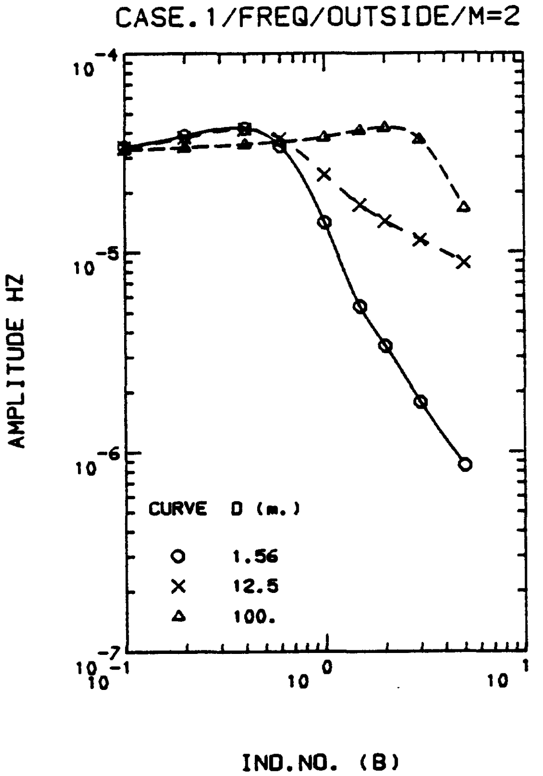
Appendix 3.-- Some sounding curve example plots

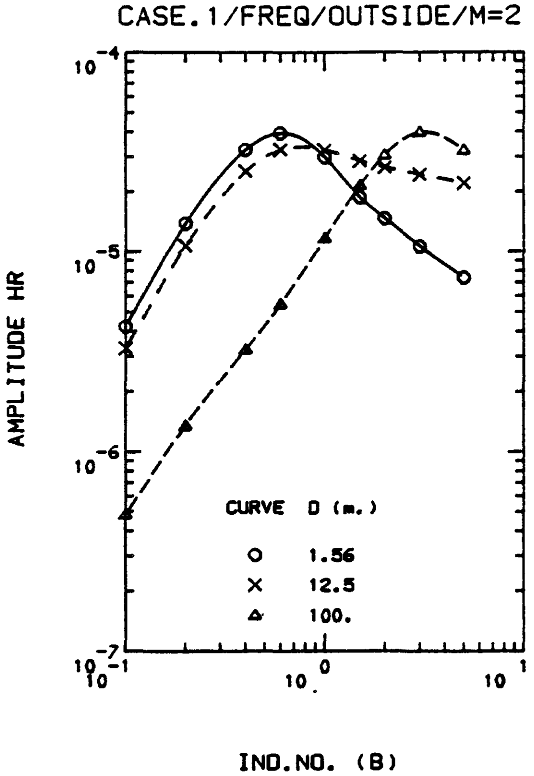
The attached plots were produced (after using IPLT>0) for Hz and Hr fields for several layered models, and curve families, by varying certain model parameters. The following table summarizes the attributes of each CASE plotted and listed in this appendix:

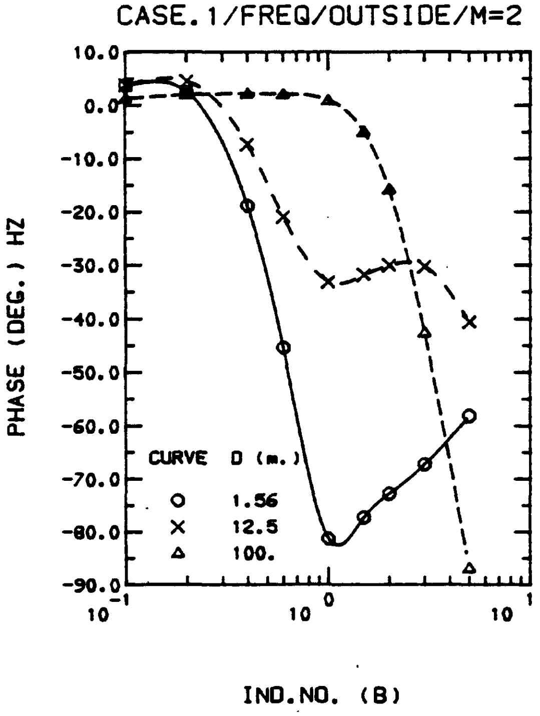
CASE	TYPE/LOCATION	M	VARY	REFERENCE
1	FREQ/OUTSIDE	2	D	Poddar (1983, Fig. 2, p. 108)
2	FREQ/inside	2	D	" (inside loop)
3	DIST/inside	3	D(2)	Poddar (1983, Fig. 3, p. 109)
4	DIST/OUTSIDE	3	D(2)	" (OUTSIDE loop)
5	DIST/NEAR	3	D(2)	" (NEAR loop)

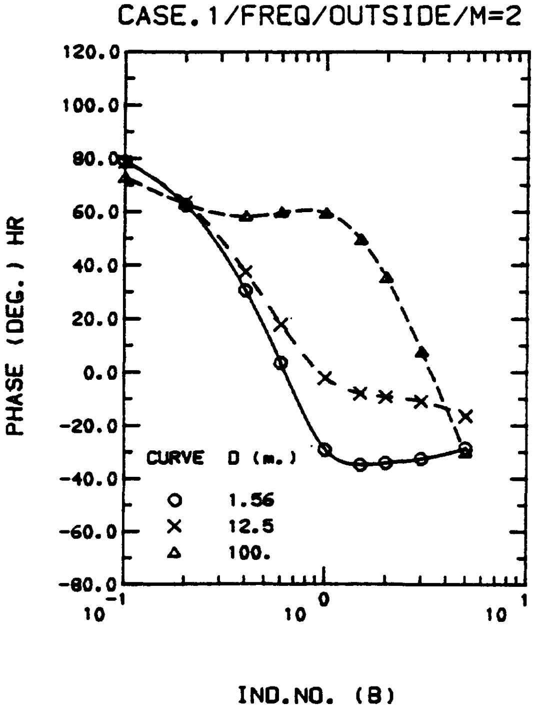
```
<HRZRECT>: CASE.1/FREQ/OUTSIDE/M=2

$PARMS
M       =          2,
SIG     =  9.9999998E-03,  0.3000000     , 8*0.0000000E+00,
D       =   1.563000     , 8*0.0000000E+00,
AX      =   10.00000     ,
BY      =   10.00000     ,
X       =  0.0000000E+00,
Y       =   100.0000     ,
DX      =  0.0000000E+00,
DY      =  0.0000000E+00,
XM      =  0.0000000E+00,
YM      =  0.0000000E+00,
.EPS    =  1.0000000E-10,
EPS2    =  9.9999997E-05,
MXEVAL  =        500,
RFAC    =   5.000000     ,
F0      =  0.0000000E+00,
NF      =          0,
FM      =  0.0000000E+00,
FNF     = 50*0.0000000E+00,
B0      =  0.0000000E+00,
NB      =         -9,
BM      =  0.0000000E+00,
BNB     =  0.1000000     ,  0.2000000     ,  0.4000000     ,  0.6000000     ,
            1.000000     ,   1.500000     ,   2.000000     ,   3.000000     ,   5.000000     ,
          41*0.0000000E+00,
INFO    =          0,
IPLT    =          2,
ICOMP   =          2
$END
```

```
<HRZRECT>:     CASE.1/FREQ/OUTSIDE/M=2                                                                            Y=  0.10000E+03

  F           B          FIELD Re           Im           AMP          AMP Z/ZO     PHZ Z/ZO     ERR (Re)     ERR (Im)      X

0.25330E+02 0.10000E+00 HZ=  0.33872E-04  0.22022E-05  0.33943E-04  0.10505E+01  0.37200E+01  0.74635E-06 -0.10985E-06  0.00000E+00
0.10132E+03 0.20000E+00 HZ=  0.38682E-04  0.16782E-05  0.38718E-04  0.11982E+01  0.24843E+01  0.86234E-06  0.00000E+00  0.00000E+00
0.40528E+03 0.40000E+00 HZ=  0.39835E-04 -0.13570E-04  0.42083E-04  0.13024E+01 -0.18812E+02  0.15699E-05 -0.31520E-06  0.00000E+00
0.91189E+03 0.60000E+00 HZ=  0.24058E-04 -0.24351E-04  0.34231E-04  0.10594E+01 -0.45347E+02  0.43047E-05 -0.86956E-06  0.00000E+00
0.25330E+04 0.10000E+01 HZ=  0.21412E-05 -0.13920E-04  0.14084E-04  0.43586E+00 -0.81255E+02  0.13595E-04 -0.34707E-05  0.00000E+00
0.56993E+04 0.15000E+01 HZ=  0.11822E-05 -0.52094E-05  0.53418E-05  0.16532E+00 -0.77214E+02  0.96278E-06 -0.26701E-05  0.00000E+00
0.10132E+05 0.20000E+01 HZ=  0.10096E-05 -0.32455E-05  0.33989E-05  0.10519E+00 -0.72721E+02  0.31893E-05 -0.26164E-05  0.00000E+00
0.22797E+05 0.30000E+01 HZ=  0.68518E-06 -0.16319E-05  0.17699E-05  0.54774E-01 -0.67224E+02  0.28454E-05 -0.26908E-05  0.00000E+00
0.63326E+05 0.50000E+01 HZ=  0.45112E-06 -0.72430E-06  0.85330E-06  0.26408E-01 -0.58084E+02  0.28981E-05 -0.25612E-05  0.00000E+00

0.25330E+02 0.10000E+00 HR=  0.81624E-06  0.41388E-05  0.42186E-05  0.13055E+00  0.78844E+02  0.14935E-06  0.00000E+00  0.00000E+00
0.10132E+03 0.20000E+00 HR=  0.63612E-05  0.12268E-04  0.13819E-04  0.42766E+00  0.62592E+02  0.00000E+00  0.95081E-07  0.00000E+00
0.40528E+03 0.40000E+00 HR=  0.27912E-04  0.16493E-04  0.32421E-04  0.10034E+01  0.30579E+02  0.00000E+00  0.16657E-05  0.00000E+00
0.91189E+03 0.60000E+00 HR=  0.39205E-04  0.24186E-05  0.39280E-04  0.12156E+01  0.35302E+01  0.00000E+00 -0.97020E-06  0.00000E+00
0.25330E+04 0.10000E+01 HR=  0.25957E-04 -0.14437E-04  0.29702E-04  0.91920E+00 -0.29083E+02  0.10530E-05 -0.26146E-06  0.00000E+00
0.56993E+04 0.15000E+01 HR=  0.15310E-04 -0.10698E-04  0.18677E-04  0.57801E+00 -0.34945E+02  0.65411E-06 -0.70264E-06  0.00000E+00
0.10132E+05 0.20000E+01 HR=  0.12110E-04 -0.82254E-05  0.14639E-04  0.45306E+00 -0.34185E+02  0.70806E-06 -0.49872E-06  0.00000E+00
0.22797E+05 0.30000E+01 HR=  0.89322E-05 -0.56943E-05  0.10593E-04  0.32783E+00 -0.32517E+02  0.63619E-06 -0.67073E-06  0.00000E+00
0.63326E+05 0.50000E+01 HR=  0.64790E-05 -0.35255E-05  0.73761E-05  0.22827E+00 -0.28552E+02  0.65607E-06 -0.69594E-06  0.00000E+00

** TOTAL INTEGRATION CPU TIME =  3.76      SEC.




<HRZRECT>:     (D=12.5)                                                                                           Y=  0.10000E+03

  F           B          FIELD Re           Im           AMP          AMP Z/ZO     PHZ Z/ZO     ERR (Re)     ERR (Im)      X

0.25330E+02 0.10000E+00 HZ=  0.33671E-04  0.22434E-05  0.33746E-04  0.10444E+01  0.38119E+01  0.64984E-06  0.00000E+00  0.00000E+00
0.10132E+03 0.20000E+00 HZ=  0.37745E-04  0.29955E-05  0.37864E-04  0.11718E+01  0.45375E+01  0.82669E-06  0.00000E+00  0.00000E+00
0.40528E+03 0.40000E+00 HZ=  0.41377E-04 -0.52972E-05  0.41715E-04  0.12910E+01 -0.72954E+01  0.11523E-05 -0.69457E-07  0.00000E+00
0.91189E+03 0.60000E+00 HZ=  0.34830E-04 -0.13313E-04  0.37288E-04  0.11540E+01 -0.20918E+02  0.18090E-05 -0.30626E-06  0.00000E+00
0.25330E+04 0.10000E+01 HZ=  0.20560E-04 -0.13405E-04  0.24544E-04  0.75958E+00 -0.33103E+02  0.26906E-05 -0.12726E-05  0.00000E+00
0.56993E+04 0.15000E+01 HZ=  0.14652E-04 -0.90799E-05  0.17237E-04  0.53346E+00 -0.31787E+02  0.25298E-05 -0.19116E-05  0.00000E+00
0.10132E+05 0.20000E+01 HZ=  0.12369E-04 -0.71457E-05  0.14285E-04  0.44209E+00 -0.30015E+02  0.25721E-05 -0.18335E-05  0.00000E+00
0.22797E+05 0.30000E+01 HZ=  0.99126E-05 -0.57895E-05  0.11480E-04  0.35526E+00 -0.30287E+02  0.26005E-05 -0.20042E-05  0.00000E+00
0.63326E+05 0.50000E+01 HZ=  0.66993E-05 -0.57285E-05  0.88146E-05  0.27279E+00 -0.40533E+02  0.28453E-05 -0.22213E-05  0.00000E+00

0.25330E+02 0.10000E+00 HR=  0.64377E-06  0.32198E-05  0.32835E-05  0.10162E+00  0.78693E+02  0.82885E-07  0.64278E-07  0.00000E+00
0.10132E+03 0.20000E+00 HR=  0.46933E-05  0.95428E-05  0.10634E-04  0.32911E+00  0.63811E+02  0.11333E-06  0.00000E+00  0.00000E+00
0.40528E+03 0.40000E+00 HR=  0.20001E-04  0.15337E-04  0.25205E-04  0.78003E+00  0.37481E+02  0.00000E+00  0.29618E-06  0.00000E+00
0.91189E+03 0.60000E+00 HR=  0.30861E-04  0.10027E-04  0.32449E-04  0.10042E+01  0.17999E+02  0.12966E-06 -0.46627E-05  0.00000E+00
0.25330E+04 0.10000E+01 HR=  0.32230E-04 -0.11906E-05  0.32252E-04  0.99812E+00 -0.21156E+01  0.23554E-06 -0.12989E-05  0.00000E+00
0.56993E+04 0.15000E+01 HR=  0.28139E-04 -0.39713E-05  0.28418E-04  0.87946E+00 -0.80332E+01  0.36812E-06 -0.48696E-06  0.00000E+00
0.10132E+05 0.20000E+01 HR=  0.26051E-04 -0.42864E-05  0.26402E-04  0.81707E+00 -0.93435E+01  0.50224E-06 -0.21269E-06  0.00000E+00
0.22797E+05 0.30000E+01 HR=  0.23838E-04 -0.46396E-05  0.24285E-04  0.75157E+00 -0.11014E+02  0.67027E-06  0.00000E+00  0.00000E+00
0.63326E+05 0.50000E+01 HR=  0.21094E-04 -0.62992E-05  0.22015E-04  0.68130E+00 -0.16627E+02  0.65228E-06 -0.27626E-06  0.00000E+00

** TOTAL INTEGRATION CPU TIME =  3.49      SEC.




<HRZRECT>:     (D=100)                                                                                            Y=  0.10000E+03

  F           B          FIELD Re           Im           AMP          AMP Z/ZO     PHZ Z/ZO     ERR (Re)     ERR (Im)      X

0.25330E+02 0.10000E+00 HZ=  0.32812E-04  0.73163E-06  0.32821E-04  0.10157E+01  0.12773E+01  0.73022E-06  0.00000E+00  0.00000E+00
0.10132E+03 0.20000E+00 HZ=  0.33741E-04  0.12159E-05  0.33763E-04  0.10449E+01  0.20638E+01  0.64991E-06  0.00000E+00  0.00000E+00
0.40528E+03 0.40000E+00 HZ=  0.35071E-04  0.13738E-05  0.35098E-04  0.10862E+01  0.22432E+01  0.66935E-06  0.00000E+00  0.00000E+00
0.91189E+03 0.60000E+00 HZ=  0.35966E-04  0.13256E-05  0.35991E-04  0.11138E+01  0.21108E+01  0.68642E-06 -0.10449E-06  0.00000E+00
0.25330E+04 0.10000E+01 HZ=  0.38001E-04  0.62336E-06  0.38006E-04  0.11762E+01  0.93978E+00  0.84741E-06 -0.99267E-07  0.00000E+00
0.56993E+04 0.15000E+01 HZ=  0.40902E-04 -0.35305E-05  0.41054E-04  0.12705E+01 -0.49333E+01  0.10444E-05 -0.24107E-06  0.00000E+00
0.10132E+05 0.20000E+01 HZ=  0.40744E-04 -0.11529E-04  0.42344E-04  0.13105E+01 -0.15800E+02  0.13960E-05 -0.20288E-06  0.00000E+00
0.22797E+05 0.30000E+01 HZ=  0.27106E-04 -0.24897E-04  0.36805E-04  0.11390E+01 -0.42568E+02  0.39088E-05 -0.71475E-06  0.00000E+00
0.63326E+05 0.50000E+01 HZ=  0.92287E-06 -0.16666E-04  0.16692E-04  0.51657E+00 -0.86831E+02 -0.26485E-04 -0.34928E-05  0.00000E+00

0.25330E+02 0.10000E+00 HR=  0.14329E-06  0.46137E-06  0.48311E-06  0.14951E-01  0.72747E+02  0.16793E-06  0.90977E-07  0.00000E+00
0.10132E+03 0.20000E+00 HR=  0.61970E-06  0.12095E-05  0.13591E-05  0.42060E-01  0.62872E+02  0.83395E-07  0.65757E-07  0.00000E+00
0.40528E+03 0.40000E+00 HR=  0.17054E-05  0.27564E-05  0.32414E-05  0.10031E+00  0.58255E+02  0.00000E+00  0.00000E+00  0.00000E+00
0.91189E+03 0.60000E+00 HR=  0.27370E-05  0.47121E-05  0.54493E-05  0.16864E+00  0.59851E+02  0.90189E-07  0.00000E+00  0.00000E+00
0.25330E+04 0.10000E+01 HR=  0.59080E-05  0.10020E-04  0.11632E-04  0.35999E+00  0.59475E+02  0.11647E-06  0.10063E-06  0.00000E+00
0.56993E+04 0.15000E+01 HR=  0.13962E-04  0.16291E-04  0.21456E-04  0.66400E+00  0.49404E+02  0.00000E+00  0.19612E-06  0.00000E+00
0.10132E+05 0.20000E+01 HR=  0.24914E-04  0.17614E-04  0.30512E-04  0.94427E+00  0.35261E+02  0.00000E+00  0.66209E-06  0.00000E+00
0.22797E+05 0.30000E+01 HR=  0.39394E-04  0.52234E-05  0.39738E-04  0.12298E+01  0.75531E+01  0.12730E-06 -0.92491E-06  0.00000E+00
0.63326E+05 0.50000E+01 HR=  0.27814E-04 -0.16326E-04  0.32251E-04  0.99811E+00 -0.30412E+02  0.10453E-05 -0.30218E-06  0.00000E+00

** TOTAL INTEGRATION CPU TIME =  3.87      SEC.


$$ TOTAL GLOBAL CPU TIME FOR ALL SOUNDINGS =        11.1      SEC.
```


CASE.1/FREQ/OUTSIDE/M=2
AMPLITUDE HZ
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-1}
10^{0}
10^{1}
IND.NO. (B)
CURVE D (m.)
○ 1.56
× 12.5
△ 100.

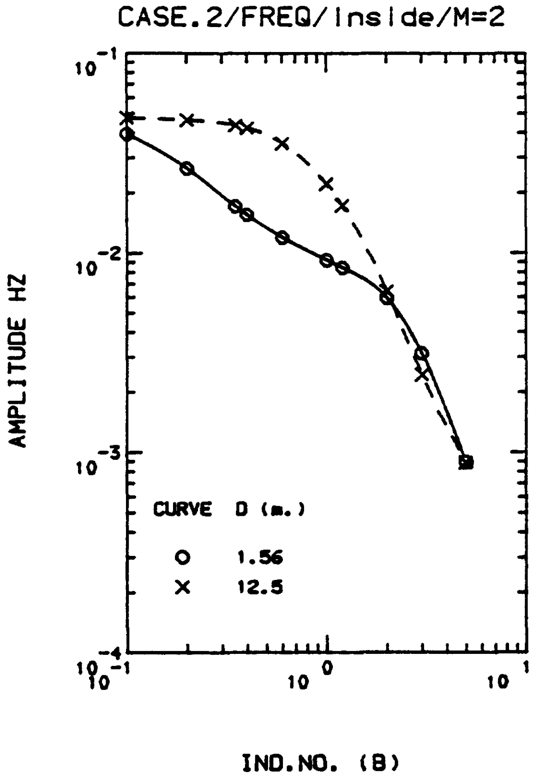

CASE.1/FREQ/OUTSIDE/M=2
AMPLITUDE HR
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-1}
10^{0}
10^{1}
IND.NO. (B)
CURVE D (m.)
○ 1.56
× 12.5
△ 100.

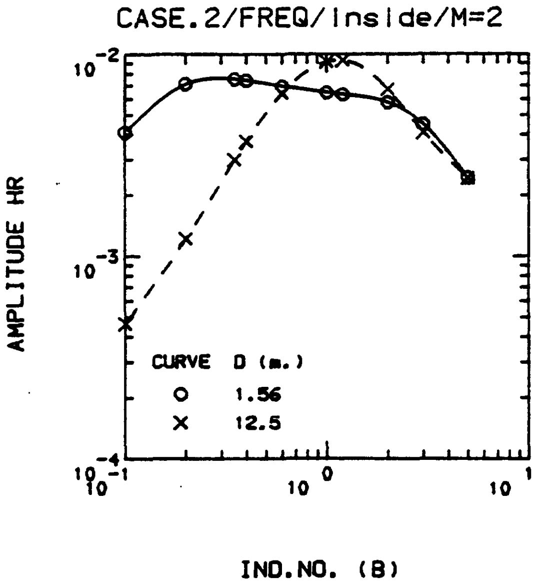

CASE.1/FREQ/OUTSIDE/M=2
PHASE (DEG.) HZ
10.0
0.0
-10.0
-20.0
-30.0
-40.0
-50.0
-60.0
-70.0
-80.0
-90.0
10^{-1}
10^{0}
10^{1}
IND.NO. (B)
CURVE D (m.)
○ 1.56
× 12.5
△ 100.

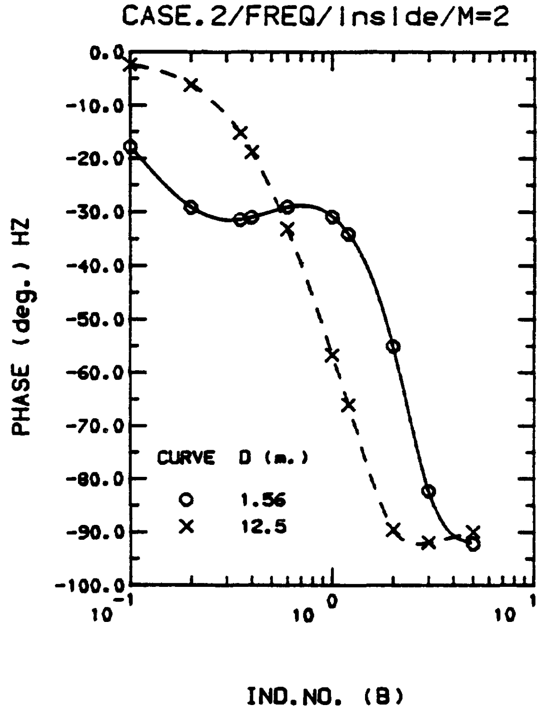

CASE.1/FREQ/OUTSIDE/M=2
PHASE (DEG.) HR
120.0
100.0
80.0
60.0
40.0
20.0
0.0
-20.0
-40.0
-60.0
-80.0
10^{-1}
10^{0}
10^{1}
IND.NO. (B)
CURVE D (m.)
○ 1.56
× 12.5
△ 100.

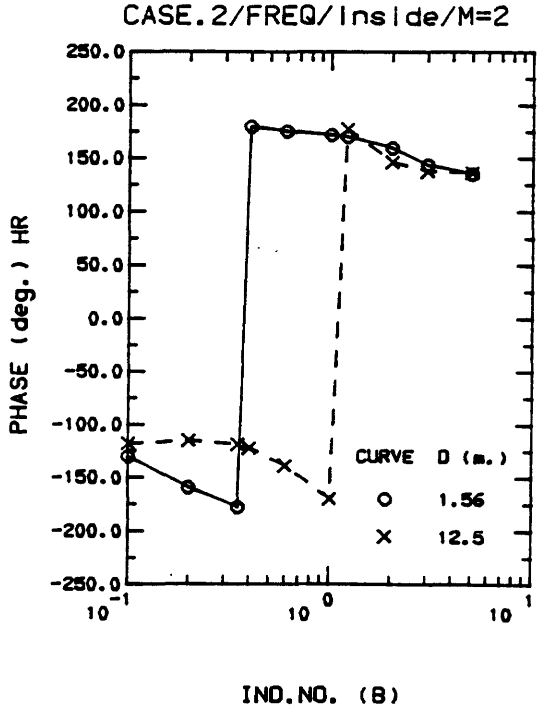
```
<HRZRECT>: CASE.2/FREQ/inside/M=2

$PARMS
M        =          2,
SIG      =  9.9999998E-03,  0.3000000     , 8*0.0000000E+00,
D        =   1.563000     , 8*0.0000000E+00,
AX       =   10.00000     ,
BY       =   10.00000     ,
X        =   2.000000     ,
Y        =   3.000000     ,
DX       =  0.0000000E+00,
DY       =  0.0000000E+00,
XM       =  0.0000000E+00,
YM       =  0.0000000E+00,
EPS      =  1.0000000E-10,
EPS2     =  9.9999997E-05,
MXEVAL   =        500,
RFAC     =   5.000000     ,
FO       =  0.0000000E+00,
NF       =          0,
FM       =  0.0000000E+00,
FNF      = 50*0.0000000E+00,
BO       =  0.0000000E+00,
NB       =        -10,
BM       =  0.0000000E+00,
BNB      =  0.1000000     ,  0.2000000     ,  0.3500000     ,  0.4000000     ,
            0.6000000     ,   1.000000     ,   1.200000     ,   2.000000     ,   3.000000     ,
             5.000000     , 40*0.0000000E+00,
INFO     =          0,
IPLT     =          2,
ICOMP    =          2
$END
```

```
<HRZRECT>:     CASE.2/FREQ/inside/M=2                                                                                Y=  0.30000E+01

  F            B          FIELD  Re              Im              AMP             AMP Z/ZO        PHZ Z/ZO        ERR (Re)        ERR (Im)        X

0.19485E+05 0.10000E+00 HZ= -0.37508E-01  0.12054E-01  0.39397E-01  0.80449E+00 -0.17816E+02  0.11153E-04 -0.83950E-06  0.20000E+01
0.77939E+05 0.20000E+00 HZ= -0.23178E-01  0.12914E-01  0.26533E-01  0.54180E+00 -0.29125E+02  0.21084E-04 -0.10799E-04  0.20000E+01
0.23869E+06 0.35000E+00 HZ= -0.14617E-01  0.89339E-02  0.17131E-01  0.34981E+00 -0.31434E+02  0.17764E-04 -0.16088E-04  0.20000E+01
0.31176E+06 0.40000E+00 HZ= -0.13301E-01  0.79752E-02  0.15509E-01  0.31669E+00 -0.30946E+02  0.17644E-04 -0.15699E-04  0.20000E+01
0.70145E+06 0.60000E+00 HZ= -0.10434E-01  0.57931E-02  0.11934E-01  0.24370E+00 -0.29040E+02  0.17817E-04 -0.16342E-04  0.20000E+01
0.19485E+07 0.10000E+01 HZ= -0.78931E-02  0.47203E-02  0.91969E-02  0.18780E+00 -0.30880E+02  0.18262E-04 -0.16862E-04  0.20000E+01
0.28058E+07 0.12000E+01 HZ= -0.69709E-02  0.47171E-02  0.84169E-02  0.17187E+00 -0.34086E+02  0.21604E-04 -0.13532E-04  0.20000E+01
0.77939E+07 0.20000E+01 HZ= -0.33910E-02  0.48627E-02  0.59283E-02  0.12106E+00 -0.55110E+02  0.29613E-04 -0.19367E-04  0.20000E+01
0.17536E+08 0.30000E+01 HZ= -0.41349E-03  0.30994E-02  0.31268E-02  0.63850E-01 -0.82401E+02  0.18984E-04 -0.25457E-04  0.20000E+01
0.48712E+08 0.50000E+01 HZ=  0.34383E-04  0.89236E-03  0.89303E-03  0.18236E-01 -0.92207E+02 -0.65028E-04 -0.21684E-04  0.20000E+01

0.19485E+05 0.10000E+00 HR=  0.26568E-02  0.31105E-02  0.40907E-02  0.83533E-01 -0.13050E+03  0.44427E-05  0.32678E-04  0.20000E+01
0.77939E+05 0.20000E+00 HR=  0.66667E-02  0.25250E-02  0.71289E-02  0.14557E+00 -0.15926E+03  0.21211E-05 -0.30502E-04  0.20000E+01
0.23869E+06 0.35000E+00 HR=  0.75415E-02  0.28968E-03  0.75471E-02  0.15411E+00 -0.17780E+03  0.33395E-06 -0.48751E-04  0.20000E+01
0.31176E+06 0.40000E+00 HR=  0.74109E-02 -0.52656E-04  0.74111E-02  0.15134E+00  0.17959E+03  0.10952E-05 -0.24170E-04  0.20000E+01
0.70145E+06 0.60000E+00 HR=  0.69030E-02 -0.59422E-03  0.69285E-02  0.14148E+00  0.17508E+03  0.55338E-05 -0.12292E-05  0.20000E+01
0.19485E+07 0.10000E+01 HR=  0.63849E-02 -0.94246E-03  0.64541E-02  0.13179E+00  0.17160E+03  0.58558E-05 -0.28396E-05  0.20000E+01
0.28058E+07 0.12000E+01 HR=  0.62109E-02 -0.11185E-02  0.63108E-02  0.12887E+00  0.16979E+03  0.59597E-05 -0.33202E-05  0.20000E+01
0.77939E+07 0.20000E+01 HR=  0.53787E-02 -0.20408E-02  0.57528E-02  0.11747E+00  0.15922E+03  0.67098E-05 -0.49899E-05  0.20000E+01
0.17536E+08 0.30000E+01 HR=  0.36081E-02 -0.26758E-02  0.44920E-02  0.91728E-01  0.14344E+03  0.69540E-05 -0.63508E-05  0.20000E+01
0.48712E+08 0.50000E+01 HR=  0.17285E-02 -0.17217E-02  0.24397E-02  0.49819E-01  0.13511E+03  0.68566E-05 -0.65675E-05  0.20000E+01

** TOTAL INTEGRATION CPU TIME =  5.28     SEC.



<HRZRECT>:     (D=12.5)                                                                                              Y=  0.30000E+01

  F            B          FIELD  Re              Im              AMP             AMP Z/ZO        PHZ Z/ZO        ERR (Re)        ERR (Im)        X

0.19485E+05 0.10000E+00 HZ= -0.47437E-01  0.19742E-02  0.47478E-01  0.96951E+00 -0.23832E+01  0.7U976E-05 -0.44282E-06  0.20000E+01
0.77939E+05 0.20000E+00 HZ= -0.46019E-01  0.49412E-02  0.46283E-01  0.94511E+00 -0.61286E+01  0.74732E-05 -0.76347E-06  0.20000E+01
0.23869E+06 0.35000E+00 HZ= -0.42056E-01  0.11454E-01  0.43587E-01  0.89006E+00 -0.15235E+02  0.87643E-05 -0.13120E-05  0.20000E+01
0.31176E+06 0.40000E+00 HZ= -0.39977E-01  0.13607E-01  0.42229E-01  0.86232E+00 -0.18797E+02  0.97065E-05 -0.15073E-05  0.20000E+01
0.70145E+06 0.60000E+00 HZ= -0.29569E-01  0.19292E-01  0.35306E-01  0.72095E+00 -0.33121E+02  0.16984E-04 -0.36946E-05  0.20000E+01
0.19485E+07 0.10000E+01 HZ= -0.12135E-01  0.18489E-01  0.22115E-01  0.45160E+00 -0.56721E+02  0.67671E-04 -0.18284E-04  0.20000E+01
0.28058E+07 0.12000E+01 HZ= -0.69796E-02  0.15776E-01  0.17251E-01  0.35227E+00 -0.66135E+02  0.53865E-04 -0.13640E-06  0.20000E+01
0.77939E+07 0.20000E+01 HZ= -0.48336E-04  0.64097E-02  0.64099E-02  0.13089E+00 -0.89568E+02 -0.53853E-04 -0.13320E-04  0.20000E+01
0.17536E+08 0.30000E+01 HZ=  0.84461E-04  0.24309E-02  0.24324E-02  0.49669E-01 -0.91990E+02  0.39046E-04 -0.19610E-06  0.20000E+01
0.48712E+08 0.50000E+01 HZ= -0.40113E-06  0.88539E-03  0.88539E-03  0.18080E-01 -0.89974E+02  0.88674E-04 -0.71021E-06  0.20000E+01

0.19485E+05 0.10000E+00 HR=  0.21693E-03  0.40822E-03  0.46228E-03  0.94397E-02 -0.11799E+03  0.00000E+00  0.91215E-05  0.20000E+01
0.77939E+05 0.20000E+00 HR=  0.51461E-03  0.11005E-02  0.12149E-02  0.24807E-01 -0.11506E+03  0.15026E-05  0.21028E-04  0.20000E+01
0.23869E+06 0.35000E+00 HR=  0.14445E-02  0.26196E-02  0.29914E-02  0.61085E-01 -0.11887E+03  0.20998E-05  0.36883E-04  0.20000E+01
0.31176E+06 0.40000E+00 HR=  0.19680E-02  0.31228E-02  0.36912E-02  0.75375E-01 -0.12222E+03  0.22426E-05  0.44804E-04  0.20000E+01
0.70145E+06 0.60000E+00 HR=  0.48083E-02  0.41952E-02  0.63812E-02  0.13030E+00 -0.13890E+03  0.11748E-04  0.92798E-04  0.20000E+01
0.19485E+07 0.10000E+01 HR=  0.90242E-02  0.15657E-02  0.91590E-02  0.18703E+00 -0.17016E+03  0.12600E-05 -0.82862E-04  0.20000E+01
0.28058E+07 0.12000E+01 HR=  0.92933E-02 -0.43555E-03  0.93035E-02  0.18998E+00  0.17732E+03  0.47549E-06 -0.12939E-04  0.20000E+01
0.77939E+07 0.20000E+01 HR=  0.55781E-02 -0.37336E-02  0.67123E-02  0.13707E+00  0.14620E+03  0.10494E-04 -0.35340E-05  0.20000E+01
0.17536E+08 0.30000E+01 HR=  0.30147E-02 -0.27671E-02  0.40921E-02  0.83562E-01  0.13745E+03  0.83941E-05 -0.79035E-05  0.20000E+01
0.48712E+08 0.50000E+01 HR=  0.17506E-02 -0.16768E-02  0.24241E-02  0.49501E-01  0.13623E+03  0.67839E-05 -0.66523E-05  0.20000E+01

** TOTAL INTEGRATION CPU TIME =  7.51     SEC.

$$ TOTAL GLOBAL CPU TIME FOR ALL SOUNDINGS =        12.8     SEC.
```


CASE.2/FREQ/Inside/M=2
AMPLITUDE HZ
10^-1
10^-2
10^-3
10^-4
10^-1
10^0
10^1
IND.NO. (B)
CURVE D (m.)
O 1.56
X 12.5


CASE.2/FREQ/Inside/M=2
AMPLITUDE HR
10^-2
10^-3
10^-4
10^-1
10^0
10^1
IND.NO. (B)
CURVE D (m.)
O 1.56
X 12.5

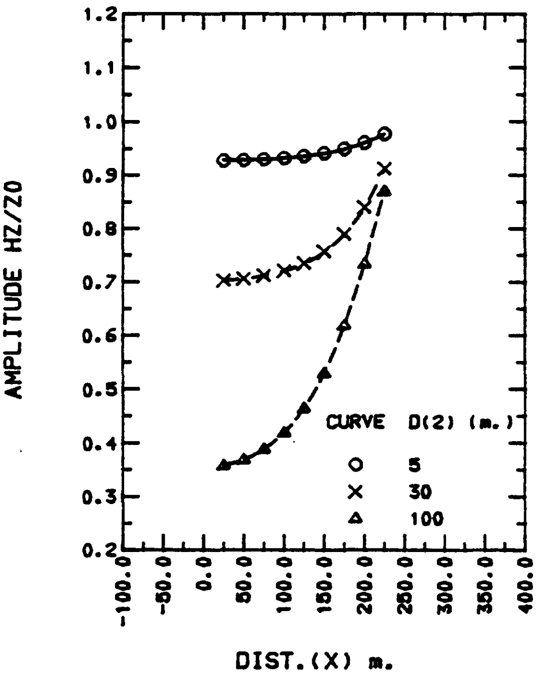

CASE.2/FREQ/Inside/M=2
PHASE (deg.) HZ
0.0
-10.0
-20.0
-30.0
-40.0
-50.0
-60.0
-70.0
-80.0
-90.0
-100.0
10^-1
10^0
10^1
IND.NO. (B)
CURVE D (m.)
O 1.56
X 12.5


CASE.2/FREQ/Inside/M=2
PHASE (deg.) HR
250.0
200.0
150.0
100.0
50.0
0.0
-50.0
-100.0
-150.0
-200.0
-250.0
10^-1
10^0
10^1
IND.NO. (B)
CURVE D (m.)
O 1.56
X 12.5

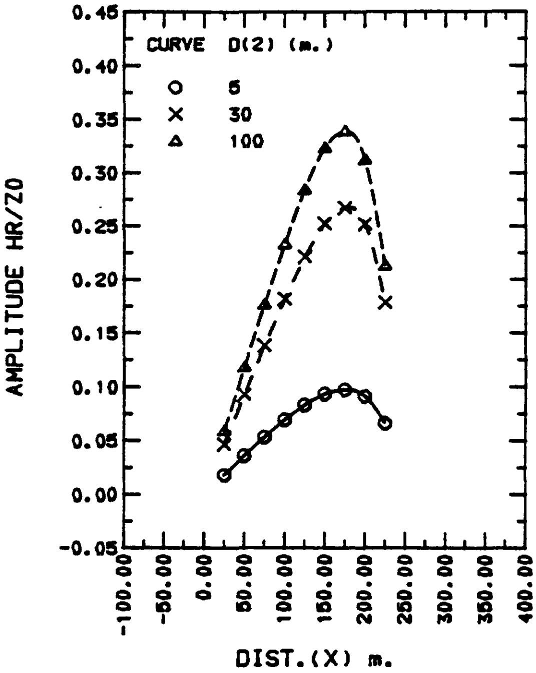
```
<HRZRECT>: CASE.3/DIST/inside/M=3

$PARMS
M       =          3,
SIG     =  9.9999998E-03,  2.9999999E-02,  1.0000000E-03, 7*0.0000000E+00,
D       =   3.000000     ,   5.000000     , 7*0.0000000E+00,
AX      =   250.0000     ,
BY      =   250.0000     ,
X       =   25.00000     ,
Y       =  0.0000000E+00,
DX      =   25.00000     ,
DY      =  0.0000000E+00,
XM      =   225.0000     ,
YM      =  0.0000000E+00,
EPS     =  1.0000000E-10,
EPS2    =  9.9999997E-05,
MXEVAL  =        500,
RFAC    =   5.000000     ,
FO      =  0.0000000E+00,
NF      =         -1,
FM      =  0.0000000E+00,
FNF     =   1344.000     , 49*0.0000000E+00,
BO      =  0.0000000E+00,
NB      =          0,
BM      =  0.0000000E+00,
BNB     = 50*0.0000000E+00,
INFO    =          0,
IPLT    =          1,
ICOMP   =          2
$END
```

```
<HRZRECT>:    CASE.3/DIST/inside/M=3                                                                                   Y=  0.00000E+00

  F             B          FIELD  Re           Im           AMP          AMP Z/ZO     PHZ Z/ZO      ERR (Re)     ERR (Im)      X

0.13440E+04 0.18210E+00 HZ= -0.15773E-02  0.58610E-03  0.16827E-02  0.92864E+00 -0.20384E+02  0.37789E-04 -0.10398E-04  0.25000E+02
0.13440E+04 0.36421E+00 HZ= -0.16152E-02  0.58438E-03  0.17176E-02  0.92956E+00 -0.19891E+02  0.15319E-04 -0.48772E-05  0.50000E+02
0.13440E+04 0.54631E+00 HZ= -0.16841E-02  0.58138E-03  0.17816E-02  0.93117E+00 -0.19046E+02  0.53545E-04 -0.13931E-04  0.75000E+02
0.13440E+04 0.72842E+00 HZ= -0.17954E-02  0.57685E-03  0.18858E-02  0.93368E+00 -0.17812E+02  0.46777E-04 -0.11016E-04  0.10000E+03
0.13440E+04 0.91052E+00 HZ= -0.19707E-02  0.57037E-03  0.20516E-02  0.93737E+00 -0.16142E+02  0.31213E-04 -0.32286E-05  0.12500E+03
0.13440E+04 0.10926E+01 HZ= -0.22550E-02  0.56120E-03  0.23238E-02  0.94274E+00 -0.13975E+02  0.39911E-04 -0.69353E-05  0.15000E+03
0.13440E+04 0.12747E+01 HZ= -0.27541E-02  0.54789E-03  0.28080E-02  0.95055E+00 -0.11251E+02  0.31810E-04 -0.64692E-05  0.17500E+03
0.13440E+04 0.14568E+01 HZ= -0.37863E-02  0.52685E-03  0.38228E-02  0.96186E+00 -0.79216E+01  0.68785E-05 -0.13226E-05  0.20000E+03
0.13440E+04 0.16389E+01 HZ= -0.69438E-02  0.48447E-03  0.69607E-02  0.97798E+00 -0.39911E+01  0.20780E-04 -0.14944E-05  0.22500E+03


0.13440E+04 0.18210E+00 HR=  0.12301E-04  0.30770E-04  0.33137E-04  0.18288E-01 -0.11179E+03  0.44767E-06  0.19621E-04  0.25000E+02
0.13440E+04 0.36421E+00 HR=  0.24578E-04  0.62367E-04  0.67035E-04  0.36279E-01 -0.11151E+03  0.46812E-05  0.44597E-04  0.50000E+02
0.13440E+04 0.54631E+00 HR=  0.36805E-04  0.95697E-04  0.10253E-03  0.53588E-01 -0.11104E+03  0.63871E-06  0.58999E-05  0.75000E+02
0.13440E+04 0.72842E+00 HR=  0.48946E-04  0.13184E-03  0.14063E-03  0.69631E-01 -0.11037E+03  0.80839E-07  0.77609E-05  0.10000E+03
0.13440E+04 0.91052E+00 HR=  0.60948E-04  0.17224E-03  0.18270E-03  0.83478E-01 -0.10949E+03  0.12898E-05  0.10066E-05  0.12500E+03
0.13440E+04 0.10926E+01 HR=  0.72734E-04  0.21899E-03  0.23076E-03  0.93617E-01 -0.10837E+03  0.23501E-05  0.89676E-05  0.15000E+03
0.13440E+04 0.12747E+01 HR=  0.84181E-04  0.27568E-03  0.28825E-03  0.97575E-01 -0.10698E+03  0.42847E-06  0.44114E-05  0.17500E+03
0.13440E+04 0.14568E+01 HR=  0.95081E-04  0.34955E-03  0.36225E-03  0.91147E-01 -0.10522E+03  0.21697E-06  0.37173E-05  0.20000E+03
0.13440E+04 0.16389E+01 HR=  0.10499E-03  0.45998E-03  0.47181E-03  0.66290E-01 -0.10286E+03  0.81536E-06  0.32448E-05  0.22500E+03

** TOTAL INTEGRATION CPU TIME =  2.08     SEC.




<HRZRECT>:    D2=30                                                                                                    Y=  0.00000E+00

  F             B          FIELD  Re           Im           AMP          AMP Z/ZO     PHZ Z/ZO      ERR (Re)     ERR (Im)      X

0.13440E+04 0.18210E+00 HZ= -0.71226E-03  0.10573E-02  0.12748E-02  0.70352E+00 -0.56032E+02  0.86510E-04 -0.13807E-04  0.25000E+02
0.13440E+04 0.36421E+00 HZ= -0.75867E-03  0.10631E-02  0.13061E-02  0.70683E+00 -0.54487E+02  0.38773E-04 -0.73252E-05  0.50000E+02
0.13440E+04 0.54631E+00 HZ= -0.84267E-03  0.10721E-02  0.13636E-02  0.71271E+00 -0.51833E+02  0.13026E-04 -0.43155E-05  0.75000E+02
0.13440E+04 0.72842E+00 HZ= -0.97660E-03  0.10827E-02  0.14581E-02  0.72194E+00 -0.47951E+02  0.97513E-04 -0.16301E-04  0.10000E+03
0.13440E+04 0.91052E+00 HZ= -0.11839E-02  0.10922E-02  0.16108E-02  0.73597E+00 -0.42693E+02  0.39476E-04 -0.19888E-05  0.12500E+03
0.13440E+04 0.10926E+01 HZ= -0.15119E-02  0.10952E-02  0.18669E-02  0.75738E+00 -0.35918E+02  0.88004E-04 -0.79662E-05  0.15000E+03
0.13440E+04 0.12747E+01 HZ= -0.20696E-02  0.10807E-02  0.23348E-02  0.79035E+00 -0.27573E+02  0.68731E-04 -0.96817E-05  0.17500E+03
0.13440E+04 0.14568E+01 HZ= -0.31794E-02  0.10250E-02  0.33405E-02  0.84051E+00 -0.17869E+02  0.18132E-04 -0.86818E-05  0.20000E+03
0.13440E+04 0.16389E+01 HZ= -0.64369E-02  0.86769E-03  0.64951E-02  0.91257E+00 -0.76771E+01  0.49594E-04 -0.18152E-05  0.22500E+03


0.13440E+04 0.18210E+00 HR=  0.69116E-04  0.48163E-04  0.84242E-04  0.46491E-01 -0.14513E+03  0.28021E-05  0.59199E-04  0.25000E+02
0.13440E+04 0.36421E+00 HR=  0.13902E-03  0.10010E-03  0.17130E-03  0.92708E-01 -0.14424E+03  0.10658E-05 -0.53639E-04  0.50000E+02
0.13440E+04 0.54631E+00 HR=  0.21041E-03  0.15992E-03  0.26428E-03  0.13813E+00 -0.14276E+03  0.26549E-05  0.85147E-04  0.75000E+02
0.13440E+04 0.72842E+00 HR=  0.28381E-03  0.23247E-03  0.36686E-03  0.18164E+00 -0.14068E+03  0.70774E-05  0.45388E-04  0.10000E+03
0.13440E+04 0.91052E+00 HR=  0.35940E-03  0.32385E-03  0.48379E-03  0.22104E+00 -0.13798E+03  0.24906E-05  0.10490E-04  0.12500E+03
0.13440E+04 0.10926E+01 HR=  0.43673E-03  0.44223E-03  0.62153E-03  0.25215E+00 -0.13464E+03  0.61177E-06  0.13555E-04  0.15000E+03
0.13440E+04 0.12747E+01 HR=  0.51415E-03  0.59894E-03  0.78936E-03  0.26721E+00 -0.13064E+03  0.46499E-05  0.49006E-04  0.17500E+03
0.13440E+04 0.14568E+01 HR=  0.58773E-03  0.80970E-03  0.10005E-02  0.25174E+00 -0.12597E+03  0.92771E-07  0.42354E-05  0.20000E+03
0.13440E+04 0.16389E+01 HR=  0.64894E-03  0.10920E-02  0.12703E-02  0.17847E+00 -0.12072E+03  0.00000E+00  0.16767E-04  0.22500E+03

** TOTAL INTEGRATION CPU TIME =  2.10     SEC.




<HRZRECT>:    D2=100                                                                                                   Y=  0.00000E+00

  F             B          FIELD  Re           Im           AMP          AMP Z/ZO     PHZ Z/ZO      ERR (Re)     ERR (Im)      X

0.13440E+04 0.18210E+00 HZ= -0.11827E-03  0.63940E-03  0.65025E-03  0.35885E+00 -0.79520E+02  0.22818E-04 -0.83112E-06  0.25000E+02
0.13440E+04 0.36421E+00 HZ= -0.16553E-03  0.66317E-03  0.68352E-03  0.36991E+00 -0.75985E+02  0.53762E-05 -0.93349E-06  0.50000E+02
0.13440E+04 0.54631E+00 HZ= -0.25327E-03  0.70118E-03  0.74552E-03  0.38965E+00 -0.70140E+02 -0.23678E-04 -0.19564E-04  0.75000E+02
0.13440E+04 0.72842E+00 HZ= -0.39723E-03  0.75027E-03  0.84894E-03  0.42033E+00 -0.62101E+02 -0.82980E-04 -0.24481E-05  0.10000E+03
0.13440E+04 0.91052E+00 HZ= -0.62540E-03  0.80455E-03  0.10190E-02  0.46560E+00 -0.52141E+02  0.52160E-04 -0.30351E-05  0.12500E+03
0.13440E+04 0.10926E+01 HZ= -0.99033E-03  0.85336E-03  0.13073E-02  0.53036E+00 -0.40751E+02 -0.11671E-04 -0.42874E-05  0.15000E+03
0.13440E+04 0.12747E+01 HZ= -0.16061E-02  0.87799E-03  0.18304E-02  0.61962E+00 -0.28664E+02 -0.10799E-04 -0.27360E-05  0.17500E+03
0.13440E+04 0.14568E+01 HZ= -0.27980E-02  0.84478E-03  0.29227E-02  0.73539E+00 -0.16800E+02  0.71102E-05  0.00000E+00  0.20000E+03
0.13440E+04 0.16389E+01 HZ= -0.61602E-02  0.68591E-03  0.61982E-02  0.87086E+00 -0.63535E+01  0.21416E-05 -0.63998E-06  0.22500E+03


0.13440E+04 0.18210E+00 HR=  0.10702E-03  0.37743E-05  0.10709E-03  0.59099E-01 -0.17798E+03  0.24763E-04 -0.78987E-04  0.25000E+02
0.13440E+04 0.36421E+00 HR=  0.21786E-03  0.14182E-04  0.21832E-03  0.11815E+00 -0.17628E+03  0.10273E-04 -0.53180E-04  0.50000E+02
0.13440E+04 0.54631E+00 HR=  0.33583E-03  0.38374E-04  0.33802E-03  0.17667E+00 -0.17348E+03  0.86361E-05 -0.45031E-04  0.75000E+02
0.13440E+04 0.72842E+00 HR=  0.46324E-03  0.84569E-04  0.47090E-03  0.23315E+00 -0.16965E+03  0.13647E-05  0.36996E-04  0.10000E+03
0.13440E+04 0.91052E+00 HR=  0.60050E-03  0.16233E-03  0.62205E-03  0.28422E+00 -0.16487E+03  0.29734E-05 -0.21487E-04  0.12500E+03
0.13440E+04 0.10926E+01 HR=  0.74520E-03  0.28270E-03  0.79702E-03  0.32335E+00 -0.15923E+03  0.24288E-04 -0.29569E-04  0.15000E+03
0.13440E+04 0.12747E+01 HR=  0.89078E-03  0.45772E-03  0.10015E-02  0.33902E+00 -0.15280E+03  0.43220E-05 -0.26354E-04  0.17500E+03
0.13440E+04 0.14568E+01 HR=  0.10249E-02  0.69883E-03  0.12405E-02  0.31212E+00 -0.14571E+03  0.39489E-04 -0.33066E-04  0.20000E+03
0.13440E+04 0.16389E+01 HR=  0.11278E-02  0.10112E-02  0.15148E-02  0.21282E+00 -0.13812E+03  0.24256E-04 -0.57156E-04  0.22500E+03

** TOTAL INTEGRATION CPU TIME =  2.29     SEC.

$$ TOTAL GLOBAL CPU TIME FOR ALL SOUNDINGS =       6.47     SEC.
```

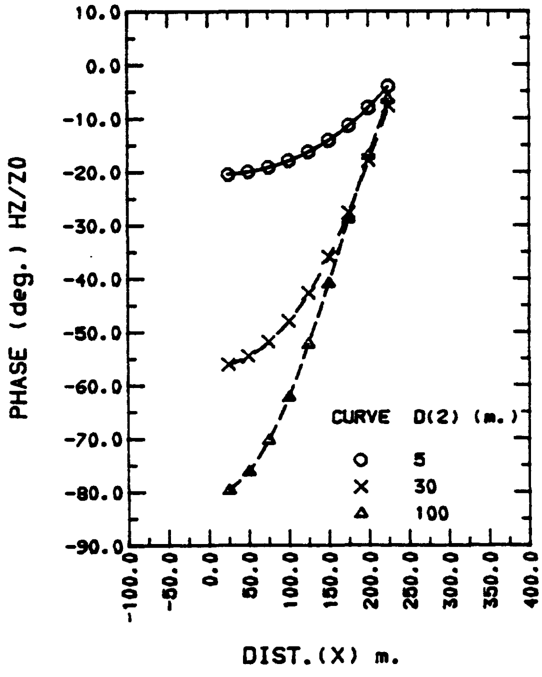
CASE.3/DIST/Inside/M=3



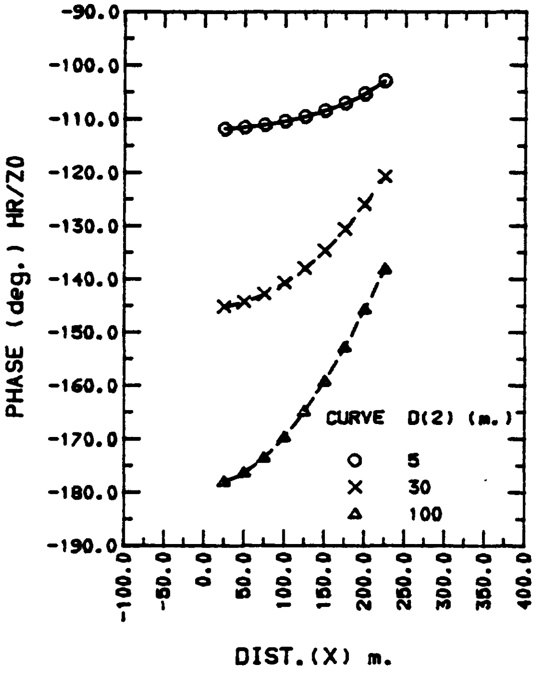
CASE.3/DIST/Inside/M=3



CASE.3/DIST/Inside/M=3



CASE.3/DIST/Inside/M=3

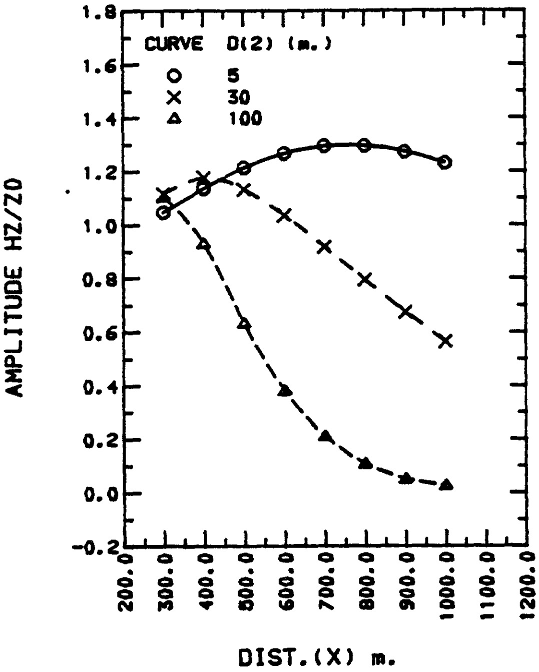


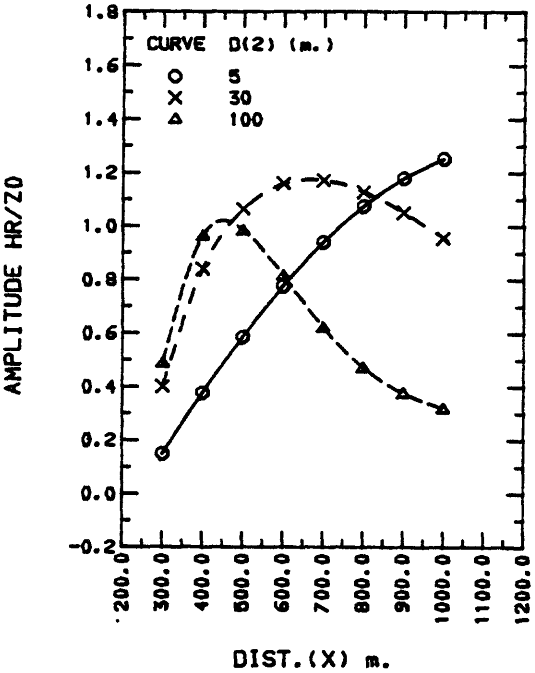
```
<HRZRECT>: CASE.4/DIST/OUTSIDE/M=3

$PARMS
M       =          3,
SIG     =  9.9999998E-03,  2.9999999E-02,  1.0000000E-03, 7*0.0000000E+00,
D       =   3.000000     ,   5.000000     , 7*0.0000000E+00,
AX      =   250.0000     ,
BY      =   250.0000     ,
X       =   300.0000     ,
Y       =  0.0000000E+00,
DX      =   100.0000     ,
DY      =  0.0000000E+00,
XM      =   1000.000     ,
YM      =  0.0000000E+00,
EPS     =  1.0000000E-10,
EPS2    =  9.9999997E-05,
MXEVAL  =        500,
RFAC    =   5.000000     ,
F0      =  0.0000000E+00,
NF      =         -1,
FM      =  0.0000000E+00,
FNF     =   1344.000     , 49*0.0000000E+00,
B0      =  0.0000000E+00,
NB      =          0,
BM      =  0.0000000E+00,
BNB     = 50*0.0000000E+00,
INFO    =          0,
IPLT    =          1,
ICOMP   =          2
$END
```

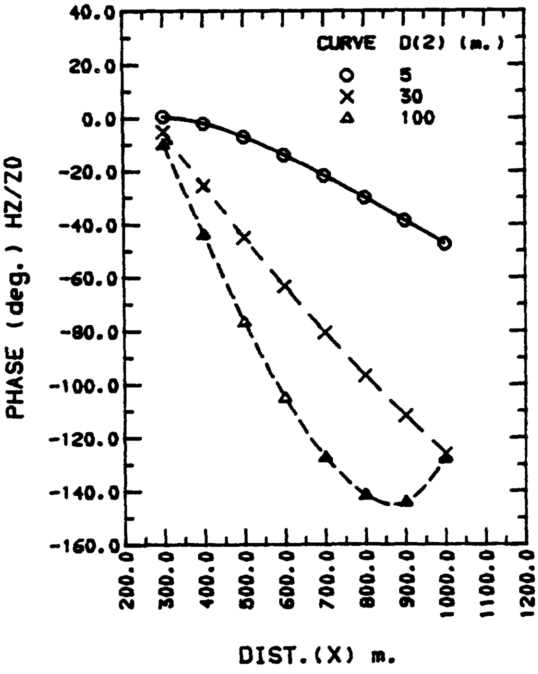
```
<HRZRECT>:     CASE.4/DIST/OUTSIDE/M=3                                                                                Y=  0.00000E+00

  F             B             FIELD  Re            Im            AMP           AMP Z/ZO      PHZ Z/ZO      ERR (Re)      ERR (Im)      X

0.13440E+04 0.21852E+01 HZ=  0.26678E-02  0.28386E-04  0.26680E-02  0.10476E+01  0.60962E+00  0.20478E-04 -0.11623E-05  0.30000E+03
0.13440E+04 0.29137E+01 HZ=  0.63158E-03 -0.21901E-04  0.63196E-03  0.11381E+01 -0.19860E+01  0.24981E-04 -0.21198E-05  0.40000E+03
0.13440E+04 0.36421E+01 HZ=  0.27648E-03 -0.34392E-04  0.27861E-03  0.12152E+01 -0.70908E+01  0.70373E-04 -0.18135E-04  0.50000E+03
0.13440E+04 0.43705E+01 HZ=  0.14637E-03 -0.36030E-04  0.15074E-03  0.12693E+01 -0.13829E+02  0.16900E-05 -0.67424E-06  0.60000E+03
0.13440E+04 0.50989E+01 HZ=  0.84353E-04 -0.33339E-04  0.90703E-04  0.12963E+01 -0.21566E+02  0.53313E-05 -0.28112E-06  0.70000E+03
0.13440E+04 0.58273E+01 HZ=  0.50460E-04 -0.28999E-04  0.58199E-04  0.12966E+01 -0.29886E+02  0.37985E-05 -0.31534E-06  0.80000E+03
0.13440E+04 0.65557E+01 HZ=  0.30482E-04 -0.24260E-04  0.38958E-04  0.12734E+01 -0.38515E+02 -0.62118E-04 -0.90145E-05  0.90000E+03
0.13440E+04 0.72842E+01 HZ=  0.18219E-04 -0.19736E-04  0.26859E-04  0.12305E+01 -0.47289E+02 -0.16860E-04 -0.66124E-05  0.10000E+04

0.13440E+04 0.21852E+01 HR=  0.11110E-03  0.36620E-03  0.38268E-03  0.15026E+00  0.73123E+02  0.60425E-06  0.81130E-05  0.30000E+03
0.13440E+04 0.29137E+01 HR=  0.97380E-04  0.18610E-03  0.21004E-03  0.37826E+00  0.62379E+02  0.32975E-05  0.12424E-04  0.40000E+03
0.13440E+04 0.36421E+01 HR=  0.81834E-04  0.10695E-03  0.13466E-03  0.58738E+00  0.52577E+02  0.22737E-05  0.91404E-05  0.50000E+03
0.13440E+04 0.43705E+01 HR=  0.67355E-04  0.63044E-04  0.92256E-04  0.77683E+00  0.43107E+02  0.82801E-05 -0.44010E-04  0.60000E+03
0.13440E+04 0.50989E+01 HR=  0.54623E-04  0.36739E-04  0.65829E-04  0.94078E+00  0.33925E+02  0.16191E-05  0.27055E-04  0.70000E+03
0.13440E+04 0.58273E+01 HR=  0.43739E-04  0.20439E-04  0.48279E-04  0.10756E+01  0.25047E+02  0.45012E-05 -0.70266E-05  0.80000E+03
0.13440E+04 0.65557E+01 HR=  0.34619E-04  0.10238E-04  0.36101E-04  0.11800E+01  0.16475E+02  0.27746E-05 -0.96440E-04  0.90000E+03
0.13440E+04 0.72842E+01 HR=  0.27102E-04  0.39211E-05  0.27384E-04  0.12545E+01  0.82323E+01  0.26454E-05 -0.11635E-05  0.10000E+04

** TOTAL INTEGRATION CPU TIME =  1.52     SEC.
```

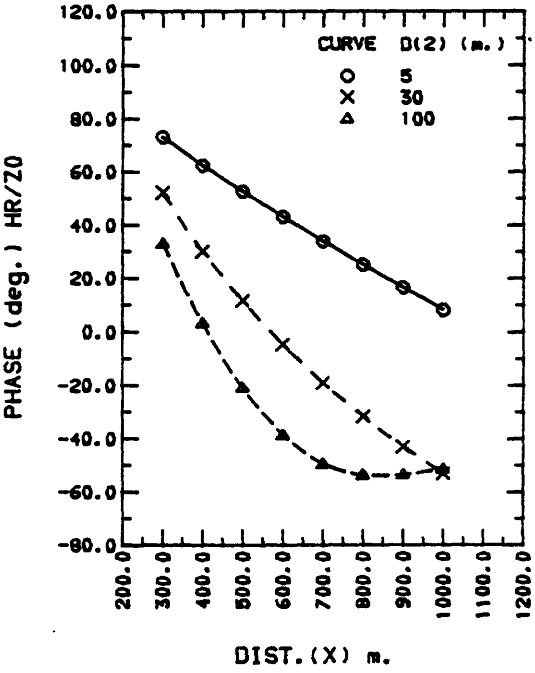
```
<HRZRECT>:     D2=30                                                                                                  Y=  0.00000E+00

  F             B             FIELD  Re            Im            AMP           AMP Z/ZO      PHZ Z/ZD      ERR (Re)      ERR (Im)      X

0.13440E+04 0.21852E+01 HZ=  0.28342E-02 -0.25410E-03  0.28456E-02  0.11173E+01 -0.51231E+01  0.61579E-04 -0.88558E-05  0.30000E+03
0.13440E+04 0.29137E+01 HZ=  0.59102E-03 -0.27946E-03  0.65376E-03  0.11773E+01 -0.25307E+02  0.58456E-05 -0.38722E-06  0.40000E+03
0.13440E+04 0.36421E+01 HZ=  0.18410E-03 -0.18307E-03  0.25963E-03  0.11325E+01 -0.44838E+02  0.51020E-05 -0.64785E-07  0.50000E+03
0.13440E+04 0.43705E+01 HZ=  0.55499E-04 -0.10989E-03  0.12311E-03  0.10366E+01 -0.63205E+02 -0.40394E-04 -0.54458E-06  0.60000E+03
0.13440E+04 0.50989E+01 HZ=  0.10718E-04 -0.63368E-04  0.64268E-04  0.91848E+00 -0.80400E+02 -0.32690E-04 -0.25755E-05  0.70000E+03
0.13440E+04 0.58273E+01 HZ= -0.40494E-05 -0.35433E-04  0.35663E-04  0.79455E+00 -0.96520E+02 -0.65786E-05 -0.32010E-05  0.80000E+03
0.13440E+04 0.65557E+01 HZ= -0.76183E-05 -0.19195E-04  0.20652E-04  0.67501E+00 -0.11165E+03 -0.42186E-06  0.42146E-04  0.90000E+03
0.13440E+04 0.72842E+01 HZ= -0.72253E-05 -0.99819E-05  0.12322E-04  0.56450E+00 -0.12590E+03 -0.13409E-04  0.72282E-04  0.10000E+04

0.13440E+04 0.21852E+01 HR=  0.62393E-03  0.80656E-03  0.10197E-02  0.40039E+00  0.52275E+02  0.25057E-06 -0.61892E-05  0.30000E+03
0.13440E+04 0.29137E+01 HR=  0.40358E-03  0.23368E-03  0.46636E-03  0.83985E+00  0.30072E+02  0.16927E-05 -0.58773E-04  0.40000E+03
0.13440E+04 0.36421E+01 HR=  0.23901E-03  0.48743E-04  0.24393E-03  0.10640E+01  0.11527E+02  0.16703E-05 -0.22211E-04  0.50000E+03
0.13440E+04 0.43705E+01 HR=  0.13736E-03 -0.11190E-04  0.13782E-03  0.11605E+01 -0.46571E+01  0.75495E-04 -0.10472E-04  0.60000E+03
0.13440E+04 0.50989E+01 HR=  0.77543E-04 -0.26673E-04  0.82002E-04  0.11719E+01 -0.18982E+02  0.53910E-04 -0.87075E-05  0.70000E+03
0.13440E+04 0.58273E+01 HR=  0.43078E-04 -0.26620E-04  0.50640E-04  0.11282E+01 -0.31714E+02  0.47273E-04 -0.67605E-04  0.80000E+03
0.13440E+04 0.65557E+01 HR=  0.23508E-04 -0.21928E-04  0.32147E-04  0.10507E+01 -0.43008E+02  0.31390E-04 -0.58596E-05  0.90000E+03
0.13440E+04 0.72842E+01 HR=  0.12552E-04 -0.16650E-04  0.20851E-04  0.95522E+00 -0.52989E+02  0.21552E-04 -0.40230E-05  0.10000E+04

** TOTAL INTEGRATION CPU TIME =  1.51     SEC.
```

```
<HRZRECT>:     D2=100                                                                                                 Y=  0.00000E+00

  F             B             FIELD  Re            Im            AMP           AMP Z/ZO      PHZ Z/ZO      ERR (Re)      ERR (Im)      X

0.13440E+04 0.21852E+01 HZ=  0.27743E-02 -0.48161E-03  0.28158E-02  0.11056E+01 -0.98483E+01  0.70343E-05  0.00000E+00  0.30000E+03
0.13440E+04 0.29137E+01 HZ=  0.37381E-03 -0.35699E-03  0.51689E-03  0.93086E+00 -0.43682E+02 -0.61696E-04 -0.11859E-05  0.40000E+03
0.13440E+04 0.36421E+01 HZ=  0.33822E-04 -0.14117E-03  0.14516E-03  0.63317E+00 -0.76526E+02 -0.12567E-05 -0.16520E-05  0.50000E+03
0.13440E+04 0.43705E+01 HZ= -0.11672E-04 -0.43965E-04  0.45488E-04  0.38303E+00 -0.10487E+03 -0.29517E-04 -0.10951E-04  0.60000E+03
0.13440E+04 0.50989E+01 HZ= -0.89590E-05 -0.11817E-04  0.14829E-04  0.21193E+00 -0.12717E+03 -0.74513E-05 -0.15450E-04  0.70000E+03
0.13440E+04 0.58273E+01 HZ= -0.38048E-05 -0.30247E-05  0.48606E-05  0.10829E+00 -0.14152E+03 -0.60574E-04 -0.34620E-05  0.80000E+03
0.13440E+04 0.65557E+01 HZ= -0.12868E-05 -0.93357E-06  0.15898E-05  0.51964E-01 -0.14404E+03 -0.20124E-04 -0.27206E-05  0.90000E+03
0.13440E+04 0.72842E+01 HZ= -0.36206E-06 -0.47031E-06  0.59353E-06  0.27190E-01 -0.12759E+03  0.22814E-04 -0.23044E-06  0.10000E+04

0.13440E+04 0.21852E+01 HR=  0.10414E-02  0.68197E-03  0.12448E-02  0.48878E+00  0.33219E+02  0.65600E-04 -0.22847E-04  0.30000E+03
0.13440E+04 0.29137E+01 HR=  0.53443E-03  0.30568E-04  0.53530E-03  0.96401E+00  0.32737E+01  0.74293E-04 -0.50033E-04  0.40000E+03
0.13440E+04 0.36421E+01 HR=  0.21081E-03 -0.80433E-04  0.22563E-03  0.98416E+00 -0.20884E+02  0.66908E-04 -0.89256E-04  0.50000E+03
0.13440E+04 0.43705E+01 HR=  0.75725E-04 -0.60467E-04  0.96905E-04  0.81597E+00 -0.38608E+02  0.24589E-04 -0.37469E-04  0.60000E+03
0.13440E+04 0.50989E+01 HR=  0.28343E-04 -0.33096E-04  0.43574E-04  0.62272E+00 -0.49423E+02  0.24316E-04 -0.14998E-04  0.70000E+03
0.13440E+04 0.58273E+01 HR=  0.12560E-04 -0.17086E-04  0.21206E-04  0.47245E+00 -0.53681E+02  0.71348E-05 -0.86339E-05  0.80000E+03
0.13440E+04 0.65557E+01 HR=  0.68642E-05 -0.92268E-05  0.11500E-04  0.37589E+00 -0.53353E+02  0.68802E-05 -0.35098E-05  0.90000E+03
0.13440E+04 0.72842E+01 HR=  0.43570E-05 -0.54170E-05  0.69517E-05  0.31847E+00 -0.51189E+02  0.66908E-04 -0.89256E-04  0.10000E+04

** TOTAL INTEGRATION CPU TIME =  1.58     SEC.

$$ TOTAL GLOBAL CPU TIME FOR ALL SOUNDINGS =        4.61     SEC.
```


CASE.4/DIST/OUTSIDE/M=3

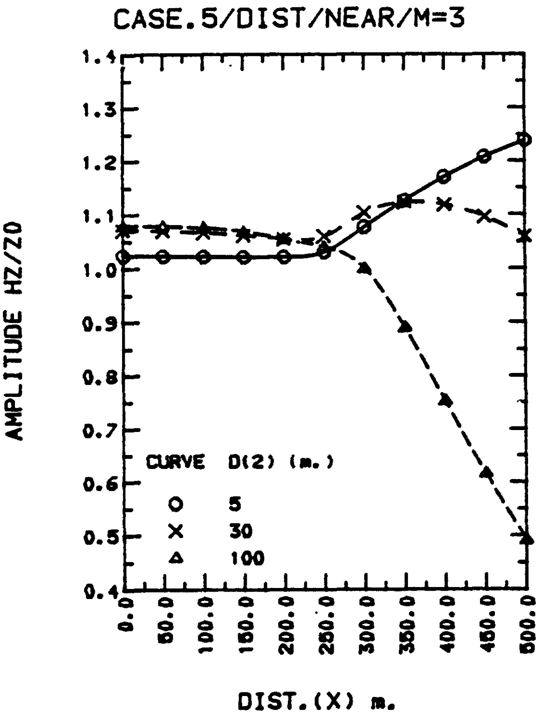

CASE.4/DIST/OUTSIDE/M=3

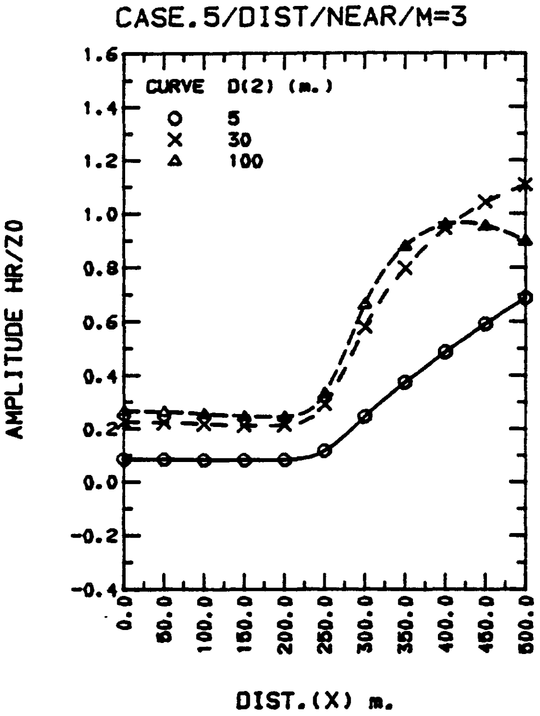

CASE.4/DIST/OUTSIDE/M=3

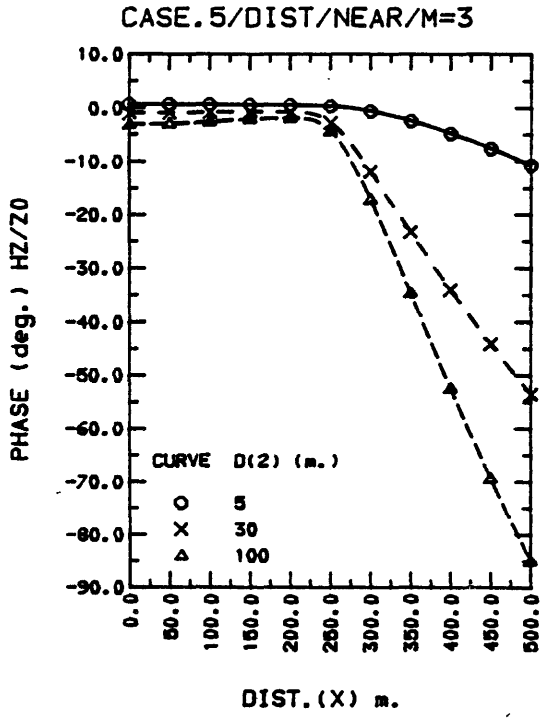

CASE.4/DIST/OUTSIDE/M=3

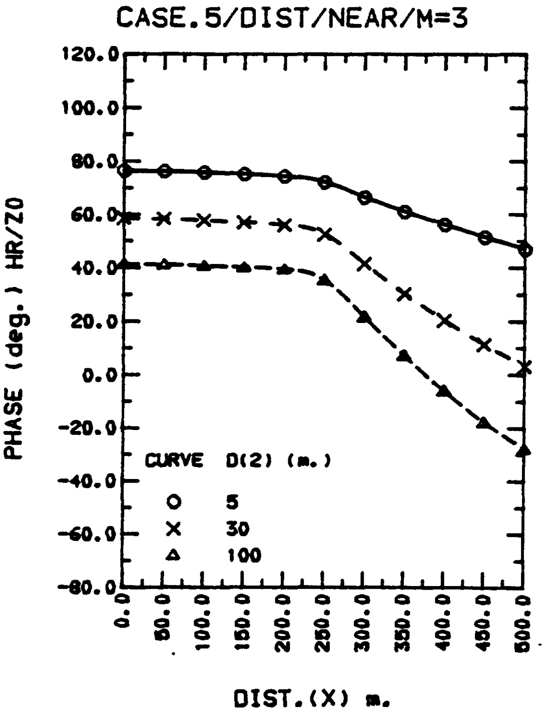
```
<HRZRECT>: CASE.5/DIST/NEAR/M=3

$PARMS
M       =           3,
SIG     =  9.9999998E-03,  2.9999999E-02,  1.0000000E-03, 7*0.0000000E+00,
D       =   3.000000    ,   5.000000    , 7*0.0000000E+00,
AX      =   250.0000    ,
BY      =   250.0000    ,
X       =  0.0000000E+00,
Y       =   275.0000    ,
DX      =   50.00000    ,
DY      =  0.0000000E+00,
XM      =   500.0000    ,
YM      =  0.0000000E+00,
EPS     =  1.0000000E-10,
EPS2    =  9.9999997E-05,
MXEVAL  =         500,
RFAC    =   5.000000    ,
FO      =  0.0000000E+00,
NF      =          -1,
FM      =  0.0000000E+00,
FNF     =   1344.000    , 49*0.0000000E+00,
BO      =  0.0000000E+00,
NB      =           0,
BM      =  0.0000000E+00,
BNB     = 50*0.0000000E+00,
INFO    =           0,
IPLT    =           1,
ICOMP   =           2
$END
```

```
<HRZRECT>:     CASE.5/DIST/NEAR/M=3                                                                                         Y=  0.27500E+03

  F            B          FIELD  Re           Im           AMP          AMP Z/ZO     PHZ Z/ZO     ERR (Re)     ERR (Im)      X

0.13440E+04 0.20031E+01 HZ=  0.58292E-02  0.70746E-04  0.58296E-02  0.10240E+01  0.69534E+00  0.24113E-04 -0.31952E-05  0.00000E+00
0.13440E+04 0.20360E+01 HZ=  0.58054E-02  0.69692E-04  0.58058E-02  0.10238E+01  0.68779E+00  0.89901E-04 -0.93669E-05  0.50000E+02
0.13440E+04 0.21315E+01 HZ=  0.57231E-02  0.66277E-04  0.57234E-02  0.10233E+01  0.66350E+00  0.84132E-04 -0.20304E-04  0.10000E+03
0.13440E+04 0.22818E+01 HZ=  0.55318E-02  0.59539E-04  0.55321E-02  0.10225E+01  0.61665E+00  0.95732E-05  0.00000E+00  0.15000E+03
0.13440E+04 0.24769E+01 HZ=  0.50250E-02  0.46628E-04  0.50252E-02  0.10224E+01  0.53164E+00  0.96897E-04 -0.36602E-04  0.20000E+03
0.13440E+04 0.27072E+01 HZ=  0.30619E-02  0.17322E-04  0.30620E-02  0.10319E+01  0.32414E+00  0.82853E-04 -0.81550E-05  0.25000E+03
0.13440E+04 0.29644E+01 HZ=  0.10953E-02 -0.11892E-04  0.10954E-02  0.10773E+01 -0.62205E+00  0.74932E-05 -0.88790E-07  0.30000E+03
0.13440E+04 0.32423E+01 HZ=  0.57802E-03 -0.24559E-04  0.57854E-03  0.11267E+01 -0.24329E+01  0.95732E-05  0.00000E+00  0.35000E+03
0.13440E+04 0.35358E+01 HZ=  0.36785E-03 -0.30995E-04  0.36915E-03  0.11706E+01 -0.48164E+01  0.84132E-04 -0.20304E-04  0.40000E+03
0.13440E+04 0.38415E+01 HZ=  0.25571E-03 -0.34223E-04  0.25799E-03  0.12080E+01 -0.76228E+01  0.91480E-04 -0.40791E-05  0.45000E+03
0.13440E+04 0.41566E+01 HZ=  0.18671E-03 -0.35473E-04  0.19005E-03  0.12387E+01 -0.10757E+02  0.24113E-04 -0.31952E-05  0.50000E+03


0.13440E+04 0.20031E+01 HR=  0.11299E-03  0.46824E-03  0.48168E-03  0.84611E-01  0.76433E+02  0.96952E-06  0.87415E-05  0.00000E+00
0.13440E+04 0.20360E+01 HR=  0.11297E-03  0.46288E-03  0.47646E-03  0.84022E-01  0.76284E+02  0.84884E-06  0.51323E-05  0.50000E+02
0.13440E+04 0.21315E+01 HR=  0.11288E-03  0.44823E-03  0.46222E-03  0.82640E-01  0.75865E+02  0.13220E-05  0.70626E-05  0.10000E+03
0.13440E+04 0.22818E+01 HR=  0.11253E-03  0.42712E-03  0.44169E-03  0.81641E-01  0.75240E+02  0.26198E-05  0.65432E-04  0.15000E+03
0.13440E+04 0.24769E+01 HR=  0.11126E-03  0.39894E-03  0.41416E-03  0.84263E-01  0.74416E+02  0.73383E-06  0.58354E-05  0.20000E+03
0.13440E+04 0.27072E+01 HR=  0.10736E-03  0.33461E-03  0.35141E-03  0.11843E+00  0.72211E+02  0.67454E-04  0.28840E-04  0.25000E+03
0.13440E+04 0.29644E+01 HR=  0.10026E-03  0.23013E-03  0.25102E-03  0.24688E+00  0.66459E+02  0.12283E-05  0.55335E-04  0.30000E+03
0.13440E+04 0.32423E+01 HR=  0.92845E-04  0.16794E-03  0.19189E-03  0.37372E+00  0.61064E+02  0.13920E-05  0.16940E-04  0.35000E+03
0.13440E+04 0.35358E+01 HR=  0.85662E-04  0.12771E-03  0.15378E-03  0.48765E+00  0.56148E+02  0.22584E-06 -0.72135E-05  0.40000E+03
0.13440E+04 0.38415E+01 HR=  0.78742E-04  0.98917E-04  0.12643E-03  0.59199E+00  0.51479E+02  0.33049E-05  0.22836E-05  0.45000E+03
0.13440E+04 0.41566E+01 HR=  0.72106E-04  0.77192E-04  0.10563E-03  0.68848E+00  0.46951E+02  0.69007E-06 -0.83451E-04  0.50000E+03

** TOTAL INTEGRATION CPU TIME =  2.83      SEC.



<HRZRECT>:     D2=30                                                                                                        Y=  0.27500E+03

  F            B          FIELD  Re           Im           AMP          AMP Z/ZO     PHZ Z/ZO     ERR (Re)     ERR (Im)      X

0.13440E+04 0.20031E+01 HZ=  0.60950E-02 -0.99719E-04  0.60958E-02  0.10708E+01 -0.93733E+00  0.65222E-04 -0.33116E-05  0.00000E+00
0.13440E+04 0.20360E+01 HZ=  0.60663E-02 -0.94420E-04  0.60671E-02  0.10699E+01 -0.89172E+00  0.91145E-04 -0.42335E-05  0.50000E+02
0.13440E+04 0.21315E+01 HZ=  0.59679E-02 -0.80188E-04  0.59684E-02  0.10671E+01 -0.76982E+00  0.40378E-04 -0.17043E-05  0.10000E+03
0.13440E+04 0.22818E+01 HZ=  0.57452E-02 -0.64065E-04  0.57456E-02  0.10620E+01 -0.63888E+00  0.92281E-04 -0.10563E-06  0.15000E+03
0.13440E+04 0.24769E+01 HZ=  0.51849E-02 -0.68519E-04  0.51853E-02  0.10550E+01 -0.75713E+00  0.98059E-04 -0.16841E-04  0.20000E+03
0.13440E+04 0.27072E+01 HZ=  0.31428E-02 -0.15128E-03  0.31465E-02  0.10604E+01 -0.27558E+01  0.84111E-04 -0.83698E-05  0.25000E+03
0.13440E+04 0.29644E+01 HZ=  0.10983E-02 -0.23130E-03  0.11224E-02  0.11038E+01 -0.11893E+02  0.84423E-05 -0.70089E-06  0.30000E+03
0.13440E+04 0.32423E+01 HZ=  0.53001E-03 -0.22747E-03  0.57676E-03  0.11233E+01 -0.23228E+02  0.11849E-04 -0.67594E-06  0.35000E+03
0.13440E+04 0.35358E+01 HZ=  0.29221E-03 -0.19726E-03  0.35255E-03  0.11180E+01 -0.34019E+02 -0.10234E-04 -0.87893E-06  0.40000E+03
0.13440E+04 0.38415E+01 HZ=  0.16792E-03 -0.16276E-03  0.23385E-03  0.10950E+01 -0.44106E+02 -0.10138E-04 -0.41095E-06  0.45000E+03
0.13440E+04 0.41566E+01 HZ=  0.96445E-04 -0.13082E-03  0.16253E-03  0.10593E+01 -0.53600E+02  0.65222E-04 -0.33116E-05  0.50000E+03


0.13440E+04 0.20031E+01 HR=  0.66686E-03  0.10903E-02  0.12780E-02  0.22450E+00  0.58548E+02  0.22988E-06 -0.78597E-05  0.00000E+00
0.13440E+04 0.20360E+01 HR=  0.66122E-03  0.10726E-02  0.12601E-02  0.22220E+00  0.58348E+02  0.15460E-05  0.72517E-04  0.50000E+02
0.13440E+04 0.21315E+01 HR=  0.64556E-03  0.10249E-02  0.12113E-02  0.21657E+00  0.57795E+02  0.73011E-07  0.18370E-04  0.10000E+03
0.13440E+04 0.22818E+01 HR=  0.62160E-03  0.95819E-03  0.11422E-02  0.21111E+00  0.57027E+02  0.31901E-04 -0.85680E-04  0.15000E+03
0.13440E+04 0.24769E+01 HR=  0.58643E-03  0.87194E-03  0.10508E-02  0.21379E+00  0.56077E+02  0.23550E-05  0.20509E-05  0.20000E+03
0.13440E+04 0.27072E+01 HR=  0.52666E-03  0.68666E-03  0.86538E-03  0.29164E+00  0.52512E+02  0.15199E-05  0.70538E-05  0.25000E+03
0.13440E+04 0.29644E+01 HR=  0.43958E-03  0.39026E-03  0.58782E-03  0.57812E+00  0.41599E+02  0.74640E-05  0.30166E-04  0.30000E+03
0.13440E+04 0.32423E+01 HR=  0.35317E-03  0.20707E-03  0.40939E-03  0.79731E+00  0.30384E+02  0.71316E-05  0.77580E-04  0.35000E+03
0.13440E+04 0.35358E+01 HR=  0.27912E-03  0.10373E-03  0.29777E-03  0.94427E+00  0.20387E+02  0.79243E-04 -0.36666E-04  0.40000E+03
0.13440E+04 0.38415E+01 HR=  0.21845E-03  0.43953E-04  0.22283E-03  0.10434E+01  0.11376E+02  0.29689E-04 -0.86653E-04  0.45000E+03
0.13440E+04 0.41566E+01 HR=  0.16973E-03  0.92439E-05  0.16998E-03  0.11079E+01  0.31174E+01  0.82201E-05 -0.82620E-04  0.50000E+03

** TOTAL INTEGRATION CPU TIME =  2.82      SEC.



<HRZRECT>:     D2=100                                                                                                       Y=  0.27500E+03

  F            B          FIELD  Re           Im           AMP          AMP Z/ZO     PHZ Z/ZO     ERR (Re)     ERR (Im)      X

0.13440E+04 0.20031E+01 HZ=  0.61361E-02 -0.32493E-03  0.61447E-02  0.10794E+01 -0.30318E+01  0.63921E-04 -0.91866E-06  0.00000E+00
0.13440E+04 0.20360E+01 HZ=  0.61099E-02 -0.30859E-03  0.61177E-02  0.10788E+01 -0.28914E+01  0.92556E-04 -0.38321E-05  0.50000E+02
0.13440E+04 0.21315E+01 HZ=  0.60143E-02 -0.26273E-03  0.60201E-02  0.10763E+01 -0.25014E+01 -0.52589E-05 -0.64194E-06  0.10000E+03
0.13440E+04 0.22818E+01 HZ=  0.57821E-02 -0.20204E-03  0.57857E-02  0.10694E+01 -0.20012E+01 -0.37748E-04 -0.10552E-04  0.15000E+03
0.13440E+04 0.24769E+01 HZ=  0.51848E-02 -0.16733E-03  0.51875E-02  0.10554E+01 -0.18485E+01  0.99064E-04 -0.15785E-04  0.20000E+03
0.13440E+04 0.27072E+01 HZ=  0.30784E-02 -0.23438E-03  0.30873E-02  0.10404E+01 -0.43540E+01  0.85369E-04 -0.65770E-05  0.25000E+03
0.13440E+04 0.29644E+01 HZ=  0.97238E-03 -0.29984E-03  0.10176E-02  0.10008E+01 -0.17137E+02 -0.15832E-04 -0.46617E-04  0.30000E+03
0.13440E+04 0.32423E+01 HZ=  0.37636E-03 -0.25978E-03  0.45731E-03  0.89063E+00 -0.34615E+02 -0.14895E-04 -0.50466E-04  0.35000E+03
0.13440E+04 0.35358E+01 HZ=  0.14490E-03 -0.18828E-03  0.23758E-03  0.75339E+00 -0.52417E+02 -0.28432E-04 -0.86562E-04  0.40000E+03
0.13440E+04 0.38415E+01 HZ=  0.46415E-04 -0.12319E-03  0.13165E-03  0.61642E+00 -0.69356E+02 -0.22896E-04 -0.94644E-04  0.45000E+03
0.13440E+04 0.41566E+01 HZ=  0.65776E-05 -0.75145E-04  0.75432E-04  0.49165E+00 -0.84998E+02 -0.14230E-04 -0.98197E-04  0.50000E+03


0.13440E+04 0.20031E+01 HR=  0.11358E-02  0.10028E-02  0.15151E-02  0.26615E+00  0.41443E+02  0.53887E-04 -0.85020E-05  0.00000E+00
0.13440E+04 0.20360E+01 HR=  0.11203E-02  0.98303E-03  0.14904E-02  0.26283E+00  0.41266E+02  0.45631E-06  0.59981E-04  0.50000E+02
0.13440E+04 0.21315E+01 HR=  0.10785E-02  0.93058E-03  0.14245E-02  0.25468E+00  0.40789E+02  0.18091E-04 -0.64264E-05  0.10000E+03
0.13440E+04 0.22818E+01 HR=  0.10180E-02  0.86022E-03  0.13328E-02  0.24634E+00  0.40198E+02  0.24036E-05  0.39430E-04  0.15000E+03
0.13440E+04 0.24769E+01 HR=  0.93510E-03  0.77164E-03  0.12124E-02  0.24666E+00  0.39529E+02  0.29025E-05  0.28985E-04  0.20000E+03
0.13440E+04 0.27072E+01 HR=  0.80636E-03  0.57303E-03  0.98923E-03  0.33338E+00  0.35399E+02  0.42828E-05  0.84358E-05  0.25000E+03
0.13440E+04 0.29644E+01 HR=  0.62761E-03  0.25021E-03  0.67565E-03  0.66450E+00  0.21735E+02  0.44558E-05  0.40418E-04  0.30000E+03
0.13440E+04 0.32423E+01 HR=  0.44846E-03  0.56474E-04  0.45200E-03  0.88028E+00  0.71774E+01  0.69198E-06  0.42033E-04  0.35000E+03
0.13440E+04 0.35358E+01 HR=  0.30160E-03 -0.32001E-04  0.30329E-03  0.96177E+00 -0.60567E+01  0.65868E-05 -0.23699E-04  0.40000E+03
0.13440E+04 0.38415E+01 HR=  0.19441E-03 -0.62172E-04  0.20411E-03  0.95569E+00 -0.17735E+02  0.38493E-04 -0.88900E-04  0.45000E+03
0.13440E+04 0.41566E+01 HR=  0.12195E-03 -0.64237E-04  0.13783E-03  0.89835E+00 -0.27778E+02  0.36483E-04 -0.58182E-04  0.50000E+03

** TOTAL INTEGRATION CPU TIME =  3.02      SEC.

$$ TOTAL GLOBAL CPU TIME FOR ALL SOUNDINGS =        8.67      SEC.
```


CASE.5/DIST/NEAR/M=3


CASE.5/DIST/NEAR/M=3


CASE.5/DIST/NEAR/M=3


CASE.5/DIST/NEAR/M=3

Appendix 4.-- Source code availability and listing

Source Code Availability

The current version of the source code may be obtained by writing directly to the author*, and enclosing a magnetic tape to be copied and returned. This method of releasing the source code was selected in order to satisfy requests for the latest (e.g., possibly updated) version. Unless otherwise requested, the magnetic tape will be recorded in the following mode:

> Industry compatible: 9-track, standard ANSI-labeled, ASCII-mode, odd-parity, 800-bpi density, 80-character card-image records (blocked 50-card images, or 4000-characters, per physical block), and contained on one-file named "HRZRECT.VAX".

------

* present address is:

U.S. Geological Survey
Mail Stop 964
Box 25046, Denver Federal Center
Denver, CO 80225

Source Listing

The attached subprograms are listed in the following order:

00000010	[MAIN PROGRAM]
00005490	SUBROUTINE PLTOUT
00005790	COMPLEX FUNCTION FF4
00005910	COMPLEX FUNCTION FA
00006010	COMPLEX FUNCTION FB
00006110	COMPLEX FUNCTION FC
00006210	COMPLEX FUNCTION FD
00006310	COMPLEX FUNCTION HZSPLN
00006920	SUBROUTINE RECUR1
00007230	SUBROUTINE ERRMSG
00007570	SUBROUTINE CPUTIME
00008210	SUBROUTINE HANKEL
00015460	SUBROUTINE IKS
00016100	SUBROUTINE MINMAX
00016200	SUBROUTINE NONBLANK
00016330	SUBROUTINE POLAR2
00016500	SUBROUTINE PRENAM
00017080	SUBROUTINE PROCINFO
00017450	SUBROUTINE SPLIN1
00018650	SUBROUTINE SPOINT
00018870	SUBROUTINE ZARRAY
00019250	COMPLEX FUNCTION ZSUBA1
00020130	SUBROUTINE KELVIN
00021930	SUBROUTINE ZQUAD1
00022530	ZQUAD PACKAGE

```
C <HRZRECT>: FAST EVAL OF HR,HZ FIELD COMPONENTS NEAR A RECTANGULAR    00000010
C LOOP SOURCE ON A LAYERED EARTH MODEL (SEE REF.1).        <3/1/84>    00000020
C THIS USES THE FHT (SEE REF.2) IMBEDDED IN REF.3 FOR BOTH HR & HZ,    00000030
C AND IS DOCUMENTATED IN MORE DETAIL IN REF.1.                         00000040
C                                                                      00000050
C--BY W.L.ANDERSON, U.S.GEOLOGICAL SURVEY, DENVER, COLORADO.           00000060
C                                                                      00000070
C--REFERENCES:                                                         00000080
C                                                                      00000090
C  1. ANDERSON, W.L.,1984, FAST EVALUATION OF HR AND HZ FIELD          00000100
C        SOUNDINGS NEAR A RECTANGULAR LOOP SOURCE ON A LAYERED         00000110
C        EARTH: USGS OPEN-FILE REPT. 84-257, 80 P.                     00000120
C  2. ANDERSON, W.L.,1982, FAST HANKEL TRANSFORMS USING RELATED AND    00000130
C        LAGGED CONVOLUTIONS: ACM-TOMS, V.8, N.4, P.344-368.           00000140
C  3. PODDAR, M.,1983, A RECTANGULAR LOOP SOURCE OF CURRENT ON         00000150
C        MULTILAYERED EARTH: GEOPHYSICS, V.48, N.1, P.107-109.         00000160
C  4. KAUAHIKAUA, J.,1978, ELECTROMAGNETIC FIELDS ABOUT A              00000170
C        HORIZONTAL ELECTRIC WIRE SOURCE OF ARBITRARY LENGTH:          00000180
C        GEOPHYSICS, V.43, N.5, P.1019-1022.                           00000190
C                                                                      00000200
```

```
C--FILES:
C
C  FOR001-TO-FOR004,FOR098--TEMPORARY WORK FILES DURING EXECUTION
C  FOR005--$PARMS INPUT--USING FOR007 TEMP SCRATCH VIA CALL PRENAM
C  FOR006--ONLINE PRINT
C  FOR016--COPY OF FOR006 (TO PRINT ON LINE PRINTER, ETC.)
C  FOR012--PLOT DATA IF IPLT>0 CONTAINING: AMP-VS-B (OR F)
C  FOR013--PLOT DATA IF IPLT>0 CONTAINING: PHZ-VS-B (OR F)
C  (SEE ANDERSON FOR USE OF FOR012/FOR013 ON USGS VAX-11 ONLY.)
C
C--FIELD SELECTION OPTION (VIA $PARMS ICOMP):
C     ICOMP=0 FOR HZ ONLY,
C     ICOMP=1 FOR HR ONLY,
C     ICOMP=2 (DEFAULT) FOR HZ,HR
C     ICOMP=3 FOR HZ,HR,HX,HY
C  (SEE REF.1 ABOVE FOR ALL $PARMS DEFINITIONS AND DEFAULT VALUES.)
C
      CHARACTER*2 FLD(4)
      CHARACTER*132 LINE
      CHARACTER*65 TITLE
      CHARACTER*40 TITLES(2),XAXIS
      COMPLEX AZ,BZ;CZ,DZ,HFLD,ZANS,ZWORK,HZSPLN,ZERR(4),ZSUBA1,
     1 ZER,ZERO,ZZ0,ZZANS,HX,HY
      DIMENSION SIG(10),D(9),FNF(50),BNB(50),RARG(200),ZANS(200,2),
     1 RK(10),DD(9),IJREL(2,2),ZWORK(283,2),XX(2),YY(2),ZZANS(400),
     2 XPLT(200,4),YAMP(200,4),YPHZ(200,4),NEVAL(4),NORD(2)
      EXTERNAL FF4,FA,FB,FC,FD
      COMMON/MODEL/RK,DD,M
      COMMON/HZSET/ZANS,RARG,F,DEL2,DEL,RMAX,RMIN,NR,ISET,JFLD,
     1 JFLD1,JFLD2
      COMMON/DAT/AX,BY,X,Y,XX2,YY2
      NAMELIST/PARMS/M,SIG,D,AX,BY,X,Y,DX,DY,XM,YM,EPS,EPS2,MXEVAL,
     1 RFAC,F0,NF,FM,FNF,B0,NB,BM,BNB,INFO,IPLT,ICOMP
      DATA M/1/,EPS/0.1E-9/,PIMU0/3.947841762E-6/,NORD/1,0/,
     1 ZERO/(0.,0.)/,EPS2/.1E-3/,IPLT/0/,IFIRST/1/,RFAC/5./,
     2 IJREL/3*0,1/INFO/0/,MXEVAL/500/,ICOMP/2/,
     3 FLD/'HZ','HR','HX','HY'/,TWOPI/6.283185307/
C--READ & CHECK PARMS
      CALL PRENAM(5,7)
      TOTCPU=0.0
1     READ(7,10,END=99) TITLE
10    FORMAT(A)
      WRITE(6,20) TITLE
20    FORMAT('1<HRZRECT>: ',A/)
      WRITE(16,20) TITLE
30    READ(7,PARMS,END=99)
      OPEN(UNIT=98,STATUS='SCRATCH')
        WRITE(98,PARMS)
C--REFMT WRITE(98,NAMELIST) TO UNIT=6 AND 16 TO BREAK OUT ARRAY LISTS
      REWIND 98
9910  READ(98,9920,END=9940) LINE
9920  FORMAT(A)
      I=INDEX(LINE,'$')
      IF(I.NE.0) GO TO 9930
      I=INDEX(LINE,'=')
      IF(I.NE.0) GO TO 9930
      LINE(11:)=LINE
      LINE(1:10)=' '
9930  CALL NONBLANK(LINE,I)
      IF(I.EQ.0) I=1
      WRITE(6,9920) LINE(1:I)
      WRITE(16,9920) LINE(1:I)
      GO TO 9910
9940  CLOSE(UNIT=98)
      IF(M.LT.1.OR.M.GT.10)CALL ERRMSG('M<1 OR >10',0,6,16)
      SIG1=SIG(1)
      IF(SIG1.LE.0.)CALL ERRMSG('SIG(1)<=0',0,6,16)
      M1=M-1
      IF(M.GT.1) THEN
        DO I=1,M1
          IF(D(I).LE.0.)CALL ERRMSG('SOME D(I)<=0',0,6,16)
        ENDDO
      ENDIF
      IF(AX.LE.0.)CALL ERRMSG('AX<=0',0,6,16)
      IF(BY.LE.0.)CALL ERRMSG('BY<=0',0,6,16)
      IF(NF.EQ.0.AND.NB.EQ.0)CALL ERRMSG('NF=NB=0',0,6,16)
      IF(IABS(NF).GT.50.OR.IABS(NB).GT.50)CALL ERRMSG(
     1 '|NF|>50 OR |NB|>50',0,6,16)
      IF(NB.NE.0) NF=0
      IF(NF.GT.0.AND.F0.LE.0.)CALL ERRMSG('NF>0 & F0<=0',0,6,16)
      IF(NF.GT.0.AND.(FM.LT.F0))CALL ERRMSG('NF>0 & FM<F0',0,6,16)
      IF(NB.GT.0.AND.B0.LE.0.)CALL ERRMSG('NB>0 & B0<=0',0,6,16)
      IF(NB.GT.0.AND.(BM.LT.B0))CALL ERRMSG('NB>0 & BM<B0',0,6,16)
      IF(RFAC.LT.1.0)CALL ERRMSG('RFAC<1',0,6,16)
      IF(ICOMP.LT.0.OR.ICOMP.GT.3)CALL ERRMSG('ICOMP<0 OR >3',0,6,16)
C--PARMS SEEM OK; PRESET AND GO FOR IT!
      JFLD1=1
      JFLD2=2
      IOPT=ICOMP-2
      IF(ICOMP.EQ.0) THEN
        JFLD2=1
      ELSE IF(ICOMP.EQ.1) THEN
        JFLD1=2
      ENDIF
      KFILE1=JFLD1
      KFILE2=JFLD2
      IF(ICOMP.EQ.3) KFILE2=4
      DO KFILE=KFILE1,KFILE2
        OPEN(UNIT=KFILE,STATUS='SCRATCH')
      ENDDO
      IF(DX.NE.0.0.OR.DY.NE.0.0) THEN
C--PRESET FOR GEOMETRIC SOUNDINGS (IGEOM=1 AND ONLY USE 1ST F OR B)
        IGEOM=1
        X0=X
        Y0=Y
        IF(DX.EQ.0.0) XM=X
        IF(DY.EQ.0.0) YM=Y
        IF((X.GE.-AX.AND.X.LE.AX).AND.(Y.GE.-BY.AND.Y.LE.BY)) THEN
          IF(XM.LT.-AX.OR.XM.GT.AX.OR.YM.LT.-BY.OR.YM.GT.BY)CALL
     1   ERRMSG('(X,Y) INSIDE BUT (XM,YM) OUTSIDE LOOP?',0,6,16)
```

```
        ELSE IF((XM.GE.-AX.AND.XM.LE.AX).AND.
     1        (YM.GE.-BY.AND.YM.LE.BY)) THEN
          IF(X.LT.-AX.OR.X.GT.AX.OR.Y.LT.-BY.OR.Y.GT.BY)CALL ERRMSG
     1    ('(XM,YM) INSIDE BUT (X,Y) OUTSIDE LOOP',0,6,16)
        ENDIF
        XMTEST=XM+0.5*DX
        YMTEST=YM+0.5*DY
      ELSE
        IGEOM=0
        XM=X
        YM=Y
      ENDIF
      XX(1)=X
      XX(2)=XM
      YY(1)=Y
      YY(2)=YM
      IF(IPLT.GT.0) THEN
        IF(IFIRST.EQ.1) THEN
          OPEN(UNIT=12,STATUS='NEW',FORM='FORMATTED',
     1    CARRIAGECONTROL='LIST')
          OPEN(UNIT=13,STATUS='NEW',FORM='FORMATTED',
     1    CARRIAGECONTROL='LIST')
          IFIRST=0
        ENDIF
        CALL NONBLANK(TITLE,I)
        IF(I.GT.40) THEN
          NT=4
          DO I=39,2,-1
            IF(TITLE(I:I).EQ.' ') THEN
              TITLES(1)=TITLE(1:I)
              TITLES(1)(40:40)='-'
              TITLES(2)=TITLE(I+1:)
              GO TO 2
            ENDIF
          ENDDO
        ELSE
          NT=3
          TITLES(1)=TITLE
        ENDIF
2       IF(IGEOM.EQ.1) THEN
          XAXIS='DIST.(X) M.'
        ELSE IF(NF.NE.0) THEN
          XAXIS='FREQ. (HZ.)'
        ELSE
          XAXIS='IND.NO. (B)'
        ENDIF
      ENDIF
      WRITE(6,40) TITLE,Y
40    FORMAT('1<HRZRECT>:     ',A,
     1 T118,'Y=',E13.5//T4,'F',T16,'B',T26,'FIELD',T33,'RE',
     2 T46,'IM',T59,'AMP',T72,'AMP Z/Z0',T85,'PHZ Z/Z0',T98,
     3 'ERR (RE)',T111,'ERR (IM)',T124,'X'/)
      WRITE(16,40) TITLE,Y
      DO I=1,M
        RK(I)=SIG(I)/SIG1
      ENDDO
      IF(EPS2.LT.1.E-8) EPS2=1.E-8
      IF(EPS2.GT.0.1) EPS2=0.1
      IF(NF.LT.0) THEN
        JMAX=IABS(NF)
        DO I=1,JMAX
          IF(FNF(I).LE.0.)CALL ERRMSG(
     1 'SOME FNF(I)<=0 FOR I=1,|NF|',0,6,16)
        ENDDO
        CALL MINMAX(FNF,JMAX,FMIN,FMAX)
      ELSE IF(NF.GT.0) THEN
        DF=EXP(2.30258509/FLOAT(NF))
        FMIN=F0
        FMAX=0.5*(FM+FM*DF)
      ELSE IF(NB.LT.0) THEN
        JMAX=IABS(NB)
        DO I=1,JMAX
          IF(BNB(I).LE.0.)CALL ERRMSG(
     1 'SOME BNB(I)<=0 FOR I=1,|NB|',0,6,16)
        ENDDO
        CALL MINMAX(BNB,JMAX,BMIN,BMAX)
      ELSE
        DB=EXP(2.30258509/FLOAT(NB))
        BMIN=B0
        BMAX=0.5*(BM+BM*DB)
      ENDIF
C--GET GLOBAL RMIN,RMAX
      DO I=1,2
        J=(I-1)*6
        XI=XX(I)
        YI=YY(I)
        AX1=(XI-AX)**2
        AX2=(XI+AX)**2
        BY1=(YI-BY)**2
        BY2=(YI+BY)**2
        RARG(1+J)=SQRT(AX1+BY1)
        RARG(2+J)=SQRT(AX2+BY1)
        RARG(3+J)=SQRT(AX2+BY2)
        RARG(4+J)=SQRT(AX1+BY2)
        IF(XI.GE.-AX.AND.XI.LE.AX) THEN
          RARG(5+J)=ABS(ABS(YI)-BY)
        ELSE
          RARG(5+J)=RARG(1+J)
        ENDIF
        IF(YI.GE.-BY.AND.YI.LE.BY) THEN
          RARG(6+J)=ABS(ABS(XI)-AX)
        ELSE
          RARG(6+J)=RARG(2+J)
        ENDIF
      ENDDO
      CALL MINMAX(RARG,12,RMIN,RMAX)
      IF(RMIN.EQ.0.0)CALL ERRMSG('RMIN=0--(X,Y) ON WIRE?',0,6,16)
      RMIN=RMIN/RFAC
      RMAX=RFAC*RMAX
      NR=AINT(5.*ALOG(RMAX/RMIN))+3
      IF(NR.GT.200)CALL ERRMSG(
     1 'COMPUTED NR>200 -- CHECK X,Y LOCATION OR RFAC',0,6,16)
C--BEGIN WITH CURRENT (X,Y) FIELD POINT (RETURNS TO 80 FOR DIST CURVES)
80    IF(X.EQ.AX.OR.X.EQ.-AX) THEN
        IF(Y.GE.-BY.AND.Y.LE.BY)CALL ERRMSG('(X,Y) ON WIRE?',0,6,16)
      ELSE IF(Y.EQ.BY.OR.Y.EQ.-BY) THEN
        IF(X.GE.-AX.AND.X.LE.AX)CALL ERRMSG('(X,Y) ON WIRE?',0,6,16)
      ENDIF
C--DETERMINE PRIMARY Z0 FIELD FOR THIS (X,Y)
      XX2=X*X
      YY2=Y*Y
      RHOXY=SQRT(XX2+YY2)
      IF(RHOXY.EQ.0.0) THEN
        IF(NB.NE.0)CALL ERRMSG('X=Y=0 CAN ONLY BE RUN WHEN NB=0',
     1  0,6,16)
        XRHO=1.0
        YRHO=1.0
      ELSE
        XRHO=X/RHOXY
        YRHO=Y/RHOXY
      ENDIF
      X1=X+AX
      X2=X-AX
      Y1=Y+BY
      Y2=Y-BY
      X12=X1*X1
      X22=X2*X2
      Y12=Y1*Y1
      Y22=Y2*Y2
      RARG(1)=SQRT(X12+Y22)
      RARG(2)=SQRT(X22+Y22)
      RARG(3)=SQRT(X12+Y12)
      RARG(4)=SQRT(X22+Y12)
      Z0=0.0
      IF(Y2.NE.0.0) Z0=-(X1/RARG(1)-X2/RARG(2))/Y2
      IF(Y1.NE.0.0) Z0=Z0+(X1/RARG(3)-X2/RARG(4))/Y1
      IF(X1.NE.0.0) Z0=Z0+(Y1/RARG(3)-Y2/RARG(1))/X1
      IF(X2.NE.0.0) Z0=Z0-(Y1/RARG(4)-Y2/RARG(2))/X2
      Z0=-Z0/12.566371
      IF(IGEOM.EQ.2) GO TO 90
      NPTS=0
      SUMCPU=0.0
      CALL SETTIME
C--BEGIN FREQ (F OR B) LOOP -- RETURN POINT IS LABEL 100 FOR NEXT F OR B
90    IF(NF.LT.0) THEN
        F=FNF(1)
        J=1
      ELSE IF(NF.GT.0) THEN
        F=F0
      ELSE IF(NB.LT.0) THEN
        B=BNB(1)
        J=1
        F=(B/RHOXY)**2/(PIMU0*SIG1)
      ELSE
        B=B0
        F=(B/RHOXY)**2/(PIMUO*SIG1)
      ENDIF
100   DEL2=1.0/(PIMUO*SIG1*F)
      DEL=SQRT(DEL2)
      IF(NF.NE.0) B=RHOXY/DEL
      IF(IGEOM.EQ.2) GO TO 110
C--IF M>1, CALL HANKEL FOR ALL RELATED & LAGGED TRANSFORMS
C  FOR R IN [RMIN,RMAX]
      IF(M.GT.1) THEN
        DO I=1,M1
          DD(I)=2.*D(I)/DEL
        ENDDO
       IF(ICOMP.LT.2) THEN
        CALL HANKEL(RMAX,NR,1,EPS,1,NORD(JFLD1),FF4,IJREL,ZWORK,
     1  ZANS(1,JFLD1),RARG,NOFUN,IERR)
        IF(IERR.NE.0)CALL ERRMSG('IERR NOT 0 AFTER CALL HANKEL?',
     2 0,6,16)
       ELSE
        CALL HANKEL(RMAX,NR,2,EPS,1,NORD,FF4,IJREL,ZWORK,
     1  ZZANS,RARG,NOFUN,IERR)
        IF(IERR.NE.0)CALL ERRMSG('IERR NOT 0 AFTER CALL HANKEL?',
     2 0,6,16)
        CALL ZARRAY(1,NR,2,200,2,ZZANS,ZANS)
       ENDIF
      ELSE
C--SPECIAL CASE M=1 -- USE DIRECT QUADRATURE SINCE NO HANKEL TRANSFORMS
        DO I=1,NR
          RARG(NR+1-I)=EXP(-.2*(I-1))*RMAX
        ENDDO
        DO JFLD=JFLD1,JFLD2
          DO I=1,NR
            ZANS(I,JFLD)=ZERO
          ENDDO
        ENDDO
      ENDIF
C--SETUP HZSPLN(R) FOR ALL SUBSEQUENT SPLINE INTEPOLATIONS, ETC.
      ISET=1
      HFLD=HZSPLN(0.0)
C--GET THE 4-DEFINITE INTEGRALS BY FAST ADAPTIVE SPLINE QUADRATURE
C  FOR EACH JFLD=JFLD1,JFLD2
C
110   JOPT=0
      DO 250 JJ=JFLD1,JFLD2
      JFLD=JJ
      IF(BY.EQ.Y.AND.JJ.EQ.1) THEN
        AZ=ZERO
        NEVAL(1)=0
        ZERR(1)=ZERO
      ELSE
        AZ=-ZSUBA1(-AX,AX,EPS2,NEVAL(1),ICK,ZERR(1),FA,MXEVAL)
        IF(ICK.LT.0)
     1 CALL ERRMSG('ICK<0 IN AZ AFTER ZSUBA1',0,6,16)
        IF(NEVAL(1).GT.MXEVAL)CALL ERRMSG(
     1 'NEVAL(1)>MXEVAL IN AZ AFTER ZSUBA1',0,6,16)
        IF(JJ.EQ.1) AZ=(BY-Y)*AZ
      ENDIF
      IF(BY.EQ.-Y.AND.JJ.EQ.1) THEN
        CZ=ZERO
        NEVAL(3)=0
        ZERR(3)=ZERO
      ELSE IF(Y.EQ.0.0) THEN
        CZ=AZ
        NEVAL(3)=0
        ZERR(3)=ZERR(1)
      ELSE
        CZ=-ZSUBA1(-AX,AX,EPS2,NEVAL(3),ICK,ZERR(3),FC,MXEVAL)
        IF(ICK.LT.0)
     1 CALL ERRMSG('ICK<0 IN CZ AFTER ZSUBA1',0,6,16)
        IF(NEVAL(3).GT.MXEVAL)CALL ERRMSG(
     1 'NEVAL(3)>MXEVAL IN CZ AFTER ZSUBA1',0,6,16)
        IF(JJ.EQ.1) CZ=(BY+Y)*CZ
      ENDIF
      IF(AX.EQ.X.AND.JJ.EQ.1) THEN
        BZ=ZERO
        NEVAL(2)=0
        ZERR(2)=ZERO
      ELSE
        BZ=-ZSUBA1(-BY,BY,EPS2,NEVAL(2),ICK,ZERR(2),FB,MXEVAL)
        IF(ICK.LT.0)
     1 CALL ERRMSG('ICK<0 IN BZ AFTER ZSUBA1',0,6,16)
        IF(NEVAL(2).GT.MXEVAL)CALL ERRMSG(
     1 'NEVAL(2)>MXEVAL IN BZ AFTER ZSUBA1',0,6,16)
        IF(JJ.EQ.1) BZ=(AX-X)*BZ
      ENDIF
      IF(AX.EQ.-X.AND.JJ.EQ.1) THEN
        DZ=ZERO
        NEVAL(4)=0
        ZERR(4)=ZERO
      ELSE IF(X.EQ.0.0) THEN
        DZ=BZ
        NEVAL(4)=0
        ZERR(4)=ZERR(2)
      ELSE
        DZ=-ZSUBA1(-BY,BY,EPS2,NEVAL(4),ICK,ZERR(4),FD,MXEVAL)
        IF(ICK.LT.0)
     1 CALL ERRMSG('ICK<0 IN DZ AFTER ZSUBA1',0,6,16)
        IF(NEVAL(4).GT.MXEVAL)CALL ERRMSG(
     1 'NEVAL(4)>MXEVAL IN DZ AFTER ZSUBA1',0,6,16)
        IF(JJ.EQ.1) DZ=(AX+X)*DZ
      ENDIF
C--SUM TO GET THE FINAL HFLD FOR THIS JFLD
      IF(JFLD.EQ.1) THEN
        HFLD=(AZ+BZ+CZ+DZ)/TWOPI
      ELSE
        HX=(-BZ+DZ)/TWOPI
        HY=(-AZ+CZ)/TWOPI
        HFLD=XRHO*HX+YRHO*HY
      ENDIF
C--ACCUMULATE CPU SECONDS FOR ALL INTEGRATIONS
249   CALL GETTIME(CPU)
```

```
      SUMCPU=SUMCPU+CPU
      ZER=ZERO
      DO I=1,4
        IF(CABS(ZERR(I)).GT.CABS(ZER)) ZER=ZERR(I)
      ENDDO
199   ZZ0=HFLD/Z0
      CALL POLAR2(ZZ0,AMPZZ0,PHZ180)
      AMP=CABS(HFLD)
C--WRITE TEMP FILE JFLD (=1,2[,3,4]) FOR LATER REWRITES ON 6 AND 16
      WRITE(JFLD,200) F,B,FLD(JFLD),HFLD,AMP,AMPZZ0,PHZ180,ZER,X
200   FORMAT(1X,E11.5,E12.5,1X,A2,'=',8E13.5)
C--PREPARE PLOT DATA IF IPLT>0
      IF(IPLT.GT.0) THEN
        IF(JFLD.EQ.1) NPTS=NPTS+1
        IF(NPTS.GT.200)CALL ERRMSG('NPTS>200 WHEN IPLT>0',0,6,16)
        IF(IGEOM.GT.0) THEN
          XPLT(NPTS,JFLD)=X
        ELSE IF(NF.NE.0) THEN
          XPLT(NPTS,JFLD)=F
        ELSE
          XPLT(NPTS,JFLD)=B
        ENDIF
        IF(IPLT.EQ.1) THEN
          YAMP(NPTS,JFLD)=AMPZZ0
        ELSE
          YAMP(NPTS,JFLD)=AMP
        ENDIF
        YPHZ(NPTS,JFLD)=PHZ180
      ENDIF
      IF(JJ.EQ.2.AND.IOPT.GT.0) THEN
        IF(JOPT.EQ.0) THEN
          HFLD=HX
          JFLD=3
          JOPT=1
          GO TO 199
        ELSE IF(JOPT.EQ.1) THEN
          HFLD=HY
          JFLD=4
          JOPT=2
          GO TO 199
        ENDIF
      ENDIF
      CALL SETTIME
250   CONTINUE        !!!! GET THE NEXT JFLD, ETC. !!!!
C
C--END OF DIST (X) LOOP (IF IGEOM>0)?
      IF(IGEOM.GT.0) THEN
        X=X+DX
        IF(X.GE.XMTEST) GO TO 300
        IGEOM=2
C--RETURN TO 80 FOR NEXT X
        CALL SETTIME
        GO TO 80
      ENDIF
C--END OF FREQ (F OR B) LOOP?
```

```
      IF(NF.LT.0) THEN
        J=J+1
        IF(J.GT.JMAX) GO TO 300
        F=FNF(J)
      ELSE IF(NF.GT.0) THEN
        F=F*DF
        IF(F.GT.FMAX) GO TO 300
      ELSE IF(NB.LT.0) THEN
        J=J+1
        IF(J.GT.JMAX) GO TO 300
        B=BNB(J)
        F=(B/RHOXY)**2/(PIMU0*SIG1)
      ELSE
        B=B*DB
        IF(B.GT.BMAX) GO TO 300
        F=(B/RHOXY)**2/(PIMU0*SIG1)
      ENDIF
C--RETURN TO 100 FOR NEXT F
      CALL SETTIME
      GO TO 100
C--OUTPUT ALL PLUS CPU TIME FOR ALL INTEGRATIONS FOR LAST SOUNDING
300   DO KFILE=KFILE1,KFILE2
        REWIND KFILE
303     READ(KFILE,304,END=305) LINE
304     FORMAT(A)
        WRITE(6,304) LINE
        WRITE(16,304) LINE
        GO TO 303
305     WRITE(6,306)
306     FORMAT(/)
        WRITE(16,306)
        REWIND KFILE
      ENDDO
      WRITE(6,992) SUMCPU
992   FORMAT(' ** TOTAL INTEGRATION CPU TIME =',G10.3,' SEC.'/)
      WRITE(16,992) SUMCPU
      TOTCPU=TOTCPU+SUMCPU
      IF(INFO.GT.0) THEN
        WRITE(6,990) RMIN,RMAX,NR,NOFUN,NEVAL
990     FORMAT(
     1 '    RMIN=',E16.8,'  RMAX=',E16.8,'  NR=',I5/
     2 '    LAST: NOFUN=',I5,'  NEVAL(1:4)=',4I5)
        WRITE(16,990) RMIN,RMAX,NR,NOFUN,NEVAL
      ENDIF
C--CONTINUE DIST (Y) LOOP (IF IGEOM>0)?
400   IF(IGEOM.GT.0) THEN
        X=X0
        Y=Y+DY
        IF(Y.GE.YMTEST) THEN
          Y=Y0
C--OUTPUT PLOT DATA ON FOR012 & FOR013 IF IPLT>0
          IF(IPLT.GT.0) THEN
            DO JFLD=KFILE1,KFILE2
              CALL PLTOUT(NT,XAXIS,TITLES,XPLT,YAMP,YPHZ,FLD,JFLD,
     1        NPTS,IPLT)
```

```
            ENDDO
          ENDIF
          GO TO 1
        ENDIF
        WRITE(6,40) TITLE,Y
        WRITE(16,40) TITLE,Y
        IGEOM=2
        NPTS=0
        SUMCPU=0.0
        CALL SETTIME
        GO TO 80
      ELSE
C--OUTPUT PLOT DATA ON FOR012 & FOR013 IF IPLT>0
          IF(IPLT.GT.0) THEN
            DO JFLD=KFILE1,KFILE2
              CALL PLTOUT(NT,XAXIS,TITLES,XPLT,YAMP,YPHZ,FLD,JFLD,
     1       NPTS,IPLT)
            ENDDO
          ENDIF
        GO TO 1
      ENDIF
99    IF(IPLT.GT.0) THEN
        WRITE(12,991)
991     FORMAT('0'/'001'/' ')
        WRITE(13,991)
      ENDIF
C--OUTPUT GLOBAL TIMES FOR ALL SOUNDINGS
      WRITE(6,993) TOTCPU
993   FORMAT('0$$ TOTAL GLOBAL CPU TIME FOR ALL SOUNDINGS =',
     1 G15.3,' SEC.')
      WRITE(16,993) TOTCPU
      CALL EXIT
      END
      SUBROUTINE PLTOUT(NT,XAXIS,TITLES,XPLT,YAMP,YPHZ,FLD,JFLD,NPTS,
     1 IPLT)
C--CALLED ONLY IF IPLT>0
      CHARACTER*40 TITLES(2),XAXIS
      CHARACTER*3 NORM
      CHARACTER*2 FLD(4)
      DIMENSION XPLT(200,4),YAMP(200,4),YPHZ(200,4)
      IF(IPLT.EQ.1) THEN
        NORM='/ZO'
      ELSE
        NORM=' '
      ENDIF
      IF(NT.EQ.4) THEN
        WRITE(12,310) NT,XAXIS,'AMPLITUDE '//FLD(JFLD)//NORM,
     1  (TITLES(I),I=1,NT-2)
310     FORMAT(I1/A/A/A/A)
        WRITE(13,310) NT,XAXIS,'PHASE (DEG.) '//FLD(JFLD)//NORM,
     1  (TITLES(I),I=1,NT-2)
      ELSE
        WRITE(12,315) NT,XAXIS,'AMPLITUDE '//FLD(JFLD)//NORM,
     1 TITLES(1)
315     FORMAT(I1/A/A/A)
```

```
        WRITE(13,315) NT,XAXIS,'PHASE (DEG.) '//FLD(JFLD)//NORM,
     1 TITLES(1)
      ENDIF
      WRITE(12,320) NPTS,(XPLT(I,JFLD),YAMP(I,JFLD),I=1,NPTS)
320   FORMAT(I/(2G16.8))
      WRITE(13,320) NPTS,(XPLT(I,JFLD),YPHZ(I,JFLD),I=1,NPTS)
      RETURN
      END
      COMPLEX FUNCTION FF4(X)
C--KERNEL USED IN HZRECT IN LAMBDA(X)-DEL FORM (I.E., G=X*DEL)
      COMPLEX V1,F1,C,ONE,ZANS
      COMMON/HZSET/ZANS(200,2),RARG(200),F,DEL2,DEL,RMAX,RMIN,NR,
     1 ISET,JFLD,JFLD1,JFLD2
      DATA ONE/(1.0,0.0)/
      G=X*DEL
      CALL RECUR1(G,V1,F1)
      C=G
      FF4=C*(V1*C*(ONE-F1))/((C+V1*F1)*(C+V1))
      RETURN
      END
      COMPLEX FUNCTION FA(X)
C--INTEGRAND FOR AZ
      COMPLEX HZSPLN,ZANS(200,2)
      COMMON/HZSET/ZANS,RARG,F,DEL2,DEL,RMAX,RMIN,NR,ISET,JFLD,
     1 JFLD1,JFLD2
      COMMON/DAT/AX,BY,XX,YY,XX2,YY2
      R=SQRT((X-XX)**2+(BY-YY)**2)
      FA=HZSPLN(R)/R
      RETURN
      END
      COMPLEX FUNCTION FB(Y)
C--INTEGRAND FOR BZ
      COMPLEX HZSPLN,ZANS(200,2)
      COMMON/HZSET/ZANS,RARG,F,DEL2,DEL,RMAX,RMIN,NR,ISET,JFLD,
     1 JFLD1,JFLD2
      COMMON/DAT/AX,BY,XX,YY,XX2,YY2
      R=SQRT((AX-XX)**2+(Y-YY)**2)
      FB=HZSPLN(R)/R
      RETURN
      END
      COMPLEX FUNCTION FC(X)
C--INTEGRAND FOR CZ
      COMPLEX HZSPLN,ZANS(200,2)
      COMMON/HZSET/ZANS,RARG,F,DEL2,DEL,RMAX,RMIN,NR,ISET,JFLD,
     1 JFLD1,JFLD2
      COMMON/DAT/AX,BY,XX,YY,XX2,YY2
      R=SQRT((X-XX)**2+(BY+YY)**2)
      FC=HZSPLN(R)/R
      RETURN
      END
      COMPLEX FUNCTION FD(Y)
C--INTEGRAND FOR DZ
      COMPLEX HZSPLN,ZANS(200,2)
      COMMON/HZSET/ZANS,RARG,F,DEL2,DEL,RMAX,RMIN,NR,ISET,JFLD,
     1 JFLD1,JFLD2
```

```
      COMMON/DAT/AX,BY,XX,YY,XX2,YY2
      R=SQRT((AX+XX)**2+(Y-YY)**2)
      FD=HZSPLN(R)/R
      RETURN
      END
      COMPLEX FUNCTION HZSPLN(R)
C--SPLINE-DEFINED FUNCTIONS VIA JFLD FOR ANY R IN [RMIN,RMAX], WHERE
C  COMMON/HZSET/ CONTAINS ALL PRECOMPUTED VALUES VIA HANKEL, ETC.
C
      SAVE
      COMPLEX ZANS(200,2),CB,N33,N3,I2,Z, I1K1,IKDIF
      REAL RARG(200),YR(200,2),YI(200,2),AR(200,2),BR(200,2),
     1 CR(200,2),D(2),
     2 AI(200,2),BI(200,2),CI(200,2),W1(200),W2(200),X(200)
      COMMON/HZSET/ZANS,RARG,F,DEL2,DEL,RMAX,RMIN,NR,ISET,JFLD,
     1 JFLD1,JFLD2
      COMMON/DAT/AX,BY,XX,YY,XX2,YY2
      DATA N33/(3.,3.)/,N3/(3.,0.)/,I2/(0.,2.)/,D/2*0.0/
      IF(ISET.EQ.1) THEN
        DO I=1,NR
          RI=RARG(I)
          X(I)=ALOG(RI)
          B=RI/DEL
          R2=RI*RI
          CB=B
         IF(JFLD1.EQ.1) THEN
          Z=ZANS(I,1)/DEL -CMPLX(0.,0.5*DEL2/RI**4)*
     1    (N3-(N3+CB*(N33+CB*I2))*CEXP(CMPLX(-B,-B)))
          YR(I,1)=REAL(Z)
          YI(I,1)=AIMAG(Z)
         ENDIF
         IF(JFLD2.EQ.2) THEN
          CALL IKS(.7071067811865475D0*DBLE(B),I1K1,IKDIF)
          Z=-CB*ZANS(I,2)-(2.*I1K1-IKDIF)/RI
          YR(I,2)=REAL(Z)
          YI(I,2)=AIMAG(Z)
         ENDIF
        ENDDO
       DO J=JFLD1,JFLD2
        CALL SPLIN1(NR,0.0,X,YR(1,J),AR(1,J),BR(1,J),CR(1,J),0,D,W1,W2)
        IF(NR.LT.0)CALL ERRMSG('NR<0 AFTER SPLIN1 IN HZSPLN?',0,6,16)
        CALL SPLIN1(NR,0.0,X,YI(1,J),AI(1,J),BI(1,J),CI(1,J),0,D,W1,W2)
        IF(NR.LT.0)CALL ERRMSG('NR<0 AFTER SPLIN1 IN HZSPLN?',0,6,16)
       ENDDO
        ISET=0
        HZSPLN=(0.,0.)
      ELSE
        IF(R.LT.RMIN.OR.R.GT.RMAX)CALL ERRMSG(
     1 'R<RMIN OR >RMAX IN HZSPLN?',0,6,16)
        RLOG=ALOG(R)
        IF(RLOG.LT.X(1)) THEN
          ANSR=YR(1,JFLD)
          ANSI=YI(1,JFLD)
        ELSE IF(RLOG.GT.X(NR)) THEN
          ANSR=YR(NR,JFLD)
          ANSI=YI(NR,JFLD)
        ELSE
          CALL SPOINT(NR,X,YR(1,JFLD),AR(1,JFLD),BR(1,JFLD),
     1    CR(1,JFLD),RLOG,ANSR)
          CALL SPOINT(NR,X,YI(1,JFLD),AI(1,JFLD),BI(1,JFLD),
     1    CI(1,JFLD),RLOG,ANSI)
        ENDIF
        HZSPLN=CMPLX(ANSR,ANSI)
      ENDIF
      RETURN
      END
      SUBROUTINE RECUR1(G,V1,F1)
C--BACKWARD RECURRENCE FOR COMPLEX V1,F1 GIVEN REAL*4 ARGUMENT G AND:
COMMON/MODEL/ PARAMETERS:
C     K(10) = NORMALIZED CONDUCTIVITY ARRAY (M VALUES,WHERE K(1)=1.0).
C     D(9)  = LAYER THICKNESS ARRAY (M-1 VALUES) D=2*THICKNESS/DEL.
C     M     = NUMBER LAYERS (M.GE.1.AND.M.LE.10)
C             SPECIAL CASE WHEN M=1 (HOMOGENEOUS--D IGNORED)
C
C--NOTE: G,K,D ARE REAL*4
C
C
      COMMON/MODEL/K,D,M
      REAL*4 K(10),D(9)
      COMPLEX C,VM,V1,F1,EVD,ONE
      DATA ONE/(1.0,0.0)/
      F1=ONE
      G2=G*G
      VM=CSQRT(CMPLX(G2,2.0*K(M)))
      IF(M.EQ.1) GO TO 2
      J=M-1
    1 V1=CSQRT(CMPLX(G2,2.0*K(J)))
      EVD=CEXP(-V1*D(J))
      C=(ONE-EVD)/(ONE+EVD)
      F1=(VM*F1+V1*C)/(V1+VM*F1*C)
      IF(J.EQ.1) GO TO 3
      J=J-1
      VM=V1
      GO TO 1
    2 V1=VM
    3 RETURN
      END
      SUBROUTINE ERRMSG(MSG,ISKIP,IUNIT1,IUNIT2)
C
C  GENERAL ERROR MESSAGE OUTPUT AND EXIT ON VAX-11/780
C
C  MSG*(*) = VARIABLE-LENGTH 'MESSAGE'
C  ISKIP = 0 FOR NO BLANK LINE BEFORE OUTPUT TO IUNIT1 & IUNIT2
C          > 0 FOR ONE BLANK LINE BEFORE.
C  IUNIT1 = 0 TO SUPPRESS OUTPUT ON IUNIT1 (>0 TO WRITE ON IUNIT1).
C  IUNIT2 = 0 TO SUPPRESS OUTPUT ON IUNIT2 (>0 TO WRITE ON IUNIT2).
C
C  MESSAGES ARE WRITTEN IN THE FORM:
C
C  {ERRMSG}: _MSG_HERE_
```

```
C
      CHARACTER*(*) MSG
      I=LEN(MSG)
      DO 1 J=1,2
        IF(J.EQ.1) THEN
       JUNIT=IUNIT1
          ELSE
       JUNIT=IUNIT2
        ENDIF
        IF(JUNIT.GT.0) THEN
          IF(ISKIP.EQ.0) THEN
            WRITE(JUNIT,2) MSG
            ELSE
       WRITE(JUNIT,3) MSG
          ENDIF
      ENDIF
1     CONTINUE
      CALL EXIT
2     FORMAT(1X,'{ERRMSG}: ',A<I>)
3     FORMAT(/1X,'{ERRMSG}: ',A<I>)
      END
      SUBROUTINE CPUTIME(I1,I2)
C
C  CPUTIME WRITES "ELAPSED & CPU" TIME FROM PREVIOUS "CALL SETTIME" ON
C  FORTRAN UNITS I1 (IF NOT 0) AND I2 (IF NOT 0).
C
C  WILL EJECT FIRST IF I1>0 (OR I2>0).
C  DOUBLE SPACE FIRST IF I1<0 (OR I2<0).
C
C  E.G., USE TO TIME ELAPSED & CPU TIME FOR PROGRAM OR CODE SEGMENTS AS:
C
C     CALL SETTIME    ! DON'T FORGET TO DO THIS!
C     >>>>> THE CODE TO TIME IS HERE <<<<<  ! USUALLY A COMPLETE PROGRAM
C     CALL CPUTIME(-6,16) ! OR USE I1 OR I2=0 TO OMIT WRITE.
C     >>>>> ALSO CAN USE CALL GETTIME(CPU) TO GET JUST THE CPU (SEC)
C           SINCE THE LAST CALL SETTIME WAS DONE.
C
      SAVE
      INTEGER*4 ABSVAL(4),INCRVAL(4)
      CALL PROCINFO(ABSVAL,INCRVAL)
      TIMES=SECNDS(TIME0)
      MIN=TIMES/60.0
      SEC=AMOD(TIMES,60.0)
      CPUSEC=INCRVAL(1)*.01
      IMIN=CPUSEC/60.0
      CSEC=AMOD(CPUSEC,60.0)
      PCPU=100.*(CPUSEC/TIMES)
      IF(I1.NE.0) THEN
        IF(I1.GT.0) THEN
          J=1
        ELSE
          J=0
        ENDIF
      WRITE(IABS(I1),60) J,TIMES,MIN,SEC,CPUSEC,IMIN,CSEC,PCPU,
     1 (INCRVAL(I),I=2,4)
```

```
60    FORMAT(I1,65('$')/' TOTAL "ELAPSED" TIME=',F16.2,' SEC. (',       00007910
     1 I4,' MIN.',F6.2,' SEC.)'/                                        00007920
     2 ' CPU_TIME=',F15.2,' SEC. (',I4,' M. ',F5.2,                     00007930
     1 ' S.)   CPU % =',F6.2,'%'/                                       00007940
     3 ' BUF.I/O_COUNT=',I10/                                           00007950
     4 ' DIR.I/O_COUNT=',I10/                                           00007960
     5 ' PAGE_FAULTS=',2X,I10/                                          00007970
     6 ' ',65('$')//)                                                   00007980
      ENDIF                                                             00007990
      IF(I2.NE.0) THEN                                                  00008000
        IF(I2.GT.0) THEN                                                00008010
          J=1                                                           00008020
        ELSE                                                            00008030
          J=0                                                           00008040
        ENDIF                                                           00008050
      WRITE(IABS(I2),60) J,TIMES,MIN,SEC,CPUSEC,IMIN,CSEC,PCPU,         00008060
     1 (INCRVAL(I),I=2,4)                                               00008070
      ENDIF                                                             00008080
      RETURN                                                            00008090
C** ENTRY 'CALL SETTIME'--MUST BE DONE BEFORE 'CALL CPUTIME(I1,I2)'     00008100
      ENTRY SETTIME()                                                   00008110
      TIME0=SECNDS(0.0)                                                 00008120
      CALL PROCINFO(ABSVAL,INCRVAL)                                     00008130
      RETURN                                                            00008140
C** ENTRY 'CALL GETTIME(CPU)'--TO GET CPU(SEC) SINCE LAST CALL SETTIME  00008150
      ENTRY GETTIME(CPU)                                                00008160
      CALL PROCINFO(ABSVAL,INCRVAL)                                     00008170
      CPU=INCRVAL(1)*.01                                                00008180
      RETURN                                                            00008190
      END                                                               00008200
      SUBROUTINE HANKEL(BMAX,NB,NREL,TOL,NTOL,NORD,FUN1,IJREL,ZWORK,    00008210
     * ZANS,ARG,NOFUN1,IERR)                                            00008220
C=======================================================================00008230
      INTEGER NB,NREL,NTOL,NORD(NREL),IJREL(2,NREL),NOFUN1,IERR         00008240
      REAL BMAX,TOL,ARG(NB)                                             00008250
      COMPLEX ZWORK(283,NREL),ZANS(NB,NREL)                             00008260
C=======================================================================00008270
C                                                                       00008280
C  PURPOSE                                                              00008290
C                                                                       00008300
C     THE PURPOSE OF SUBPROGRAM HANKEL IS TO PROVIDE IN SINGLE PRECISION00008310
C    A GENERAL ALGORITHM FOR FAST COMPLEX HANKEL TRANSFORMS OF ORDERS   00008320
C    0 AND 1 USING RELATED AND LAGGED CONVOLUTIONS.                     00008330
C                                                                       00008340
C  AUTHOR                                                               00008350
C                                                                       00008360
C    ANDERSON, W.L., U.S. GEOLOGICAL SURVEY, DENVER, COLORADO.          00008370
C                                                                       00008380
C  REFERENCES (REF.3 DESCRIBES THE HANKEL ALGORITHM IN DETAIL.)         00008390
C                                                                       00008400
C    1. ANDERSON, W.L., IMPROVED DIGITAL FILTERS FOR EVALUATING         00008410
C       FOURIER AND HANKEL TRANSFORM INTEGRALS. N.T.I.S REPT.           00008420
C       PB-242-800, SPRINGFIELD, VA., 1975.                             00008430
C                                                                       00008440
C    2. ANDERSON, W.L., NUMERICAL INTEGRATION OF RELATED HANKEL         00008450
```

```
C       TRANSFORMS OF ORDERS 0 AND 1 BY ADAPTIVE DIGITAL FILTERING.     00008460
C       GEOPHYSICS 44 (JULY 1979), 1287-1305.                            00008470
C                                                                        00008480
C    3. ANDERSON, W.L., FAST HANKEL TRANSFORMS USING RELATED AND        00008490
C       LAGGED CONVOLUTIONS, ACM TRANS. ON MATH. SOFTWARE,              00008500
C       VOL.8, NO.4 (DEC. 1982), P.344-368.                              00008510
C                                                                        00008520
C  LANGUAGE                                                              00008530
C                                                                        00008540
C    ANS-FORTRAN (X3.9-1966) IS USED, WITH THE EXCEPTION OF THE          00008550
C    CHARACTERS <,[,&,:,],> APPEARING IN SOME COMMENT STATEMENTS.       00008560
C                                                                        00008570
C  ABSTRACT                                                              00008580
C                                                                        00008590
C      BY COMBINING BOTH ADAPTIVE LAGGED CONVOLUTION (SEE [1]) AND      00008600
C    ADAPTIVE RELATED CONVOLUTION (SEE [2]), SUBPROGRAM HANKEL          00008610
C    MINIMIZES EXTERNAL FUN1 CALLS (NOFUN1 AT EXIT) IN EVALUATING A     00008620
C    TOTAL OF NB*NREL COMPLEX HANKEL TRANSFORMS OF ORDERS 0 AND (OR) 1, 00008630
C    WHERE NB IS THE NUMBER OF LAGGED CONVOLUTIONS, AND NREL IS THE     00008640
C    NUMBER OF RELATED CONVOLUTIONS.                                     00008650
C      DIRECT CONVOLUTION METHODS (SEE [1],[2]) DO NOT REQUIRE BESSEL   00008660
C    FUNCTION EVALUATIONS, AND HENCE ARE GENERALLY AN ORDER OF          00008670
C    MAGNITUDE FASTER TO COMPUTE THAN MOST DIRECT NUMERICAL             00008680
C    INTEGRATION METHODS.  BY USING PREVIOUSLY SAVED TRANSFORM INPUT    00008690
C    FUNCTION EVALUATIONS, BOTH LAGGED CONVOLUTION AND RELATED          00008700
C    CONVOLUTION FURTHER REDUCE SIGNIFICANTLY THE NUMBER OF TRANSFORM   00008710
C    INPUT FUNCTION EVALUATIONS REQUIRED OVER DIRECT CONVOLUTION.       00008720
C      LAGGED CONVOLUTION IS SELECTED WHEN NB>1, WHICH DEFINES ARG(NB)  00008730
C    OVER ANY DESIRED TRANSFORM ARGUMENT RANGE (BMIN,BMAX). RESULTS ARE 00008740
C    STORED AT THE FILTER SPACING IN ARRAYS ARG(NB) AND ZANS(NB,NREL)   00008750
C    FOR LATER USE IN SPLINE INTERPOLATION, ETC.  GIVEN BMAX,NB, THE    00008760
C    VALUE OF BMIN (NOT GIVEN) CAN BE COMPUTED FROM THE EXPRESSION      00008770
C    BMIN=BMAX*EXP(-.2*(NB-1)), WHICH MUST BE .GT. 0.0 FOR THE          00008780
C    GIVEN MACHINE EXPONENT RANGE.                                       00008790
C      RELATED CONVOLUTION IS SELECTED WHEN NREL>1, AND BY GIVEN        00008800
C    SIMPLE ALGEBRAIC RELATIONSHIPS BETWEEN FUN1 AND EACH NREL RELATED  00008810
C    TRANSFORM INPUT FUNCTION, DEFINED AS G**I * FUN1(G)**J, WHERE      00008820
C    FUN1(G) IS THE FIRST TRANSFORM INPUT FUNCTION, AND ARRAY           00008830
C    IJREL(2,NREL)= PAIRS OF I,J INTEGERS (NEGATIVE, 0, OR POSITIVE).   00008840
C    THE ORDER OF ALL RELATED CONVOLUTIONS MUST BE GIVEN IN NORD(NREL), 00008850
C    AND MUST BE EITHER 0 OR 1, BUT CAN BE IN ANY DESIRED SEQUENCE.     00008860
C    HIGHER INTEGER ORDERS MAY BE EXPRESSED IN TERMS OF ORDERS 0 AND 1  00008870
C    BY USING THE RECURRENCE RELATION JN-1(X)+JN+1(X)=2*N*JN(X)/X.      00008880
C      THE EQUALLY-SPACED J0,J1 FILTER ABSCISSAS ARE GENERATED IN       00008890
C    DOUBLE-PRECISION (TO CONSERVE STORAGE AND REDUCE ROUND-OFF),       00008900
C    BUT ARE USED IN SINGLE-PRECISION IN THE COMPLEX FUNCTION FUN1.     00008910
C    BOTH J0 AND J1 FILTER RESPONSE FUNCTIONS (WEIGHTS) WERE            00008920
C    DESIGNED TO HAVE IDENTICAL ABSCISSA VALUES IN [2].  THE            00008930
C    STORED J0,J1 FILTER WEIGHTS USED IN SUBPROGRAM HANKEL ARE TAKEN    00008940
C    FROM [2].  MUCH OF THE LOGIC USED IN HANKEL FOLLOWS THE            00008950
C    CODING USED IN [1] AND [2].  THE MAJOR DIFFERENCES ARE IN THE      00008960
C    DEFINITION OF RELATED INPUT FUNCTIONS (SEE IJREL,ZWORK), AND       00008970
C    FOR HANDLING OSCILLATING FUNCTIONS (SEE NTOL,ITOL).                 00008980
C                                                                        00008990
C      FOUR GENERAL CASES ARE POSSIBLE USING SUBPROGRAM HANKEL.          00009000
```

```
C                                                                       00009010
C    CASE 1.  SINGLE DIRECT CONVOLUTION AT B=BMAX=BMIN (NB=1,NREL=1).   00009020
C    CASE 2.  RELATED CONVOLUTIONS AT A CONSTANT B=BMAX (NB=1,NREL>1).  00009030
C    CASE 3.  LAGGED CONVOLUTIONS IN (BMIN,BMAX) (NB>1,NREL=1).         00009040
C    CASE 4.  BOTH RELATED AND LAGGED CONVOLUTIONS (NB>1,NREL>1).       00009050
C                                                                       00009060
C  MACHINE DEPENDENT REMARKS                                            00009070
C                                                                       00009080
C      THIS SUBPROGRAM WAS IMPLEMENTED AND TESTED ON A 32-BIT WORD      00009090
C    MACHINE WITH EXP-RANGE APPROXIMATELY 10**-38 TO 10**+38 AND 24-BIT 00009100
C    MANTISSA (ABOUT 7-DECIMAL DIGITS).  ONLY SINGLE-PRECISION          00009110
C    COMPLEX ARITHMETIC IS USED.  DOUBLE-PRECISION REAL WORDS HAVE THE  00009120
C    SAME EXP-RANGE, WITH A 56-BIT MANTISSA (ABOUT 16-DECIMAL DIGITS).  00009130
C    FOR MACHINES WITH OTHER WORD SIZES, CHANGES IN THE NUMBER OF       00009140
C    DIGITS RETAINED IN SOME DATA STATEMENTS MAY BE REQUIRED.           00009150
C                                                                       00009160
C  DESCRIPTION OF PARAMETERS                                            00009170
C                                                                       00009180
C    INPUT                                                              00009190
C                                                                       00009200
C      BMAX - INITIAL HANKEL TRANSFORM ARGUMENT B=BMAX>0.0 (ANY CASE),  00009210
C             USED IN INTEGRAL FROM 0 TO INFINITY OF                    00009220
C             FUN1(G)*JN(G*B)*DG, WHERE JN=BESSEL FUNCTION OF ORDER N,  00009230
C             N=0 OR 1, AND B>0.0. (SEE FUN1 DEFINITION BELOW).         00009240
C        NB - NUMBER OF LAGGED CONVOLUTIONS DESIRED (NB.GE.1).  USE     00009250
C             NB=1 IF B=BMIN=BMAX (I.E., CASE 1 OR 2).  USE             00009260
C             NB>1 IF B IS LAGGED IN (BMIN,BMAX), WHERE                 00009270
C             BMIN=BMAX*EXP(-.2*(NB-1)) DOES NOT UNDERFLOW THE EXPONENT 00009280
C             RANGE.  THE B-LAGGED SPACING IS .2 IN LOG-SPACE. FOR      00009290
C             CONVENIENCE IN SPLINE INTERPOLATION LATER, EACH B IN      00009300
C             (BMIN,BMAX) IS RETURNED IN ARRAY ARG(I),I=1,NB, WHERE     00009310
C             ARG(I+1)/ARG(I)=EXP(.2) FOR ALL I.  IF BMAX>BMIN>0 IS     00009320
C             GIVEN, THEN AN EFFECTIVE VALUE OF NB IS DETERMINED AS     00009330
C             NB=AINT(5.*ALOG(BMAX/BMIN))+I, WHERE I>1 IS RECOMMENDED,  00009340
C             PARTICULARLY IF USING SUBSEQUENT SPLINE INTERPOLATION FOR 00009350
C             A DIFFERENT B-SPACING THAN USED IN THE SAMPLED FILTERS. IF00009360
C             SPLINE INTERPOLATION IS TO BE USED LATER, IT IS GENERALLY 00009370
C             BEST TO USE ALOG(ARG(I)) INSTEAD OF ARG(I) -VS- ZANS(I,J),00009380
C             FOR I=1,NB, AND FOR ANY GIVEN J BETWEEN 1 AND NREL.  NOTE 00009390
C             NB IS USED AS AN ADJUSTABLE DIMENSION IN ZANS(NB,NREL).   00009400
C      NREL - NUMBER OF RELATED CONVOLUTIONS DESIRED (NREL.GE.1).  USE  00009410
C             NREL=1 IF ONLY A SINGLE COMPLEX HANKEL TRANSFORM IS USED. 00009420
C             NREL>1 REQUIRES ARRAY IJREL(2,NREL)  (SEE BELOW).         00009430
C             NOTE NREL IS USED AS ADJUSTABLE DIMENSIONS IN ARRAYS      00009440
C             ZANS(NB,NREL),ZWORK(283,NREL),NORD(NREL),IJREL(2,NREL).   00009450
C       TOL - REQUESTED REAL TRUNCATION TOLERANCE AT BOTH FILTER TAILS  00009460
C             FOR ADAPTIVE CONVOLUTION FOR ALL NB*NREL TRANSFORMS.  THE 00009470
C             TRUNCATION CRITERION IS ESTABLISHED DURING CONVOLUTION IN 00009480
C             A FIXED ABSCISSA RANGE (USING WEIGHTS 131-149) OF EITHER  00009490
C             ORDER FILTER AS THE MAXIMUM ABSOLUTE CONVOLVED PRODUCT    00009500
C             TIMES TOL.  THE CONVOLUTION SUMMATION IS TERMINATED       00009510
C             ON EITHER SIDE OF THE FIXED RANGE WHENEVER THE ABSOLUTE   00009520
C             PRODUCT .LE. THE TRUNCATION CRITERION.  BOTH REAL AND     00009530
C             IMAGINARY PARTS OF THE COMPLEX SUMMATION MUST SATISFY     00009540
C             THE TRUNCATION CRITERION INDEPENDENTLY. IN GENERAL, A     00009550
```

```
C             DECREASING TOLERANCE WILL PRODUCE HIGHER ACCURACY SINCE   00009560
C             MORE FILTER WEIGHTS ARE USED (UNLESS EXPONENT UNDERFLOW   00009570
C             OCCURS IN THE TRANSFORM INPUT FUNCTION EVALUATION).       00009580
C             ONE MAY SET TOL=0.0 TO OBTAIN MAXIMUM ACCURACY FOR ALL    00009590
C             NB*NREL COMPLEX HANKEL TRANSFORMS IN ZANS(NB,NREL).       00009600
C             HOWEVER, THE ACTUAL RELATIVE ERRORS CANNOT BE EXPECTED    00009610
C             TO BE SMALLER THAN ABOUT .1E-7 REGARDLESS OF THE TOLERANCE00009620
C             VALUE USED, SINCE SINGLE-PRECISION FILTER WEIGHTS AND     00009630
C             SINGLE-PRECISION COMPLEX FUNCTIONS ARE USED. IN ANY EVENT,00009640
C             ONE SHOULD ALWAYS CHOOSE TOL<<DESIRED RELATIVE ERROR.     00009650
C          ** ACCURACY WARNING **  SOME HIGHLY OSCILLATORY FUNCTIONS    00009660
C             FUN1(G) AND (OR) LIMITING CASES OF B NEAR MACHINE-ZERO    00009670
C             (OR INFINITY) SHOULD BE AVOIDED, OTHERWISE UNSATISFACTORY 00009680
C             RESULTS (E.G., RELATIVE & ABSOLUTE ERRORS>>TOL) MAY OCCUR.00009690
C      NTOL - NUMBER OF CONSECUTIVE TIMES THE TRUNCATION CRITERION (TOL)00009700
C             IS TO BE MET AT EITHER FILTER TAIL BEFORE FILTER          00009710
C             TRUNCATION OCCURS.  NTOL=1 SHOULD BE USED FOR INPUT       00009720
C             FUNCTIONS THAT DO NOT HAVE MANY ZEROS IN (0,INFINITY). FOR00009730
C             OSCILLATORY FUNCTIONS WITH MANY ZEROS, NTOL>1 MAY BE USED 00009740
C             TO INSURE A PREMATURE CUTOFF DOES NOT OCCUR FOR TRUNCATION00009750
C             (SEE USE OF ITOL,NTOL,TOL IN THE CODE BELOW).             00009760
C      NORD - INTEGER ARRAY NORD(NREL) GIVING THE NREL ORDERS (0 OR 1)  00009770
C             OF EACH RELATED HANKEL TRANSFORM. IF ANY NORD(I),I=1,NREL,00009780
C             IS NOT 0 OR 1, THEN ORDER 1 WILL BE ASSUMED.              00009790
C      FUN1 - NAME OF AN EXTERNAL DECLARED COMPLEX FUNCTION OF A REAL   00009800
C             ARGUMENT DEFINING THE 1ST TRANSFORM INPUT FUNCTION OF THE 00009810
C             SET OF RELATED TRANSFORMS TO BE EVALUATED.  AN EXTERNAL   00009820
C             FUN1 STATEMENT MUST APPEAR IN THE CALLING PROGRAM.  THE   00009830
C             COMPLEX FUNCTION FUN1(G) SUBPROGRAM MUST BE CODED BY THE  00009840
C             USER AND MUST BE A CONTINUOUS DECREASING COMPLEX FUNCTION 00009850
C             FOR ALL REAL G>0.0.  THE VALUE OF G MUST BE UNCHANGED     00009860
C             UPON RETURN FROM FUN1. A MULTIPLE-POLE OF FUN1(G) AT G=0.000009870
C             CAN EXIST, PROVIDED THE HANKEL TRANSFORM CONVERGES (NOTE  00009880
C             FUN1(0.0) IS NOT USED). A SINGLE REAL FUNCTION F1(G) MAY  00009890
C             BE CODED AS FUN1=CMPLX(F1(G),0.0). TWO INDEPENDENT REAL   00009900
C             FUNCTIONS F1(G),F2(G) MAY BE INTEGRATED IN PARALLEL BY    00009910
C             CODING FUN1=CMPLX(F1(G),F2(G)).  GENERALLY, FUN1(G)       00009920
C             IS DEFINED ANALYTICALLY FOR ALL G>0.0.  HOWEVER,          00009930
C             DISCRETELY DEFINED FUNCTIONS MAY BE USED IF FUN1(G)       00009940
C             RETURNS A SMOOTH INTERPOLATION VALUE (E.G., VIA CUBIC     00009950
C             SPLINES) WHICH SATISFIES THE CONTINUITY CONDITION FOR ALL 00009960
C             G>0, AND PROVIDED THE PROPER LIMITING VALUE OF FUN1(G) IS 00009970
C             GIVEN AS G TENDS TO INFINITY.  PARAMETERS OTHER           00009980
C             THAN G NEEDED IN FUN1(G) MAY BE INCLUDED BY USING LABELED 00009990
C             COMMON IN FUN1 AND IN THE USERS CALLING PROGRAM.  IF      00010000
C             FUN1(G) IS AN OSCILLATING FUNCTION, THEN THE HIGHEST      00010010
C             FREQUENCY COMPONENT (IN LOG-SPACE) SHOULD NOT EXCEED THE  00010020
C             FILTER NYQUIST FREQUENCY, 1/(2*0.2).  IN GENERAL,         00010030
C             SUBPROGRAM HANKEL PERFORMS BEST WHEN USING SMOOTH, WELL-  00010040
C             BEHAVED FUNCTIONS FUN1(G), THAT ARE CHARACTERIZED AS      00010050
C             MONOTONICALLY DECREASING FUNCTIONS WITH RELATIVELY FEW    00010060
C             ZEROS FOR G>0. (SEE THE ACCURACY WARNING UNDER TOL, AND   00010070
C             ERROR CONDITION (4).)                                     00010080
C     IJREL - INTEGER ARRAY IJREL(2,NREL) USED WHEN NREL>1 TO DEFINE    00010090
C             THE PAIR OF I,J INTEGER EXPONENTS FOR EACH RELATED INPUT  00010100
```

```
C           FUNCTION.  THE RELATED INPUT FUNCTIONS ARE ASSUMED         00010110
C           TO BE SIMPLY RELATED IN TERMS OF FUN1 VIA THE INTEGER      00010120
C           ARRAY IJREL(2,K),K=2,NREL.  THAT IS, WE ASSUME THE K-TH    00010130
C           RELATED INPUT FUNCTION IS GIVEN (SEE STATEMENT 160) AS     00010140
C           FUNK(G)=G**IJREL(1,K) * FUN1(G)**IJREL(2,K), WHERE         00010150
C           THE INTEGER EXPONENTS MAY BE POSITIVE, ZERO, OR NEGATIVE.  00010160
C           IN THIS WAY, ONLY FUN1 NEED BE DECLARED AN EXTERNAL        00010170
C           COMPLEX FUNCTION. MORE COMPLICATED CODE COULD BE USED FOR  00010180
C           THE RELATED FUNCTIONS, PROVIDED THE MEANING OF IJREL(2,K)  00010190
C           IS REDEFINED AND STATEMENT 160 IS CHANGED (ALSO, SEE       00010200
C           ERROR CONDITION (3) BELOW). WHEN NREL=1, ARRAY             00010210
C           IJREL IS A DUMMY NAME (I.E., NOT REFERENCED).              00010220
C           IF NREL>1, THEN THE STATEMENT AT LABEL 160                 00010230
C           IS DEFINED ONLY FOR K=J=2,...,NREL. THAT IS,               00010240
C           IJREL(1,1),IJREL(2,1) ARE NOT USED IN THIS VERSION.        00010250
C    ZWORK - COMPLEX WORK ARRAY ZWORK(283,NREL), WHICH IS USED TO HOLD 00010260
C           VARIOUS COMPUTED FUNCTIONAL VALUES DURING RELATED AND      00010270
C           LAGGED CONVOLUTIONS.  A STORAGE ROLL FEATURE USING         00010280
C           ZWORK(283,NREL) AND INTERNAL ARRAY KEY(283) ALLOWS FOR     00010290
C           ANY B RANGE (BMIN,BMAX) TO BE USED DURING CONVOLUTION.     00010300
C                                                                      00010310
C  OUTPUT                                                              00010320
C                                                                      00010330
C     ZANS - THE COMPLEX ARRAY ZANS(NB,NREL) IS RETURNED GIVING THE    00010340
C           NB*NREL COMPLEX HANKEL TRANSFORMS, WITH CORRESPONDING      00010350
C           B ARGUMENTS GIVEN IN REAL ARRAY ARG(NB).                   00010360
C      ARG - REAL ARRAY ARG(NB) IS RETURNED GIVING THE RESULTING       00010370
C           B ARGUMENTS IN (BMIN,BMAX), WHERE ARG(I+1)/ARG(I)=EXP(.2),00010380
C           I=1,NB-1 (THIS ARRAY COULD BE ELIMINATED TO SAVE STORAGE   00010390
C           AND REGENERATED AFTER THE CALL HANKEL, IF DESIRED).        00010400
C   NOFUN1 - NUMBER OF DIRECT FUN1 EVALUATIONS USED FOR ALL NB*NREL    00010410
C           COMPLEX HANKEL TRANSFORMS.  NOFUN1 IS USUALLY NOT MORE     00010420
C           THAN THE NUMBER OF WEIGHTS NEEDED FOR A SINGLE DIRECT      00010430
C           CONVOLUTION, FOR ANY NB AND NREL.                          00010440
C     IERR - ERROR RETURN CODE.  THE FOLLOWING CODES ARE POSSIBLE --   00010450
C           = 0, NO ERROR IN INPUT PARAMETERS. ZANS,ARG COMPUTED.      00010460
C           = 1, IMPROPER INPUT PARAMETERS (I.E., NB<1,NREL<1,BMAX<=0,00010470
C             OR BMAX*EXP(-.2*(NB-1))<=0.0).  ZANS,ARG NOT COMPUTED.   00010480
C                                                                      00010490
C                                                                      00010500
C  ERROR CONDITIONS                                                    00010510
C                                                                      00010520
C   (1)  IMPROPER INPUT PARAMETERS GIVEN (SEE IERR=1 ABOVE).           00010530
C   (2)  UNDERFLOW CONDITIONS ARE POSSIBLE DURING CONVOLUTION, DUE TO  00010540
C        THE BEHAVIOR OF FUN1, VALUE OF B IN (BMIN,BMAX), TOL, AND     00010550
C        NTOL. EXPONENT AND (OR) ARITHMETIC UNDERFLOW TRAPS MUST RETURN00010560
C        A VALUE OF 0.0 FOR THE COMPUTER SYSTEM BEING USED.  NOTE THAT 00010570
C        UNDERFLOW MAY ALSO OCCUR IN THE USERS EXTERNAL FUNCTION       00010580
C        FUN1(G) FOR ANY VALUE OF G AS SET BY SUBPROGRAM HANKEL.       00010590
C   (3)  AN UNRECOVERABLE OVERFLOW CONDITION CAN OCCUR IN EXECUTING    00010600
C        STATEMENT 160, DEPENDING ON THE VALUE OF B IN (BMIN,BMAX),    00010610
C        TOL, OR THE INTEGER EXPONENTS USED IN IJREL(2,NREL),NREL>1.   00010620
C        IN GENERAL, EXTREMELY SMALL OR LARGE VALUES OF B SHOULD BE    00010630
C        AVOIDED (SEE ACCURACY WARNING UNDER TOL ABOVE).  ALSO, IN MANY00010640
C        CASES, EXPONENT OVERFLOW CAN BE AVOIDED BY PROPER CHOICE OF   00010650
```

```
C         FUN1 AND THE RELATED INPUT FUNCTION ORDERING DEFINED BY
C         THE IJREL SIGNED INTEGER EXPONENTS.
C    (4)  UNDETECTED ERRORS ARE POSSIBLE IF FUN1 IS IMPROPERLY CODED, OR
C         DOES NOT YIELD SINGLE-PRECISION COMPLEX ACCURACY, OR IS
C         NOT A CONTINUOUS DECREASING COMPLEX FUNCTION FOR ALL G>0.
C
C  USAGE
C
C    SUBPROGRAM HANKEL IS CALLED AS FOLLOWS (USE NUMERICAL VALUES FOR
C    <EXPRESSION>, EXCLUDING < AND >, IN DECLARATIONS) --
C
C      COMPLEX ZANS(<NB>,<NREL> -OR- <NB*NREL>),ZWORK(283,<NREL>)
C      DIMENSION ARG(<NB>),NORD(<NREL>),IJREL(2,<NREL>)
C      EXTERNAL ZFUN1
CC-----READ/LOAD INPUT PARAMETERS FOR HANKEL AS REQUIRED
C      ...
C      CALL HANKEL(BMAX,NB,NREL,TOL,NTOL,NORD,ZFUN1,IJREL,ZWORK,
C     * ZANS,ARG,NOFUN1,IERR)
C      IF(IERR.EQ.1) STOP
C      ...
C      END
C      COMPLEX FUNCTION ZFUN1(G)
CC-----INSERT USER SUPPLIED CODE FOR EVALUATION OF ZFUN1(G),G>0.0
C      END
C
C=======================================================================
      COMPLEX C,CMAX,ZSUM,ZERO,FUN1
      INTEGER KEY(283)
      DOUBLE PRECISION E,ER,Y1,Y,ABSCIS
      DIMENSION T(2),TMAX(2)
      DIMENSION WT0(283),WT1(283)
C-----WE DEFINE COMPLEX C,CMAX TO BE EQUIVALENT TO REAL ARRAYS T(2),
C     TMAX(2), RESPECTIVELY, FOR USE IN THE TRUNCATION CRITERION TESTS,
C     WHERE C IS ANY CONVOLUTION PRODUCT AND CMAX IS THE MAXIMUM
C     CONVOLVED PRODUCT IN THE FIXED ABSCISSA RANGE (SEE PARAMETER TOL).
      EQUIVALENCE (C,T(1)),(CMAX,TMAX(1))
      DATA ZERO/(0.0,0.0)/
C-----ABSCIS=BASE CONSTANT FOR FILTER ABSCISSA GENERATION
      DATA ABSCIS/0.7358852661479794460D0/
C-----E=DEXP(.2D0), ER=1.0D0/E (ALSO USED IN ABSCISSA GENERATION)
      DATA E,ER/1.221402758160169834D0,0.818730753077981859D0/
C-----WT0(I)=J0 HANKEL TRANSFORM FILTER WEIGHTS FOR I=1,283
      DATA
     * WT0(  1),WT0(  2),WT0(  3),WT0(  4),
     * WT0(  5),WT0(  6),WT0(  7),WT0(  8),
     * WT0(  9),WT0( 10),WT0( 11),WT0( 12),
     * WT0( 13),WT0( 14),WT0( 15),WT0( 16),
     * WT0( 17),WT0( 18),WT0( 19),WT0( 20),
     * WT0( 21),WT0( 22),WT0( 23),WT0( 24),
     * WT0( 25),WT0( 26),WT0( 27),WT0( 28),
     * WT0( 29),WT0( 30),WT0( 31),WT0( 32),
     * WT0( 33),WT0( 34),WT0( 35),WT0( 36)/
     * 2.1969101E-11, 4.1201161E-09,-6.1322980E-09, 7.2479291E-09,
     *-7.9821627E-09, 8.5778983E-09,-9.1157294E-09, 9.6615250E-09,
     *-1.0207546E-08, 1.0796633E-08,-1.1393033E-08, 1.2049873E-08,
      *-1.2708789E-08, 1.3446466E-08,-1.4174300E-08, 1.5005577E-08,        00011210
      *-1.5807160E-08, 1.6747136E-08,-1.7625961E-08, 1.8693427E-08,        00011220
      *-1.9650840E-08, 2.0869789E-08,-2.1903555E-08, 2.3305308E-08,        00011230
      *-2.4407377E-08, 2.6033678E-08,-2.7186773E-08, 2.9094334E-08,        00011240
      *-3.0266804E-08, 3.2534013E-08,-3.3672072E-08, 3.6408936E-08,        00011250
      *-3.7425022E-08, 4.0787921E-08,-4.1543242E-08, 4.5756842E-08/        00011260
       DATA                                                                00011270
      * WTO( 37),WTO( 38),WTO( 39),WTO( 40),                               00011280
      * WTO( 41),WTO( 42),WTO( 43),WTO( 44),                               00011290
      * WTO( 45),WTO( 46),WTO( 47),WTO( 48),                               00011300
      * WTO( 49),WTO( 50),WTO( 51),WTO( 52),                               00011310
      * WTO( 53),WTO( 54),WTO( 55),WTO( 56),                               00011320
      * WTO( 57),WTO( 58),WTO( 59),WTO( 60),                               00011330
      * WTO( 61),WTO( 62),WTO( 63),WTO( 64),                               00011340
      * WTO( 65),WTO( 66),WTO( 67),WTO( 68),                               00011350
      * WTO( 69),WTO( 70),WTO( 71),WTO( 72)/                               00011360
      *-4.6035233E-08, 5.1425075E-08,-5.0893896E-08, 5.7934897E-08,        00011370
      *-5.6086570E-08, 6.5475248E-08,-6.1539913E-08, 7.4301996E-08,        00011380
      *-6.7117043E-08, 8.4767837E-08,-7.2583120E-08, 9.7366568E-08,        00011390
      *-7.7553611E-08, 1.1279873E-07,-8.1416723E-08, 1.3206914E-07,        00011400
      *-8.3217217E-08, 1.5663185E-07,-8.1482581E-08, 1.8860593E-07,        00011410
      *-7.3963141E-08, 2.3109673E-07,-5.7243707E-08, 2.8867452E-07,        00011420
      *-2.6163525E-08, 3.6808773E-07, 2.7049871E-08, 4.7932617E-07,        00011430
      * 1.1407365E-07, 6.3720626E-07, 2.5241961E-07, 8.6373487E-07,        00011440
      * 4.6831433E-07, 1.1916346E-06, 8.0099716E-07, 1.6696015E-06/        00011450
       DATA                                                                00011460
      * WTO( 73),WTO( 74),WTO( 75),WTO( 76),                               00011470
      * WTO( 77),WTO( 78),WTO( 79),WTO( 80),                               00011480
      * WTO( 81),WTO( 82),WTO( 83),WTO( 84),                               00011490
      * WTO( 85),WTO( 86),WTO( 87),WTO( 88),                               00011500
      * WTO( 89),WTO( 90),WTO( 91),WTO( 92),                               00011510
      * WTO( 93),WTO( 94),WTO( 95),WTO( 96),                               00011520
      * WTO( 97),WTO( 98),WTO( 99),WTO(100),                               00011530
      * WTO(101),WTO(102),WTO(103),WTO(104),                               00011540
      * WTO(105),WTO(106),WTO(107),WTO(108)/                               00011550
      * 1.3091334E-06, 2.3701475E-06, 2.0803829E-06, 3.4012978E-06,        00011560
      * 3.2456774E-06, 4.9240402E-06, 5.0005198E-06, 7.1783540E-06,        00011570
      * 7.6367633E-06, 1.0522038E-05, 1.1590021E-05, 1.5488635E-05,        00011580
      * 1.7510398E-05, 2.2873836E-05, 2.6368006E-05, 3.3864387E-05,        00011590
      * 3.9610390E-05, 5.0230379E-05, 5.9397373E-05, 7.4612122E-05,        00011600
      * 8.8951409E-05, 1.1094809E-04, 1.3308026E-04, 1.6511335E-04,        00011610
      * 1.9895671E-04, 2.4587195E-04, 2.9728181E-04, 3.6629770E-04,        00011620
      * 4.4402013E-04, 5.4589361E-04, 6.6298832E-04, 8.1375348E-04,        00011630
      * 9.8971624E-04, 1.2132772E-03, 1.4772052E-03, 1.8092022E-03/        00011640
       DATA                                                                00011650
      * WTO(109),WTO(110),WTO(111),WTO(112),                               00011660
      * WTO(113),WTO(114),WTO(115),WTO(116),                               00011670
      * WTO(117),WTO(118),WTO(119),WTO(120),                               00011680
      * WTO(121),WTO(122),WTO(123),WTO(124),                               00011690
      * WTO(125),WTO(126),WTO(127),WTO(128),                               00011700
      * WTO(129),WTO(130),WTO(131),WTO(132),                               00011710
      * WTO(133),WTO(134),WTO(135),WTO(136),                               00011720
      * WTO(137),WTO(138),WTO(139),WTO(140),                               00011730
      * WTO(141),WTO(142),WTO(143),WTO(144)/                               00011740
      * 2.2045122E-03, 2.6980811E-03, 3.2895354E-03, 4.0238764E-03,        00011750
```

```
     * 4.9080203E-03, 6.0010999E-03, 7.3216878E-03, 8.9489225E-03,      00011760
     * 1.0919448E-02, 1.3340696E-02, 1.6276399E-02, 1.9873311E-02,      00011770
     * 2.4233627E-02, 2.9555699E-02, 3.5990069E-02, 4.3791529E-02,      00011780
     * 5.3150319E-02, 6.4341372E-02, 7.7506720E-02, 9.2749987E-02,      00011790
     * 1.0980561E-01, 1.2791555E-01, 1.4525830E-01, 1.5820085E-01,      00011800
     * 1.6058576E-01, 1.4196085E-01, 8.9781222E-02,-1.0238278E-02,      00011810
     *-1.5083434E-01,-2.9059573E-01,-2.9105437E-01,-3.7973244E-02,      00011820
     * 3.8273717E-01, 2.2014118E-01,-4.7342635E-01, 1.9331133E-01/      00011830
      DATA                                                              00011840
     * WT0(145),WT0(146),WT0(147),WT0(148),                             00011850
     * WT0(149),WT0(150),WT0(151),WT0(152),                             00011860
     * WT0(153),WT0(154),WT0(155),WT0(156),                             00011870
     * WT0(157),WT0(158),WT0(159),WT0(160),                             00011880
     * WT0(161),WT0(162),WT0(163),WT0(164),                             00011890
     * WT0(165),WT0(166),WT0(167),WT0(168),                             00011900
     * WT0(169),WT0(170),WT0(171),WT0(172),                             00011910
     * WT0(173),WT0(174),WT0(175),WT0(176),                             00011920
     * WT0(177),WT0(178),WT0(179),WT0(180)/                             00011930
     * 5.3839527E-02,-1.1909845E-01, 9.9317051E-02,-6.6152628E-02,      00011940
     * 4.0703241E-02,-2.4358316E-02, 1.4476533E-02,-8.6198067E-03,      00011950
     * 5.1597053E-03,-3.1074602E-03, 1.8822342E-03,-1.1456545E-03,      00011960
     * 7.0004347E-04,-4.2904226E-04, 2.6354444E-04,-1.6215439E-04,      00011970
     * 9.9891279E-05,-6.1589037E-05, 3.7996921E-05,-2.3452250E-05,      00011980
     * 1.4479572E-05,-8.9417427E-06, 5.5227518E-06,-3.4114252E-06,      00011990
     * 2.1074101E-06,-1.3019229E-06, 8.0433617E-07,-4.9693681E-07,      00012000
     * 3.0702417E-07,-1.8969219E-07, 1.1720069E-07,-7.2412496E-08,      00012010
     * 4.4740283E-08,-2.7643004E-08, 1.7079403E-08,-1.0552634E-08/      00012020
      DATA                                                              00012030
     * WT0(181),WT0(182),WT0(183),WT0(184),                             00012040
     * WT0(185),WT0(186),WT0(187),WT0(188),                             00012050
     * WT0(189),WT0(190),WT0(191),WT0(192),                             00012060
     * WT0(193),WT0(194),WT0(195),WT0(196),                             00012070
     * WT0(197),WT0(198),WT0(199),WT0(200),                             00012080
     * WT0(201),WT0(202),WT0(203),WT0(204),                             00012090
     * WT0(205),WT0(206),WT0(207),WT0(208),                             00012100
     * WT0(209),WT0(210),WT0(211),WT0(212),                             00012110
     * WT0(213),WT0(214),WT0(215),WT0(216)/                             00012120
     * 6.5200311E-09,-4.0284597E-09, 2.4890232E-09,-1.5378695E-09,      00012130
     * 9.5019040E-10,-5.8708696E-10, 3.6273937E-10,-2.2412348E-10,      00012140
     * 1.3847792E-10,-8.5560821E-11, 5.2865474E-11,-3.2664392E-11,      00012150
     * 2.0182948E-11,-1.2470979E-11, 7.7057678E-12,-4.7611713E-12,      00012160
     * 2.9415274E-12,-1.8170081E-12, 1.1221034E-12,-6.9271067E-13,      00012170
     * 4.2739744E-13,-2.6344388E-13, 1.6197105E-13,-9.9147443E-14,      00012180
     * 6.0487998E-14,-3.6973097E-14, 2.2817964E-14,-1.4315547E-14,      00012190
     * 9.1574735E-15,-5.9567236E-15, 3.9209969E-15,-2.5911739E-15,      00012200
     * 1.6406939E-15,-8.8248590E-16, 3.0195409E-16, 2.2622634E-17/      00012210
      DATA                                                              00012220
     * WT0(217),WT0(218),WT0(219),WT0(220),                             00012230
     * WT0(221),WT0(222),WT0(223),WT0(224),                             00012240
     * WT0(225),WT0(226),WT0(227),WT0(228),                             00012250
     * WT0(229),WT0(230),WT0(231),WT0(232),                             00012260
     * WT0(233),WT0(234),WT0(235),WT0(236),                             00012270
     * WT0(237),WT0(238),WT0(239),WT0(240),                             00012280
     * WT0(241),WT0(242),WT0(243),WT0(244),                             00012290
     * WT0(245),WT0(246),WT0(247),WT0(248),                             00012300
     * WTO(249),WTO(250),WTO(251),WTO(252)/                             00012310
     *-8.0942556E-17,-3.7172363E-17, 1.9299542E-16,-3.3388160E-16,      00012320
     * 4.6174116E-16,-5.8627358E-16, 7.2227767E-16,-8.7972941E-16,      00012330
     * 1.0211793E-15,-1.0940039E-15, 1.0789555E-15,-9.7089714E-16,      00012340
     * 7.4110927E-16,-4.1700094E-16, 8.5977184E-17, 1.3396469E-16,      00012350
     *-1.7838410E-16, 4.8975421E-17, 1.9398153E-16,-5.0046989E-16,      00012360
     * 8.3280985E-16,-1.1544640E-15, 1.4401527E-15,-1.6637066E-15,      00012370
     * 1.7777129E-15,-1.7322187E-15, 1.5247247E-15,-1.1771155E-15,      00012380
     * 6.9747910E-16,-1.2088956E-16,-4.8382957E-16, 1.0408292E-15,      00012390
     *-1.5220450E-15, 1.9541597E-15,-2.4107448E-15, 2.9241438E-15/      00012400
      DATA                                                              00012410
     * WTO(253),WTO(254),WTO(255),WTO(256),                             00012420
     * WTO(257),WTO(258),WTO(259),WTO(260),                             00012430
     * WTO(261),WTO(262),WTO(263),WTO(264),                             00012440
     * WTO(265),WTO(266),WTO(267),WTO(268),                             00012450
     * WTO(269),WTO(270),WTO(271),WTO(272),                             00012460
     * WTO(273),WTQ(274),WTO(275),WTO(276),                             00012470
     * WTO(277),WTO(278),WTO(279),WTO(280),                             00012480
     * WTO(281),WTO(282),WTO(283)/                                      00012490
     *-3.5176475E-15, 4.2276125E-15,-5.0977851E-15, 6.1428456E-15,      00012500
     *-7.3949962E-15, 8.8597601E-15,-1.0515959E-14, 1.2264584E-14,      00012510
     *-1.3949870E-14, 1.5332490E-14,-1.6146782E-14, 1.6084121E-14,      00012520
     *-1.4962523E-14, 1.2794804E-14,-9.9286701E-15, 6.8825809E-15,      00012530
     *-4.0056107E-15, 1.5965079E-15,-7.2732961E-18,-4.0433218E-16,      00012540
     *-6.5679655E-16, 3.3011866E-15,-7.3545910E-15, 1.2394851E-14,      00012550
     *-1.7947697E-14, 2.3774303E-14,-3.0279168E-14, 3.9252831E-14,      00012560
     *-5.5510504E-14, 9.0505371E-14,-1.7064873E-13/                     00012570
C-----WT1(I)=J1 HANKEL TRANSFORM FILTER WEIGHTS FOR I=1,283             00012580
      DATA                                                              00012590
     * WT1(  1),WT1(  2),WT1(  3),WT1(  4),                             00012600
     * WT1(  5),WT1(  6),WT1(  7),WT1(  8),                             00012610
     * WT1(  9),WT1( 10),WT1( 11),WT1( 12),                             00012620
     * WT1( 13),WT1( 14),WT1( 15),WT1( 16),                             00012630
     * WT1( 17),WT1( 18),WT1( 19),WT1( 20),                             00012640
     * WT1( 21),WT1( 22),WT1( 23),WT1( 24),                             00012650
     * WT1( 25),WT1( 26),WT1( 27),WT1( 28),                             00012660
     * WT1( 29),WT1( 30),WT1( 31),WT1( 32),                             00012670
     * WT1( 33),WT1( 34),WT1( 35),WT1( 36)/                             00012680
     *-4.2129715E-16, 5.3667031E-15,-7.1183962E-15, 8.9478500E-15,      00012690
     *-1.0767891E-14, 1.2362265E-14,-1.3371129E-14, 1.3284178E-14,      00012700
     *-1.1714302E-14, 8.4134738E-15,-3.7726725E-15,-1.4263879E-15,      00012710
     * 6.1279163E-15,-9.1102765E-15, 9.9696405E-15,-9.3649955E-15,      00012720
     * 8.6009018E-15,-8.9749846E-15, 1.1153987E-14,-1.4914821E-14,      00012730
     * 1.9314024E-14,-2.3172388E-14, 2.5605477E-14,-2.6217555E-14,      00012740
     * 2.5057768E-14,-2.2485539E-14, 1.9022752E-14,-1.5198084E-14,      00012750
     * 1.1422464E-14,-7.9323958E-15, 4.8421406E-15,-2.1875032E-15,      00012760
     *-3.2177842E-17, 1.8637565E-15,-3.3683643E-15, 4.6132219E-15/      00012770
      DATA                                                              00012780
     * WT1( 37),WT1( 38),WT1( 39),WT1( 40),                             00012790
     * WT1( 41),WT1( 42),WT1( 43),WT1( 44),                             00012800
     * WT1( 45),WT1( 46),WT1( 47),WT1( 48),                             00012810
     * WT1( 49),WT1( 50),WT1( 51),WT1( 52),                             00012820
     * WT1( 53),WT1( 54),WT1( 55),WT1( 56),                             00012830
     * WT1( 57),WT1( 58),WT1( 59),WT1( 60),                             00012840
     * WT1( 61),WT1( 62),WT1( 63),WT1( 64),                             00012850
     * WT1( 65),WT1( 66),WT1( 67),WT1( 68),                              00012860
     * WT1( 69),WT1( 70),WT1( 71),WT1( 72)/                              00012870
     *-5.6209538E-15, 6.4192841E-15,-6.8959928E-15, 6.9895792E-15,      00012880
     *-6.5355935E-15, 5.6125163E-15,-4.1453931E-15, 2.6358827E-15,      00012890
     *-9.5104370E-16, 1.4600474E-16, 5.6166519E-16, 8.2899246E-17,      00012900
     * 5.0032100E-16, 4.3752205E-16, 2.1052293E-15,-9.5451973E-16,      00012910
     * 6.4004437E-15,-2.1926177E-15, 1.1651003E-14, 5.8415433E-16,      00012920
     * 1.8044664E-14, 1.0755745E-14, 3.0159022E-14, 3.3506138E-14,      00012930
     * 5.8709354E-14, 8.1475200E-14, 1.2530006E-13, 1.8519112E-13,      00012940
     * 2.7641786E-13, 4.1330823E-13, 6.1506209E-13, 9.1921659E-13,      00012950
     * 1.3698462E-12, 2.0447427E-12, 3.0494477E-12, 4.5501001E-12/      00012960
      DATA                                                               00012970
     * WT1( 73),WT1( 74),WT1( 75),WT1( 76),                              00012980
     * WT1( 77),WT1( 78),WT1( 79),WT1( 80),                              00012990
     * WT1( 81),WT1( 82),WT1( 83),WT1( 84),                              00013000
     * WT1( 85),WT1( 86),WT1( 87),WT1( 88),                              00013010
     * WT1( 89),WT1( 90),WT1( 91),WT1( 92),                              00013020
     * WT1( 93),WT1( 94),WT1( 95),WT1( 96),                              00013030
     * WT1( 97),WT1( 98),WT1( 99),WT1(100),                              00013040
     * WT1(101),WT1(102),WT1(103),WT1(104),                              00013050
     * WT1(105),WT1(106),WT1(107),WT1(108)/                              00013060
     * 6.7870250E-12, 1.0126237E-11, 1.5104976E-11, 2.2536053E-11,      00013070
     * 3.3617368E-11, 5.0153839E-11, 7.4818173E-11, 1.1161804E-10,      00013080
     * 1.6651222E-10, 2.4840923E-10, 3.7058109E-10, 5.5284353E-10,      00013090
     * 8.2474468E-10, 1.2303750E-09, 1.8355034E-09, 2.7382502E-09,      00013100
     * 4.0849867E-09, 6.0940898E-09, 9.0913020E-09, 1.3562651E-08,      00013110
     * 2.0233058E-08, 3.0184244E-08, 4.5029477E-08, 6.7176304E-08,      00013120
     * 1.0021488E-07, 1.4950371E-07, 2.2303208E-07, 3.3272689E-07,      00013130
     * 4.9636623E-07, 7.4049804E-07, 1.1046805E-06, 1.6480103E-06,      00013140
     * 2.4585014E-06, 3.6677163E-06, 5.4714550E-06, 8.1626422E-06/      00013150
      DATA                                                               00013160
     * WT1(109),WT1(110),WT1(111),WT1(112),                              00013170
     * WT1(113),WT1(114),WT1(115),WT1(116),                              00013180
     * WT1(117),WT1(118),WT1(119),WT1(120),                              00013190
     * WT1(121),WT1(122),WT1(123),WT1(124),                              00013200
     * WT1(125),WT1(126),WT1(127),WT1(128),                              00013210
     * WT1(129),WT1(130),WT1(131),WT1(132),                              00013220
     * WT1(133),WT1(134),WT1(135),WT1(136),                              00013230
     * WT1(137),WT1(138),WT1(139),WT1(140),                              00013240
     * WT1(141),WT1(142),WT1(143),WT1(144)/                              00013250
     * 1.2176782E-05, 1.8166179E-05, 2.7099223E-05, 4.0428804E-05,      00013260
     * 6.0307294E-05, 8.9971508E-05, 1.3420195E-04, 2.0021123E-04,      00013270
     * 2.9860417E-04, 4.4545291E-04, 6.6423156E-04, 9.9073275E-04,      00013280
     * 1.4767050E-03, 2.2016806E-03, 3.2788147E-03, 4.8837292E-03,      00013290
     * 7.2596811E-03, 1.0788355E-02, 1.5973323E-02, 2.3612041E-02,      00013300
     * 3.4655327E-02, 5.0608141E-02, 7.2827752E-02, 1.0337889E-01,      00013310
     * 1.4207357E-01, 1.8821315E-01, 2.2996815E-01, 2.5088500E-01,      00013320
     * 2.0334626E-01, 6.0665451E-02,-2.0275683E-01,-3.5772336E-01,      00013330
     *-1.8280529E-01, 4.7014634E-01, 7.2991233E-03,-3.0614594E-01/      00013340
      DATA                                                               00013350
     * WT1(145),WT1(146),WT1(147),WT1(148),                              00013360
     * WT1(149),WT1(150),WT1(151),WT1(152),                              00013370
     * WT1(153),WT1(154),WT1(155),WT1(156),                              00013380
     * WT1(157),WT1(158),WT1(159),WT1(160),                              00013390
     * WT1(161),WT1(162),WT1(163),WT1(164),                              00013400
     * WT1(165),WT1(166),WT1(167),WT1(168),
     * WT1(169),WT1(170),WT1(171),WT1(172),
     * WT1(173),WT1(174),WT1(175),WT1(176),
     * WT1(177),WT1(178),WT1(179),WT1(180)/
     * 2.4781735E-01,-1.1149185E-01, 2.5985386E-02, 1.0850279E-02,
     *-2.2830217E-02, 2.4644647E-02,-2.2895284E-02, 2.0197032E-02,
     *-1.7488968E-02, 1.5057670E-02,-1.2953923E-02, 1.1153254E-02,
     *-9.6138436E-03, 8.2952090E-03,-7.1628361E-03, 6.1882910E-03,
     *-5.3482055E-03, 4.6232056E-03,-3.9970542E-03, 3.4560118E-03,
     *-2.9883670E-03, 2.5840861E-03,-2.2345428E-03, 1.9323046E-03,
     *-1.6709583E-03, 1.4449655E-03,-1.2495408E-03, 1.0805480E-03,
     *-9.3441130E-04, 8.0803899E-04,-6.9875784E-04, 6.0425624E-04,
     *-5.2253532E-04, 4.5186652E-04,-3.9075515E-04, 3.3790861E-04/
      DATA
     * WT1(181),WT1(182),WT1(183),WT1(184),
     * WT1(185),WT1(186),WT1(187),WT1(188),
     * WT1(189),WT1(190),WT1(191),WT1(192),
     * WT1(193),WT1(194),WT1(195),WT1(196),
     * WT1(197),WT1(198),WT1(199),WT1(200),
     * WT1(201),WT1(202),WT1(203),WT1(204),
     * WT1(205),WT1(206),WT1(207),WT1(208),
     * WT1(209),WT1(210),WT1(211),WT1(212),
     * WT1(213),WT1(214),WT1(215),WT1(216)/
     *-2.9220916E-04, 2.5269019E-04,-2.1851585E-04, 1.8896332E-04,
     *-1.6340753E-04, 1.4130796E-04,-1.2219719E-04, 1.0567099E-04,
     *-9.1379828E-05, 7.9021432E-05,-6.8334412E-05, 5.9092726E-05,
     *-5.1100905E-05, 4.4189914E-05,-3.8213580E-05, 3.3045496E-05,
     *-2.8576356E-05, 2.4711631E-05,-2.1369580E-05, 1.8479514E-05,
     *-1.5980307E-05, 1.3819097E-05,-1.1950174E-05, 1.0334008E-05,
     *-8.9364160E-06, 7.7278366E-06,-6.6827083E-06, 5.7789251E-06,
     *-4.9973715E-06, 4.3215167E-06,-3.7370660E-06, 3.2316575E-06,
     *-2.7946015E-06, 2.4166539E-06,-2.0898207E-06, 1.8071890E-06/
      DATA
     * WT1(217),WT1(218),WT1(219),WT1(220),
     * WT1(221),WT1(222),WT1(223),WT1(224),
     * WT1(225),WT1(226),WT1(227),WT1(228),
     * WT1(229),WT1(230),WT1(231),WT1(232),
     * WT1(233),WT1(234),WT1(235),WT1(236),
     * WT1(237),WT1(238),WT1(239),WT1(240),
     * WT1(241),WT1(242),WT1(243),WT1(244),
     * WT1(245),WT1(246),WT1(247),WT1(248),
     * WT1(249),WT1(250),WT1(251),WT1(252)/
     *-1.5627811E-06, 1.3514274E-06,-1.1686576E-06, 1.0106059E-06,
     *-8.7392952E-07, 7.5573750E-07,-6.5353002E-07, 5.6514528E-07,
     *-4.8871388E-07, 4.2261921E-07,-3.6546333E-07, 3.1603732E-07,
     *-2.7329579E-07, 2.3633470E-07,-2.0437231E-07, 1.7673258E-07,
     *-1.5283091E-07, 1.3216174E-07,-1.1428792E-07, 9.8831386E-08,
     *-8.5465227E-08, 7.3906734E-08,-6.3911437E-08, 5.5267923E-08,
     *-4.7793376E-08, 4.1329702E-08,-3.5740189E-08, 3.0906612E-08,
     *-2.6726739E-08, 2.3112160E-08,-1.9986424E-08, 1.7283419E-08,
     *-1.4945974E-08, 1.2924650E-08,-1.1176694E-08, 9.6651347E-09/
      DATA
     * WT1(253),WT1(254),WT1(255),WT1(256),
     * WT1(257),WT1(258),WT1(259),WT1(260),
     * WT1(261),WT1(262),WT1(263),WT1(264),
     * WT1(265),WT1(266),WT1(267),WT1(268),
     * WT1(269),WT1(270),WT1(271),WT1(272),
     * WT1(273),WT1(274),WT1(275),WT1(276),
     * WT1(277),WT1(278),WT1(279),WT1(280),
     * WT1(281),WT1(282),WT1(283)/
     *-8.3580023E-09, 7.2276490E-09,-6.2501673E-09, 5.4048822E-09,
     *-4.6739154E-09, 4.0418061E-09,-3.4951847E-09, 3.0224895E-09,
     *-2.6137226E-09, 2.2602382E-09,-1.9545596E-09, 1.6902214E-09,
     *-1.4616324E-09, 1.2639577E-09,-1.0930164E-09, 9.4519327E-10,
     *-8.1736202E-10, 7.0681930E-10,-6.1122713E-10, 5.2856342E-10,
     *-4.5707937E-10, 3.9526267E-10,-3.4180569E-10, 2.9557785E-10,
     *-2.5560176E-10, 2.2103233E-10,-1.9113891E-10, 1.6528994E-10,
     *-1.4294012E-10, 1.2361991E-10,-8.2740936E-11/
C
      NOFUN1=0
C-----ERROR CHECKS
      IF(NB.LT.1.OR.NREL.LT.1.OR.BMAX.LE.0.0) GO TO 9999
      Y=DBLE(BMAX)*ER**(NB-1)
      IF(Y.LE.0.0D0) GO TO 9999
      IERR=0
C-----INITIALIZE RELATED CONVOLUTION WITHIN LAGGED CONVOLUTION LOOPS
      DO 10 I=1,283
   10   KEY(I)=0
      NB1=NB+1
      LAG=-1
C-----PRESET INITIAL FILTER ABSCISSA FOR STARTING BMAX, GENERATED IN
C     DOUBLE-PRECISION (TO REDUCE ROUND-OFF), BUT USED IN SINGLE-
C     PRECISION IN THE COMPLEX FUNCTION FUN1(G).  NOTE THE ABSCISSAS
C     ARE EQUALLY SPACED (E=DEXP(.2D0), ER=1.0D0/E) IN LOG-SPACE.
      Y1=ABSCIS/DBLE(BMAX)
C-----LAGGED CONVOLUTION, OUTERMOST LOOP 1010
      DO 1010 ILAG=1,NB
        LAG=LAG+1
        ISTORE=NB1-ILAG
        IF(LAG.GT.0) Y1=Y1*E
        ARG(ISTORE)=ABSCIS/Y1
C-------RELATED CONVOLUTION, INNERMOST LOOP 1000
        DO 1000 JREL=1,NREL
C---------SPECIAL CASE FLAG NONE=1 IS SET IF FUN1(G)=0 FOR ALL G IN
C         FILTER FIXED RANGE (USING WEIGHTS 131-149).
          NONE=0
          ITOL=NTOL
          ZSUM=ZERO
          CMAX=ZERO
          Y=Y1
C---------BEGIN RIGHT SIDE CONVOLUTION AT WEIGHT 131 (M=RETURN LABEL)
          ASSIGN 20 TO M
          I=131
          Y=Y*E
C---------CALL PSEUDO SUBROUTINE AT 100 (RETURN TO 20 VIA M ASSIGNED)
          GO TO 100
   20     TMAX(1)=AMAX1(ABS(T(1)),TMAX(1))
          TMAX(2)=AMAX1(ABS(T(2)),TMAX(2))
          I=I+1
          Y=Y*E
```

```
C---------CALL PSEUDO SUBROUTINE AT 100 (RETURN TO 20 VIA M ASSIGNED)
          IF(I.LE.149) GO TO 100
          IF(TMAX(1).EQ.0.0.AND.TMAX(2).EQ.0.0) NONE=1
C---------ESTABLISH TRUNCATION CRITERION (CMAX=CMPLX(TMAX(1),TMAX(2))
          CMAX=TOL*CMAX
          ASSIGN 30 TO M
C---------CALL PSEUDO SUBROUTINE AT 100 (RETURN TO 30 VIA M ASSIGNED)
          GO TO 100
C---------CHECK FOR FILTER TRUNCATION AT RIGHT END
   30     IF(ABS(T(1)).LE.TMAX(1).AND.ABS(T(2)).LE.TMAX(2)) GO TO 50
          ITOL=NTOL
   40     I=I+1
          Y=Y*E
C---------CALL PSEUDO SUBROUTINE AT 100 (RETURN TO 30 VIA M ASSIGNED)
          IF(I.LE.283) GO TO 100
   50     ITOL=ITOL-1
          IF(ITOL.GT.0.AND.I.LT.283) GO TO 40
          ITOL=NTOL
          Y=Y1
C---------CONTINUE WITH LEFT SIDE CONVOLUTION AT WEIGHT 130
          ASSIGN 60 TO M
          I=130
C---------CALL PSEUDO SUBROUTINE AT 100 (RETURN TO 60 VIA M ASSIGNED)
          GO TO 100
C---------CHECK FOR FILTER TRUNCATION AT LEFT END
   60     IF(ABS(T(1)).LE.TMAX(1).AND.ABS(T(2)).LE.TMAX(2).AND.
     *     NONE.EQ.0) GO TO 80
          ITOL=NTOL
   70     I=I-1
          Y=Y*ER
C---------CALL PSEUDO SUBROUTINE AT 100 (RETURN TO 60 VIA M ASSIGNED)
          IF(I.GT.0) GO TO 100
   80     ITOL=ITOL-1
          IF(ITOL.GT.0.AND.I.GT.1) GO TO 70
C---------NORMALIZE ZSUM BY ARG(ISTORE) TO ACCOUNT FOR INTEGRATION
C         RANGE CHANGE, AND STORE IN ZANS(ISTORE,JREL)
          ZANS(ISTORE,JREL)=ZSUM/ARG(ISTORE)
C---------SKIP OVER PSEUDO SUBROUTINE TO END OF DO 1000 INNERMOST LOOP
          GO TO 1000
C
C=======================================================================
C=====STORE/RETRIEVE PSEUDO SUBROUTINE FOR RELATED/LAGGED CONVOLUTION.
C     THE INTERNAL (PSEUDO) SUBROUTINE ENTRY IS LABEL 100, AND RETURNS
C     TO THE LABEL ASSIGNED TO M.  THIS CALLING MECHANISM COULD OCCUR
C     A MAXIMUM OF 283*NB*NREL TIMES, WHERE PARAMETERS NB>0 AND NREL>0
C     CAN BE ARBITRARILY LARGE.  IF A MORE-STRUCTURED STANDARD FORTRAN
C     SUBROUTINE CALL WAS USED, THEN THE USUAL COMPILER LINKAGE
C     CONVENTION COULD GENERATE A MAXIMUM OF 283*NB*NREL MACHINE-
C     LANGUAGE INSTRUCTIONS FOR REGISTER SAVES/RESTORES AND OTHER
C     MEMORY REFERENCES.  FOR MOST COMPILERS, TIMING TESTS REVEAL THAT
C     THE PSEUDO-CALL METHOD USED HERE GENERATED FASTER MACHINE CODE
C     THAN WITH USING EXTERNAL SUBROUTINE CALLS (E.G., CALL LINKAGE
C     VERSUS PSEUDO-CALL RATIO WAS 2.6:1 ON A VAX-11/780 USING
C     NB=50,NREL=61, AND NOFUN1=199).
C
```

```
C=========PSEUDO-CALL ENTRY POINT AT 100 (RETURNS VIA GO TO M BELOW)
  100      LOOK=I+LAG
           IQ=LOOK/284
           IR=MOD(LOOK,284)
           IF(IR.EQ.0) IR=1
           IROLL=IQ*283
           IF(KEY(IR).LE.IROLL) GO TO 150
C=========USE EXISTING SAVED FUNCTIONAL VALUES IN ZWORK(IR,JREL)
  110      IF(NORD(JREL)) 130,120,130
  120      C=ZWORK(IR,JREL)*WT0(I)
           GO TO 140
  130      C=ZWORK(IR,JREL)*WT1(I)
  140      ZSUM=ZSUM+C
C=========RETURN CONVOLUTION CONTROL VIA ASSIGNED M VALUE, AND WITH
C          THE LAST CONVOLUTION PRODUCT C=CMPLX(T(1),T(2)).
           GO TO M,(20,30,60)
C=========COMPUTE EXTERNAL FUN1 DIRECTLY ONLY WHEN NECESSARY
  150      KEY(IR)=IROLL+IR
           G=Y
           ZWORK(IR,1)=FUN1(G).
           NOFUN1=NOFUN1+1
           IF(NREL.EQ.1) GO TO 110
C=========FILL-IN REMAINING RELATED ZWORK(IR,J),J=2,NREL FOR THIS IR
           DO 160 J=2,NREL
C***********FOR OTHER THAN SIMPLE RELATIONS, THE FOLLOWING STATEMENT
C             COULD BE CHANGED (AND ALSO THE MEANING OF IJREL(2,NREL)).
  160         ZWORK(IR,J)=CMPLX(G**IJREL(1,J),0.0)*ZWORK(IR,1)**IJREL(2,J)
           GO TO 110
C=========END OF PSEUDO SUBROUTINE (ENTRY 100, RETURN GO TO M ABOVE)
C=====================================================================
C
C-------END LOOP 1000 (GET REMAINING RELATED CONVOLUTIONS FOR THIS ARG)
 1000    CONTINUE
C-----END LOOP 1010 (GET REMAINING LAGGED CONVOLUTIONS FOR NEXT ARG)
 1010 CONTINUE
C-----EXIT WITH ZANS(NB,NREL),ARG(NB) COMPLETED WITH MINIMAL FUN1 CALLS
      RETURN
 9999 IERR=1
      RETURN
      END
      SUBROUTINE IKS(B8,I1K1,IKDIF)
C--COMPUTE MODIFIED BESSEL FUNCTION (I & K) SPECIAL COMBINATIONS FOR
C  PARAMETERS
C     B8    = DOUBLE PRECISION ARGUMENT (=B/DSQRT(2.D0) HERE)
C     I1K1  = I1*K1 COMPLEX RESULT
C     IKDIF = 4*I1*K1-(B8*DSQRT(I))*(I0*K1-I1*K0) COMPLEX RESULT DONE IN
C             DP BEFORE CMPLX"ING.
C--SUBROUTINE KELVIN CALLED
C
      DOUBLE PRECISION B8,BB(8),BETA,Q1,Q2,R1,R2
      COMPLEX I1K1,IKDIF,CAMBDA,DENOM,DENOM1,TERM0,TERM1,TERM11
      COMPLEX S11,SI0,SI1,SK0,SK1,ONE
      DATA ONE/(1.0,0.0)/
      IF(B8.GT.20.D0) GO TO 10
      CALL KELVIN(B8,8,BB)
      Q1=-BB(6)*BB(8)+BB(5)*BB(7)
      Q2= BB(5)*BB(8)+BB(6)*BB(7)
      I1K1=CMPLX(SNGL(Q1),SNGL(Q2))
      R1=-BB(1)*BB(8)-BB(2)*BB(7) -
     &    BB(6)*BB(3)-BB(5)*BB(4)
      R2=-BB(2)*BB(8)+BB(1)*BB(7) +
     &    BB(5)*BB(3)-BB(6)*BB(4)
      BETA=.7071067811865475D0*B8
      Q1=4.0D0*Q1-BETA*(R1-R2)
      Q2=4.0D0*Q2-BETA*(R1+R2)
      IKDIF=CMPLX(SNGL(Q1),SNGL(Q2))
      RETURN
   10 B=SNGL(B8/0.7071067811865475D0)
      TOL=1.E-6
C--FOR LARGE ARGUMENTS, USE ABRAMOWITZ AND STEGUN
C          ASYMPTOTIC FORMULAS FOR LARGE ARGUMENTS
C          9.7.1 THROUGH 9.7.5, P. 377-378.
      CAMBDA=B*CMPLX(1.0,1.0)/2.
      IKDIF=CMPLX(100.,0.)
      ISIGN=1
      DENOM=8.*CAMBDA
      DENOM1=(2.*CAMBDA)**2
      NODD=1
      TERM0=ONE
      TERM1=ONE
      TERM11=ONE
      S11=ONE
      SI0=ONE
      SI1=ONE
      SK0=ONE
      SK1=ONE
    1 NODD2=NODD*NODD
      OIKDIF=CABS(IKDIF)
      TERM1=TERM1*CMPLX(4.-NODD2,0.)/DENOM
      TERM0=TERM0*CMPLX(-FLOAT(NODD2),0.)/DENOM
      TERM11=TERM11*CMPLX(NODD*(4.-NODD2)/(NODD+1),0.)/DENOM1
      ISIGN=-ISIGN
      S11=S11+ISIGN*TERM11
      SI0=SI0+ISIGN*TERM0
      SI1=SI1+ISIGN*TERM1
      SK0=SK0+TERM0
      SK1=SK1+TERM1
      IKDIF=SI0*SK1-SK0*SI1
      NODD=NODD+2
      IF(ABS(OIKDIF-CABS(IKDIF)).GT.TOL) GO TO 1
      I1K1=S11/(CAMBDA*CMPLX(2.,0.))
      IKDIF=CMPLX(4.,0.)*I1K1-IKDIF/CMPLX(2.,0.)
      RETURN
      END
      SUBROUTINE MINMAX(A,N,AMIN,AMAX)
      DIMENSION A(1)
      AMIN=A(1)
      AMAX=AMIN
      DO 1 I=2,N
      AMIN=AMIN1(AMIN,A(I))
      AMAX=AMAX1(AMAX,A(I))
    1 CONTINUE
      RETURN
      END
      SUBROUTINE NONBLANK(C,NB)
C--DETERMINE NON-BLANK CHAR LENGTH (=NB ON EXIT) OF C*(*)
C  NOTE THAT NB WILL BE IN [0,LEN(C)].
C
      CHARACTER*(*) C
      L=LEN(C)
      DO 10 I=L,1,-1
        NB=I
        IF(C(I:I).NE.' ') RETURN
10    CONTINUE
      NB=0
      RETURN
      END
      SUBROUTINE POLAR2(Z,AMP,PHZ180)
C
C     Z= COMPLEX ARGUMENT OR Z=CMPLX(X,Y)
C     AMP= COMPUTED AMPLITUDE
C     PHZ180= COMPUTED PHASE IN (-180.,180.) DEGREES
C
      COMPLEX Z
      ZR=REAL(Z)
      ZI=AIMAG(Z)
      IF(ZR.EQ.0.0.AND.ZI.EQ.0.0) GO TO 9
      AMP=SQRT(ZR*ZR+ZI*ZI)
      PHZ180=57.29577951*ATAN2(ZI,ZR)
      RETURN
9     AMP=0.0
      PHZ180=0.0
      RETURN
      END
      SUBROUTINE PRENAM(INUNIT,ITMP)
C--PRENAM CAN BE CALLED PRIOR TO READ(ITMP,NAME,...) TO SHIFT ALL
C  NAMELIST INPUT $NAME ... FROM COL.1 AND BEYOND ON INUNIT TO
C  NAMELIST INPUT $NAME ... FROM COL.2 AND BEYOND ON ITMP (UNIT=ITMP
C  IS DELETED AFTER CLOSING OR END OF PROCESS).  NOTE ITMP MAY BE
C  ANY UNIT NUMBER NOT BEING USED (BUT CANNOT BE INUNIT OR 6).
C
C--USAGE:
C
C     NAMELIST/ANYNAME/...
C     ...
C     CALL PRENAM(5,1)
C     ...
C     READ(1,ANYNAME,END=99,ERR=999)
C     ...
C
C--NOTE:  BECAUSE EARLIER VERSIONS (<3.0) OF VAX-11 FORTRAN-77
C     DID NOT HAVE NAMELIST AVAILABLE, A SIMULATED CALL NAMELIST WAS
C     USED BY MANY USERS.  IN PARTICULAR, W.L.ANDERSON USED A
C     SIMULATED CALL NAMELIST(INUNIT,'$ANYNAM',*99) SUBROUTINE WHICH
C     COULD CONTAIN $ANYNAM LISTS BEGINNING IN COL.1 TO 80;  BUT
```

```
C     SINCE VERSION 3.0 OF VAX-11 FORTRAN-77 REQUIRES THE INPUT
C     $ANYNAM LIST TO BEGIN IN COL.2 OR BEYOND, SUBROUTINE PRENAM
C     CAN BE USED ONCE TO MEET THIS REQUIREMENT, AND BECOMES
C     TRANSPARENT TO THE END USER'S INPUT FILE PREPARATION (COL.1-ON)
C
      CHARACTER*200 C
      IF(ITMP.EQ.6.OR.ITMP.EQ.INUNIT) CALL ERRMSG(
     1 '{PRENAM}: ITMP=6 OR ITMP=INUNIT VIOLATION',0,6,0)
      OPEN(UNIT=ITMP,STATUS='SCRATCH',FILE='PRENAM.TMP',ERR=999)
      NONAME=0
10    READ(INUNIT,20,END=99,ERR=888) C
20    FORMAT(A)
      CALL NONBLANK(C,NC)
      IF(NC.EQ.0) NC=1
      IF(NONAME.EQ.0) THEN
        I=INDEX(C,'$')
        IF(I.EQ.0) THEN
          WRITE(ITMP,30) C
30        FORMAT(A<NC>)
        ELSE
          NONAME=1
          WRITE(ITMP,40) C(I:NC)
40        FORMAT(1X,A)
          I=INDEX(C(I+1:NC),'$')
          IF(I.NE.0) NONAME=0
        ENDIF
      ELSE
        WRITE(ITMP,40) C(1:NC)
        I=INDEX(C,'$')
        IF(I.NE.0) NONAME=0
      ENDIF
      GO TO 10
99    REWIND ITMP
      RETURN
888   CALL ERRMSG('{PRENAM}: ERROR IN READING INUNIT',0,6,0)
999   CALL ERRMSG('{PRENAM}: CANNOT OPEN UNIT=ITMP',0,6,0)
      END
      SUBROUTINE PROCINFO(ABS_VALUES,INCR_VALUES)
C
C** SUBROUTINE TO OBTAIN ABSOLUTE AND INCREMENTAL VALUES OF PROCESS
C   PARAMETERS: CPU TIME, BUFFERED I/O COUNT, DIRECT I/O COUNT, AND
C   PAGE FAULTS.
C
      IMPLICIT INTEGER*2(W),INTEGER*4(L)
      PARAMETER (JPI$_CPUTIM = '00000407'X,
     1 JPI$_BUFIO = '0000040C'X,JPI$_DIRIO = '0000040B'X,
     2 JPI$_PAGEFLTS= '0000040A'X)
      INTEGER*4 ABS_VALUES(4),INCR_VALUES(4),LCL_VALUES(4)
      COMMON/ITEMLIST/
     1 W_LEN1,W_CODE1,L_ADDR1,L_LENADDR1,
     2 W_LEN2,W_CODE2,L_ADDR2,L_LENADDR2,
     3 W_LEN3,W_CODE3,L_ADDR3,L_LENADDR3,
     4 W_LEN4,W_CODE4,L_ADDR4,L_LENADDR4,
     5 W_LEN5,W_CODE5
      DATA W_LEN1,W_LEN2,W_LEN3,W_LEN4,W_LEN5/5*4/
      DATA W_CODE1/JPI$_CPUTIM/,
     1 W_CODE2/JPI$_BUFIO/,
     2 W_CODE3/JPI$_DIRIO/,
     3 W_CODE4/JPI$_PAGEFLTS/,
     4 W_CODE5/0/
      DATA L_LENADDR1,L_LENADDR2,L_LENADDR3,L_LENADDR4/4*0/
      L_ADDR1=%LOC(LCL_VALUES(1))
      L_ADDR2=%LOC(LCL_VALUES(2))
      L_ADDR3=%LOC(LCL_VALUES(3))
      L_ADDR4=%LOC(LCL_VALUES(4))
C** PERFORM THE SYSTEM SERVICE CALL
      CALL SYS$GETJPI(,,,W_LEN1,,,)
C** ASSIGN THE NEW VALUES TO THE ARGUMENTS
      DO I=1,4
        INCR_VALUES(I)=LCL_VALUES(I)-ABS_VALUES(I)
        ABS_VALUES(I)=LCL_VALUES(I)
      END DO
      RETURN
      END
      SUBROUTINE SPLIN1(M,H,X,Y,A,B,C,IT,D,P,S)
C--ONE DIMENSIONAL CUBIC SPLINE COEFFICIENT DETERMINATION.
C
C        BY  W.L.ANDERSON, U.S. GEOLOGICAL SURVEY, DENVER, COLORADO
C
C  PARMS--- M= NUMBER OF DATA POINTS .GT. 2
C           H= EQUAL INTERVAL OPTION WHEN H.GT.0. (USE DUMMY X HERE),
C              UNEQUAL INTERVALS IF H=0. (X REQUIRED STORAGE)
C           X= INDEP.VAR WHEN H=0. (DIM .GE. M).
C           Y= DEPENDENT VARIABLE  (DIM .GE. M).
C           A,B,C=COEFF.ARRAYS (EACH DIM .GE. M)
C                 RESULTS ARE RETURNED IN 1ST(M-1) ELEMENTS OF A,B,&C.
C                 ALSO USED AS WORK ARRAYS DURING EXECUTION.
C           IT= TYPE OF BOUNDARY CONDITION SUPPLIED IN D ARRAY. USE
C              IT=1 IF 1ST DERIVATIVES GIVEN AT END POINTS, OR
C              IT=0 IF 2ND DERIVATIVES GIVEN AT END POINTS.
C           D= BOUNDARY ARRAY (DIM 2) AT POINT 1 AND M RESPECTIVELY.
C           P,S= WORK ARRAYS (EACH DIM=M).
C--ERROR RETURN WITH M=-(ABS(M)) IF ANY PARM OUT OF RANGE.
C  THE RESULTING CUBIC SPLINE IS OF THE FORM:
C     Y=Y(I)+A(I)*(X-X(I))+B(I)*(X-X(I))**2+C(I)*(X-X(I))**3
C       FOR I=1,2,...,M-1
C
C
      REAL*4  X(1),Y(1),A(1),B(1),C(1),D(2),P(1),S(1),MUL
      IF(IT.LT.0.OR.IT.GT.1.OR.H.LT.0..OR.M.LT.3) GO TO 999
      N=M-1
      IF(IT.EQ.0) GO TO 20
C--1ST DERIVATIVE BOUNDARIES GIVEN
      NE=N-1
      IF(H) 999,11,1
C--EQUAL SPACING H .GT. 0. AND IT=1
    1 HH=3.0/H
      DO 2 I=1,NE
      B(I)=4.0
      C(I)=1.0
      A(I)=1.0
    2 P(I)=HH*(Y(I+2)-Y(I))
      P(1)=P(1)-D(1)
      P(NE)=P(NE)-D(2)
C--SOLUTION OF TRIDIAGONAL MATRIX EQ. OF ORDER NE
    3 C(1)=C(1)/B(1)
      P(1)=P(1)/B(1)
      DO 4 I=2,NE
      MUL=1.0/(B(I)-A(I)*C(I-1))
      C(I)=MUL*C(I)
    4 P(I)=MUL*(P(I)-A(I)*P(I-1))
C--OBTAIN SPLINE COEFFICIENTS
      A(NE+IT)=P(NE)
      I=NE-1
    5 A(I+IT)=P(I)-C(I)*A(I+IT+1)
      I=I-1
       IF(I.GE.1) GO TO 5
      IF(IT.EQ.0) GO TO 6
      A(1)=D(1)
      A(M)=D(2)
    6 IF(H.EQ.0.) GO TO 14
      HH=1.0/H
      DO 7 I=1,N
      MUL=HH*(Y(I+1)-Y(I))
      B(I)=HH*(3.0*MUL-(A(I+1)+2.0*A(I)))
    7 C(I)=HH*HH*(-2.0*MUL+A(I+1)+A(I))
      RETURN
C--UNEQUAL SPACING H=0.. AND IT=1
   11 DO 12 I=1,N
   12 S(I+1)=X(I+1)-X(I)
      DO 13 I=1,NE
      B(I)=2.0*(S(I+1)+S(I+2))
      C(I)=S(I+1)
      A(I)=S(I+2)
   13 P(I)=3.0*(S(I+1)**2*(Y(I+2)-Y(I+1))+S(I+2)**2*(Y(I+1)-Y(I)))/
     $ (S(I+1)*S(I+2))
      P(1)=P(1)-S(3)*D(1)
      P(NE)=P(NE)-S(N)*D(2)
      GO TO 3
   14 DO 15 I=1,N
      HH=1.0/S(I+1)
      MUL=(Y(I+1)-Y(I))*HH**2
      B(I)=3.0*MUL-(A(I+1)+2.0*A(I))*HH
   15 C(I)=-2.0*MUL*HH+(A(I+1)+A(I))*HH**2
      RETURN
C--2ND DERIVATIVE BOUNDARIES GIVEN
   20 NE=N+1
      IF(H) 999,31,21
C--EQUAL SPACING H .GT. 0 AND IT=0
   21 HH=3.0/H
      DO 22 I=2,N
      B(I)=4.0
      C(I)=1.0
      A(I)=1.0
   22 P(I)=HH*(Y(I+1)-Y(I-1))
```

```
      B(1)=2.0
      B(NE)=2.0
      C(1)=1.0
      C(NE)=1.0
      A(NE)=1.0
      P(1)=HH*(Y(2)-Y(1))-0.5*H*D(1)
      P(NE)=HH*(Y(M)-Y(N))+0.5*H*D(2)
      GO TO 3
C--UNEQUAL SPACING H=0 AND IT=0
   31 DO 32 I=1,N
   32 S(I+1)=X(I+1)-X(I)
      N1=N-1
      DO 33 I=1,N1
      B(I+1)=2.0*(S(I+1)+S(I+2))
      C(I+1)=S(I+1)
      A(I+1)=S(I+2)
   33 P(I+1)=3.0*(S(I+1)**2*(Y(I+2)-Y(I+1))+S(I+2)**2*(Y(I+1)-Y(I)))/
     *     (S(I+1)*S(I+2))
      B(1)=2.0
      B(NE)=2.0
      C(1)=1.0
      C(NE)=1.0
      A(NE)=1.0
      P(1)=3.0*(Y(2)-Y(1))/S(2)-0.5*S(2)*D(1)
      P(NE)=3.0*(Y(M)-Y(N))/S(M)+0.5*S(M)*D(2)
      GO TO 3
  999 M=-IABS(M)
      RETURN
      END
      SUBROUTINE SPOINT(M,X,Y,A,B,C,XX,YY)
C--GIVEN CUBIC SPLINE COEFF'S A,B,C,AND M OBS.DATA ARRAYS X,Y
C  SPOINT EVALUATES THE PIECEWISE CUBIC SPLINE ORDINATE YY AT THE
C  ABSCISSA XX, WHERE XX IS IN THE CLOSED INTERVAL (X(1),X(M)).
C  NOTE: IF COMPUTING OVER EQUAL INTERVALS, USE THE SUBR 'CUBIC'
C  WHICH REQUIRES ONLY ONE CALL.
C
      DIMENSION X(1),Y(1),A(1),B(1),C(1)
      IF(XX.LT.X(1).OR.XX.GT.X(M)) GO TO 9
      M1=M-1
      DO 1 I=1,M1
      J=I
      IF(XX.LE.X(I+1)) GO TO 2
    1 CONTINUE
    9 WRITE(6,60) XX,X(1),X(M)
   60 FORMAT('0ERROR IN SPOINT CALL--XX=',E16.8,' NOT IN CLOSED INTERVAL
     * (',E16.8,',',E16.8,')')
      RETURN
    2 Z=XX-X(J)
      YY=Y(J)+((C(J)*Z+B(J))*Z+A(J))*Z
      RETURN
      END
      SUBROUTINE ZARRAY(MODE,I,J,N,M,S,D)
C--CONVERT SINGLE TO DOUBLE DIM COMPLEX ARRAY (OR VICE VERSA).
C  (IBM SSP MANUAL, P.98 SAME AS THIS BUT IN COMPLEX)
C
```

```
C  MODE=1 TO CONVERT S TO D ARRAYS (SINGLE TO DOUBLE)
C      =2 TO CONVERT D TO S ARRAYS (DOUBLE TO SINGLE)
C         NOTE: S AND D MAY BE THE SAME LOCATION ARRAYS
C      I=NO. OF ROWS IN ACTUAL 2D MATRIX
C      J=NO. OF COLS IN ACTUAL 2D MATRIX
C      N=NO. OF ROWS IN DIM FOR D(N,M) IN CALLING PROGRAM
C      M=NO. OF COLS IN DIM FOR D(N,M) IN CALLING PROGRAM
C      S= COMPLEX VECTOR USED AS S(N*M) BUT ACTUAL IS S(I,J)
C      D= COMPLEX MATRIX DIMENSIONED AS D(N,M)
C
      COMPLEX S(1),D(1)
      NI=N-I
      IF(MODE-1) 100,100,120
C--CONVERT FROM SINGLE TO DOUBLE DIMENSION
100   IJ=I*J+1
      NM=N*J+1
      DO 110 K=1,J
      NM=NM-NI
      DO 110 L=1,I
      IJ=IJ-1
      NM=NM-1
110   D(NM)=S(IJ)
      GO TO 140
C--CONVERT FROM DOUBLE TO SINGLE DIMENSION
120   IJ=0
      NM=0
      DO 130 K=1,J
      DO 125 L=1,I
      IJ=IJ+1
      NM=NM+1
125   S(IJ)=D(NM)
130   NM=NM+NI
140   RETURN
      END
      COMPLEX FUNCTION ZSUBA1(A, B, EPSIL, NPTS, ICHECK, RELERR, F,MEV)
      COMPLEX RELERR,F,RESULT,ESTIM,COMP
C   THIS FUNCTION ROUTINE PERFORMS AUTOMATIC INTEGRATION
C OVER A FINITE INTERVAL USING THE BASIC INTEGRATION
C ALGORITHM ZQUAD1 TOGETHER WITH,  IF NECESSARY AN ADAPTIVE
C SUBDIVISION PROCESS.  IT IS GENERALLY MORE EFFICIENT THAN
C THE NON-ADAPTIVE ALGORITHM ZSUB1 BUT IS LIKELY TO BE LESS
C RELIABLE(SEE COMP.J.,14,189,1971).
      DIMENSION RESULT(8), STACK(100)
      EXTERNAL F
      DATA ISMAX/100/
      CALL ZQUAD1(A, B, RESULT, K, EPSIL, NPTS, ICHECK, F,MEV)
      ZSUBA1 = RESULT(K)
      RELERR = (0.0,0.0)
      IF(REAL(ZSUBA1).NE.0.0.AND.AIMAG(ZSUBA1).NE.0.0) RELERR=
     $ CMPLX(ABS(REAL(RESULT(K)-RESULT(K-1)))/REAL(ZSUBA1),
     $ ABS(AIMAG(RESULT(K)-RESULT(K-1)))/AIMAG(ZSUBA1))
C CHECK IF SUBDIVISION IS NEEDED
      IF (ICHECK.EQ.0) RETURN
C SUBDIVIDE
      ESTIM=ZSUBA1*EPSIL
      ESTIM=CMPLX(ABS(REAL(ESTIM)),ABS(AIMAG(ESTIM)))
      RELERR = (0.0,0.0)
      ZSUBA1 = (0.0,0.0)
      IS = 1
      IC = 1
      SUB1 = A
      SUB3 = B
   10 SUB2 = (SUB1+SUB3)*0.5
      CALL  ZQUAD1(SUB1, SUB2, RESULT, K, EPSIL, NF, ICHECK, F,MEV)
      NPTS = NPTS + NF
      IF(NPTS.GE.MEV) GO TO 50
      COMP = (RESULT(K)-RESULT(K-1))
      COMP=CMPLX(ABS(REAL(COMP)),ABS(AIMAG(COMP)))
      IF (ICHECK.EQ.0) GO TO 30
      IF(REAL(COMP).LE.REAL(ESTIM).AND.
     $ AIMAG(COMP).LE.AIMAG(ESTIM)) GO TO 70
      IF (IS.GE.ISMAX) GO TO 20
C STACK SUBINTERVAL (SUB1,SUB2) FOR FUTURE EXAMINATION
      STACK(IS) = SUB1
      IS = IS + 1
      STACK(IS) = SUB2
      IS = IS + 1
      GO TO 40
   20 IC = -IABS(IC)
   30 ZSUBA1 = ZSUBA1 + RESULT(K)
      RELERR = RELERR + COMP
   40 CALL ZQUAD1(SUB2, SUB3, RESULT, K, EPSIL, NF, ICHECK, F,MEV)
      NPTS = NPTS + NF
      IF(NPTS.GE.MEV) GO TO 50
      COMP = (RESULT(K)-RESULT(K-1))
      COMP=CMPLX(ABS(REAL(COMP)),ABS(AIMAG(COMP)))
      IF (ICHECK.EQ.0) GO TO 50
      IF(REAL(COMP).LE.REAL(ESTIM).AND.
     $ AIMAG(COMP).LE.AIMAG(ESTIM)) GO TO 80
C SUBDIVIDE INTERVAL (SUB2,SUB3)
      SUB1 = SUB2
      GO TO 10
   50 ZSUBA1 = ZSUBA1 + RESULT(K)
      RELERR = RELERR + COMP
      IF(NPTS.GE.MEV) RETURN
      IF (IS.EQ.1) GO TO 60
C SUBDIVIDE THE DELINQUENT INTERVAL LAST STACKED
      IS = IS - 1
      SUB3 = STACK(IS)
      IS = IS - 1
      SUB1 = STACK(IS)
      GO TO 10
C SUBDIVISION RESULT
   60 ICHECK = IC
      IF(REAL(ZSUBA1).EQ.0.0) GO TO 62
      IF(AIMAG(ZSUBA1).EQ.0.0) GO TO 64
      RELERR=CMPLX(REAL(RELERR)/ABS(REAL(ZSUBA1)),
     $ AIMAG(RELERR)/ABS(AIMAG(ZSUBA1)))
      RETURN
   62 IF(AIMAG(ZSUBA1).EQ.0.0) GO TO 66
```

```
      RELERR=CMPLX(0.0,AIMAG(RELERR)/ABS(AIMAG(ZSUBA1)))
      RETURN
   64 RELERR=CMPLX(REAL(RELERR)/ABS(REAL(ZSUBA1)),0.0)
      RETURN
   66 RELERR=(0.0,0.0)
      RETURN
C RELAXED CONVERGENCE
   70 IC = ISIGN(2,IC)
      GO TO 30
   80 IC = ISIGN(2,IC)
      GO TO 50
      END
      SUBROUTINE KELVIN(X,M,B)
C--COMPUTES M(.LE.8) KELVIN FUNCTIONS (ORDERS 0,1) CONSECUTIVELY STORED
C  IN ARRAY B(M) WHERE:
C
C     X      = DP-ARGUMENT .GT. 0.0D0 (ASYMPTOTIC FORM USED IF X.GE.8.0)
C     M      = NUMBER OF B'S TO COMPUTE AS DEFINED BELOW (1.GE.M.LE.8)
C     B(M)   = COMPUTED DP-FUNCTIONS WHERE B IS DEFINED:
C             B(1) = BER(X)  --  ORDER 0
C             B(2) = BEI(X)  --  ORDER 0
C             B(3) = KER(X)  --  ORDER 0
C             B(4) = KEI(X)  --  ORDER 0
C             B(5) = BER1(X)  --  ORDER 1
C             B(6) = BEI1(X)  --  ORDER 1
C             B(7) = KER1(X)  --  ORDER 1
C             B(8) = KEI1(X)  --  ORDER 1
C ** ACCURACY GOOD TO AT LEAST 14 FIGURES FOR ALL X **
C NOTE: THIS METHOD OF GENERATING MULTIPLE KELVIN FUNCTIONS WAS CHOSEN
C       TO REDUCE TOTAL CPU-TIME SINCE MOST APPLICATIONS REQUIRE
C       MULTIPLE FUNCTION USE AND IS THEREFORE ACCOMPLISHED BY ONE CALL.
C E.G: TO OBTAIN BER(X),BEI(X),KER(X), AND KEI(X):  CALL KELVIN(X,4,B)
C  IF X OR M OUT OF RANGE, ROUTINE EXITS WITHOUT ACTION.
C
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 B(8),CN(8),SN(8)
      DATA CN          /.7071067811865475D0,0.D0,-.7071067811865475D0,
     * -1.D0,-.7071067811865475D0,0.D0,.7071067811865475D0,1.D0/,
     * SN   /.7071067811865475D0,1.D0,.7071067811865475D0,0.D0,
     * -.7071067811865475D0,-1.D0,-.7071067811865475D0,0.D0/
      DATA PI4/.7853981633974483D0/,R22/.7071067811865475D0/,
     * E/0.5D-14/,
     * PI1/.3183098861837907D0/
      IF(M.LT.1.OR.M.GT.8.OR.X.LE.0.0D0) GO TO 9
      IF(X.GE.8.0D0) GO TO 8
C--SERIES METHODS (X.GT.0.0.AND.X.LT.8.0D0)
      X2=0.5D0*X
      X4=X2**4
      T1=-0.25D0*X4
      S1=T1
      T2=0.0D0
      T3=0.0D0
      T4=0.0D0
      T15=0.0D0
      T26=0.0D0
      T75=0.0D0
      T86=0.0D0
      IF(M.EQ.1) GO TO 100
      T2=X2**2
      S2=T2
      IF(M.EQ.2) GO TO 100
      T5=1.5D0
      S5=T1*T5
      IF(M.EQ.3) GO TO 100
      T6=1.0D0
      S6=T2
      IF(M.EQ.4) GO TO 100
      T3=-0.5D0*X2**3
      S3=T3
      T4=X2
      S4=T4
      IF(M.LE.6) GO TO 100
      T7=-0.25D0*X2**3
      S7=2.0D0*T7*T5
      T8=X2
      S8=T8
  100 TK=2.0D0
  101 TK2=TK+TK
      TK21=TK2-1.0D0
      TK22=TK2-2.0D0
      RK2=1.0D0/TK2
      RK21=1.0D0/TK21
      RK22=1.0D0/TK22
      R1=-X4*(RK21*RK2)**2
      T1=T1*R1
      S1=S1+T1
      IF(M.EQ.1) GO TO 200
      R2=-X4*(RK22*RK21)**2
      T2=T2*R2
      S2=S2+T2
      IF(M.EQ.2) GO TO 200
      T5=T5+RK21+RK2
      T15=T1*T5
      S5=S5+T15
      IF(M.EQ.3) GO TO 200
      T6=T6+RK22+RK21
      T26=T2*T6
      S6=S6+T26
      IF(M.EQ.4) GO TO 200
      T3=T3*(-X4*(RK22*RK21**2*RK2))
      S3=S3+T3
      T4=T4*(-X4*RK22**2*RK21/(TK2-3.0D0))
      S4=S4+T4
      IF(M.LE.6) GO TO 200
      T7=T7*R1
      T75=TK2*T7*T5
      S7=S7+T75
      T8=T8*R2
      T86=TK21*T8*T6
      S8=S8+T86
  200 TK=TK+1.0D0
      IF(DABS(T1).GT.E.OR.DABS(T2).GT.E.OR.DABS(T15).GT.E.OR.
     * DABS(T26).GT.E.OR.DABS(T3).GT.E.OR.DABS(T4).GT.E.OR.
     * DABS(T75).GT.E.OR.DABS(T86).GT.E) GO TO 101
      B(1)=1.0D0+S1
      IF(M.EQ.1) GO TO 9
      B(2)=S2
      IF(M.EQ.2) GO TO 9
      C=0.11593151565841240D0-DLOG(X)
      B(3)=C*B(1)+PI4*B(2)+S5
      IF(M.EQ.3) GO TO 9
      B(4)=C*B(2)-PI4*B(1)+S6
      IF(M.EQ.4) GO TO 9
      B(5)=R22*(S3-S4)
      IF(M.EQ.5) GO TO 9
      B(6)=R22*(S3+S4)
      IF(M.EQ.6) GO TO 9
      S7=C*S3-B(1)/X+PI4*S4+S7
      S8=C*S4-B(2)/X-PI4*S3+S8
      B(7)=R22*(S7-S8)
      IF(M.EQ.7) GO TO 9
      B(8)=R22*(S7+S8)
    9 RETURN
C--GENERAL ASYMPTOTIC FORM FOR NU=0,1:
    8 NU=0
      X2=R22*X
      X8=8.0D0*X
      SX=DSQRT(X)
      EX2=DEXP(-X2)
      C1=1.2533141373155000D0*EX2/SX
      C2=1.0D0/(2.5066282746310010D0*SX*EX2+1.0D-38)
      MAXK=30
      IF(X.LT.15.0D0) MAXK=X+X
    1 XNU=NU
      XMU=4.0D0*XNU
      ALP=X2+PI4*(XNU+XNU-0.5D0)
      BETA=ALP+PI4
      CB=DCOS(BETA)
      CA=DCOS(ALP)
      SB=DSIN(BETA)
      SA=DSIN(ALP)
      N4=4*NU
      FM=0.0D0
      FP=0.0D0
      GM=0.0D0
      GP=0.0D0
      TM=1.0D0
      TP=1.0D0
      K=1
    2 TK=K
      T=(XMU-(TK+TK-1.0D0)**2)/(TK*X8)
      TPL=DABS(TP)
      TP=-TP*T
      IF(DABS(TP).GT.TPL) GO TO 21
      TM=TM*T
      N=MOD(K,8)
      IF(N.EQ.0) N=8
      T1=TP*CN(N)
      FP=FP+T1
      T2=TM*CN(N)
      FM=FM+T2
      T3=TP*SN(N)
      GP=GP+T3
      T4=TM*SN(N)
      GM=GM+T4
      K=K+1
      IF(K.GT.MAXK) GO TO 3
      GO TO 2
   21 FP=FP-T1
      FM=FM-T2
      GP=GP-T3
      GM=GM-T4
    3 FP=FP+1.0D0
      FM=FM+1.0D0
      B(N4+4)=C1*(-FM*SB-GM*CB)
      B(N4+3)=C1*(FM*CB-GM*SB)
      B(N4+2)=C2*(FP*SA-GP*CA)+PI1*B(N4+3)
      B(N4+1)=C2*(FP*CA+GP*SA)-PI1*B(N4+4)
      IF(NU.EQ.1.OR.M.LE.4) GO TO 9
      NU=1
      GO TO 1
      END
      SUBROUTINE ZQUAD1(A,B,RESULT,K,EPSIL,NPTS,ICHECK,F,MEV)
      COMPLEX F,RESULT,FUNCT,FZERO,ACUM
      DIMENSION FUNCT(127),P(381),RESULT(8)
      COMMON/ZQUADP/P
C--FOLLOWING CALL ONLY FOR MULTICS SYSTEM:
      CALL ZBLOCK
      ICHECK = 0
C CHECK FOR TRIVIAL CASE.
      IF (A.EQ.B) GO TO 70
C SCALE FACTORS.
      SUM = (B+A)/2.0
      DIFF = (B-A)/2.0
C 1-POINT GAUSS
      FZERO = F(SUM)
      RESULT(1) = 2.0*FZERO*DIFF
      I = 0
      IOLD = 0
      INEW = 1
      K = 2
      ACUM = (0.0,0.0)
      GO TO 30
   10 IF (K.EQ.8) GO TO 50
      IF(INEW+IOLD.GE.MEV) GO TO 60
      K = K + 1
      ACUM = (0.0,0.0)
C CONTRIBUTION FROM FUNCTION VALUES ALREADY COMPUTED.
      DO 20 J=1,IOLD
        I = I + 1
        ACUM = ACUM + P(I)*FUNCT(J)
   20 CONTINUE
C CONTRIBUTION FROM NEW FUNCTION VALUES.
   30 IOLD = IOLD + INEW
      DO 40 J=INEW,IOLD
        I = I + 1
        X = P(I)*DIFF
        FUNCT(J) = F(SUM+X) + F(SUM-X)
        I = I + 1
        ACUM = ACUM + P(I)*FUNCT(J)
   40 CONTINUE
      INEW = IOLD + 1
      I = I + 1
      RESULT(K) = (ACUM+P(I)*FZERO)*DIFF
C CHECK FOR CONVERGENCE.
      IF(ABS(REAL(RESULT(K))-REAL(RESULT(K-1))).LE.EPSIL*
     $ABS(REAL(RESULT(K))).AND.
     $ ABS(AIMAG(RESULT(K))-AIMAG(RESULT(K-1))).LE.EPSIL*
     $ABS(AIMAG(RESULT(K)))) GO TO 60
      GO TO 10
C CONVERGENCE NOT ACHIEVED.
   50 ICHECK = 1
C NORMAL TERMINATION.
   60 NPTS = INEW + IOLD
      RETURN
C TRIVIAL CASE
   70 K = 2
      RESULT(1) = (0.0,0.0)
      RESULT(2) = (0.0,0.0)
      NPTS = 0
      RETURN
      END
C--ZQUAD PACKAGE (ZBLOCK,ZQUAD1,ZSUB1,ZSUBA1,ZQUAD2,ZSUB2,ZSUBA2)
C  FOR AUTOMATIC COMPLEX GAUSSIAN DOUBLE INTEGRATION OVER A
C  FINITE INTERVAL.
C
C--MODIFIED BY W.L.ANDERSON, U.S.GEOLOGICAL SURVEY, DENVER, COLORADO
C  12/30/75.
C
C--USAGE:
C
C  USE "ZSUB1" OR "ZSUBA1" FOR 1ST COMPLEX INTEGRATION (CALLS ZQUAD1)
C  AND "ZSUB2" OR "ZSUBA2" FOR 2ND COMPLEX INTEGRATION (CALLS ZQUAD2)
C
C--REFERENCES:
C
C     (1)       PATTERSON,T.N.L, 1973, ALGORITHM FOR AUTOMATIC
C               NUMERICAL INTEGRATION OVER A FINITE INTERVAL ^D1?:
C               ACM COMM. V.16, NO.11, P.694-699.
C     (2)       ANDERSON,W.L., 1974, ELECTROMAGNETIC FIELDS ABOUT A
C               FINITE ELECTRIC WIRE SOURCE:
C               N.T.I.S REPORT PB-238199, 209P.
C
C--NOTES:
C
```

```
C     (A).  SEE REF(1) FOR A COMPLETE DISCUSSION OF THE BASIC
C               ALGORITHM(S) AS ORIGINALLY DEVELOPED FOR
C               SINGLE REAL FUNCTION AUTOMATIC GAUSSIAN INTEGRATION.
C     (B).  SEE REF(2) FOR A MODIFIED VERSION FOR SINGLE COMPLEX
C               FUNCTION AUTOMATIC GAUSSIAN INTEGRATION.
C     (C).  ALL CALLING PARMS USED BELOW IN THE ZQUAD PACKAGE ARE
C               IDENTICAL TO THOSE USED IN REF(2).  THEREFORE, SEE
C               REF(2) FOR COMMENTS ON THESE ANALOGOUS ROUTINES.
C               REF(1) MAY ALSO BE USED FOR DEFINITIONS OF MOST OF
C               THE PARMS...
C
C-- THE VAX VERSION USES CALL ZBLOCK TO INITILIZE COMMON/ZQUADP.
C   FOR SOME SYSTEMS, ONE MAY CHANGE SUBROUTINE ZBLOCK TO A
C   BLOCK DATA SUBPROGRAM -- AND REMOVE THE ASSIGNMENT STATEMENTS.
C
      SUBROUTINE ZBLOCK
      DIMENSION P(381)
      COMMON/ZQUADP/Q(381)
      DATA IVAX/0/
      DATA
     * P( 1),P( 2),P( 3),P( 4),P( 5),P( 6),P( 7),
     * P( 8),P( 9),P(10),P(11),P(12),P(13),P(14),
     * P(15),P(16),P(17),P(18),P(19),P(20),P(21),
     * P(22),P(23),P(24),P(25),P(26),P(27),P(28)/
     *  0.77459666924148337704E 00,0.55555555555555555556E 00,
     *  0.88888888888888888889E 00,0.26848808986833344073E 00,
     *  0.96049126870802028342E 00,0.10465622602646726519E 00,
     *  0.43424374934680255800E 00,0.40139741477596222291E 00,
     *  0.45091653865847414235E 00,0.13441525524378422036E 00,
     *  0.51603282997079739697E-01,0.20062852937698902103E 00,
     *  0.99383196321275502221E 00,0.17001719629940260339E-01,
     *  0.88845923287225699889E 00,0.92927195315124537686E-01,
     *  0.62110294673722640294E 00,0.17151190913639138079E 00,
     *  0.22338668642896688163E 00,0.21915685840158749640E 00,
     *  0.22551049979820668739E 00,0.67207754295990703540E-01,
     *  0.25807598096176653565E-01,0.10031427861179557877E 00,
     *  0.84345657393211062463E-02,0.46462893261757986541E-01,
     *  0.85755920049990351154E-01,0.10957842105592463824E 00/
      DATA
     * P(29),P(30),P(31),P(32),P(33),P(34),P(35),
     * P(36),P(37),P(38),P(39),P(40),P(41),P(42),
     * P(43),P(44),P(45),P(46),P(47),P(48),P(49),
     * P(50),P(51),P(52),P(53),P(54),P(55),P(56)/
     *  0.99909812496766759766E 00,0.25447807915618744154E-02,
     *  0.98153114955374010687E 00,0.16446049854387810934E-01,
     *  0.92965485742974005667E 00,0.35957103307129322097E-01,
     *  0.83672593816886873550E 00,0.56979509494123357412E-01,
     *  0.70249620649152707861E 00,0.76879620499003531043E-01,
     *  0.53131974364437562397E 00,0.93627109981264473617E-01,
     *  0.33113539325797683309E 00,0.10566989358023480974E 00,
     *  0.11248894313318662575E 00,0.11195687302095345688E 00,
     *  0.11275525672076869161E 00,0.33603877148207730542E-01,
     *  0.12903800100351265626E-01,0.50157139305899537414E-01,
     *  0.42176304415588548391E-02,0.23231446639910269443E-01,
     *  0.42877960025007734493E-01,0.54789210527962865032E-01,
*  0.12651565562300680114E-02,0.82230079572359296693E-02,           00023310
*  0.17978551568128270333E-01,0.28489754745833548613E-01/           00023320
 DATA                                                               00023330
* P(57),P(58),P(59),P(60),P(61),P(62),P(63),                        00023340
* P(64),P(65),P(66),P(67),P(68),P(69),P(70),                        00023350
* P(71),P(72),P(73),P(74),P(75),P(76),P(77),                        00023360
* P(78),P(79),P(80),P(81),P(82),P(83),P(84)/                        00023370
*  0.38439810249455532039E-01,0.46813554990628012403E-01,           00023380
*  0.52834946790116519862E-01,0.55978436510476319408E-01,           00023390
*  0.99987288812035761194E 00,0.36322148184553065969E-03,           00023400
*  0.99720625937222195908E 00,0.25790497946856882724E-02,           00023410
*  0.98868475754742947994E 00,0.61155068221172463397E-02,           00023420
*  0.97218287474858179658E 00,0.10498246909621321898E-01,           00023430
*  0.94634285837340290515E 00,0.15406750466559497802E-01,           00023440
*  0.91037115695700429250E 00,0.20594233915912711149E-01,           00023450
*  0.86390793819369047715E 00,0.25869679327214746911E-01,           00023460
*  0.80694053195021761186E 00,0.31073551111687964880E-01,           00023470
*  0.73975604435269475868E 00,0.36064432780782572640E-01,           00023480
*  0.66290966002478059546E 00,0.40715510116944318934E-01,           00023490
*  0.57719571005204581484E 00,0.44914531653632197414E-01,           00023500
*  0.48361802694584102756E 00,0.48564330406673198716E-01/           00023510
 DATA                                                               00023520
* P( 85),P( 86),P( 87),P( 88),P( 89),P( 90),P( 91),                 00023530
* P( 92),P( 93),P( 94),P( 95),P( 96),P( 97),P( 98),                 00023540
* P( 99),P(100),P(101),P(102),P(103),P(104),P(105),                 00023550
* P(106),P(107),P(108),P(109),P(110),P(111),P(112)/                 00023560
*  0.38335932419873034692E 00,0.51583253952048458777E-01,           00023570
*  0.27774982202182431507E 00,0.53905499335266063927E-01,           00023580
*  0.16823525155220746498E 00,0.55481404356559363988E-01,           00023590
*  0.56344313046592789972E-01,0.56277699831254301273E-01,           00023600
*  0.56377628360384717388E-01,0.16801938574103865271E-01,           00023610
*  0.64519000501757369228E-02,0.25078569652949768707E-01,           00023620
*  0.21088152457266328793E-02,0.11615723319955134727E-01,           00023630
*  0.21438980012503867246E-01,0.27394605263981432516E-01,           00023640
*  0.63260731936263354422E-03,0.41115039786546930472E-02,           00023650
*  0.89892757840641357233E-02,0.14244877372916774306E-01,           00023660
*  0.19219905124727766019E-01,0.23406777495314006201E-01,           00023670
*  0.26417473395058259931E-01,0.27989218255238159704E-01,           00023680
*  0.18073956444538835782E-03,0.12895240826104173921E-02,           00023690
*  0.30577534101755311361E-02,0.52491234548088591251E-02/           00023700
 DATA                                                               00023710
* P(113),P(114),P(115),P(116),P(117),P(118),P(119),                 00023720
* P(120),P(121),P(122),P(123),P(124),P(125),P(126),                 00023730
* P(127),P(128),P(129),P(130),P(131),P(132),P(133),                 00023740
* P(134),P(135),P(136),P(137),P(138),P(139),P(140)/                 00023750
*  0.77033752332797418482E-02,0.10297116957956355524E-01,           00023760
*  0.12934839663607373455E-01,0.15536775555843982440E-01,           00023770
*  0.18032216390391286320E-01,0.20357755058472159467E-01,           00023780
*  0.22457265826816098707E-01,0.24282165203336599358E-01,           00023790
*  0.25791626976024229388E-01,0.26952749667633031963E-01,           00023800
*  0.27740702178279681994E-01,0.28138849915627150636E-01,           00023810
*  0.99998243035489159858E 00,0.50536095207862517625E-04,           00023820
*  0.99959879967191068325E 00,0.37774664632698466027E-03,           00023830
*  0.99831663531840739253E 00,0.93836984854238150079E-03,           00023840
*  0.99572410469840718851E 00,0.16811428654214699063E-02,           00023850
```

```
     *  0.99149572117810613240E 00,0.25687649437940203731E-02,          00023860
     *  0.98537149959852037111E 00,0.35728927835172996494E-02,          00023870
     *  0.97714151463970571416E 00,0.46710503721143217474E-02,          00023880
     *  0.96663785155841656709E 00,0.58434498758356395076E-02/          00023890
      DATA                                                              00023900
     * P(141),P(142),P(143),P(144),P(145),P(146),P(147),                00023910
     * P(148),P(149),P(150),P(151),P(152),P(153),P(154),                00023920
     * P(155),P(156),P(157),P(158),P(159),P(160),P(161),                00023930
     * P(162),P(163),P(164),P(165),P(166),P(167),P(168)/                00023940
     *  0.95373000642576113641E 00,0.70724899954335554680E-02,          00023950
     *  0.93832039777959288365E 00,0.83428387539681577056E-02,          00023960
     *  0.92034002547001242073E 00,0.96411777297025366953E-02,          00023970
     *  0.89974489977694003664E 00,0.10955733387837901648E-01,          00023980
     *  0.87651341448470526974E 00,0.12275830560082770087E-01,          00023990
     *  0.85064449476835027976E 00,0.13591571009765546790E-01,          00024000
     *  0.82215625436498040737E 00,0.14893641664815182035E-01,          00024010
     *  0.79108493379984836143E 00,0.16173218729577719942E-01,          00024020
     *  0.75748396638051363793E 00,0.17421930159464173747E-01,          00024030
     *  0.72142308537009891548E 00,0.18631848256138790186E-01,          00024040
     *  0.68298743109107922809E 00,0.19795495048097499488E-01,          00024050
     *  0.64227664250975951377E 00,0.20905851445812023852E-01,          00024060
     *  0.59940393024224289297E 00,0.21956366305317824939E-01,          00024070
     *  0.55449513263193254887E 00,0.22940964229387748761E-01/          00024080
      DATA                                                              00024090
     * P(169),P(170),P(171),P(172),P(173),P(174),P(175),                00024100
     * P(176),P(177),P(178),P(179),P(180),P(181),P(182),                00024110
     * P(183),P(184),P(185),P(186),P(187),P(188),P(189),                00024120
     * P(190),P(191),P(192),P(193),P(194),P(195),P(196)/                00024130
     *  0.50768775753371660215E 00,0.23854052106038540080E-01,          00024140
     *  0.45913001198983233287E 00,0.24690524744487676909E-01,          00024150
     *  0.40897982122988867241E 00,0.25445769965464765813E-01,          00024160
     *  0.35740383783153215238E 00,0.26115673376706097680E-01,          00024170
     *  0.30457644155671404334E 00,0.26696622927450359906E-01,          00024180
     *  0.25067873030348317661E 00,0.27185513229624791819E-01,          00024190
     *  0.19589750271110015392E 00,0.27579749566481873035E-01,          00024200
     *  0.14042423315256017459E 00,0.27877251476613701609E-01,          00024210
     *  0.84454040083710883710E-01,0.28076455793817246607E-01,          00024220
     *  0.28184648949745694339E-01,0.28176319033016602131E-01,          00024230
     *  0.28188814180192358694E-01,0.84009692870519326354E-02,          00024240
     *  0.32259500250878684614E-02,0.12539284826474884353E-01,          00024250
     *  0.10544076228633167722E-02,0.58078616599775673635E-02,          00024260
     *  0.10719490006251933623E-01,0.13697302631990716258E-01/          00024270
      DATA                                                              00024280
     * P(197),P(198),P(199),P(200),P(201),P(202),P(203),                00024290
     * P(204),P(205),P(206),P(207),P(208),P(209),P(210),                00024300
     * P(211),P(212),P(213),P(214),P(215),P(216),P(217),                00024310
     * P(218),P(219),P(220),P(221),P(222),P(223),P(224)/                00024320
     *  0.31630366082226447689E-03,0.20557519893273465236E-02,          00024330
     *  0.44946378920320678616E-02,0.71224386864583871532E-02,          00024340
     *  0.96099525623638830097E-02,0.11703388747657003101E-01,          00024350
     *  0.13208736697529129966E-01,0.13994609127619079852E-01,          00024360
     *  0.90372734658751149261E-04,0.64476204130572477933E-03,          00024370
     *  0.15288767050877655684E-02,0.26245617274044295626E-02,          00024380
     *  0.38516876166398709241E-02,0.51485584789781777618E-02,          00024390
     *  0.64674198318036867274E-02,0.77683877779219912200E-02,          00024400
     *  0.90161081951956431600E-02,0.10178877529236079733E-01,          00024410
     *  0.11228632913408049354E-01,0.12141082601668299679E-01,          00024420
     *  0.12895813488012114694E-01,0.13476374833816515982E-01,          00024430
     *  0.13870351089139840997E-01,0.14069424957813575318E-01,          00024440
     *  0.25157870384280661489E-04,0.18887326450650491366E-03,          00024450
     *  0.46918492424785040975E-03,0.84057143271072246365E-03/          00024460
      DATA                                                              00024470
     * P(225),P(226),P(227),P(228),P(229),P(230),P(231),                00024480
     * P(232),P(233),P(234),P(235),P(236),P(237),P(238),                00024490
     * P(239),P(240),P(241),P(242),P(243),P(244),P(245),                00024500
     * P(246),P(247),P(248),P(249),P(250),P(251),P(252)/                00024510
     *  0.12843824718970101768E-02,0.17864463917586498247E-02,          00024520
     *  0.23355251860571608737E-02,0.29217249379178197538E-02,          00024530
     *  0.35362449977167777340E-02,0.41714193769840788528E-02,          00024540
     *  0.48205888648512683476E-02,0.54778666939189508240E-02,          00024550
     *  0.61379152800413850435E-02,0.67957855048827733948E-02,          00024560
     *  0.74468208324075910174E-02,0.80866093647888599710E-02,          00024570
     *  0.87109650797320868736E-02,0.93159241280693950932E-02,          00024580
     *  0.98977475240487497440E-02,0.10452925722906011926E-01,          00024590
     *  0.10978183152658912470E-01,0.11470482114693874380E-01,          00024600
     *  0.11927026053019270040E-01,0.12345262372243838455E-01,          00024610
     *  0.12722884982732382906E-01,0.13057836688353048840E-01,          00024620
     *  0.13348311463725179953E-01,0.13592756614812395910E-01,          00024630
     *  0.13789874783240936517E-01,0.13938625738306850804E-01,          00024640
     *  0.14038227896908623303E-01,0.14088159516508301065E-01/          00024650
      DATA                                                              00024660
     * P(253),P(254),P(255),P(256),P(257),P(258),P(259),                00024670
     * P(260),P(261),P(262),P(263),P(264),P(265),P(266),                00024680
     * P(267),P(268),P(269),P(270),P(271),P(272),P(273),                00024690
     * P(274),P(275),P(276),P(277),P(278),P(279),P(280)/                00024700
     *  0.99999759637974846462E 00,0.69379364324108267170E-05,          00024710
     *  0.99994399620705437576E 00,0.53275293669780613125E-04,          00024720
     *  0.99976049092443204733E 00,0.13575491094922871973E-03,          00024730
     *  0.99938033802502358193E 00,0.24921240048299729402E-03,          00024740
     *  0.99874561446809511470E 00,0.38974528447328229322E-03,          00024750
     *  0.99780535449595727456E 00,0.55429531493037471492E-03,          00024760
     *  0.99651414591489027385E 00,0.74028280424450333046E-03,          00024770
     *  0.99483150280062100052E 00,0.94536151685852538246E-03,          00024780
     *  0.99272134428278861533E 00,0.11674841174299594077E-02,          00024790
     *  0.99015137040077015918E 00,0.14049079956551446427E-02,          00024800
     *  0.98709252795403406719E 00,0.16561127281544526052E-02,          00024810
     *  0.98351865757863272876E 00,0.19197129710138724125E-02,          00024820
     *  0.97940628167086268381E 00,0.21944069253638388388E-02,          00024830
     *  0.97473445975240266776E 00,0.24789582266575679307E-02/          00024840
      DATA                                                              00024850
     * P(281),P(282),P(283),P(284),P(285),P(286),P(287),                00024860
     * P(288),P(289),P(290),P(291),P(292),P(293),P(294),                00024870
     * P(295),P(296),P(297),P(298),P(299),P(300),P(301),                00024880
     * P(302),P(303),P(304),P(305),P(306),P(307),P(308)/                00024890
     *  0.96948465950245923177E 00,0.27721957645934509940E-02,          00024900
     *  0.96364062156981213252E 00,0.30730184347025783234E-02,          00024910
     *  0.95718821610986096274E 00,0.33803979910869203823E-02,          00024920
     *  0.95011529752129487656E 00,0.36933779170256508183E-02,          00024930
     *  0.94241156519108305981E 00,0.40110687240750233989E-02,          00024940
     *  0.93406843615772578800E 00,0.43326409680929828545E-02,          00024950
     *  0.92507893290707565236E 00,0.46573172997568547773E-02,          00024960
     *  0.91543758715576504064E 00,0.49843645647655386012E-02,          00024970
     *  0.90514035881326159519E 00,0.53130866051870565663E-02,          00024980
     *  0.89418456833555902286E 00,0.56428181013844441585E-02,          00024990
     *  0.88256884024734190684E 00,0.59729195655081658049E-02,          00025000
     *  0.87029305554811390585E 00,0.63027734490857587172E-02,          00025010
     *  0.85735831088623215653E 00,0.66317812429018878941E-02,          00025020
     *  0.84376688267270860104E 00,0.69593614093904229394E-02/          00025030
      DATA                                                              00025040
     * P(309),P(310),P(311),P(312),P(313),P(314),P(315),                00025050
     * P(316),P(317),P(318),P(319),P(320),P(321),P(322),                00025060
     * P(323),P(324),P(325),P(326),P(327),P(328),P(329),                00025070
     * P(330),P(331),P(332),P(333),P(334),P(335),P(336)/                00025080
     *  0.82952219463740140018E 00,0.72849479805538070639E-02,          00025090
     *  0.81462878765513741344E 00,0.76079896657190565832E-02,          00025100
     *  0.79909229096084140180E 00,0.79279493342948491103E-02,          00025110
     *  0.78291939411828301639E 00,0.82443037630328680306E-02,          00025120
     *  0.76611781930376009072E 00,0.85565435613076896192E-02,          00025130
     *  0.74869629361693660282E 00,0.88641732094824942641E-02,          00025140
     *  0.73066452124218126133E 00,0.91667111635607884067E-02,          00025150
     *  0.71203315536225203459E 00,0.94636899938300652943E-02,          00025160
     *  0.69281376977911470289E 00,0.97546565363174114611E-02,          00025170
     *  0.67301883023041847920E 00,0.10039172044056840798E-01,          00025180
     *  0.65266166541001749610E 00,0.10316812330947621682E-01,          00025190
     *  0.63175643771119423041E 00,0.10587167904885197931E-01,          00025200
     *  0.61031811371518640016E 00,0.10849844089337314099E-01,          00025210
     *  0.58836243444766254143E 00,0.11104461134006926537E-01/          00025220
      DATA                                                              00025230
     * P(337),P(338),P(339),P(340),P(341),P(342),P(343),                00025240
     * P(344),P(345),P(346),P(347),P(348),P(349),P(350),                00025250
     * P(351),P(352),P(353),P(354),P(355),P(356),P(357),                00025260
     * P(358),P(359),P(360),P(361),P(362),P(363),P(364)/                00025270
     *  0.56590588542365442262E 00,0.11350654315980596602E-01,          00025280
     *  0.54296566649831149049E 00,0.11588074033043952568E-01,          00025290
     *  0.51955966153745702199E 00,0.11816385890830235763E-01,          00025300
     *  0.49570640791876146017E 00,0.12035270785279562630E-01,          00025310
     *  0.47142506587165887693E 00,0.12244424981611985899E-01,          00025320
     *  0.44673538766202847374E 00,0.12443560190714035263E-01,          00025330
     *  0.42165768662616330006E 00,0.12632403643542078765E-01,          00025340
     *  0.39621280605761593918E 00,0.12810698163877361967E-01,          00025350
     *  0.37042208795007823014E 00,0.12978202239537399286E-01,          00025360
     *  0.34430734159943802278E 00,0.13134690091960152836E-01,          00025370
     *  0.31789081206847668318E 00,0.13279951743930530650E-01,          00025380
     *  0.29119514851824668196E 00,0.13413793085110098513E-01,          00025390
     *  0.26424337241092676194E 00,0.13536035934956213614E-01,          00025400
     *  0.23705884558982972721E 00,0.13646518102571291428E-01/          00025410
      DATA                                                              00025420
     * P(365),P(366),P(367),P(368),P(369),P(370),P(371),                00025430
     * P(372),P(373),P(374),P(375),P(376),P(377),P(378),                00025440
     * P(379),P(380),P(381)/                                            00025450
     *  0.20966523824318119477E 00,0.13745093443001896632E-01,          00025460
     *  0.18208649675925219825E 00,0.13831631909506428676E-01,          00025470
     *  0.15434681148137810869E 00,0.13906019601325461264E-01,          00025480
     *  0.12647058437230196685E 00,0.13968158806516938516E-01,          00025490
     *  0.98482396598119202090E-01,0.14017968039456608810E-01,          00025500
```

```
      *  0.70406976042855179063E-01,0.14055382072649964277E-01,
      *  0.42269164765363603212E-01,0.14080351962553661325E-01,
      *  0.14093886410782462614E-01,0.14092845069160408355E-01,
      *  0.14094407090096179347E-01/
       IF(IVAX.EQ.1) RETURN
       DO 1 I=1,381
1      Q(I)=P(I)
       IVAX=1
       RETURN
       END
$
```