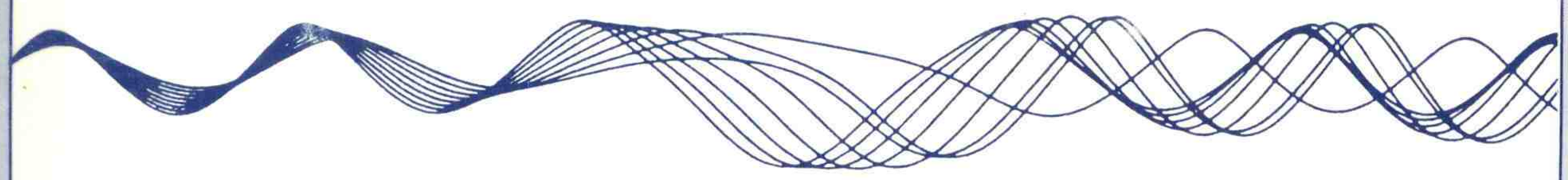
HARPO

A Versatile Three-Dimensional Hamiltonian Ray-Tracing Program for Acoustic Waves in an Ocean with Irregular Bottom

R. Michael Jones
J.P. Riley
T.M. Georges



U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration
Environmental Research Laboratories

QC
Z43
.J67
1986

HARPO

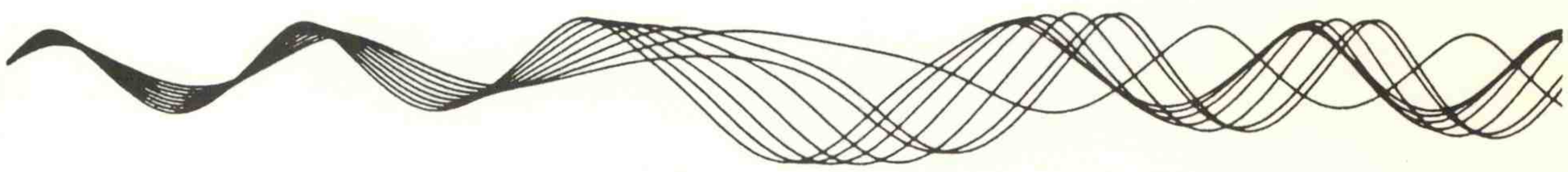
A Versatile Three-Dimensional Hamiltonian Ray-Tracing Program for Acoustic Waves in an Ocean with Irregular Bottom

R. Michael Jones
J.P. Riley
T.M. Georges

Wave Propagation Laboratory
Boulder, Colorado

October 1986

LIBRARY
SEP 27 1988
N.O.A.A.
U.S. Dept. of Commerce



U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration
Environmental Research Laboratories

NOTICES

Mention of a commercial company or product does not constitute an endorsement by NOAA Environmental Research Laboratories. Use for publicity or advertising purposes of information from this publication concerning proprietary products or the tests of such products is not authorized.

The program documented in this report is offered "as is," with no performance guarantees. The authors make no claim that it is suitable for any particular application, or that it will run on any computing system without modification. If you encounter true "bugs," please report them to the authors. If you have specific applications that require consultation or program modification, the authors are willing to discuss arrangements for meeting your specific need.

For sale by the National Technical Information Service, 5285 Port Royal Road
Springfield, VA 22161

CONTENTS

Page

ACKNOWLEDGMENTS..... xii

ABSTRACT..... 1

PART 1. WHAT THIS RAY-TRACING PROGRAM CAN DO

1. Introduction to Hamiltonian Ray Tracing..... 3
   1.1 Rationale..... 3
   1.2 What Is Ray Tracing?..... 4
   1.3 What Approximations Are Involved in Ray Tracing?..... 5
   1.4 When Should You Use Ray Tracing?..... 6
   1.5 What Is Hamiltonian Ray Tracing?..... 7
   1.6 What This Program Does..... 10
   1.7 What This Program Does Not Do..... 11
   1.8 History of the Program..... 12
   1.9 Scope of This Report..... 13

2. A Sample Run Illustrating an Application of HARPO..... 15
   2.1 The Ocean Model for the Sample Case..... 15
   2.2 The Ray-Tracing Order Form for the Sample Case..... 28
   2.3 Rayplot Order Form for the Sample Case..... 32
   2.4 Setting Up the Input Data File (W Array) for the Sample Case.... 35
   2.5 Printed Output for the Sample Case..... 40
      2.5.1 Definitions of Quantities Listed in Printout..... 42
      2.5.2 Meanings of Special Events Along a Raypath..... 43
   2.6 Rayplots for the Sample Case..... 44
   2.7 Machine-Readable Output for the Sample Case..... 47

PART II. HOW TO USE THIS PROGRAM

3. How To Get This Program Running on Your Computer..... 51
   3.1 How To Get a Copy of the Program..... 51
   3.2 ANSI-FORTRAN 77 Compatibility..... 51
      3.2.1 Machine- and System-Dependent Programming..... 52
      3.2.2 Word Length..... 52
      3.2.3 Execution Speed..... 53
      3.2.4 Graphics..... 53
   3.3 Unpacking the Program Tape and File Organization..... 54
      3.3.1 Files on the HARPO Distribution Tape..... 54
      3.3.2 Setting Up a Run Module..... 54
      3.3.3 If You Are Using Cards to Input Data..... 56
      3.3.4 If You Are Using Permanent Disk Files..... 56

4. How To Construct an Ocean Model..... 57
   4.1 Choosing From the Available Ocean Models..... 57
      4.1.1 Model Check Numbers..... 60
      4.1.2 Tabular Input to Models..... 60
   4.2 Designing Your Own Models..... 60

4.2.1 How To Write an Ocean Model Subroutine So It Can Receive Tabular Data Read Into Common Blocks by READW1..... 66
4.2.2 Designing Your Own Ocean Bottom and Ocean-Surface Models. 67

5. How to Specify the Input Data and Set Up an Input Data File..... 69
5.1 Editing the Input Data File..... 69
5.2 Input Data Formats..... 73
5.3 Specifying the W-Array Input..... 73
5.3.1 Initialization of the Input Data Parameters..... 75
5.3.2 Explanation of the Input Data Parameters..... 76
5.4 Specifying Tabular Input..... 82
5.4.1 Changing to the Tabular Format..... 82
5.4.2 The Format Line..... 82
5.4.3 The Units Line..... 85
5.4.4 The Data Lines..... 85
5.4.5 The Return Line..... 85

PART III. HOW THE PROGRAM WORKS

6. The Ray-Tracing Equations..... 87
6.1 Hamilton's Equations in Spherical Polar Coordinates..... 87
6.1.1 Phase Path..... 90
6.1.2 Absorption..... 90
6.1.3 Path Length..... 91
6.2 Numerical Integration..... 92
6.3 The Ocean Sound-Speed Models..... 99
6.4 Ocean Dispersion Relations..... 99
6.4.1 ANCNL - Acoustic, No Currents, No Losses..... 100
6.4.2 AWCNL - Acoustic, With Currents, No Losses..... 101
6.4.3 ANCWL - Acoustic, No Currents, With Losses and AWCWL - Acoustic, With Currents With Losses..... 103
6.5 Reflecting Rays From the Bottom..... 104
6.5.1 Detecting Ray Intersections With the Bottom..... 104
6.5.2 The Surface Model..... 109
6.5.3 Reflecting the Ray From the a Surface..... 111
6.5.4 Unit-Normal Directions From the Surface..... 114
6.6 Coordinate Systems..... 115

7. Structure of the Program..... 117
7.1 Description of the Subroutines..... 117
7.1.1 Ray-Tracing Core..... 117
7.1.2 Dispersion Relation Routines..... 120
7.1.3 Ocean Model Subroutines..... 121
7.2 HARPO Organization Diagrams..... 122
7.3 Flow Charts for Program RAYTRC and Subroutines TRACE and BACKUP..... 122
7.4 Common-Block Structure and Usage..... 125

REFERENCES..... 169

APPENDIX A. PRINTOUT AND RAYSET LISTING FOR THE SAMPLE CASE..... 175

APPENDIX B. BLANK INPUT PARAMETER FORMS..... 211

3-D RAYPATH CALCULATION..... 213

RAYPATH PLOT PROJECTION..... 214

OCEAN MODELS..... 215

BACKGROUND CURRENT MODELS

WLINEAR -- A linear profile of u; constant v and w..... 216
VVORTX3 -- A vertical vortex with a solid-rotating core..... 217
WGAUSS2 -- A jet of u that decays in three dimensions..... 218

BACKGROUND SOUND-SPEED MODELS

CTANH -- C profile with linear segments smoothly joined..... 219
CSTANH -- C^2 profile with linear segments smoothly joined..... 220
CSSPOKE -- C^2 as a function of angle from horizontal..... 221
CSSPOK2 -- C^2 as a function of angle from horizontal..... 222
CSMUNK1 -- Canonical sound channel; linear parameters in longitude..... 223
CSMUNK2 -- Canonical sound channel; linear C with longitude.... 224
CTABLE -- A tabular sound-speed profile in cubic segments..... 225

SOUND-SPEED PERTURBATION MODELS

CBLOB2 -- A blob of C^2 increase that decays in three dimensions..... 226
CBLOB3 -- Up to 3 blobs of C increase that decay in three dimensions..... 227
CBLOB4 -- Up to 3 blobs of C^2 increase that decay in three dimensions..... 228

ABSORPTION MODEL

SLLOSS -- Skretting-Leroy model of acoustic absorption..... 229

BOTTOM MODELS

GHORIZ -- A horizontal surface at a specified height above sea level..... 230
GLORENZ -- A Lorentzian ridge along a latitude line..... 231
GTANH -- Linear segments smoothly joined..... 232

OCEAN-SURFACE MODELS

SHORIZ -- A horizontal surface at a fixed height above sea level.... 233

RECEIVER-SURFACE MODELS

RHORIZ -- A surface at a fixed height above mean sea level..... 234
RBOTM -- A surface at a fixed height above the bottom..... 235
RVERT -- A vertical receiver surface at a fixed range from an origin..... 236

APPENDIX C. FR-80 PLOT PACKAGE AND DISSPLA INTERFACE..... 237

APPENDIX D. FORTRAN SOURCE-CODE LISTINGS..... 243

D.1 RAY-TRACING CORE (Tape File 3)..... 245
D.2 DISPERSION-RELATION ROUTINES (Tape File 4)..... 343
D.3 OCEAN MODEL ROUTINES (Tape File 5)..... 357
D.4 TAPRD -- Graphics file read routine (Tape File 6)..... 432
D.5 DDSPLA -- DISSPLA interface routines (Tape File 7)..... 435
D.6 DDALT -- Skeleton graphic interface routines (Tape File 8)..... 439

APPENDIX E. ERRATA FOR HARPA REPORT..... 449

INDEX..... 451

FIGURES

Page

Figure 1.1. Relation of HARPO to its inputs and outputs, as well as its supporting and supplementary programs..... 14

Figure 2.1. Sample of completed form to specify an ocean model, listing models used in the sample case..... 16

Figure 2.2. Sample of completed form to specify ocean current model VVORTX3..... 18

Figure 2.3. Contours of ocean current for model VVORTX3 in the vertical ray-transmission plane..... 19

Figure 2.4. Contours of ocean current for model VVORTX3 in a horizontal plane at a depth of 1 km..... 19

Figure 2.5. Sample of completed form to specify input data for ocean sound-speed model CTANH..... 20

Figure 2.6. Sound-speed profile for model CTANH with parameters used in the sample case..... 21

Figure 2.7. Sample of completed form to specify sound-speed perturbation model CBLOB2..... 23

Figure 2.8. Contours of total sound speed in the vertical ray-transmission plane using models CTANH and CBLOB2..... 24

Figure 2.9. Contours of total sound speed in a horizontal plane at 1 km depth, corresponding to the sound-speed models of the sample case..... 24

Figure 2.10. Sample of completed form to specify absorption model SLLOSS..... 25

Figure 2.11. Frequency dependence of ocean absorption model SLLOSS..... 26

Figure 2.12. Sample of completed form to specify ocean-bottom model GLORENZ..... 27

Figure 2.13. Sample of completed form to specify ocean-surface model SHORIZ..... 29

Figure 2.14. Sample of completed form to specify a three-dimensional raypath calculation..... 30

Figure 2.15. Sample of completed form to specify receiver surface model RHORIZ..... 31

Figure 2.16. Sample of completed form to specify a vertical rayplot projection..... 33

Figure 2.17. Sample of completed form to specify a horizontal rayplot projection..... 34

Figure 2.18. Plan view of the major features of the sample-case model, the source location, the transmission plane, and the plot-projection plane..... 36

Figure 2.19. Input Data File for the sample case..... 37

Figure 2.20. Projection of the rays of the sample case onto the vertical plane shown in Figure 2.18..... 45

Figure 2.21. Projection of the rays of the sample case onto a horizontal plane whose axis is shown in Figure 2.18..... 45

Figure 2.22. Ray-apogee plot for the sample case..... 46

Figure 2.23. Rayset output for the first run set of the sample case..... 48

Figure 2.24. Definitions and format of a transmitter rayset..... 49

Figure 2.25. Definitions and format for a receiver rayset..... 50

Figure 3.1. Configuration of a run module, assembled from parts consisting either of disk-file modules or punched-card decks..... 55

Figure 4.1. Listing for model sound-speed subroutine CTANH..... 63

Figure 5.1. Input Data File (W array) for the sample case..... 70

Figure 6.1. Flow chart for subroutine RKAM..... 94

Figure 6.2. Flow chart for subroutine RKAM1..... 95

Figure 6.3. Flow chart for the Runge-Kutta procedure..... 96

Figure 6.4. Flow chart for the Adams-Moulton integration method..... 97

Figure 6.5. Flow chart for the single-step error analysis..... 98

Figure 6.6. Three steps in calculating the intersection and reflection of a ray at a surface..... 105

Figure 6.7. A ray crossing a bottom or receiver surface and crossing back again between integration steps..... 106

Figure 6.8. A ray that nearly intersects a bottom or receiver surface..... 106

Figure 6.9. A ray crossing a bottom or receiver surface and ending closer to a second intersection..... 106

Figure 6.10. Examples of bottom (or receiver-surface) encounters that HARPO's intersection algorithm will not correctly handle..... 107

Figure 7.1. Block diagram of the ray-tracing program showing the relation of the main program to other subroutines during the initialization stage..... 126

Figure 7.2. Continuation of the block diagram of the ray-tracing program (Fig. 7.1), showing the relations among the ocean model subroutines during the initialization stage..... 127

Figure 7.3. Block diagram of the ray-tracing program showing the relations among the main program and the subroutines during the raypath calculation stage..... 128

Figure 7.4. Continuation of the block diagram of the ray-tracing program (Fig. 7.3) that shows the relations among the ocean model subroutines during the raypath calculation stage..... 129

Figure 7.5. Flow chart for program RAYTRC..... 130

Figure 7.6. Flow chart for subroutine TRACE..... 132

Figure 7.7. Flow chart showing some details of handling surface crossings in subroutine TRACE..... 133

Figure 7.8. Details of finding crossings of and closest approaches to receiver surface..... 134

Figure 7.9. Flow chart for subroutine BACKUP..... 135

Figure 7.10. Flow chart for the algorithm that steps the ray by numerical integration to a crossing of a specified height..... 136

Figure 7.11. Flow chart for the algorithm that steps the ray by numerical integration to a point of closest approach to a specified height..... 137

Figure C1. An organization chart that shows how graphical output is produced by a series of programs, some of which are a part of HARPO, and others of which are either supplied along with HARPO or are commercial packages..... 238

TABLES

Page

Table 4.1. Available ocean models..... 58

Table 4.2. Allocation of common blocks for tabular input to the various ocean models..... 61

Table 4.3. Allocation of common blocks for output from the various atmospheric models..... 62

Table 4.4. Assignment of entry point names and input parameter blocks in the W array for the ocean models..... 66

Table 4.5. Definitions of the parameters in common block /B3/..... 68

Table 5.1. Units conversion on input..... 74

Table 5.2. Description of the identifying numbers in the first three columns of the Input Data File..... 83

Table 5.3. Tabular input data formats available..... 84

Table 6.1. The more important symbols and their definitions..... 88

Table 6.2. Run time vs. accuracy..... 92

Table 7.1. Definitions of the parameters in blank common..... 138

Table 7.2. Definitions of the parameters in common block /MCONST/..... 139

Table 7.3. Definitions of the parameters in common block /PCONST/..... 139

Table 7.4. Common block usage by the core routines..... 140

Table 7.5. Input and output variables to routines in common blocks.... 141

Table 7.6. Definitions of the parameters in common block /HDR/..... 146

Table 7.7. Definitions of the parameters in common block /UCONC/..... 147

Table 7.8. Definitions of the parameters in common block /UCONV/..... 147

Table 7.9. Definitions of the parameters in common block /RKAMS/..... 148

Table 7.10. Definitions of the parameters in common block /CGET/..... 148

Table 7.11. Definitions of the parameters in common block /CRMACH/..... 149

Table 7.12. Definitions of the parameters in common block /CRKTIME/.... 149

Table 7.13. Definitions of the parameters in common block /TRLOCAL/.... 150

Table 7.14. Definitions of the parameters in common block /RK/..... 151

Table 7.15. Definitions of the parameters in common block /TRAC/..... 152

Table 7.16. Definitions of the parameters in common block /FNDER/..... 153

Table 7.17. Definitions of the parameters in common block /RIN/..... 154

Table 7.18. Definitions of the parameters in common block /ERR/..... 155

Table 7.19. Definitions of the parameters in common block /RAYDEV/..... 155

Table 7.20. Definitions of the parameters in common block /FLGP/..... 156

Table 7.21. Definitions of the parameters in common block /RINPL/..... 156

Table 7.22. Definitions of the parameters in common block /FLG/..... 157

Table 7.23. Definitions of the parameters in common block /HDRC/..... 157

Table 7.24. Definitions of the parameters in common block /FILEC/..... 158

Table 7.25. Definitions of the parameters in common block /PLT/..... 158

Table 7.26. Definitions of the parameters in common block /RAYCON/..... 159

Table 7.27. Definitions of the parameters in common block /ANNCTC/..... 159

Table 7.28. Definitions of the parameters in common block /ANNCTL/..... 159

Table 7.29. Definitions of the parameters in common block /LABCLT/..... 160

Table 7.30a. Definitions of the parameters in common block /DD/..... 161

Table 7.30b. Definitions of the parameters in common block /DDREZ/..... 161

Table 7.31. Definitions of the parameters in common block /KNKN/..... 162

Table 7.32. Definitions of the parameters in common block /DDLIM/..... 162

Table 7.33. Definitions of the parameters in common block /UU/..... 163

Table 7.34. Definitions of the parameters in common block /CC/..... 164

Table 7.35. Definitions of the parameters in common block /LL/..... 165

Table 7.36. Definitions of the parameters in common block /SS/..... 166

Table 7.37. Definitions of the parameters in common block /RR/..... 167

Table 7.38. Definitions of the parameters in common block /GG/..... 168

Table D1. Alphabetical list of source-code modules..... 244

ACKNOWLEDGMENTS

Part of the organization of this program into subroutines follows that of the program of Dudziak (1961). Also, the coordinate transformation in subroutine PRINTR and the method for data input via the W array are taken from the program of Dudziak (1961). The term "rayset," the idea of outputting computer-readable results of each hop for each ray trace, and the idea of automatically plotting raypaths come from the program of Croft and Gregory (1963). Subroutine RKAM1 is a modification of subroutine RKAMSUB, written by G. J. Lastman, and is available through the CDC CO-OP library (the CO-OP identification is D2 UTEX RKAMSUB). Subroutine GAUSEL was written by L. David Lewis, NOAA Space Environment Laboratory. Judith Stephenson wrote much of the original ionospheric ray-tracing program, upon which HARPO is based. Richard Lindzen devised the method upon which models CSTANH, CTANH, and GTANH are based. We also thank the many users of earlier versions of the program who provided helpful feedback. Special thanks go to the Editorial Staff of Publication Services for extensive help in clarifying the expression of our ideas, and to Ms. Mildred Birchfield for her excellent typing and layout of the manuscript.

HARPO -- A VERSATILE THREE-DIMENSIONAL HAMILTONIAN RAY-TRACING PROGRAM FOR ACOUSTIC WAVES IN AN OCEAN WITH IRREGULAR BOTTOM

R. Michael Jones, J. P. Riley, and T. M. Georges

ABSTRACT

HARPO stands for Hamiltonian Acoustic Ray-tracing Program for the Ocean. This FORTRAN computer program traces the three-dimensional paths of acoustic rays through model oceans by numerically integrating Hamilton's equations, which are a differential expression of Fermat's principle. The user specifies an ocean model by writing closed-form formulas for its three-dimensional current and sound-speed distribution, and by defining the depth of the bottom as a function of geographic latitude and longitude. Some general-purpose models are provided, or users can easily design their own.

Because it uses continuous models, the Hamiltonian method avoids the false caustics and discontinuous raypath properties encountered in conventional ray-tracing methods, which use layers or cells where each acoustic-raypath segment can be computed in closed form. Furthermore, computational speed can be traded for accuracy, without changing the model of the medium, by specifying the maximum allowable integration error per step.

In addition to computing the geometry of each raypath, the program can calculate pulse travel time, phase time, Doppler shift (if the ocean varies in time), absorption, and geometrical path length. Amplitude is not explicitly computed, but the contributions by absorption, reflection losses, and focusing are separately available for each ray. Only geometrical effects are accounted for; that is, no diffraction or partial-reflection corrections are applied. The program prints out a step-by-step account of each ray's progress, and it can plot the projection of a set of rays on any horizontal or vertical plane. Furthermore, it can output each ray's properties in machine-readable form for further processing (amplitude and eigenray calculations, for example).

This report describes the ray-tracing equations and the structure of the program and provides complete instructions for using it, illustrated by a sample

case. The program is modular and can be adapted to model propagation through other media by changing the routine that defines the medium's dispersion relation.

PART I: WHAT THIS RAY-TRACING PROGRAM CAN DO

1. Introduction to Hamiltonian Ray Tracing

1.1 Rationale

Many practical problems in ocean acoustics submit to a straightforward application of geometrical acoustics, or ray theory. No other propagation-modeling tool provides such an intuitive and graphic portrayal of the paths that acoustic energy follows through inhomogeneous media. Even in situations where ray theory does not strictly apply, a picture of the acoustic raypaths often provides a useful first look at where sound goes in the ocean, and it gives insight into where higher order computations may be required. Some calculations cannot be easily made in any other way, for example, computing multipath pulse travel time or showing which parts of the ocean affect each pulse arrival.

Yet most of the ray-tracing computer programs in common use fail to take full advantage of the power of geometrical acoustics. Many are essentially automated versions of graphical techniques that patch together closed-form raypath solutions for layers or cells with simple refractive-index gradients (Roberts, 1974; Cornyn, 1973). In such models, gradient discontinuities at cell boundaries can introduce false caustics and cause discontinuous behavior of ray properties as launch angle varies (Pedersen, 1961). Furthermore, it is difficult to extend such models to three-dimensional oceans, to account for currents, and to compute reflections from complicated bottom models.

This report describes a general-purpose underwater acoustic ray-tracing program called HARPO -- for Hamiltonian Acoustic Ray-tracing Program for the Ocean -- that we have designed to overcome these limitations. It computes acoustic raypaths by numerically integrating Hamilton's equations, which are a differential expression of Fermat's principle. The user defines an ocean model by writing closed-form expressions for its sound-speed and current distribution in three dimensions, and by defining the bottom depth (or bathymetry) as a function of latitude and longitude. Several simple but generally useful models with user-specifiable parameters are described in this report; users can pattern their own models after them.

HARPO is the companion to a similar program we have developed for the atmosphere, known as HARPA. The main differences between the two programs are in the models available for the two media, and in provisions for reflections from an upper surface (in the ocean case). HARPA is documented in a separate report (Jones et al., 1986b).

1.2 What Is Ray Tracing?

Although ray tracing has a long history, many people outside the field do not know what ray tracing is or what it can do. In ray theory, waves are treated like particles (photons of light, phonons of sound) that travel along geometric trajectories called rays. In material media, the particles travel at a speed determined by the medium's "refractive index." Gradients in refractive index bend rays, giving rise to the problem of computing ray trajectories through a known spatial distribution of refractive index. Ray tracing is any method, graphical or numerical, for solving that problem.

Originally, lensmakers used ray tracing to find out how light rays travel through optical systems. They used graphical techniques based on Snell's law to compute the bending that light rays suffer when they encounter abrupt changes in refractive index, as at the surface of a lens. By constructing bundles of such rays, lensmakers could simulate the magnification, reduction, and focusing of their lens designs without actually building them.

Modern ray-tracing applications, whether acoustic or electromagnetic, serve basically the same function: they allow one to simulate the propagation of waves through media whose refractive-index structure varies in a complicated way, without actually performing the physical measurement. Modern ray-tracing computations are usually performed by programs written for digital computers that can graphically display the results of their computations in informative ways.

Today's ray-tracing programs do much more than compute how rays bend as they cross interfaces; they can model media whose refractive index varies continuously in space and even with time. In dissipative media, they integrate absorption along the raypath. They can also integrate phase and pulse travel time, as well as wave amplitude. In time-varying media, they can integrate

the rate of change of phase, or Doppler shift. Some programs (including HARPO) produce machine-readable output so that the results of many raypath computations can be processed by other programs to display field observables, such as amplitude.

The most advanced applications of ray-tracing computer programs have been in the fields of ionospheric radiowave propagation, seismic wave propagation, and the propagation of acoustic or sound waves in the ocean and the atmosphere. In the Hamiltonian formalism, the ray-tracing equations for acoustic, seismic, and electromagnetic waves are identical. To go from one kind of ray-tracing program to another, all you have to do is change the modules that describe the wave dispersion relation and how the medium varies in space.

1.3 What Approximations Are Involved in Ray Tracing?

Solving a wave equation with arbitrary boundary conditions is still an impractical task, even for the most modern computers. Therefore, practical problems in wave propagation are often solved by making simplifying approximations to the wave equation. Examples of such approximations are the parabolic-equation (P.E.) method (Tappert, 1977), normal-mode theory (Tolstoy and Clay, 1966; Pierce, 1965), fast-field methods for numerical integration of the wave equation in range-independent environments (Raspet et al., 1985), and ray theory.

Ray theory, sometimes called the WKB or eikonal method, results from making a high-frequency approximation in the solution of arbitrary elliptic or hyperbolic partial differential equations (Budden, 1961). Ray tracing is related to the "method of characteristics" for solving such equations because the acoustic raypaths are the bi-characteristic rays of the differential equations in the infinite-frequency, infinite-wave-number limits. In some fields, ray tracing is called the "shooting method" because (as with shooting a gun) the location of the end point is found by trial and error while the initial conditions of a ray are varied.

In the case of the wave equation, the approximation gives rise to the fields of geometrical acoustics or geometrical optics, which are concerned

with the trajectories of bundles of acoustic or electromagnetic energy radiated in infinitesimal angular beams. Such rays experience no diffraction but produce sharp shadow boundaries when they encounter solid objects. Ray theory can be extended to include the effects of diffraction, for example, by using the Geometrical Theory of Diffraction (GTD) (Keller, 1962).

In ray theory, one assumes conservation of energy within a bundle of rays called a flux tube, so that wave intensity is inversely proportional to the cross-sectional area of the flux tube. When that cross-sectional area becomes zero (a "caustic"), ray theory predicts infinite energy density. (This does not prevent rays from being traced through caustics, however.) Higher order corrections to ray theory can give more accurate field estimates near caustics when needed. For example, the field near a surface caustic can be calculated in terms of Airy functions (Ludwig, 1966; White and Pedersen, 1981).

Without such corrections, ray tracing accounts only for refraction by large-scale gradients in the medium and not for diffraction and scattering by changes in the medium over scales that are small compared to a Fresnel zone. Even so, ray theory provides a useful first look at many complicated propagation problems and gives a kind of graphical insight lacking in other propagation models.

1.4 When Should You Use Ray Tracing?

Ray tracing is best suited to modeling sound propagation in oceans whose refractive index can be described deterministically in one, two, or three dimensions, and where changes in refractive index are small within an acoustic Fresnel zone (this is the WKB criterion). (This means that ray models are most accurate at high frequencies.) In such environments, ray tracing gives accurate information about the geometrical paths followed by acoustic rays (energy), about shadow boundaries and reflections from surfaces, and about phase, intensity, pulse travel time, absorption and Doppler shift (for time-varying media) integrated along those paths.

In models where multiple rays reach a receiver location of interest, additional computations, external to the ray tracing, may be required to combine field information from multiple rays. When the number of multipath rays be-

comes large, alternative formulations of the problem (P.E. or normal-mode theory, for example) are more appropriate for continuous-wave amplitude calculations. For pulse transmissions, ray theory is useful for describing the distinct geometric paths corresponding to each pulse arrival and for computing multipath travel times.

In situations where the applicability of ray theory is doubtful, a raypath picture can tell which regions must be treated with higher order methods, such as GTD or the Airy-function approximation to the field near a caustic. Furthermore, there are standard formulas to estimate how close to a caustic amplitude calculations are accurate (Budden, 1961; 1972).

Even when ray calculations of one wave quantity become inaccurate, they can give useful estimates of others. For example, when amplitude estimates break down (as at surface caustics), other information, such as travel time or phase, may still be reliable and can be tracked through caustics. Furthermore, Budden (1961, pp. 325-326) shows that the ray-computed phase must be advanced by 90° every time a ray passes through a surface caustic.

1.5 What Is Hamiltonian Ray Tracing?

An alternative to cellular methods requires the ocean to be modeled as a continuous three-dimensional function with continuous gradients and computes each raypath by numerically integrating Hamilton's equations with a different set of initial conditions. This method has been called Hamiltonian ray tracing. Hamilton's equations are the same for all kinds of wave propagation; only the definition of the Hamiltonian varies when going from one wave type to another.

Although Hamilton's equations are more familiar in mechanics, they have a long history of application to more general problems, including wave propagation. There, the point of view is that in a high-frequency limit, waves behave like particles and travel along rays, according to equations that exactly parallel those governing changes of position and momentum in mechanical systems (Lighthill, 1978, Sec. 4.5). Two steps show that integrating Hamilton's equations can lead to approximate solutions of a wave equation:

(1) The first step is to show that solutions to the wave equation are related to paths that satisfy a particular stationary principle, usually called

Fermat's principle. There are at least two standard methods for demonstrating that relation.

(a) First is the method of characteristics (see, for example, Courant and Hilbert, 1962; Garabedian, 1964), in which the solution is related to initial-value data chosen on some appropriate surface. Specifying a surface and constructing a solution requires first constructing the characteristic surfaces that are wave fronts of the wave. These characteristic surfaces can be constructed by first constructing bi-characteristic rays that satisfy a stationary principle. The bi-characteristic rays are the same as the geometrical raypaths whenever all terms in the wave equation are proportional to a derivative of the wave function, or in the limit of infinite frequency and wave number.

(b) Second is the path-integral method (see, for example, Feynman and Hibbs, 1965), in which a solution to the wave equation is constructed as an integral over all possible paths (not just raypaths) that connect the source and observer. Making a saddlepoint (or stationary phase) approximation to the path integral finds the paths that contribute most to the path integral. Such paths are those for which the action (phase) is stationary for variations of the path; that is, they satisfy Fermat's principle.

(2) The second step is to show that Hamilton's equations can be integrated to construct paths that satisfy a variational principle, such as Fermat's principle. This is done in standard texts (for example, Lighthill, 1978). First, the variational principle is expressed as an integral of a Lagrangian along the path (specified in terms of generalized coordinates, q_i). This determines the form of the Lagrangian for the problem, which for the wave equation is usually some simple function of the phase refractive index. Then the generalized momenta p_i are defined, which for the wave equation correspond to components k_i of the wave number vector. Then a Hamiltonian $H(q_i,p_i)$ is constructed from the Lagrangian. For the wave equation, the Hamiltonian is usually a function that gives the dispersion relation for the wave in question when it is set to zero. Integrating Hamilton's equations then gives a path that satisfies the variational (Fermat's) principle.

In Cartestian coordinates, Hamilton's equations take the particularly simple form (Lighthill, 1978)

$$\frac{dx_i}{d\tau} = \frac{\partial H}{\partial k_i} \quad ; \quad \frac{dk_i}{d\tau} = -\frac{\partial H}{\partial x_i} \quad , \quad i = 1 \text{ to } 3 \quad , \tag{1.1}$$

where τ is a parameter (sometimes proportional to time) whose physical meaning depends on the how the Hamiltonian, H, is defined, k_i are the wave-number components, and x_i are the coordinates of a point on the raypath. Transforming to spherical polar coordinates complicates the equations considerably. The full set of equations for spherical coordinates can be found in Chapter 6.

To solve (1.1) for the raypath, one chooses initial values for the six quantities x_i and k_i and performs a numerical integration of the system (1.1) of six total differential equations. For acoustic waves in the ocean, the Hamiltonian (which is constant along a raypath) is defined as

$$H(x_i, k_j) = [\omega - \vec{k} \cdot \vec{V}(x_i)]^2 - C^2(x_i)\, k^2 = 0 \quad , \tag{1.2}$$

where $\vec{V}(x_i)$ is the ocean current, $C(x_i)$ is the sound-speed field, and ω is the angular wave frequency ($\vec{V}$ and C may also depend on time). Thus, the effects of a three-dimensional vector-current field are automatically included in the definition of the Hamiltonian.

There is an alternative to Hamilton's equations for a differential form of the ray equation, namely the eikonal equation (see, for example, Garabedian, 1964, p. 166; Felsen and Marcuvitz, 1973, p. 126). The eikonal equation is derived by first assuming an approximate solution to the wave equation in terms of an asymptotic series. Substituting the asymptotic series into the wave equation leads to the eikonal equation, which determines the raypaths, and a transport equation, which determines an approximate solution to the wave equation. The eikonal equation is equivalent to Hamilton's equations for determining the raypath. The transport equation is equivalent to methods mentioned above for determining an approximate solution to the wave equation.

In addition to allowing continuous three-dimensional models of the refractive-index field and two-dimensional models of reflecting surfaces, Hamiltonian ray tracing by numerical integration permits the user to trade computing speed for accuracy by specifying the maximum allowed integration error per step. In other words, you can have a fast but crude ray trace or a slower and more accurate one. HARPO automatically adjusts the integration

step length along the raypath to keep the error within specified bounds. In regions where the refractive index varies quickly, small steps are required, but where it varies slowly, large steps save computation. If the quantity being integrated varies monotonically along the raypath, the specified relative accuracy will be preserved in integrated quantities, such as travel time.

1.6 What This Program Does

HARPO computes the paths of acoustic rays, one at a time, through a user-defined model of the ocean, given initial conditions that include the source location (latitude, longitude, and depth), wave frequency, direction of transmission (elevation and azimuth), the receiver-surface model, and the maximum number of hops (intersections with the receiver surface). The input data specification forms in Chapter 2 illustrate the generality of acceptable input.

The mechanics of the raypath calculation have been completely separated from the modeling of the medium (sound-speed, current-velocity, and bottom models). This allows the user to select models from those we have developed or to develop new models simply by writing new (or altering existing) subroutines to define those models.

The modular structure of the program allows the user to extend the program easily to other types of geophysical ray tracing simply by substituting new subroutines for defining the Hamiltonian and the model of the medium.

The method for putting data into the program is easy to learn. The user simply specifies the magnitude and units of the elements of an Input Data File that correspond to physical or mathematical quantities that tell the program what models to use, what rays to trace, and in what form to present its results. We provide input parameter forms for making sure that all the required quantities are specified.

At the user's option, HARPO produces three kinds of output: (1) The printout reproduces the input data set and gives detailed information about each raypath computed, in columnar form, with each line corresponding to a "snapshot" of the ray's progress after a specified number of integration steps. (2) Computer-readable output permits further processing of raypath

data by supplementary programs, without recomputing the raypaths. (3) The raypath plots show projections of any part of the raypaths on any vertical or horizontal plane, with any desired magnification. These plots give the user a quick view of the raypath geometries.

Chapter 2 illustrates more fully what the program does by going through the setup and execution of a representative application.

1.7 What This Program Does Not Do

HARPO's computations lie entirely within the scope of geometrical acoustics (ray theory). It applies no corrections for diffraction or partial reflections. The ocean model must be deterministic, not random.

There are no provisions built into HARPO for explicit computations of acoustic amplitude. This would normally be done with a supplementary program that processes HARPO's machine-readable output. Total amplitude at a receiver would be computed by combining flux-tube focusing, reflection losses, and absorption, and the user would normally decide whether to add coherently or incoherently the contributions of multipath rays. Because there are so many ways to compute amplitude, we think it is most useful to keep the various factors separate and let the users combine them however they wish.

Because the numerical integration of Hamilton's equations requires ocean models with continuous gradients, HARPO cannot presently handle refraction at discontinuities of refractive index or its gradients. If such discontinuities are included in a model, the integration routine will attempt to handle them by taking extremely small steps when a ray encounters a discontinuity, and the results may not be reliable. In general, one can approximate discontinuous functions with continuous functions to any desired accuracy, and HARPO will adjust its step length to accommodate them. Our algorithms for reflecting rays from arbitrary bottom surfaces could be generalized to compute refraction at discontinuities in refractive index.

HARPO is not now equipped to model penetration of rays into the bottom or to account for partial reflections from sub-bottom layers. However, the user can specify a complex (to account for phase and amplitude) bottom-reflection coefficient that is a function of frequency and angle of incidence. Since

reflection coefficients do not affect the raypaths, their effects can be added (and varied) after raypath calculations.

HARPO cannot directly compute the raypaths that connect a specified source and receiver. To find such "eigenrays," one usually launches a fan of rays at small increments in elevation and/or azimuth angle and linearly interpolates for the rays that reach the desired receiver location (range, azimuth and depth). We have developed an eigenray program that processes the "rayset" output of HARPO, interpolating in elevation angle only, and that will be documented elsewhere. Some shortcuts for finding three-dimensional eigenrays when azimuthal deflections are small are described by Georges et al. (1986).

HARPO makes no checks to see if ocean models satisfy physical conservation laws and boundary conditions, or that current and sound-speed models are geostrophically consistent. (Accurate raypaths can be computed through physically impossible models.) It is the users' responsibility to make their models as physically realistic as their application demands.

1.8 History of the Program

HARPO has a long history of development. We started by learning from the programs of Dudziak (1961) and Croft and Gregory (1963). Jones (1966) documented our first version of a three-dimensional ray-tracing program for radio waves in the ionosphere, which included anisotropy caused by the earth's magnetic field. Jones (1968) documented improvements in the original program. Georges (1971) converted the ionospheric radio program to trace raypaths for acoustic-gravity waves in an atmosphere with winds and changed the ray-tracing equations into Hamiltonian form. Jones and Stephenson (1975) documented further significant improvements in the ionospheric program. Jones et al. (1982) documented a Hamiltonian acoustic ray-tracing program for an atmosphere over a spherical earth.

Through its history, HARPO and its predecessors have found application in the propagation of radio waves through the ionosphere (Georges, 1967; 1970; Georges and Stephenson, 1968; Stephenson and Georges, 1969), acoustic propagation through the atmosphere (Georges, 1972; Georges and Beasley, 1977) and acoustic propagation in the ocean (Georges et al., 1986; Jones et al., 1984,

1986a). In extending the utility of ray theory, Jones (1970) has treated ray propagation in lossy media (ray tracing in complex space), bending of rays in random, inhomogeneous media (Jones, 1981a), and the frequency shift suffered by pulses propagating in dispersive media (Jones, 1981b). Jones (1983) has also surveyed existing techniques for underwater acoustic ray tracing.

HARPO combines the improvements made by Jones and Stephenson with the acoustic-wave capability and ocean models developed by Georges, and it includes modularity features that make it easier to convert the program to trace rays in other media. It also includes algorithms developed by Jones (1982) for reflecting rays from arbitrary bottom surfaces. In addition, we have developed real-time graphics routines and improved access by time-share graphics terminals as that technology has advanced.

1.9 Scope of This Report

This report documents only the ray-tracing program HARPO, its supporting subroutines, and its various forms of input and output. The main intent of this report is to show what HARPO can do and to explain how to use it. We illustrate its capabilities with a comprehensive sample case. We also show how to extend and modify the program to the user's specific needs.

Not documented here are supplementary programs that we have designed to plot properties of the ocean models and to process the computer-readable output of HARPO. Examples of such programs are packages to compute eigenrays, to plot range vs. elevation angle of transmission, range vs. travel time, and amplitude calculations (Georges et al., 1986; Jones et al., 1984). We have not documented here our programs for editing input to HARPO or our procedure files for running it on our computer. Nevertheless, the package documented here is self-contained and has everything needed to compute and display raypaths through arbitrary three-dimensional model oceans.

Figure 1.1 shows an organization chart of HARPO in relation to its supporting modules. The dotted line encloses the portion documented in this report. Separate reports will document the remaining modules, which are the same for both HARPA and HARPO.

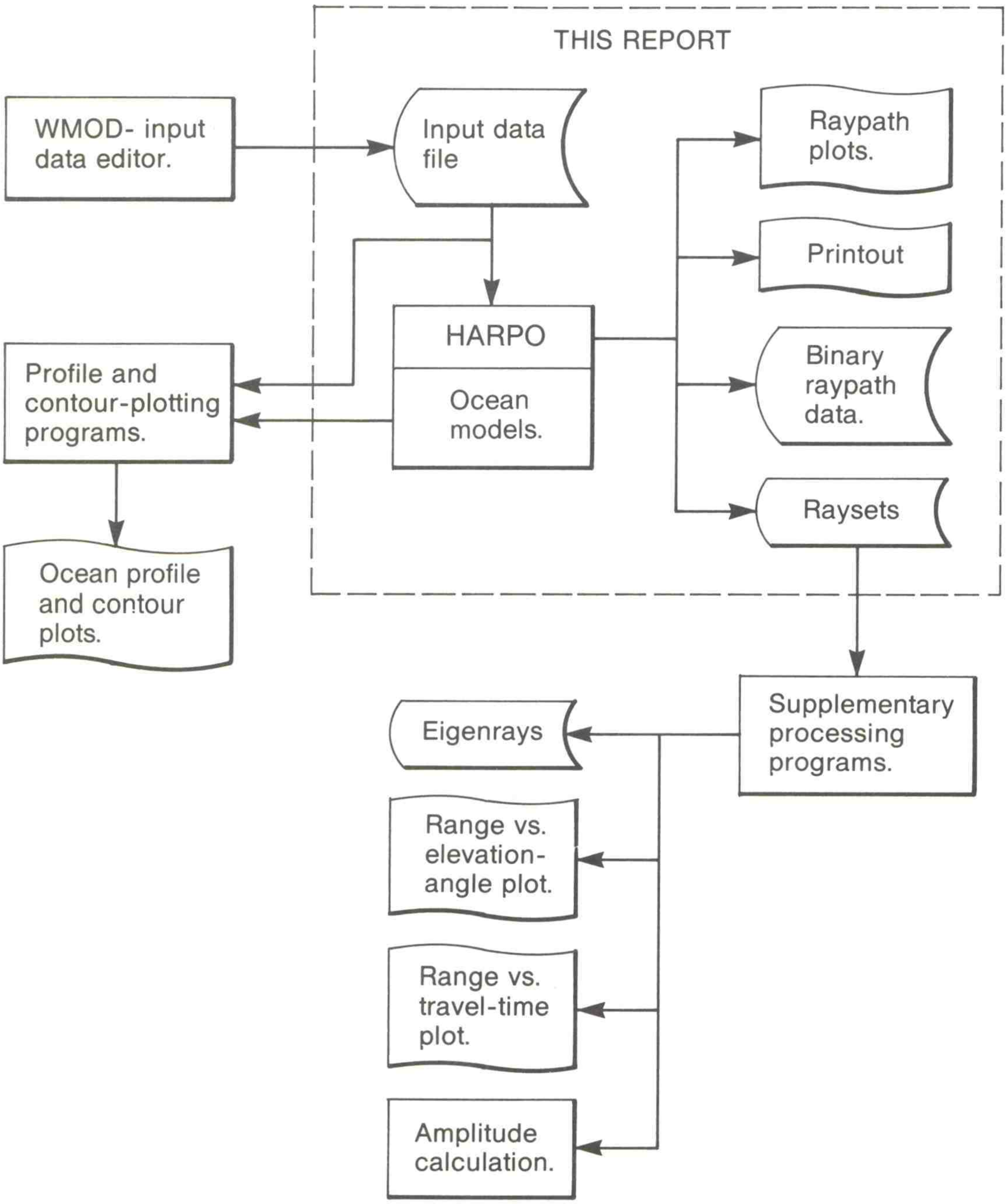


Figure 1.1. Relation of HARPO to its inputs and outputs, as well as its supporting and supplementary programs. The dashed line shows the scope of this report.

2. A Sample Run Illustrating an Application of HARPO

A sample case serves several purposes: it introduces new users to the capabilities of the program in terms of physical models they can understand; it shows the new user how to set up and run the program and what output to expect; and it provides a comprehensive test case to exercise the program and make sure everything works on a new machine. New users should run the sample case (provided with the program) first and make sure that the program's output is identical to that shown in this report. Varying the input parameters, one by one, from the sample case is an instructive way to explore the program's capabilities.

The usual procedure in defining models is to fill out "order forms" corresponding to the sound-speed, current, absorption and bottom models you want to use, then create an "Input Data File" from the information on the order forms. Models can be selected from the general-purpose ones we have created, or you can design your own. The following pages show filled-out forms for the models used by the sample case, and blank order forms for all our ocean models are supplied in Appendix B. We recommend that both beginning and advanced users fill out these forms, because they make sure that you specify all the required model parameters. They also help you keep track of the models you create.

2.1 The Ocean Model for the Sample Case

The ocean model described here is designed more to exercise features of the program than to represent any physically realistic situation. It combines a three-dimensional current and sound-speed field with a bottom model containing a Lorentzian-shaped ridge. The sample case also includes a simple absorption model that depends only on acoustic frequency.

Refer now to the FORM TO SPECIFY AN OCEAN MODEL (Figure 2.1). This form is filled in with the names of all the subroutines required to specify the ocean model for the sample case, including an ocean-bottom model, an ocean-surface model, and a receiver-surface model. Data-set ID numbers uniquely specify the particular set of parameter values used by each subroutine for the sample case. The entire set of models and parameters that constitute the ocean model for the sample case is also given a unique ID number, which is N01. The references to W

FORM TO SPECIFY AN OCEAN MODEL
(including bottom and upper surface model)

Name GEORGES Date 8-19-86 Model ID (3 characters) NØ1

Dispersion Relation
__ ANCNL ✓ AWCWL __AWCNL __ANCWL __Other _____

Coordinates of the north pole pole of the computational coordinate system:
North geographic latitude = 90. rad, km, (deg) (W24)
East geographic longitude = 0. rad, km, deg (W25)

Data Set ID	Model Subroutine Name				
Current Velocity	__	WLINEAR	(1.)	__	WGAUSS2 (8.)
W102 (W100)	3.	VVORTX3	(9.)	__	Other _____ ( )
Current Perturbation	0.	NPCURR	(0.)	__	Other _____ ( )
W127 (W125)					
Sound Speed	__	CSTANH	(2.)	__	CSMUNK1 (5.)
W152 (W150)	__	CSSPOKE	(3.)	__	CSMUNK2 (6.)
	__	CSSPOK2	(4.)	__	CTABLE (8.)
	1.	CTANH	(7.)	__	Other _____ ( )
Sound-Speed Perturbation					
W177 (W175)	__	NPSPEED	(0.)	7.	CBLOB2 (2.)
	__	CBLOB3	(3.)	__	Other _____ ( )
Receiver Surface*	✓	RHORIZ	(1.)		RTERR (2.)
(W275)		RVERT	(3.)		Other _____ ( )
Ocean Bottom	__	GHORIZ	(1.)	__	GTANH (2.)
W302 (W300)	3.	GLORENZ	(3.)	__	Other _____ ( )
Ocean-Bottom Perturbation					
W327 (W325)	0.	NPBOTM	(0.)	__	Other _____ ( )
Ocean Surface					
W352 (W350)	1.	SHORIZ	(1.)	__	Other _____ ( )
Ocean-Surface Perturbation					
W377 (W375)	0.	NPSURF	(0.)	__	Other _____ ( )
Absorption (loss)					
W502 (W500)	1.	SLLOSS	(1.)	__	Other _____ ( )
Absorption Perturbation					
W527 (W525)	0.	NPABSR	(0.)	__	Other _____ ( )
Graph Annotation*					
W75	__	SMPANN		.15	FULANN

* The receiver-surface and graph-annotation models are not considered part of the ocean-model ID.

Figure 2.1. Sample of completed form to specify an ocean model, with entries for the sample case.

followed by numbers in these forms correspond to specific input data parameters, as described in Section 5.3.

The first subroutine name on this form specifies the acoustic-wave dispersion relation to be used. In the sample case, we specify AWCWL, which means "Acoustic, With Currents, With Losses." This means that we will use a model ocean with currents, and we will calculate acoustic absorption. More efficient versions of the dispersion relation should be selected when currents or absorption models are not used (Sec. 6.4). The remaining subroutine names filled out on this form refer to ocean model subroutines, to be discussed next.

Refer next to the FORM TO SPECIFY INPUT DATA FOR CURRENT VELOCITY MODEL VVORTX3 (Fig. 2.2). This current model represents a vortex with a vertical axis and a solid-rotating core blending with a potential-vortex velocity falloff beyond a specified radius. A gaussian height profile of vortex intensity is also allowed.

VVORTX3 requires the user to specify six parameters: U_o, r_o, ϕ_o, λ_o, W_H and h_{max}. For the sample case, we select the maximum tangential speed to be 1.02 m/s, the radius r_o of the vortex core (where the tangential velocity reaches its maximum value, U_o) to be 50 km, the latitude of the vortex center, $\lambda_o = 0$, the longitude of the vortex center, $\phi_o = 150$ km (that is, 150 km east of the prime meridian, as measured at the Equator), the gaussian width in height, $W_H = 1$ km, and the height of the maximum velocity, $h_{max} = -1$ km (1 km below mean sea level). The resulting current field is illustrated in the contour plots of Figures 2.3 and 2.4. (The program that provided these plots is part of a set of peripheral programs that will be documented in another report.) Because we use no current-perturbation model in the sample case, we select the do-nothing current-perturbation model NPCURR.

Refer next to the FORM TO SPECIFY INPUT DATA FOR BACKGROUND SOUND-SPEED MODEL CTANH (Fig. 2.5). We use this model to specify the background sound-speed field of the ocean, in this case a horizontally uniform sound channel with a surface layer. CTANH is a general-purpose model that fits tabular sound-speed profiles with linear segments that join smoothly to form an analytic function. The formula for the function is given on the CTANH order form. For the sample case, we fit six segments to an average profile derived from in-situ ocean

FORM TO SPECIFY INPUT DATA FOR
CURRENT-VELOCITY MODEL VVORTX3

This subroutine models a vortex with a viscous core and a Gaussian intensity profile in the vertical. The axis of the vortex is vertical and may be positioned above any geographic latitude and longitude. The vortex rotates anticlockwise looking down. The core (inside r_o) is essentially a solid-rotating fluid, while outside r_o, $|u|$ falls off as the inverse radius.

$$u_\theta = -\frac{1.397\ R_e U_o r_o}{r^2}\left(1 - e^{-1.26\ r^2/r_o^2}\right)(\phi - \phi_o)e^{-\left(\frac{h - h_{max}}{w_H}\right)^2}$$

$$u_\phi = \frac{1.397\ R_e U_o r_o}{r^2}\left(1 - e^{-1.26\ r^2/r_o^2}\right)(\theta - \theta_o)e^{-\left(\frac{h - h_{max}}{w_H}\right)^2},$$

where $\theta_o = \pi/2 - \lambda_o$ and r is the radial distance from the vortex center. The numerical constants normalize the function so that $|U| = U_o$ at $r = r_o$. R_e is the radius of the Earth, θ is the colatitude, ϕ is the longitude, and h is the height above sea level.

Specify--

the model check for VVORTX3 = 9.0 (W100)

the input data-format code = _____ (W101)

an input data-set identification number = 3. (W102)

an 80-character description of the model, including description of parameter values:

VORTEX AT LONGITUDE 150 KM E, UMAX = 1.02 M/S, R = 50 KM

the maximum tangential current, U_o = .00102 (km/s) m/s (W103)

the radius of the vortex core (to $u = U_o$), r_o = 50. km (W104)

the latitude of the vortex center, λ_o = 0. rad, deg, km N (W105)

the longitude of the vortex center, ϕ_o = 150. rad, deg, (km) E (W106)

the Gaussian width in height of the vortex, W_H = 1. (km) m (W107)

the height of the vortex, h_{max} = −1. (km) m (W108)

OTHER MODELS REQUIRED: Any current-perturbation model. Use NPCURR if no perturbation is desired.

Figure 2.2. Sample of completed form to specify ocean current model VVORTX3.

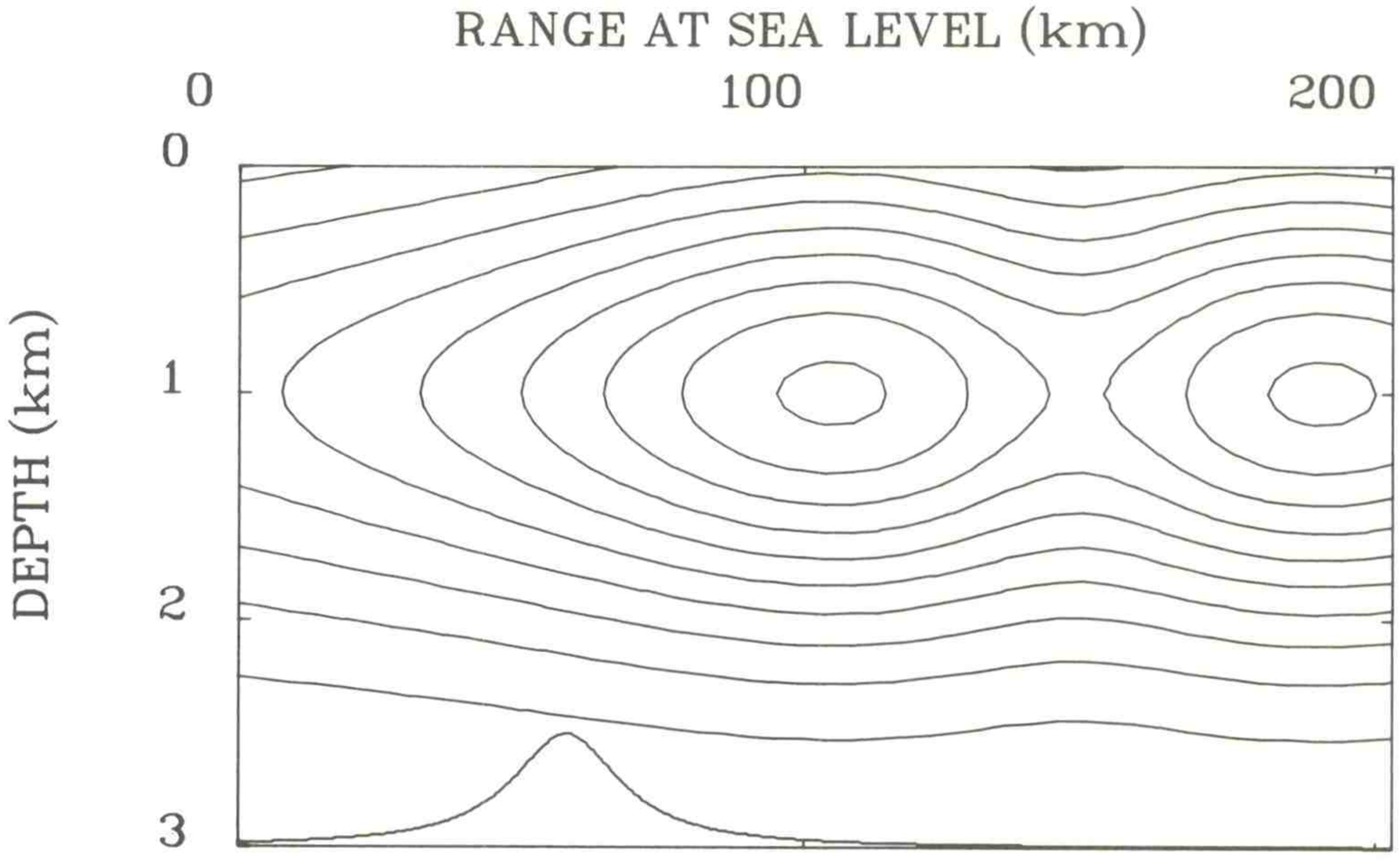


Figure 2.3. Contours of ocean current for model VVORTX3 in the vertical ray-transmission plane (Fig. 2.18).

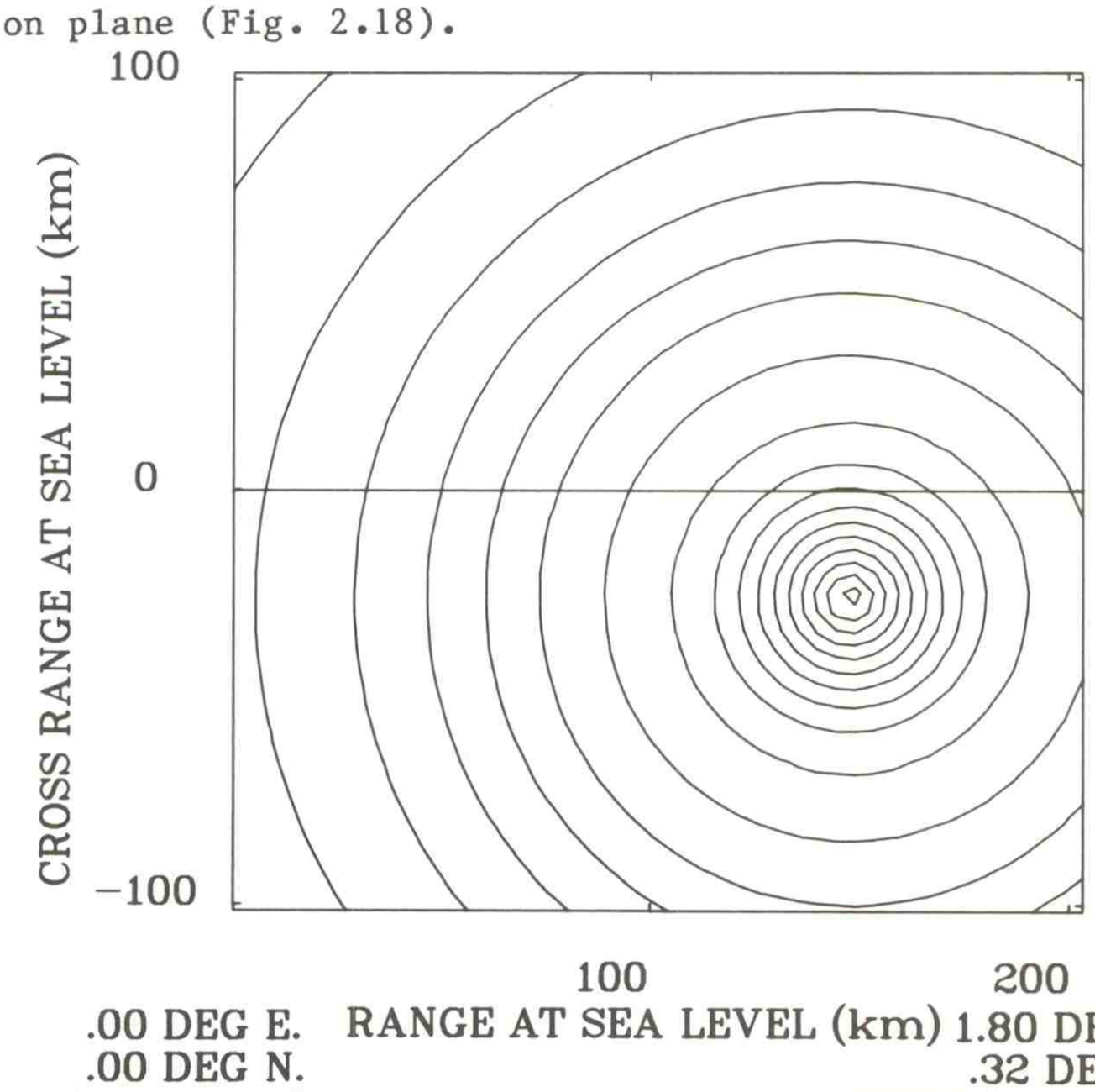


Figure 2.4. Contours of ocean current for model VVORTX3 in a horizontal plane at a depth of 1 km.

FORM TO SPECIFY INPUT DATA FOR BACKGROUND
SOUND-SPEED MODEL CTANH

This model represents the sound-speed profile by a sequence of linear segments that are smoothly joined by hyperbolic functions:

$$C = C_o + \frac{b_1}{2}(z-z_o) + \sum_{i=1}^{n} \delta_i \left(\frac{b_{i+1} - b_i}{2}\right) \ln \left\{ \frac{\cosh\left(\frac{z-z_i}{\delta_i}\right)}{\cosh\left(\frac{z_i-z_o}{\delta_i}\right)} \right\} + \frac{b_{n+1}}{2}(z-z_o)$$

$$\frac{dC}{dz} = b_1 + \sum_{i=1}^{n} \left(\frac{b_{i+1} - b_i}{2}\right) \left\{\tanh\left(\frac{z-z_i}{\delta_i}\right) + 1\right\}$$

$$b_i = (C_i - C_{i-1})/(z_i - z_{i-1}) \quad .$$

$z = r - r_e$, where r_e is the Earth radius, and r is the radial coordinate of the ray point. Thus, δ_i is the half-thickness of a region centered at approximately z_i km, in which dC/dz changes from b_i to b_{i+1}. Start by drawing a profile with linear segments, and get C_i and z_i from the corners. Then select δ_i to round the corners. The final profile will not go through (C_i, z_i).

Specify--

the model check for CTANH = 7.0 (W150)

the input data-format code = _____(W151)

an input data-set identification number = 1. (W152)

an 80-character description of the model with parameters:

EL NINO BACKGROUND SOUND-SPEED PROFILE

and the profile values:

the number of points in the profile -2 = n = 5

the profile:

i	z_i (km, m)	C_i (km/s, m/s)	δ_i (km, m)
0	0	1532.	0
1	-20.	1531.5	-7.
2	-50.	1509.	-20.
3	-250.	1503.	-40.
4	-450.	1485.	-300.
5	-1500.	1485.	-400.
6	-3000.	1508.	0

OTHER MODELS REQUIRED: Any sound-speed perturbation model. Use NPSPEED if no perturbation is desired. FUNCTION ALCOSH.

Figure 2.5. Sample of completed form to specify input data for ocean sound-speed model CTANH.

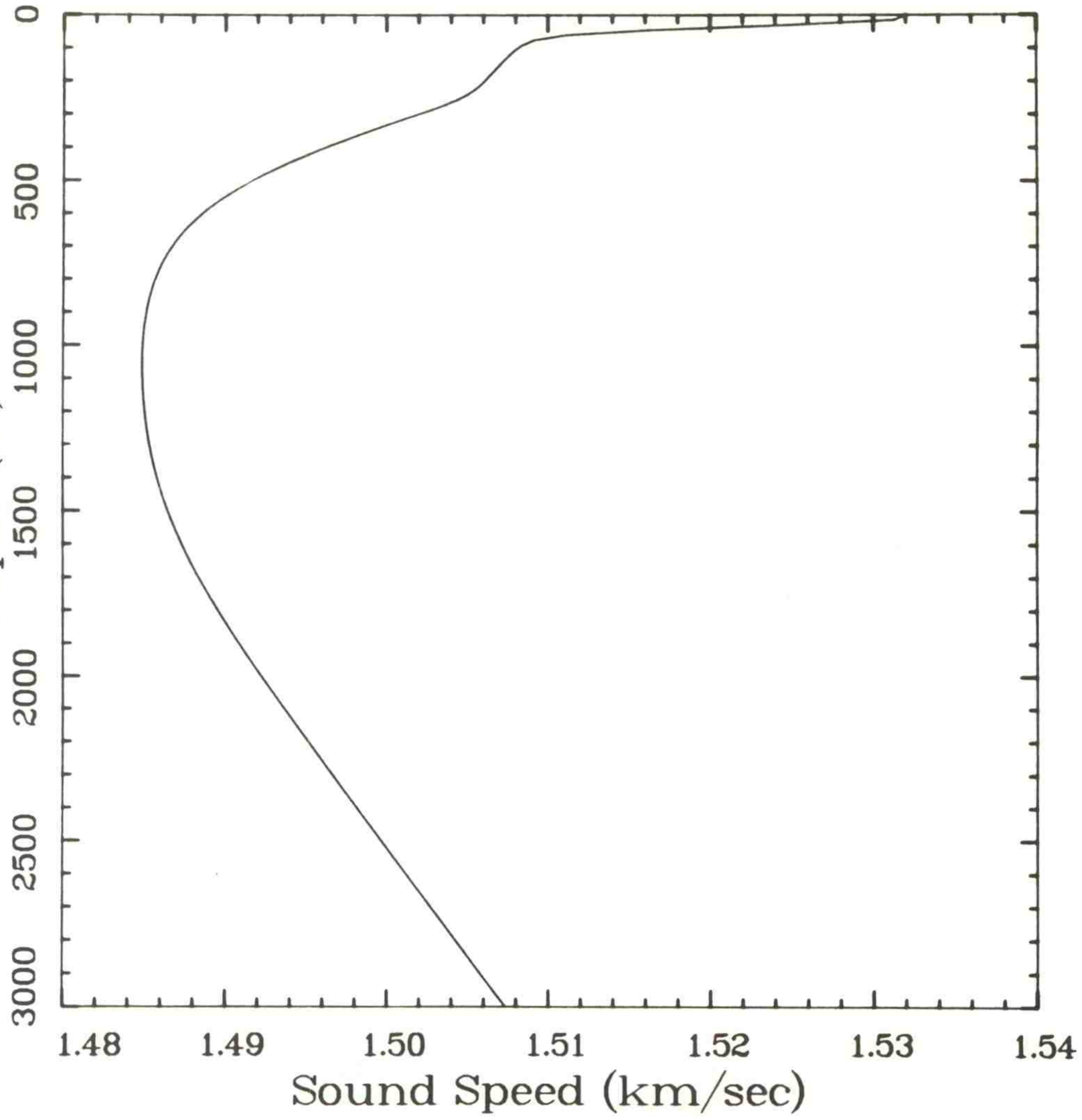


Figure 2.6. Sound-speed profile for model CTANH with parameters used in the sample case.

measurements made off the coast of Ecuador during the EPOCS exercise (NOAA, 1985). The resulting model profile is shown in Figure 2.6.

Refer next to the FORM TO SPECIFY INPUT DATA FOR SOUND-SPEED PERTURBATION MODEL CBLOB2 (Figure 2.7). CBLOB2 represents a local increase (or decrease) of sound speed that decays in a gaussian manner in all three spatial dimensions and can be located anywhere in the ocean. The perturbation model is superimposed on the background sound-speed model.

The user must specify seven input parameters: Δ, the strength of the fractional increase in squared sound speed, z_o, the height of the maximum effect; λ_o, the latitude of the maximum effect, ϕ_o, the longitude of the maximum effect; W_z, the gaussian width in height; W_θ, the gaussian width in the north-south direction; and W_ϕ, the gaussian width in the east-west direction. If any of the W's is given a zero value, the result is that there is no spatial variation in that direction. The formula is given on the CBLOB2 order form. For the sample case, we select $\Delta = 0.02$, $z_o = -1$ km, $\lambda_o = 0$, $\phi_o = 50$ km, $W_Z = 1$ km, $W_\theta = 50$ km, and $W_\phi = 50$ km. The location and scale of this perturbation are designed to coincide with the VVORTX3 current model just described. Figures 2.8 and 2.9 show total-sound-speed contours (including the background model) for the sample case.

Refer next to the FORM TO SPECIFY INPUT DATA FOR OCEAN ABSORPTION MODEL SLLOSS (Figure 2.10). This is a simple empirical model of absorption that depends only on the acoustic frequency according to the formula given on the SLLOSS order form. It requires no other ocean models, but other absorption models could be devised that depend on the concentrations of ocean constituents. Figure 2.11 shows the frequency dependence of this absorption model.

Refer next to the FORM TO SPECIFY INPUT DATA FOR OCEAN BOTTOM MODEL GLORENZ (Figure 2.12). This bottom model superimposes a Lorentzian ridge on a spherical surface at any height. The ridge, defined by the formula on the GLORENZ order form, runs along a latitude line, chosen to be 10 km N for the sample case. The other input parameters are z_o, the height of the ridge, chosen to be 0.5 km, and $\Delta\theta$, the width of the ridge, chosen to be 10 km for the sample case, and z_B, the height of the base of the ridge, chosen to be -3 km for the sample case.

FORM TO SPECIFY INPUT DATA FOR SOUND-SPEED
PERTURBATION MODEL CBLOB2

An increase (or decrease) in sound speed in a localized region that decays in a Gaussian manner in all three spatial directions.

$$C^2(r,\theta,\phi) = C_o^2(r,\theta,\phi)\left(1 + \Delta \exp\left\{-\left(\frac{z-z_o}{W_z}\right)^2 - \left(\frac{\theta-\theta_o}{W_\theta}\right)^2 - \left(\frac{\phi-\phi_o}{W_\phi}\right)^2\right\}\right)$$

$C_o^2(r,\theta,\phi)$ is the square of the sound speed specified by a sound-speed model. (r,θ,ϕ) are the coordinates of the ray point in an Earth-centered spherical polar-coordinate system. $\theta_o = \pi/2 - \lambda_o$ and $z = r - r_e$, where r_e is the Earth radius.

Specify--

the model check for subroutine CBLOB2 = 2.0 (W175)

the input data-format code = _____ (W176)

an input data-set identification number = 7. (W177)

an 80-character description for the sound-speed perturbation model, including description of parameter values:

2% INCREASE IN C SQUARED AT 150 KM RANGE, 1 KM DEPTH, 50 KM WIDE

the strength of the fractional increase (or decrease), Δ = .02 (W178)

the height of maximum effect, z_o = -1. km (W179)

the latitude of maximum effect, λ_o = 0 rad, deg, km N (W180)

the longitude of maximum effect, ϕ_o = 150. rad, deg, (km) E (W181)

the Gaussian width in height of the effect, W_z = 1. km (W182)*

the meridional width of the effect, W_θ = 50. rad, deg, (km) (W183)*

the zonal width of the effect, W_ϕ = 50. rad, deg, (km) (W184)*

OTHER MODELS REQUIRED: none.

---

* Setting W_z, W_θ, or W_ϕ = zero results in no space variation in that direction.

Figure 2.7. Sample of completed form to specify sound-speed perturbation model CBLOB2.

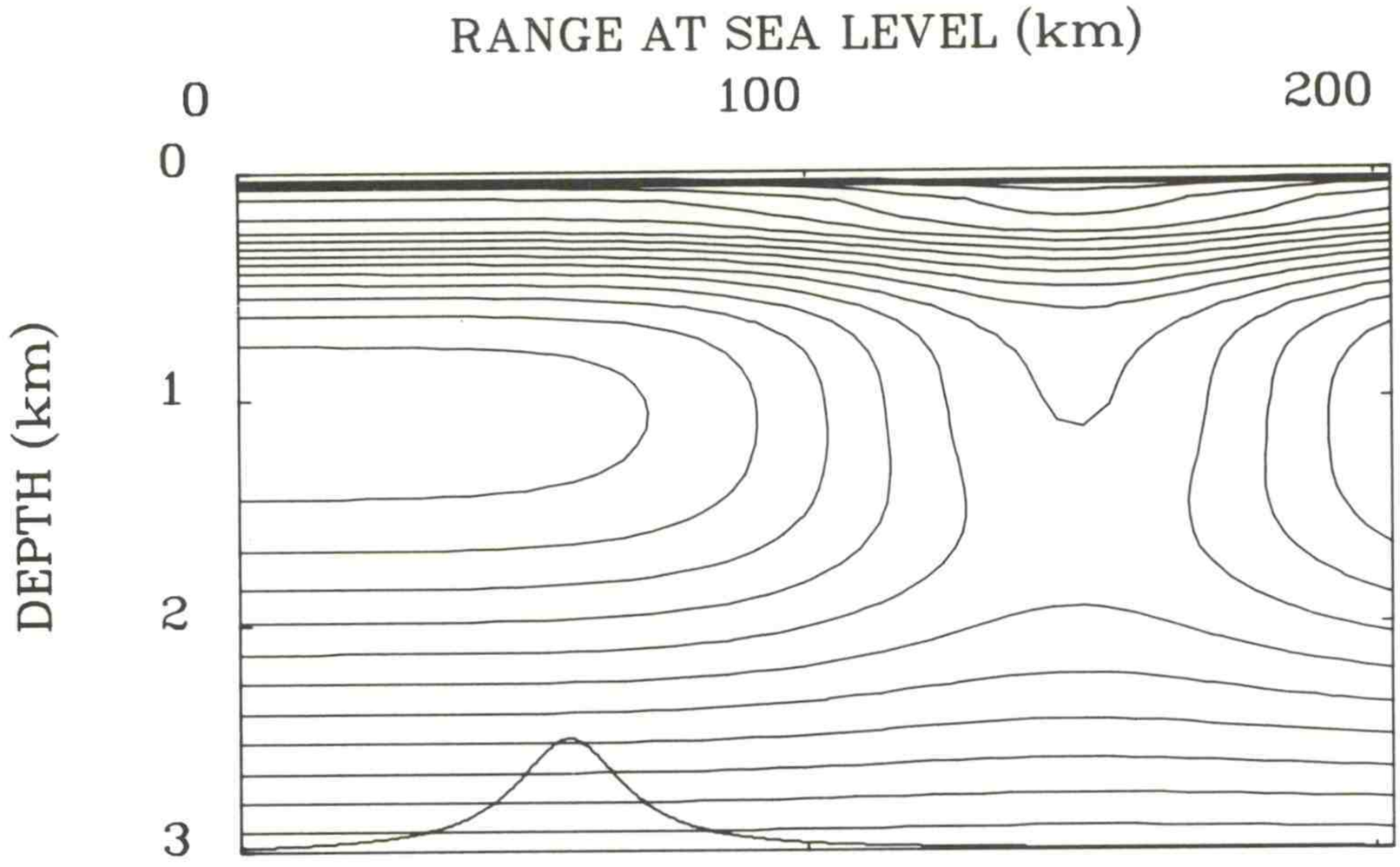


Figure 2.8. Contours of total sound speed in the vertical ray-transmission plane (Figure 2.18), using models CTANH and CBLOB2.

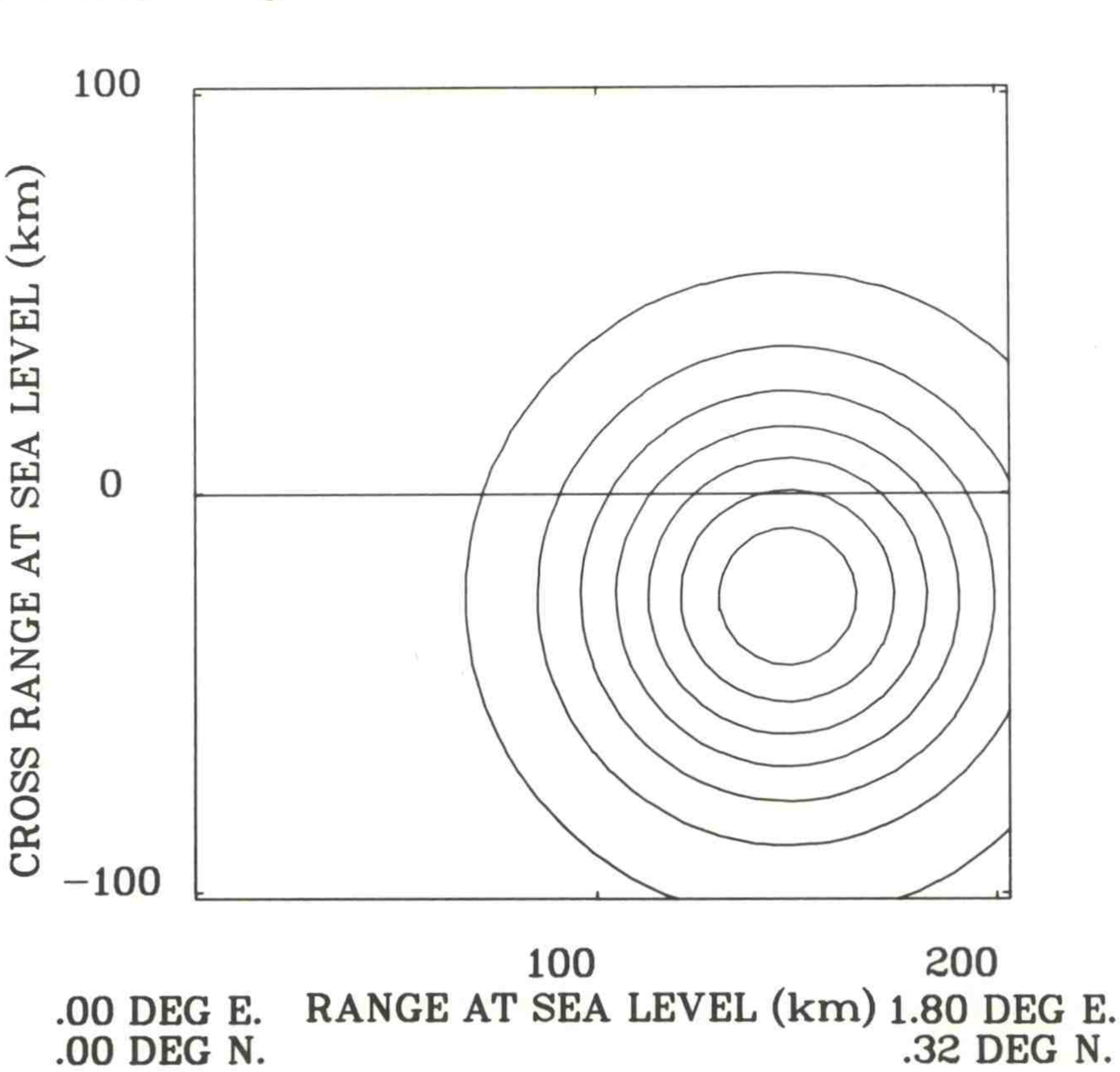


Figure 2.9. Contours of total sound speed in a horizontal plane at 1 km depth, corresponding to the sound-speed models of the sample case.

FORM TO SPECIFY INPUT DATA FOR
OCEAN ABSORPTION MODEL SLLOSS

This absorption model depends only on the acoustic wave frequency ω (rad/s), according to the formula

$$\alpha = a \frac{\omega^2}{\omega_1^2} + b \frac{\omega^2}{\omega_2^2 + \omega^2}$$

The following values for the coefficients correspond to the model of Skretting and Leroy (1971)*:

a = 0.006 dB/km ; ω_1 = 6283.2 rad/s (= 1000.0 Hz)

b = 0.2635 dB/km ; ω_2 = 10,681.4 rad/s (= 1700.0 Hz)

Specify--

the model check for SLLOSS = 1. (W500)

the input data-format code = _____ (W501)

an input data-set identification number = 1.0 (W502)

an 80-character description of the model, including parameter values:

SKRETTING- LEROY ABSORPTION FORMULA

a = 0.006 nepers/km, (dB/km) (W503)

b = 0.2635 nepers/km, (dB/km) (W504)

ω_1 = 1000 rad/s (Hz) (505)

ω_2 = 1700 rad/s (Hz) (W506)

Other models required: any absorption perturbation model. Use NPABSR if no perturbation is desired.

*see References.

Figure 2.10. Sample of completed form to specify absorption model SLLOSS.

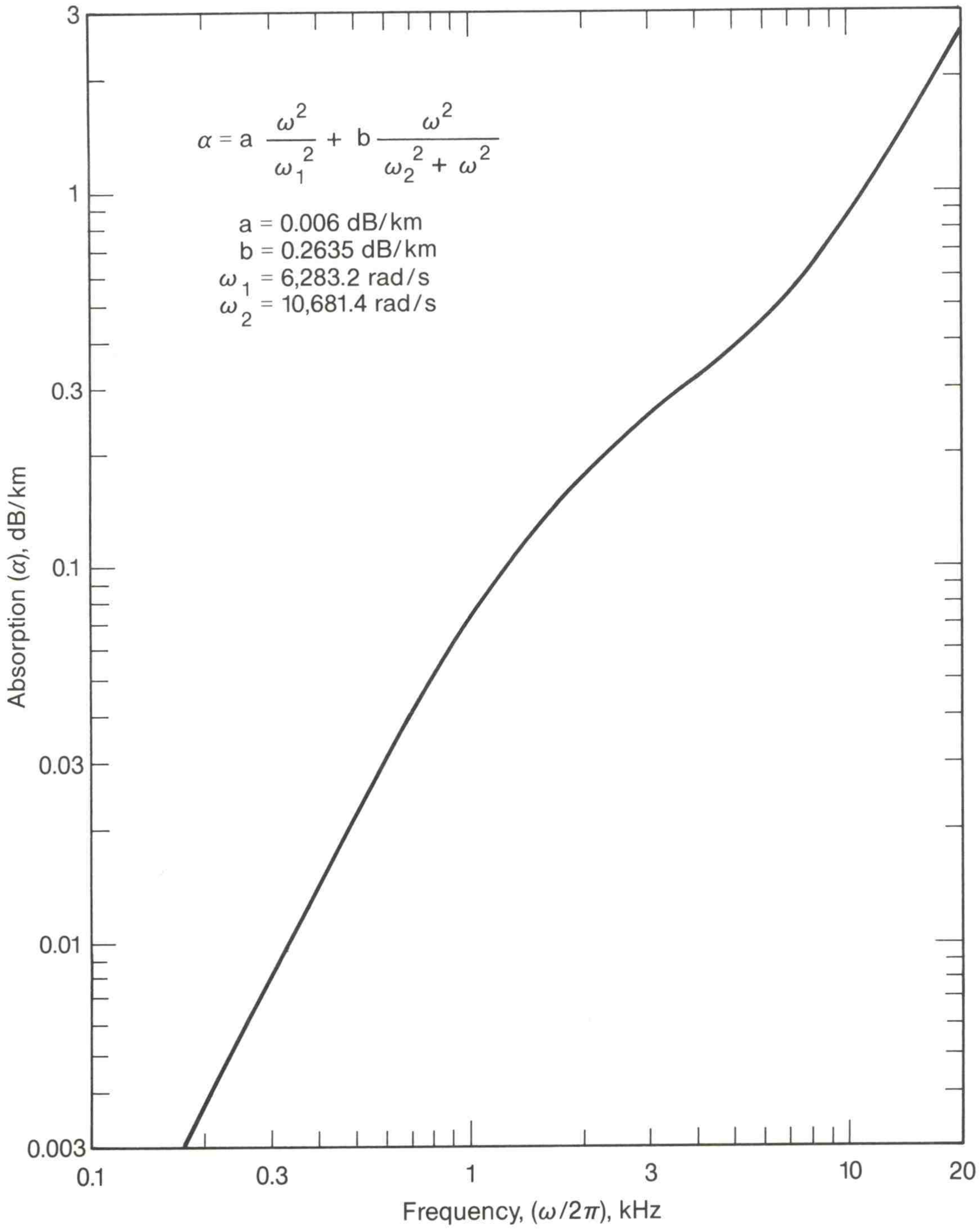


Figure 2.11. Frequency dependence of acoustic absorption with model SLLOSS, using the parameters indicated.

FORM TO SPECIFY INPUT DATA FOR
BOTTOM MODEL GLORENZ

An east-west Lorentzian-shaped ridge.

$g(r,\theta,\phi) = h - z$,

where $h = r - r_e$,

$z = z_o/(1 + ((\theta-\theta_o)/\Delta\theta)^2) + z_B$

$\theta_o = \pi/2 - \lambda_o$,

and r_e is the radius of the Earth.

Specify --

the model check number for GLORENZ = 4.0 (W300)
the data input format code number = _____ (W301)
the data set identification number = 3. (W302)
an 80-character description of the model including parameters:
RIDGE .5 KM HIGH, 2 KM WIDE AT 10KM LATITUDE, BASE = -3 KM
the height of the ridge, z_o = .5 (km) m (W303)
the latitude of the ridge center, λ_o = 10. rad, deg, (km) (W304)
the half-width of the ridge, $\Delta\theta$ = 2. rad, deg, (km) (W305)
the height above sea level of the
base of the ridge (negative if below sea level) z_B = -3. m (km) (W306)

OTHER MODELS REQUIRED: Any bottom perturbation model. Use NPBOTM if no perturbation is desired.

Figure 2.12. Sample of completed form to specify ocean bottom model GLORENZ.

Refer next to the FORM TO SPECIFY INPUT DATA FOR OCEAN SURFACE MODEL SHORIZ (Figure 2.13). This model specifies a spherical ocean surface at a specified height above mean sea level. For the sample case, the height is zero.

2.2 The Ray-Tracing Order Form for the Sample Case

Refer now to the FORM TO SPECIFY DATA FOR A THREE-DIMENSIONAL RAYPATH CALCULATION (Figure 2.14). The form has been filled out with values selected for the sample case. Let's go through each parameter specification on this form:

We choose a transmitter height of 2.0 km above the ocean bottom (as specified in the GLORENZ bottom model), at a latitude of zero, and a longitude of zero. (Alternatively, a transmitter depth below mean sea level (MSL, $r = r_e$) could be specified.) The acoustic frequency is 400 Hz, with no stepping in frequency. The azimuth angle of transmission is 80° (clockwise from north), with no stepping in azimuth. The elevation angle of transmission is stepped from 2° to 16° in steps of 2°. (If azimuth and frequency stepping are used, elevation angle stepping is performed first, then azimuth angle, then frequency.)

We want to keep track of all ray intersections with a receiver surface 1 km below MSL, and we want to stop tracing rays that go below 5 km depth or above 5 km above MSL. (Actually, neither event should happen unless something goes wrong in the bottom- or surface-reflection logic, so this is just a precaution.) We want to stop rays that go beyond 50 hops, a hop being defined as an intersection with the receiver surface, and those that go beyond 210 km range, measured at mean sea level.

Because we have selected a receiver surface at a fixed height above MSL, we also fill out the FORM TO SPECIFY INPUT DATA FOR RECEIVER SURFACE MODEL RHORIZ (Figure 2.15) and specify a height of -1.0 km. If we had wanted a receiver surface at a fixed height above the bottom, we would have specified the receiver surface model RBOTM.

FORM TO SPECIFY INPUT DATA
FOR OCEAN SURFACE MODEL SHORIZ

An ocean surface model that is a horizontal (i.e., a sphere concentric with the earth). The ocean surface is where the following function is zero.

$$s(r,\theta,\phi) = z_s - h$$

where

$$h = r - r_e$$

and

r_e is the earth radius

Specify--

the model check number for subroutine SHORIZ = 1.0 (W350)

the data input format code number = _____ (W351)

the data set identification number = 1.0 (W352)

an 80-character description of the model including parameters:

OCEAN SURFACE = SPHERE AT MSL

the height of the ocean surface above

mean sea level, z_s = 0.0 km, m (W353)

OTHER MODELS REQUIRED: Any surface-perturbation model. Use NPSURF if no perturbation is desired.

Figure 2.13. Sample of completed form to specify ocean surface model SHORIZ.

FORM TO SPECIFY INPUT DATA FOR A
THREE-DIMENSIONAL RAYPATH CALCULATION

Name _____
Ocean ID (3 characters) NØ1 Date AUG 19, 86

Title (77 characters) SAMPLE CASE FOR HARPO DOCUMENTATION

Transmitter: Height	2.	(km), nm, ft	(W3)
	✓	above bottom	
		above sea level	
Latitude	0.	rad, deg, km	(W4)
Longitude	0.	rad, deg, km	(W5)
Frequency, initial	400.	rad/s, (Hz), s	(W7)
final			(W8)
step			(W9)
Azimuth angle, initial	80.	rad, (deg) clockwise of north	(W11)
final			(W12)
step			(W13)
Elevation angle, initial	2.	rad, (deg)	(W15)
final	16.		(W16)
step	2.		(W17)
Receiver: Height	-1.	km, nm, ft	(W20)
	✓	above sea level (rcvr model RHORIZ)	
		above bottom (rcvr model RBOTM)	
Distance from origin		rad, deg, km	(W278) (rcvr model RVERT)
Latitude of origin		rad, deg, km	(W279) (rcvr model RVERT)
Longitude of origin		rad, deg, km	(W280) (rcvr model RVERT)
Stop elevation-angle stepping when ray goes out of bounds	0.		(W21 = 1.)
Stop ray if it hits the bottom	0.		(W19 = 1.)
Maximum height	5.	km	(W26)
Minimum height	-5.	km	(W27)
Maximum range	210.	km	(W28)
Maximum number of hops	50.		(W22)
Maximum number of steps per hop	1000.		(W23)
Maximum allowable error per step	10^{-6}		(W42)

Additional calculations: = 1. to integrate
= 2. to integrate and print

Phase path	2.	(W57)
Absorption	2.	(W58)
Doppler shift	0.	(W59)
Path length	2.	(W60)

Printout: Every 50. steps of the ray trace (W71)

Computer readable output (raysets):	1.	(W72 = 1.)
Diagnostic printing:	0.	(W73 = 1.)
Suppress all printout	0.	(W74 = 1.)

Figure 2.14. Sample of completed form to specify a three-dimensional raypath calculation.

FORM TO SPECIFY INPUT DATA
FOR RECEIVER-SURFACE MODEL RHORIZ

A receiver-surface model that is a horizontal surface (i.e., a sphere concentric with the Earth).

$$f(r,\theta,\phi) = h - z_R \ ,$$

where

$$h = r - r_e$$

and

r_e is the Earth radius

$$\frac{\partial f}{\partial t} = \frac{\partial f}{\partial \theta} = \frac{\partial f}{\partial \phi} = 0$$

$$\frac{\partial f}{\partial r} = 1.0 \quad .$$

Specify--

the model check number for subroutine RHORIZ = 1.0 (W275)

the input data-format code number = _____ (W276)

an 80-character description of the model including parameters:

RECEIVER SURFACE = SPHERE 1 KM BELOW MSL

the receiver surface height, z_R = -1. km (W20)

OTHER MODELS REQUIRED: none.

Figure 2.15. Sample of completed form to specify receiver surface model RHORIZ.

We set the maximum number of steps per hop to 1000 (usually a large number that we don't expect to be exceeded under normal conditions but which guards against "accidents"). We set the maximum allowable integration error per step to 10^{-6}, which means that integrated quantities are computed with at least that relative accuracy. We want to integrate and print phase path, path length and absorption, but not to calculate Doppler shift (which would be zero for the sample case, which has no time-dependent models). The printed output will display the raypath status every 50th step, in addition to printing at special events, such as reflections, apogees and perigees (turning points). Machine-readable "raysets" will also be produced.

2.3 Rayplot Order Form for the Sample Case

Two kinds of raypath plots are available and are specified for the sample case on the FORM TO SPECIFY INPUT PARAMETERS FOR PLOTTING A PROJECTION OF THE RAY PATH (Figure 2.16 and Figure 2.17). The same form is used twice: once for a projection of raypaths on a vertical plane, and once for a projection on a horizontal plane.

The vertical plane is specified by the geographic coordinates of its left and right edges and the height above MSL for the bottom of the graph. In the sample case, we want the left edge to be at latitude zero, longitude zero; the right edge is to be at latitude 35.27 km (north), longitude 200 km (east). This makes the plane of the vertical projection coincide with the plane of initial ray transmission (azimuth 80°). The bottom of the graph is to be at -3 km, and the top of the graph is at sea level (0 km). Because rays in the ocean are nearly horizontal, rayplots are easier to see if we expand the vertical dimension. We select an expansion factor of 40. We want tick marks every 100 km along the horizontal axis, and every 1 km along the vertical.

The horizontal projection plane is specified by the location of the centers of its left and right edges. For the sample case, we select the left and right edges of the horizontal plot to coincide with the coordinates of the left and right edges of the vertical plot, as specified above.

FORM TO SPECIFY INPUT PARAMETERS FOR PLOTTING A PROJECTION OF THE RAYPATH

Model ID: N01

Plot directly during raypath calculations ✓, or

plot from precomputed raypaths _____

in disk file _____

Normal or apogee plots: Normal ✓ (W80=0.0)

Plot apogees only _____ (W80=1.0)

Projection:

Vertical plane, polar plot, rectangular expansion _____ (W81=1.0)

Horizontal plane, lateral expansion _____ (W81=2.0)

Vertical plane, polar plot, radial expansion _____ (W81=3.0)

Vertical plane, rectangular plot ✓ (W81=4.0)

Superimpose these raypath plots on the graph of the previous runset:

Yes _____ (W81 negative.)

No ✓ (W81 positive.)

Vertical or lateral expansion factor 40. (W82)

Coordinates of the left edge of the graph:

Latitude = 0. (rad, deg, km) north (W83)

Longitude = 0. (rad, deg, km) east (W84)

Coordinates of the right edge of the graph:

Latitude = 35.1 (rad, deg, (km)) north (W85)

Longitude = 200. (rad, deg, (km)) east (W86)

Distance between horizontal tick marks = 50. rad, deg, (km) (W87)

Height above sea level of bottom of graph = -3. km (W88)

Height above sea level of top of graph = 0. km (W89)

Distance between vertical tick marks = /. km (W96)

Figure 2.16. Sample of completed form to specify a vertical rayplot projection.

FORM TO SPECIFY INPUT PARAMETERS FOR PLOTTING A PROJECTION OF THE RAYPATH

Model ID: NØ1

Plot directly during raypath calculations ✓, or

plot from precomputed raypaths ____

in disk file ____

Normal or apogee plots:

Normal ✓ (W80=0.0)

Plot apogees only ____ (W80=1.0)

Projection:

Vertical plane, polar plot, rectangular expansion ____ (W81=1.0)

Horizontal plane, lateral expansion ✓ (W81=2.0)

Vertical plane, polar plot, radial expansion ____ (W81=3.0)

Vertical plane, rectangular plot ____ (W81=4.0)

Superimpose these raypath plots on the graph of the previous runset:

Yes ____ (W81 negative.)

No ✓ (W81 positive.)

Vertical or lateral expansion factor 40. (W82)

Coordinates of the left edge of the graph:

Latitude = 0. (rad, deg, km) north (W83)

Longitude = 0. (rad, deg, km) east (W84)

Coordinates of the right edge of the graph:

Latitude = 35.1 (rad, deg, (km) north (W85)

Longitude = 200. (rad, deg, (km) east (W86)

Distance between horizontal tick marks = 50. rad, deg, (km) (W87)

Height above sea level of bottom of graph = -1. km (W88)

Height above sea level of top of graph = 0. km (W89)

Distance between vertical tick marks = 1. km (W96)

Figure 2.17. Sample of completed form to specify a horizontal rayplot projection.

Figure 2.18 shows a plan view of the region of the earth's surface near latitude zero, longitude zero, including the transmitter location, the ray-launch azimuth, the locations of the centers of the CBLOB2 and VVORTX3 perturbations, the axis of the ridge at 10 km N, and the locations of the left and right edges of the two plot projections (L and R).

2.4 Setting Up the Input Data File (W Array) for the Sample Case

Now that we have defined all the parameters needed to specify the ocean model, the raypaths desired, as well as the printed, plotted and machine-readable output, we can look at how these input data are communicated to HARPO. To run HARPO, you have to create a file like the one shown for the sample case in Figure 2.19. (Such a file for the sample case comes with the program.) HARPO reads this file into an array named W(n), where n, the array subscript, is the first value on each line (columns 1-3), and W(n) is the second value on each line (columns 4-17).

The values of n corresponding to each input parameter are indicated on each of the input parameter forms we have just looked at. You will notice that not all values of n are listed in Figure 2.19 for the sample case; those not explicitly defined in the Input Data File assume an initial value that is usually (but not always) zero. The initialization scheme is explained in Section 5.3.1.

Besides the values of n and W(n), the table contains in columns 18-24 provisions for unit conversion on input, that is, for entering data in various units. For example, the notation AN KM means that a central-earth angle has been entered as a great-circle distance in kilometers and will be converted to radians by the program. For W(3), the height of the transmitter, a T in col. 17 implies that the height specified is a height above the bottom (as specified by the bottom model). Finally, in columns 25-80, descriptive comments identify the data for easy data entry.

The standard set of units used by the program are: angles in radians, distances in kilometers, and frequency in radians per second. It is important

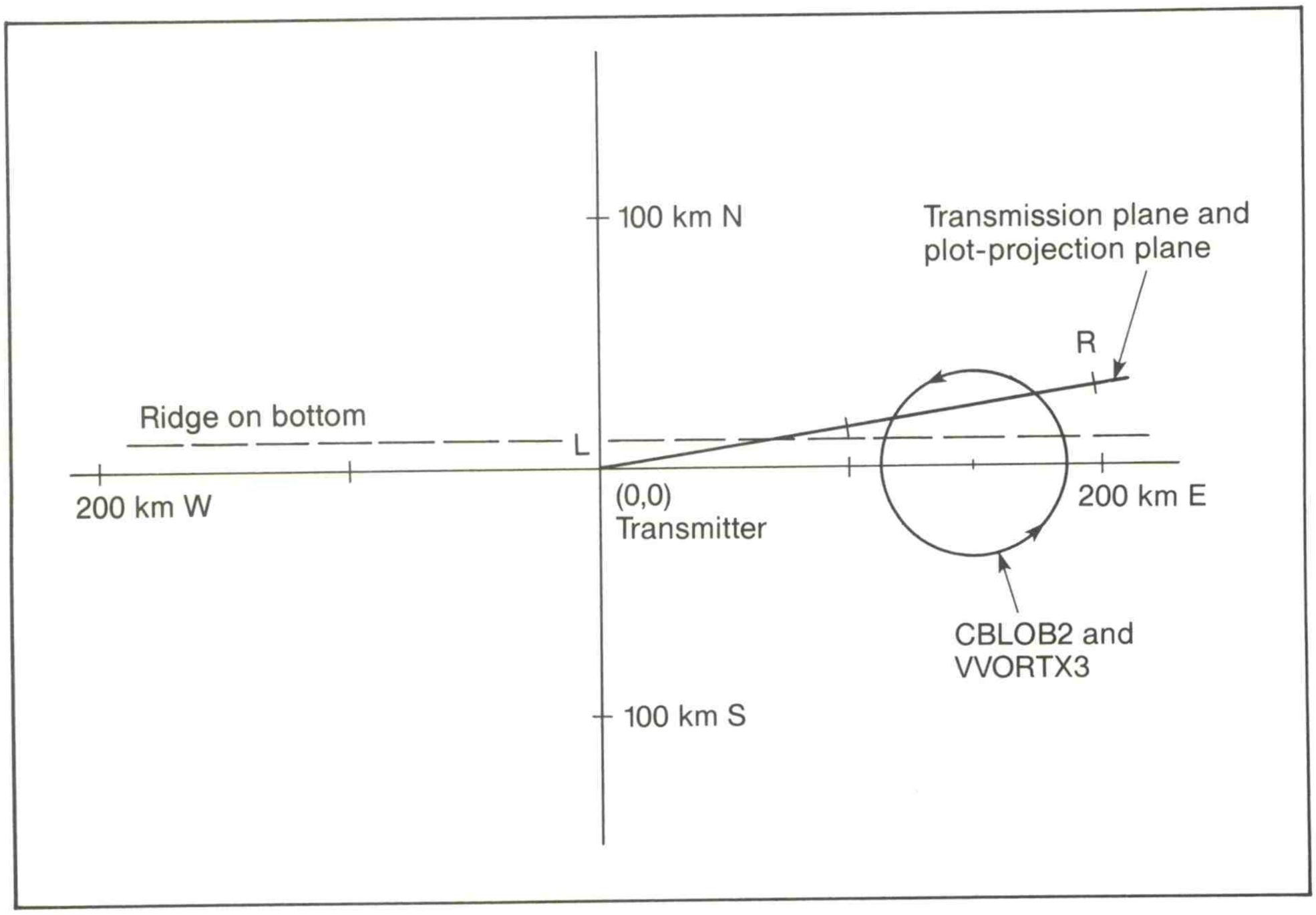


Figure 2.18. Plan view of the major features of the sample case model, the source location, the transmission plane, and the plot-projection plane.

```
col.     col.      col.        col.
1-3      4-17      18-24       25-80
 n       W(n)      UNITS       DESCRIPTION
```

```
GEORGES    RB3     X6437
N01-1       SAMPLE CASE FOR HARPO DOCUMENTATION          REV. 8-19-86
  1       6370.              EARTH RADIUS TO MSL, KM (6370.)
  3          2.            TTRANSMITTER HEIGHT ABOVE MSL (T=ABOVE BOTTOM), KM
  4          0.     AN KM   N. TRANSMITTER LATITUDE, KM
  5          0.     AN KM   E. TRANSMITTER LONGITUDE, KM
  7        400.     FQ HZ   INITIAL FREQUENCY, HZ
 11         80.     AN DG   INITIAL AZIMUTH ANGLE, DEG
 15          2.     AN DG   INITIAL ELEVATION ANGLE, DEG
 16         16.     AN DG   FINAL   ELEVATION ANGLE, DEG
 17          2.     AN DG   STEP IN ELEVATION ANGLE, DEG
 19          0.             STOP RAYS THAT STRIKE BOTTOM (1=YES; 0=NO)
 20         -1.             RECEIVER HEIGHT ABOVE MSL, KM
 22         50.             MAXIMUM NUMBER OF HOPS (1.)
 23       1000.             MAXIMUM NUMBER OF STEPS PER HOP (1000.)
 26          5.             MAXIMUM RAY HEIGHT ABOVE MSL, KM
 27         -5.             MINIMUM RAY HEIGHT ABOVE MSL, KM
 28        210.             MAXIMUM RANGE AT MSL, KM
 29          0.             DO: EIGRAY/RNG-TIM/RNG-ELV/NEW-PROJ/RAYTRC/CONT/PROF
 33         20.             MAXIMUM ABSORPTION, DB (999.999)
 42        1.0E-06          MAXIMUM SINGLE-STEP INTEGRATION ERROR (1.0E-4)
 44         .1              INITIAL INTEGRATION STEP SIZE, KM (1.0)
 57          2.             PHASE PATH  (0=NO; 1=INTEGRATE; 2=INTEGRATE/PRINT)
 58          2.             ABSORPTION  (0=NO; 1=INTEGRATE; 2=INTEGRATE/PRINT)
 60          2.             PATH LENGTH (0=NO; 1=INTEGRATE; 2=INTEGRATE/PRINT)
 71         50.             NUMBER OF INTEGRATION STEPS PER PRINT [1.E31]
 72          1.             OUTPUT RAYSETS (1=YES; 0=NO)
 73          0.             DIAGNOSTIC PRINTOUT (1=YES; 0=NO)
 74          0.             PRINT EVERY W(71) RAY STEPS (0=YES; 1=N0)
 75        .15              FULANN PLOT LETTERING; HEIGHT= [0.15 IN]
 76          0.             BINARY RAY OUTPUT (1=YES; 0=NO)
 77         57.             LINES PER PAGE OF PRINTOUT (57.)
 81          4.             RAYPLOT PROJECTION PLANE (4 = VERT. RECTANGULAR)
 82         40.              PLOT-ORDINATE EXPANSION FACTOR [1.]
 83          0.     AN KM    N. LATITUDE OF LEFT PLOT EDGE, KM
 84          0.     AN KM    E. LONGITUDE OF LEFT PLOT EDGE, KM
 85       35.1      AN KM    N. LATITUDE OF RIGHT PLOT EDGE, KM
 86       200.      AN KM    E. LONGITUDE OF RIGHT PLOT EDGE, KM
 87         50.     AN KM    DISTANCE BETWEEN RANGE TICKS, KM (0 = AUTO)
 88         -3.              HEIGHT ABOVE MSL OF BOTTOM OF GRAPH, KM
 89          0.              HEIGHT ABOVE MSL OF TOP OF GRAPH, KM
 96          1.              DISTANCE BETWEEN DEPTH TICKS, KM (0 = AUTO)
100          9.             VVORTX3 MODEL CHECK NUMBER
102          3.              VVORTX3 BACKGROUND CURRENT DATA SET ID
103       1.02       LN  M   MAXIMUM TANGENTIAL CURRENT, M/S
104         50.              RADIUS OF VORTEX CORE, KM
105          0.     AN KM    LATITUDE OF VORTEX CENTER, KM
106        150.     AN KM    LONGITUDE OF VORTEX CENTER, KM
107          1.              VERTICAL HALF-WIDTH OF VORTEX, KM
```

Figure 2.19. Input Data File for the sample case.

```
108        -1.                HEIGHT OF VORTEX CENTER ABOVE MSL, KM
150         7.               CTANH SOUND SPEED MODEL CHECK NUMBER
152         1.                CTANH BACKGROUND SOUND SPEED DATA SET ID
175         2.               CBLOB2 SOUND SPEED PERTURBATION MODEL CHECK NUMBER
177         7.                CBLOB2 PERTURBATION SOUND SPEED DATA SET ID
178        .02                MAXIMUM FRACTIONAL INCREASE IN C SQUARED
179        -1.                HEIGHT OF MAX EFFECT ABOVE MSL, KM
180         0.     AN KM      LATITUDE OF MAX EFFECT, KM
181       150.     AN KM      LONGITUDE OF MAX EFFECT, KM
182         1.                VERTICAL HALF-WIDTH, KM
183        50.     AN KM       N-S  HALF-WIDTH, KM
184        50.     AN KM       E-W  HALF-WIDTH, KM
275         1.               RHORIZ RECEIVER MODEL CHECK NUMBER
300         4.               GLORENZ BOTTOM MODEL CHECK NUMBER
302         3.                GLORENZ BOTTOM MODEL DATA SET ID
303        .5                 HEIGHT OF RIDGE, KM ABOVE BASE
304        10.     AN KM      N. LATITUDE OF RIDGE CENTER, KM
305         2.     AN KM      HALF-WIDTH OF THE RIDGE, KM
306        -3.                HEIGHT ABOVE MSL OF BASE OF RIDGE, KM
350         1.               SHORIZ MODEL CHECK NUMBER
352         1.                SHORIZ OCEAN SURFACE DATA SET ID
353         0.                HEIGHT OF OCEAN SURFACE ABOVE MSL, KM
500         1.               SLLOSS ABSORPTION MODEL CHECK NUMBER
502         1.                SLLOSS ABSORPTION DATA SET ID
503     0.006      AM DB      A COEFFICIENT, DB
504    0.2635      AM DB      B COEFFICIENT, DB
505     1000.      FQ HZ      OMEGA1, HZ
506     1700.      FQ HZ      OMEGA2, HZ
 -1                          DATA SUBSET FOR BACKGROUND CURRENT MODEL
 A  VORTEX AT LONGITUDE 150 KM E, UMAX= 1.02 M/S, R= 50 KM
 0        RETURN TO W-ARRAY DATA SET
-2                           DATA SUBSET FOR PERTURBATION CURRENT MODEL
 A  NO CURRENT PERTURBATION
 0        RETURN TO W-ARRAY DATA SET
-3                           DATA SUBSET FOR BACKGROUND SOUND-SPEED MODEL
 A  EL NINO BACKGROUND SOUND-SPEED PROFILE
 3     999.0
  LN  M         LN  M         LN  M
  0.            1532.         0.
  -20.          1531.5        -7.
  -50.          1509.         -20.
  -250.         1503.         -40.
  -450.         1485.         -300.
  -1500.        1485.         -400.
  -3000.        1508.         0.
  999.0
 0        RETURN TO W-ARRAY DATA SET
-4                           DATA SUBSET FOR SOUND-SPEED PERTURBATION MODEL
 A  2% INCREASE IN C-SQUARED AT 150 KM LON., 1 KM DEPTH, 50 KM WIDE
 0        RETURN TO W-ARRAY DATA SET
-8                           DATA SUBSET FOR RECEIVER-SURFACE MODEL
 A  RECEIVER SURFACE = SPHERE 1 KM BELOW MSL
 0        RETURN TO W-ARRAY DATA SET
-9                           DATA SUBSET FOR BACKGROUND BOTTOM MODEL
 A  RIDGE .5 KM HIGH, 2 KM WIDE AT 10 KM N LATITUDE; BASE= -3 KM
```

Figure 2.19. Input Data File (continued).

```
  0       RETURN TO W-ARRAY DATA SET
-10                     DATA SUBSET FOR BOTTOM PERTURBATION MODEL
  A  NO BOTTOM PERTURBATION
  0       RETURN TO W-ARRAY DATA SET
-11                     DATA SUBSET FOR OCEAN SURFACE MODEL
  A  OCEAN SURFACE = SPHERE AT MSL
  0       RETURN TO W-ARRAY DATA SET
-12                     DATA SUBSET FOR OCEAN SURFACE PERTURBATION MODEL
  A  NO OCEAN SURFACE PERTURBATION
  0       RETURN TO W-ARRAY DATA SET
-17                     DATA SUBSET FOR OCEAN ABSORPTION MODEL
  A  SKRETTING-LEROY ABSORPTION FORMULA
  0       RETURN TO W-ARRAY DATA SET
-18                     DATA SUBSET FOR PERTURBATION ABSORPTION MODEL
  A  NO ABSORPTION PERTURBATION
  0       RETURN TO W-ARRAY DATA SET
  0                     *********  END OF RUN SET NUMBER 1  *********
N01-2     SAMPLE CASE FOR HARPO DOCUMENTATION          REV. 8-19-86
 71        0.           NUMBER OF INTEGRATION STEPS PER PRINT [1.E31]
 72        0.           OUTPUT RAYSETS (1=YES; 0=NO)
 73        1.           DIAGNOSTIC PRINTOUT (1=YES; 0=NO)
 81        2.           RAYPLOT PROJECTION PLANE (2 = HORIZONTAL)
 88       -1.           HEIGHT OF HORIZONTAL PLOT SECTION ABOVE MSL, KM
  0                     *********  END OF RUN SET NUMBER 2  *********
```

Figure 2.19. Input Data File (continued).

to remember to convert the units of the spherical coordinates θ or ϕ when they are entered as distances. For example, entering CBLOB2 longitude ϕ_o in kilometers requires you to use the AN KM unit conversion.

There are special values of n (such as zero or negative values) that contain instructions for interpreting what follows in the Input Data File. They will be described in detail in Chapter 5. A negative number in col. 1-3 indicate that tabular or text data follow. A zero in col. 1-3 indicates the end of tabular data or the end of a "run set," which is the name we give to the input data for a set of ray calculations. For example, the rays for one run set will all appear on a single ray plot. A new run set is necessary to create different plots or to change model parameters. In Figure 2.19 new run sets start with the lines that begin with "N01." Each of the two run sets for the sample case generates a different ray plot projection. Only the W values that change from the previous run set need be specified; the others remain unchanged from the previous run set.

For now, you need only be aware of this tabular procedure for entering data into HARPO; you may find it instructive to verify that the values entered into the order forms for the sample case correspond to the entries in the W(n) input data listing.

When you run the program with this input data set, three kinds of output will be produced: a step-by-step printed account of each ray's progress, plots of the raypaths on vertical and horizontal planes, and machine-readable data, including "raysets."

2.5 Printed Output for the Sample Case

Appendix A shows the complete printed output, or "printout," for the sample case. Sections 2.5.1 and 2.5.2 define the terms and quantities used in the printout.

Page 1 of the printout contains the program title block and a list of all the models used for the first run set. The model list includes subroutine names, a number identifying the set of parameters defining that particular model, and comments describing each model.

Pages 2-4 of the printout reproduce the Input Data File for run set number 1.

Page 5 of the printout is a list of n and W(n) for all nonzero W(n). The values of W(n) have been converted to the units used by the program; for example, angles (like latitude and longitude) input in kilometers or degrees have been converted to radians.

Pages 6-10 of the printout are in a columnar format that gives a step-by-step account of each ray's progress, with each line showing important raypath quantities at user-specified intervals along the raypath. The meanings of the quantities printed out are explained in Section 2.5.1. The user can specify how often along the ray a line is printed (we specified every 50 integration steps). In addition, a line is printed out every time a ray experiences a "special event," such as a bottom or surface reflection, a turnover (apogee) or turnunder (perigee), intersections with the receiver surface, or a few other events. Section 2.5.2 explains the exact meanings of all the special events.

Each ray calculation terminates when one of the termination conditions is met, such as the maximum number of hops or maximum range, whichever occurs first. In the sample case, all of the rays terminate because they reached the maximum range specified. At the end of each ray calculation, the printout shows how much CPU time was used for that ray computation (1.736 s, in this case).

Look at page 1 of the printout in Appendix A. Verify that the models indicated at the bottom of the page coincide with those we specified as input. Verify that the wave frequency and initial azimuth and elevation angles on page 6 are what we want. Look down the first (ERROR) column of the tabular printout (page 6) and verify that none of the numbers in this column greatly exceeds the maximum allowable single-step integration error, W(42), which for the sample case is 10^{-6}. This means that the numerical integration is proceeding correctly.

Look at the first entries in the ELEVATION columns and verify that the ray starts at the correct height above the bottom (2.00 km in this case), as well as the correct height above MSL (slightly above -1 km in this case, because of the bottom model). The RANGE column should begin with zero range from the transmitter and indicate an increasing range from the transmitter with successive steps.

A general idea of the sequence of events along this ray can be read in the EVENT column. These notations mark the special events along the raypath, which cause printout regardless of the step number. Reading down this column, we see that the first ray begins at the transmitter (XMTR), then executes a series of APOGEE, WAVE REV, RCVR, PERIGEE, etc., indicating ray ducting in the sound channel. The ray apogee is at a depth of 0.77 km and its perigee is at a depth of 1.38 km. Notice that the apogee and perigee depths change because of the model's range dependence. The ray stops when it reaches the specified maximum range (210 km).

The numbers in the AZIMUTH DEVIATION columns indicate that the ray deviates from the azimuth of transmission because of sound-speed and current

gradients. The ELEVATION ANGLE columns show changes in the local wave-normal direction and the elevation angle of the ray point measured from the transmitter.

PULSE TIME gives the time for a pulse or wave packet to reach that point, and PHASE TIME gives the time for a wave phase front to reach that point. The wave phase can be derived from PHASE TIME by multiplying by the wave frequency in appropriate units (cycles per second to get wave phase in cycles, etc.) and removing the integer part. The PATH LENGTH gives the physical length of the ray path (which cannot be directly measured).

The second run set shown in the sample-case printout uses the same initial ray conditions, but it changes the rayplot to a horizontal projection, and it adds diagnostic printout that can be useful for studying the details of ocean bottom or receiver-surface intersections.

2.5.1 Definitions of Quantities Listed in Printout (in Appendix A)

AZIMUTH ANGLE OF TRANSMISSION -- Azimuth angle (degrees clockwise from north) of the initial ray-launch direction.

ELEVATION ANGLE OF TRANSMISSION -- Elevation angle (degrees upward) between the initial ray-launch direction and local horizontal at the transmitter.

TRANSMITTER LATITUDE -- North geographic latitude of the transmitter.

TRANSMITTER LONGITUDE -- East geographic longitude of the transmitter.

FREQUENCY -- Acoustic wave frequency in hertz.

SINGLE-STEP ERROR -- Maximum allowable single-step integration error.

ERROR -- Normalized difference between the wave number k computed by numerical integration and k computed from the dispersion relation [Eq. (6.23)].

EVENT -- Nature of special event along the raypath (see Sec. 2.5.2).

HEIGHT -- Height of the ray point above mean sea level (or above the bottom).

RANGE -- Great-circle distance, measured at mean sea level, between the transmitter and the ray point.

AZIMUTH DEVIATION (XMTR) -- Azimuth angle of the direction of transmission in degrees clockwise from the great circle between the transmitter and the ray point.

AZIMUTH DEVIATION (LOCAL) -- Azimuth angle of the wave normal in degrees clockwise from the great circle between the transmitter and the ray point.

ELEVATION (XMTR) -- Elevation angle (degrees) of the ray point from local horizontal at the transmitter.

ELEVATION (LOCAL) -- Elevation angle (degrees) of the wave normal from the local horizontal at the ray point.

PULSE TIME -- The time (seconds) required for a wave packet (pulse) to travel from the transmitter to the ray point (Sec. 6.1).

PHASE TIME -- The time (seconds) required for a wave front to travel from the transmitter to the ray point (Sec. 6.1.1).

ABSORPTION -- Decrease in acoustic intensity (dB) from the transmitter to the ray point caused by volume dissipation only (Sec. 6.1.2).

PATH LENGTH -- Geometric length of the ray path (kilometers) from the transmitter to the ray point (Sec. 6.1.3).

2.5.2 Meanings of Special Events Along a Raypath

XMTR -- Ray is at the transmitter.

RCVR -- Ray is at the receiver surface.

BOTM REF -- Ray has reflected from the ocean bottom.

SURF REF -- Ray has reflected from the ocean surface.

APOGEE -- Ray has passed through a maximum in height.

PERIGEE -- Ray has passed through a minimum in height.

WAVE REV -- Vertical, southward, or eastward component of the wave-normal vector has changed sign.

MAX LAT -- Ray has passed through a maximum (or minimum) in latitude.

MAX LONG -- Ray has passed through a maximum (or minimum) in longitude.

EXTINC -- Absorption has exceeded the maximum allowable.

MAX HOP -- Ray has executed the requested number of hops (receiver-surface crossings).

MAX RANG -- Ray has exceeded the maximum allowable horizontal range.

MAX HT -- Ray has exceeded the maximum allowable height.

MIN HT -- Ray has gone below the minimum allowable height.

MIN DIST -- Ray has made a closest approach to the receiver surface.

BOTM ABS -- Total absorption of ray by ocean bottom.

ADDITIONAL EVENTS IN DIAGNOSTIC PRINTOUT

BACK UP0 -- At call to subroutine BACKUP.

BACK UP1 -- Before each numerical integration step in subroutine BACKUP.

GRAZE 0 -- At call to ENTRY point GRAZE in subroutine BACKUP.

GRAZE 1 -- Before each numerical integration step after ENTRY point GRAZE.

BACK UP2 -- After unsuccessfully trying to find a closest approach to the receiver surface.

BACK UP3 -- Before each numerical integration step after BACK UP2.

2.6 Rayplots for the Sample Case

We requested two rayplots, a projection on a vertical plane in rectangular coordinates and a projection on a horizontal plane. These two plots are shown in Figures 2.20 and 2.21. Because we selected the FULANN (full annotation) option in W(75), we have produced a plot with publication-quality lettering. This capability requires the DISSPLA plotting package.

Rayplots are often the most useful output from a ray-tracing program, particularly when the medium is complicated. Depending on the user's plotting and display facilities, rayplots can be produced on paper, microfilm, or video displays.

Because the ocean and bottom models used in the sample case cause the acoustic raypaths to behave in complicated ways, a few features of the two rayplot projections call for some explanation. Figure 2.18 shows how the planes of these plots are related to the transmitter location and the features of the ocean and bottom models. The letters L and R show the locations of the left and right edges of the two plots; the line connecting the L and R represents the plane of the vertical projection (Figure 2.20) as

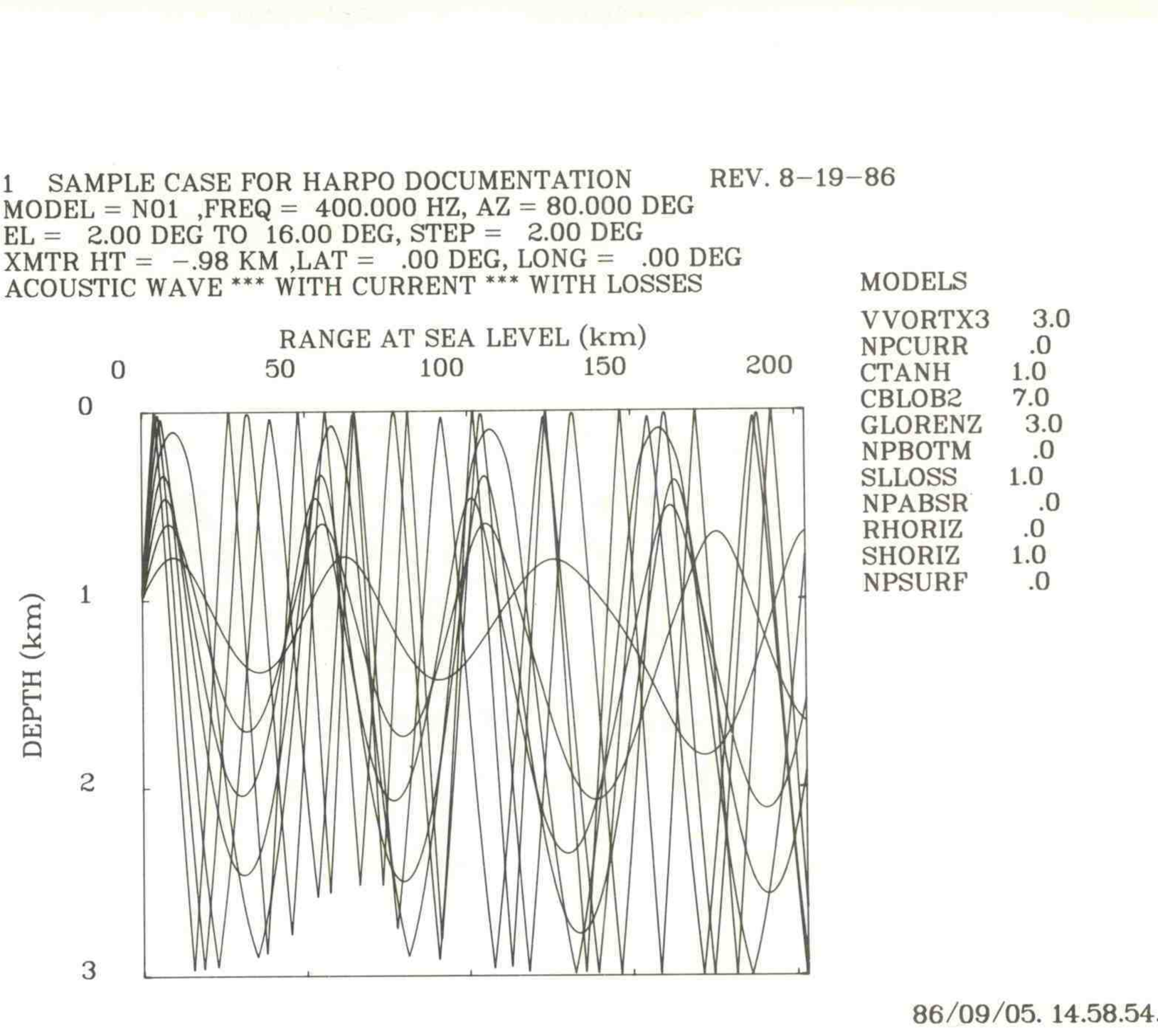


Figure 2.20. Projection of the rays of the sample case onto the vertical plane shown in Figure 2.18.

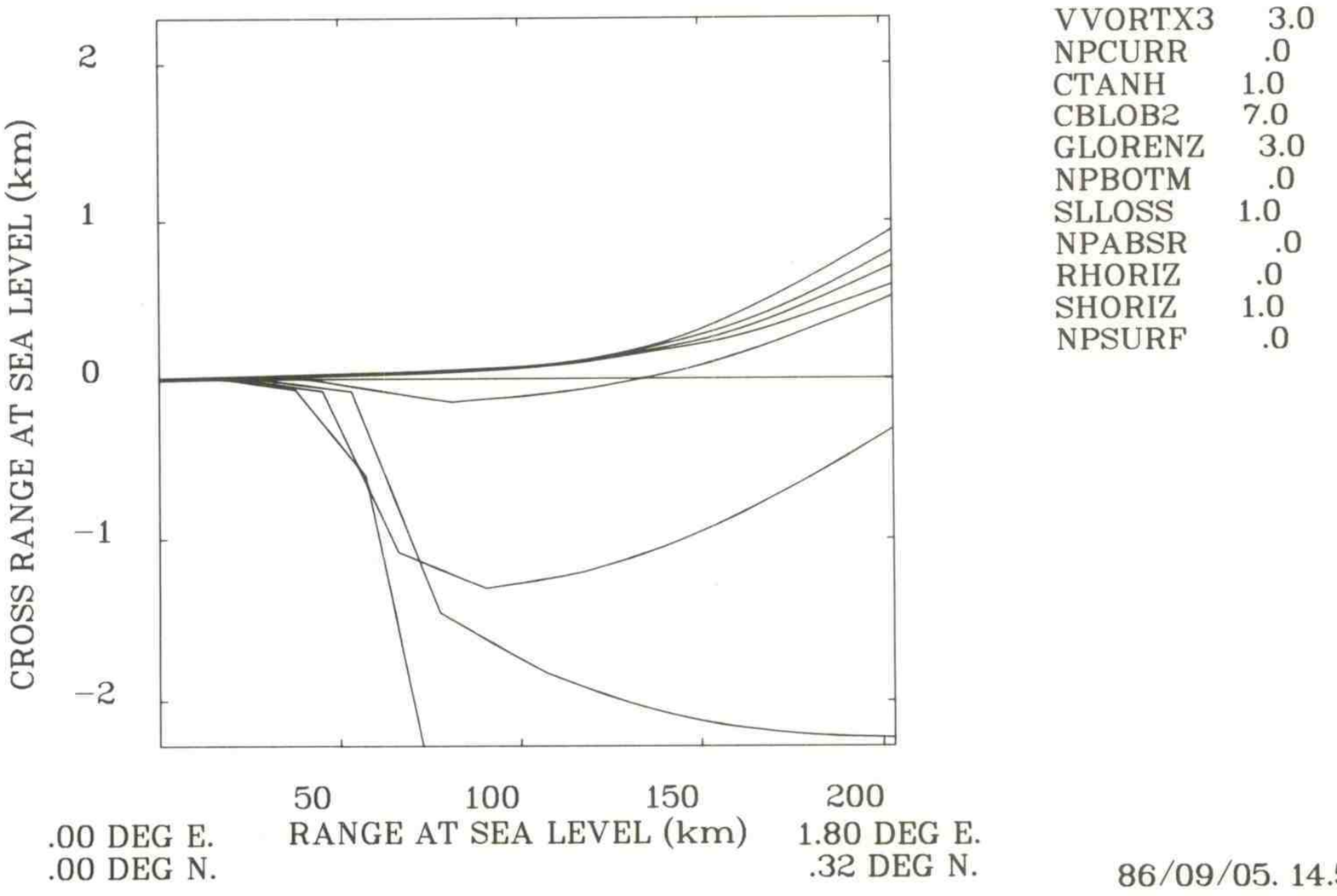


Figure 2.21. Projection of the rays of the sample case onto a horizontal plane whose axis is shown in Figure 2.18.

well as the line across the center of the horizontal projection (Figure 2.21).

The two rayplots show ray ducting in the sound channel, with rays launched at different elevation angles reaching different depths. The ray launched at 14° elevation angle reflects from the ocean surface on its third apogee. The horizontal projection shows the lateral deviations caused by the eddy model's sound-speed gradients and by reflections from the ridge on the ocean bottom. A more detailed picture of the effects of bottom reflections could be obtained by launching a dense fan of rays between 12° and 14°. This will be left as an exercise for the user.

Figure 2.22 shows an "apogee plot," a plotting option (see Fig. 2.16, W80) that gives only the loci of ray apogees and perigees. Such a diagram is useful for showing the ocean regions probed by a set of rays, without the clutter of intersecting raypaths.

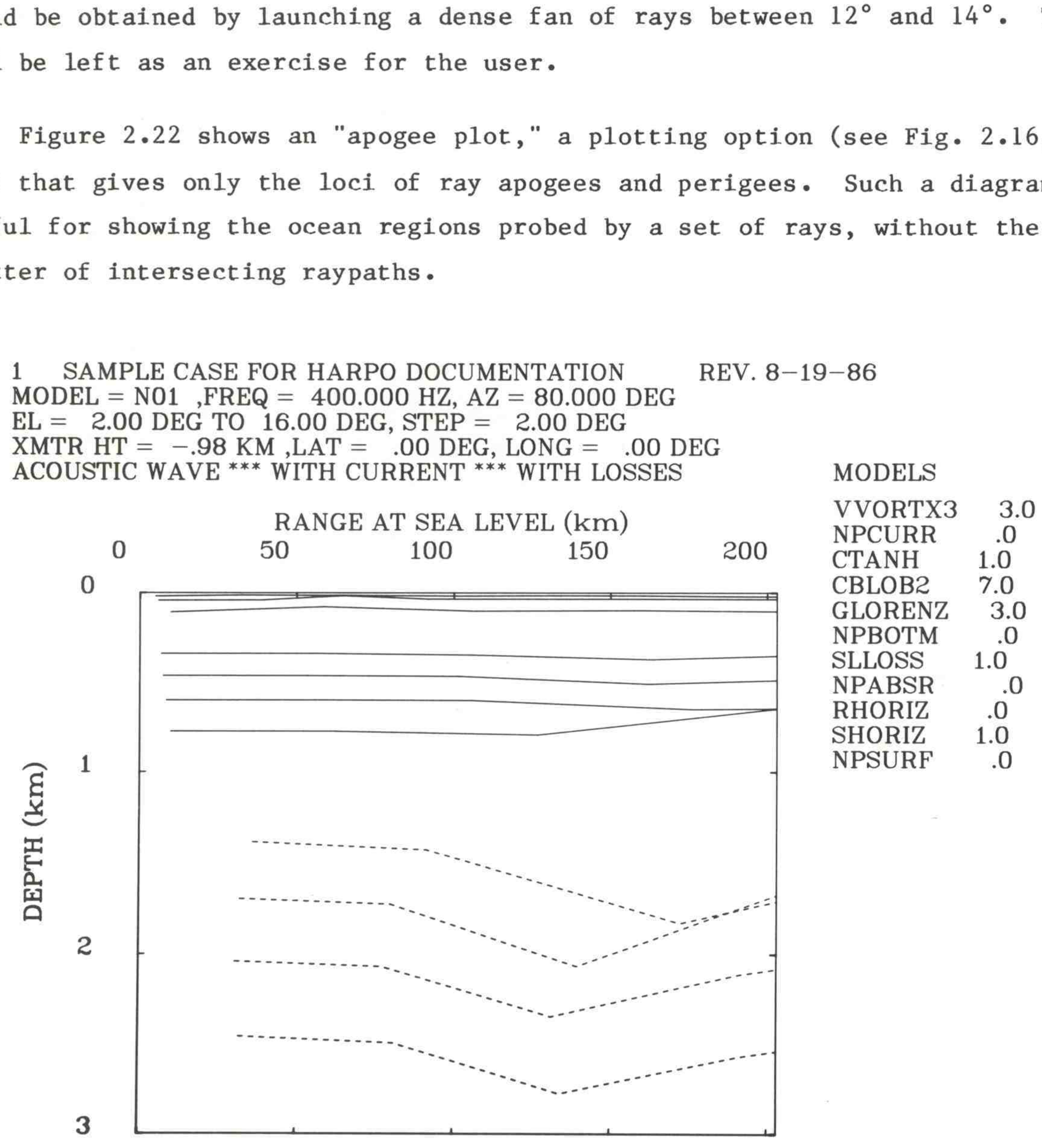


Figure 2.22. Ray-apogee plot for the sample case.

2.7 Machine-Readable Output for the Sample Case

HARPO produces two kinds of machine-readable output. One kind is called "raysets," which summarize in compressed form some useful ray parameters at each special event (as defined above in Section 2.5.2) along the raypath. The other kind of machine-readable output is called "binary raypath data," which permits a complete reconstruction of the raypaths by a supplementary processing program. When stored in machine-readable form (punched cards, magnetic tape, disk files), raysets (as a file named PUNCH) and binary raypath data (as a file named TAPE6) form the input to supplementary processing programs and extend the utility of ray-tracing calculations. Examples are supplementary programs to compute amplitude, to plot range versus elevation angle of transmission and range versus travel time, as well as programs that interpolate in elevation angle to estimate eigenrays that reach a specified range. These supplementary capabilities will be documented in other reports.

Figure 2.23 shows a portion of the printout of the raysets for the sample case. The complete rayset output for the sample case is given in Appendix A. Each ray begins with a "transmitter rayset," the lines beginning with "N01" in the example. Additional 80-column lines are produced whenever a ray reflects from the bottom or the ocean surface or crosses the receiver height and at the end of each ray trace. Because of the way hops are counted, two identical raysets are produced each time a ray executes a closest approach to the receiver surface.

The compressed rayset format is generally meant to be read by machines, not humans, so it can be rather difficult to inspect for information. However, since this is occasionally necessary (and to aid the user in designing supplementary programs to process raysets), Figures 2.24 and 2.25 provide the key for reading rayset printouts.

Notice that the last 3 items preceding the hop identifier in the receiver raysets contain all zeroes for the sample case. The first of these items is Doppler shift, which is zero because we did not use a time-varying ocean model. The transverse polarization is always zero for pure acoustic waves, but is nonzero for acoustic-gravity waves (Jones et al., 1982, Sec. 4.1).

```
N01-1     SAMPLE CASE FOR HARPO DOCUMENTATION          REV. 8-19-86
VVORTX3     3.0 NPCURR       .0   CTANH       1.0 CBLOB2      7.0
NO MODL      .0 NO MODL      .0   NO MODL      .0 NO MODL      .0
NO MODL      .0 NO MODL      .0   SLLOSS      1.0 NPABSR      .0
GLORENZ     3.0 NPBOTM       .0   RHORIZ       .0 NO MODL      .0
  VORTEX AT LONGITUDE 150 KM E, UMAX= 1.02 M/S, R= 50 KM
  NO CURRENT PERTURBATION
  EL NINO BACKGROUND SOUND-SPEED PROFILE
  2% INCREASE IN C-SQUARED AT 150 KM LON., 1 KM DEPTH, 50 KM WIDE
  RIDGE .5 KM HIGH, 2 KM WIDE AT 10 KM N LATITUDE; BASE= -3 KM
N013    -9808     0      0   -10000 25132741   8000000    200000        0    050T
   -10000   194734   -18   -20 -2022   1311291   1311291   288    0    0    0 1R
   -10000   520417   -16   -27  2010   3504472   3504471   769    0    0    0 2R
   -10000   723109   -15   -42 -1990   4868914   4868914  1069    0    0    0 3R
   -10000  1127595     6  -147  1418   7586966   7586966  1667    0    0    0 4R
   -10000  1406506    51  -324 -1281   9454219   9454219  2079    0    0    0 5R
   -10000  1941240   197  -503  3128  13036463  13036463  2870    0    0    0 6R
    -8511  2101960   240  -509 -2971  14116463  14116463  3108    0    0    0 7F
N013    -9808     0      0   -10000 25132741   8000000    400000        0    050T
   -10000   165446   -17   -19 -4012   1114351   1114351   245    0    0    0 1R
   -10000   468372   -15   -22  4001   3154502   3154502   693    0    0    0 2R
   -10000   637278   -14   -31 -3985   4291986   4291986   943    0    0    0 3R
   -10000   963209    -6   -74  3618   6484653   6484653  1425    0    0    0 4R
   -10000  1149018     7  -148 -3476   7731899   7731898  1700    0    0    0 5R
   -10000  1652653    91  -346  2658  11104455  11104455  2446    0    0    0 6R
   -10000  1864722   142  -439 -2819  12525535  12525535  2759    0    0    0 7R
   -15734  2105919   200  -455  1823  14145535  14145535  3116    0    0    0 8F
N013    -9808     0      0   -10000 25132741   8000000    600000        0    050T
   -10000   145209   -16   -17 -6008    978585    978585   215    0    0    0 1R
   -10000   458199   -12   -18  5996   3086200   3086200   679    0    0    0 2R
   -10000   605599   -13   -25 -5983   4079462   4079462   898    0    0    0 3R
   -10000   934379    -6   -55  5593   6291447   6291447  1385    0    0    0 4R
   -10000  1090494     1  -108 -5472   7340500   7340500  1616    0    0    0 5R
   -10000  1530604    52  -242  4301  10289292  10289292  2268    0    0    0 6R
   -10000  1709365    87  -352 -4402  11486110  11486110  2533    0    0    0 7R
   -10000  2087795   168  -388  5537  14026691  14026691  3093    0    0    0 8R
    -8578  2102600   171  -389  5404  14126691  14126691  3115    0    0    0 9F
```

Model Identification Header

Transmitter Raysets

Receiver Raysets

Figure 2.23. Portion of the rayset output for the first run set of the sample case.

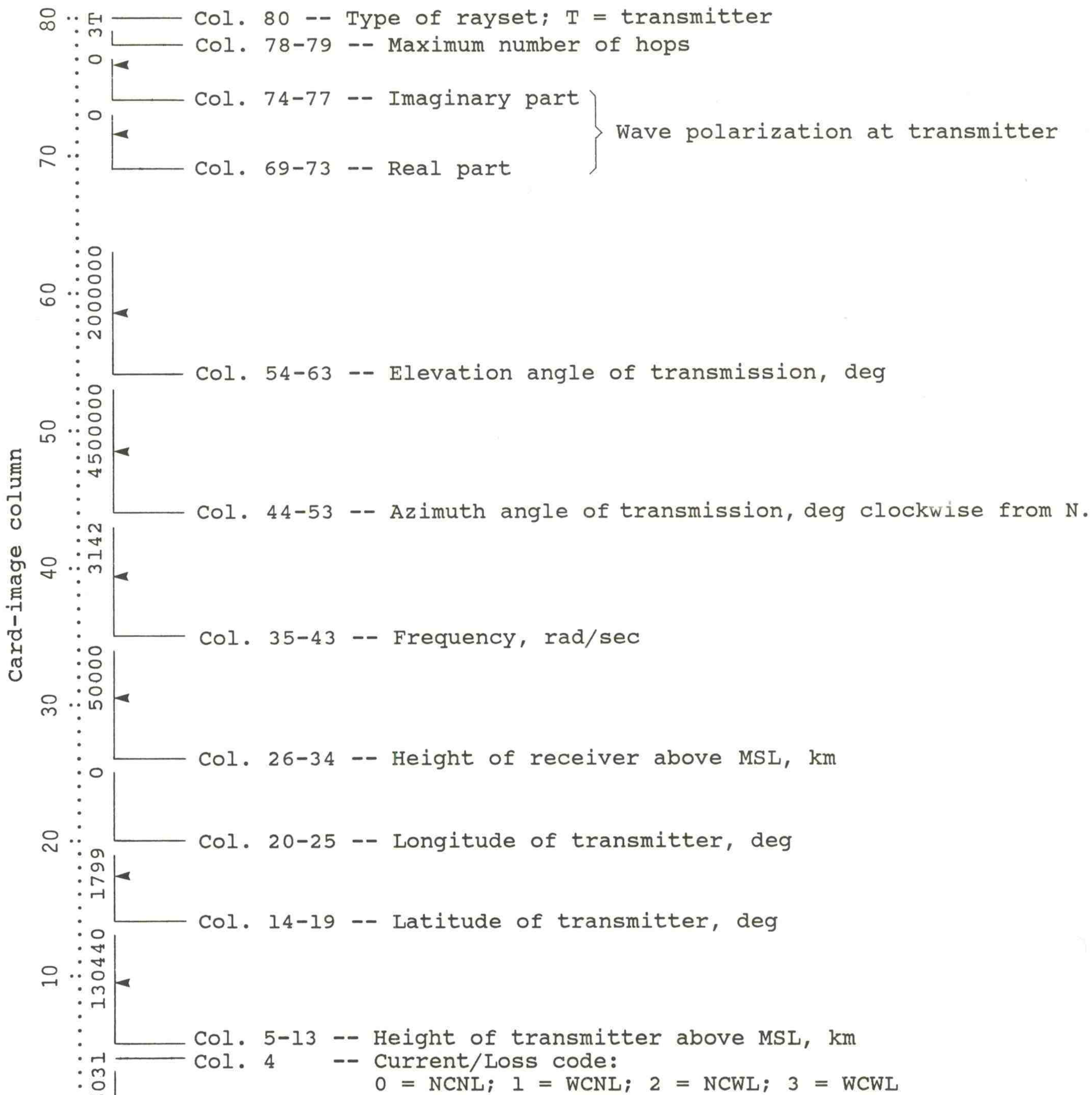


Figure 2.24. Definitions and format of a transmitter rayset.
▲ = implied decimal point.

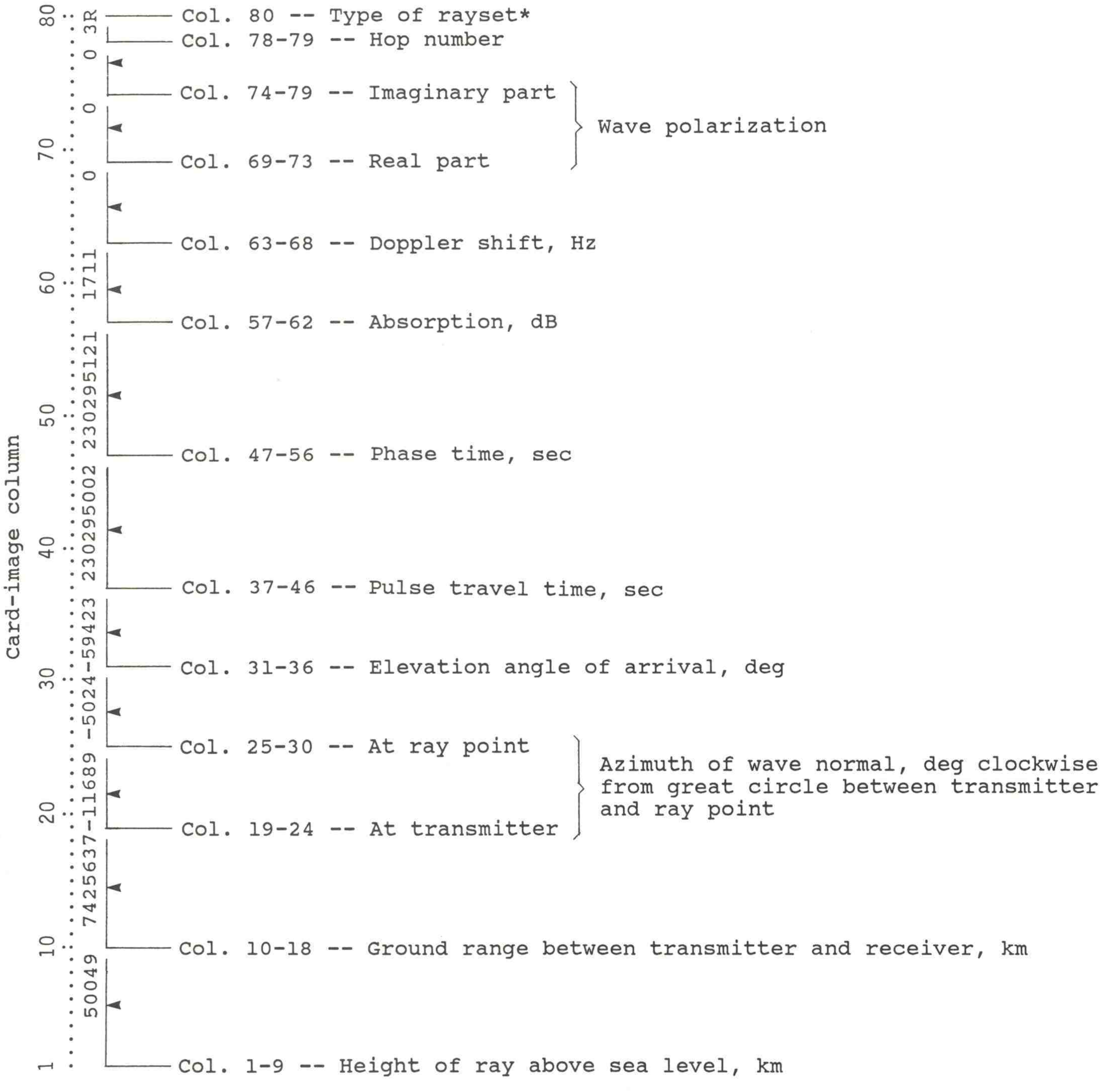


Figure 2.25. Definitions and format for a receiver rayset. * type of rayset: G = bottom reflection; M = closest approach to receiver height; R = at receiver height; S = maximum number of steps; E = extinction; F = exceeded maximum range; U = exceeded maximum height; D = went below minimum height; A = ocean-surface reflection; X = bottom absorption; ▲ = implied decimal point.

PART II: HOW TO USE THIS PROGRAM

3. How to Get This Program Running on Your Computer

This chapter explains how to get the FORTRAN program off the distribution tape and onto your computer, and how to get as far as running the sample case. It also deals with the machine-dependent aspects of running HARPO and suggests ways to deal with different computing environments.

3.1 How To Get a Copy of the Program

The FORTRAN program for the version of HARPO documented in this report and the Input Data File for the sample case are available on magnetic tape. For ordering information, contact the authors at the Wave Propagation Laboratory, Propagation Studies Program Area, 325 Broadway, Boulder, Colorado 80303.

The format of the distribution tape is 0.5 in × 700 ft, 9 track, 1600 bpi, ASCII character set, block size 1600 bytes, logical record length 80 bytes, no parity.

3.2 ANSI-FORTRAN 77 Compatibility

HARPO was designed to run on a Control Data Corp. (CDC) CYBER 700-800 series with a CDC FORTRAN 77 compiler. It should compile with any FORTRAN compiler that adheres to the ANSI FORTRAN 77 standard, including microcomputer FORTRAN compilers that claim such compatibility.

To ensure portability, we have made many changes in portions of the program that were written before the ANSI standard was established. However, where such changes would have been arduous, and where de facto standards that exist on many systems permit deviations from the ANSI standard, we have retained some non-ANSI code. Following are some exceptions to the ANSI standard:

(1) Some variable and subroutine names have seven characters (six is the ANSI standard).

(2) Some alphanumeric characters are stored eight characters per word in numeric (not character) variables and are output using A8 format.

(3) Some machine-dependent constants are entered in nonstandard format (see the following section).

(4) Sometimes a function is called as though it were a subroutine.

(5) Some real variables are EQUIVALENCEd to integer variables.

3.2.1 Machine- and System-Dependent Programming

We have tried to consolidate any machine- or operating-system-dependent code into two subroutines to make it easier to identify and adapt to new environments.

Before attempting to run HARPO, the user must modify SUBROUTINE DFCNST, which defines machine-dependent constants. The version supplied on the distribution tape is for the CYBER 700-800 series with NOS 2x. Instructions for modifying this routine for several popular machines are included as comments in SUBROUTINE DFCNST (Appendix D).

Another subroutine, DFSYS, contains some operating-system-dependent functions, such as clock and date functions and system-dependent I/O (input/output). Users should also check DFSYS and make changes appropriate to their own operating systems.

3.2.2 Word Length

Some problems may arise with machines that have word lengths shorter than the 60-bit word used in our CYBER 840. The numerical integration subroutine (RKAM1) uses double-precision arithmetic to accumulate numerically integrated quantities, but this is almost certainly not always necessary with a 60-bit word for ordinary precision requirements. We have not investigated what errors might occur if less precision were used. We recommend testing the precision on a different machine by running the sample case for smaller and smaller values of the single-step integration error, W(42), and verifying that the error value in the first column of the printout maintains at least the accuracy specified by W(42).

The level where that accuracy first breaks down is probably the precision limit imposed by the computer's word length.

3.2.3 Execution Speed

For many applications, HARPO runs fast enough on our CYBER 840 to allow virtually interactive (machine load permitting) ray tracing using a graphics terminal for editing program input and for viewing graphical output. Although HARPO may compile and run on smaller machines, its speed may be so slow that interactive ray tracing may no longer be practical. The run times shown on the printout (Appendix A) for the sample case (at the end of each ray) allow you to compare run times between your computer and ours. Tests on a CRAY XMP-48 indicate a speedup by a factor of about 7 over a CYBER 840.

3.2.4 Graphics

The graphics programs included with HARPO were designed to run on the CYBER series computer and use the DISSPLA graphics package (by ISSCO, Inc.) and the Information International Inc. Model FR-80 Microfilm Plotting Unit. However, HARPO will run regardless of the plotting facilities you have.

If you have DISSPLA, you can produce the graphic output by running the supplementary program TAPRD combined with DDSPLA subroutines, supplied as Files 6 and 7 of the distribution tape. This program reads a graphics file called TAPE5, which HARPO produces when plots are requested (see Fig. C1).

If you don't have DISSPLA, you can still run HARPO and get the printed and machine-readable outputs, but you won't get any graphics output. Just ignore the TAPE5 file. If you have other graphics devices, you can modify the DD-prefix plotting routines (in DDALT, supplied as File 8 of the distribution tape) to drive them. The functions of these routines and further details about the FR-80 plot package and the DISSPLA interface are given in Appendix C.

3.3 Unpacking the Program Tape and File Organization

The distribution tape contains the files listed in Section 3.3.1. Although HARPO continues to evolve, the source code on the distribution tape will always correspond exactly to the version described in this report. Updates and errata will be documented separately and included in a dated update tape file. Section 7.1 gives a list of the programs and subroutines on the distribution tape. Normally, you would transfer tape files 1-8 to punched cards or permanent disk files, depending on which medium you will use to run the program.

3.3.1 Files on the HARPO Distribution Tape

File #:

1. FORTRAN source code for the Sample Case, including its models, ready to compile, with common and data blocks included.

2. Input Data File for the Sample Case.

3. FORTRAN source code for the "Ray-tracing Core" programs, including plotting (graphics write) routines.

4. FORTRAN source code for four dispersion-relation routines.

5. FORTRAN source code for all ocean-model routines.

6. FORTRAN source code for graphics-file read routine TAPRD for reading the Graphics Output File (TAPE5).

7. FORTRAN source code for DDSPLA routines for users with DISSPLA.

8. FORTRAN source code for DDALT, a set of skeleton routines allowing users to insert plotting modules for their own plotting system.

3.3.2 Setting Up a Run Module

A run module is the subset of programs that you submit to your computer to run a particular application, along with input data and the job-control commands your computer needs to compile and/or run a program. Files 1 and 2 of the distribution tape constitute the run module (minus the job-control cards) for the sample case. It consists of core routines (Sec. 7.1.1) that must always be present to trace rays, and a set of selectable routines (Secs. 7.1.2 and 7.1.3) that describe the particular ocean model (those for the sample case in this example). Figure 3.1 shows the parts of a representative run module.

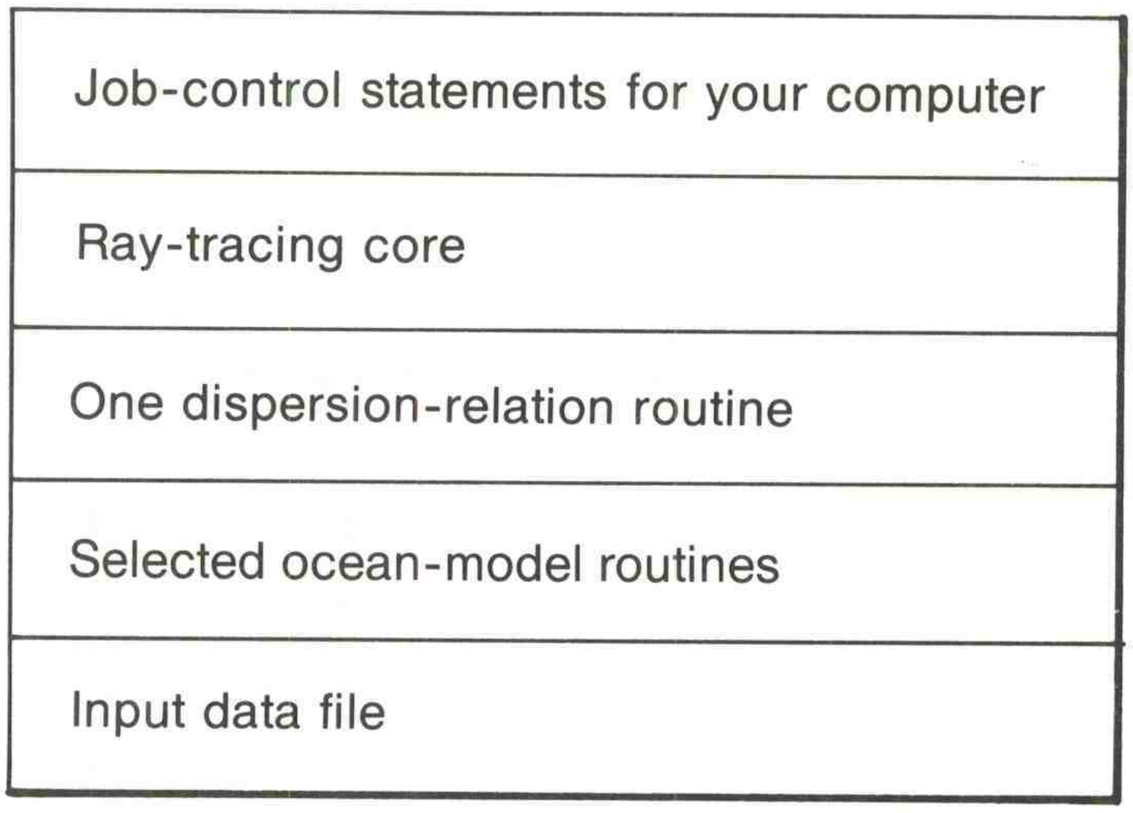


Figure 3.1. Configuration of a run module, assembled from parts consisting either of disk-file modules or punched-card decks.

The selectable part of the run module means that you select only the routines that describe the model you want to use. The run module must contain one and only one (with exceptions noted) of the following kinds of model routines:

a dispersion-relation routine
a background sound-speed model
a background current model (if you use AWCWL or AWCNL dispersion-relation models)
a background bottom model
a background ocean-surface model
a perturbation sound-speed model (NPSPEED does nothing)
a perturbation current model (NPCURR does nothing) (if you use AWCWL or AWCNL dispersion-relation models)
a perturbation bottom model (NPBOTM does nothing)
a perturbation ocean-surface model (NPSURF does nothing)
a receiver-surface model.
a loss model (if you use ANCWL or AWCWL dispersion-relation models)

In addition, you need any other models that are called by any of the above routines. Look at the bottom of the model input parameter forms (Appendix B) to see what other models a given model needs.

3.3.3 If You Are Using Cards To Input Data

If you have no permanent disk storage on your computer, you can load a previously compiled version of HARPO (if you don't change the program itself) into your computer from tape each time you want to run it, and have it read the Input Data File from punched cards. For each run, you have to edit the Input Data File (Deck) by punching new cards for the data you change from a previous run. The Input Data File is arranged in an 80-column format with one input parameter per card so that the deck is easy to edit if you have the contents of the cards interpreted (printed across the card tops).

3.3.4 If You Are Using Permanent Disk Files

It is far more convenient to store both the program and the Input Data File in permanent disk files. A run module (Figure 3.1) can be constructed by a batch or procedure program that selects the appropriate routines from the HARPO "library." Procedure (or batch) programs also simplify the manipulation of HARPO input and output, but because they depend on the operating system you use, you will generally have to write your own procedures.

4. How to Construct An Ocean Model

The easy way to set up an ocean model is to select from a general-purpose set of models we have designed (Table 4.1) and choose the model parameters that fit your needs. This requires no programming whatsoever and we encourage that choice whenever possible. Alternatively, you can design your own ocean model by writing a FORTRAN subroutine that defines the ocean property and its spatial derivatives in a form that is compatible with the rest of the program. This chapter describes both ways.

4.1 Choosing From the Available Ocean Models

We have designed some generic ocean models that can be adapted to represent common ocean structures simply by selecting the appropriate model and its parameters in the Input Data File. They are closed-form expressions for an ocean property as a function of geographic latitude, longitude, and depth; some accept input parameters in tabular form. There are models for current and sound-speed fields, absorption, and top and bottom surfaces. Though not strictly considered part of the ocean model, three models for the receiver surface are also provided. All of these models are described in Appendix B.

Most of the models come in two kinds, "background" and "perturbation." Perturbation models generally superimpose more structure on a background model. Table 4.1 lists the ocean models that come with HARPO. To run HARPO, you always have to specify one background model and one perturbation model for each required ocean property, even if the perturbation is a do-nothing version.

To put together an ocean model from the subroutines we have supplied, first copy the FORM TO SPECIFY AN OCEAN MODEL from Appendix B and fill in the name of a model you want to use for each ocean parameter, selecting from the choices listed in Table 4.1. Full mathematical descriptions of each model can be found in Appendix B on the order form listed under the model name, and FORTRAN listings for the model subroutines can be found in Appendix D. For now, leave off the numbers of the Data Set ID column until you have selected the models' parameters.

Table 4.1--Available ocean models

Model type number	Start of W array parameter block	Model check number	Subroutine name	Description
1.	100			Background current models:
		1.	WLINEAR	Constant upward and northward current, linear eastward current profile
		9.	VVORTX3	Vertical, cylindrical current vortex
		8.	WGAUSS2	Localized (Gaussian) eastward current field
2.	125			Current-perturbation model:
		0.	NPCURR	Do-nothing version
				Sound-speed models:
3.	150	2.	CSTANH	Linear C^2 segments joined by hyperbolic functions
		3.	CSSPOKE	C^2 function of angle from horizontal
		4.	CSSPOK2	C^2 function of angle from horizontal
		7.	CTANH	Linear C segments, smoothly joined
		5.	CSMUNK1	Canonical sound channel; interpolates parameters
		6.	CSMUNK2	Canonical sound channel; interpolates C
		8.	CTABLE	Tabular sound-speed profile in cubic segments
4.	175			Sound-speed perturbation models:
		0.	NPSPEED	Do-nothing version
		2.	CBLOB2	Localized (Gaussian) sound-speed perturbations
		3.	CBLOB3	
8.	275			Receiver-surface models:
		1.	RHORIZ	A sphere concentric with the earth
		2.	RTERR	A fixed height above the bottom
		3.	RVERT	A vertical surface at a specified fixed range from a specified geographic point
9.	300			Bottom models:
		1.	GHORIZ	A sphere concentric with the earth
		3.	GTANH	A profile of linear segments joined by hyperbolic functions
		4.	GLORENZ	An east-west Lorentzian-shaped ridge

Table 4.1--Available ocean models (continued)

Model type number	Start of W array parameter block	Model check number	Subroutine name	Description
10.	325	0.	NPBOTM	Bottom perturbation: Do-nothing version
11.	350	1.	SHORIZ	Ocean-surface model: A sphere concentric with the earth
12.	375	0.	NPSURF	Ocean-surface perturbation model: Do-nothing version
17.	500	1.	SLLOSS	Ocean absorption model: Skretting-Leroy absorption formula
18.	525	0.	NPABSR	Ocean absorption perturbation model: Do-nothing version

Next, select and copy the blank Input Parameter Forms from Appendix B for the models you have selected and fill in the values of the variable parameters that you want. Next transfer the input parameters to a new Input Data File, either constructing one from scratch, according to the format shown in Figure 2.19, or modifying an old one. (If you use an old one, make sure that unused parameters are removed.) Remember to assign an input data-set identification number, which uniquely identifies that set of input parameters for each model, and to assign an Ocean Identification (ID) for the entire set of models. Put these ID numbers on the FORM TO SPECIFY AN OCEAN MODEL and save all these forms as a record of the models you have defined.

Here are a few guidelines for selecting models. If you want a model with no current, use no current model and select a dispersion-relation routine with no current (ANCNL or ANCWL). If you want no perturbation model for current, sound speed, ocean surface, bottom, or absorption, use the corresponding do-nothing perturbation models NPCURR, NPSPEED, NPSURF, NPBOTM, or NPABSR. If you are storing HARPO on a permanent disk file, you have to select only the subroutines that define your ocean model (and the correct dispersion-relation routine) and assemble them into a separate "run module" (Sec. 3.3.2). If you are storing the programs on punched cards, you should select and submit only the decks for the

model subroutines you want to use. It is convenient to think of HARPO as consisting of a core of ray-tracing routines that are always used, and a set of selectable model-related routines from which you select the ones appropriate to the models you want. The specific routines that fall into each category are listed in Chapter 7.

4.1.1 Model Check Numbers

To guard against accidentally selecting the wrong model subroutine for a run module, each model is assigned a permanent Model Check Number, which is entered on each model input data form (Appendix B). If a model subroutine is selected whose Check Number does not match that specified in the Input Data File, then the program will stop and give an error message.

Another number, called the Input Data Format Code, is not now being used or checked, but may be used in the future.

4.1.2 Tabular Input to Models

Some models, like CTANH and CTABLE, can accept so many input parameters that it is inconvenient to specify each one as a separate line in the Input Data File, so a general provision has been made for entering data in tabular form. The use of CTANH in the sample case is an example. Tabular data are entered into the Input Data File in a special format illustrated by Figure 2.19 and described fully in Chapter 5.

4.2 Designing Your Own Models

HARPO will accept any ocean model specification that provides the desired ocean property as a continuous function of the earth-centered spherical-polar coordinates r,θ,ϕ and time t, as well as its first spatial and temporal derivatives. (Bottom models require specifying continuous second derivatives as well.) There are three important considerations in writing model subroutines: (a) All first spatial derivatives must be not only continuous but also analytically consistent with the formulas for the atmospheric property itself; any errors or approximations in those calculations will result in larger-than-desired integration errors, as displayed in the first column of the ray-tracing

printout. (b) The input data for the models must come from the part of the W array assigned to that type of model (Table 4.1) and from the tabular-input common blocks assigned to those models (Table 4.2). (c) The output from the ocean model must go to the appropriate data-output common block (Table 4.3). Because the model routines are called many times, efficient programming here pays off in execution efficiency.

If you want a new model of current or sound speed that depends only on depth, you would normally design only a new background model. If you want a three-dimensional model, you could use one of the background models we have supplied and design only a new perturbation model. Conceivably, you could

Table 4.2--Allocation of common blocks for tabular input to the various ocean models*

Common block name	Ocean model
/B1/	Current velocity
/B2/	Perturbation to the current velocity
/B3/	Sound speed
/B4/	Perturbations to the sound speed
/B8/	Receiver surface
/B9/	Ocean bottom
/B10/	Ocean-bottom perturbation
/B11/	Ocean surface
/B12/	Ocean-surface perturbation
/B17/	Absorption
/B18/	Absorption perturbation

* In the first 31 elements of each of these common blocks, each ocean model indicates the structure of the rest of the common block. Subroutine READW1 stores input data that it reads starting in element 32 of the common block.

Table 4.3--Allocation of common blocks for output from the various ocean models

Common block name	Location of description (table number)	Ocean model
/UU/	7.33	Current velocity
/CC/	7.34	Sound speed
/RR/	7.37	Receiver surface
/GG/	7.38	Bottom
/LL/	7.35	Absorption (losses)
/SS/	7.36	Ocean surface

design both a new background and a new perturbation model, but the safe way to proceed is to do one at a time.

Those designing a new model should pattern their subroutine after a similar one that comes with HARPO. We will use the CTANH model (Fig. 4.1) as an example and discuss its structure in detail to illustrate how to write a model subroutine. In the following paragraphs, general statements will be followed in square brackets by the specific examples from CTANH.

There are no restrictions on naming models, but it is useful to assign a name that suggests the model's function. [CTANH is a sound-speed model that used TANH (hyperbolic tangent) functions to smooth sound-speed profiles that have linear segments.] Each model subroutine has three entry points whose standard names are given in Table 4.4. These names must be used when designing new subroutines. The first entry point [ENTRY SETSPD] is to transfer values from local variables to labeled common, since Fortran 77 forbids references in DATA statements to labeled-common variables. The second entry point [ENTRY IPSPEED], whose name begins with an "I," is for initialization after new input data have been read in, and that entry point is called each time new data are read before the second entry point [ENTRY SPEED] is called. The third entry point [ENTRY SPEED] enters the routine to compute the ocean parameter [CS] and its time and space derivatives [PCST, PCSR, PCSTH, PCSPH] according to the formulas given on the model order form.

```
      SUBROUTINE CTANH                                                      CTANH  2
C     SPEED PROFILE REPRESENTED BY A SEQUENCE OF LINEAR SEGMENTS            CTANH  3
C     SMOOTHLY JOINED BY HYPERBOLIC FUNCTIONS. PARAMETERS ARE INPUT         CTANH  4
C     AS TABULAR DATA WITH SLOPES COMPUTED FROM SPEED DATA.                 CTANH  5
C     REFERENCE SPEED C0 IS READ FROM TABULAR DATA.                         CTANH  6
C                                                                           CTANH  7
      REAL CO(20), TM(19), Z(19), DL(19)                                    CTANH  8
C                                                                           CTANH  9
C     COMMON DECK "RKAM" INSERTED HERE                                      RKAMCOM2
      REAL KR,KTH,KPH                                                       RKAMCOM4
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)           RKAMCOM5
C     COMMON DECK "CC" INSERTED HERE                                        CCC    2
      REAL MODC                                                             CCC    4
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                            CCC    5
C     COMMON DECK "WW" INSERTED HERE                                        CWW    2
      PARAMETER (NWARSZ=1000)                                               CWW1   3
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                       CWW1   4
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                    CWW2   2
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),        CWW2   3
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),          CWW2   4
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),              CWW2   5
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),             CWW2   6
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                     CWW2   7
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))    CWW2   8
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                          CWW2   9
     6 (HMIN,W(27)),(RGMAX,W(28)),                                          CWW2  10
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                         CWW2  11
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),           CWW2  12
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))            CWW2  13
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),            CWW2  14
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                  CWW2  15
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                       CWW2  16
      REAL MMODEL,MFORM,MID                                                 CWW3   2
C                                                                           CWW3   3
C     WIND        100-124                                                   CWW3   4
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)               CWW3   5
C                                                                           CWW3   6
C     DELTA WIND        125-149                                           CWW3   7
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)            CWW3   8
C                                                                           CWW3   9
C     SOUND SPEED 150-174                                                   CWW3  10
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)               CWW3  11
      EQUIVALENCE (W(153),REFC)                                             CWW3  12
C                                                                           CWW3  13
C     DELTA SOUND SPEED 175-199                                             CWW3  14
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)            CWW3  15
C                                                                           CWW3  16
C     TEMPERATURE    200-224                                                CWW3  17
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)               CWW3  18
C                                                                           CWW3  19
C     DELTA TEMPERATURE    225-249                                          CWW3  20
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)            CWW3  21
C                                                                           CWW3  22
C     MOLECULAR   250-274                                                   CWW3  23
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)               CWW3  24
C                                                                           CWW3  25
C     RECEIVER HEIGHT   275-299                                             CWW3  26
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)               CWW3  27
C                                                                           CWW3  28
C     TOPOGRAPHY    300-324                                                 CWW3  29
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)               CWW3  30
C                                                                           CWW3  31
C     DELTA TOPOGRAPHY   325-349                                            CWW3  32
```

Figure 4.1. Listing for model sound-speed subroutine CTANH.

```
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)          CWW3  33
                                                                          CWW3  34
C
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                           CWW3  35
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)             CWW3  36
                                                                          CWW3  37
C
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                           CWW3  38
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)          CWW3  39
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                CWW3  40
C                                                                         CWW3  41
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                         CWW3  42
C     ABSORPTION         500-524                                          CWW3  43
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)             CWW3  44
                                                                          CWW3  45
C
C     DELTA ABSORPTION         525-549                                    CWW3  46
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)          CWW3  47
C                                                                         CWW3  48
C     PRESSURE          550-574                                           CWW3  49
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)             CWW3  50
                                                                          CWW3  51
C
C     DELTA PRESSURE          575-599                                     CWW3  52
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)          CWW3  53
                                                                          CWW3  54
C
C     COMMON DECK "B3" INSERTED HERE                                      CB3    2
      INTEGER CMX,CNTBL,CITBL,CFRMTBL,IDSC(10)                            CB3    4
      COMMON/B3/CMX,CNTBL(10),CITBL(10),CFRMTBL(10),CGP(512)              CB3    5
      EQUIVALENCE (CGP,IDSC),(ANC,CGP(11))                                CB3    6
                                                                          CTANH 14
C
      EQUIVALENCE (Z0,CGP(12)),(TM,CGP(33))                               CTANH 15
      EQUIVALENCE (Z,CGP(13)),(CO,CGP(32)),(DL,CGP(53))                   CTANH 16
                                                                          CTANH 17
C
      INTEGER CPX    ,CQTBL(10),CLTBL(10),CIRMTBL(10)                     CTANH 18
      DATA RECOGC,N/7.0,0/                                                CTANH 19
                                                                          CTANH 20
C
      DATA AQC/0.0/                                                       CTANH 21
      DATA CPX/2/                                                         CTANH 22
      DATA CQTBL/1,11,72,7*0/                                             CTANH 23
      DATA CLTBL/1,20,8*0/                                                CTANH 24
      DATA CIRMTBL/1,2,8*0/                                               CTANH 25
                                                                          CTANH 26
C
      COSH (X) = (EXP (X) + 1. / (EXP (X))) / 2.                          CTANH 27
                                                                          CTANH 28
C
C                                                                         CTANH 29
      ENTRY SETSPD                                                        CTANH 30
                                                                          CTANH 31
C
      ANC=AQC                                                             CTANH 32
      CMX=CPX                                                             CTANH 33
      CALL IMOVE(CNTBL,CQTBL,10)                                          CTANH 34
      CALL IMOVE(CITBL,CLTBL,10)                                          CTANH 35
      CALL IMOVE(CFRMTBL,CIRMTBL,10)                                      CTANH 36
      CALL SETPSP                                                         CTANH 37
                                                                          CTANH 38
C
      RETURN                                                              CTANH 39
                                                                          CTANH 40
C
      ENTRY ISPEED                                                        CTANH 41
                                                                          CTANH 42
C
C     IF HAD PREVIOUS CALL BUT NOTHING THIS TIME, EXIT NOW                CTANH 43
C     RETAINING PREVIOUS TABULAR DATA COUNT                               CTANH 44
C                                                                         CTANH 45
      CALL IPSPEED                                                        CTANH 46
                                                                          CTANH 47
C
      IF(N.GT.0 .AND. ANC.EQ.0.0) RETURN                                  CTANH 48
                                                                          CTANH 49
C
      IF(RECOGC .NE. CMODEL)                                              CTANH 50
```

Figure 4.1. Listing for model sound-speed subroutine CTANH (continued).

```
     1    CALL RERROR('SPEED   ','WRNG MODEL',RECOGC)                    CTANH 51
C                                                                         CTANH 52
      MODC(1)=7HCTANH                                                     CTANH 53
      MODC(2)=CID                                                         CTANH 54
C                                                                         CTANH 55
      N=(ANC+1)/3 - 2                                                     CTANH 56
C                                                                         CTANH 57
      IF(N.LE.0)                                                          CTANH 58
     1    CALL RERROR('CTANH','BAD N VALUE',FLOAT(N))                    CTANH 59
C                                                                         CTANH 60
      ANC=0.0                                                             CTANH 61
C                                                                         CTANH 62
C     CONVERT 'CGP' ARRAY INPUT(OVERLAYS 'C' ARRAY) TO 'C' ARRAY          CTANH 63
C                                                                         CTANH 64
      C0=CO(1)                                                            CTANH 65
      TIM1=C0                                                             CTANH 66
      ZIM1=Z0                                                             CTANH 67
      NP1=N+1                                                             CTANH 68
      DO 10 I=1,NP1                                                       CTANH 69
         TI=TM(I)                                                         CTANH 70
         ZI=Z(I)                                                          CTANH 71
         CO(I)=(TI-TIM1)/(ZI-ZIM1)                                        CTANH 72
         TIM1=TI                                                          CTANH 73
10       ZIM1=ZI                                                          CTANH 74
C                                                                         CTANH 75
      RETURN                                                              CTANH 76
C                                                                         CTANH 77
      ENTRY SPEED                                                         CTANH 78
      H = R - EARTHR                                                      CTANH 79
      SUM = 0.                                                            CTANH 80
C                                                                         CTANH 81
C     LOOP TO SUM OVER ALL COEFFICIENTS                                   CTANH 82
C     USE SPECIAL FUNCTION 'ALCOSH' WHICH ALLOWS FOR LARGE ARGUMENTS.     CTANH 83
      DO 1 I = 1, N                                                       CTANH 84
   1  SUM = SUM + DL(I) * (CO(I + 1) - CO(I)) / 2. *(ALCOSH((H - Z        CTANH 85
     1(I)) / DL(I)) - ALCOSH((Z(I)-Z0) / DL(I)))                          CTANH 86
C                                                                         CTANH 87
      C = C0 + SUM + (CO(1) + CO(N + 1)) * (H - Z0) * 0.5                 CTANH 88
C                                                                         CTANH 89
      SUM = 0.                                                            CTANH 90
      DO 2 I = 1, N                                                       CTANH 91
   2  SUM = SUM + (CO(I + 1) - CO(I)) / 2. * (1. + TANH ((H - Z(I)) /DL CTANH 92
     1 (I)))                                                              CTANH 93
C                                                                         CTANH 94
      CS=C*C                                                              CTANH 95
C                                                                         CTANH 96
      PCST=0.0                                                            CTANH 97
      PCSR = 2.0*C*(CO(1) + SUM)                                          CTANH 98
      PCSTH=0.0                                                           CTANH 99
      PCSPH=0.0                                                           CTANH100
C                                                                         CTANH101
      CALL PSPEED                                                         CTANH102
      RETURN                                                              CTANH103
      END                                                                 CTANH104
```

Figure 4.1. Listing for model sound-speed subroutine CTANH (continued).

The input to each subroutine (geographic coordinates r, θ, ϕ) is through blank common, and output [sound speed squared and its derivatives] is through the labeled common blocks [/CC/], named for each kind of model and listed in Table 4.3. If you need more input parameters than will fit in the assigned

Table 4.4--Assignment of entry point names and input parameter blocks in the W array for the ocean models

Ocean model	Entry point names	Input parameter block in the W array
Current	SETWND, WINDR, IWINDR	100-124
Current perturbation	SETPWN, PWINDR, IPWINDR	125-149
Sound-speed	SETSPD, SPEED, ISPEED	150-174
Sound speed perturbation	SETPSP, PSPEED, IPSPEED	175-199
Receiver surface	SETRCV, RECVR, IRECVR	275-299
Bottom	SETTOP, TOPOG, ITOPOG	300-324
Bottom perturbation	SETPTP, PTOPOG, IPTOPOG	325-349
Ocean surface	SETSUR, SURFACE, ISURFAC	300-374
Ocean surface perturbation	SETPSR, PSURFC, IPSURFC	375-399
Absorption	SETABS, ABSRP, IABSRP	500-524
Absorption perturbation	SETPAB, PABSRP, IPABSRP	525-549

block of the Input Data Table [150-174], then you should use a tabular input format, which uses the labeled common blocks [/B3/] listed in Table 4.2. Tabular data are read into this common block from the Input Data File according to the format described in Chapter 5 and illustrated near the end of Figure 2.19 for the sample case.

4.2.1 How To Write an Ocean Model Subroutine So It Can Receive Tabular Data Read Into Common Blocks by Subroutine READW1

If you want to write an ocean model subroutine that uses tabular input data, then you have to observe some special precautions. In what follows, general statements are exemplified in square brackets for the case of the sound-speed subroutine. You can use this example as a guide in developing new model subroutines that use tabular data.

Tabular [sound-speed] data are read into a model-related common block [/B3/] by subroutine READW1. Table 4.2 gives the names of the common blocks associated with the different model types. The format of the tabular input data can be selected by the user and is determined by a format number [3] in columns 1-3 of the Input Data File (see Fig. 2.19). [In the case of CTANH, a three-column format makes sense because there are three input parameters for each profile point]. READW1 interprets this format number according to the formats listed in Table 5.3. The model subroutine [CTANH] must tell READW1 how it wants the tabular input data stored in the common block [/B3/] in an array [CGP]. It does so by setting (in DATA statements) the values in variable CMX and in arrays CNTBL, CITBL and CFRMTBL for sound speed models (or corresponding names for other model types, in which the first letter is different). The model variables [ZO,TM,C,DL] are EQUIVALENCEd to elements of a GP (for general-purpose) array [called CGP for sound-speed models]. The first element [ANC] of each numeric data block is set by SUBROUTINE READW1 to the number of data values actually read. Table 4.5 defines the structure of common block /B3/, which transmits these variables between READW1 and CTANH, and it explains how to set the data-block parameters.

4.2.2 Designing Your Own Ocean-Bottom and Ocean-Surface Models

An ocean-bottom model specifies a function $g(r,\theta,\phi)$ such that $g=0$ for a point (r,θ,ϕ) on the bottom, $g>0$ for a point (r,θ,ϕ) above the bottom, and $g<0$ for a point (r,θ,ϕ) below the bottom. To be an allowed model, g must be continuous through second derivatives. A subroutine for a background or perturbation bottom model must calculate g, its three first derivatives, and its six second derivatives for any values of r,θ,ϕ. All of our present bottom models define g to be the height above the bottom, but more general definitions are allowed to handle cliffs, overhangs, and caves. To design a simple model with a few parameters, follow the example of SUBROUTINE GLORENZ. To design a more elaborate model that needs tabular input data, follow the example of SUBROUTINE GTANH. Source-Code listings for these models are in Appendix D.

Models for the ocean surface follow the same rules as those for the bottom, with both background and perturbation models allowed. Ordinarily, the ocean surface is represented by a sphere at MSL (or some other radius), and we provide SUBROUTINE SHORIZ for this purpose. A do-nothing surface-perturbation model NPSURF is provided. Other ocean-surface models can be patterned after these examples.

Table 4.5--Definitions of the parameters in common block /B3/*

Position in common	Variable name	Definition	Value
1	CMX	Maximum number of data blocks in /B3/	2
2-11	CNTBL	An array that contains the beginning locations (in CGP array) of data blocks within the common block	
2	CNTBL(1)	Beginning location (in CGP array) of data block 1	1
3	CNTBL(2)	Beginning location (in CGP array) of data block 2	11
4	CNTBL(3)	Beginning location (in CGP array) of data block 3**	12 + number of data columns times the maximum number of rows of data
5 -11	CNTBL(4)-CNTBL(10)	Not used	0
12-21	CITBL	An array that contains the maximum number of rows of data in the data blocks (if there is more than one array in the data block, then this is the dimension of the arrays)	
12	CITBL(1)	Maximum number of rows in data block 1	1
13	CITBL(2)	Maximum number of rows in data block 2	= array dimension for data columns
14-21	CITBL(3)-CITBL(10)	Not used	0
22-31	CFRMTBL	An array that contains the input format type numbers for the data blocks within the common block	
22	CFRMTBL(1)	Format type*** for data block 1	1
23	CFRMTBL(2)	Format type*** for data block 2	2
24-31	CFRMTBL(3)-CFRMTBL(10)	Not used	0
32-	CGP	An array (of length at least CNTBL(3)-1) containing CMX number of data blocks for tabular input data for ocean models	
42	CGP(11)	The number of data values actually read into data block 2	roughly equal the number of data columns times the number of rows of data read

* The values of the first 31 elements in /B3/ define the block structure for the array CGP, and they are set in the ocean models and used in the data read-in routines READW and READW1. The common blocks /B1/, /B2/, /B4/, /B5/, /B6/, and /B7/ have the same structure as /B3/, but have different names for the variables.

** Only two data blocks are now available to use; however, the beginning location of the first data block not used must be specified to indicate the length of the last data block used.

*** Format type 1 implies format number A (see Table 5.3).
Format type 2 implies format numbers 1, 2, or 3 (see Table 5.3).

5. How to Specify the Input Data and Set Up an Input Data File

To give HARPO an ocean model and to tell it what rays to trace, you have to construct an Input Data File, like the one shown in Figure 5.1 (same as Figure 2.19, reproduced here for your convenience) for the sample case. An Input Data File may contain one or more "run sets," each of which can specify a different ocean model (but the same ocean-model subroutines), different plotting modes, or different initial ray conditions, but which will all be executed as a single computer run. The sample case contains two run sets. After the first run set, only the parameters whose values differ from those specified in the preceding run set need be specified.

After setting up an Input Data File, you run HARPO by combining the Input Data File with other ray-tracing modules to form a "run module," as explained in Section 3.3.2. All the necessary modules to run the sample case are contained in Files 1 and 2 of the distribution tape (Sec. 3.3.1).

5.1 Editing the Input Data File

Because HARPO contains no built-in way to construct or edit an Input Data File, you have to use a text editor of your own to do so. We have designed a specialized editor, called WMOD, for this purpose. It not only permits editing the Input Data File, but it also sets up a "run module" that includes appropriate ray-tracing and model routines and job-submission procedures for our computer. This program, which could run on a local microcomputer, will be documented in another report.

The Input Data File can take the form of either a deck of punched cards or a disk file to be read by HARPO. Some suggestions for those using punched cards are given in Section 3.3.3. Henceforth, we will assume that the Input Data File will be created as a disk file. There are no formal differences between the two methods, however.

Rather than start from scratch, we recommend that you modify an existing Input Data File. After you have run the sample case and have verified that its

col. n	col. W(n)	col. UNITS	col. DESCRIPTION
GEORGES	RB3	X6437	
N01-1			SAMPLE CASE FOR HARPO DOCUMENTATION REV. 8-19-86
1	6370.		EARTH RADIUS TO MSL, KM (6370.)
3	2.		TTRANSMITTER HEIGHT ABOVE MSL (T=ABOVE BOTTOM), KM
4	0.	AN KM	N. TRANSMITTER LATITUDE, KM
5	0.	AN KM	E. TRANSMITTER LONGITUDE, KM
7	400.	FQ HZ	INITIAL FREQUENCY, HZ
11	80.	AN DG	INITIAL AZIMUTH ANGLE, DEG
15	2.	AN DG	INITIAL ELEVATION ANGLE, DEG
16	16.	AN DG	FINAL ELEVATION ANGLE, DEG
17	2.	AN DG	STEP IN ELEVATION ANGLE, DEG
19	0.		STOP RAYS THAT STRIKE BOTTOM (1=YES; 0=NO)
20	-1.		RECEIVER HEIGHT ABOVE MSL, KM
22	50.		MAXIMUM NUMBER OF HOPS (1.)
23	1000.		MAXIMUM NUMBER OF STEPS PER HOP (1000.)
26	5.		MAXIMUM RAY HEIGHT ABOVE MSL, KM
27	-5.		MINIMUM RAY HEIGHT ABOVE MSL, KM
28	210.		MAXIMUM RANGE AT MSL, KM
29	0.		DO: EIGRAY/RNG-TIM/RNG-ELV/NEW-PROJ/RAYTRC/CONT/PROF
33	20.		MAXIMUM ABSORPTION, DB (999.999)
42	1.0E-06		MAXIMUM SINGLE-STEP INTEGRATION ERROR (1.0E-4)
44	.1		INITIAL INTEGRATION STEP SIZE, KM (1.0)
57	2.		PHASE PATH (0=NO; 1=INTEGRATE; 2=INTEGRATE/PRINT)
58	2.		ABSORPTION (0=NO; 1=INTEGRATE; 2=INTEGRATE/PRINT)
60	2.		PATH LENGTH (0=NO; 1=INTEGRATE; 2=INTEGRATE/PRINT)
71	50.		NUMBER OF INTEGRATION STEPS PER PRINT [1.E31]
72	1.		OUTPUT RAYSETS (1=YES; 0=NO)
73	0.		DIAGNOSTIC PRINTOUT (1=YES; 0=NO)
74	0.		PRINT EVERY W(71) RAY STEPS (0=YES; 1=N0)
75	.15		FULANN PLOT LETTERING; HEIGHT= [0.15 IN]
76	0.		BINARY RAY OUTPUT (1=YES; 0=NO)
77	57.		LINES PER PAGE OF PRINTOUT (57.)
81	4.		RAYPLOT PROJECTION PLANE (4 = VERT. RECTANGULAR)
82	40.		PLOT-ORDINATE EXPANSION FACTOR [1.]
83	0.	AN KM	N. LATITUDE OF LEFT PLOT EDGE, KM
84	0.	AN KM	E. LONGITUDE OF LEFT PLOT EDGE, KM
85	35.1	AN KM	N. LATITUDE OF RIGHT PLOT EDGE, KM
86	200.	AN KM	E. LONGITUDE OF RIGHT PLOT EDGE, KM
87	50.	AN KM	DISTANCE BETWEEN RANGE TICKS, KM (0 = AUTO)
88	-3.		HEIGHT ABOVE MSL OF BOTTOM OF GRAPH, KM
89	0.		HEIGHT ABOVE MSL OF TOP OF GRAPH, KM
96	1.		DISTANCE BETWEEN DEPTH TICKS, KM (0 = AUTO)
100	9.		VVORTX3 MODEL CHECK NUMBER
102	3.		VVORTX3 BACKGROUND CURRENT DATA SET ID
103	1.02	LN M	MAXIMUM TANGENTIAL CURRENT, M/S
104	50.		RADIUS OF VORTEX CORE, KM
105	0.	AN KM	LATITUDE OF VORTEX CENTER, KM
106	150.	AN KM	LONGITUDE OF VORTEX CENTER, KM
107	1.		VERTICAL HALF-WIDTH OF VORTEX, KM

Figure 5.1. Input Data File (W array) for the sample case.

```
 108       -1.              HEIGHT OF VORTEX CENTER ABOVE MSL, KM
 150        7.             CTANH SOUND SPEED MODEL CHECK NUMBER
 152        1.              CTANH BACKGROUND SOUND SPEED DATA SET ID
 175        2.             CBLOB2 SOUND SPEED PERTURBATION MODEL CHECK NUMBER
 177        7.              CBLOB2 PERTURBATION SOUND SPEED DATA SET ID
 178       .02              MAXIMUM FRACTIONAL INCREASE IN C SQUARED
 179       -1.              HEIGHT OF MAX EFFECT ABOVE MSL, KM
 180        0.    AN KM     LATITUDE OF MAX EFFECT, KM
 181      150.    AN KM     LONGITUDE OF MAX EFFECT, KM
 182        1.              VERTICAL HALF-WIDTH, KM
 183       50.    AN KM     N-S  HALF-WIDTH, KM
 184       50.    AN KM     E-W  HALF-WIDTH, KM
 275        1.             RHORIZ RECEIVER MODEL CHECK NUMBER
 300        4.             GLORENZ BOTTOM MODEL CHECK NUMBER
 302        3.              GLORENZ BOTTOM MODEL DATA SET ID
 303       .5               HEIGHT OF RIDGE, KM ABOVE BASE
 304       10.    AN KM     N. LATITUDE OF RIDGE CENTER, KM
 305        2.    AN KM     HALF-WIDTH OF THE RIDGE, KM
 306       -3.              HEIGHT ABOVE MSL OF BASE OF RIDGE, KM
 350        1.             SHORIZ MODEL CHECK NUMBER
 352        1.              SHORIZ OCEAN SURFACE DATA SET ID
 353        0.              HEIGHT OF OCEAN SURFACE ABOVE MSL, KM
 500        1.             SLLOSS ABSORPTION MODEL CHECK NUMBER
 502        1.              SLLOSS ABSORPTION DATA SET ID
 503    0.006     AM DB     A COEFFICIENT, DB
 504   0.2635     AM DB     B COEFFICIENT, DB
 505    1000.     FQ HZ     OMEGA1, HZ
 506    1700.     FQ HZ     OMEGA2, HZ
-1                          DATA SUBSET FOR BACKGROUND CURRENT MODEL
 A  VORTEX AT LONGITUDE 150 KM E, UMAX= 1.02 M/S, R= 50 KM
 0        RETURN TO W-ARRAY DATA SET
-2                          DATA SUBSET FOR PERTURBATION CURRENT MODEL
 A  NO CURRENT PERTURBATION
 0        RETURN TO W-ARRAY DATA SET
-3                          DATA SUBSET FOR BACKGROUND SOUND-SPEED MODEL
 A  EL NINO BACKGROUND SOUND-SPEED PROFILE
 3     999.0
  LN  M        LN  M        LN  M
  0.           1532.        0.
  -20.         1531.5       -7.
  -50.         1509.        -20.
  -250.        1503.        -40.
  -450.        1485.        -300.
  -1500.       1485.        -400.
  -3000.       1508.        0.
  999.0
 0        RETURN TO W-ARRAY DATA SET
-4                          DATA SUBSET FOR SOUND-SPEED PERTURBATION MODEL
 A  2% INCREASE IN C-SQUARED AT 150 KM LON., 1 KM DEPTH, 50 KM WIDE
 0        RETURN TO W-ARRAY DATA SET
-8                          DATA SUBSET FOR RECEIVER-SURFACE MODEL
 A  RECEIVER SURFACE = SPHERE 1 KM BELOW MSL
 0        RETURN TO W-ARRAY DATA SET
-9                          DATA SUBSET FOR BACKGROUND BOTTOM MODEL
 A  RIDGE .5 KM HIGH, 2 KM WIDE AT 10 KM N LATITUDE; BASE= -3 KM
```

Figure 5.1. Input Data File (W array) for the sample case (continued).

```
  0       RETURN TO W-ARRAY DATA SET
-10                     DATA SUBSET FOR BOTTOM PERTURBATION MODEL
  A  NO BOTTOM PERTURBATION
  0       RETURN TO W-ARRAY DATA SET
-11                     DATA SUBSET FOR OCEAN SURFACE MODEL
  A  OCEAN SURFACE = SPHERE AT MSL
  0       RETURN TO W-ARRAY DATA SET
-12                     DATA SUBSET FOR OCEAN SURFACE PERTURBATION MODEL
  A  NO OCEAN SURFACE PERTURBATION
  0       RETURN TO W-ARRAY DATA SET
-17                     DATA SUBSET FOR OCEAN ABSORPTION MODEL
  A  SKRETTING-LEROY ABSORPTION FORMULA
  0       RETURN TO W-ARRAY DATA SET
-18                     DATA SUBSET FOR PERTURBATION ABSORPTION MODEL
  A  NO ABSORPTION PERTURBATION
  0       RETURN TO W-ARRAY DATA SET
  0                     **********  END OF RUN SET NUMBER 1  *********
N01-2     SAMPLE CASE FOR HARPO DOCUMENTATION        REV. 8-19-86
 71        0.           NUMBER OF INTEGRATION STEPS PER PRINT [1.E31]
 72        0.           OUTPUT RAYSETS (1=YES; 0=NO)
 73        1.           DIAGNOSTIC PRINTOUT (1=YES; 0=NO)
 81        2.           RAYPLOT PROJECTION PLANE (2 = HORIZONTAL)
 88       -1.           HEIGHT OF HORIZONTAL PLOT SECTION ABOVE MSL, KM
  0                     **********  END OF RUN SET NUMBER 2  *********
```

Figure 5.1. Input Data File (W array) for the sample case (continued).

output agrees with that given in Appendix A, you can modify the Input Data File for the sample case to make the raypath calculations you want.

The best way to be sure you have input all the required data is to fill out the forms for specifying all the ocean-model and ray parameters, as discussed for the sample case in Chapter 2. Then translate the data from those forms into the format of the Input Data File. We provide blank forms for all models and procedures in Appendix B.

5.2 Input Data Formats

The Input Data File is read by a FORTRAN program and so must conform to precise format specifications. Originally, the Input Data File consisted of a deck of 80-column punched cards, with one input parameter per card, so the data format is still specified in terms of data fields in card-image columns, even though we no longer use cards. Figure 5.1 for the sample case is an example of the proper format.

Looking at Figure 5.1, you will notice that the first part of the file consists of a series of lines that begin with a positive integer. Each of these lines specifies an element of a Data Input Array, W(n). This format goes as far as the line that begins with 506. At the line beginning with -1, the data format changes to accept tabular input data. First, we will explain how to specify data to be read into W(n); then we will explain how to enter data in the tabular format.

5.3 Specifying the W-Array Input

The W-array input format consists of a single 80-column line with four data fields: n, W(n), unit conversion characters, and a description field.

The first three card-image columns contain the first data field in I3 format and specify the index, n, of the array W(n). The value of n must be between 0 and 999, and if there are fewer than three digits, the entry must be right-justified, or else FORTRAN will assume that trailing zeroes are appended to fill out the three columns. If two or more lines begin with the same value of n, the last one prevails.

The second field, columns 4-17, contains the value of W(n) in E14.7 format. The value can be entered in either E or F format, but if the E format is used, the exponent must be right-justified within the 14 spaces, or else zeroes will be appended.

The third field, columns 18-24, contains characters that tell the program to convert the units of the data, as input, to units used by the program. The present choices available for input in this field are given in Table 5.1; the characters must be input in exactly the columnar format shown.

Table 5.1--Units conversion on input

Units of input value*	Meaning	Conversion needed	Value stored by read-in routine
AN RD	Angle in radians	None	V_i
AN DG	Angle in degrees	deg to rad or deg/s to rad/s	V_i $\pi/180$**
AN KM	Central earth angle in kilometers	km to rad	V_i/r_e***
LN KM	Length in kilometers	None	V_i
LN M	Length in meters	m to km	$V_i/1000$
LN NM	Length in nautical miles	nmi to km	$1.852\ V_i$
LN FT	Length in feet	ft to km	$3.048006096 \times 10^{-4}\ V_i$
FQ HZ	Frequency in hertz	Hz to rad/s	$2\pi V_i$
FQ S	Frequency expressed as a period in seconds	Period in seconds to frequency in rad/s	$2\pi/V_i$
AM NP	Amplitude in nepers	None	V_i
AM DB	Amplitude in decibels	dB to nepers	$V_i \log_c 10/10$
T****	Transmitter height relative to bottom instead of mean sea level	Add bottom height to transmitter height	V_i (also, a flag is set)*****

* The five characters listed are to be put in card-image columns 18 through 22 of the W-array input value to be converted, or put above the data-input column of tabular input. For three-column tabular input, for example, the five characters should be in columns 1-13, 14-26, and 27-39. The five characters are automatically put in the appropriate place when using the WMOD editor.

** V_i is the input value.

*** r_e is the radius of the earth. The current value of W(1) in the W array is used for this conversion.

**** Applies only for input to W(3) (transmitter height). The "T" must be put in card-image column 24.

***** At the start of each ray, the status of the flag is checked. If the flag is set, then the bottom height (negative) at the longitude and latitude of the transmitter is added to the transmitter height. For general bottom models, the bottom height at a given longitude and latitude can only be estimated. For all of the presently available bottom models, the estimate gives an exact result, however, because $\partial g/\partial r$ is constant.

The fourth field (columns 25-80) contains descriptive comments, which aid the user in setting up the table. These comments are optional and arbitrary as far as HARPO is concerned, but for n ≥ 100 and divisible by 25, the first word in the comment field is read when WMOD is used for editing and must be a valid model name. This convention will be described in a report about the supplementary programs. Where practical, the comments should describe the function of all acceptable values of the parameter, not just the present value, so that the comment would not have to be changed when the parameter is changed. We have included nonzero initial values in the comment field, where applicable. To make it easier to see the model groupings, we have adopted the convention of indenting the comments that describe model parameters.

5.3.1 Initialization of the Input Data Parameters

Before reading the Input Data File, the program initializes all of the input parameters, W(n). Most are set to zero, but a few are given nonzero initial values that correspond to common usage. An example is the latitude of the north pole of the computational coordinate system, W(24), which usually has a value of $\pi/2$. Section 5.3.2 denotes those nonzero initial values by parentheses. These initial values can be overridden by the Input Data File (including a value of zero), but if no value is specified for a W(n) in the Input Data File, then its initial value prevails.

In addition, some initial values are given "zero-override" priority, which means that W(n) assumes its nonzero initial value if no value is input, <u>but also if a zero is input</u>. This zero override operates when a zero value would produce meaningless results or cause difficulty in program execution. An example is the plot expansion factor, W(82). Fig. 5.1 and Sec. 5.3.2 denotes by square brackets the nonzero initial values that override zero.

To help the user keep track of the unit conversions and initializations, all nonzero W(n) values, in the units actually used by HARPO, are listed at the beginning of the printout (Appendix A). In the examples given above, if W(24) were given a value of zero in the Input Data File, no value would be printed for W(24), indicating that HARPO will use W(24) = 0. On the other hand, if W(82) is given a zero value in the Input Data File, a message is printed indicating "INPUT OVERRIDDEN," and the nonzero override value is printed.

5.3.2 Explanation of the Input Data Parameters

Because HARPO has evolved from ray-tracing programs for other media, some values of the input parameter index, n, are not used in HARPO, but may be used in other versions of the program. As far as possible, n is assigned consistently among the different versions, and blocks of n are assigned to groups of related parameters.

Following is a list of all the input parameters used by HARPO, with a description of their meanings and idiosyncrasies. Those with nonzero initial values need not be entered in the Input Data File, if the value is what you want. If no initial value is indicated, a zero will be assigned if you leave it out of the Input Data File. The default units given in parentheses are those which are assumed if no unit conversions are put into columns 18-24. Also included in the table is the FORTRAN name (where one exists) assigned (in EQUIVALENCE statements) to each variable in the program. Those labeled "not used" can be assigned to additional input parameters, but those labeled "used by other programs" or "used internally" should not be used.

W(1) EARTHR (6370.) -- Radius (kilometers) of the earth, r_e, to mean sea level (MSL). Can be set to a very large value for a "flat-earth" approximation.

W(2) RAY -- Used by other programs.

W(3) XMTRH -- Height (kilometers) of the transmitter (source) above mean sea level. Depth values are negative. If there is a T in column 24, it is the height above the ocean bottom.

W(4) TLAT -- North geographic latitude (radians) of the transmitter. Can be entered in kilometers (or degrees) by putting AN KM (or AN DG) beginning in column 18.

W(5) TLON -- East geographic longitude (radians) of the transmitter. Can be entered in kilometers (or degrees) by putting AN KM (or AN DG) beginning in column 18.

W(6) OW -- Used internally.

W(7) FBEG -- Initial acoustic wave frequency (rad/s). Can be entered in Hz (or period in seconds) by putting FQ HZ (or FQ S) beginning in column 18.

W(8) FEND -- Final frequency (rad/s). Can be entered in Hz (or period in seconds) by putting FQ HZ (or FQ S) beginning in column 18.

W(9) FSTEP -- Step in frequency (rad/s). Can be entered in Hz (or period in seconds) by putting FQ HZ (or FQ S) beginning in col 18. Set = 0 for no stepping.

W(10) AZ1 -- Used internally.

W(11) AZBEG -- Initial azimuth angle (radians east of north) of transmission. Can be entered in degrees by putting AN DG beginning in column 18.

W(12) AZEND -- Final azimuth angle (radians east of north) of transmission. Can be entered in degrees by putting AN DG beginning in column 18.

W(13) AZSTEP -- Step in azimuth angle (radians east of north) of transmission. Can be entered in degrees by putting AN DG beginning in column 18. Set = 0 for no stepping.

W(14) BETA -- Used internally.

W(15) ELBEG -- Initial elevation angle (radians above horizontal) of transmission. Can be entered in degrees by putting AN DG beginning in column 18.

W(16) ELEND -- Final elevation angle (radians above horizontal) of transmission. Can be entered in degrees by putting AN DG beginning in column 18.

W(17) ELSTEP -- Step in elevation angle (radians) of transmission. Can be entered in degrees by putting AN DG beginning in column 18. Set = 0 for no stepping.

W(18) -- Used internally.

W(19) -- Set = 1 to stop a ray when it strikes the ocean bottom, printing BOTM ABS.

W(20) RCVRH -- Height (kilometers) above sea level of the receiver surface when model RHORIZ is used; height of the receiver surface above the bottom when model RTERR is used.

W(21) ONLY -- Set = 1 to stop elevation-angle stepping when ray goes out of bounds.

W(22) HOP -- Maximum number of ray hops (intersections with or closest approaches to the receiver surface); ray calculation stops when reached, printing MAX HOPS. Closest approaches count as two hops.

W(23) MAXSTP (1000.) -- Maximum number of integration steps per hop; ray calculation stops when reached, printing STEP MAX.

W(24) PLAT ($\pi/2$) -- Geographic latitude (radians) of the north pole of the computational coordinate system.

W(25) PLON -- Geographic longitude (radians) of the north pole of the computational coordinate system.

W(26) HMAX (500.) -- Maximum ray height (kilometers) above MSL; ray calculation stops if exceeded, printing MAX HT.

W(27) HMIN -- Minimum ray height (kilometers) above MSL; calculation stops if ray goes below this height, printing MIN HT.

W(28) RGMAX -- Maximum ground range (kilometers at MSL) of the ray from the transmitter; ray calculation stops if exceeded, printing MAX RANG.

W(29) RAYFNC -- A seven-digit number used to select execution of HARPO and supplementary programs. To run HARPO, use 100. Setting = 0 is the same as all ones and will run HARPO.

W(30)-W(32) -- Used by other programs.

W(33) EXTINC (999.999) -- Maximum absorption (dB); ray calculation stops if value exceeded, printing EXTINC. Set = 0 for no maximum.

W(34)-W(40) -- Not used.

W(41) INTYP (3.) -- Integration type --
= 1 for Runge-Kutta integration without error checking;
= 2 for Adams-Moulton integration without error checking;
= 3 for Adams-Moulton integration with relative-error checking;
= 4 for Adams-Moulton integration with absolute-error checking.

W(42) MAXERR (1.E-4) -- Maximum allowable integration error per step. RKAM routine decreases step size to achieve this error.

W(43) ERATIO (50.) [50.] -- Ratio of maximum to minimum single-step integration error; RKAM increases step size when error is smaller than W(42) by this factor.

W(44) STEP1 (1.0) -- Initial integration step size (seconds).

W(45) STPMAX (100.) -- Maximum integration step size (seconds).

W(46) STPMIN (1.E-8) -- Minimum integration step size (seconds).

W(47) FACTR (.5) [0.5] -- Factor multiplying integration step size when decreasing step size.

W(48)-W(56) Not used.

W(57) -- Phase-time integration: 0 to not integrate; 1 to integrate; 2 to integrate and print.

W(58) -- Absorption integration: 0 to not integrate; 1 to integrate; 2 to integrate and print.

W(59) -- Doppler shift integration: 0 to not integrate; 1 to integrate; 2 to integrate and print.

W(60) -- Path-length integration: 0 to not integrate; 1 to integrate; 2 to integrate and print.

W(61)-W(70) -- Assigned to future integration options.

W(71) SKIP -- [1.E31] Number of integration steps between printed lines. Set = 1 to print every step; = 0 to suppress periodic printing.

W(72) RAYSET -- Write machine-readable raysets to file PUNCH: 1 = yes; 0 = no.

W(73) PCNTRW -- Add diagnostic printout lines: 1 = yes; 0 = no.

W(74) PRTSRP -- Produce normal printout every W(71) steps: 0 = yes; 1 = no. Also produces printout at special events.

W(75) HITLET [.15] -- Height (inches on our plotter) of lettering on graphs. "FULANN" in description field activates publication-quality lettering on graphs when read by WMOD. Any other comment in description field produces draft-quality lettering.

W(76) BINRAY -- Write binary raypath description to file TAPE6: 1 = yes; 0 = no.

W(77) PAGLIN (66.) -- Page length (lines) for printout.

W(78)-W(79) Not used.

W(80) APOG, PRIGEE - 0 for normal rayplots; 1 for apogee plots.

W(81) PLT -- Rayplot projection:
1 = vertical plane, polar projection, rectangular expansion;
2 = horizontal plane, lateral expansion;
3 = vertical plane, polar plot, radial expansion;
4 = vertical plane, rectangular plot.
Make W(81) negative to superimpose plot on that from previous runset.

W(82) PFACTW [1.] -- Vertical or lateral expansion factor for rayplot.

W(83) LLAT -- North latitude (radians) of left edge of plot. To enter in degrees (kilometers) put AN DG (AN KM) beginning in column 18.

W(84) LLON -- East longitude (radians) of left edge of plot. To enter in degrees (kilometers) put AN DG (AN KM) beginning in column 18.

W(85) RLAT -- North latitude (radians) of right edge of plot. To enter in degrees (kilometers) put AN DG (AN KM) beginning in column 18.

W(86) RLON -- East longitude (radians) of right edge of plot. To enter in degrees (kilometers) put AN DG (AN KM) beginning in column 18.

W(87) TIC -- Distance (radians) between tick marks on horizontal axis of plot. To enter in kilometers, put AN KM beginning in column 18.

W(88) HB -- Height (kilometers) of the bottom of the graph above MSL.

W(89) HT -- Height (kilometers) of the top of the graph above MSL.

W(90)-W(95) -- Used by other programs.

W(96) TICV -- Distance (kilometers) between tick marks on vertical axis of plot. Notice that the default units are kilometers for the vertical ticks and radians for the horizontal ticks.

W(97)-W(99) -- Used by other programs.

W(100) UMODEL -- Check number for background current model.

W(101) UFORM -- format code for background current model.

W(102) UID -- Data-set ID for background current model.

W(103)-W(124) Parameters for background current model.

W(125) DUMODEL -- Check number for perturbation current model.

W(126) DUFORM -- Format code for perturbation current model.

W(127) DUID -- Data-set perturbation current model.

W(128)-W(149) -- Parameters for perturbation model.

W(150) CMODEL -- Check number for background sound speed model.

W(151) CFORM -- Format code for background sound speed model.

W(152) CID -- Data-set ID for background sound speed model.

W(153)-W(174) -- Parameters for background sound speed model.

W(175) DCMODEL -- Check number for perturbation sound speed model.

W(176) DCFORM -- Format code for perturbation sound speed model.

W(177) DCID -- Data-set for perturbation sound speed model.

W(178)-W(199) -- Parameters for perturbation sound speed model.

W(275) RMODEL -- Check number for receiver surface model.

W(276) RFORM -- Format code for receiver surface model.

W(277) RID -- Data-set ID for receiver surface model.

W(278)-W(299) -- Parameters for receiver surface model.

W(300) GMODEL -- Check number for background bottom model.

W(301) GFORM -- Format code for background bottom model.

W(302) GID -- Data-set ID for background bottom model.

W(303)-W(324) -- Parameters for background bottom model.

W(325) DGMODEL -- Check number for perturbation bottom model.

W(326) DGFORM -- Format code for perturbation bottom model.

W(327) DGID -- Data-set ID for perturbation bottom model.

W(328)-W(349) -- Parameters for perturbation bottom model.

W(350) SMODEL -- Check number for background ocean surface model.

W(351) SFORM -- Format code for background ocean surface model.

W(352) SID -- Data-set ID for background ocean surface model.

W(353)-W(374) -- Parameters for background ocean surface model.

W(375) -- Check number for ocean-surface perturbation model.

W(376) -- Format code for ocean-surface perturbation model.

W(377) -- Data-set ID for ocean-surface perturbation model.

W(378)-(399) -- Parameters for ocean-surface perturbation model.

W(400)-W(500) -- Parameters for supplementary programs.

W(500) AMODEL -- Check number for absorption model.

W(501) AFORM -- Format code for absorption model.

W(502) AID -- Data-set ID for absorption model.

W(503)-W(524) -- Parameters for absorption model.

W(525) DAMODEL -- Check number for perturbation absorption model.

W(526) DAFORM -- Format code for perturbation absorption model.

W(527) DAID -- Data-set ID for perturbation absorption model.

W(528)-W(549) -- Parameters for perturbation absorption model.

5.4 Specifying Tabular Input

In addition to providing values to the W(n) array, the Input Data File lets you enter tabular data to be used by ocean model subroutines. In the following discussion, refer to the Input Data File for the sample case (Fig. 5.1) for examples of tabular data.

5.4.1 Changing to the Tabular Format

When the sign of the W-array index n is read (in columns 1-3), it is checked for a valid negative value, which signals a change in the format of the data to follow. Any valid negative value selects a corresponding "input common block" that has been dedicated to a particular model subroutine. Table 5.2 shows which values select which common blocks. The line in Figure 5.1 that begins with -1 and has the comment "ENTER DATA SUBSET FOR BACKGROUND CURRENT MODEL" is an example selecting common block /B1/ to receive data. Because this line must conform to the W-array format, comments must begin after column 24.

5.4.2 The Format Line

The line following the negative value of n begins with an integer that specifies the format of the data to follow. The formats are numbered according to the method shown in Table 5.3.

Format A is alphanumeric and lets you enter descriptive comments on the rest of the line that begins with the format number. That comment is reproduced at the beginning of the program printout. The comments in Figure 5.1 that follow an A in col. 3 are examples of brief model descriptions and correspond to those enetered on the forms specifying ocean models.

Formats 1 through 6 specify 1 to 6 columns, respectively, of floating-point data. If format 1 through 6 is selected, then columns 4-13 of the format line must contain an end-of-data terminator, such as 999.0 in the sample case. The rest of the format line should be blank.

Table 5.2--Description of the identifying numbers in the first three columns of the Input Data File

Code number	Prefix for common-block variables*	Description
1-999		Input to elements 1-999 of the W array as described in Table 4.4 and Section 5.3.2
-1	U	Signals start of tabular input to common block /B1/ (current velocity) (see Table 4.2)
-2	DU	Signals start of tabular input to common block /B2/ (perturbation current velocity)
-3	C	Signals start of tabular input to common block /B3/ (sound speed)
-4	DC	Signals start of tabular input to common block /B4/ (perturbation sound speed)
-8	R	Signals start of tabular input to common block /B8/ (receiver surface)
-9	G	Signals start of tabular input to common block /B9/ (bottom)
-10	DG	Signals start of tabular input to common block /B10/ (bottom perturbation)
-11	S	Signals start of tabular data input to common block /B11/ (ocean surface)
-12	DS	Signals start of tabular data input to common block /B12/ (ocean-surface perturbation)
-17	V	Signals start of tabular input to common block /B17/ (absorption)
-18	DV	Signals start of tabular input to common block /B18/ (absorption perturbation)
0		Signals end of tabular input to one of the above common blocks, or if that is not appropriate, end of the input data

* See Table 4.5.

Table 5.3--Tabular input data formats available*

Format number	Purpose of format	Number of data lines used	Line number	Format	Card columns	Data read
0	signify end of tabular input	1	1(return line)	I3	1 -3	format number (=0)
A	enter comments that describe the ocean model	1	1	A	1 -3 4 -80	format number (=A) descriptive comments
1-6	enter 1-6 column tabular data	variable	1	I3	1 -3	format number
				G13.6	4 -16	terminator
			2(units line)	A	1 -13	units conversion for first data column
				A	14-26	units conversion for second data column
				A	27-39	units conversion for third data column
				A	40-52	units conversion for fourth data column
				A	53-65	units conversion for fifth data column
				A	60-78	units conversion for sixth data column
			3-(data lines)	BZ,10G13.6	1 -78	up to 6 columns of data
			last	BZ,10G13.6	1 -78	terminator (in data column after the last data column, but not in a data column larger than the format number)

*Table 4.5 explains how the data is stored in a common block such as /B3/. The descriptive comments are stored in elements 32-41 of the common block. The number of tabular data elements read is stored in element 42 of the common block. The first data column is stored beginning in element 43 of the common block. The second data column is stored beginning in element number (= 43 + maximum allowed number of rows of tabular data) in the common block. The maximum number of rows of tabular data is set by the model subroutine (see Table 4.5).

5.4.3 The Units Line

Following the format line is a line containing unit-conversion specifications for the columnar data to follow. The conversion specifications follow the conventions described in Table 5.1. The 5 characters can be placed anywhere in the same 13-character columns as the tabular data to be converted, as long as at least one space separates adjacent conversion specifications.

5.4.4 The Data Lines

Any number of data lines can follow the format line. They must contain numeric data in the format specified in the format line (Table 5.3) and must be terminated with the number given in the format line as specifying end-of-data. Data values are read until the terminator is encountered. The data following the "EL NINO BACKGROUND SOUND-SPEED PROFILE" comment in Figure 5.1 is an example of 3-column data entry (format 3) into the common block /B3/, which contains data used by sound-speed model CTANH. If the number of data values read in exceeds the maximum allocated in the model subroutine, an error will occur, and the program will stop.

5.4.5 The Return Line

After the end-of-data number signifies the end of tabular input, you will usually enter a line beginning with a 0 in column 3 to return to the W-array data format.

PART III: HOW THE PROGRAM WORKS

6. The Ray-Tracing Equations

6.1 Hamilton's Equations in Spherical Polar Coordinates

HARPO calculates raypaths by numerically integrating Hamilton's equations. Lighthill (1965, 1978) gives Hamilton's equations in four dimensions (three spatial and one temporal) for Cartesian coordinates. Haselgrove (1954) gives Hamilton's equations in three dimensions for spherical polar coordinates, a more useful coordinate system for geophysical media. Combining the two gives Hamilton's equations in four dimensions in which the three spatial coordinates are earth-centered spherical polar (see Table 6.1 for a definition of the symbols):

$$\frac{dr}{d\tau} = \frac{\partial H}{\partial k_r} , \tag{6.1}$$

$$\frac{d\theta}{d\tau} = \frac{1}{r} \frac{\partial H}{\partial k_\theta} , \tag{6.2}$$

$$\frac{d\phi}{d\tau} = \frac{1}{r \sin\theta} \frac{\partial H}{\partial k_\phi} , \tag{6.3}$$

$$\frac{dt}{d\tau} = - \frac{\partial H}{\partial \omega} , \tag{6.4}$$

$$\frac{dk_r}{d\tau} = - \frac{\partial H}{\partial r} + k_\theta \frac{d\theta}{d\tau} + k_\phi \sin\theta \frac{d\phi}{d\tau} , \tag{6.5}$$

$$\frac{dk_\theta}{d\tau} = \frac{1}{r} \left(- \frac{\partial H}{\partial \theta} - k_\theta \frac{dr}{d\tau} + k_\phi \, r \cos\theta \frac{d\phi}{d\tau} \right) , \tag{6.6}$$

$$\frac{dk_\phi}{d\tau} = \frac{1}{r \sin\theta} \left(- \frac{\partial H}{\partial \phi} - k_\phi \sin\theta \frac{dr}{d\tau} - k_\phi \, r \cos\theta \frac{d\theta}{d\tau} \right) , \tag{6.7}$$

$$\frac{d\omega}{d\tau} = \frac{\partial H}{\partial t} \quad . \tag{6.8}$$

The variables r, θ, ϕ are the (Earth-centered) spherical polar coordinates of a point on the raypath; k_r, k_θ, and k_ϕ are the local Cartesian components of the propagation vector (a vector whose magnitude,

Table 6.1--The more important symbols and their definitions

A	In Section 6.1.2, absorption in dB
C	Sound speed
C_{ref}	A reference sound speed
f	Wave frequency (in Hz)
Δf	Frequency shift of a wave due to a time-varying medium
H	Hamiltonian
k_r, k_θ, k_ϕ	Components of the propagation vector in the r, θ, ϕ directions--a vector normal to the wave front having a magnitude $2\pi/\lambda = \omega/v$
k_{disp}	Complex wave number determined by the dispersion relation
P	Phase path length, phase of the wave divided by the reference wave number ($2\pi/\lambda_o = \omega/C_{ref}$)
P'	Group path length = $C_{ref}t$
r, θ, ϕ	Spherical polar coordinates of a raypath point
s	Geometric raypath length
t	Time of travel of a wave packet (in some cases, used to express the time dependence of the propagation medium)
V	Current velocity
V_r, V_θ, V_ϕ	Components of the current velocity in the r, θ, ϕ direction
v	Wave phase velocity
θ	Colatitude in spherical polar coordinates
λ	Wavelength
$\lambda_o = (2\pi/\omega)C_{ref}$	Reference wavelength
τ	Independent variable in Hamilton's equations (no physical significance)
ϕ	Longitude in spherical polar coordinates
Ω	$= \omega - \vec{k}\cdot\vec{V}$, the intrinsic wave frequency, the wave frequency as seen by an observer moving with the medium
$\omega = 2\pi f$	Radian wave frequency
$\Delta\omega = 2\pi\Delta f$	Radian frequency shift

$$k = \sqrt{k_r^2 + k_\theta^2 + k_\phi^2} = 2\pi/\lambda \quad , \tag{6.9}$$

is the wave number, and that points in the wave normal direction) in the r, θ, and ϕ directions; t is time--in (6.4) it is the propagation time of a wave packet; in (6.8) it expresses the variation with time of a time-varying medium; τ is a parameter whose value depends on the choice of the Hamiltonian H. Section 6.4 explains how the Hamiltonian is defined.

For actual calculation, HARPO uses group path $P' = C_{ref}\ t$ (where C_{ref} is a standard reference speed) as the independent variable because the derivatives with respect to P' are independent of the choice of Hamiltonian, allowing the program to switch Hamiltonians in the middle of a path. This choice automatically causes the program to take smaller steps in real path length where the calculations are more critical, as when refractive index varies rapidly. These equations are obtained by dividing (6.1) through (6.8) by C_{ref} times (6.4):

$$\frac{dr}{dP'} = -\frac{1}{C_{ref}} \frac{\partial H/\partial k_r}{\partial H/\partial \omega} \quad , \tag{6.10}$$

$$\frac{d\theta}{dP'} = -\frac{1}{rC_{ref}} \frac{\partial H/\partial k_\theta}{\partial H/\partial \omega} \quad , \tag{6.11}$$

$$\frac{d\phi}{dP'} = -\frac{1}{rC_{ref}\ \sin\theta} \frac{\partial H/\partial k_\phi}{\partial H/\partial \omega} \quad , \tag{6.12}$$

$$\frac{dk_r}{dP'} = \frac{1}{C_{ref}} \frac{\partial H/\partial r}{\partial H/\partial \omega} + k_\theta \frac{d\theta}{dP'} + k_\phi \sin\theta \frac{d\phi}{dP'} \quad , \tag{6.13}$$

$$\frac{dk_\theta}{dP'} = \frac{1}{r} \left(\frac{1}{C_{ref}} \frac{\partial H/\partial \theta}{\partial H/\partial \omega} - k_\theta \frac{dr}{dP'} + k_\phi\ r\ \cos\theta \frac{d\phi}{dP'}\right) \quad , \tag{6.14}$$

$$\frac{dk_\phi}{dP'} = \frac{1}{r\ \sin\theta} \left(\frac{1}{C_{ref}} \frac{\partial H/\partial \phi}{\partial H/\partial \omega} - k_\phi \sin\theta \frac{dr}{kP'} - k_\phi\ r\ \cos\theta\ \frac{d\theta}{dP'}\right) \quad , \tag{6.15}$$

$$\frac{d(\Delta f)}{dP'} = \frac{1}{2\pi} \frac{d\Delta\omega}{dP'} = \frac{1}{2\pi} \frac{d\omega}{dP'} = -\frac{1}{2\pi} \frac{\partial H/\partial t}{\partial H/\partial \omega} \quad . \tag{6.16}$$

Equation (6.16) for the frequency shift of a wave propagating through a time-varying medium follows directly from Hamilton's equations (6.4) and (6.8). An alternative derivation is given by Bennett (1967). Large fre-

quency shifts should be accumulated along the raypath and the shifted frequency used in calculations at each point on the raypath. Equations (6.1) through (6.8) imply that all eight dependent variables vary along the path, and that at each point on the path the instantaneous value of all parameters (including frequency) is used in further evaluations of the equations. However, the time variation of the ocean due to natural causes is so slow that the resulting frequency shifts have negligible effect on the propagation. For this reason, HARPO calculates frequency shift to compare with frequency-shift measurements, but does not adjust the frequency of the wave used in the propagation calculations.

The first six differential equations, (6.10) through (6.15), are always integrated. By setting W(59), the user can choose whether to have the program integrate (6.16) to calculate the frequency shift.

6.1.1 Phase Path

Three other quantities can be calculated by integration along the raypath. The phase path P (phase divided by the reference wave number $2\pi/\lambda_o = \omega/C_{ref}$) is calculated by integrating

$$\frac{dP}{dP'} = \frac{C_{ref}}{\omega}\left(k_r \frac{dr}{dP'} + k_\theta\, r \frac{d\theta}{dP'} + k_\phi\, r \sin\theta \frac{d\phi}{dP'}\right)$$

$$= -\frac{1}{\omega}\frac{k_r \frac{\partial H}{\partial k_r} + k_\theta \frac{\partial H}{\partial k_\theta} + k_\phi \frac{\partial H}{\partial k_\phi}}{\partial H/\partial\omega} \quad . \qquad (6.17)$$

6.1.2 Absorption

If the absorption per wavelength is small (as it must be for this type of ray tracing to be valid), then an approximate formula can be integrated to give the absorption in decibels:

$$\frac{dA}{dP'} = -\frac{10}{\log_e 10}\,\frac{\omega}{C_{ref}}\,\frac{\text{imag}\,(k^2_{disp})}{k_r^2 + k_\theta^2 + k_\phi^2}\,\frac{dP}{dP'}$$

$$= \frac{10}{\log_e 10}\,\frac{\text{imag}\,(k^2_{disp})}{k_r^2 + k_\theta^2 + k_\phi^2}\,\frac{k_r\,\frac{\partial H}{\partial k_r} + k_\theta\,\frac{\partial H}{\partial k_\theta} + k_\phi\,\frac{\partial H}{\partial k_\phi}}{C_{ref}\,\partial H/\partial\omega}\quad , \qquad (6.18)$$

where k_{disp} is the (complex) wave number determined by the dispersion relation.

There are two ways to calculate the imaginary part of K^2_{disp} to give the absorption in (6.18). Whenever possible, one calculates K^2_{disp} from a complex dispersion relation such as the Appleton-Hartree formula for radio waves in the ionosphere or from the Navier-Stokes equations for acoustic-gravity waves in the atmosphere. When that is not possible, one uses

$$K^2_{imag} = -\alpha\,\sqrt{K^2_{real}}\quad , \qquad (6.19)$$

where α is an absorption coefficient in nepers per kilometer, derived either from a theoretical or empirical model or from measurements.

6.1.3 Path Length

The geometrical path length of the ray can be calculated by integrating

$$\frac{ds}{dP'} = \sqrt{\left(\frac{dr}{dP'}\right)^2 + r^2\left(\frac{d\theta}{dP'}\right)^2 + r^2\sin^2\theta\left(\frac{d\phi}{dP'}\right)^2}$$

$$= \frac{\sqrt{\left(\frac{\partial H}{\partial k_r}\right)^2 + \left(\frac{\partial H}{\partial k_\theta}\right)^2 + \left(\frac{\partial H}{\partial k_\phi}\right)^2}}{C_{ref}\,|\partial H/\partial\omega|}\quad . \qquad (6.20)$$

The user can choose to integrate and print frequency shift, phase time, absorption, or path length using Equations (6.16), (6.17), (6.18), or (6.20) by setting the appropriate values of W(59), W(57), W(58), W(60), respectively, in the Input Data File (Figures 2.14 and 2.19).

The user can add other differential equations to the program by modifying HAMLTN, the subroutine that evaluates Hamilton's equations.

The Hamiltonian and its derivatives are calculated by one of the versions of dispersion-relation subroutine (with entry point DISPER), which also calculates k_{disp}^2.

6.2 Numerical Integration

Subroutines RKAM and RKAM1 integrate the differential equations numerically using an Adams-Moulton predictor-corrector method with a Runge-Kutta starter. RKAM1 was adapted from a program in the CDC CO-OP library called RKAMSUB, written by G. J. Lastman and dated March 1964. The program executes one integration step by one of four methods the user can specify using W(41). The subroutine is called once for each advance of the independent variable. The flow charts of Figures 6.1 through 6.5 show how the integration routine works.

Usually, RKAM1 is run in mode 3, that is, Adams-Moulton integration with relative-error checking. The user can trade execution time for accuracy by varying the single-step integration error, W(42). RKAM1 will increase or decrease the step length to maintain the specified error. Table 6.2 gives an

Table 6.2--Run time vs. accuracy

Maximum Integration Error, W(42)	Run Time Sec	Range error
10^{-9}	9.9	$<10^{-7}$
10^{-8}	7.1	$<10^{-7}$
10^{-7}	4.5	1×10^{-7}
10^{-6}	3.3	1×10^{-6}
10^{-5}	2.3	6×10^{-6}
10^{-4}	1.7	1.7×10^{-4}
10^{-3}	1.2	5.7×10^{-4}

Note: Data obtained using model OT2; elevation angle = 4.46°; range = 1000 km; computer: CYBER 750.

idea of how the tradeoff works for a single ray calculation using HARPO and a model of the ocean sound channel described by Georges et al. (1986).

The user can vary other parameters, W(43) - W(47), that control the way RKAM1 adjusts its step length and controls errors (see Sec. 5.3.2), but the initial values assigned in the sample case have been found to work best for most cases met in practice. If the scale of the model differs greatly from the sample case, the initial step length, W(44), should be adjusted.

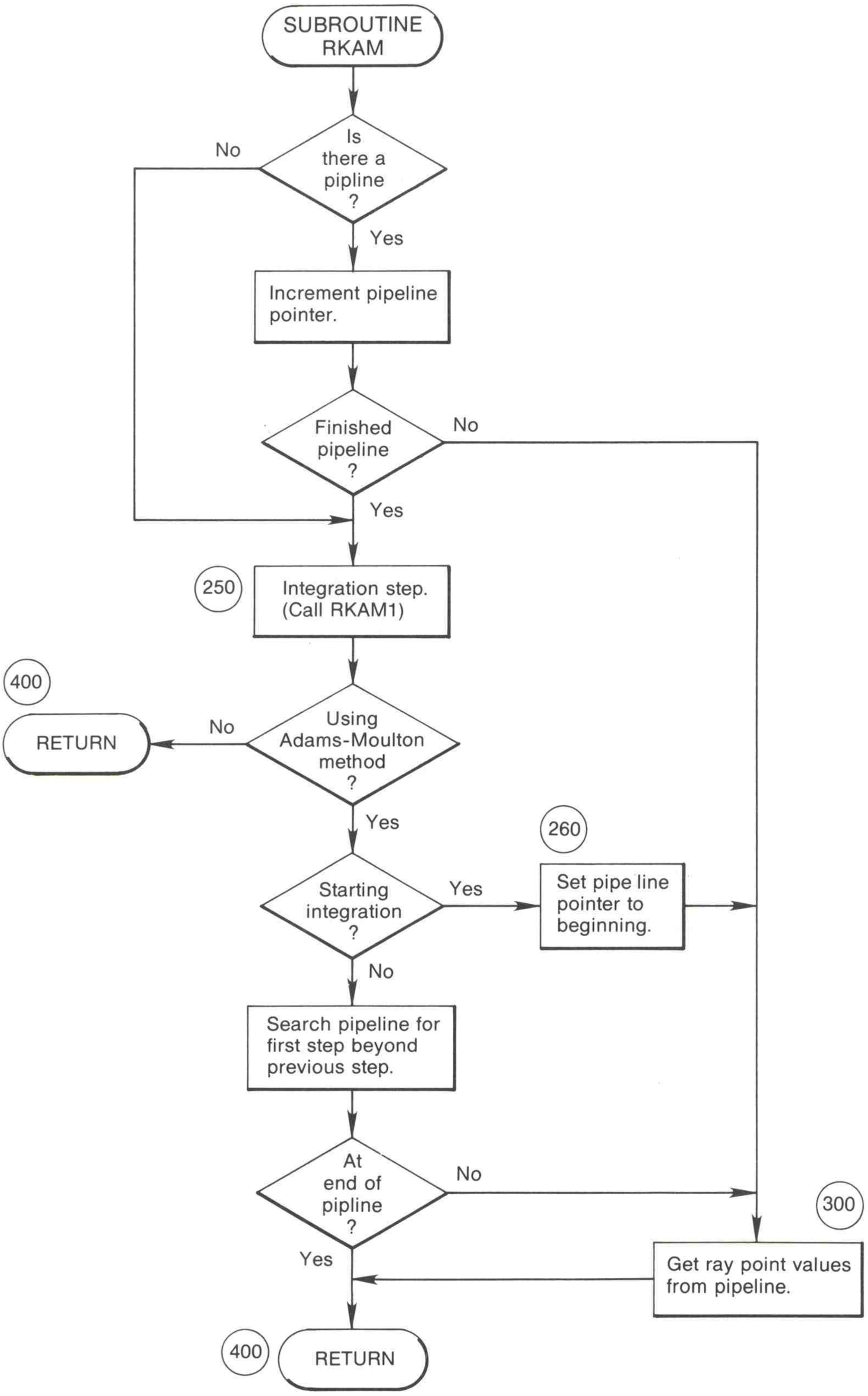


Figure 6.1. Flow chart for subroutine RKAM. The term "pipeline" refers to the sequence of four consecutive integration steps kept by the Adams-Moulton integration method. Circled numbers refer to program statement numbers.

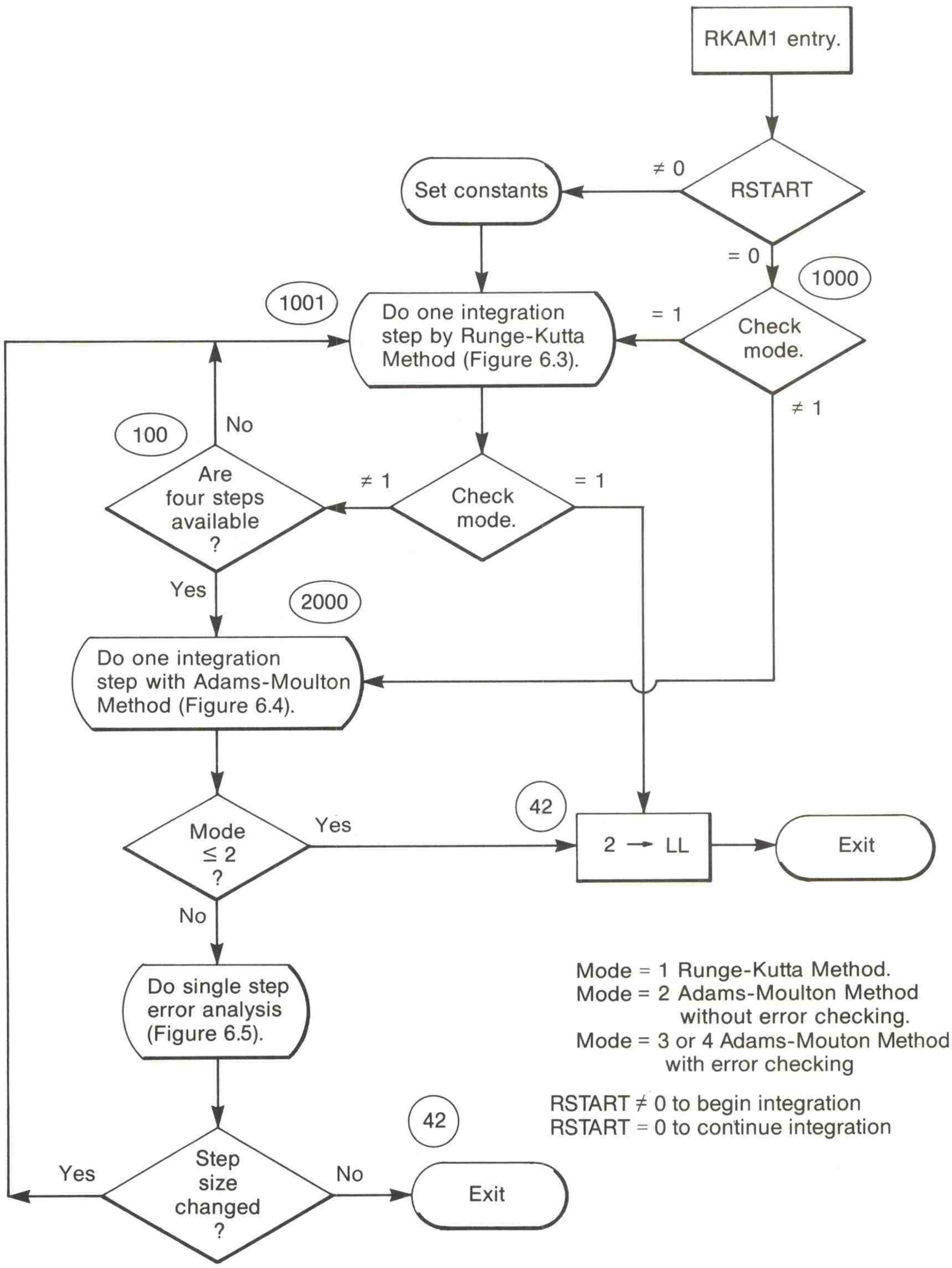


Figure 6.2. Flow chart for subroutine RKAM1. Circled numbers refer to program statement numbers.

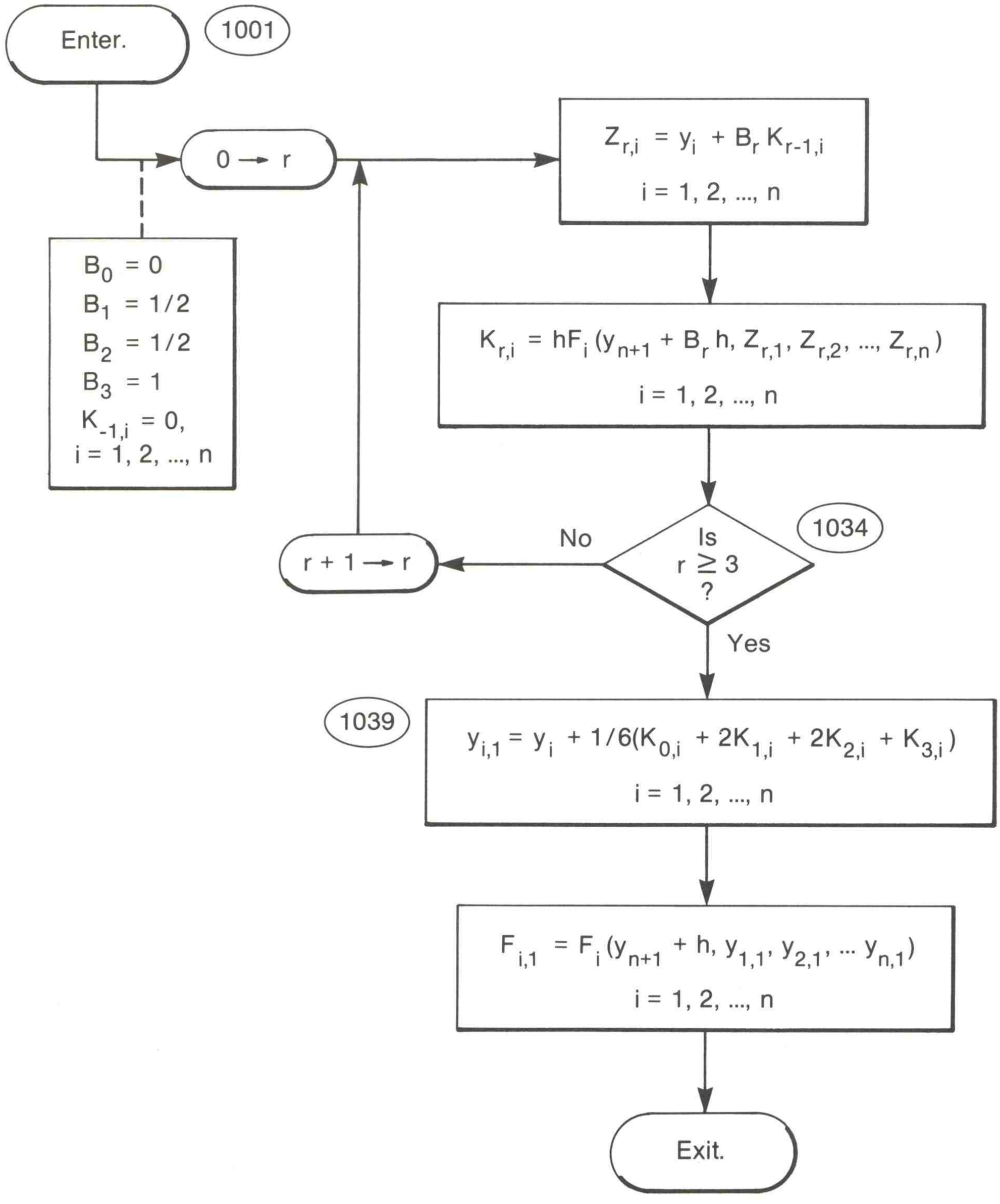


Figure 6.3. Flow chart for the Runge-Kutta procedure. The variables y_1, y_2, ... y_n are the dependent variables, n is the order of the system, and y_{n+1} is the independent variable. Circled numbers refer to program statement numbers. h is the step size. $F_{i,1}$ is the derivative of y_i. $Z_{j,i}$ are 4 estimates of y_i at the beginning, middle, and end of a step. $K_{j,i}$ is Δy_i estimated at $Z_{j,i}$.

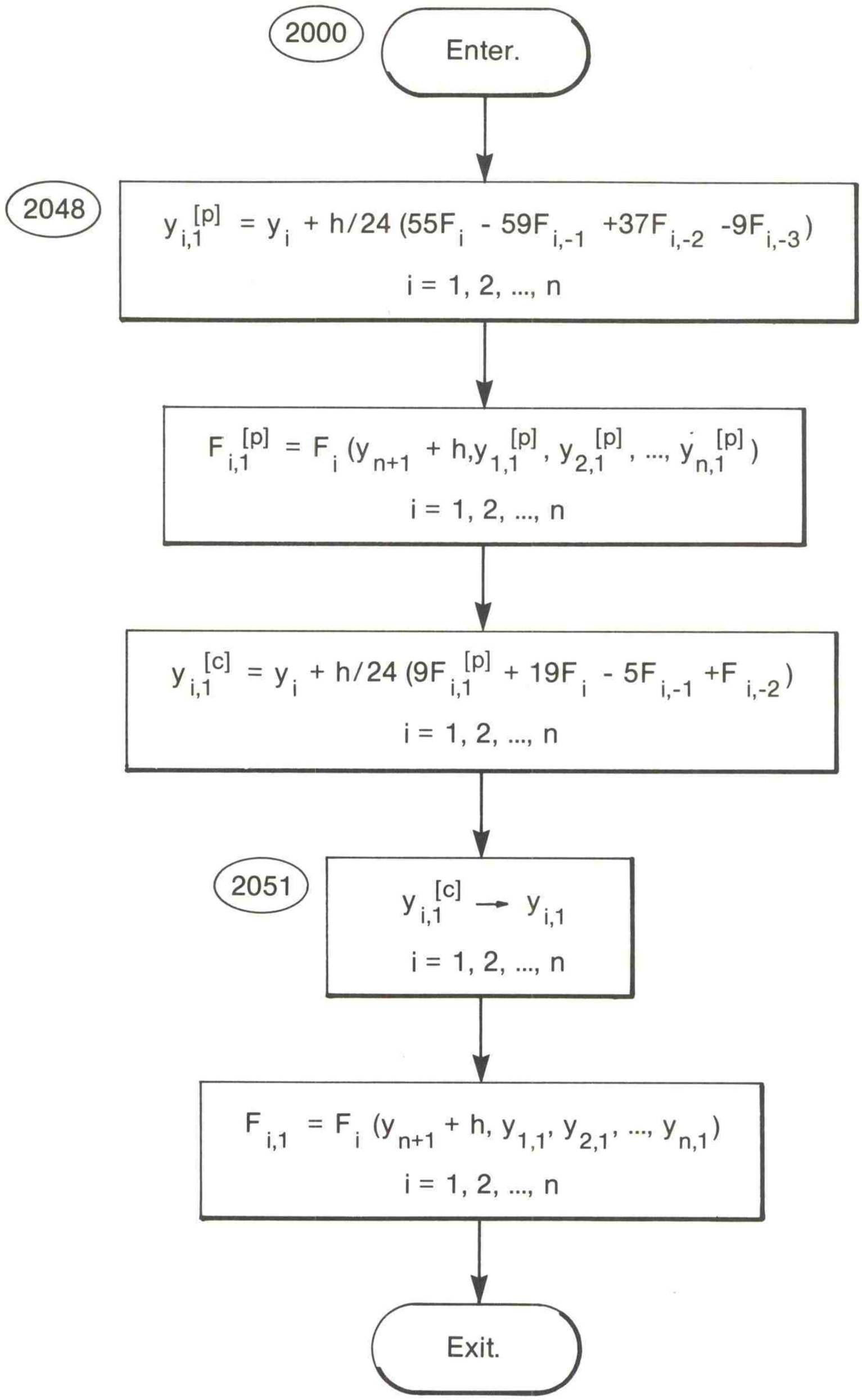
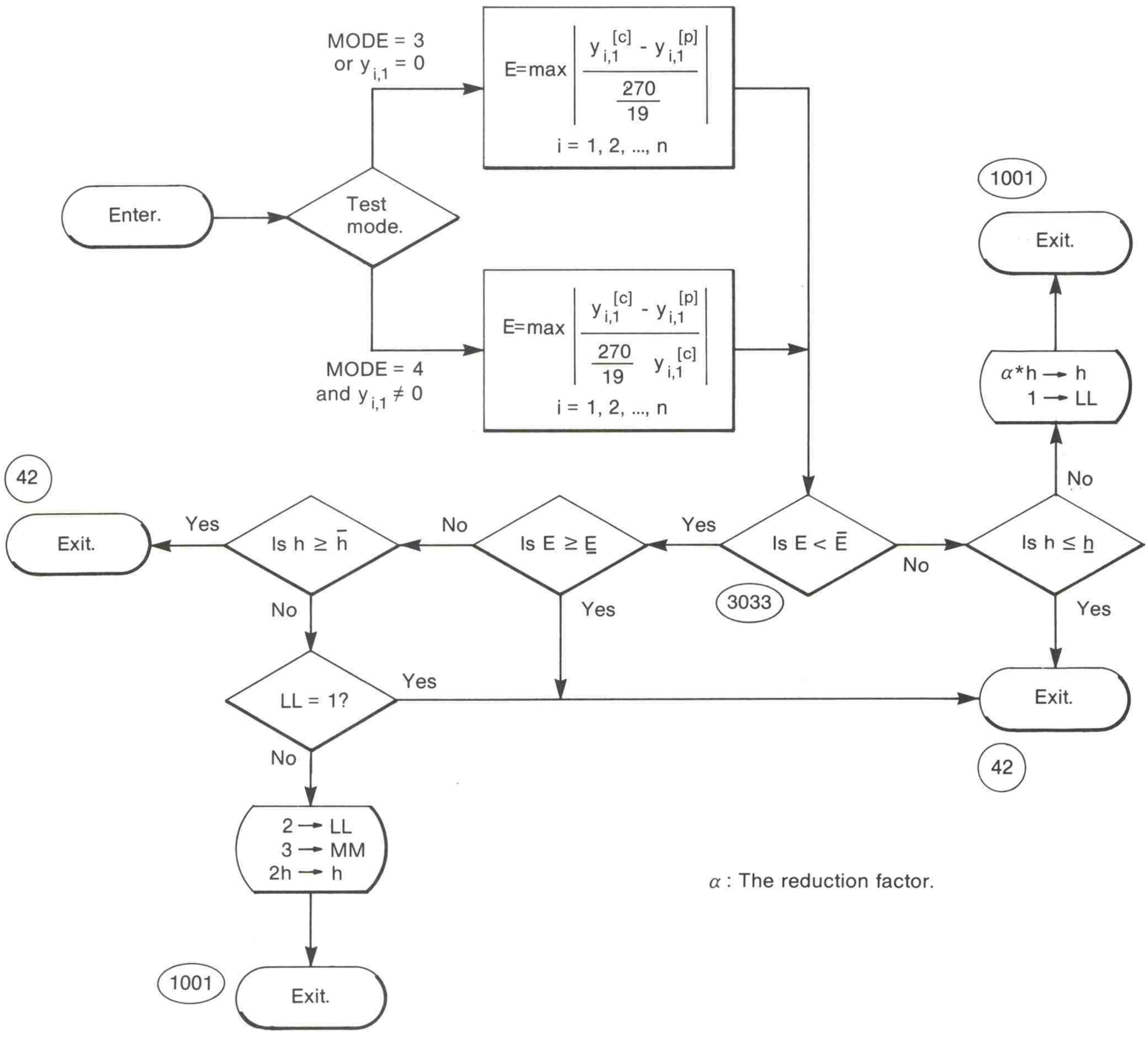


Figure 6.4. Flow chart for the Adams-Moulton integration method. The four starting values needed for this method are supplied by the Runge-Kutta method. Circled numbers refer to program statement numbers. [P] means predicted. [C] means corrected. n is the number of equations being integrated. $y_i \rightarrow y_n$ are the dependent variables. y_{n+1} is the independent variable. h is the step size. $F_{i,j}$ is the derivative of y_i j steps forward.



α : The reduction factor.

Figure 6.5. Flow chart for the single-step error analysis. The symbol E represents the single-step error, and $\bar{E}$, $\underline{E}$, h, and $\underline{h}$ represent the maximum and minimum acceptable single-step error and the maximum and minimum mesh size, respectively. Circled numbers refer to program statement numbers.

6.3 The Ocean Sound-Speed Models

Ocean sound-speed models can be specified either directly as $C(r,\theta,\phi,t)$, or they can call models of other ocean variables (such as temperature, salinity and density) upon which sound speed depends. In this version of HARPO, we provide only $C(r,\theta,\phi,t)$ background and perturbation models.

6.4 Ocean Dispersion Relations

HARPO gains versatility without sacrificing speed by having several versions of some of the subroutines. For example, the four versions of the dispersion-relation subroutine allow the user to decide in making up a run module (Sec. 3.3.2) whether to include or ignore currents and absorption. If there are no currents (or absorption) in the calculation, it is much cheaper to leave them out of the equations than it is to make the full calculations with zero current or zero absorption.

The input to the dispersion-relation subroutine is through blank common and common blocks /UU/, /CC/ and /LL/. Output is through common block /RIN/. The dispersion-relation subroutine is called through the entry point DISPER. The subroutine names are used only for user identification.

HARPO has four versions of the dispersion-relation subroutine.

(1) ANCNL - Acoustic waves, No Current, No Losses
(2) AWCNL - Acoustic waves, With Current, No Losses
(3) ANCWL - Acoustic waves, No Current, With Losses
(4) AWCWL - Acoustic Waves, With Current, With Losses

All of these versions calculate a Hamiltonian and its derivatives and the square of the wave number that satisfies the dispersion relation. All of the above variables and some others are in the common block /RIN/ (described in Table 7.17), which has all of the output from the dispersion-relation subroutines.

6.4.1 ANCNL - Acoustic, No Current, No Losses

The dispersion relation for pure acoustic waves is

$$k^2 = k_r^2 + k_\theta^2 + k_\phi^2 = \frac{\omega^2}{C^2} , \qquad (6.21)$$

where $C(t,r,\theta,\phi)$ is the speed of sound (provided by one of the sound-speed model subroutines).

At the beginning of the numerical integration, the magnitude of $\vec{k}$ is automatically set by the program so that the dispersion relation (6.21) is satisfied. During the numerical integration, the components of $\vec{k}$ are allowed to vary according to Hamilton's equations. Because of integration errors, there will be slight differences between k^2 and

$$k_{disp}^2 = \frac{\omega^2}{C^2} , \qquad (6.22)$$

the value it would have according to the dispersion relation. As a check on the accuracy of the numerical integration and on the consistency of the equations, the quantity

$$\text{ERROR} \equiv \frac{k^2}{k_{disp}^2} - 1 \qquad (6.23)$$

is printed at each step of the raypath calculation. It is possible, however, for ERROR to exceed somewhat the maximum allowable single-step integration error (W42) because k does not vary monotonically along the raypath. ERROR serves mainly as a check that the integration is proceeding correctly and that there are no errors in the formulas that compute derivatives in the model subroutines. If ERROR is generally smaller than W(42), then integrated quantities that vary monotonically (like travel time) will be computed with the accuracy given by W(42).

We use the following form of the dispersion relation for the Hamiltonian:

$$H(t,r,\theta,\phi,\omega,k_r,k_\theta,k_\phi) = \omega^2 - (k_r^2 + k_\theta^2 + k_\phi^2)\, C^2(t,r,\theta,\phi) \qquad (6.24)$$

The partial derivatives of the Hamiltonian are

$$\frac{\partial H}{\partial t} = - k^2 \frac{\partial C^2}{\partial t} \tag{6.25}$$

$$\frac{\partial H}{\partial r} = - k^2 \frac{\partial C^2}{\partial r} \tag{6.26}$$

$$\frac{\partial}{\partial \theta} = - k^2 \frac{\partial C^2}{\partial \theta} \tag{6.27}$$

$$\frac{\partial H}{\partial \phi} = - k^2 \frac{\partial C^2}{\partial \phi} \tag{6.28}$$

$$\frac{\partial H}{\partial \omega} = 2\omega \tag{6.29}$$

$$\frac{\partial H}{\partial k_r} = - 2 C^2 k_r \tag{6.30}$$

$$\frac{\partial H}{\partial k_\theta} = - 2 C^2 k_\theta \tag{6.31}$$

$$\frac{\partial H}{\partial k_\phi} = - 2 C^2 k_\phi \tag{6.32}$$

$$k \bullet \frac{\partial H}{\partial k} = -2 C^2 k^2 \quad . \tag{6.33}$$

6.4.2 AWCNL - Acoustic, With Current, No Losses

The dispersion relation for pure acoustic waves in terms of the sound speed in the presence of currents is

$$\Omega^2 - C^2 k^2 = 0 \ , \tag{6.34}$$

where

$$k^2 = k_r^2 + k_\theta^2 + k_\phi^2 \tag{6.35}$$

and

$$\Omega = \omega - \vec{k} \bullet \vec{V} = \omega - k_r V_r - k_\theta V_\theta - k_\phi V_\phi \tag{6.36}$$

is the intrinsic frequency of the wave (the frequency seen by an observer moving with the current). $\vec{V}(t,r,\theta,\phi)$ is the current velocity (provided by a current velocity model subroutine).

At the beginning of the numerical integration, the magnitude of k is set by the program so that the dispersion relation (6.34) is satisfied. During the numerical integration, the components of $\vec{k}$ are allowed to vary according

to Hamilton's equations. Because of integration errors, there will be slight differences between k^2 and

$$k^2_{disp} = \frac{\omega^2}{\left(C + \frac{\vec{k}\cdot\vec{V}}{k}\right)^2} \quad , \tag{6.37}$$

the value it would have according to the dispersion relation. Notice that

$$\frac{\vec{k}\cdot\vec{V}}{k} = \frac{k_r V_r + k_\theta V_\theta + k_\phi V_\phi}{\sqrt{k_r^2 + k_\theta^2 + k_\phi^2}} \tag{6.38}$$

is independent of the magnitude of $\vec{k}$, as it should be.

As a check on the accuracy of the numerical integration and on the consistency of the equations, the quantity in (6.23) is printed at each step of the raypath calculation.

We use the following form of the dispersion relation for the Hamiltonian:

$$H(t,r,\theta,\phi,\omega,k_r,k_\theta,k_\phi) =$$

$$\Omega^2(t,r,\theta,\phi,\omega,k_r,k_\theta,k_\phi) - (k_r^2+k_\theta^2+k_\phi^2)\, C^2(t,r,\theta,\phi), \tag{6.39}$$

where the intrinsic frequency Ω is given by (6.36)

The partial derivatives of the Hamiltonian are as follows.

$$\frac{\partial H}{\partial t} = 2\Omega \frac{\partial \Omega}{\partial t} - k^2 \frac{\partial C^2}{\partial t} \tag{6.40}$$

$$\frac{\partial H}{\partial r} = 2\Omega \frac{\partial \Omega}{\partial r} - k^2 \frac{\partial C^2}{\partial r} \tag{6.41}$$

$$\frac{\partial H}{\partial \theta} = 2\Omega \frac{\partial \Omega}{\partial \theta} - k^2 \frac{\partial C^2}{\partial \theta} \tag{6.42}$$

$$\frac{\partial H}{\partial \phi} = 2\Omega \frac{\partial \Omega}{\partial \phi} - k^2 \frac{\partial C^2}{\partial \phi} \tag{6.43}$$

$$\frac{\partial H}{\partial \omega} = 2\Omega \tag{6.44}$$

$$\frac{\partial H}{\partial k_r} = -\,2(\Omega V_r + C^2\, k_r) \tag{6.45}$$

$$\frac{\partial H}{\partial k_\theta} = -\,2(\Omega V_\theta + C^2\, k_\theta) \tag{6.46}$$

$$\frac{\partial H}{\partial k_\phi} = -\,2(\Omega V_\phi + C^2\, k_\phi) \tag{6.47}$$

$$k \cdot \frac{\partial H}{\partial k} = k_r \frac{\partial H}{\partial k_r} + k_\theta \frac{\partial H}{\partial k_\phi} + k_\phi \frac{\partial}{\partial k_\phi} = -\,2(\Omega \vec{k} \cdot \vec{V} + C^2\, k^2)\ , \tag{6.48}$$

where

$$\frac{\partial \Omega}{\partial t} = -\,k_r \frac{\partial V_r}{\partial t} - k_\theta \frac{\partial V_\theta}{\partial t} - k_\phi \frac{\partial V_\phi}{\partial t} \tag{6.49}$$

$$\frac{\partial \Omega}{\partial r} = -\,k_r \frac{\partial V_r}{\partial r} - k_\theta \frac{\partial V_\theta}{\partial r} - k_\phi \frac{\partial V_\phi}{\partial r} \tag{6.50}$$

$$\frac{\partial \Omega}{\partial \theta} = -\,k_r \frac{\partial V_r}{\partial \theta} - k_\theta \frac{\partial V_\theta}{\partial \theta} - k_\phi \frac{\partial V_\phi}{\partial \theta} \tag{6.51}$$

$$\frac{\partial \Omega}{\partial \phi} = -\,k_r \frac{\partial V_r}{\partial \phi} - k_\theta \frac{\partial V_\theta}{\partial \phi} - k_\phi \frac{\partial V_\phi}{\partial \phi}\ . \tag{6.52}$$

6.4.3 ANCWL - Acoustic, No Currents, With Losses and AWCWL - Acoustic, With Currents, With Losses

Part of the transmission loss for acoustic waves in the ocean is caused by absorption due to various physical and chemical relaxation processes (Urick, 1967). This absorption is often modeled with empirical formulas that give absorption as a function only of acoustic frequency. One such formula is given by Skretting and Leroy (1971) and Wakeley (1978):

$$\alpha = .006\ f^2 + \frac{.155(1.7f^2)}{(1.7)^2 + f^2}\ , \tag{6.53}$$

where α is the absorption coefficient in dB/km, and f is the acoustic frequency in kHz. Other treatments of absorption are reviewed by Jones (1983).

Subroutine SLLOSS computes α from (6.53) when called by DISPER, DISPER calculates k_{imag}^2 using (6.19), and HAMLTN provides an equation (6.18) which when integrated gives the absorption along the raypath. Some kind of absorption subroutine, such as SLLOSS, is required whenever you use ANCWL or AWCWL.

6.5 Reflecting Rays From the Bottom

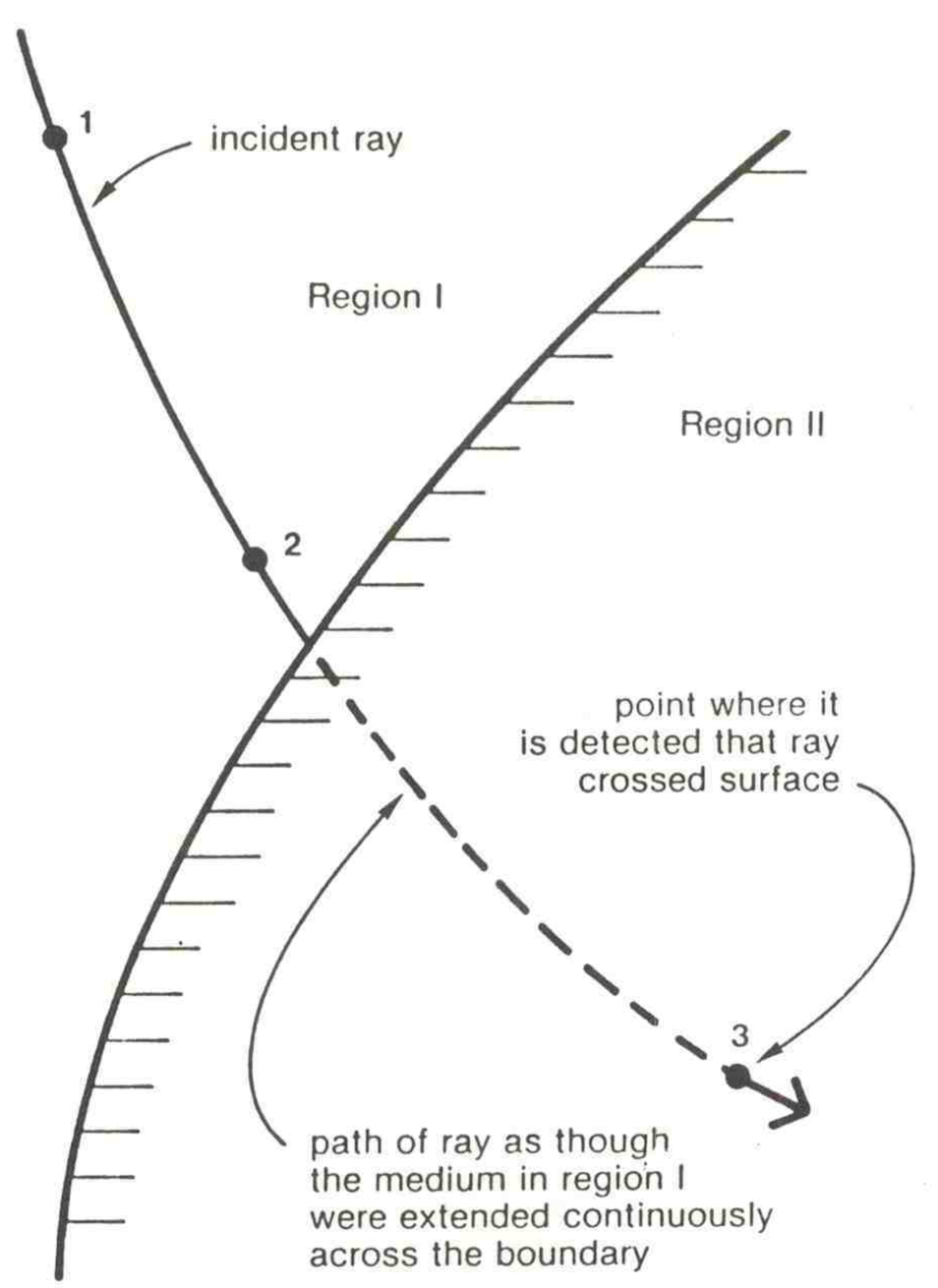
This section describes the method HARPO uses to find the intersections of a ray with any bottom surface and to compute the correct reflected ray, even in an anisotropic medium (an ocean with currents). A more detailed discussion of bottom-reflection algorithms is given in the report by Jones (1982). Essentially the same methods are used to detect receiver-surface crossings and to compute reflections from the ocean surface.

6.5.1 Detecting Ray Intersections With the Bottom

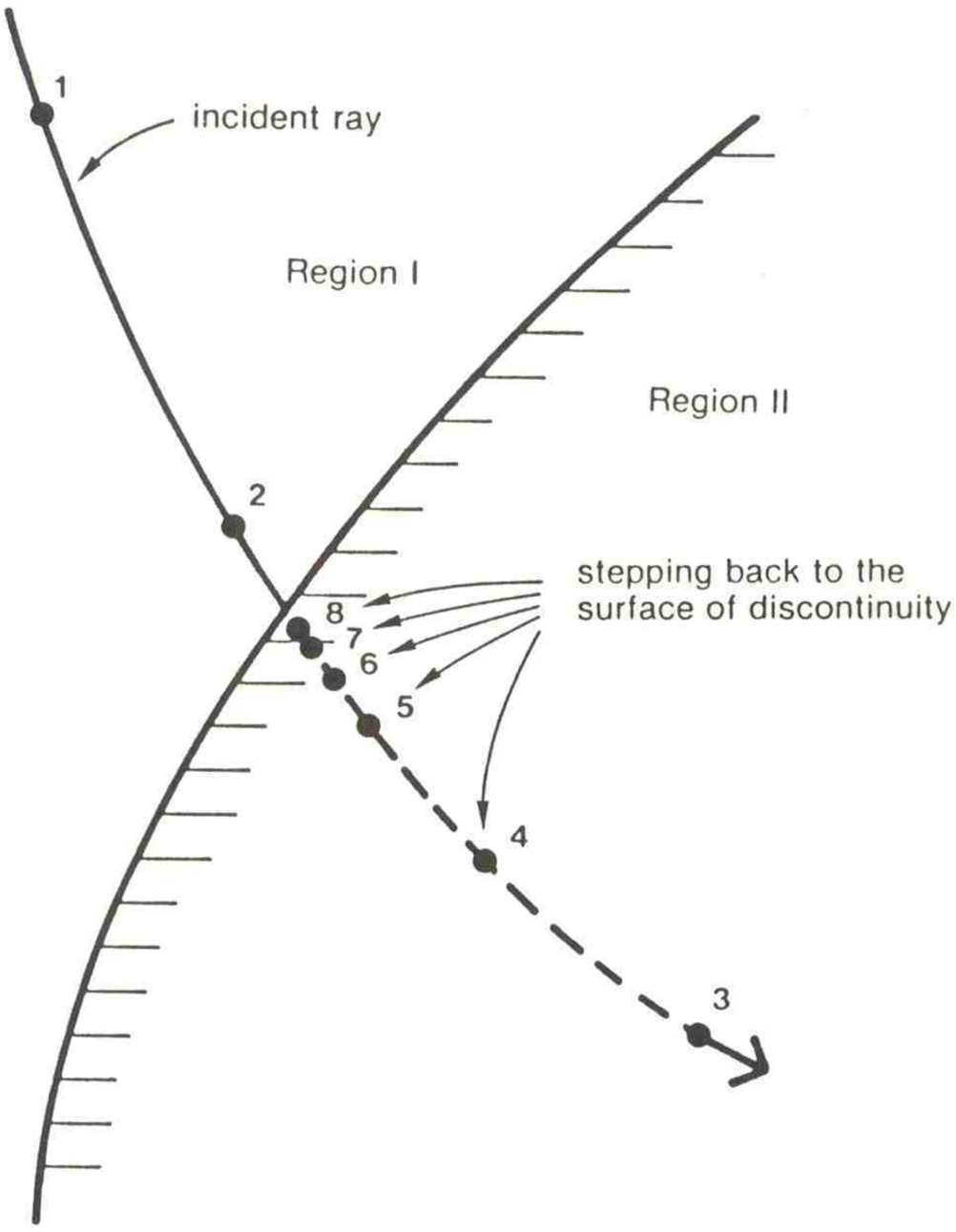
Detecting that the ray has crossed the bottom surface is straightforward for the simple case shown in Figure 6.6. Figure 6.7 shows a more difficult example in which the ray crosses the bottom surface twice. It is difficult to distinguish such a raypath from the example in Figure 6.8, knowing only the raypath coordinates and ray directions between the integration steps (shown by dots in Figures 6.7 and 6.8). A further difficulty is that iteration of the numerical integration in Figure 6.7 might lead to finding the wrong intersection with the surface of discontinuity, a problem shared by the example in Figure 6.9.

Although infrequent, the difficulties illustrated in Figures 6.7, 6.8 and 6.9 occur often enough that any useful algorithm must be able to handle them. These difficult cases obviously occur more often when large (relative to bottom structure) integration steps are used. This section derives algorithms that can handle these difficult cases in addition to the straightforward cases. Figure 6.10 illustrates cases that HARPO's intersection algorithm will not correctly handle unless step length is decreased or the bottom model is made smoother.

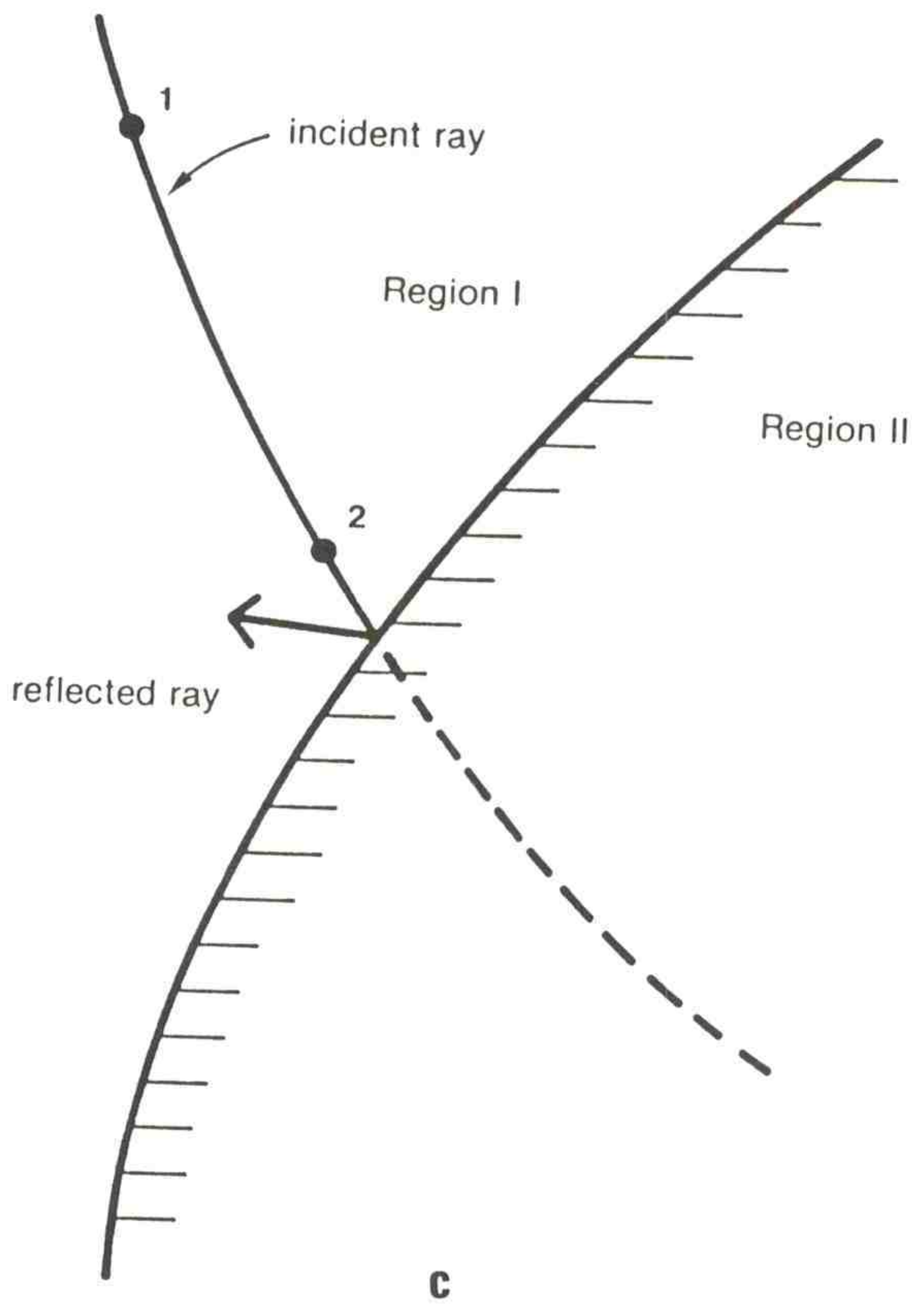
HARPO accepts only bottom models whose surface is continuous and whose slope and curvature are continuous. Thus, wedge-shaped surfaces, for example, are not allowed. Not only are diffraction effects important at such edges, but



a



b



c

Figure 6.6. Three steps in calculating the intersection and reflection of a ray at a surface: (a) recognition that the ray has crossed the surface (numbers indicate successive positions of the ray after each integration step; the raypath is curved because the ocean is inhomogeneous); (b) iteration by numerical integration to find the intersection of the ray with the surface; and (c) computation of the reflected ray ready to start numerical integration in a new direction in the ocean. The same algorithm could be used to compute ray refraction at discontinuities of refractive index.

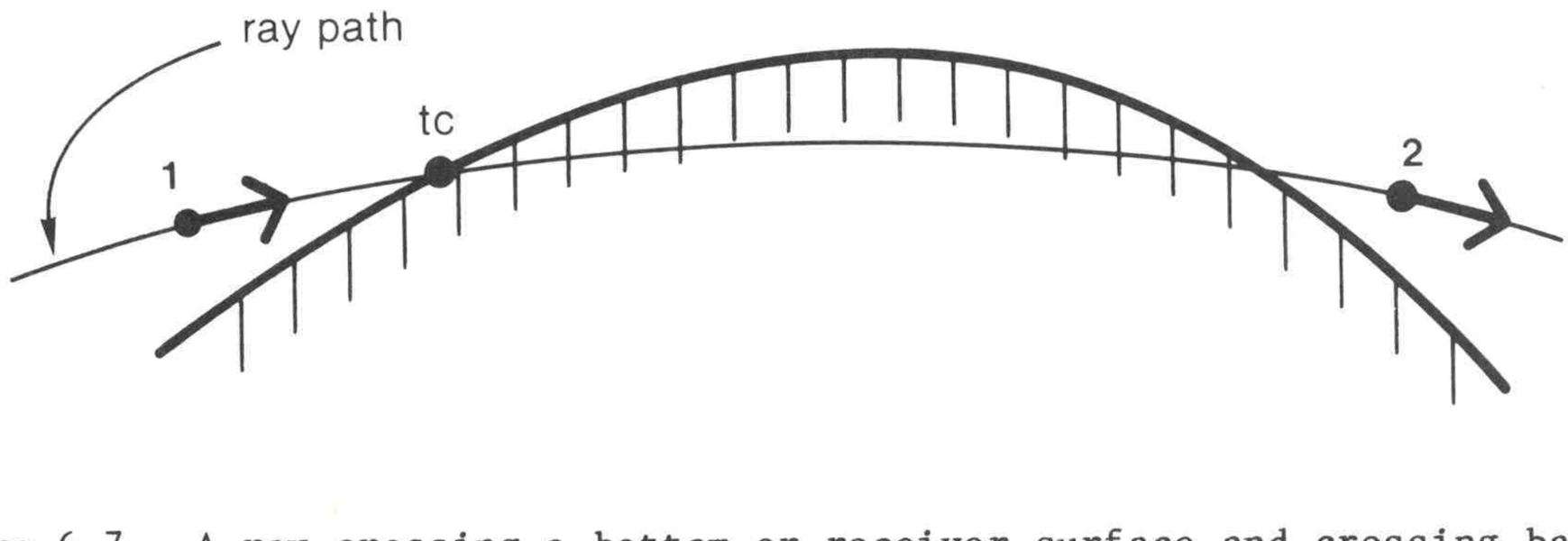


Figure 6.7. A ray crossing a bottom or receiver surface and crossing back again between integration steps. An algorithm that checked only to see if the ray at point 2 was in a different medium than point 1 would miss the intersections.

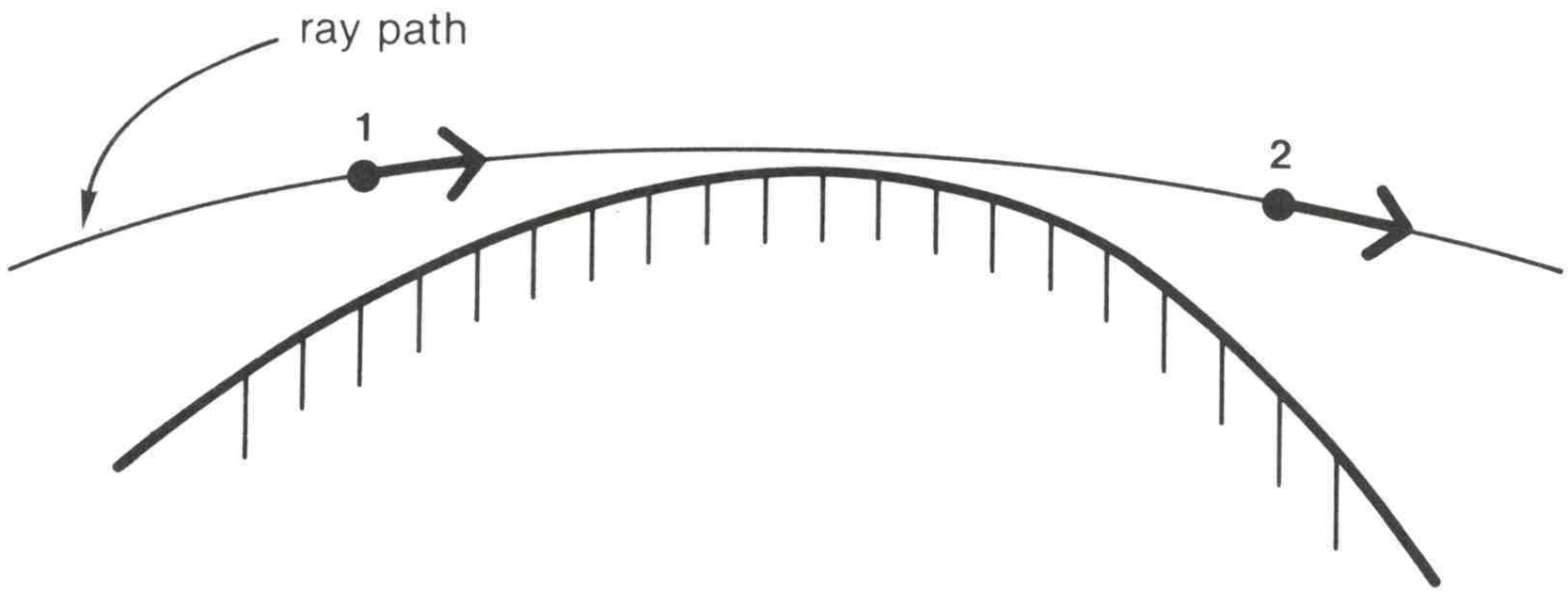


Figure 6.8. A ray that nearly intersects a bottom or receiver surface. A useful intersection algorithm must be able to distinguish this case from the one depicted in Figure 6.7.

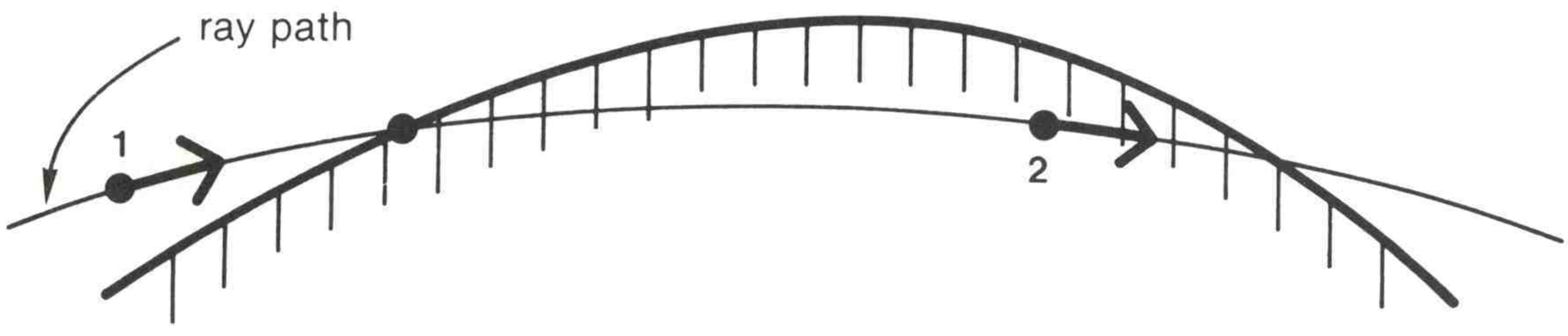


Figure 6.9. A ray crossing a bottom or receiver surface and ending closer to a second intersection. A useful algorithm must step backward and find the first intersection. HARPO will correctly handle the three cases on this page.

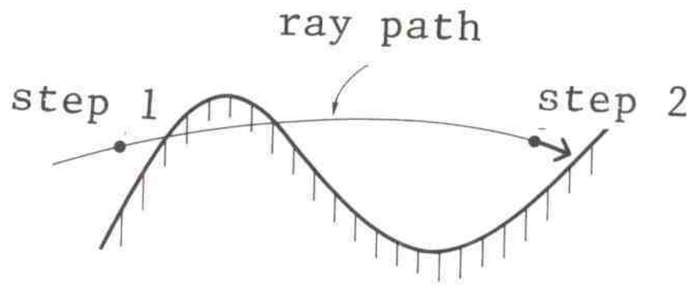


Figure 6.10. Examples of bottom (or receiver-surface) encounters that HARPO's intersection algorithm will not correctly handle. Step length must be decreased, or the bottom model must be smoothed.

some of the algorithms used by HARPO may not work properly for surfaces with edges. In addition, the algorithms used by HARPO may not always properly handle surfaces that contain caves or tunnels.

At each integration step in the raypath calculation, we assume that the following information is available:

(1) The position of the ray point (r, θ, ϕ) in spherical polar (Earth-centered) coordinates,

(2) The local Cartesian components (k_r, k_θ, k_ϕ) of the wave vector (a vector pointing in the wave-normal direction and normalized so that the magnitude equals the wave number),

(3) The accumulated group time delay, t, of the ray, (which is also the independent variable for the numerical integration), and

(4) The derivatives

$$\dot{r} \equiv dr/dt \tag{6.54}$$

$$\dot{\theta} \equiv d\theta/dt \tag{6.55}$$

$$\dot{\phi} \equiv d\phi/dt \tag{6.56}$$

$$\dot{k}_r \equiv dk_r/dt \tag{6.57}$$

$$\dot{k}_\theta \equiv dk_\theta/dt \tag{6.58}$$

$$\dot{k}_\phi \equiv dk_\phi/dt \ . \tag{6.59}$$

We assume also that the values of any of these variables can be saved from one step to the next for comparison. In particular, we make the following approximations,

$$\ddot{r} \approx (\dot{r}(t_2) - \dot{r}(t_1))/t_2 - t_1) \tag{6.60}$$

$$\ddot{\theta} \approx (\dot{\theta}(t_2) - \dot{\theta}(t_1))/(t_2 - t_1) \tag{6.61}$$

$$\ddot{\phi} \approx (\dot{\phi}(t_2) - \dot{\phi}(t_1))/(t_2 - t_1) \ , \tag{6.62}$$

which allow us to estimate the curvature of ray. In using these algorithms, it would probably be useful to incorporate a test of the validity of the above approximations by evaluating both sides of (6.60) - (6.62), but HARPO does not do that.

The simplest method for predicting whether an extension of the raypath from a point will intersect the bottom surface (or the receiver surface) is to extend the ray in a straight line and see if it meets the bottom when the surface is extended as a plane by using the local slope. However, whenever the curvature of the ray and the bottom surface are small enough for that approximation to be useful, the ray would probably eventually go below the bottom on one of the integration steps, and the prediction from the simplest algorithm would not be needed.

Because we want an algorithm sophisticated enough to estimate the time of nearest intersection (if one occurs) in cases like those in Figures 6.7 or 6.8, we should include at least the local curvature in any approximations. Also, because it would be difficult to deal with a higher order approximation, a quadratic approximation to both the ray and the bottom seems to be the best compromise.

In addition, when searching for an intersection with the bottom (or receiver) surface, the intersection must be in the correct direction. In the case of the bottom, the intersection will always be into the bottom. In the case of the receiver surface, the correct direction of crossing will alternate from one crossing to the next, but for each crossing, the direction will be known. For the purposes of the present development, it is useful to define a parameter S that is equal to +1 if the wanted crossing is upward and -1 if the wanted crossing is downward. Thus, the value of S will always be -1 for a bottom crossing.

6.5.2 The Surface Model

HARPO expresses an arbitrary model of a bottom, an ocean surface or a receiver surface in the form

$$f(r, \theta, \phi) = 0 \ . \tag{6.63}$$

Because f is zero only on the surface, it always has one sign on one side of the surface and the opposite sign on the other side. Let us call the side of the bottom surface that is underground the inside and the other side the outside. Then we can arbitrarily require that f be positive outside of the surface and negative inside the surface. We can similarly designate an inside and an outside for the receiver and ocean surfaces and make the same requirement on f.

Thus, we have

$$f(r, \theta, \phi) > 0 \tag{6.64}$$

outside the surface and

$$f(r, \theta, \phi) < 0 \tag{6.65}$$

inside the surface. The time derivative of f (which indicates the rate that the ray is moving away from the surface) is

$$\dot{f} = f_r \dot{r} + f_\theta \dot{\theta} + f_\phi \dot{\phi} \tag{6.66}$$

where a subscript indicates a partial derivative with respect to the subscript. The time derivative of (6.66) is

$$\ddot{f} = f_r \ddot{r} + f_\theta \ddot{\theta} + f_\phi \ddot{\phi} + f_{rr} \dot{r}^2 + f_{\theta\theta} \dot{\theta}^2 + f_{\phi\phi} \dot{\phi}^2 + 2f_{r\theta} \dot{r}\dot{\theta} + 2f_{r\phi} \dot{r}\dot{\phi} + 2f_{\theta\phi} \dot{\theta}\dot{\phi}. \tag{6.67}$$

Assuming that (6.67) is nearly constant locally, we have

$$f = f_1 + \dot{f}_1(t - t_1) + \tfrac{1}{2} \ddot{f}_1(t - t_1)^2 , \tag{6.68}$$

where the subscript 1 refers to the values at the time t_1. We want to find the value of t for which the ray intersects the surface, that is, where f = 0. Let t_c be the value for which f = 0. Then

$$0 = f_1 + \dot{f}_1(t_c - t_1) + \tfrac{1}{2} \ddot{f}_1(t_c - t_1)^2 . \tag{6.69}$$

For simplicity, let us drop the subscript 1 in (6.69), but remember that f and t without subscripts refer to the integration step on the raypath where we are trying to estimate the time t_c where the ray will intersect the surface. Then (6.69) becomes

$$0 = f + \dot{f}(t_c - t) + \tfrac{1}{2} \ddot{f}(t_c - t)^2 . \tag{6.70}$$

The solution of (6.70), for which the ray is crossing from inside to outside when S = +1 and crossing from outside to inside when S = -1, is

$$t_c - t = \frac{-\dot{f} + S\sqrt{\dot{f}^2 - 2f\ddot{f}}}{\ddot{f}} . \tag{6.71}$$

Within the approximation made here, the ray will intersect the surface if

$$\dot{f}^2 - 2f\ddot{f} > 0 \tag{6.72}$$

but will make a closest approach to the surface if

$$\dot{f}^2 - 2f\ddot{f} < 0 . \tag{6.73}$$

In the latter case, we can estimate the time t_p at which f is a minimum.

$$t_p - t = -\frac{\dot{f}}{\ddot{f}} . \tag{6.74}$$

This is close to the time when the ray makes a closest approach to the surface. If the second derivative in (6.67) is very small, then the formula in (6.71) may be impractical. In that case, the solution

$$t_c - t = \frac{-2f}{\dot{f} + S\sqrt{\dot{f}^2 - 2f\ddot{f}}} \tag{6.75}$$

is more useful. The advantage of (6.75) is that it is uniformly valid as $\ddot{f}$ approaches zero.

In HARPO, different formulas are used in different circumstances: statement 501 in TRACE uses (6.75) to estimate t_c. Subroutine BACKUP uses

$$t_c = t - f/\dot{f} \tag{6.76}$$

when it is stepping to an intersection. Subroutine BACKUP uses (6.74) to step to a closest approach. When BACKUP has tried to step to a closest approach and fails after 10 tries, it tries to find an intersection. In that case, it uses (6.71) to give a first estimate for t_c, then uses (6.76) for iterating. Statement 30 in TRACE uses (6.74) to estimate the time of closest approach. See Section 7.3 for flow charts of these algorithms.

6.5.3 Reflecting the Ray From a Surface

Once the intersection with the ocean surface or bottom has been found, the ray must be properly reflected. For an isotropic medium, this is straightforward. The algorithm must first project the wave vector into the two components parallel to the surface and the component perpendicular to the surface. It then changes sign on the component perpendicular to the surface.

An anisotropic medium (such as an ocean with currents) is more difficult. The two components parallel to the surface remain unchanged, as before, but the component perpendicular to the surface must be changed so that the dispersion relation is satisfied. Although this principle is the same for all media, the solution depends on the dispersion relation. At this point, we must specialize

to the particular medium of interest, namely, acoustic waves in the presence of currents.

We need to first separate the wave vector $\vec{k}$ into components perpendicular and parallel to the surface. Let $\vec{n}$ be a unit vector pointing out of the surface. Then the component of $\vec{k}$ normal to the surface is

$$k_\perp = \vec{k}\cdot\vec{n} \tag{6.77}$$

$$\vec{k}_\perp = (\vec{k}\cdot\vec{n})\ \vec{n} \tag{6.78}$$

and the part parallel to the surface is

$$\vec{k}_\parallel = \vec{k} - (\vec{k}\cdot\vec{n})\ \vec{n}\ . \tag{6.79}$$

The dispersion relation for acoustic waves in the presence of currents is

$$-\Omega^2 + C^2\ k^2 = 0\ , \tag{6.80}$$

where $$\Omega \equiv \omega - \vec{k}\cdot\vec{v} \tag{6.81}$$

is the intrinsic frequency, ω is the wave frequency, and $\vec{v}$ is the current velocity.

With the help of (6.81), we can separate (6.80) as follows:

$$-(\omega - \vec{k}_\perp\cdot\vec{v} - \vec{k}_\parallel\cdot\vec{v})^2 + C^2\ k_\perp^{\,2} + C^2\ k_\parallel^{\,2} = 0\ . \tag{6.82}$$

We want to solve (6.82) for $k_\perp$, assuming $\vec{k}_\parallel$ to be known. We can rewrite (6.80) as

$$(C^2 - v_\perp^{\,2})\ k_\perp^{\,2} + 2\ \Omega_\parallel\ v_\perp\ k_\perp - \Omega_\parallel^{\,2} + C^2\ k_\parallel^{\,2} = 0\ , \tag{6.83}$$

where

$$\Omega_\parallel \equiv \omega - \vec{k}_\parallel\cdot\vec{v} \tag{6.84}$$

and

$$\vec{v}_\perp \equiv (\vec{v}\cdot\vec{n})\ \vec{n} \quad , \quad v_\perp \equiv \vec{v}\cdot\vec{n} \tag{6.85}$$

is the component of current normal to the surface. The quadratic formula gives

the solution to (6.83) as

$$k_\perp = \frac{-\Omega_\parallel v_\perp \pm c\sqrt{\Omega_\parallel^2 - k_\parallel^2 (c^2 - v_\perp^2)}}{c^2 - v_\perp^2} \quad . \tag{6.86}$$

One solution of (6.86) should be the normal component of $\vec{k}$ for the incident wave, the other the normal component of $\vec{k}$ for the reflected wave. To convert from the incident wave to the reflected wave, it is necessary simply to change the sign of

$$k_\perp + \frac{\Omega_\parallel v_\perp}{c^2 - v_\perp^2} \quad . \tag{6.87}$$

To do this, we can use (6.78) and (6.79) to write

$$\vec{k} = \vec{k} - (\vec{k}\cdot\vec{n})\vec{n} + (\vec{k}\cdot\vec{n})\vec{n} \quad . \tag{6.88}$$

This is equivalent to

$$\vec{k} = \vec{k} - (\vec{k}\cdot\vec{n})\vec{n} - \frac{\Omega_\parallel v_\perp}{c^2 - v_\perp^2}\vec{n} + (\vec{k}\cdot\vec{n})\vec{n} + \frac{\Omega_\parallel v_\perp \vec{n}}{c^2 - v_\perp^2} \quad . \tag{6.89}$$

Let us assume that (6.89) applies to the incident wave, that is,

$$\vec{k}_{inc} = \vec{k}_{inc} - (\vec{k}_{inc}\cdot\vec{n})\vec{n} - \frac{\Omega_\parallel v_\perp}{c^2 - v_\perp^2}\vec{n} + (\vec{k}_{inc}\cdot\vec{n})\vec{n} + \frac{\Omega_\parallel v_\perp \vec{n}}{c^2 - v_\perp^2} \quad , \tag{6.90}$$

where the subscript inc signifies the incident wave. To get the wave vector for the reflected wave, we need only reverse the sign of the last two terms in (6.90), that is,

$$\vec{k}_{ref} = \vec{k}_{inc} - 2(\vec{k}_{inc}\cdot\vec{n})\vec{n} - 2\frac{\Omega_\parallel v_\perp \vec{n}}{c^2 - v_\perp^2} \quad , \tag{6.91}$$

where the subscript ref signifies the reflected wave.

To be more explicit, we can write (6.91) in terms of components. We assume an earth-centered spherical polar-coordinate system (r, θ, ϕ). We then

consider Cartesian components of $\vec{k}$, (k_r, k_θ, k_ϕ) in the r, θ, and ϕ directions. We also consider components of n, (n_r, n_θ, n_ϕ) in the r, θ, and ϕ directions. Then (6.91) is equivalent to

$$k_{r\ ref} = k_{r\ inc} - 2(\vec{k}_{inc} \cdot \vec{n}) n_r - \frac{2\Omega_\parallel v_\perp}{c^2 - v_\perp^2} n_r \qquad (6.92a)$$

$$k_{\theta\ ref} = k_{\theta\ inc} - 2(\vec{k}_{inc} \cdot \vec{n}) n_\theta - \frac{2\Omega_\parallel v_\perp}{c^2 - v_\perp^2} n_\theta \qquad (6.92b)$$

$$k_{\phi\ ref} = k_{\phi\ inc} - 2(\vec{k}_{inc} \cdot \vec{n}) n_\phi - \frac{2\Omega_\parallel v_\perp}{c^2 - v_\perp^2} n_\phi \ . \qquad (6.92c)$$

One might wonder whether it is realistic to allow a component of the current normal (or even parallel) to the ocean surface or bottom. Rather than debate this point here, we simply point out that HARPO does not check any models to see if they satisfy continuity, geostrophy, or physical boundary conditions (in fact, many of our models don't). This is the responsibility of those who design models, if such conditions are important in their applications. The last term in (6.92a,b,c) guarantees that the dispersion relation will be satisfied for the reflected wave, even if the current does not vanish at the surface.

6.5.4 Unit-Normal Directions From the Surface

The previous section requires unit normal directions to the surface. Because f is a constant along the surface, the gradient of f is in the same direction as the unit normal. That is

$$\vec{\nabla} f = f_r \hat{i}_r + \frac{f_\theta}{r} \hat{i}_\theta + \frac{f_\phi}{r \sin \theta} \hat{i}_\theta \propto \vec{n} \ . \qquad (6.93)$$

Taking the ratio of components gives

$$n_\theta = \frac{f_\theta}{r\ f_r} n_r \qquad (6.94)$$

and

$$n_\phi = \frac{f_\phi}{f_r \, r \sin\theta} \, n_r \, . \tag{6.95}$$

The solution of (6.94), and (6.95), while requiring $n_r^2 + n_\theta^2 + n_\phi^2 = 1$, is

$$n_r = \frac{f_r}{\sqrt{f_r^2 + \frac{f_\theta^2}{r^2} + \frac{f_\phi^2}{r^2 \sin^2\theta}}} \, , \tag{6.96}$$

$$n_\theta = \frac{f_\theta}{r\sqrt{f_r^2 + \frac{f_\theta^2}{r^2} + \frac{f_\phi^2}{r^2 \sin^2\theta}}} \, , \tag{6.97}$$

$$n_\phi = \frac{f_\phi}{r \sin\theta \sqrt{f_r^2 + \frac{f_\theta^2}{r^2} + \frac{f_\phi^2}{r^2 \sin^2\theta}}} \, . \tag{6.98}$$

6.6 Coordinate Systems

HARPO uses two different earth-centered spherical polar-coordinate systems, one geographic and one computational, because it is easier to express some models in a different coordinate system. Input data for the coordinates of the transmitter, W(4) and W(5), and input data for the coordinates of the north pole of the computational coordinate system, W(24) and W(25), are entered in geographic coordinates. Setting W(25) equal to 0° and W(24) equal to 90° superimposes the two north poles and equates the two coordinate systems.

When the two coordinate systems do not coincide, the ocean models calculate current, sound speed, bottom surface, upper surface, and receiver surface in terms of the computational coordinate system. Dudziak (1961) describes the transformations between these coordinate systems.

For some long-range ray calculations in the ocean, a spherical model of the earth may not be accurate enough. Some applications may require ocean models to be expressed in geodetic (e.g., spheroidal) coordinates, which would

be transformed to spherical coordinates for ray tracing. However, for paths of a few thousand kilometers and less, we would recommend using a spherical computational coordinate system with W(1) set to the local radius of curvature of the geoid in the propagation direction.

7. Structure of the Program

This chapter explains how the parts of HARPO work together, including a brief description of each program subroutine and detailed flow charts of the central ray-tracing parts. Also included are hierarchical or organization diagrams that show the calling sequences of the principal ray-tracing operations and details of the common-block structure and usage.

7.1 Description of the Subroutines

Following is a list of all the programs, subroutines, and functions that comprise HARPO, that is, the functions inside the dashed lines of Figure 1.1. They are listed in the order they appear in Files 3, 4, and 5 of the distribution tape and in the source-code listings of Appendix D. The routines are divided into a RAY-TRACING "CORE," the set of programs that is always required to do ray tracing, DISPERSION-RELATION ROUTINES, from which you must select one, and OCEAN MODEL SUBROUTINES, from which you select the routines that correspond to the ocean models you want to use. A more detailed description of the distribution-tape structure is given in Sec. 3.3.1.

7.1.1 Ray-Tracing Core (Tape File 3)

PROGRAM RAYTRC -- The main program; sets the initial conditions for each ray and calls TRACE.

SUBROUTINE DFSYS -- Contains system-dependent functions such as date, time (user must modify).

SUBROUTINE DFCNST -- Defines machine-dependent constants (user must modify).

SUBROUTINE READW1 -- Reads variable-length tabular data from input data table.

SUBROUTINE GTUNIT -- Interprets units line in tabular input data.

SUBROUTINE SREAD1 -- Handles unassigned labeled common blocks.

FUNCTION READW -- Reads input data table into W array, converting units.

SUBROUTINE CLEAR -- Sets n elements of an array to zero.

FUNCTION ND2B -- Converts decimal digits to positionally equivalent binary numbers when reading W(29).

FUNCTION UCON -- Provides keyword units conversion for input data.

SUBROUTINE TRACE -- Calculates a raypath for the requested number of hops.

FUNCTION PCROSS -- Tests whether ray crosses a surface.

SUBROUTINE RCROSS -- Estimates point where ray crosses a surface.

SUBROUTINE HAMLTN -- Calculates Hamilton's equations for ray tracing and other quantities to be integrated.

SUBROUTINE RKAM -- Keeps track of integration steps performed by RKAM1 and makes them available to calling subroutines.

SUBROUTINE RKAM1 -- Numerically integrates Hamilton's equations.

SUBROUTINE BACKUP -- Moves the ray point to the desired intersection with a surface.

FUNCTION REFLECT -- Computes normal and parallel components of k vector at reflections from a surface.

SUBROUTINE FIT -- Computes 3 types of parabolic fit to raypath relative to a surface.

FUNCTION GET -- Gets the value of a surface function and its derivatives; calls the surface model subroutine if necessary.

FUNCTION GET1 -- Second version of GET to avoid self-calls.

FUNCTION ITEST -- Passes integer values through for variables typed real.

FUNCTION ITOC -- Integer to character conversion.

SUBROUTINE CONBLK -- Data-initialization and file-opening service routine.

SUBROUTINE SETTRC -- Initializes /TRAC/ labeled common block.

FUNCTION WCHANGE -- Determines whether two W arrays are the same (needed for producing raysets).

FUNCTION RENORM -- Normalizes a vector to a specified magnitude.

SUBROUTINE SET2 -- Sets n components of vector to a specified single value.

SUBROUTINE PRINTR -- Prints details of the raypath calculation at specified intervals and produces computer-readable output (raysets).

SUBROUTINE OCNHD -- Prints page headings.

SUBROUTINE PUTDES -- Prints model information on printout header.

FUNCTION NUMSTG -- Converts a numeric value to a string.

SUBROUTINE SFILL -- Fills a string with n specified characters.

FUNCTION STRIM -- Determines position of last nonblank character of a string.

FUNCTION RERR -- Computes for subroutine PRINTR the largest relative integration error.

SUBROUTINE RERROR -- Reports error conditions and stops program.

SUBROUTINE STOPIT -- Prints error condition and stops program.

SUBROUTINE PUTKST -- Multiple ENTRY points to produce line-printer output while accounting for line count, new page, etc.

SUBROUTINE PUTKCT -- Put out a centered line and count.

SUBROUTINE OPNREP -- Either opens a new disk file or replaces an ole one, as necessary.

SUBROUTINE ZAPFIL -- Test for existence of file and delete.

SUBROUTINE OVERRD -- Tests for "zero-override" condition in input data (Sec. 5.3.1).

SUBROUTINE SFILTR -- Filters extraneous characters from plot labels.

FUNCTION ALCOSH -- Compute log(cosh(x)) and uses large-argument approximation.

SUBROUTINE GAUSEL -- Calculates coefficients of cubic functions to fit points in CTABLE.

[PLOTTING ROUTINES]

SUBROUTINE RAYPLT -- Main plotting program; initializes, reads input, plots projections of rays on a vertical or horizontal plane.

SUBROUTINE PLOT -- XY plotting routine, called by RAYPLT.

SUBROUTINE LABPLT -- Labels rayplots.

[TICK/ANNOTATION ROUTINES]

SUBROUTINE PLTHLB -- Plots horizontal ticks and annotation for rayplot.

SUBROUTINE PLTANH -- Generic horizontal tick annotation.

SUBROUTINE SETXY -- Plot initialization; sets projection parameters.

SUBROUTINE TIKLINE -- Draws straight line with ticks at intervals.

SUBROUTINE PLTANOT -- Puts general annotations on plots.

SUBROUTINE DRAWTKS -- Draws plot boundary, ticks, and labels for horizontal ray projection.

SUBROUTINE PLTLB -- Puts vertical tick annotations on rayplots.

SUBROUTINE ARCTIC -- Draws curved range axis for rayplot.

[GRAPHICS WRITE ROUTINES]

SUBROUTINE DDINIT -- Initializes plotting process (writes header line to TAPE5).

SUBROUTINE DDBP -- Sets a vector origin (writes IX,IY to TAPE5).

SUBROUTINE DDVC -- Plots a vector (writes IX,IY for vector end point to TAPE5).

SUBROUTINE DDTEXT -- Writes an array (character string) to TAPE 5 in tabular text mode.

SUBROUTINE DDTAB -- Sends instruction to TAPE5 that initializes tabular (text) plotting.

SUBROUTINE DDFR -- Sends instruction to TAPE5 to advance a microfilm frame.

SUBROUTINE DDEND -- Empties plot buffer and releases plotting command file to microfilm plot queue.

SUBROUTINE DASH -- Sets dashed-line mode; that is, all plotted curves will be dashed instead of solid after a call to subroutine DASH.

SUBROUTINE RESET('DASH') -- Sets solid-line mode; that is, all plotted curves will be solid lines after this call.

SUBROUTINE HEIGHT -- If you do not have the DISSPLA* plotting package, load SUBROUTINE SMPANN instead of SUBROUTINE FULANN, and you can ignore this routine.

SUBROUTINE MX1ALF -- If you do not have the DISSPLA* plotting package, load SUBROUTINE SMPANN instad of SUBROUTINE FULANN, and you can ignore this routine.

SUBROUTINE MX2ALF -- If you do not have the DISSPLA* plotting package, load SUBROUTINE SMPANN instead of SUBROUTINE FULANN, and you can ignore this routine.

SUBROUTINE SCMPLX -- If you do not have the DISSPLA* plotting package, load SUBROUTINE SMPANN instead of SUBROUTINE FULANN, and you can ignore this routine.

7.1.2 Dispersion Relation Routines (Tape File 4)

SUBROUTINE ANCNL -- Acoustic wave, no current, no losses.

SUBROUTINE AWCNL -- Acoustic wave, with current, no losses.

---

* DISSPLA is the proprietary product of ISSCO, Inc.

SUBROUTINE ANCWL -- Acoustic wave, no current, with losses.

SUBROUTINE AWCWL -- Acoustic wave, with current, with losses.

7.1.3 Ocean Model Subroutines (Tape File 5)

SUBROUTINE WLINEAR -- Constant upward or northward background current; linear eastward current profile.

SUBROUTINE VVORTX3 -- Vertical vortex current model with viscous core and Gaussian height profile.

SUBROUTINE WGAUSS2 -- Eastward current model that decays in three dimensions (jet).

SUBROUTINE NPCURR -- Do-nothing current perturbation model.

SUBROUTINE CTANH -- Background sound-speed profile with linear segments joined smoothly.

SUBROUTINE CSTANH -- Background sound-speed-squared profile with linear segments joined smoothly.

SUBROUTINE CSSPOKE -- Background sound speed as a function of angle from horizontal.

SUBROUTINE CSSPOK2 -- Background sound speed as a function of angle from horizontal.

SUBROUTINE CSMUNK1 -- Canonical sound channel; linear parameters in longitude.

SUBROUTINE CSMUNK2 -- Canonical sound channel; linear C with longitude.

SUBROUTINE CTABLE -- Tabular C profile in cubic segments.

SUBROUTINE NPSPEED -- Do-nothing sound-speed-perturbation model.

SUBROUTINE CBLOB2 -- Sound-speed perturbation with Gaussian decay in three dimensions.

SUBROUTINE CBLOB3 -- Up to 3 blobs of C increase that decay in three dimensions.

SUBROUTINE CBLOB4 -- Up to 3 blobs of C^2 increase that decay in three dimensions.

[ABSORPTION MODELS]

SUBROUTINE SLLOSS -- Skretting-Leroy formula for acoustic absorption.

SUBROUTINE NPABSR -- Do-nothing absorption-perturbation model.

[BOTTOM, OCEAN-SURFACE, AND RECEIVER-SURFACE MODELS]

SUBROUTINE GHORIZ -- Bottom model = a sphere at fixed depth below MSL (mean sea level).

SUBROUTINE GLORENZ -- Lorentzian-ridge bottom model.

SUBROUTINE GTANH -- 2-D bottom model with a series of linear segments joined smoothly.

SUBROUTINE NPBOTM -- Do-nothing bottom-perturbation model.

SUBROUTINE SHORIZ -- Ocean surface = a sphere at fixed height above MSL.

SUBROUTINE NPSURF -- Do-nothing ocean-surface perturbation model.

SUBROUTINE RHORIZ -- Horizontal receiver-surface model.

SUBROUTINE RBOTM -- Receiver-surface model at fixed height above the bottom.

SUBROUTINE RVERT -- Vertical (conical) receiver surface at a fixed central-earth angle from a specified geographic location (latitude and longitude).

[ANNOTATION MODELS]

SUBROUTINE SMPANN -- Initializes plot in draft mode (must be used if you don't have DISSPLA).

SUBROUTINE FULANN -- Initializes plot in publication-quality mode (requires DISSPLA).

7.2 HARPO Organization Diagrams

This section contains hierarchical diagrams, Figures 7.1 through 7.4, that show how the principal subroutines are interrelated by calling sequences. These diagrams are not flow charts; they show how control passes among the program modules. Not all subroutines are shown, only the major ones that perform the ray-tracing function.

7.3 Flow Charts for Program RAYTRC and Subroutines TRACE and BACKUP

These three routines contain the central logic of the raypath calculations and so are described in detail in flow-chart form.

This ray-tracing program consists of various subroutines that perform specific tasks in calculating raypaths. The division of labor makes it easier

to modify the program to solve specific problems. Often it may be necessary to change only one or two subroutines to convert the program to a different use.

The main program (RAYTRC) sets up the initial conditions (transmitter location, wave frequency, and direction of transmission) for each ray trace. In setting up the initial conditions for each ray trace, the main program (RAYTRC) steps frequency, azimuth angle of transmission, and elevation angle of transmission (see Figure 7.5). Then subroutine TRACE calculates one raypath for the requested number of crossings of the specified receiver height. Subroutine TRACE is the heart of the ray-tracing program. It is the most complicated subroutine included, but also the most important to understand. The flow charts in Figures 7.6, 7.7, and 7.8 explain how TRACE works.

Subroutine RKAM integrates the differential equations numerically using an Adams-Moulton predictor-corrector method with a Runge-Kutta starter. Subroutine HAMLTN evaluates the differential equations to be integrated. A subroutine with entry point DISPER calculates the Hamiltonian and its derivatives, the wave number from the dispersion relation, and the wave polarization. (Four versions of this subroutine are included, as noted in Sec. 6.4.) Subroutines with entry points WINDR, SPEED, RECVR, TOPOG, SURFACE and ABSRP calculate the ocean current velocity, sound speed, receiver surface, bottom, upper surface, and absorption. Several versions of these six subroutines are included, and it is straightforward to add more. Subroutine BACKUP finds an intersection of the ray with the receiver surface, ocean bottom, or upper surface. The flow charts in Figures 7.9 through 7.11 and Section 6.5 explain how BACKUP works.

Subroutine PRINTR prints information describing the raypath and outputs the results in computer-readable form (raysets). Subroutine RAYPLT plots the raypath. The block diagrams in Figures 7.1 through 7.4 show the relation among these (and other) subroutines.

The listings of most of the subroutines have comments that should help in understanding how they work. In addition, Tables 7.1 through 7.38 define the variables in the common blocks.

7.4 Common-Block Structure and Usage

We use common blocks instead of calling sequences to pass information between subroutines and functions because it is faster. However, the added complexity requires a detailed description of how those common blocks are organized, which blocks link which program modules, and the variables in each block.

Table 7.1 defines the variables in blank common. These are mostly the dependent variables in the numerical integration and their derivatives. Nearly all of the subprograms use this common block.

Table 7.2 describes the common block /MCONST/, which contains mathematical constants. Table 7.3 describes the common block /PCONST/, which contains physical constants.

Many common blocks are used to cummunicate among the various routines in the program. Table 7.4 lists those common blocks and shows which routines use those common blocks. Blank common, common block /WW/, and the model-related common blocks listed in Tables 4.2 and 4.3 are not included in Table 7.4. Table 7.5 lists the variable names in these common blocks that are used for input and output by each routine. Tables 7.6 through 7.32 list all of the variables in these common blocks and give the meanings of those variables.

Table 7.17 describes common block /RIN/, which contains parameters output by all of the versions of the dispersion relation subroutines (all of which have the entry point name DISPER). Tables 7.33 through 7.38 describe the common blocks /UU/, /CC/, /LL/, /SS/, /RR/, and /GG/, which contain the parameters output by the various ocean models.

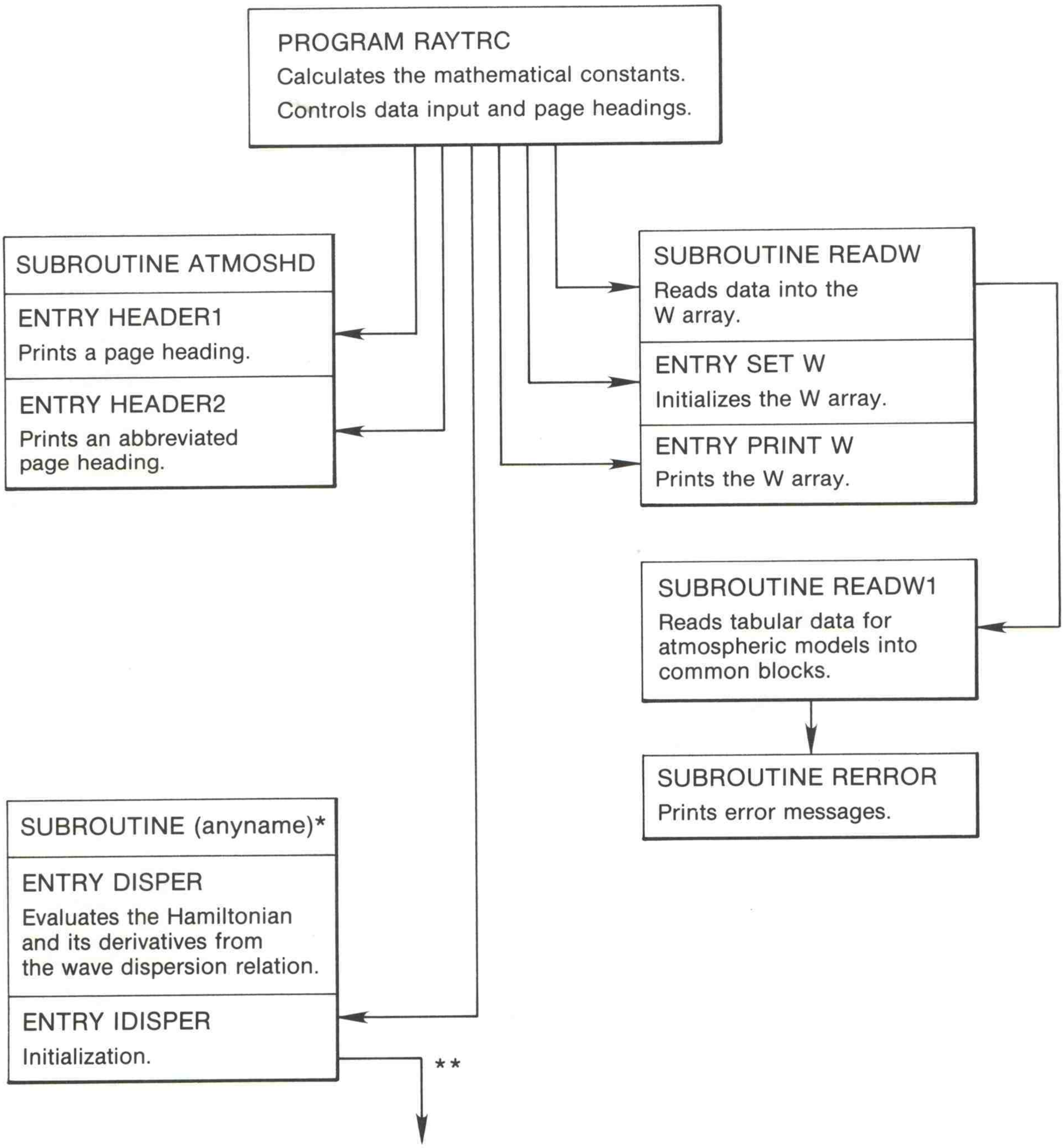


Figure 7.1. Block diagram (not a flow chart) of the ray-tracing program showing the relation (hierarchy, what calls what) of the main program to other subroutines during the initialization stage (immediately after new input data are read in).

* ANCNL (Acoustic, No Currents, No Losses), AWCNL (Acoustic, With Currents, No Losses), ANCWL, and AWCWL are the names of the versions of the dispersion-relation subroutines available.

** Figure 7.2 shows the continuation of this block diagram.

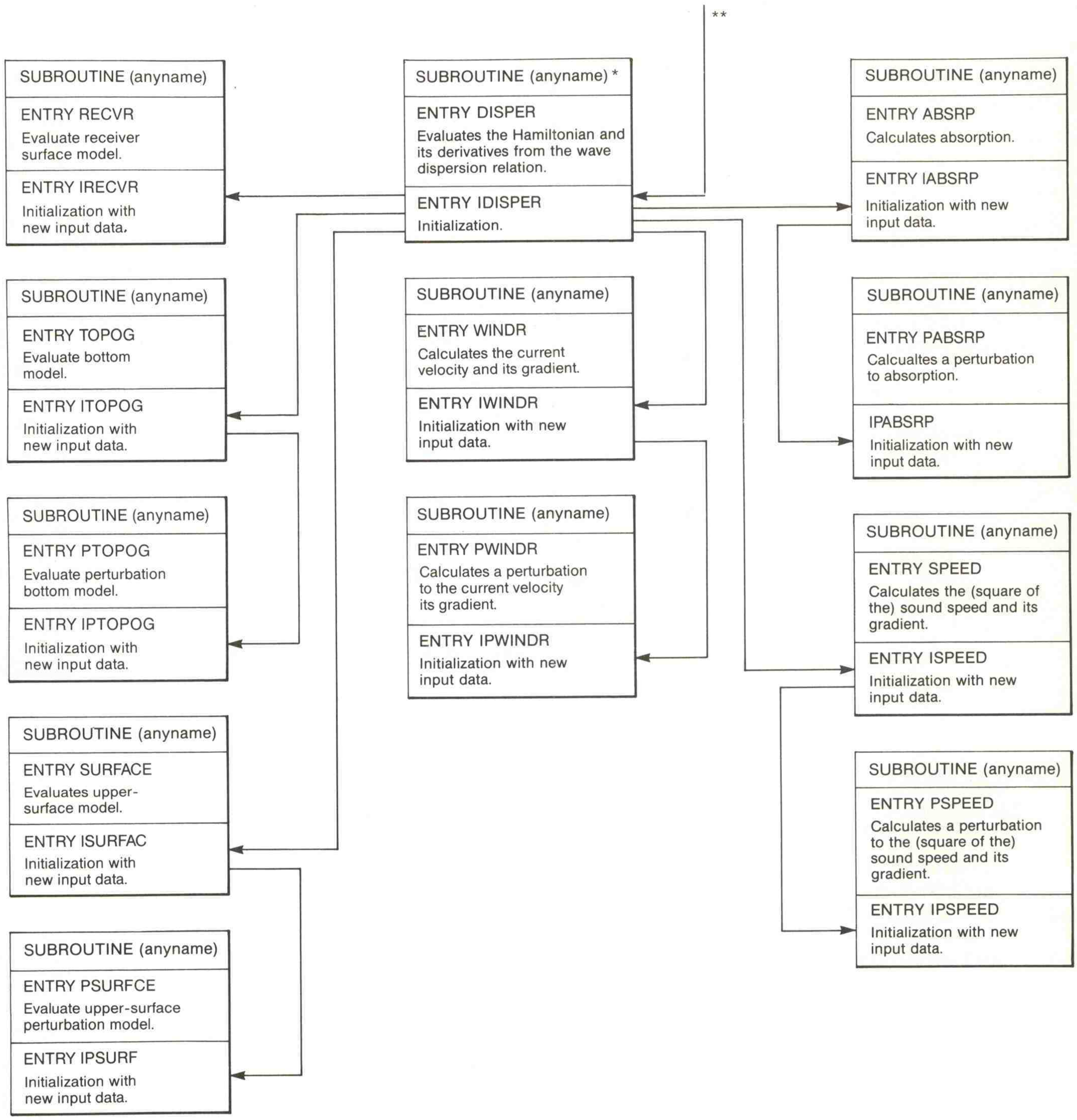


Figure 7.2. Continuation of the block diagram of the ray-tracing program (Fig. 7.1), showing the relations among the atmospheric model subroutines during the initialization stage (immediately after new input data are read in).

* ANCNL (Acoustic, No Currents, No Losses), AWCNL (Acoustic, With Currents, No Losses), ANCWL, and AWCWL are the names of the versions of the dispersion-relation subroutine available with this version of the program.

** Continued from Figure 7.1.

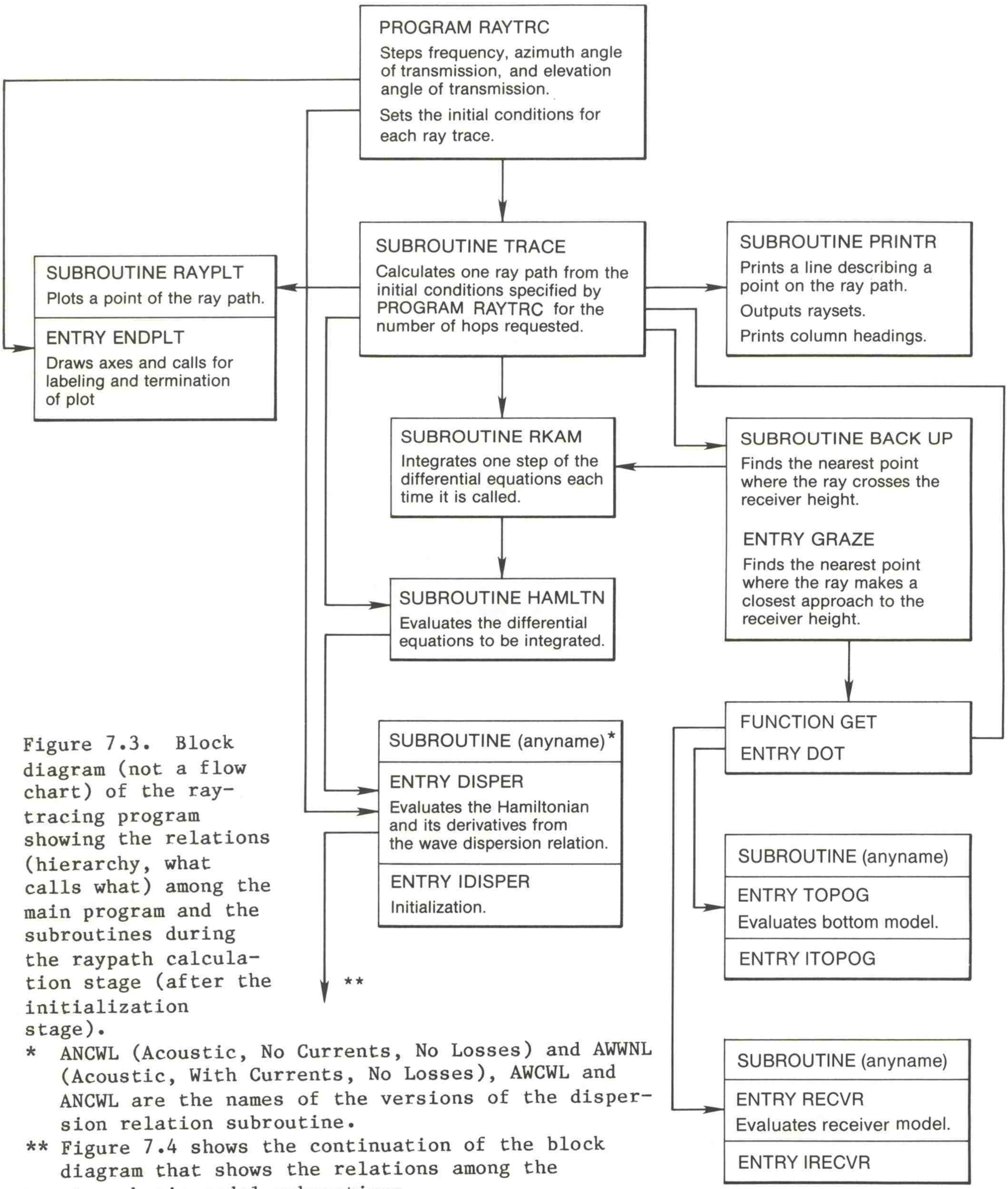


Figure 7.3. Block diagram (not a flow chart) of the ray-tracing program showing the relations (hierarchy, what calls what) among the main program and the subroutines during the raypath calculation stage (after the initialization stage).

* ANCWL (Acoustic, No Currents, No Losses) and AWWNL (Acoustic, With Currents, No Losses), AWCWL and ANCWL are the names of the versions of the dispersion relation subroutine.

** Figure 7.4 shows the continuation of the block diagram that shows the relations among the atmospheric model subroutines.

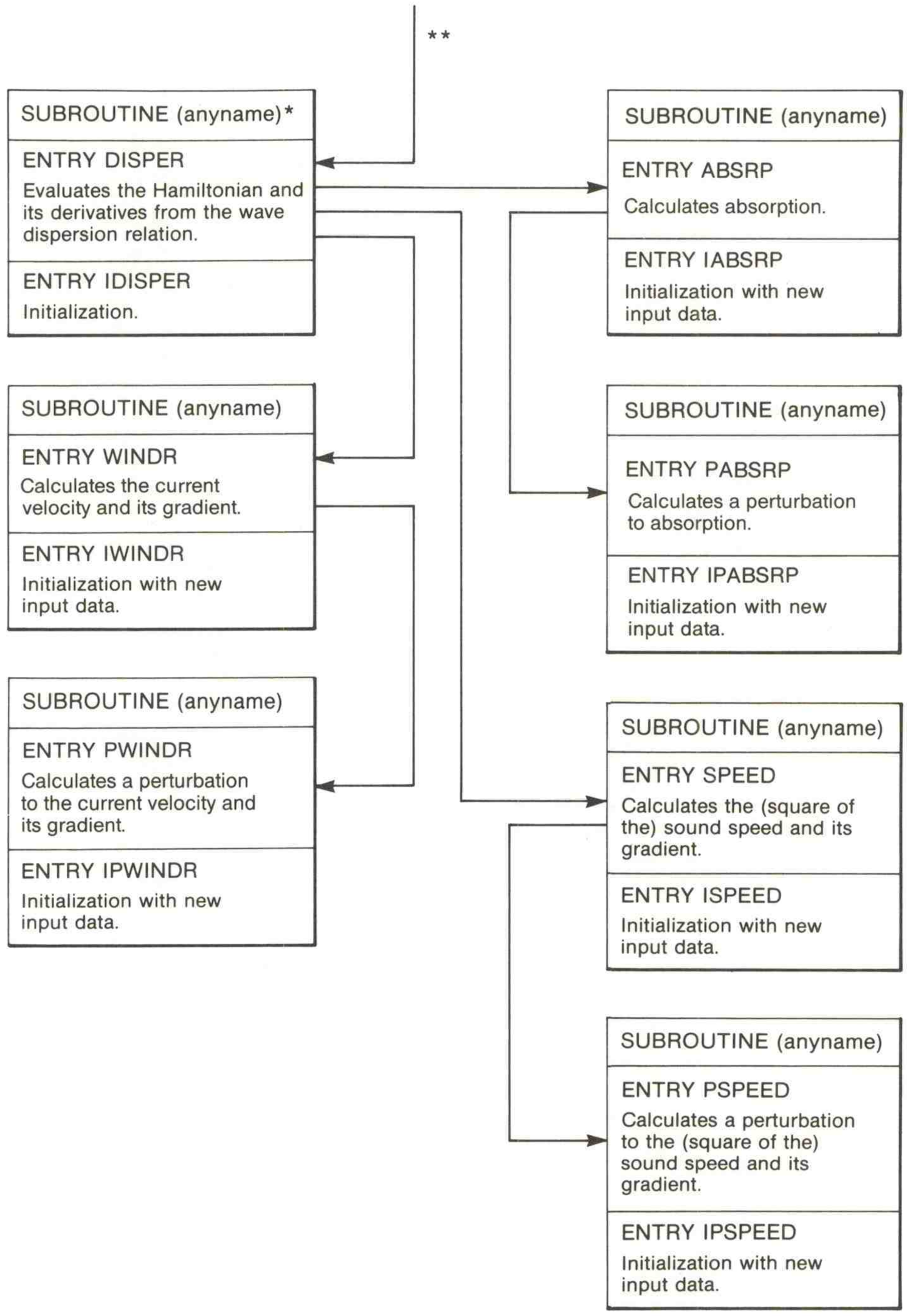


Figure 7.4. Continuation of the block diagram of the ray-tracing program (Fig. 7.3) that shows the relations (hierarchy, what calls what, a block diagram is not a flow chart) among the ocean model subroutines during the raypath calculation stage (after the initialization stage).
* ANCNL (Acoustic, No Currents, No Losses) and AWCNL (Acoustic, With Currents, No Losses), AWCWL and ANCWL are the names of the versions of the dispersion-relation subroutine.
** Continued from Figure 7.3.

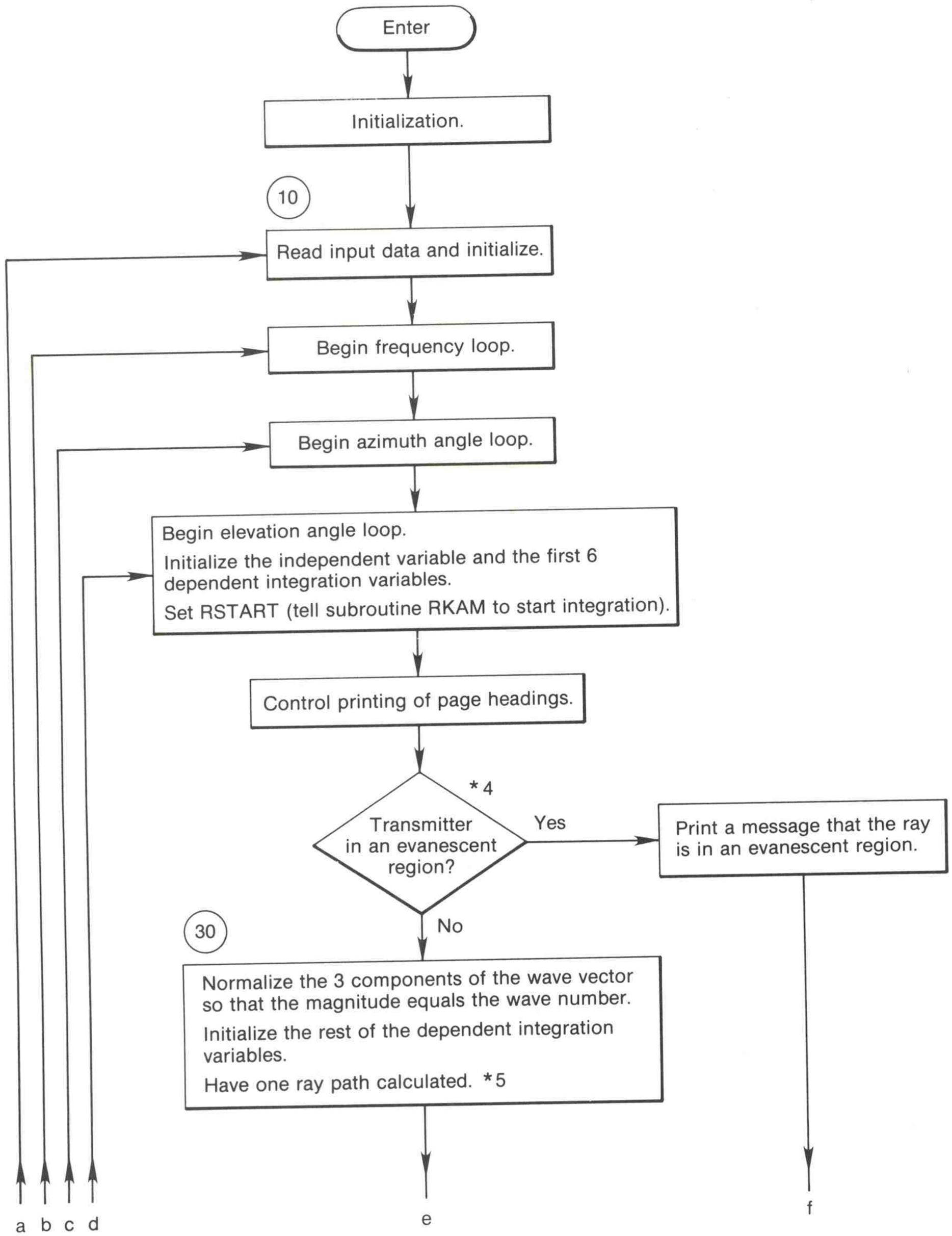


Figure 7.5. Flow chart for program RAYTRC. Circled block numbers correspond to program statement numbers.
*4 There are no evanescent regions for pure acoustic waves (with no cutoff frequency).
*5 Subroutine TRACE calculates one raypath.

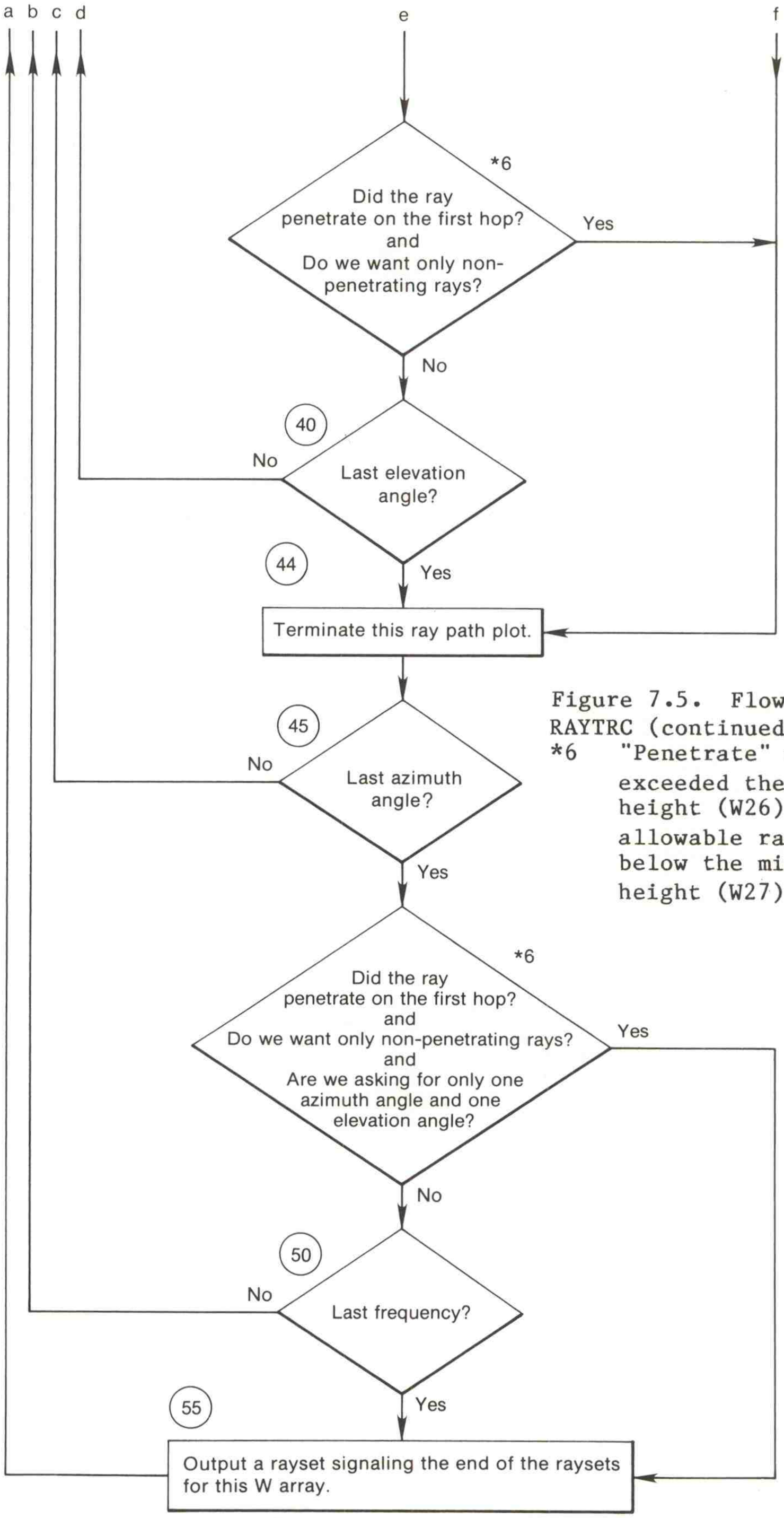


Figure 7.5. Flow chart for program RAYTRC (continued).

*6 "Penetrate" means the ray exceeded the maximum allowable height (W26) or the maximum allowable range (W28) or went below the minimum allowable height (W27).

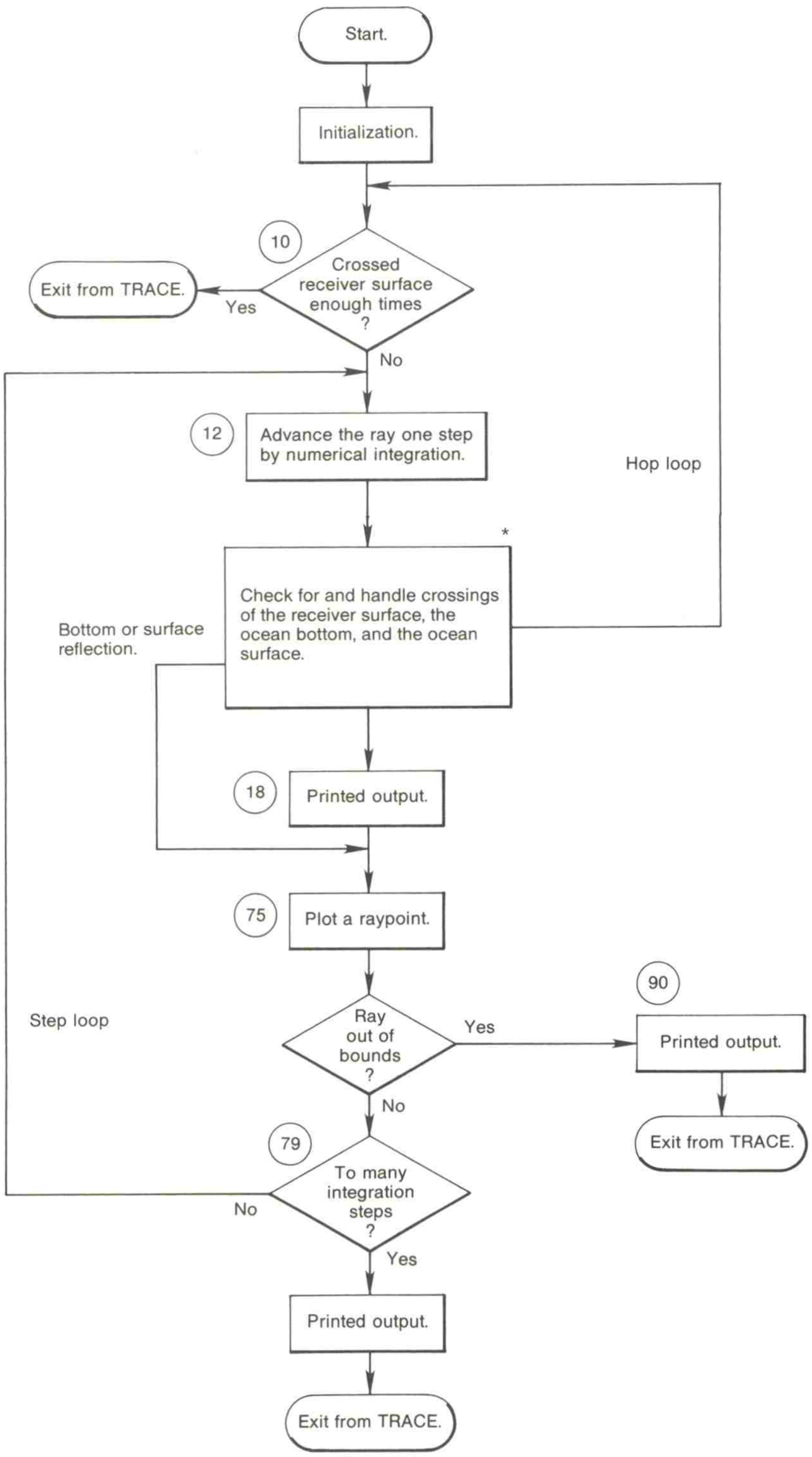


Figure 7.6. Flow chart for subroutine TRACE. Circled block numbers correspond to program statement numbers in subroutine TRACE.
* See Figure 7.7 for details.

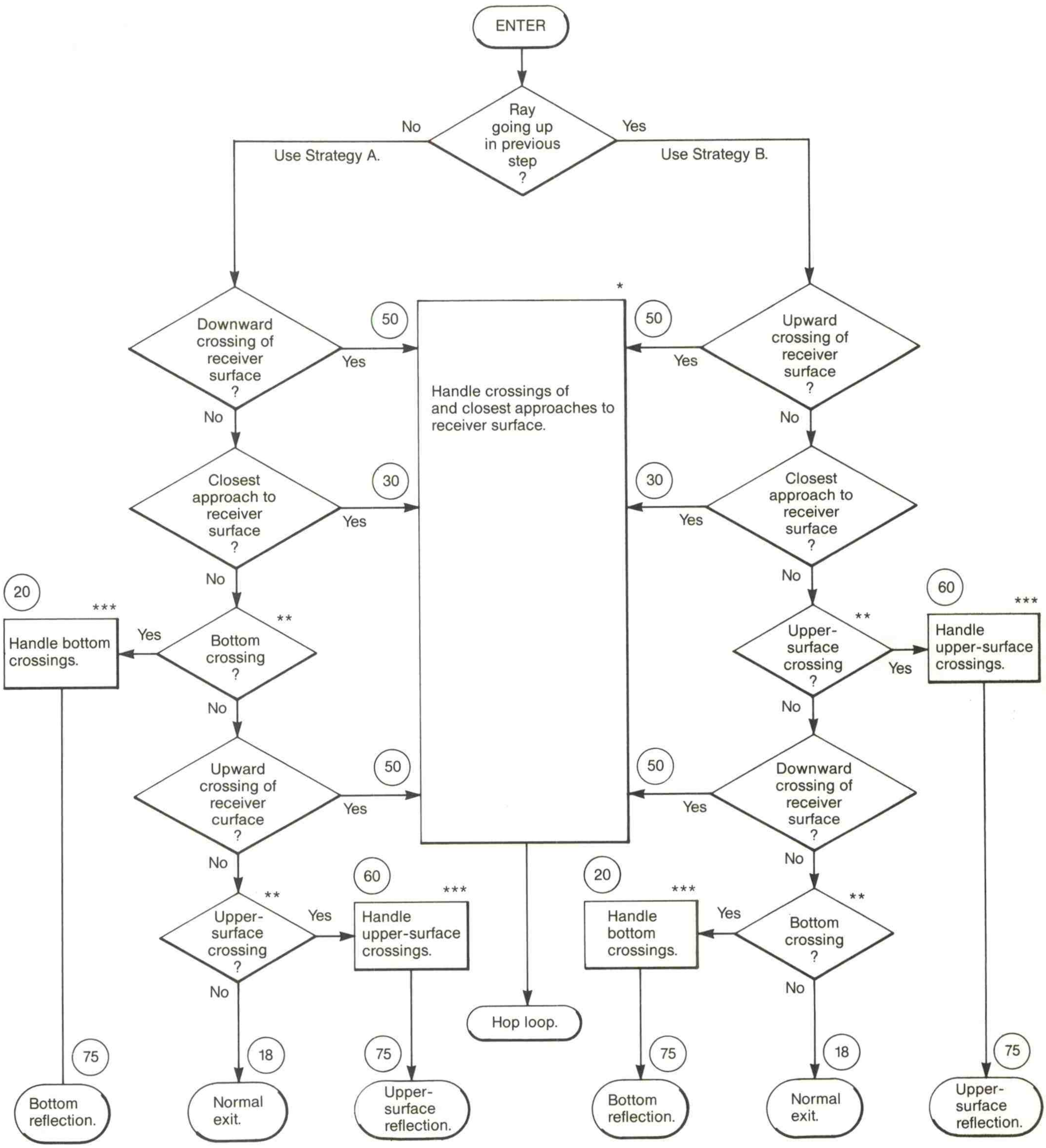


Figure 7.7. Flow chart for details of handling surface crossings in subroutine TRACE. Circled block numbers correspond to program statement numbers.

* See Figure 7.8 for details
** Logical function PCROSS estimates whether a bottom crossing or upper-surface crossing has occurred.
*** Subroutine RCROSS handles bottom crossings and upper-surface crossings.

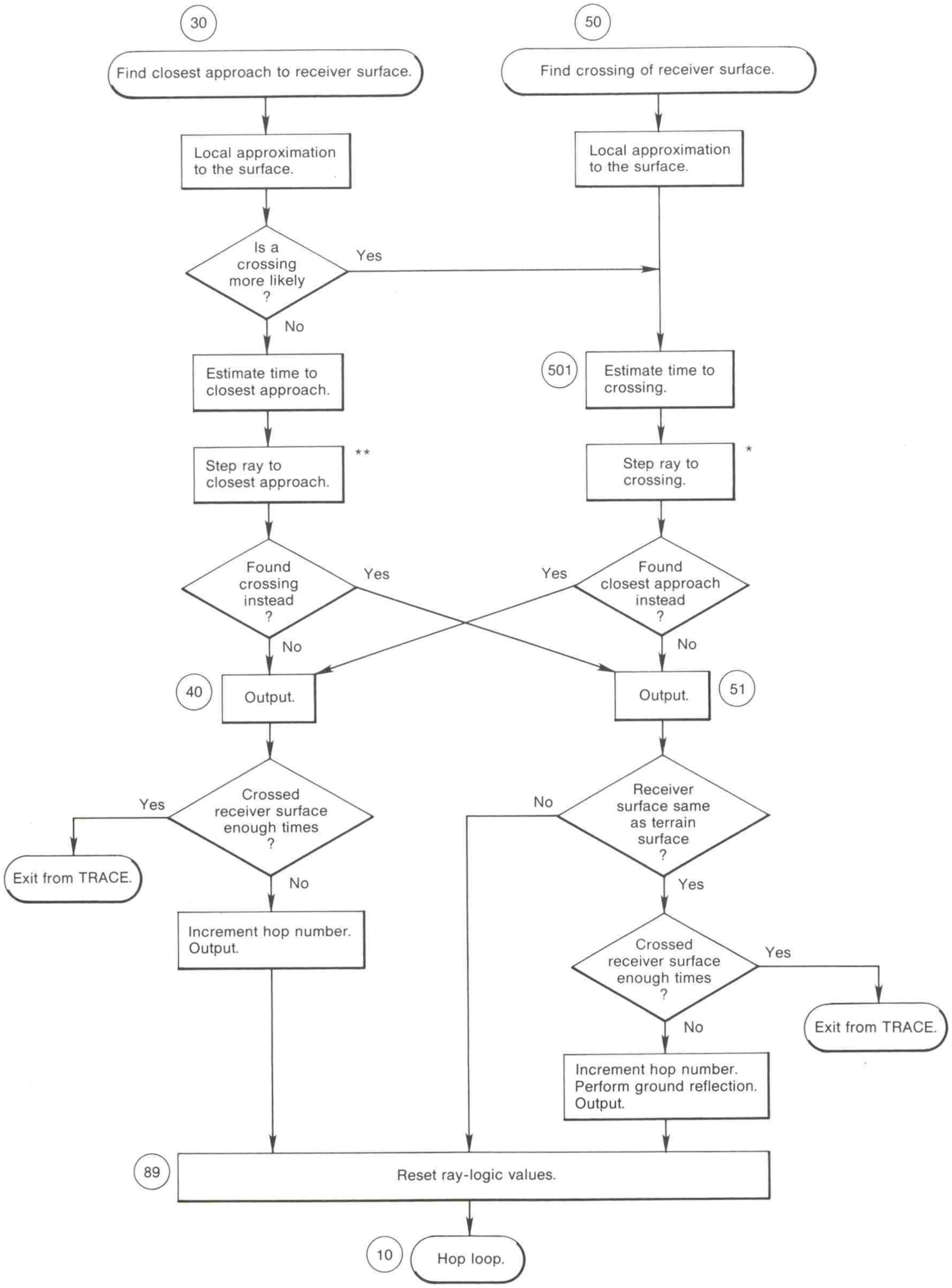


Figure 7.8. Details of finding crossings of and closest approaches to receiver surface. Circled block numbers correspond to program statement numbers in subroutine TRACE.
* See Figure 7.9 for details (subroutine BACKUP).
** See Figure 7.9 for details (entry point GRAZE in subroutine BACKUP).

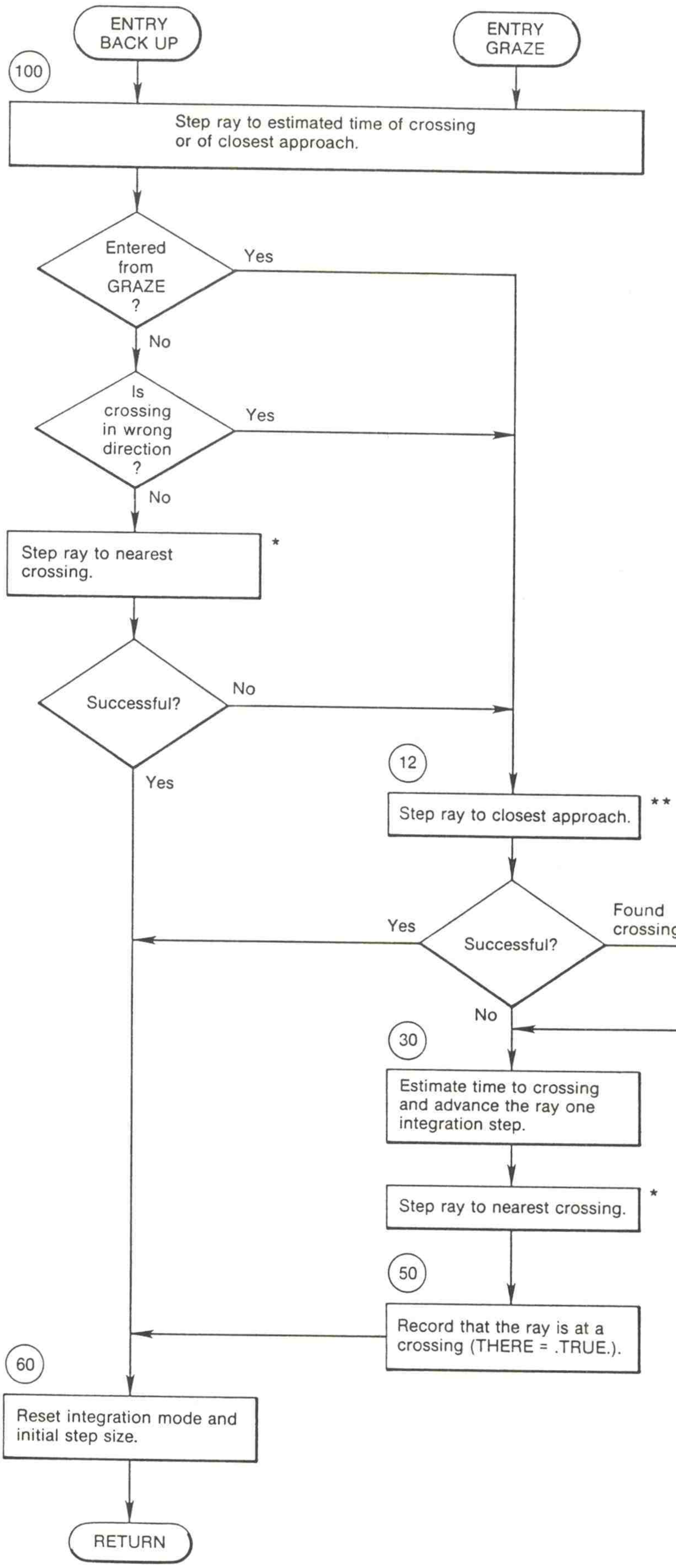


Figure 7.9. Flow chart for subroutine BACKUP. Circled block numbers correspond to program statement numbers. Entry BACKUP steps the ray to a crossing with a specified height. Entry GRAZE steps the ray to a point of closest approach to the specified surface. The calling routine (subroutine TRACE) specifies the height with which the ray is to intersect, the direction of crossing (up or down), and estimates the time of crossing (group time delay). Asterisks identify supplementary procedures.

* See Figure 7.10 for details of the algorithm that steps the ray by numerical integration to the nearest crossing of a specified height.

** See Figure 7.11 for details of the algorithm that steps the ray by numerical integration to a closest approach to a specified height.

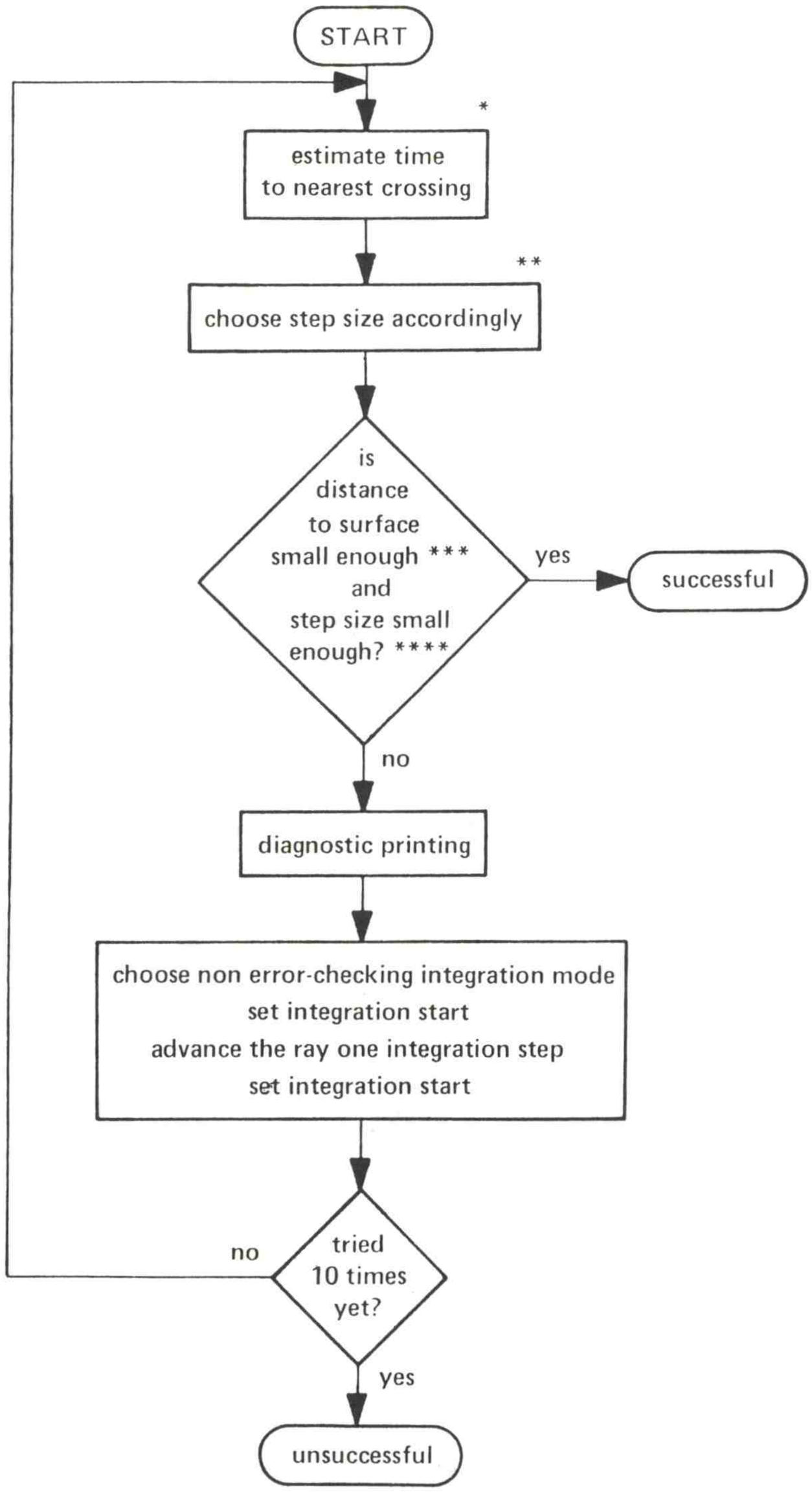


Figure 7.10. Flow chart for the algorithm that steps the ray by numerical integration to a crossing of a specified surface. Circled block numbers correspond to program statement numbers.

* See Equation (6.76) to estimate the time of the nearest crossing of the specified height.

** The step size should be no larger than that being used by the numerical integration routine to maintain accuracy in the error-checking mode.

*** 0.5×10^{-4} km.

**** Small enough to ensure the required accuracy and smaller than the smallest allowable step size.

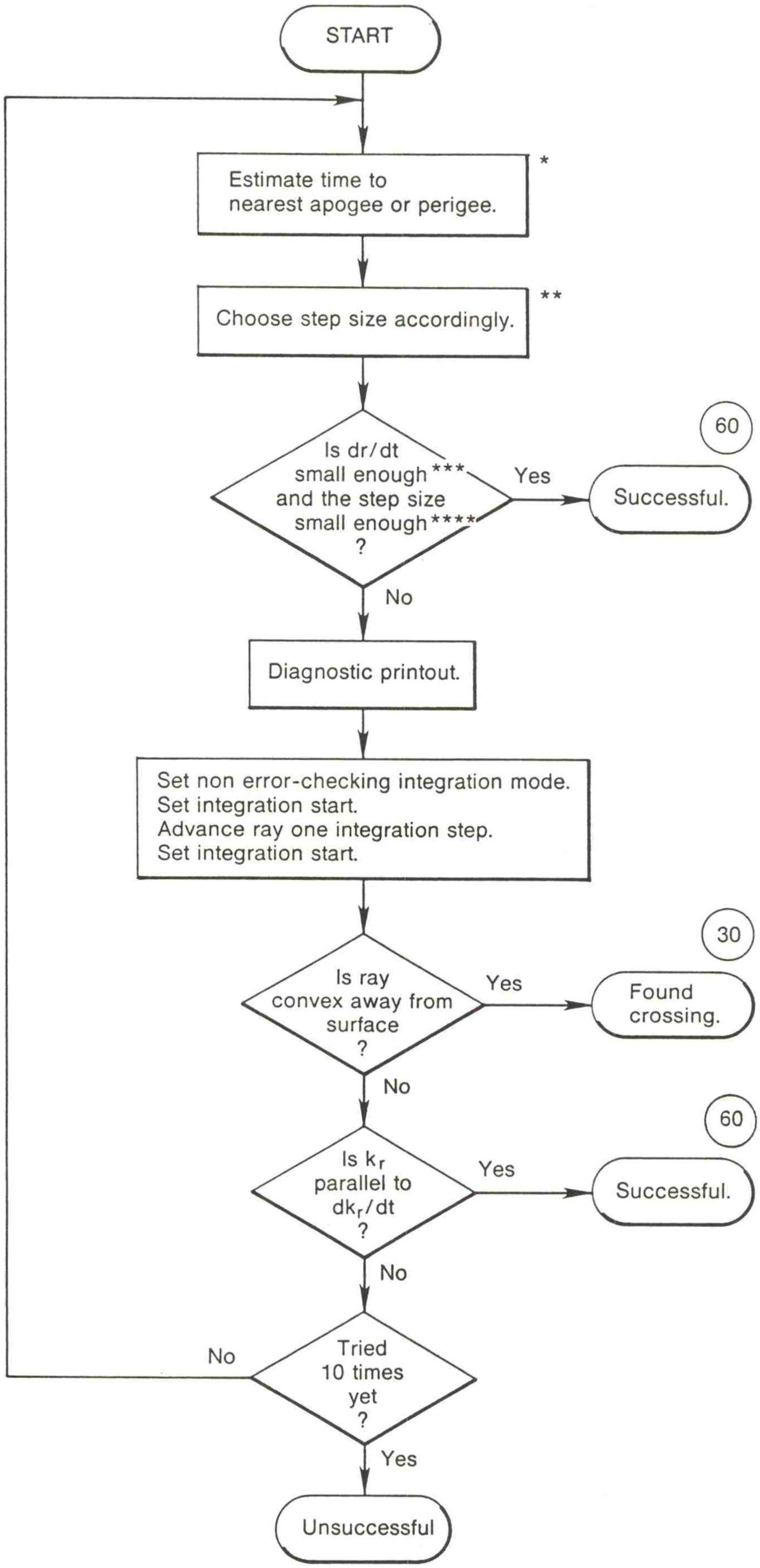


Figure 7.11. Flow chart for the algorithm that steps the ray by numerical integration to a point of closest approach to a specified surface. Circled block numbers correspond to program statement numbers.

* See Equation (6.74) to estimate the time of the specified surface.

** The step size should be no larger than that being used by the numerical integration routine to maintain accuracy in the error-checking mode.

*** 10^{-6} km/km.

**** Small enough to ensure the required accuracy and smaller than the smallest allowable step size.

Table 7.1--Definitions of the parameters in blank common

Position in common	Variable name	Definition
1-20	R	The dependent variables in the differential equations being integrated--the definitions of the first six are fixed, but the others may be varied by the program user
1	R(1) or R	r
2	R(2) or TH	θ
3	R(3) or PH	ϕ
4	R(4) or KR	k_r
5	R(5) or KTH	k_θ
6	R(6) or KPH	k_ϕ
7-12	R(7)-R(12) or RKVARS(1)-RKVARS(6)	Those variables the user has chosen to integrate, taken in the following order: P -phase path in kilometers A -absorption in decibels Δf -Doppler shift in hertz s -geometrical path length in kilometers
13-20	R(13)-R(20)	Reserved for future expansion
21	TPULSE	Group path length in kilometers (the independent variable in the differential equations)
22	CSTEP	Step length in group path
23-42	DRDT	The derivatives of the dependent variables with respect to the independent variable TPULSE

R and TPULSE are initialized in program RAYTRC and changed in subroutines TRACE, RKAM, and BACKUP.

CSTEP is calculated in subroutine RKAM.

DRDT is calculated in subroutine HAMLTN and used in subroutine RKAM.

Table 7.2--Definitions of the parameters in common block /MCONST/*

Position in common	Variable name	Definition
1	PI	π
2	PIT2	2π
3	PID2	$\pi/2$
4	DEGS	$180.0/\pi$
5	RAD	$\pi/180.0$
6	ALN10	$\log_e 10$

* These parameters are set in program RAYTRC.

Table 7.3--Definitions of the parameters in common block /PCONST/

Position in common	Variable name	Definition
1	CREF	A reference sound speed (1.500 km/s), used to convert group delay time in seconds to an equivalent group path length in kilometers, and to convert a phase time in seconds to an equivalent phase-path length in kilometers
2	RGAS	The gas constant, = 8.31436×10^{-3} kg (kg mole)$^{-1}$ km^2 s^{-2} K^{-1} (this value of the gas constant gives a sound speed in km/s). Used by HARPA.
3	GAMMA	γ, the ratio of specific heat at constant pressure to that at constant density, = 1.4. Used by HARPA.

Table 7.4--Common block usage by the core routines

Routine	Common block	HDR	UCON	UCOM	RKAM	CGEOM	CRMCHEL	CRMKTOC	CRMLOC	TRKRAD	TFNDER	RINRR	ERRAYDE	FLGPCE	FLGNRL	FLGRCE	HDIRLE	PLTDAT	RANYCHK	AANNTOR	LABCOM	DDRZ	DKNKE	DDLMI
RAYTRC		O							B			I	I	I	O		B	O	O				
CONBLK		O	O	O			O				O			B			O		O				
DFCNST						O																		
UCON			I	B																				
HAMLTN											I													
RKAM					B		B		B															
RKAM1					O				B															
TRACE							B	B	B	B									O					
PCROSS										I														
RCROSS								O	O	B									O					
BACKUP									B	B														
REFLECT										B	I													
FIT								B	I															
GET			O	I	I			O																
GET1			O	I	I																			
READW1												I												
READW												I	I		I									
PRINTR						I			I	O					B									
OCNHD									I				I	I	B	I								
RAYPLT															B		I	O						
PLOT																		B			O	O		
LABPLT									I							I					O			
PLTANH																	I			I	O			
SETXY																				O				
PLTANOT																			I	I	I	B		
PLTLB																					I	B		
PLOTBL																							O	O
DDBP																						I	B	B B
DDVC																						I	B	B B
DDTEXT																						I		B

Notes:

1. "I" signifies that the routine uses information from the common block.
2. "O" signifies that the routine puts information into the common block.
3. "B" signifies that the routine both puts information into the common block and uses information from the common block.
4. Blank common, common blocks /MCONST/, /PCONST/, /WW/, and the common blocks listed in Tables 4.2 and 4.3 are not included in this table.

Table 7.5--Input and output variables to routines in common blocks other than blank common and labeled common blocks /MCONST/, /PCONST/, and /WW/ and other than the common blocks listed in Tables 4.2 and 4.3

Routine	Input parameters		Output parameters	
	common block	parameter name	common block	parameter name
RAYTRC	/FLG/	IHOP	/HDRC/	DAT
	/RIN/	KAY2	/HDRC/	INITID
	/FLG/	LINES	/FLG/	LINES
	/RK/	NEQS	/FLG/	NEWWP
	/ERR/	NERG	/FLG/	NEWWR
	/ERR/	NERP	/FILEC/	NPLTDP
	/ERR/	NERR	/FLGP/	NSET
	/ERR/	NERT	/HDR/	SEC
	/RAYDEV/	NRYIND	/HDRC/	TOD
	/RIN/	OMEGMAX	/RK/	RSTART
	/RIN/	OMEGMIN		
	/FLG/	PENET		
	/RIN/	SGN		
DFCNST			/CRMACH/	RMACH
READW1	/RAYDEV/	NDEVTMP		
	/RAYDEV/	NRYIND		
READW	/B1/→/B20/		/B1/→/B20/	
	/FLG/	LINES		
	/RAYDEV/	NDEVTMP		
	/RAYDEV/	NFRMAT		
	/RAYDEV/	NRYIND		
	/FLGP/	NSET		
UCON	/UCONC/	CNVC	/UCONV/	CNVV
	/UCONV/	CNVV		
	/UCONC/	PCV		
TRACE	/TRAC/	D2Z	/TRAC/	DROLD
	/TRAC/	GROUND	/RK/	E1MAX
	/CRKTIME/	IRKTIME	/RK/	E1MIN
	/TRAC/	RAD	/RK/	E2MAX
	/TRAC/	RAD1	/RK/	E2MIN
	/TRAC/	THERE	/RK/	FACT
	/TRAC/	ZDOT	/TRLOCAL/	FDOT
	/TRLOCAL/	RSIGN	/TRLOCAL/	GDOLD
	/TRLOCAL/	HOME	/TRLOCAL/	GDOT
	/TRLOCAL/	FDOT	/TRLOCAL/	GOLD
			/TRAC/	GROUND
			/TRAC/	HOME
			/FLG/	HPUNCH
			/FLG/	IHOP

Table 7.5--(continued)

Routine	Input parameters common block	parameter name	Output parameters common block	parameter name
TRACE (continued)			/CRKTIME/	IRKTIME
			/RK/	MODE
			/TRAC/	NEWRAY
			/FLG/	NEWTRC
			/TRAC/	PENET
			/TRLOCAL/	ROLD
			/RK/	RSIGN
			/RK/	RSTART
			/TRAC/	STEP
			/TRAC/	THERE
				TOLD
PCROSS	/TRAC/	OSMT		
	/TRAC/	SMT		
RCROSS	/TRAC/	RAD1	/TRLOCAL/	FDOT
	/TRAC/	ZDOT	/TRLOCAL/	HOME
			/FLG/	HPUNCH
			/TRLOCAL/	RSIGN
			/RK/	RSTART
HAMLTN	/RIN/			
RKAM	/RKAMS/	ALPHA	/RK/	E1MAX
	/RK/	E1MAX	/CRKTIME/	IRKTIME
	/RKAMS/	FV	/RKAMS/	MM
	/CRKTIME/	IRKTIME	/RK/	MODE
	/RKAMS/	MM	/RK/	NEQS
	/RK/	MODE	/RKAMS/	XV
	/RK/	NEQS		
	/RK/	RSTART		
	/RK/	STEP		
	/RKAMS/	XV		
	/RKAMS/	YU		
RKAM1	/RK/	E1MAX	/RKAMS/	ALPHA
	/RK/	E1MIN	/RKAMS/	EPM
	/RK/	E2MAX	/RK/	E1MIN
	/RK/	E2MIN	/RK/	FACT
	/RK/	FACT	/RKAMS/	FV
	/RK/	MODE	/RKAMS/	MM
	/RK/	NN	/RK/	RSTART
	/RK/	RSTART	/RKAMS/	XV
	/RK/	SPACE	/RKAMS/	YU

Table 7.5--(continued)

Routine	Input parameters		Output parameters	
	common block	parameter name	common block	parameter name
BACKUP	/TRAC/	DROLD	/RK/	MODE
	/TRAC/	D2Z	/RK/	RSTART
	/RK/	E1MAX	/RK/	STEP
	/RK/	E2MIN	/TRAC/	THERE
	/RK/	FACT	/TRAC/	TOLD
	/RK/	MODE	/TRAC/	ZDOT
	/TRAC/	RAD1		
REFLECT	/FNDER/	NPZPH		
	/FNDER/	NPZR		
	/FNDER/	NPZTH		
FIT	/TRAC/	DROLD	/TRAC/	D2Z
	/FNDER/	NPZPH	/TRAC/	OSMT
	/FNDER/	NPZPHPH	/TRAC/	RAD
	/FNDER/	NPZR	/TRAC/	RAD1
	/FNDER/	NPZRPH	/TRAC/	SMT
	/FNDER/	NPZRR	/TRAC/	ZDOT
	/FNDER/	NPZRTH		
	/FNDER/	NPZTH		
	/FNDER/	NPZTHPH		
	/FNDER/	NPZTHTH		
	/FNDER/	NZ		
	/TRAC/	TOLD		
GET	/CRKTIME/	IRKTIME	/FNDER/	NPZPH
	/CRKTIME/	RKTIME	/FNDER/	NPZPHPH
	/CRMACH/	RMACH	/FNDER/	NPZR
			/FNDER/	NPZRPH
			/FNDER/	NPZRR
			/FNDER/	NPZRTH
			/FNDER/	NPZTH
			/FNDER/	NPZTHPH
			/FNDER/	NPZTHTH
			/FNDER/	NSELECT
			/FNDER/	NTIME
			/FNDER/	NZ
			/CGET/	ZERO
GET1	/CRKTIME/	IRKTIME	/CGET/	ZERO
	/CRKTIME/	RKTIME		
	/CRMACH/	RMACH		
CONBLK	/RAYDEV/	NRYIND	/UCONC/	CNVC
			/UCONV/	CNVV
			/CRKTIME/	IRKTIME
			/RIN/	KVECT

Table 7.5--(continued)

Routine	Input parameters common block	parameter name	Output parameters common block	parameter name
CONBLK (continued)			/RIN/	LPOLAR
			/RAYCON/	MCONP
			/RAYDEV/	NDEVTMP
			/RAYDEV/	NFRMAT
			/RAYDEV/	NRYIND
			/RIN/	OMEGMAX
			/RIN/	OMEGMIN
			/UCONC/	PCV
			/RIN/	POLAR
			/RIN/	RAYNAME
			/HDRC/	DAT
			/HDRC/	INITID
			/PROCFL/	LIST
			/PROCFL/	PITBL
			/PROCFL/	PNTBL
			/HDR/	SEC
			/HDRC/	TOD
PRINTR	/FLG/	HPUNCH	/FLG/	LINES
	/FLG/	IHOP	/ERR/	NERG
	/RIN/	KAY2	/ERR/	NERP
	/FLG/	LINES	/ERR/	NERR
	/RK/	NEQS	/ERR/	NERT
	/FLG/	NTYP	/FLG/	NEWWP
	/RIN/	POLAR		
	/RIN/	TYPE		
OCNHD	/HDRC/	DAT	/FLG/	LINES
	/RINPL/	DISPM	/FLG/	NTYP
	/FLG/	LINES		
	/RIN/	MODRIN		
	/FLGP/	NSET		
	/RIN/	RAYNAME		
	/HDRC/	TOD		
PUTKST	/FLG/	LINES	/FLG/	LINES
RAYPLT	/FLG/	NEWTRC	/PLT/	ALPHA
	/FLG/	NEWWR	/PLT/	APLT
	/FILEC/	NPLTDP	/FLG/	NEWTRC
			/FLG/	NEWWR
			/PLT/	PRESET
			/PLT/	RMAX
			/PLT/	RMIN
			/PLT/	XL
			/PLT/	XR
			/PLT/	YB
			/PLT/	YT

Table 7.5--(continued)

Routine	Input parameters		Output parameters	
	common block	parameter name	common block	parameter name
PLOT	/PLT/	ALPHA	/DD/	IX
	/PLT/	APLT	/DD/	IY
	/PLT/	RESET	/PLT/	RESET
	/PLT/	RMAX	/DDREZ/	DDHIX
	/PLT/	RMIN	/DDREZ/	DDHIY
	/PLT/	XMAXO		
	/PLT/	XMINO		
	/PLT/	YMAXO		
	/PLT/	YMINO		
LABPLT	/HDRC/	DAT	/DD/	IOR
	/RIN/	MODRIN	/DD/	IS
			/DD/	IT
			/DD/	IX
			/DD/	IY
PLTANH	/RAYCON/	MCONP	/DD/	IOR
	/LABCLT/	PROJCT	/DD/	IX
	/LABCLT/	RMAX	/DD/	IY
	/LABCLT/	RMIN		
	/LABCLT/	THMAX		
	/LABCLT/	THMIN		
SETXY			/LABCLT/	PROJCT
			/LABCLT/	RMAX
			/LABCLT/	RMIN
			/LABCLT/	THMAX
			/LABCLT/	THMIN
PLOTANOT	/ANNCTC/	ANOTES	/DD/	IOR
	/ANNCTC/	HNOTES	/DD/	IX
	/DD/	IX	/DD/	IY
	/DD/	IY		
	/ANNCTL/	LENA		
	/ANNCTL/	LENHA		
	/LABCLT/	THMAX		
	/LABCLT/	THMIN		
PLTLB	/DD/	IX	/DD/	IOR
	/DD/	IY	/DD/	IX
	/LABCLT/	PROJCT		
	/LABCLT/	RMAX		
	/LABCLT/	RMIN		

Table 7.5--(continued)

Routine	Input parameters common block	parameter name	Output parameters common block	parameter name
PLOTBL			/KNKN/	
			/DDLIM/	
DDBP	/DD/	IX	/KNKN/	KNBP
	/DD/	IY	/DDLIM/	MNIX
	/KNKN/	KNBP	/DDLIM/	MNIY
	/DDLIM/	MNIX	/DDLIM/	MXIX
	/DDLIM/	MNIY	/DDLIM/	MXIY
	/DDLIM/	MXIX	/DDREZ/	DDHIX
	/DDLIM/	MXIY	/DDREZ/	DDHIY
	/DDREZ/	DDHIX		
	/DDREZ/	DDHIY		
DDVC	/DD/	IX	/KNKN/	KNVC
	/DD/	IY	/DDLIM/	MNIX
	/KNKN/	KNVC	/DDLIM/	MNIY
	/DDLIM/	MNIX	/DDLIM/	MXIX
	/DDLIM/	MNIY	/DDLIM/	MXIY
	/DDLIM/	MXIX	/DDREZ/	DDHIX
	/DDLIM/	MXIY	/DDREZ/	DDHIY
	/DDREZ/	DDHIX		
	/DDREZ/	DDHIY		
DDTEXT	/DD/	IOR	/KNKN/	KNDT
	/DD/	IX		
	/DD/	IY		
	/KNKN/	KNDT		

Table 7.6--Definitions of the parameters in common block /HDR/

Position in common	Variable name	Definition
1	SEC	Total elapsed computer calculation time at end of calculating previous raypath

Table 7.7--Definitions of the parameters in common block /UCONC/

Position in common	Variable name	Definition
1-4	PCV	List of valid unit types for units conversion: blank (no conversion), AN (angle), LN (length), FQ (frequency, AM (amplitude)
5-20	CNVC	An array of lists of valid physical units for each unit type for units conversion: blank (no conversion), M (meters), KM (kilometers), DG (degrees), etc

Table 7.8--Definitions of the parameters in common block /UCONV/

Position in common	Variable name	Definition
1-16	CNVV	An array of units conversion factors corresponding to the array CNVC in common block /UCONC/

Table 7.9--Definitions of the parameters in common block /RKAMS/

Position in common	Variable name	Definition
1-5	XV	Values of independent variable for 5 integration steps
6-85	FV	Values of the derivatives of the 20 dependent variables for 4 integration steps
86-285	YU	Values of the 20 dependent variables for 5 integration steps (in double precision)
286	EPM	The amount by which the independent variable changed during the previous call to SUBROUTINE RKAM1
287	ALPHA	Value of the independent variable at the beginning of the latest integration step
288	MM	Relative integration step number (varies from 1 to 4)

Table 7.10--Definitions of the parameters in common block /CGET/

Position in common	Variable name	Definition
1	ZERO	A great circle distance at sea level corresponding to a central earth angle that is twice the smallest floating point variable that can be stored in one single precision word in the computer being used

Table 7.11--Definitions of the parameters in common block /CRMACH/

Position in common	Variable name	Definition
1	RMACH(1)	Smallest positive magnitude = B**(EMIN-1)
2	RMACH(2)	Largest magnitude = B**EMAX*(1-B**(-T))
3	RMACH(3)	Smallest relative spacing = B**(-T)
4	RMACH(4)	Largest relative spacing = B**(1-T)
5	RMACH(5)	$Log_{10}B = Log_{10}2$

Notes:
1. B = the number base used by the computer (= 2 for most computers)
2. T = the number of bits in the mantissa of a floating point number
3. EMIN = the most negative allowable exponent
4. EMAX = the largest allowable positive exponent

Table 7.12--Definitions of the parameters in common block /CRKTIME/

Position in common	Variable name	Definition
1	IRKTIME	The number of times that SUBROUTINE RKAM has been called (used to compare FTIME or GTIME with to know whether F or G need to be updated)
1	RKTIME	Floating point name of IRKTIME

Table 7.13--Definitions of the parameters in common block /TRLOCAL/

Position in common	Variable name	Definition
1	RSIGN	+1 if next receiver-surface crossing is going up; -1 if going down
2	HOME	.TRUE. if ray is going away from receiver surface; .FALSE. otherwise
3	FDOT	Rate of change of distance of ray above the receiver surface
4	GDOT	Rate of change of distance of ray above the terrain
5	GOLD	Value of G at previous integration step (= distance of ray above terrain)
6	GDOLD	Value of GDOT at previous integration step

Table 7.14--Definitions of the parameters in common block /RK/*

Position in common	Variable name	Definition
1	N	The number of equations being integrated
2	STEP	The initial step in group path in kilometers
3	MODE	Defines type of integration used (same as W41), see Section 5.3.2
4	E1MAX	Maximum allowable single step error (same as W42)
5	E1MIN	Minimum allowable single step error (= W42/W43)
6	E2MAX	Maximum step length (same as W45)
7	E2MIN	Minimum step length (same as W46)
8	FACT	Factor to use to decrease step length (same as W47)
9	RSTART	Nonzero to initialize numerical integration, zero to continue integration

* These parameters are calculated in subroutine READW (some are temporarily reset in subroutine BACKUP) and are used in subroutine RKAM.

Table 7.15--Definitions of the parameters in common block /TRAC/*

Position in common	Variable name	Definition
1	GROUND	.TRUE. if the ray is on the ocean bottom
2	PERIGE	.TRUE. if the ray has just made a perigee
3	THERE	.TRUE. if the ray is at the receiver height
4	MINDIS	.TRUE. if the ray has just made a closest approach to the receiver height
5	NEWRAY	Not used in this version of the program
6	SMT	An estimation of the vertical distance to an apogee or perigee of the ray
7	OSMT	Value of SMT at previous integration step
8-27	ROLD	Value of R(1) (=r in r,θ,ϕ earth-centered spherical polar coordinate system) at previous integration step
28-47	DROLD	Value of dr/dt at previous integration step
48	TOLD	Value of t (= independent variable for numerical integration) at previous integration step
49	ZDOT	dZ/dt (= dF/dt or dG/dt, depending on the situation)
50	D2Z	d^2Z/dt^2 (= d^2F/dt^2 or d^2G/dt^2, depending on the situation)
51	RAD	$(dZ/dt)^2 - 2\ Z\ d^2Z/dt^2$
52	RAD1	$\sqrt{RAD}$

* These parameters are used for communication between subroutines TRACE and BACKUP.

Table 7.16--Definitions of the parameters in common block /FNDER/

Position in common	Variable name	Definition	Value
1-20	GY	Dummy variable to protect the other variables from being klobbered by the CRAY XMP computer	---
21	NZ	Relative position of F in common block /RR/ (or G in common block /GG/)	1
22	NPZR	Relative position of PFR in common block /RR/ (or PGR in common block /GG/)	2
23	NPZRR	Relative position of PFRR in common block /RR/ (or PGRR in common block /GG/)	3
24	NPZRTH	Relative position of PFRTH in common block /RR/ (or PGRTH in common block /GG/)	4
25	NPZRPH	Relative position of PFRPH in common block /RR/ (or PGRPH in common block /GG/)	5
26	NPZTH	Relative position of PFTH in common block /RR/ (or PGTH in common block /GG/)	6
27	NPZPH	Relative position of PFPH in common block /RR/ (or PGPH in common block /GG/)	7
28	NPZTHTH	Relative position of PFTHTH in common block /RR/ (or PGTHTH in common block /GG/)	8
29	NPZPHPH	Relative position of PFPHPH in common block /RR/ (or PGPHPH in common block /GG/)	9
30	NPZTHPH	Relative position of PFTHPH in common block /RR/ (or PGTHPH in common block /GG/)	10
31	NSELECT	Relative position of FSELECT in common block /RR/ (or GSELECT in common block /GG/)	11
32	NTIME	Relative position of FTIME in common block /RR/ (or GTIME in common block /GG/)	12

Table 7.17--Definitions of the parameters in common block /RIN/*

Position in common	Variable name	Definition
1-8	MODRIN	Description of version of DISPER in BCD
9-14	RAYNAME	Hollerith names of the characteristic rays in a birefringent medium (= blank for this version of the program)
15-17	TYPE	= Hollerith 1 or 3 if this version of DISPER includes wind, = Hollerith 2 or 3 if this version of DISPER includes losses
18	SPACE	TRUE, if the ray is in a homogeneous non-dissipative medium (Unconditionally set to FALSE in this version of the program)
19	OMEGMIN	Minimum frequency for nonevanescent propagation (= 0 for this version of the program)
20	OMEGMAX	Maximum frequency for nonevanescent propagation (not applicable for this version of the program, set to 0)
21,22	KAY2	k^2, square of the complex wave number
23,24	H	Hamiltonian (complex)
25,26	PHPT	$\partial H/\partial t$ (complex)
27,28	PHPR	$\partial H/\partial r$ (complex)
29,30	PHPTH	$\partial H/\partial \theta$ (complex)
31,32	PHPPH	$\partial H/\partial \phi$ (complex)
33,34	PHPOM	$\partial H/\partial \omega$ (complex)
35,36	PHPKR	$\partial H/\partial k_r$ (complex)
37,38	PHPKTH	$\partial H/\partial k_\theta$ (complex)
39,40	PHPKPH	$\partial H/\partial k_\phi$ (complex)
41,42	KPHPK	$\vec{k} \cdot \partial H/\partial \vec{k}$ (complex) $= k_r \partial H/\partial k_r + k_\theta \partial H/\partial k_\theta + k_\phi \partial H/\partial k_\phi$
43,44	POLAR	Characteristic transverse polarization of the wave (complex) (= 0 for this version of the program)
45,46	LPOLAR	Characteristic longitudinal polarization of the wave (complex) (= 1 for this version of the program)
47	SGN	= +1 or -1; used for ray tracing in complex space

* These parameters are calculated in subroutine DISPER and used in subroutine HAMLTN.

Note: In some subroutines, the real and imaginary parts of the complex variables have separate names.

Table 7.18--Definitions of the parameters in common block /ERR/

Position in common	Variable name	Definition
1	NERG	Index number for the dependent variable for the integration that gives G
2	NERR	Index number for the dependent variable for the integration that gives $\partial G/\partial r$
3	NERT	Index number for the dependent variable for the integration that gives $\partial G/\partial \theta$
4	NERP	Index number for the dependent variable for the integration that gives $\partial G/\partial \phi$

Table 7.19--Definitions of the parameters in common block /RAYDEV/

Position in common	Variable name	Definition
1	NRYIND	Device unit number for input data
2	NDEVTMP	Device unit number for temporary output and input
3	NFRMAT	Device unit number for secondary input file (not used by ray tracing program)
4	NDEVGRP	Device unit number for graphics output
5	NDEVBIN	Device unit number for binary raypath coordinate output

Table 7.20--Definitions of the parameters in common block /FLGP/

Position in common	Variable name	Definition
1	NSET	Runset number

Table 7.21--Definitions of the parameters in common block /RINPL/

Position in common	Variable name	Definition
1	DISPM	Character string identifier for the dispersion relation model

Table 7.22--Definitions of the parameters in common block /FLG/*

Position in common	Variable	Definition
1	NTYP	Wave polarization indicator (not used in this version of program)
2	NEWWR	Set equal to .TRUE. to tell subroutine RAYPLT there is a new W array
3	NEWWP	Set equal to .TRUE. to tell subroutine PRINTR there is a new W array
4	PENET	Set equal to .TRUE. if the ray left the allowed region of the ocean
5	LINES	Number of lines printed on the current page
6	IHOP	Hop number (at the beginning of each ray, subroutine TRACE sets this parameter to zero so that subroutine RAYPLT will begin a new line in plotting the raypath, and subroutine PRINTR will print column headings and punch a transmitter rayset)
7	HPUNCH	The height to be output on the raysets

* The parameters are used to communicate between various subroutines.

Table 7.23--Definitions of the parameters in common block /HDRC/

Position in common	Variable name	Definition
1	INITID	Character string for user name and phone number identifier for graphics output
2	DAT	Character string for the date of the computer run
3	TOD	Character string for the time of day of the computer run

Table 7.24--Definitions of the parameters in common block /FILEC/

Position in common	Variable name	Definition
1	NPLTDP	Set equal to the device unit number for binary raypath coordinate output (variable in position 5 of common block /RAYDEV/) if binary raypath coordinate output has been requested, set to zero otherwise

Table 7.25--Definitions of the parameters in common block /PLT/*

Position in common	Variable name	Definition
1	XMINO,XL	The x-coordinate of the left side of the plotting area in kilometers
2	XMAXO,XR	The x-coordinate of the right side of the plotting area in kilometers
3	XMINO,YB	The y-coordinate of the bottom of the plotting area in kilometers
4	XMAXO,YT	The y-coordinate of the top of the plotting area in kilometers
5	RESET	Set equal to one whenever the plotting area is changed

* These parameters are used for communication between subroutine RAYPLT and subroutine PLOT.

Table 7.26--Definitions of the parameters in common block /RAYCON/

Position in common	Variable name	Definition
1	MCONP	Set to zero for the raytracing program to indicate that the abscissa in raypath plots is a central-earth angle in radians, set non-zero for the contouring program to indicate that the abscissa in contour plots is a great-circle distance in kilometers

Table 7.27--Definitions of the parameters in common block /ANNCTC/

Position in common	Variable name	Definition
1-8	ANOTES	Character strings to label the ordinate of raypath plots
9-20	HNOTES	Character strings to label the abscissa of raypath plots

Table 7.28--Definitions of the parameters in common block /ANNCTL/

Position in common	Variable name	Definition
1-4	LENA	Lengths of the character strings that label the ordinate of raypath plots
5-7	LENHA	Lengths of the character strings that label the abscissa of raypath plots

Table 7.29--Definitions of the parameters in common block /LABCLT/

Position in common	Variable name	Definition
1	PROJCT	Number that indicates which type of projection is being used for raypath plots
2	THMIN	θ_{min}, minimum value of the abscissa of a raypath plot
3	THMAX	θ_{max}, maximum value of the abscissa of a raypath plot
4	RMIN	r_{min}, minimum value of the ordinate of a raypath plot
5	RMAX	r_{max}, maximum value of the ordinate of a raypath plot

Table 7.30a--Definitions of the parameters in common block /DD/

Position in common	Variable name	Definition
1	IN	Intensity IN = 0 specifies normal intensity IN = 1 specifies high intensity
2	IOR	Orientation IOR = 0 specifies upright orientation IOR = 1 specifies rotated orientation (90° counterclockwise)
3	IT	Italics (Font) IT= 0 specifies non-italic (Roman) symbols IT = 1 specifies italic symbols
4	IS	Symbol size IS = 0 specifies miniature size IS = 1 specifies small size IS = 2 specifies medium size IS = 3 specifies large size
5	IC	Symbol case IC = 0 specifies uppercase IC = 1 specifies lowercase
6	ICC	Character code, 0-63 (R1 format) ICC and IC together specify the symbol plotted
7	IX	X-coordinate, 0-1023
8	IY	Y-coordinate, 0-1023

Table 7.30b--Definitions of the parameters in common block /DDREZ/

Position in common	Variable name	Definition
1	DDHIX	X-coordinate (real), 0.0-1023.0
2	DDHIY	Y-coordinate (real), 0.0-1023.0

Table 7.31--Definitions of the parameters in common block /KNKN/

Position in common	Variable name	Definition
1	KNBP	Number of times SUBROUTINE DDBP was called
2	KNVC	Number of times SUBROUTINE DDVC was called
3	KNDT	Number of times SUBROUTINE DDTEXT was called

Table 7.32--Definitions of the parameters in common block /DDLIM/

Position in common	Variable name	Definition
1	MXIX	Maximum value of IX
2	MXIY	Maximum value of IY
3	MNIX	Minimum value of IX
4	MNIY	Minimum value of IY

Table 7.33--Definitions of the parameters in common block /UU/

Position in common	Variable name	Definition
1-4	MODU	Current-velocity model and parameter identification
1	MODU(1)	Name of current-velocity subroutine
2	MODU(2)	Current-velocity parameter identification number
3	MODU(3)	Name of current-velocity perturbation subroutine
4	MODU(4)	Current-velocity perturbation parameter identification number
5	V	$\|V\|$, current speed in km/s
6	PVT	$\partial\|V\|/\partial t$
7	PVR	$\partial\|V\|/\partial r$
8	PVTH	$\partial\|V\|/\partial\theta$
9	PVPH	$\partial\|V\|/\partial\phi$
10	VR	V_r, upward component of current velocity
11	PVRT	$\partial V_r/\partial t$
12	PVRR	$\partial V_r/\partial r$
13	PVRRTH	$\partial V_r/\partial\theta$
14	PVRPH	$\partial V_r/\partial\phi$
15	VTH	V_θ, southward component of current velocity
16	PVTHT	$\partial V_\theta/\partial t$
17	PVTHR	$\partial V_\theta/\partial r$
18	PVTHTH	$\partial V_\theta/\partial\theta$
19	PVTHPH	$\partial V_\theta/\partial\phi$
20	VPH	V_ϕ, eastward component of current velocity
21	PVPHT	$\partial V_\phi/\partial t$
22	PVPHR	$\partial V_\phi/\partial r$
23	PVPHTH	$\partial V_\phi/\partial\theta$
24	PVPHPH	$\partial V_\phi/\partial\phi$

Table 7.34--Definitions of the parameters in common block /CC/

Position in common	Variable name	Definition common
1-4	MODC	Sound-speed model and parameter identification
1	MODC(1)	Name of sound-speed subroutine
2	MODC(2)	Sound-speed parameter identification number
3	MODC(3)	Name of sound-speed perturbation subroutine
4	MODC(4)	Sound-speed perturbation parameter identification number
5	CS	C^2, square of sound speed in km^2/s^2
6	PCST	$\partial C^2/\partial t$
7	PCSR	$\partial C^2/\partial r$
8	PCSTH	$\partial C^2/\partial \theta$
9	PCSPH	$\partial C^2/\partial \phi$

Table 7.35--Definitions of the parameters in common block /LL/

Position in common	Variable name	Definition
1-4	MODL	Absorption model and parameter identification
1	MODL(1)	Name of absorption subroutine
2	MODL(2)	Absorption parameter identification number
3	MODL(3)	Name of absroption-perturbation subroutine
4	MODL(4)	Absorption-perturbation-parameter identification number
5	ALPHA	α, absorption coefficient in nepers per km

Table 7.36--Definitions of the parameters in common block /SS/

Position in common	Variable name	Definition
1-4	MODSURF	Ocean-surface model and parameter identification
1	MODSURF(1)	Name of ocean-surface subroutine
2	MODSURF(2)	Ocean surface parameter identification number
3	MODSURF(3)	Name of ocean surface perturbation subroutine
4	MODSURF(4)	Ocean surface perturbation parameter identification number
5	U	$u(r,\theta,\phi)$ defined in the same way as $f(r,\theta,\phi)$ in (6.72)-(6.74)
6	PUR	$\partial u/\partial r$
7	PURR	$\partial^2 u/\partial r^2$
8	PURTH	$\partial^2 u/\partial r \partial \theta$
9	PURPH	$\partial^2 u/\partial r \partial \phi$
10	PUTH	$\partial u/\partial \theta$
11	PUPH	$\partial u/\partial \phi$
12	PUTHTH	$\partial^2 u/\partial \theta^2$
13	PUPHPH	$\partial^2 u/\partial \phi^2$
14	PUTHPH	$\partial^2 u/\partial \theta \partial \phi$
15	USELECT	= "SURFACE"
16	UTIME	An integer that is initialized to equal -1 at the beginning of each raypath calculation and is incremented by 1 at each integration step so that it is possible to determine whether the variables in this common block are current

Table 7.37--Definitions of the parameters in common block /RR/

Position in common	Variable name	Definition
1-4	MODREC	Receiver-surface model and parameter identification
1	MODREC(1)	Name of receiver-surface subroutine
2	MODREC(2)	Parameter identification number for receiver-surface model
3	MODREC(3)	Unused now
4	MODREC(4)	Unused now
5	F	$f(r,\theta,\phi)$ defined in (6.72)-(6.74)
6	PFR	$\partial f/\partial r$
7	PFRR	$\partial^2 f/\partial r^2$
8	PFRTH	$\partial^2 f/\partial r \partial \theta$
9	PFRPH	$\partial^2 f/\partial r \partial \phi$
10	PFTH	$\partial f/\partial \theta$
11	PFPH	$\partial f/\partial \phi$
12	PFTHTH	$\partial^2 f/\partial \theta^2$
13	PFPHPH	$\partial^2 f/\partial \phi^2$
14	PFTHPH	$\partial^2 f/\partial \theta \partial \phi$
15	FSELECT	= "RECEIVER"
16	FTIME	An integer that is initialized to equal -1 at the beginning of each raypath calculation and is incremented by 1 at each integration step so that it is possible to determine whether the variables in this common block are current

Table 7.38--Definitions of the parameters in common block /GG/

Position in common	Variable name	Definition
1-4	MODG	Bottom model and parameter identification
1	MODG(1)	Name of terrain subroutine
2	MODG(2)	Bottom-parameter identification number
3	MODG(3)	Name of terrain-perturbation subroutine
4	MODG(4)	Bottom-perturbation parameter identification number
5	G	$g(r,\theta,\phi)$ defined in the same way as $f(r,\theta,\phi)$ in (6.72)-(6.74)
6	PGR	$\partial g/\partial r$
7	PGRR	$\partial^2 g/\partial r^2$
8	PGRTH	$\partial^2 g/\partial r \partial \theta$
9	PGRPH	$\partial^2 g/\partial r \partial \phi$
10	PGTH	$\partial g/\partial \theta$
11	PGPH	$\partial g/\partial \phi$
12	PGTHTH	$\partial^2 g/\partial \theta^2$
13	PGPHPH	$\partial^2 g/\partial \phi^2$
14	PGTHPH	$\partial^2 g/\partial \theta \partial \phi$
15	GSELECT	= "BOTTOM"
16	GTIME	An integer that is initialized to equal -1 at the beginning of each raypath calculation and is incremented by 1 at each integration step so that it is possible to determine whether the variables in this common block are current

REFERENCES

Bennett, J. A. (1967): The calculation of Doppler shifts due to a changing ionosphere. J. Atmos. Terr. Phys., 29, 887-891.

Budden, K. G. (1961): Radio Waves in the Ionosphere. University Press, Cambridge, England.

Budden, K. G. (1972): The theory of coupling of characteristic radio waves in the ionosphere. J. Atmos. Terr. Phys., 34, 1909-1921.

Croft, T. A., and L. Gregory (1963): A fast, versatile ray-tracing program for IBM 7090 digital computers. Rep. SEL-63-107 (TR 82), Contract No. 225(64), Stanford Electronics Laboratories, Stanford, Calif.

Cornyn, J. J. (1973): GRASS: A digital computer ray-tracing and transmission-loss-prediction system. Vol. 1, NRL Report 7621, Naval Research Laboratory, Washington, D.C.

Courant, R., and D. Hilbert (1962): Methods of Mathematical Physics, Vol. II, Partial Differential Equations. Interscience Publishers, John Wiley and Sons, New York.

Dudziak, W. F. (1961): Three-dimensional ray trace computer program for electromagnetic wave propagation studies. RM 61 TMP-32, DASA 1232, G. E. TEMPO, Santa Barbara, Calif.

Felsen, L. B., and N. Marcuvitz (1973): Radiation and Scattering of Waves. Prentice-Hall, Englewood Cliffs, N.J.

Feynman, R. P., and A. R. Hibbs (1965): Quantum Mechanics and Path Integrals. McGraw-Hill, New York.

Garabedian, P. R. (1964): Partial Differential Equations. John Wiley and Sons, New York.

Georges, T. M. (1967): Ionospheric effects of atmospheric waves. ESSA Tech. Rept. IER 57-ITSA 54, Boulder, Colo.

Georges, T. M., and J. J. Stephenson (1968): HF radar signatures of traveling ionospheric irregularities: 3D ray-tracing simulation. Radio Sci., 4, 679-696.

Georges, T. M. (1970): Amplification of ionospheric heating and triggering of 'spread-F' by natural irregularities. J. Geophys. Res., 75, 6436-6438.

Georges, T. M. (1971): A program for calculating three-dimensional acoustic-gravity ray paths in the atmosphere. NOAA Tech. Rep. ERL 212-WPL 16, NOAA Environmental Research Laboratories, Boulder, Colo.

Georges, T. M. (1972): Acoustic ray paths through a model vortex with a viscous core. J. Acoust. Soc. Am., 51, 206-209.

Georges, T. M., and W. H. Beasley (1977): Refraction of infrasound by upper atmospheric winds. J. Acoust. Soc. Am., 61, 28-34.

Georges, T. M., R. M. Jones, and J. P. Riley (1986): Simulating ocean acoustic tomography measurements with Hamiltonian ray tracing. IEEE J. Oceanic Engr., OE-11, 58-71.

Gossard, E. E., and W. H. Hooke (1975): Waves in The Atmosphere. Elsevier, New York.

Haselgrove, J. (1954): Ray theory and a new method for ray tracing. Report of Conference on the Physics of the Ionosphere (London Physical Society), 355-364.

Jones, R. M. (1966): A three-dimensional ray tracing computer program. ESSA Tech. Rep. IER 17-ITSA 17, Boulder, Colo.

Jones, R. M. (1968): Modifications to the three-dimensional ray tracing program described in IER 17-ITSA 17, ESSA Tech. Memo. ERLTM-ITS 134, ESSA Research Laboratories, Institute for Telecommunication Sciences, Boulder, Colo.

Jones, R. M. (1970): Ray theory for lossy media. Radio Sci., 6, 793-801.

Jones, R. M., and J. J. Stephenson (1975): A versatile three-dimensional ray tracing computer program for radio waves in the ionosphere. OT Rep. 75-76, U.S. Department of Commerce/Office of Telecommunications, Boulder, Colo.

Jones, R. M., and T. M. Georges (1976): Infrasound from convective storms. III. Propagation to the ionosphere. J. Acoust. Soc. Am., 59, 765-779.

Jones, R. M. (1981a): Bending of a ray in a random inhomogeneous medium. NOAA Tech. Memo. ERL WPL-77, NOAA Environmental Research Laboratories, Boulder, Colo.

Jones, R. M. (1981b): The frequency shift of a pulse by a time-independent dispersive, lossy medium. NOAA Tech. Memo. ERL WPL-80, NOAA Environmental Research Laboratories, Boulder, Colo.

Jones, R. M. (1982): Algorithms for reflecting rays from general topographic surfaces in a ray-tracing program. NOAA Tech. Memo. ERL WPL-98, NOAA Environmental Research Laboratories, Boulder, Colo., 58 pp.

Jones, R. M., J. P. Riley, and T. M. Georges (1982): A versatile three-dimensional Hamiltonian ray-tracing computer program for acoustic waves in the atmosphere. NOAA Tech. Memo. ERL WPL-103, NOAA Environmental Research Laboratories, Boulder, Colo., 242 pp.

Jones, R. M. (1983): A survey of underwater-acoustic ray-tracing techniques. NOAA Tech. Memo. ERL WPL-111, NOAA Environmental Research Laboratories, Boulder, Colo.

Jones, R. M., T. M. Georges, and J. P. Riley (1984): Modeling acoustic remote sensing in the Florida Straits with ray tracing. IEEE Trans. Geosci. Remote Sens. GE-22, 633-640.

Jones, R. M., T. M. Georges, and J. P. Riley (1986a): Inverting vertical-slice tomography measurements for asymmetric ocean sound-speed profiles. Deep-Sea Research, 33, 601-619.

Jones, R. M., J. P. Riley, and T. M. Georges (1986b): HARPA--A versatile three-dimensional Hamiltonian ray-tracing program for acoustic waves in the atmosphere above irregular terrain. NOAA Rep., Boulder, Colo.

Keller, J. B. (1962): Geometrical theory of diffraction. J. Opt. Soc. Am., 52, 116-130.

Lighthill, M. J. (1965): Group velocity. J. Inst. Math. Appl., 1, 1-28.

Lighthill, M. J. (1978): Waves in Fluids. Cambridge Univ. Press, Sec. 4.5.

Ludwig, D. (1966): Uniform asymptotic expansions at a caustic. Commun. Pure Appl. Math., 19, 215-250.

NOAA (1985): Progress and Plans. Environmental Research Laboratories Report, Bolder, Colo.

NOAA, NASA, and USAF (1976): "U.S. Standard Atmosphere, 1976". U.S. Government Printing Office, Washington, D.C. October 1976, p. 19.

Pedersen, M. A. (1961): Acoustic intensity anomalies introduced by constant velocity gradients. J. Acoust. Soc. Am., 33, 465-474.

Pierce, A. D. (1965): Extension of the method of normal modes to sound propagation in an almost-stratified medium. J. Acoust. Soc. Am., 37, 19-27.

Raspet, R., S. W. Lee, E. Kuester, D. C. Chang, W. F. Richards, R. Gilbert, and N. Bong (1985): Fast-field program for sound propagation in a layered atmosphere above an impedance ground. J. Acoust. Soc. Am., 77, 345-352.

Roberts, B. G., Jr. (1974): Horizontal-gradient acoustical ray-trace program TRIMAIN. NRL Report 7827, Naval Research Laboratory, Washington, DC.

Sears, F. W. (1956): An Introduction to Thermodynamics, the Kinetic Theory of Gases, and Statistical Mechanics. Second Edition, Addison-Wesley, Reading, Mass.

Skretting, A., and C. C. Leroy (1971): Sound attenuation between 200 Hz and 10 kHz. J. Acoust. Soc. Am., Vol. 69, 276-282.

Stephenson, J. J., and T. M. Georges (1969): Computer routines for synthesizing ground-backscatter from three-dimensional raysets. ESSA Tech. Rep. ERL 120-ITS84, NOAA Environmental Research Laboratories, Boulder, Colo.

Tappert, F. D. (1977): The parabolic approximation method. In Wave Propagation in Underwater Acoustics, edited by J. B. Keller and J. S. Papadakis, Springer-Verlag, New York.

Tolstoy, I., and C. S. Clay (1966): Ocean Acoustics, Theory and Experiment in Underwater Sound. McGraw-Hill, N.Y.

Urick, R. J. (1975): Principles of Underwater Sound. McGraw-Hill, N.Y.

Valley, S. L. (1965): Handbook of Geophysics and Space Environments. McGraw-Hill, New York.

Wakeley, J., Jr. (1978): Coherent ray tracing--measured and predicted shallow-water frequency spectrum. J. Acoust. Soc. Am., Vol. 63, 1820-1823.

White, D., and M. Pedersen (1981): Evaluation of shadow-zone fields by uniform asymptotics and complex rays. J. Acoust. Soc. Am., 69, 1029-1059.

APPENDIX A: PRINTOUT AND RAYSET LISTING FOR THE SAMPLE CASE

This appendix contains the printout and rayset listing for the sample case. Users should compare their sample-case output with this printout to be sure they are identical. The meanings of the printed quantities are explained in Sections 2.5.1 and 2.5.2, and the meanings of rayset quantities are listed in Figures 2.24 and 2.25.

***** H A R P O *****
HAMILTONIAN ACOUSTIC RAY-TRACING PROGRAM FOR THE OCEAN

BY
R. M. JONES, J. P. RILEY AND T. M. GEORGES
WAVE PROPAGATION LABORATORY
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
BOULDER, COLORADO 80303

RUN SET NUMBER 1

OCEAN MODEL ID — N01

OCEAN MODEL DESCRIPTION — SAMPLE CASE FOR HARPO DOCUMENTATION REV. 8-19-86

MODEL TYPE	SUBROUTINE NAME	DATA SET ID	DESCRIPTION
DISPERSION RELATION	AWCWL		ACOUSTIC WAVE *** WITH CURRENT *** WITH LOSSES
BACKGROUND CURRENT VELOCITY	VVORTX3	3.00	VORTEX AT LONGITUDE 150 KM E, UMAX= 1.02 M/S, R= 50 KM
CURRENT VELOCITY PERTURBATION	NPCURR	.00	NO CURRENT PERTURBATION
BACKGROUND SOUND SPEED	CTANH	1.00	EL NINO BACKGROUND SOUND-SPEED PROFILE
SOUND SPEED PERTURBATION	CBLOB2	7.00	2% INCREASE IN C-SQUARED AT 150 KM LON., 1 KM DEPTH, 50 KM WIDE
BACKGROUND BOTTOM	GLORENZ	3.00	RIDGE .5 KM HIGH, 2 KM WIDE AT 10 KM N LATITUDE; BASE= -3 KM
BOTTOM PERTURBATION	NPBOTM	.00	NO BOTTOM PERTURBATION
ABSORPTION	SLLOSS	1.00	SKRETTING-LEROY ABSORPTION FORMULA
ABSORPTION PERTURBATION	NPABSR	.00	NO ABSORPTION PERTURBATION
RECEIVER SURFACE	RHORIZ	.00	RECEIVER SURFACE = SPHERE 1 KM BELOW MSL
OCEAN SURFACE	SHORIZ	1.00	OCEAN SURFACE = SPHERE AT MSL
OCEAN SURFACE PERTURBATION	NPSURF	.00	NO OCEAN SURFACE PERTURBATION

INPUT DATA FILE FOR RUN SET NUMBER 1

```
N01-1     SAMPLE CASE FOR HARPO DOCUMENTATION          REV. 8-19-86
  1      6370.            EARTH RADIUS TO MSL, KM (6370.)
  3         2.           TTRANSMITTER HEIGHT ABOVE MSL (T=ABOVE BOTTOM), KM
  4         0.     AN KM  N. TRANSMITTER LATITUDE, KM
  5         0.     AN KM  E. TRANSMITTER LONGITUDE, KM
  7       400.     FQ HZ  INITIAL FREQUENCY, HZ
 11        80.     AN DG  INITIAL AZIMUTH ANGLE, DEG
 15         2.     AN DG  INITIAL ELEVATION ANGLE, DEG
 16        16.     AN DG  FINAL   ELEVATION ANGLE, DEG
 17         2.     AN DG  STEP IN ELEVATION ANGLE, DEG
 19         0.            STOP RAYS THAT STRIKE BOTTOM (1=YES; 0=NO)
 20        -1.            RECEIVER HEIGHT ABOVE MSL, KM
 22        50.            MAXIMUM NUMBER OF HOPS (1.)
 23      1000.            MAXIMUM NUMBER OF STEPS PER HOP (1000.)
 26         5.            MAXIMUM RAY HEIGHT ABOVE MSL, KM
 27        -5.            MINIMUM RAY HEIGHT ABOVE MSL, KM
 28       210.            MAXIMUM RANGE AT MSL, KM
 29         0.            DO: EIGRAY/RNG-TIM/RNG-ELV/NEW-PROJ/RAYTRC/CONT/PROF
 33        20.            MAXIMUM ABSORPTION, DB (999.999)
 42       1.0E-06         MAXIMUM SINGLE-STEP INTEGRATION ERROR (1.0E-4)
 44        .1             INITIAL INTEGRATION STEP SIZE, KM (1.0)
 57         2.            PHASE PATH  (0=NO; 1=INTEGRATE; 2=INTEGRATE/PRINT)
 58         2.            ABSORPTION  (0=NO; 1=INTEGRATE; 2=INTEGRATE/PRINT)
 60         2.            PATH LENGTH (0=NO; 1=INTEGRATE; 2=INTEGRATE/PRINT)
 71        50.            NUMBER OF INTEGRATION STEPS PER PRINT [1.E31]
 72         1.            OUTPUT RAYSETS (1=YES; 0=NO)
 73         0.            DIAGNOSTIC PRINTOUT (1=YES; 0=NO)
 74         0.            PRINT EVERY W(71) RAY STEPS (0=YES; 1=N0)
 75       .15             FULANN PLOT LETTERING; HEIGHT= [0.15 IN]
 76         0.            BINARY RAY OUTPUT (1=YES; 0=NO)
 77        57.            LINES PER PAGE OF PRINTOUT (57.)
 81         4.            RAYPLOT PROJECTION PLANE (4 = VERT. RECTANGULAR)
 82        40.             PLOT-ORDINATE EXPANSION FACTOR [1.]
 83         0.     AN KM   N. LATITUDE OF LEFT PLOT EDGE, KM
 84         0.     AN KM   E. LONGITUDE OF LEFT PLOT EDGE, KM
 85      35.1      AN KM   N. LATITUDE OF RIGHT PLOT EDGE, KM
 86       200.     AN KM   E. LONGITUDE OF RIGHT PLOT EDGE, KM
 87        50.     AN KM   DISTANCE BETWEEN RANGE TICKS, KM (0 = AUTO)
 88        -3.             HEIGHT ABOVE MSL OF BOTTOM OF GRAPH, KM
 89         0.             HEIGHT ABOVE MSL OF TOP OF GRAPH, KM
 96         1.             DISTANCE BETWEEN DEPTH TICKS, KM (0 = AUTO)
100         9.            VVORTX3 MODEL CHECK NUMBER
102         3.             VVORTX3 BACKGROUND CURRENT DATA SET ID
103      1.02     LN  M    MAXIMUM TANGENTIAL CURRENT, M/S
104        50.             RADIUS OF VORTEX CORE, KM
105         0.     AN KM   LATITUDE OF VORTEX CENTER, KM
106       150.     AN KM   LONGITUDE OF VORTEX CENTER, KM
107         1.             VERTICAL HALF-WIDTH OF VORTEX, KM
108        -1.             HEIGHT OF VORTEX CENTER ABOVE MSL, KM
150         7.            CTANH SOUND SPEED MODEL CHECK NUMBER
```

```
152        1.              CTANH BACKGROUND SOUND SPEED DATA SET ID
175        2.             CBLOB2 SOUND SPEED PERTURBATION MODEL CHECK NUMBER
177        7.              CBLOB2 PERTURBATION SOUND SPEED DATA SET ID
178       .02              MAXIMUM FRACTIONAL INCREASE IN C SQUARED
179       -1.              HEIGHT OF MAX EFFECT ABOVE MSL, KM
180        0.     AN KM    LATITUDE OF MAX EFFECT, KM
181      150.     AN KM    LONGITUDE OF MAX EFFECT, KM
182        1.              VERTICAL HALF-WIDTH, KM
183       50.     AN KM    N-S  HALF-WIDTH, KM
184       50.     AN KM    E-W  HALF-WIDTH, KM
275        1.             RHORIZ RECEIVER MODEL CHECK NUMBER
300        4.             GLORENZ BOTTOM MODEL CHECK NUMBER
302        3.              GLORENZ BOTTOM MODEL DATA SET ID
303        .5              HEIGHT OF RIDGE, KM ABOVE BASE
304       10.     AN KM    N. LATITUDE OF RIDGE CENTER, KM
305        2.     AN KM    HALF-WIDTH OF THE RIDGE, KM
306       -3.              HEIGHT ABOVE MSL OF BASE OF RIDGE, KM
350        1.             SHORIZ MODEL CHECK NUMBER
352        1.              SHORIZ OCEAN SURFACE DATA SET ID
353        0.              HEIGHT OF OCEAN SURFACE ABOVE MSL, KM
500        1.             SLLOSS ABSORPTION MODEL CHECK NUMBER
502        1.              SLLOSS ABSORPTION DATA SET ID
503    0.006      AM DB    A COEFFICIENT, DB
504   0.2635      AM DB    B COEFFICIENT, DB
505     1000.     FQ HZ    OMEGA1, HZ
506     1700.     FQ HZ    OMEGA2, HZ
 -1                       DATA SUBSET FOR BACKGROUND CURRENT MODEL
  A  VORTEX AT LONGITUDE 150 KM E, UMAX= 1.02 M/S, R= 50 KM
  0        RETURN TO W-ARRAY DATA SET
 -2                       DATA SUBSET FOR PERTURBATION CURRENT MODEL
  A  NO CURRENT PERTURBATION
  0        RETURN TO W-ARRAY DATA SET
 -3                       DATA SUBSET FOR BACKGROUND SOUND-SPEED MODEL
  A  EL NINO BACKGROUND SOUND-SPEED PROFILE
  3     999.0
   LN  M        LN  M        LN  M
   0.           1532.        0.
   -20.         1531.5       -7.
   -50.         1509.        -20.
   -250.        1503.        -40.
   -450.        1485.        -300.
   -1500.       1485.        -400.
   -3000.       1508.        0.
   999.0
  0       RETURN TO W-ARRAY DATA SET
 -4                       DATA SUBSET FOR SOUND-SPEED PERTURBATION MODEL
  A  2% INCREASE IN C-SQUARED AT 150 KM LON., 1 KM DEPTH, 50 KM WIDE
  0       RETURN TO W-ARRAY DATA SET
 -8                       DATA SUBSET FOR RECEIVER-SURFACE MODEL
  A  RECEIVER SURFACE = SPHERE 1 KM BELOW MSL
  0       RETURN TO W-ARRAY DATA SET
 -9                       DATA SUBSET FOR BACKGROUND BOTTOM MODEL
  A  RIDGE .5 KM HIGH, 2 KM WIDE AT 10 KM N LATITUDE; BASE= -3 KM
  0       RETURN TO W-ARRAY DATA SET
-10                       DATA SUBSET FOR BOTTOM PERTURBATION MODEL
```

```
  A  NO BOTTOM PERTURBATION
  0       RETURN TO W-ARRAY DATA SET
-11                    DATA SUBSET FOR OCEAN SURFACE MODEL
  A  OCEAN SURFACE = SPHERE AT MSL
  0       RETURN TO W-ARRAY DATA SET
-12                    DATA SUBSET FOR OCEAN SURFACE PERTURBATION MODEL
  A  NO OCEAN SURFACE PERTURBATION
  0       RETURN TO W-ARRAY DATA SET
-17                    DATA SUBSET FOR OCEAN ABSORPTION MODEL
  A  SKRETTING-LEROY ABSORPTION FORMULA
  0       RETURN TO W-ARRAY DATA SET
-18                    DATA SUBSET FOR PERTURBATION ABSORPTION MODEL
  A  NO ABSORPTION PERTURBATION
  0       RETURN TO W-ARRAY DATA SET
  0                    **********  END OF RUN SET NUMBER 1  *********
```

INITIAL VALUES FOR THE W ARRAY
ONLY NONZERO VALUES PRINTED — ALL ANGLES IN RADIANS

N	W(N)
1	.637000000000000E+04
3	−.980769230769170E+00
7	.251327412287184E+04
11	.139626340159545E+01
15	.349065850398864E−01
16	.279252680319091E+00
17	.349065850398864E−01
20	−.100000000000000E+01
22	.500000000000000E+02
23	.100000000000000E+04
24	.157079632679490E+01
26	.500000000000000E+01
27	−.500000000000000E+01
28	.210000000000000E+03
33	.200000000000000E+02
41	.300000000000000E+01
42	.999999999999997E−06
43	.500000000000000E+02
44	.100000000000000E+00
45	.100000000000000E+03
46	.100000000000000E−07
47	.500000000000000E+00
57	.200000000000000E+01
58	.200000000000000E+01
60	.200000000000000E+01
71	.500000000000000E+02
72	.100000000000000E+01
75	.150000000000000E+00
77	.570000000000000E+02
81	.400000000000000E+01
82	.400000000000000E+02
85	.551020408163264E−02
86	.313971742543171E−01
87	.784929356357927E−02
88	−.300000000000000E+01
96	.100000000000000E+01
100	.900000000000000E+01
102	.300000000000000E+01
103	.102000000000001E−02
104	.500000000000000E+02
106	.235478806907378E−01
107	.100000000000000E+01
108	−.100000000000000E+01
150	.700000000000000E+01
152	.100000000000000E+01
175	.200000000000000E+01
177	.700000000000000E+01
178	.200000000000000E−01
179	−.100000000000000E+01
181	.235478806907378E−01
182	.100000000000000E+01
183	.784929356357927E−02
184	.784929356357927E−02
275	.100000000000000E+01
300	.400000000000000E+01
302	.300000000000000E+01
303	.500000000000000E+00
304	.156985871271585E−02
305	.313971742543171E−03
306	−.300000000000000E+01
350	.100000000000000E+01
352	.100000000000000E+01
500	.100000000000000E+01
502	.100000000000000E+01
503	.138155106185094E−02
504	.606731174662869E−01
505	.628318530717958E+04
506	.106814150222053E+05

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 2.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
.00E+00	XMTR	-.9808	2.0000	.0000				2.000	.0000	.0000	.0000	.0000
.85E-09	APOGEE	-.7737	2.1992	9.5046	-.018	-.019	1.205	-.014	6.4000	6.4000	.1405	9.5061
.85E-09	WAVE REV	-.7737	2.1992	9.5046	-.018	-.019	1.205	-.014	6.4000	6.4000	.1405	9.5061
-.1E-07		-.9379	2.0238	17.6732	-.018	-.020	.060	-1.916	11.9000	11.9000	.2613	17.6757
-.2E-07	RCVR	-1.0000	1.9582	19.4734	-.018	-.020	-.144	-2.022	13.1129	13.1129	.2879	19.4767
-.1E-07	PERIGEE	-1.3869	1.5043	35.8077	-.017	-.023	-.811	.016	24.1129	24.1129	.5294	35.8134
-.1E-07	WAVE REV	-1.3869	1.5043	35.8077	-.017	-.023	-.811	.016	24.1129	24.1129	.5294	35.8134
-.1E-07		-1.1622	1.5593	47.3932	-.017	-.025	-.433	1.908	31.9129	31.9129	.7007	47.3992
-.2E-07	RCVR	-1.0000	1.5965	52.0417	-.016	-.027	-.255	2.010	35.0447	35.0447	.7694	52.0498
-.1E-07	APOGEE	-.7732	1.7938	62.2163	-.016	-.033	-.089	-.023	41.8947	41.8947	.9199	62.2261
-.1E-07	WAVE REV	-.7732	1.7938	62.2163	-.016	-.033	-.089	-.023	41.8947	41.8947	.9199	62.2261
-.3E-07	RCVR	-1.0000	1.8084	72.3109	-.015	-.042	-.340	-1.990	48.6891	48.6891	1.0691	72.3225
-.3E-07	PERIGEE	-1.4302	1.5166	91.0432	-.009	-.070	-.692	.029	61.2891	61.2891	1.3461	91.0571
-.3E-07	WAVE REV	-1.4302	1.5166	91.0432	-.009	-.070	-.692	.029	61.2891	61.2891	1.3461	91.0571
-.3E-07		-1.2877	1.6800	101.4652	-.004	-.098	-.630	1.332	68.2891	68.2891	1.5001	101.4781
-.3E-07	RCVR	-1.0000	1.9792	112.7595	.006	-.147	-.517	1.418	75.8697	75.8697	1.6671	112.7741
-.3E-07	APOGEE	-.7919	2.1946	126.4921	.024	-.226	-.483	-.038	85.0697	85.0697	1.8701	126.5068
-.3E-07	WAVE REV	-.7919	2.1946	126.4921	.024	-.226	-.483	-.038	85.0697	85.0697	1.8701	126.5068
-.3E-07		-.9275	2.0625	137.2533	.044	-.299	-.595	-1.160	92.2697	92.2697	2.0292	137.2677
-.3E-07	RCVR	-1.0000	1.9907	140.6506	.051	-.324	-.640	-1.281	94.5422	94.5422	2.0794	140.6653
-.3E-07		-1.8317	1.1635	171.7311	.135	-.469	-1.056	-.077	115.3422	115.3422	2.5390	171.7518
-.3E-07	PERIGEE	-1.8317	1.1635	172.3289	.136	-.470	-1.058	.068	115.7422	115.7422	2.5478	172.3494
-.3E-07	WAVE REV	-1.8317	1.1635	172.3289	.136	-.470	-1.058	.068	115.7422	115.7422	2.5478	172.3494
-.5E-07	RCVR	-1.0000	1.9967	194.1240	.197	-.503	-.879	3.128	130.3646	130.3646	2.8702	194.1592
-.4E-07	APOGEE	-.6456	2.3515	203.6491	.223	-.510	-.822	-.031	136.7646	136.7646	3.0112	203.6915
-.4E-07	WAVE REV	-.6456	2.3515	203.6491	.223	-.510	-.822	-.031	136.7646	136.7646	3.0112	203.6915
-.5E-07		-.7780	2.2193	208.7102	.236	-.510	-.883	-2.632	140.1646	140.1646	3.0860	208.7544
-.5E-07	MAX RANG	-.8511	2.1462	210.1960	.240	-.509	-.910	-2.971	141.1646	141.1646	3.1080	210.2419

THIS RAY CALCULATION TOOK 3.124 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 4.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
.00E+00	XMTR	-.9808	2.0000	.0000				4.000	.0000	.0000	.0000	.0000
.16E-07	APOGEE	-.5981	2.3760	8.2406	-.017	-.018	2.622	-.098	5.5500	5.5500	.1220	8.2504
.16E-07	WAVE REV	-.5981	2.3760	8.2406	-.017	-.018	2.622	-.098	5.5500	5.5500	.1220	8.2504
.15E-07		-.6839	2.2867	11.5129	-.016	-.018	1.425	-2.712	7.7500	7.7500	.1704	11.5238
-.4E-08	RCVR	-1.0000	1.9635	16.5446	-.017	-.019	-.141	-4.012	11.1435	11.1435	.2449	16.5649
-.4E-08	PERIGEE	-1.6999	1.2170	31.8384	-.015	-.019	-1.437	.069	21.4435	21.4435	.4712	31.8750
-.4E-08	WAVE REV	-1.6999	1.2170	31.8384	-.015	-.019	-1.437	.069	21.4435	21.4435	.4712	31.8750
.14E-07		-1.3886	1.4488	41.0550	-.014	-.020	-.754	3.453	27.6435	27.6435	.6075	41.0959
-.6E-07	RCVR	-1.0000	1.7344	46.8372	-.015	-.022	-.234	4.001	31.5450	31.5450	.6932	46.8901
-.4E-07	APOGEE	-.5978	1.9213	55.3751	-.015	-.026	.147	-.094	37.2950	37.2950	.8195	55.4384
-.4E-07	WAVE REV	-.5978	1.9213	55.3751	-.015	-.026	.147	-.094	37.2950	37.2950	.8195	55.4384
-.5E-07		-.7986	1.7342	60.7261	-.015	-.029	-.101	-3.572	40.8950	40.8950	.8987	60.7935
-.6E-07	RCVR	-1.0000	1.6081	63.7278	-.014	-.031	-.304	-3.985	42.9199	42.9199	.9432	63.8015
-.2E-07	PERIGEE	-1.7286	1.1645	79.7738	-.012	-.043	-.896	.058	53.7199	53.7199	1.1806	79.8643
-.2E-07	WAVE REV	-1.7286	1.1645	79.7738	-.012	-.043	-.896	.058	53.7199	53.7199	1.1806	79.8643
-.9E-08		-1.3278	1.6191	91.0825	-.008	-.059	-.628	3.420	61.3199	61.3199	1.3479	91.1790
-.6E-07	RCVR	-1.0000	1.9593	96.3209	-.006	-.074	-.445	3.618	64.8465	64.8465	1.4255	96.4267
-.5E-07	APOGEE	-.5997	2.3730	105.5524	-.001	-.106	-.268	-.032	71.0465	71.0465	1.5621	105.6674
-.5E-07	WAVE REV	-.5997	2.3730	105.5524	-.001	-.106	-.268	-.032	71.0465	71.0465	1.5621	105.6674
-.5E-07		-.7378	2.2395	110.3261	.003	-.125	-.370	-2.873	74.2465	74.2465	1.6327	110.4432
-.6E-07	RCVR	-1.0000	1.9806	114.9018	.007	-.148	-.526	-3.476	77.3190	77.3190	1.7004	115.0258
-.2E-07	PERIGEE	-2.0665	.9238	138.7830	.042	-.232	-1.072	.018	93.3190	93.3190	2.0538	138.9293
-.2E-07	WAVE REV	-2.0665	.9238	138.7830	.042	-.232	-1.072	.018	93.3190	93.3190	2.0538	138.9293
-.2E-07		-1.7343	1.2583	150.7473	.062	-.260	-.964	2.799	101.3190	101.3190	2.2307	150.8958
-.3E-07	RCVR	-1.0000	1.9945	165.2653	.091	-.346	-.750	2.658	111.0446	111.0446	2.4455	165.4295
-.3E-07	APOGEE	-.6490	2.3465	176.1659	.116	-.403	-.684	-.067	118.3446	118.3446	2.6068	176.3358
-.3E-07	WAVE REV	-.6490	2.3465	176.1659	.116	-.403	-.684	-.067	118.3446	118.3446	2.6068	176.3358
-.3E-07		-.7340	2.2618	180.4961	.127	-.419	-.733	-2.007	121.2446	121.2446	2.6708	180.6667
-.3E-07	RCVR	-1.0000	1.9962	186.4722	.142	-.439	-.845	-2.819	125.2554	125.2553	2.7592	186.6482
-.1E-07	PERIGEE	-1.6644	1.3327	205.2309	.188	-.458	-1.114	.087	137.8554	137.8553	3.0367	205.4174
-.1E-07	WAVE REV	-1.6644	1.3327	205.2309	.188	-.458	-1.114	.087	137.8554	137.8553	3.0367	205.4174
-.1E-07	MAX RANG	-1.5734	1.4239	210.5919	.200	-.455	-1.108	1.823	141.4554	141.4553	3.1159	210.7782

THIS RAY CALCULATION TOOK 2.467 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 6.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
.00E+00	XMTR	-.9808	2.0000	.0000				6.000	.0000	.0000	.0000	.0000
.64E-07	APOGEE	-.4612	2.5139	7.2720	-.016	-.016	4.055	-.170	4.9000	4.9000	.1078	7.2936
.64E-07	WAVE REV	-.4612	2.5139	7.2720	-.016	-.016	4.055	-.170	4.9000	4.9000	.1078	7.2936
.65E-07		-.5558	2.4167	9.8061	-.015	-.016	2.438	-3.882	6.6000	6.6000	.1453	9.8297
-.3E-07	RCVR	-1.0000	1.9667	14.5209	-.016	-.017	-.141	-6.008	9.7858	9.7858	.2153	14.5651
-.2E-07	PERIGEE	-2.0399	.8848	30.2627	-.013	-.016	-2.141	.056	20.3858	20.3858	.4486	30.3459
-.2E-07	WAVE REV	-2.0399	.8848	30.2627	-.013	-.016	-2.141	.056	20.3858	20.3858	.4486	30.3459
-.3E-07		-1.9737	.9313	33.8448	-.012	-.016	-1.833	2.055	22.7858	22.7858	.5016	33.9277
-.6E-07	RCVR	-1.0000	1.7566	45.8199	-.012	-.018	-.230	5.996	30.8620	30.8620	.6792	45.9421
.28E-07	APOGEE	-.4609	2.1032	53.2398	-.013	-.021	.320	-.089	35.8620	35.8620	.7892	53.3846
.28E-07	WAVE REV	-.4609	2.1032	53.2398	-.013	-.021	.320	-.089	35.8620	35.8620	.7892	53.3846
.30E-07		-.5837	1.9240	56.2203	-.013	-.023	.152	-4.288	37.8620	37.8620	.8333	56.3680
-.6E-07	RCVR	-1.0000	1.5297	60.5599	-.013	-.025	-.291	-5.983	40.7946	40.7946	.8977	60.7272
-.5E-07	PERIGEE	-2.0686	.7995	76.9038	-.010	-.033	-1.156	.085	51.7946	51.7946	1.1399	77.1109
-.5E-07	WAVE REV	-2.0686	.7995	76.9038	-.010	-.033	-1.156	.085	51.7946	51.7946	1.1399	77.1109
-.5E-07		-1.9934	.9072	80.7867	-.009	-.034	-1.082	2.116	54.3946	54.3946	1.1973	80.9934
-.7E-07	RCVR	-1.0000	1.9531	93.4379	-.006	-.055	-.432	5.593	62.9145	62.9145	1.3849	93.6828
-.5E-09	APOGEE	-.4648	2.5026	101.2476	-.003	-.078	-.163	-.072	68.1645	68.1645	1.5007	101.5133
-.5E-09	WAVE REV	-.4648	2.5026	101.2476	-.003	-.078	-.163	-.072	68.1645	68.1645	1.5007	101.5133
.66E-09		-.5476	2.4231	103.7882	-.002	-.086	-.228	-3.460	69.8645	69.8645	1.5382	104.0555
-.7E-07	RCVR	-1.0000	1.9762	109.0494	.001	-.108	-.501	-5.472	73.4050	73.4050	1.6163	109.3358
-.6E-07	PERIGEE	-2.3504	.6375	130.5265	.024	-.156	-1.188	.209	87.8050	87.8050	1.9345	130.8599
-.6E-07	WAVE REV	-2.3504	.6375	130.5265	.024	-.156	-1.188	.209	87.8050	87.8050	1.9345	130.8599
-.6E-07		-2.3469	.6412	131.1263	.025	-.156	-1.187	.467	88.2050	88.2050	1.9434	131.4594
-.2E-07	RCVR	-1.0000	1.9929	153.0604	.052	-.242	-.696	4.301	102.8929	102.8929	2.2682	153.4338
.17E-07	APOGEE	-.5077	2.4864	162.1001	.069	-.302	-.562	-.053	108.9429	108.9429	2.4021	162.4884
.17E-07	WAVE REV	-.5077	2.4864	162.1001	.069	-.302	-.562	-.053	108.9429	108.9429	2.4021	162.4884
.19E-07		-.5695	2.4249	164.6463	.074	-.316	-.597	-2.599	110.6429	110.6429	2.4397	165.0353
-.2E-08	RCVR	-1.0000	1.9950	170.9365	.087	-.352	-.775	-4.402	114.8611	114.8611	2.5329	171.3399
-.2E-07	PERIGEE	-2.1120	.8844	191.2150	.132	-.390	-1.199	.188	128.4611	128.4611	2.8332	191.6505
-.2E-07	WAVE REV	-2.1120	.8844	191.2150	.132	-.390	-1.199	.188	128.4611	128.4611	2.8332	191.6505
-.2E-07		-2.0668	.9298	194.2054	.138	-.388	-1.194	1.547	130.4611	130.4611	2.8774	194.6404
-.5E-07	RCVR	-1.0000	1.9972	208.7795	.168	-.388	-.944	5.537	140.2669	140.2669	3.0934	209.2537
-.5E-07	MAX RANG	-.8578	2.1395	210.2600	.171	-.389	-.912	5.404	141.2669	141.2669	3.1154	210.7408

THIS RAY CALCULATION TOOK 2.942 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 8.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
-.7E-14	XMTR	-.9808	2.0000	.0000				8.000	.0000	.0000	.0000	.0000
.13E-07	APOGEE	-.3379	2.6378	6.6740	-.015	-.015	5.473	-.090	4.5000	4.5000	.0992	6.7104
.13E-07	WAVE REV	-.3379	2.6378	6.6740	-.015	-.015	5.473	-.090	4.5000	4.5000	.0992	6.7104
.15E-07		-.3648	2.6099	7.7979	-.014	-.015	4.482	-2.630	5.2500	5.2500	.1158	7.8347
.30E-07	RCVR	-1.0000	1.9682	13.4059	-.015	-.016	-.142	-8.006	9.0405	9.0405	.1993	13.4802
.78E-07	PERIGEE	-2.4608	.4591	31.2440	-.010	-.013	-2.853	.258	21.0405	21.0405	.4641	31.3913
.78E-07	WAVE REV	-2.4608	.4591	31.2440	-.010	-.013	-2.853	.258	21.0405	21.0405	.4641	31.3913
.75E-07		-2.3208	.5684	36.0378	-.009	-.012	-2.292	3.086	24.2405	24.2405	.5349	36.1858
-.6E-07	RCVR	-1.0000	1.7001	48.2230	-.009	-.014	-.240	7.989	32.4673	32.4673	.7161	48.4435
-.5E-07	APOGEE	-.3378	2.1865	55.0449	-.010	-.018	.422	-.087	37.0673	37.0673	.8175	55.3032
-.5E-07	WAVE REV	-.3378	2.1865	55.0449	-.010	-.018	.422	-.087	37.0673	37.0673	.8175	55.3032
-.5E-07		-.3888	2.1152	56.6176	-.010	-.018	.344	-3.568	38.1173	38.1173	.8408	56.8768
-.3E-07	RCVR	-1.0000	1.5573	61.7971	-.010	-.022	-.296	-7.975	41.6204	41.6204	.9179	62.0931
.20E-07	PERIGEE	-2.4981	.3966	79.9390	-.007	-.026	-1.447	.143	53.8204	53.8204	1.1872	80.3095
.20E-07	WAVE REV	-2.4981	.3966	79.9390	-.007	-.026	-1.447	.143	53.8204	53.8204	1.1872	80.3095
.18E-07		-2.3999	.5206	84.1373	-.006	-.026	-1.345	2.527	56.6204	56.6204	1.2493	84.5076
-.7E-07	RCVR	-1.0000	1.9625	98.0715	-.004	-.048	-.452	7.435	66.0059	66.0059	1.4563	98.5132
-.6E-07	APOGEE	-.3456	2.6267	105.2099	-.002	-.072	-.127	-.050	70.8059	70.8059	1.5624	105.6865
-.6E-07	WAVE REV	-.3456	2.6267	105.2099	-.002	-.072	-.127	-.050	70.8059	70.8059	1.5624	105.6865
-.6E-07		-.3876	2.5863	106.7110	-.001	-.076	-.161	-3.102	71.8059	71.8059	1.5846	107.1882
-.5E-07	RCVR	-1.0000	1.9788	112.3615	.002	-.102	-.515	-7.311	75.6110	75.6110	1.6686	112.8725
-.4E-08	PERIGEE	-2.7759	.2131	134.1448	.024	-.134	-1.370	.298	90.2110	90.2110	1.9919	134.7405
-.4E-08	WAVE REV	-2.7759	.2131	134.1448	.024	-.134	-1.370	.298	90.2110	90.2110	1.9919	134.7405
-.5E-08		-2.7513	.2382	135.9504	.025	-.133	-1.358	1.260	91.4110	91.4110	2.0185	136.5455
-.3E-07	RCVR	-1.0000	1.9933	155.7112	.045	-.192	-.707	6.434	104.6557	104.6557	2.3118	156.3860
-.3E-07	APOGEE	-.3717	2.6225	163.4742	.056	-.242	-.522	-.149	109.8557	109.8557	2.4270	164.1784
-.3E-07	WAVE REV	-.3717	2.6225	163.4742	.056	-.242	-.522	-.149	109.8557	109.8557	2.4270	164.1784
-.2E-07		-.4361	2.5584	165.4268	.059	-.251	-.555	-3.517	111.1557	111.1557	2.4559	166.1322
-.2E-07	RCVR	-1.0000	1.9950	171.0190	.069	-.283	-.776	-6.530	114.9135	114.9135	2.5390	171.7528
.30E-08	PERIGEE	-2.5674	.4291	191.5992	.105	-.307	-1.336	.195	128.7135	128.7135	2.8443	192.4012
.30E-08	WAVE REV	-2.5674	.4291	191.5992	.105	-.307	-1.336	.195	128.7135	128.7135	2.8443	192.4012
.28E-08		-2.5636	.4329	192.1999	.106	-.306	-1.336	.521	129.1135	129.1135	2.8531	193.0017
-.1E-06	MAX RANG	-1.0281	1.9692	210.0554	.134	-.300	-.958	7.721	141.1135	141.1135	3.1182	210.9320

THIS RAY CALCULATION TOOK 3.476 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 10.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
-.7E-14	XMTR	-.9808	2.0000	.0000				10.000	.0000	.0000	.0000	.0000
.36E-07		-.2056	2.7708	5.9196	-.014	-.014	7.435	2.972	4.0000	4.0000	.0883	5.9749
.33E-07	APOGEE	-.1048	2.8679	9.6483	-.012	-.013	5.145	-.028	6.4750	6.4750	.1435	9.7053
.33E-07	WAVE REV	-.1048	2.8679	9.6483	-.012	-.013	5.145	-.028	6.4750	6.4750	.1435	9.7053
.33E-07		-.1089	2.8631	10.3268	-.011	-.012	4.780	-.652	6.9250	6.9250	.1535	10.3838
.37E-07		-.3567	2.6089	15.2873	-.011	-.012	2.269	-6.411	10.2250	10.2250	.2269	15.3521
.56E-07	RCVR	-1.0000	1.9585	19.3456	-.012	-.014	-.144	-10.005	12.9848	12.9848	.2877	19.4613
.15E-06	BOTM REF	-2.8979	.0000	34.8791	-.009	.232	-3.304	3.400	23.4576	23.4576	.5192	35.1183
.13E-06		-2.3562	.4852	40.8644	-.043	.197	-2.112	6.941	27.4576	27.4576	.6080	41.1266
-.2E-07	RCVR	-1.0000	1.6789	49.3788	-.080	.159	-.244	10.224	33.2447	33.2447	.7354	49.7475
.30E-07		-.1682	2.3537	55.6707	-.100	.136	.586	3.262	37.4947	37.4947	.8293	56.0996
.23E-07	APOGEE	-.0756	2.4244	58.2324	-.106	.129	.629	-.035	39.1947	39.1947	.8672	58.6633
.23E-07	WAVE REV	-.0756	2.4244	58.2324	-.106	.129	.629	-.035	39.1947	39.1947	.8672	58.6633
.23E-07		-.0757	2.4243	58.2890	-.106	.129	.627	-.194	39.2322	39.2322	.8681	58.7199
.34E-07		-.1931	2.3371	61.1890	-.113	.121	.462	-3.552	41.1572	41.1572	.9110	61.6225
.36E-07	RCVR	-1.0000	1.6805	67.0788	-.126	.105	-.318	-10.188	45.1390	45.1391	.9989	67.5716
.15E-06	BOTM REF	-2.8954	.0000	81.1497	-.145	-.210	-1.717	3.021	54.6401	54.6401	1.2089	81.7738
.13E-06		-2.3276	.6044	87.7358	-.129	-.195	-1.274	6.794	59.0401	59.0401	1.3066	88.3829
-.7E-07	RCVR	-1.0000	1.9576	96.4893	-.113	-.197	-.445	9.505	64.9763	64.9763	1.4374	97.2352
-.2E-07		-.2083	2.7599	102.7961	-.102	-.206	-.032	2.790	69.2263	69.2263	1.5315	103.5961
-.3E-07	APOGEE	-.0990	2.8741	106.9837	-.094	-.208	-.009	-.028	72.0013	72.0013	1.5934	107.7855
-.3E-07	WAVE REV	-.0990	2.8741	106.9837	-.094	-.208	-.009	-.028	72.0013	72.0013	1.5934	107.7855
-.3E-07		-.0995	2.8739	107.2103	-.094	-.208	-.011	-.224	72.1513	72.1513	1.5967	108.0120
-.4E-07		-.2706	2.7074	112.2643	-.085	-.213	-.142	-3.946	75.5013	75.5013	1.6715	113.0695
-.7E-08	RCVR	-1.0000	1.9817	117.6220	-.076	-.234	-.538	-9.212	79.1147	79.1147	1.7515	118.4782
.92E-07	BOTM REF	-2.9882	.0000	132.2094	-.048	-.265	-1.465	4.554	88.9443	88.9443	1.9691	133.2007
.14E-06		-2.0084	.9821	140.5627	-.032	-.259	-1.051	8.423	94.5443	94.5443	2.0934	141.6097
.23E-07	RCVR	-1.0000	1.9918	147.1174	-.019	-.295	-.669	8.695	98.9793	98.9793	2.1914	148.2400
.40E-07		-.2199	2.7730	153.7493	-.005	-.332	-.408	2.573	103.4293	103.4293	2.2902	154.9216
.36E-07	APOGEE	-.0959	2.8977	158.6998	.005	-.344	-.394	-.027	106.7043	106.7043	2.3634	159.8741
.36E-07	WAVE REV	-.0959	2.8977	158.6998	.005	-.344	-.394	-.027	106.7043	106.7043	2.3634	159.8741
.36E-07		-.0960	2.8976	158.7376	.006	-.344	-.395	-.057	106.7293	106.7293	2.3640	159.9119
.43E-07		-.2285	2.7657	163.7248	.017	-.355	-.473	-2.749	110.0293	110.0293	2.4377	164.9012
.58E-07	RCVR	-1.0000	1.9949	170.0700	.031	-.383	-.771	-8.865	114.2929	114.2929	2.5323	171.2967
.15E-06	BOTM REF	-2.9961	.0000	186.1151	.068	-.394	-1.458	3.353	125.0875	125.0875	2.7713	187.4683
.14E-06		-2.3201	.6764	193.2994	.083	-.382	-1.266	7.320	129.8875	129.8875	2.8780	194.6828
-.9E-07	RCVR	-1.0000	1.9969	201.7061	.100	-.384	-.913	9.692	135.5923	135.5923	3.0038	203.1913
-.5E-07		-.2011	2.7960	208.0081	.113	-.386	-.721	2.795	139.8423	139.8423	3.0978	209.5489
-.6E-07	MAX RANG	-.1257	2.8715	210.0424	.117	-.384	-.711	1.425	141.1923	141.1923	3.1278	211.5846

THIS RAY CALCULATION TOOK 5.659 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 12.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
-.7E-14	XMTR	-.9808	2.0000	.0000				12.000	.0000	.0000	.0000	.0000
.65E-07		-.1252	2.8521	4.8540	-.014	-.014	9.975	6.814	3.3000	3.3000	.0729	4.9310
.12E-06	APOGEE	-.0412	2.9353	5.7848	-.013	-.013	9.200	-.152	3.9188	3.9188	.0867	5.8661
.12E-06	WAVE REV	-.0412	2.9353	5.7848	-.013	-.013	9.200	-.152	3.9188	3.9188	.0867	5.8661
.12E-06		-.0446	2.9318	5.9175	-.013	-.013	8.964	-2.724	4.0063	4.0063	.0887	5.9989
.17E-06		-.2287	2.7462	7.5281	-.012	-.012	5.672	-7.461	5.0813	5.0813	.1126	7.6201
.46E-07	RCVR	-1.0000	1.9705	11.6447	-.013	-.014	-.147	-12.004	7.8897	7.8897	.1746	11.8093
.22E-06	BOTM REF	-2.9508	.0000	22.7433	-.010	.193	-5.054	7.184	15.4322	15.4322	.3412	23.0790
.19E-06		-1.7328	1.1932	30.1346	-.058	.144	-1.565	11.426	20.4322	20.4322	.4519	30.5691
-.7E-07	RCVR	-1.0000	1.9079	33.5994	-.075	.127	-.184	12.169	22.8150	22.8150	.5042	34.1099
.33E-07		-.1048	2.7643	38.5988	-.094	.107	1.127	6.968	26.2150	26.2150	.5794	39.1911
.71E-07	APOGEE	-.0394	2.8222	39.3422	-.096	.105	1.194	-.089	26.7087	26.7088	.5904	39.9378
.71E-07	WAVE REV	-.0394	2.8222	39.3422	-.096	.105	1.194	-.089	26.7087	26.7088	.5904	39.9378
.89E-07		-.0481	2.8112	39.5601	-.096	.104	1.173	-4.292	26.8525	26.8525	.5936	40.1559
.13E-06		-.2570	2.5826	41.2430	-.101	.099	.820	-8.014	27.9775	27.9775	.6187	41.8519
.64E-07	RCVR	-1.0000	1.7803	45.0772	-.111	.088	-.227	-12.165	30.5973	30.5973	.6764	45.7579
.26E-06	BOTM REF	-2.5790	.0000	53.2686	-.127	3.158	-1.959	11.314	36.1914	36.1914	.7997	54.0993
.46E-06		-1.1398	1.3625	59.5115	-.460	2.824	-.421	14.052	40.4914	40.4914	.8946	60.5132
.44E-06	RCVR	-1.0000	1.5006	60.0692	-.486	2.797	-.288	14.060	40.8789	40.8790	.9031	61.0887
.65E-06		-.1205	2.3975	64.0004	-.660	2.621	.482	10.070	43.5789	43.5790	.9627	65.1224
.61E-06		-.0188	2.5081	64.7692	-.691	2.590	.560	1.791	44.0914	44.0915	.9742	65.8995
.61E-06	APOGEE	-.0153	2.5149	65.0273	-.701	2.579	.558	-.032	44.2602	44.2602	.9780	66.1579
.61E-06	WAVE REV	-.0153	2.5149	65.0273	-.701	2.579	.558	-.032	44.2602	44.2602	.9780	66.1579
.61E-06		-.0215	2.5130	65.3427	-.714	2.567	.547	-2.690	44.4664	44.4665	.9827	66.4737
.70E-06		-.1440	2.4026	66.1744	-.746	2.534	.427	-10.169	45.0227	45.0227	.9951	67.3159
.47E-06		-.9233	1.6847	69.6713	-.875	2.403	-.266	-14.021	47.4227	47.4227	1.0481	70.9025
.48E-06	RCVR	-1.0000	1.6137	69.9780	-.885	2.392	-.330	-14.041	47.6356	47.6357	1.0528	71.2189
.34E-06	BOTM REF	-2.7472	.0000	77.6090	-1.122	-.362	-1.653	8.925	52.8837	52.8837	1.1686	79.0521
.49E-06		-1.4621	1.3883	84.3663	-1.094	-.338	-.706	12.251	57.4837	57.4837	1.2703	85.9295
-.3E-07	RCVR	-1.0000	1.8714	86.4716	-1.086	-.335	-.402	12.427	58.9327	58.9327	1.3021	88.0845
.12E-06		-.0930	2.8136	91.3049	-1.069	-.328	.146	7.760	62.2202	62.2202	1.3749	93.0040
.18E-06	APOGEE	-.0340	2.8762	91.9411	-1.067	-.327	.177	-.165	62.6420	62.6421	1.3843	93.6435
.18E-06	WAVE REV	-.0340	2.8762	91.9411	-1.067	-.327	.177	-.165	62.6420	62.6421	1.3843	93.6435
.21E-06		-.0448	2.8667	92.1690	-1.067	-.326	.167	-5.051	62.7920	62.7921	1.3877	93.8717
.29E-06		-.2739	2.6460	93.8128	-1.061	-.323	.010	-8.877	63.8920	63.8921	1.4122	95.5315
.29E-08	RCVR	-1.0000	1.9346	97.4114	-1.050	-.323	-.449	-12.356	66.3503	66.3503	1.4665	99.2029
-.5E-07	BOTM REF	-2.9598	.0000	107.4338	-1.020	-.497	-1.539	8.489	73.1776	73.1776	1.6175	109.4142
.18E-07		-1.6122	1.3583	114.8063	-.989	-.476	-.831	11.745	78.1776	78.1776	1.7282	116.9076
-.4E-06	RCVR	-1.0000	1.9735	117.7288	-.977	-.482	-.539	11.826	80.1784	80.1784	1.7724	119.8930
-.3E-06		-.0952	2.8826	122.7290	-.958	-.491	-.139	7.698	83.5659	83.5659	1.8475	124.9759
-.2E-06	APOGEE	-.0340	2.9443	123.3849	-.955	-.490	-.115	-.161	84.0003	84.0003	1.8573	125.6353
-.2E-06	WAVE REV	-.0340	2.9443	123.3849	-.955	-.490	-.115	-.161	84.0003	84.0003	1.8573	125.6353
-.2E-06		-.0431	2.9354	123.5942	-.954	-.490	-.121	-4.673	84.1378	84.1378	1.8604	125.8448
-.2E-06		-.2641	2.7155	125.2234	-.948	-.490	-.235	-8.485	85.2253	85.2253	1.8847	127.4891
-.3E-06	RCVR	-1.0000	1.9819	129.0483	-.934	-.502	-.589	-11.733	87.8248	87.8248	1.9422	131.3844
-.2E-06	BOTM REF	-2.9865	.0000	139.2201	-.896	-.541	-1.452	9.044	94.7420	94.7420	2.0954	141.7468
-.2E-06		-1.0505	1.9391	149.2267	-.859	-.545	-.698	11.551	101.5420	101.5420	2.2461	151.9372
-.3E-06	RCVR	-1.0000	1.9896	149.4738	-.858	-.546	-.680	11.543	101.7106	101.7106	2.2498	152.1894

-.2E-06		-.1072	2.8836	154.4643	-.839	-.565	-.371	7.760	105.0856	105.0856	2.3248	157.2604
-.1E-06	APOGEE	-.0338	2.9571	155.2048	-.837	-.565	-.348	-.071	105.5762	105.5762	2.3358	158.0053
-.1E-06	WAVE REV	-.0338	2.9571	155.2048	-.837	-.565	-.348	-.071	105.5762	105.5762	2.3358	158.0053
-.9E-07		-.0398	2.9512	155.3763	-.836	-.565	-.352	-3.853	105.6887	105.6887	2.3383	158.1769
-.4E-07		-.2519	2.7394	156.9699	-.830	-.566	-.440	-8.306	106.7512	106.7512	2.3621	159.7848
-.1E-06	RCVR	-1.0000	1.9920	160.9212	-.816	-.582	-.731	-11.578	109.4330	109.4331	2.4215	163.8067
-.8E-07	BOTM REF	-2.9935	.0000	171.3287	-.778	-.596	-1.444	8.749	116.5054	116.5054	2.5782	174.4020
-.1E-06		-1.1889	1.8057	180.7432	-.746	-.588	-.879	11.949	122.9054	122.9054	2.7199	183.9868
-.4E-06	RCVR	-1.0000	1.9946	181.6361	-.743	-.591	-.823	11.929	123.5170	123.5170	2.7334	184.8993
-.3E-06		-.1074	2.8877	186.5385	-.727	-.597	-.571	7.638	126.8420	126.8420	2.8070	189.8840
-.2E-06	APOGEE	-.0350	2.9601	187.2827	-.724	-.596	-.553	-.091	127.3358	127.3358	2.8181	190.6324
-.2E-06	WAVE REV	-.0350	2.9601	187.2827	-.724	-.596	-.553	-.091	127.3358	127.3358	2.8181	190.6324
-.2E-06		-.0398	2.9554	187.4350	-.724	-.596	-.555	-3.432	127.4358	127.4358	2.8204	190.7847
-.2E-06		-.2454	2.7499	189.0078	-.719	-.594	-.627	-8.296	128.4858	128.4858	2.8438	192.3712
-.4E-06	RCVR	-1.0000	1.9956	192.8983	-.706	-.597	-.873	-12.030	131.1364	131.1364	2.9024	196.3350
-.5E-06	BOTM REF	-2.9962	.0000	203.5948	-.673	-.588	-1.483	7.919	138.4099	138.4099	3.0633	207.2161
-.5E-06		-1.9368	1.0597	209.8062	-.655	-.572	-1.205	11.338	142.6099	142.6099	3.1564	213.5160
-.5E-06	MAX RANG	-1.8778	1.1188	210.0987	-.655	-.572	-1.190	11.474	142.8099	142.8099	3.1608	213.8144

THIS RAY CALCULATION TOOK 8.958 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 14.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
-.7E-14	XMTR	-.9808	2.0000	.0000				14.000	.0000	.0000	.0000	.0000
.18E-06		-.1184	2.8596	3.9008	-.014	-.014	12.450	9.917	2.6750	2.6750	.0591	3.9959
.96E-07		-.0202	2.9572	4.6611	-.013	-.013	11.624	1.442	3.1813	3.1813	.0704	4.7634
.96E-07	APOGEE	-.0187	2.9586	4.7902	-.013	-.013	11.335	-.054	3.2656	3.2656	.0723	4.8926
.96E-07	WAVE REV	-.0187	2.9586	4.7902	-.013	-.013	11.335	-.054	3.2656	3.2656	.0723	4.8926
.11E-06		-.0490	2.9279	5.2560	-.012	-.013	10.029	-8.283	3.5719	3.5719	.0792	5.3598
.28E-06		-.3004	2.6753	6.6651	-.012	-.012	5.799	-11.090	4.5219	4.5219	.1004	6.7911
.17E-07	RCVR	-1.0000	1.9727	9.6474	-.013	-.014	-.158	-14.003	6.5782	6.5782	.1457	9.8545
-.1E-06	BOTM REF	-2.9600	.0000	18.5493	-.010	.206	-6.176	10.186	12.6798	12.6798	.2804	18.9691
-.1E-06		-1.7236	1.2230	24.4126	-.061	.156	-1.853	13.542	16.6798	16.6798	.3690	24.9601
.74E-07	RCVR	-1.0000	1.9375	27.3220	-.079	.137	-.163	14.155	18.6970	18.6970	.4133	27.9576
.27E-06		-.1226	2.7988	31.2187	-.098	.117	1.434	10.151	21.3720	21.3720	.4724	31.9527
.20E-06		-.0182	2.8993	31.9974	-.102	.114	1.579	1.947	21.8908	21.8908	.4840	32.7392
.20E-06	APOGEE	-.0128	2.9026	32.4041	-.103	.112	1.565	-.017	22.1564	22.1564	.4900	33.1460
.20E-06	WAVE REV	-.0128	2.9026	32.4041	-.103	.112	1.565	-.017	22.1564	22.1564	.4900	33.1460
.20E-06		-.0144	2.8997	32.6338	-.104	.111	1.550	-.844	22.3064	22.3064	.4934	33.3757
.18E-06		-.0793	2.8309	33.3339	-.106	.109	1.399	-9.868	22.7689	22.7689	.5038	34.0799
.24E-06		-.4217	2.4771	35.1051	-.113	.102	.755	-12.532	23.9689	23.9689	.5305	35.8841
.48E-07	RCVR	-1.0000	1.8806	37.4756	-.120	.094	-.198	-14.153	25.6089	25.6089	.5665	38.3239
-.1E-06	BOTM REF	-2.7790	.0000	45.2665	-.138	2.653	-2.479	12.635	30.9631	30.9631	.6847	46.3144
-.4E-08		-1.6227	1.0820	49.9072	-.386	2.405	-.962	15.204	34.1631	34.1631	.7554	51.1002
.49E-07	RCVR	-1.0000	1.6618	52.1548	-.491	2.300	-.256	15.588	35.7338	35.7338	.7899	53.4340
.40E-06		-.0905	2.4992	55.7357	-.640	2.149	.665	11.971	38.2088	38.2088	.8446	57.1319
.32E-06		-.0065	2.5735	56.2362	-.659	2.130	.740	6.560	38.5432	38.5432	.8521	57.6400
.30E-06	SURF REF	.0000	2.5788	56.2926	-.661	2.128	.745	-6.517	38.5803	38.5803	.8529	57.6968
.31E-06		-.0488	2.5233	56.6468	-.675	2.114	.688	-10.794	38.8146	38.8146	.8582	58.0547
.54E-06		-.3052	2.2457	57.8238	-.718	2.070	.409	-13.112	39.6146	39.6146	.8760	59.2601
.17E-06	RCVR	-1.0000	1.5146	60.4317	-.808	1.979	-.290	-15.580	41.4271	41.4271	.9160	61.9605
.89E-07	BOTM REF	-2.5168	.0000	66.1771	-.981	-.413	-1.628	11.913	45.4148	45.4148	1.0038	67.9048
.29E-06		-1.0613	1.5984	72.2693	-.948	-.384	-.389	14.338	49.6148	49.6148	1.0964	74.1674
.28E-06	RCVR	-1.0000	1.6659	72.5091	-.947	-.383	-.341	14.335	49.7813	49.7813	1.1001	74.4149
.40E-06		-.1073	2.6484	76.4054	-.928	-.369	.311	10.473	52.4563	52.4563	1.1592	78.4130
.37E-06		-.0147	2.7542	77.0815	-.925	-.366	.371	3.080	52.9063	52.9063	1.1693	79.0962
.34E-06	SURF REF	.0000	2.7745	77.3854	-.924	-.365	.378	-2.616	53.1049	53.1049	1.1738	79.4004
.39E-06		-.0285	2.7550	77.8822	-.922	-.363	.350	-5.771	53.4299	53.4299	1.1811	79.8981
.53E-06		-.1877	2.6111	78.7936	-.918	-.359	.222	-10.778	54.0424	54.0424	1.1948	80.8236
.27E-06	RCVR	-1.0000	1.8466	82.2672	-.904	-.351	-.383	-14.302	56.4312	56.4312	1.2476	84.3914
.17E-06	BOTM REF	-2.9130	.0000	90.5056	-.872	-1.030	-1.630	10.348	62.0907	62.0908	1.3726	92.8480
.38E-06		-1.4652	1.4772	97.2370	-.802	-.967	-.723	13.513	66.6907	66.6908	1.4743	99.7329
.32E-06	RCVR	-1.0000	1.9482	99.1598	-.784	-.956	-.457	13.622	68.0193	68.0194	1.5036	101.7111
.50E-06		-.1302	2.8276	103.1436	-.747	-.934	.009	9.972	70.7443	70.7444	1.5639	105.7899
.41E-06		-.0204	2.9390	103.9411	-.740	-.928	.062	2.046	71.2756	71.2756	1.5758	106.5958
.41E-06	APOGEE	-.0169	2.9429	104.1662	-.738	-.926	.062	-.051	71.4225	71.4225	1.5791	106.8209
.41E-06	WAVE REV	-.0169	2.9429	104.1662	-.738	-.926	.062	-.051	71.4225	71.4225	1.5791	106.8209
.41E-06		-.0277	2.9328	104.5105	-.735	-.924	.052	-4.479	71.6475	71.6475	1.5842	107.1656
.55E-06		-.2094	2.7531	105.6202	-.726	-.917	-.057	-10.224	72.3912	72.3912	1.6009	108.2905
.30E-06	RCVR	-1.0000	1.9680	109.1722	-.697	-.904	-.501	-13.547	74.8237	74.8237	1.6547	111.9302
.16E-06	BOTM REF	-2.9775	.0000	118.0022	-.629	-.958	-1.500	10.589	80.8701	80.8701	1.7884	120.9783

.36E-06		-1.3162	1.6666	125.6139	-.571	-.921	-.718	13.227	86.0701	86.0701	1.9036	128.7684
.45E-06	RCVR	-1.0000	1.9835	126.9610	-.561	-.922	-.580	13.183	86.9964	86.9964	1.9240	130.1520
.63E-06		-.1308	2.8547	131.0325	-.533	-.922	-.218	9.947	89.7714	89.7714	1.9856	134.3163
.55E-06		-.0196	2.9662	131.8406	-.527	-.919	-.175	2.104	90.3089	90.3089	1.9977	135.1329
.55E-06	APOGEE	-.0149	2.9711	132.1571	-.525	-.918	-.176	-.035	90.5151	90.5152	2.0023	135.4495
.55E-06	WAVE REV	-.0149	2.9711	132.1571	-.525	-.918	-.176	-.035	90.5151	90.5152	2.0023	135.4495
.55E-06		-.0182	2.9678	132.4161	-.523	-.917	-.179	-1.642	90.6839	90.6839	2.0062	135.7085
.61E-06		-.1040	2.8823	133.1137	-.518	-.915	-.221	-9.862	91.1464	91.1464	2.0166	136.4123
.54E-06		-.5923	2.3950	135.5879	-.502	-.913	-.446	-12.752	92.8214	92.8214	2.0539	138.9348
.43E-06	RCVR	-1.0000	1.9879	137.3491	-.490	-.917	-.626	-13.126	94.0299	94.0299	2.0806	140.7425
.30E-06	BOTM REF	-2.9904	.0000	146.2585	-.433	-.926	-1.445	10.886	100.1225	100.1225	2.2155	149.8706
.47E-06		-1.5123	1.4795	152.9963	-.392	-.903	-.887	13.196	104.7225	104.7225	2.3175	156.7677
.43E-06	RCVR	-1.0000	1.9922	155.1897	-.379	-.910	-.705	13.086	106.2289	106.2289	2.3508	159.0199
.60E-06		-.1342	2.8587	159.2642	-.355	-.917	-.412	9.927	109.0039	109.0039	2.4124	163.1864
.51E-06		-.0195	2.9735	160.1005	-.350	-.916	-.376	1.968	109.5602	109.5602	2.4249	164.0316
.51E-06	APOGEE	-.0156	2.9775	160.3739	-.348	-.915	-.376	-.042	109.7383	109.7383	2.4289	164.3050
.51E-06	WAVE REV	-.0156	2.9775	160.3739	-.348	-.915	-.376	-.042	109.7383	109.7383	2.4289	164.3050
.51E-06		-.0215	2.9717	160.6807	-.346	-.915	-.381	-2.625	109.9383	109.9383	2.4334	164.6119
.59E-06		-.1441	2.8492	161.5250	-.342	-.913	-.430	-9.955	110.5008	110.5008	2.4461	165.4658
.34E-06		-.8617	2.1320	164.9413	-.322	-.917	-.700	-13.106	112.8258	112.8258	2.4977	168.9575
.37E-06	RCVR	-1.0000	1.9938	165.5342	-.319	-.919	-.751	-13.134	113.2332	113.2332	2.5067	169.5663
.13E-06	BOTM REF	-2.9948	.0000	174.5912	-.269	-.914	-1.446	10.549	119.4250	119.4250	2.6438	178.8397
.29E-06		-1.3307	1.6647	182.2004	-.231	-.891	-.929	13.422	124.6250	124.6250	2.7589	186.6280
.33E-06	RCVR	-1.0000	1.9956	183.5857	-.224	-.893	-.832	13.413	125.5800	125.5800	2.7800	188.0522
.55E-06		-.1330	2.8629	187.6118	-.204	-.891	-.585	9.843	128.3300	128.3300	2.8409	192.1718
.46E-06		-.0209	2.9751	188.4473	-.200	-.889	-.556	1.564	128.8863	128.8863	2.8533	193.0158
.46E-06	APOGEE	-.0192	2.9767	188.5766	-.199	-.889	-.556	-.040	128.9706	128.9707	2.8553	193.1451
.46E-06	WAVE REV	-.0192	2.9767	188.5766	-.199	-.889	-.556	-.040	128.9706	128.9707	2.8553	193.1451
.47E-06		-.0492	2.9468	189.0334	-.197	-.888	-.568	-8.228	129.2706	129.2707	2.8620	193.6034
.58E-06		-.2966	2.6994	190.4456	-.190	-.884	-.651	-10.778	130.2206	130.2207	2.8832	195.0373
.32E-06	RCVR	-1.0000	1.9963	193.5470	-.175	-.882	-.876	-13.489	132.3496	132.3497	2.9302	198.2177
.20E-06	BOTM REF	-2.9968	.0000	202.7494	-.134	-.862	-1.482	9.968	138.6453	138.6453	3.0694	207.6343
.24E-06		-1.6529	1.3442	209.2034	-.107	-.839	-1.125	13.342	143.0453	143.0453	3.1669	214.2261
.38E-06	MAX RANG	-1.4448	1.5522	210.0720	-.104	-.837	-1.071	13.577	143.6453	143.6453	3.1801	215.1191

THIS RAY CALCULATION TOOK 12.853 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 16.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
-.7E-14	XMTR	-.9808	2.0000	.0000				16.000	.0000	.0000	.0000	.0000
.20E-06		-.0835	2.8949	3.4327	-.014	-.014	14.634	12.436	2.3750	2.3750	.0525	3.5486
.17E-06		-.0050	2.9730	3.8770	-.013	-.014	14.111	7.448	2.6719	2.6719	.0591	4.0000
.15E-06	SURF REF	.0000	2.9780	3.9152	-.013	-.014	14.047	-7.420	2.6970	2.6970	.0597	4.0386
.15E-06		-.0555	2.9222	4.2683	-.013	-.013	12.213	-11.786	2.9314	2.9314	.0650	4.3961
.38E-06		-.3556	2.6210	5.5701	-.012	-.012	6.379	-14.064	3.8189	3.8189	.0847	5.7321
-.3E-07	RCVR	-1.0000	1.9745	7.8975	-.013	-.014	-.175	-16.003	5.4410	5.4410	.1204	8.1469
-.3E-06	BOTM REF	-2.9653	.0000	15.3961	-.010	.213	-7.416	12.811	10.6304	10.6304	.2350	15.8979
-.4E-07		-1.3042	1.6493	21.7624	-.075	.148	-.950	16.069	15.0304	15.0304	.3323	22.4762
-.4E-07	RCVR	-1.0000	1.9510	22.8151	-.083	.140	-.151	16.139	15.7682	15.7682	.3485	23.5718
.11E-06		-.0827	2.8586	26.2807	-.103	.119	1.839	12.606	18.1682	18.1682	.4015	27.1573
.16E-06		-.0031	2.9368	26.7249	-.105	.117	1.975	7.730	18.4651	18.4651	.4081	27.6088
.15E-06	SURF REF	.0000	2.9398	26.7475	-.105	.117	1.980	-7.714	18.4800	18.4800	.4085	27.6316
.14E-06		-.0596	2.8789	27.1094	-.107	.115	1.824	-12.156	18.7206	18.7206	.4139	27.9985
.38E-06		-.3785	2.5552	28.4642	-.113	.109	1.084	-14.413	19.6456	19.6456	.4345	29.3904
.46E-07	RCVR	-1.0000	1.9244	30.6799	-.122	.099	-.174	-16.138	21.1918	21.1918	.4685	31.6915
-.2E-06	BOTM REF	-2.8791	.0000	37.7222	-.143	1.485	-3.051	13.972	26.0731	26.0731	.5762	38.9792
-.2E-06		-1.4867	1.3450	42.7673	-.319	1.308	-.870	16.689	29.5731	29.5731	.6536	44.2132
-.3E-06	RCVR	-1.0000	1.8118	44.3760	-.367	1.260	-.224	16.889	30.7047	30.7047	.6784	45.8940
-.3E-06		-.0693	2.6918	47.7025	-.456	1.169	.880	13.415	33.0172	33.0172	.7295	49.3494
-.2E-06	SURF REF	.0000	2.7549	48.0587	-.465	1.161	.953	-9.230	33.2556	33.2556	.7349	49.7125
-.1E-06		-.0077	2.7463	48.1059	-.466	1.159	.943	-9.268	33.2869	33.2869	.7356	49.7604
-.2E-07		-.0989	2.6473	48.5373	-.477	1.149	.823	-13.665	33.5775	33.5775	.7421	50.2015
-.5E-06		-.8743	1.8155	51.3273	-.540	1.085	-.112	-16.849	35.5150	35.5150	.7849	53.0982
-.4E-06	RCVR	-1.0000	1.6806	51.7418	-.549	1.075	-.254	-16.886	35.8066	35.8066	.7914	53.5313
-.3E-06	BOTM REF	-2.5573	.0000	57.1345	-.652	5.462	-1.838	16.468	39.5718	39.5718	.8743	59.1442
-.5E-06		-1.1110	1.4095	61.6638	-1.055	5.057	-.398	18.382	42.7718	42.7718	.9449	63.9162
-.5E-06	RCVR	-1.0000	1.5183	61.9964	-1.083	5.030	-.297	18.383	43.0087	43.0087	.9501	64.2680
-.4E-06		-.0607	2.4436	65.0092	-1.318	4.793	.519	15.127	45.1274	45.1274	.9969	67.4348
-.4E-06	SURF REF	.0000	2.5035	65.2683	-1.337	4.774	.567	-11.851	45.3027	45.3027	1.0008	67.7019
-.5E-06		-.0188	2.4845	65.3570	-1.343	4.768	.549	-12.108	45.3620	45.3620	1.0022	67.7929
-.2E-06		-.2051	2.2966	66.0428	-1.393	4.718	.376	-15.817	45.8339	45.8339	1.0127	68.5059
-.9E-06	RCVR	-1.0000	1.5003	68.5421	-1.566	4.543	-.324	-18.376	47.5969	47.5969	1.0516	71.1365
-.7E-06	BOTM REF	-2.5168	.0000	73.2669	-1.861	1.370	-1.531	15.683	50.9337	50.9337	1.1251	76.1111
-.9E-06	RCVR	-1.0000	1.5937	78.2658	-1.950	1.277	-.366	17.518	54.4373	54.4373	1.2024	81.3353
-.7E-06		-.0628	2.5914	81.4574	-2.001	1.222	.279	14.213	56.6623	56.6623	1.2516	84.6628
-.8E-06	SURF REF	.0000	2.6597	81.7521	-2.005	1.217	.320	-10.579	56.8600	56.8600	1.2560	84.9642
-.7E-06		-.0186	2.6430	81.8509	-2.007	1.216	.305	-10.852	56.9256	56.9256	1.2575	85.0648
-.5E-06		-.1767	2.4966	82.4834	-2.016	1.206	.188	-14.797	57.3569	57.3569	1.2671	85.7170
-.9E-06	RCVR	-1.0000	1.7211	85.2403	-2.056	1.161	-.396	-17.499	59.2834	59.2834	1.3097	88.5949
-.8E-06	BOTM REF	-2.8039	.0000	91.2552	-2.133	-1.725	-1.555	13.679	63.4879	63.4879	1.4025	94.8744
-.6E-06		-1.1211	1.7570	97.4561	-2.024	-1.626	-.521	16.054	67.7879	67.7879	1.4975	101.3007
-.6E-06	RCVR	-1.0000	1.8817	97.8766	-2.017	-1.620	-.451	16.051	68.0819	68.0819	1.5040	101.7384
-.4E-06		-.0850	2.8212	101.3112	-1.963	-1.577	.051	12.950	70.4569	70.4569	1.5566	105.2944
-.5E-06		-.0008	2.9081	101.7598	-1.956	-1.571	.094	8.329	70.7569	70.7569	1.5633	105.7512
-.5E-06	SURF REF	.0000	2.9090	101.7652	-1.956	-1.571	.095	-8.326	70.7605	70.7604	1.5634	105.7567
-.5E-06		-.0680	2.8431	102.1449	-1.951	-1.566	.053	-12.749	71.0136	71.0136	1.5691	106.1427
-.3E-06		-.4618	2.4578	103.7530	-1.927	-1.546	-.180	-15.059	72.1136	72.1136	1.5936	107.7989

-.6E-06	RCVR	-1.0000	1.9283	105.6589	-1.899	-1.527	-.486	-16.008	73.4412	73.4412	1.6229	109.7798
-.9E-06	BOTM REF	-2.9512	.0000	112.8133	-1.803	-1.841	-1.508	13.397	78.3987	78.3987	1.7325	117.1962
-.7E-06		-1.1041	1.8610	119.7562	-1.696	-1.754	-.598	15.571	83.1987	83.1987	1.8387	124.3822
-.7E-06	RCVR	-1.0000	1.9656	120.1295	-1.691	-1.752	-.549	15.561	83.4584	83.4583	1.8445	124.7699
-.6E-06		-.0862	2.8842	123.6498	-1.641	-1.723	-.142	12.758	85.8834	85.8833	1.8983	128.4087
-.6E-06		-.0036	2.9674	124.0942	-1.635	-1.718	-.107	8.097	86.1803	86.1802	1.9049	128.8611
-.6E-06	SURF REF	.0000	2.9710	124.1192	-1.635	-1.718	-.105	-8.086	86.1967	86.1967	1.9053	128.8864
-.7E-06		-.0611	2.9103	124.4767	-1.630	-1.714	-.136	-12.375	86.4342	86.4342	1.9107	129.2493
-.5E-06		-.3875	2.5855	125.8629	-1.611	-1.701	-.296	-14.243	87.3780	87.3780	1.9317	130.6740
-.9E-06	RCVR	-1.0000	1.9752	128.1144	-1.581	-1.688	-.585	-15.517	88.9375	88.9375	1.9663	133.0082
-.1E-05	BOTM REF	-2.9809	.0000	135.4746	-1.489	-1.724	-1.455	13.598	94.0267	94.0266	2.0789	140.6313
-.9E-06		-1.6847	1.2991	140.4060	-1.428	-1.674	-.919	15.436	97.4267	97.4266	2.1543	145.7308
-.8E-06	RCVR	-1.0000	1.9850	142.8894	-1.399	-1.666	-.650	15.337	99.1495	99.1495	2.1924	148.3074
-.7E-06		-.0845	2.9020	146.4535	-1.358	-1.652	-.308	12.720	101.5995	101.5995	2.2468	151.9888
-.7E-06		-.0012	2.9855	146.9079	-1.353	-1.648	-.279	8.085	101.9027	101.9026	2.2537	152.4513
-.7E-06	SURF REF	.0000	2.9867	146.9161	-1.353	-1.648	-.278	-8.082	101.9081	101.9080	2.2538	152.4596
-.8E-06		-.0651	2.9217	147.2923	-1.349	-1.645	-.306	-12.469	102.1581	102.1580	2.2595	152.8417
-.5E-06		-.4230	2.5644	148.8073	-1.332	-1.637	-.454	-14.360	103.1893	103.1893	2.2825	154.3990
-.9E-06	RCVR	-1.0000	1.9881	150.9431	-1.309	-1.632	-.686	-15.335	104.6667	104.6667	2.3152	156.6119
-.1E-05	BOTM REF	-2.9902	.0000	158.3933	-1.231	-1.623	-1.439	13.562	109.8135	109.8135	2.4292	164.3244
-.1E-05		-1.7753	1.2160	163.0372	-1.184	-1.585	-1.012	15.427	113.0135	113.0135	2.5002	169.1249
-.9E-06	RCVR	-1.0000	1.9918	165.8384	-1.157	-1.578	-.752	15.416	114.9585	114.9585	2.5431	172.0318
-.7E-06		-.0842	2.9082	169.3993	-1.123	-1.568	-.459	12.642	117.4085	117.4085	2.5975	175.7102
-.8E-06		-.0024	2.9901	169.8489	-1.119	-1.565	-.434	7.931	117.7085	117.7085	2.6043	176.1675
-.8E-06	SURF REF	.0000	2.9925	169.8664	-1.119	-1.565	-.433	-7.923	117.7200	117.7200	2.6045	176.1853
-.8E-06		-.0620	2.9306	170.2334	-1.116	-1.562	-.456	-12.297	117.9638	117.9637	2.6100	176.5577
-.5E-06		-.3859	2.6069	171.6202	-1.103	-1.556	-.573	-14.144	118.9075	118.9075	2.6311	177.9824
-.9E-06	RCVR	-1.0000	1.9931	173.8890	-1.082	-1.551	-.788	-15.459	120.4791	120.4791	2.6659	180.3335
-.1E-05	BOTM REF	-2.9941	.0000	181.4326	-1.016	-1.525	-1.452	13.200	125.6912	125.6911	2.7812	188.1371
-.9E-06		-1.1587	1.8361	188.3780	-.959	-1.483	-.901	15.678	130.4912	130.4911	2.8874	195.3217
-.9E-06	RCVR	-1.0000	1.9949	188.9434	-.955	-1.482	-.856	15.669	130.8852	130.8852	2.8961	195.9090
-.8E-06		-.0845	2.9107	192.4767	-.927	-1.467	-.599	12.533	133.3227	133.3227	2.9501	199.5606
-.8E-06		-.0045	2.9908	192.9211	-.924	-1.465	-.578	7.682	133.6196	133.6195	2.9568	200.0125
-.8E-06	SURF REF	.0000	2.9953	192.9544	-.923	-1.464	-.577	-7.663	133.6416	133.6415	2.9573	200.0462
-.8E-06		-.0572	2.9381	193.3075	-.921	-1.462	-.596	-11.983	133.8759	133.8759	2.9626	200.4041
-.4E-06		-.3653	2.6301	194.6382	-.911	-1.455	-.694	-14.055	134.7822	134.7821	2.9828	201.7704
-.8E-06	RCVR	-1.0000	1.9956	196.9598	-.893	-1.446	-.891	-15.715	136.3953	136.3952	3.0184	204.1778
-.9E-06	BOTM REF	-2.9961	.0000	204.5749	-.838	-1.412	-1.485	12.804	141.6603	141.6602	3.1347	212.0512
-.7E-06	MAX RANG	-1.5880	1.4085	210.0885	-.801	-1.378	-1.110	15.626	145.4603	145.4602	3.2189	217.7421

THIS RAY CALCULATION TOOK 13.791 SEC

***** H A R P O *****
HAMILTONIAN ACOUSTIC RAY-TRACING PROGRAM FOR THE OCEAN

BY
R. M. JONES, J. P. RILEY AND T. M. GEORGES
WAVE PROPAGATION LABORATORY
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
BOULDER, COLORADO 80303

RUN SET NUMBER 2

OCEAN MODEL ID — N01

OCEAN MODEL DESCRIPTION — SAMPLE CASE FOR HARPO DOCUMENTATION REV. 8-19-86

MODEL TYPE	SUBROUTINE NAME	DATA SET ID	DESCRIPTION
DISPERSION RELATION	AWCWL		ACOUSTIC WAVE *** WITH CURRENT *** WITH LOSSES
BACKGROUND CURRENT VELOCITY	VVORTX3	3.00	VORTEX AT LONGITUDE 150 KM E, UMAX= 1.02 M/S, R= 50 KM
CURRENT VELOCITY PERTURBATION	NPCURR	.00	NO CURRENT PERTURBATION
BACKGROUND SOUND SPEED	CTANH	1.00	EL NINO BACKGROUND SOUND-SPEED PROFILE
SOUND SPEED PERTURBATION	CBLOB2	7.00	2% INCREASE IN C-SQUARED AT 150 KM LON., 1 KM DEPTH, 50 KM WIDE
BACKGROUND BOTTOM	GLORENZ	3.00	RIDGE .5 KM HIGH, 2 KM WIDE AT 10 KM N LATITUDE; BASE= -3 KM
BOTTOM PERTURBATION	NPBOTM	.00	NO BOTTOM PERTURBATION
ABSORPTION	SLLOSS	1.00	SKRETTING-LEROY ABSORPTION FORMULA
ABSORPTION PERTURBATION	NPABSR	.00	NO ABSORPTION PERTURBATION
RECEIVER SURFACE	RHORIZ	.00	RECEIVER SURFACE = SPHERE 1 KM BELOW MSL
OCEAN SURFACE	SHORIZ	1.00	OCEAN SURFACE = SPHERE AT MSL
OCEAN SURFACE PERTURBATION	NPSURF	.00	NO OCEAN SURFACE PERTURBATION

INPUT DATA FILE FOR RUN SET NUMBER 2

```
N01-2     SAMPLE CASE FOR HARPO DOCUMENTATION        REV. 8-19-86
 71        0.          NUMBER OF INTEGRATION STEPS PER PRINT [1.E31]
 72        0.          OUTPUT RAYSETS (1=YES; 0=NO)
 73        1.          DIAGNOSTIC PRINTOUT (1=YES; 0=NO)
 81        2.          RAYPLOT PROJECTION PLANE (2 = HORIZONTAL)
 88       -1.          HEIGHT OF HORIZONTAL PLOT SECTION ABOVE MSL, KM
  0                    **********  END OF RUN SET NUMBER 2  *********
```

INITIAL VALUES FOR THE W ARRAY
ONLY NONZERO VALUES PRINTED — ALL ANGLES IN RADIANS

N	W(N)	
1	.637000000000000E+04	
3	-.980769230769170E+00	
7	.251327412287184E+04	
11	.139626340159545E+01	
15	.349065850398864E-01	
16	.279252680319091E+00	
17	.349065850398864E-01	
20	-.100000000000000E+01	
22	.500000000000000E+02	
23	.100000000000000E+04	
24	.157079632679490E+01	
26	.500000000000000E+01	
27	-.500000000000000E+01	
28	.210000000000000E+03	
33	.200000000000000E+02	
41	.300000000000000E+01	
42	.999999999999997E-06	
43	.500000000000000E+02	
44	.100000000000000E+00	
45	.100000000000000E+03	
46	.100000000000000E-07	
47	.500000000000000E+00	
57	.200000000000000E+01	
58	.200000000000000E+01	
60	.200000000000000E+01	
71	.999999999999999E+30	INPUT OVERRIDDEN
73	.100000000000000E+01	
75	.150000000000000E+00	
77	.570000000000000E+02	
81	.200000000000000E+01	
82	.400000000000000E+02	
85	.551020408163264E-02	
86	.313971742543171E-01	
87	.784929356357927E-02	
88	-.100000000000000E+01	
96	.100000000000000E+01	
100	.900000000000000E+01	
102	.300000000000000E+01	
103	.102000000000001E-02	
104	.500000000000000E+02	
106	.235478806907378E-01	
107	.100000000000000E+01	
108	-.100000000000000E+01	
150	.700000000000000E+01	
152	.100000000000000E+01	
175	.200000000000000E+01	
177	.700000000000000E+01	
178	.200000000000000E-01	
179	-.100000000000000E+01	
181	.235478806907378E-01	
182	.100000000000000E+01	
183	.784929356357927E-02	
184	.784929356357927E-02	
275	.100000000000000E+01	
300	.400000000000000E+01	
302	.300000000000000E+01	
303	.500000000000000E+00	
304	.156985871271585E-02	
305	.313971742543171E-03	
306	-.300000000000000E+01	
350	.100000000000000E+01	
352	.100000000000000E+01	
500	.100000000000000E+01	
502	.100000000000000E+01	
503	.138155106185094E-02	
504	.606731174662869E-01	
505	.628318530717958E+04	
506	.106814150222053E+05	

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 2.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
.00E+00	XMTR	-.9808	2.0000	.0000				2.000	.0000	.0000	.0000	.0000
.85E-09	APOGEE	-.7737	2.1992	9.5046	-.018	-.019	1.205	-.014	6.4000	6.4000	.1405	9.5061
.85E-09	WAVE REV	-.7737	2.1992	9.5046	-.018	-.019	1.205	-.014	6.4000	6.4000	.1405	9.5061
-.2E-07	BACK UP0	-1.0204	1.9367	20.0479	-.018	-.020	-.203	-2.037	13.5000	13.5000	.2964	20.0514
-.2E-07	BACK UP1	-1.0000	1.9582	19.4734	-.018	-.020	-.144	-2.022	13.1129	13.1129	.2879	19.4767
-.2E-07	RCVR	-1.0000	1.9582	19.4734	-.018	-.020	-.144	-2.022	13.1129	13.1129	.2879	19.4767
-.1E-07	PERIGEE	-1.3869	1.5043	35.8077	-.017	-.023	-.811	.016	24.1129	24.1129	.5294	35.8134
-.1E-07	WAVE REV	-1.3869	1.5043	35.8077	-.017	-.023	-.811	.016	24.1129	24.1129	.5294	35.8134
-.2E-07	BACK UP0	-.9965	1.5972	52.1429	-.016	-.027	-.252	2.007	35.1129	35.1129	.7709	52.1510
-.2E-07	BACK UP1	-1.0000	1.5965	52.0417	-.016	-.027	-.255	2.010	35.0447	35.0447	.7694	52.0498
-.2E-07	RCVR	-1.0000	1.5965	52.0417	-.016	-.027	-.255	2.010	35.0447	35.0447	.7694	52.0498
-.1E-07	APOGEE	-.7732	1.7938	62.2163	-.016	-.033	-.089	-.023	41.8947	41.8947	.9199	62.2261
-.1E-07	WAVE REV	-.7732	1.7938	62.2163	-.016	-.033	-.089	-.023	41.8947	41.8947	.9199	62.2261
-.3E-07	BACK UP0	-1.0158	1.7998	72.7647	-.015	-.042	-.355	-2.002	48.9947	48.9947	1.0759	72.7765
-.3E-07	BACK UP1	-1.0000	1.8084	72.3109	-.015	-.042	-.340	-1.990	48.6891	48.6891	1.0691	72.3225
-.3E-07	RCVR	-1.0000	1.8084	72.3109	-.015	-.042	-.340	-1.990	48.6891	48.6891	1.0691	72.3225
-.3E-07	PERIGEE	-1.4302	1.5166	91.0432	-.009	-.070	-.692	.029	61.2891	61.2891	1.3461	91.0571
-.3E-07	WAVE REV	-1.4302	1.5166	91.0432	-.009	-.070	-.692	.029	61.2891	61.2891	1.3461	91.0571
-.3E-07	BACK UP0	-.9993	1.9799	112.7886	.006	-.147	-.517	1.417	75.8891	75.8891	1.6676	112.8031
-.3E-07	BACK UP1	-1.0000	1.9792	112.7596	.006	-.147	-.517	1.418	75.8697	75.8697	1.6671	112.7741
-.3E-07	RCVR	-1.0000	1.9792	112.7595	.006	-.147	-.517	1.418	75.8697	75.8697	1.6671	112.7741
-.3E-07	APOGEE	-.7919	2.1946	126.4921	.024	-.226	-.483	-.038	85.0697	85.0697	1.8701	126.5068
-.3E-07	WAVE REV	-.7919	2.1946	126.4921	.024	-.226	-.483	-.038	85.0697	85.0697	1.8701	126.5068
-.3E-07	BACK UP0	-1.0043	1.9865	140.8411	.052	-.325	-.643	-1.287	94.6697	94.6697	2.0823	140.8559
-.3E-07	BACK UP1	-1.0000	1.9907	140.6506	.051	-.324	-.640	-1.281	94.5422	94.5422	2.0794	140.6653
-.3E-07	RCVR	-1.0000	1.9907	140.6506	.051	-.324	-.640	-1.281	94.5422	94.5422	2.0794	140.6653
-.3E-07	PERIGEE	-1.8317	1.1635	172.3289	.136	-.470	-1.058	.068	115.7422	115.7422	2.5478	172.3494
-.3E-07	WAVE REV	-1.8317	1.1635	172.3289	.136	-.470	-1.058	.068	115.7422	115.7422	2.5478	172.3494
-.5E-07	BACK UP0	-.9856	2.0111	194.3880	.198	-.504	-.876	3.118	130.5422	130.5422	2.8742	194.4236
-.5E-07	BACK UP1	-1.0000	1.9967	194.1240	.197	-.503	-.879	3.128	130.3646	130.3646	2.8702	194.1592
-.5E-07	RCVR	-1.0000	1.9967	194.1240	.197	-.503	-.879	3.128	130.3646	130.3646	2.8702	194.1592
-.4E-07	APOGEE	-.6456	2.3515	203.6491	.223	-.510	-.822	-.031	136.7646	136.7646	3.0112	203.6915
-.4E-07	WAVE REV	-.6456	2.3515	203.6491	.223	-.510	-.822	-.031	136.7646	136.7646	3.0112	203.6915
-.5E-07	MAX RANG	-.8511	2.1462	210.1960	.240	-.509	-.910	-2.971	141.1646	141.1646	3.1080	210.2419

THIS RAY CALCULATION TOOK 2.697 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 4.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
.00E+00	XMTR	-.9808	2.0000	.0000				4.000	.0000	.0000	.0000	.0000
.16E-07	APOGEE	-.5981	2.3760	8.2406	-.017	-.018	2.622	-.098	5.5500	5.5500	.1220	8.2504
.16E-07	WAVE REV	-.5981	2.3760	8.2406	-.017	-.018	2.622	-.098	5.5500	5.5500	.1220	8.2504
-.4E-08	BACK UP0	-1.0007	1.9629	16.5543	-.017	-.019	-.143	-4.012	11.1500	11.1500	.2450	16.5745
-.4E-08	BACK UP1	-1.0000	1.9635	16.5446	-.017	-.019	-.141	-4.012	11.1435	11.1435	.2449	16.5649
-.4E-08	RCVR	-1.0000	1.9635	16.5446	-.017	-.019	-.141	-4.012	11.1435	11.1435	.2449	16.5649
-.4E-08	PERIGEE	-1.6999	1.2170	31.8384	-.015	-.019	-1.437	.069	21.4435	21.4435	.4712	31.8750
-.4E-08	WAVE REV	-1.6999	1.2170	31.8384	-.015	-.019	-1.437	.069	21.4435	21.4435	.4712	31.8750
-.6E-07	BACK UP0	-.9898	1.7412	46.9831	-.015	-.023	-.222	3.996	31.6435	31.6435	.6953	47.0363
-.6E-07	BACK UP1	-1.0000	1.7344	46.8372	-.015	-.022	-.234	4.001	31.5450	31.5450	.6932	46.8901
-.6E-07	RCVR	-1.0000	1.7344	46.8372	-.015	-.022	-.234	4.001	31.5450	31.5450	.6932	46.8901
-.4E-07	APOGEE	-.5978	1.9213	55.3751	-.015	-.026	.147	-.094	37.2950	37.2950	.8195	55.4384
-.4E-07	WAVE REV	-.5978	1.9213	55.3751	-.015	-.026	.147	-.094	37.2950	37.2950	.8195	55.4384
-.6E-07	BACK UP0	-1.0181	1.5972	63.9874	-.014	-.031	-.321	-3.992	43.0950	43.0950	.9470	64.0617
-.6E-07	BACK UP1	-1.0000	1.6081	63.7278	-.014	-.031	-.304	-3.985	42.9199	42.9199	.9432	63.8015
-.6E-07	RCVR	-1.0000	1.6081	63.7278	-.014	-.031	-.304	-3.985	42.9199	42.9199	.9432	63.8015
-.2E-07	PERIGEE	-1.7286	1.1645	79.7738	-.012	-.043	-.896	.058	53.7199	53.7199	1.1806	79.8643
-.2E-07	WAVE REV	-1.7286	1.1645	79.7738	-.012	-.043	-.896	.058	53.7199	53.7199	1.1806	79.8643
-.6E-07	BACK UP0	-.9931	1.9664	96.4298	-.006	-.075	-.441	3.614	64.9199	64.9199	1.4271	96.5359
-.6E-07	BACK UP1	-1.0000	1.9593	96.3209	-.006	-.074	-.445	3.618	64.8465	64.8465	1.4255	96.4267
-.6E-07	RCVR	-1.0000	1.9593	96.3209	-.006	-.074	-.445	3.618	64.8465	64.8465	1.4255	96.4267
-.5E-07	APOGEE	-.5997	2.3730	105.5524	-.001	-.106	-.268	-.032	71.0465	71.0465	1.5621	105.6674
-.5E-07	WAVE REV	-.5997	2.3730	105.5524	-.001	-.106	-.268	-.032	71.0465	71.0465	1.5621	105.6674
-.6E-07	BACK UP0	-1.0116	1.9692	115.0917	.007	-.149	-.533	-3.482	77.4465	77.4465	1.7032	115.2161
-.6E-07	BACK UP1	-1.0000	1.9806	114.9018	.007	-.148	-.526	-3.476	77.3190	77.3190	1.7004	115.0258
-.6E-07	RCVR	-1.0000	1.9806	114.9018	.007	-.148	-.526	-3.476	77.3190	77.3190	1.7004	115.0258
-.2E-07	PERIGEE	-2.0665	.9238	138.7830	.042	-.232	-1.072	.018	93.3190	93.3190	2.0538	138.9293
-.2E-07	WAVE REV	-2.0665	.9238	138.7830	.042	-.232	-1.072	.018	93.3190	93.3190	2.0538	138.9293
-.3E-07	BACK UP0	-.9810	2.0135	165.6749	.091	-.349	-.745	2.644	111.3190	111.3190	2.4516	165.8395
-.3E-07	BACK UP1	-1.0000	1.9945	165.2653	.091	-.346	-.750	2.658	111.0446	111.0445	2.4455	165.4295
-.3E-07	RCVR	-1.0000	1.9945	165.2653	.091	-.346	-.750	2.658	111.0446	111.0446	2.4455	165.4295
-.3E-07	APOGEE	-.6490	2.3465	176.1659	.116	-.403	-.684	-.067	118.3446	118.3446	2.6068	176.3358
-.3E-07	WAVE REV	-.6490	2.3465	176.1659	.116	-.403	-.684	-.067	118.3446	118.3446	2.6068	176.3358
-.3E-07	BACK UP0	-1.0286	1.9676	187.0517	.143	-.441	-.856	-2.835	125.6446	125.6446	2.7678	187.2282
-.3E-07	BACK UP1	-1.0000	1.9962	186.4722	.142	-.439	-.845	-2.819	125.2553	125.2553	2.7592	186.6482
-.3E-07	RCVR	-1.0000	1.9962	186.4722	.142	-.439	-.845	-2.819	125.2554	125.2553	2.7592	186.6482
-.1E-07	PERIGEE	-1.6644	1.3327	205.2309	.188	-.458	-1.114	.087	137.8554	137.8553	3.0367	205.4174
-.1E-07	WAVE REV	-1.6644	1.3327	205.2309	.188	-.458	-1.114	.087	137.8554	137.8553	3.0367	205.4174
-.1E-07	MAX RANG	-1.5734	1.4239	210.5919	.200	-.455	-1.108	1.823	141.4554	141.4553	3.1159	210.7782

THIS RAY CALCULATION TOOK 2.438 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 6.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
.00E+00	XMTR	-.9808	2.0000	.0000				6.000	.0000	.0000	.0000	.0000
.64E-07	APOGEE	-.4612	2.5139	7.2720	-.016	-.016	4.055	-.170	4.9000	4.9000	.1078	7.2936
.64E-07	WAVE REV	-.4612	2.5139	7.2720	-.016	-.016	4.055	-.170	4.9000	4.9000	.1078	7.2936
-.3E-07	BACK UP0	-1.0022	1.9644	14.5418	-.016	-.017	-.150	-6.009	9.8000	9.8000	.2156	14.5861
-.3E-07	BACK UP1	-1.0000	1.9667	14.5209	-.016	-.017	-.141	-6.008	9.7858	9.7858	.2153	14.5651
-.3E-07	RCVR	-1.0000	1.9667	14.5209	-.016	-.017	-.141	-6.008	9.7858	9.7858	.2153	14.5651
-.2E-07	PERIGEE	-2.0399	.8848	30.2627	-.013	-.016	-2.141	.056	20.3858	20.3858	.4486	30.3459
-.2E-07	WAVE REV	-2.0399	.8848	30.2627	-.013	-.016	-2.141	.056	20.3858	20.3858	.4486	30.3459
-.6E-07	BACK UP0	-.9808	1.7719	46.0028	-.012	-.018	-.207	5.989	30.9858	30.9858	.6819	46.1260
-.6E-07	BACK UP1	-1.0000	1.7566	45.8199	-.012	-.018	-.230	5.996	30.8620	30.8620	.6792	45.9421
-.6E-07	RCVR	-1.0000	1.7566	45.8199	-.012	-.018	-.230	5.996	30.8620	30.8620	.6792	45.9421
.28E-07	APOGEE	-.4609	2.1032	53.2398	-.013	-.021	.320	-.089	35.8620	35.8620	.7892	53.3846
.28E-07	WAVE REV	-.4609	2.1032	53.2398	-.013	-.021	.320	-.089	35.8620	35.8620	.7892	53.3846
-.6E-07	BACK UP0	-1.0104	1.5212	60.6594	-.013	-.026	-.301	-5.986	40.8620	40.8620	.8992	60.8273
-.6E-07	BACK UP1	-1.0000	1.5297	60.5599	-.013	-.025	-.291	-5.983	40.7946	40.7946	.8977	60.7272
-.6E-07	RCVR	-1.0000	1.5297	60.5599	-.013	-.025	-.291	-5.983	40.7946	40.7946	.8977	60.7272
-.5E-07	PERIGEE	-2.0686	.7995	76.9038	-.010	-.033	-1.156	.085	51.7946	51.7946	1.1399	77.1109
-.5E-07	WAVE REV	-2.0686	.7995	76.9038	-.010	-.033	-1.156	.085	51.7946	51.7946	1.1399	77.1109
-.7E-07	BACK UP0	-.9884	1.9650	93.5567	-.006	-.055	-.425	5.588	62.9946	62.9946	1.3867	93.8021
-.7E-07	BACK UP1	-1.0000	1.9531	93.4379	-.006	-.055	-.432	5.593	62.9145	62.9145	1.3849	93.6828
-.7E-07	RCVR	-1.0000	1.9531	93.4379	-.006	-.055	-.432	5.593	62.9145	62.9145	1.3849	93.6828
-.5E-09	APOGEE	-.4648	2.5026	101.2476	-.003	-.078	-.163	-.072	68.1645	68.1645	1.5007	101.5133
-.5E-09	WAVE REV	-.4648	2.5026	101.2476	-.003	-.078	-.163	-.072	68.1645	68.1645	1.5007	101.5133
-.7E-07	BACK UP0	-1.0085	1.9678	109.1376	.001	-.108	-.505	-5.474	73.4645	73.4645	1.6176	109.4244
-.7E-07	BACK UP1	-1.0000	1.9762	109.0494	.001	-.108	-.501	-5.472	73.4050	73.4050	1.6163	109.3358
-.7E-07	RCVR	-1.0000	1.9762	109.0494	.001	-.108	-.501	-5.472	73.4050	73.4050	1.6163	109.3358
-.6E-07	PERIGEE	-2.3504	.6375	130.5265	.024	-.156	-1.188	.209	87.8050	87.8050	1.9345	130.8599
-.6E-07	WAVE REV	-2.3504	.6375	130.5265	.024	-.156	-1.188	.209	87.8050	87.8050	1.9345	130.8599
-.2E-07	BACK UP0	-.9874	2.0055	153.2275	.053	-.244	-.692	4.295	103.0050	103.0050	2.2707	153.6014
-.2E-07	BACK UP1	-1.0000	1.9929	153.0604	.052	-.242	-.696	4.301	102.8929	102.8929	2.2682	153.4338
-.2E-07	RCVR	-1.0000	1.9929	153.0604	.052	-.242	-.696	4.301	102.8929	102.8929	2.2682	153.4338
.17E-07	APOGEE	-.5077	2.4864	162.1001	.069	-.302	-.562	-.053	108.9429	108.9429	2.4021	162.4884
.17E-07	WAVE REV	-.5077	2.4864	162.1001	.069	-.302	-.562	-.053	108.9429	108.9429	2.4021	162.4884
-.2E-08	BACK UP0	-1.0209	1.9742	171.2073	.088	-.353	-.783	-4.411	115.0429	115.0429	2.5369	171.6114
-.2E-08	BACK UP1	-1.0000	1.9950	170.9365	.087	-.352	-.775	-4.402	114.8611	114.8611	2.5329	171.3399
-.2E-08	RCVR	-1.0000	1.9950	170.9365	.087	-.352	-.775	-4.402	114.8611	114.8611	2.5329	171.3399
-.2E-07	PERIGEE	-2.1120	.8844	191.2150	.132	-.390	-1.199	.188	128.4611	128.4611	2.8332	191.6505
-.2E-07	WAVE REV	-2.1120	.8844	191.2150	.132	-.390	-1.199	.188	128.4611	128.4611	2.8332	191.6505
-.5E-07	BACK UP0	-.9722	2.0251	209.0670	.168	-.388	-.938	5.525	140.4611	140.4611	3.0977	209.5425
-.5E-07	BACK UP1	-1.0000	1.9972	208.7795	.168	-.388	-.944	5.537	140.2669	140.2669	3.0934	209.2537
-.5E-07	RCVR	-1.0000	1.9972	208.7795	.168	-.388	-.944	5.537	140.2669	140.2669	3.0934	209.2537
-.5E-07	MAX RANG	-.8578	2.1395	210.2600	.171	-.389	-.912	5.404	141.2669	141.2669	3.1154	210.7408

THIS RAY CALCULATION TOOK 2.922 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 8.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
-.7E-14	XMTR	-.9808	2.0000	.0000				8.000	.0000	.0000	.0000	.0000
.13E-07	APOGEE	-.3379	2.6378	6.6740	-.015	-.015	5.473	-.090	4.5000	4.5000	.0992	6.7104
.13E-07	WAVE REV	-.3379	2.6378	6.6740	-.015	-.015	5.473	-.090	4.5000	4.5000	.0992	6.7104
.30E-07	BACK UP0	-1.0227	1.9453	13.5670	-.015	-.016	-.238	-8.010	9.1500	9.1500	.2017	13.6428
.30E-07	BACK UP1	-1.0000	1.9682	13.4059	-.015	-.016	-.142	-8.006	9.0405	9.0405	.1993	13.4802
.30E-07	RCVR	-1.0000	1.9682	13.4059	-.015	-.016	-.142	-8.006	9.0405	9.0405	.1993	13.4802
.78E-07	PERIGEE	-2.4608	.4591	31.2440	-.010	-.013	-2.853	.258	21.0405	21.0405	.4641	31.3913
.78E-07	WAVE REV	-2.4608	.4591	31.2440	-.010	-.013	-2.853	.258	21.0405	21.0405	.4641	31.3913
-.6E-07	BACK UP0	-.9643	1.7293	48.4777	-.009	-.014	-.199	7.977	32.6405	32.6405	.7199	48.7006
-.6E-07	BACK UP1	-1.0000	1.7001	48.2230	-.009	-.014	-.240	7.989	32.4673	32.4673	.7161	48.4435
-.6E-07	RCVR	-1.0000	1.7001	48.2230	-.009	-.014	-.240	7.989	32.4673	32.4673	.7161	48.4435
-.5E-07	APOGEE	-.3378	2.1865	55.0449	-.010	-.018	.422	-.087	37.0673	37.0673	.8175	55.3032
-.5E-07	WAVE REV	-.3378	2.1865	55.0449	-.010	-.018	.422	-.087	37.0673	37.0673	.8175	55.3032
-.3E-07	BACK UP0	-1.0406	1.5239	62.0868	-.010	-.022	-.334	-7.980	41.8173	41.8173	.9222	62.3855
-.3E-07	BACK UP1	-1.0000	1.5573	61.7971	-.010	-.022	-.296	-7.975	41.6204	41.6204	.9179	62.0931
-.3E-07	RCVR	-1.0000	1.5573	61.7971	-.010	-.022	-.296	-7.975	41.6204	41.6204	.9179	62.0931
.20E-07	PERIGEE	-2.4981	.3966	79.9390	-.007	-.026	-1.447	.143	53.8204	53.8204	1.1872	80.3095
.20E-07	WAVE REV	-2.4981	.3966	79.9390	-.007	-.026	-1.447	.143	53.8204	53.8204	1.1872	80.3095
-.7E-07	BACK UP0	-.9972	1.9653	98.0930	-.004	-.048	-.451	7.434	66.0204	66.0204	1.4566	98.5349
-.7E-07	BACK UP1	-1.0000	1.9625	98.0715	-.004	-.048	-.452	7.435	66.0059	66.0059	1.4563	98.5132
-.7E-07	RCVR	-1.0000	1.9625	98.0715	-.004	-.048	-.452	7.435	66.0059	66.0059	1.4563	98.5132
-.6E-07	APOGEE	-.3456	2.6267	105.2099	-.002	-.072	-.127	-.050	70.8059	70.8059	1.5624	105.6865
-.6E-07	WAVE REV	-.3456	2.6267	105.2099	-.002	-.072	-.127	-.050	70.8059	70.8059	1.5624	105.6865
-.5E-07	BACK UP0	-1.0370	1.9420	112.6498	.002	-.103	-.535	-7.319	75.8059	75.8059	1.6729	113.1631
-.5E-07	BACK UP1	-1.0000	1.9788	112.3615	.002	-.102	-.515	-7.311	75.6110	75.6110	1.6686	112.8725
-.5E-07	RCVR	-1.0000	1.9788	112.3615	.002	-.102	-.515	-7.311	75.6110	75.6110	1.6686	112.8725
-.4E-08	PERIGEE	-2.7759	.2131	134.1448	.024	-.134	-1.370	.298	90.2110	90.2110	1.9919	134.7405
-.4E-08	WAVE REV	-2.7759	.2131	134.1448	.024	-.134	-1.370	.298	90.2110	90.2110	1.9919	134.7405
-.3E-07	BACK UP0	-.9740	2.0193	155.9419	.045	-.194	-.699	6.426	104.8110	104.8110	2.3153	156.6182
-.3E-07	BACK UP1	-1.0000	1.9933	155.7112	.045	-.192	-.707	6.434	104.6557	104.6557	2.3118	156.3860
-.3E-07	RCVR	-1.0000	1.9933	155.7112	.045	-.192	-.707	6.434	104.6557	104.6557	2.3118	156.3860
-.3E-07	APOGEE	-.3717	2.6225	163.4742	.056	-.242	-.522	-.149	109.8557	109.8557	2.4270	164.1784
-.3E-07	WAVE REV	-.3717	2.6225	163.4742	.056	-.242	-.522	-.149	109.8557	109.8557	2.4270	164.1784
-.2E-07	BACK UP0	-1.0242	1.9708	171.2300	.069	-.284	-.785	-6.537	115.0557	115.0557	2.5422	171.9652
-.2E-07	BACK UP1	-1.0000	1.9950	171.0190	.069	-.283	-.776	-6.530	114.9135	114.9135	2.5390	171.7528
-.2E-07	RCVR	-1.0000	1.9950	171.0190	.069	-.283	-.776	-6.530	114.9135	114.9135	2.5390	171.7528
.30E-08	PERIGEE	-2.5674	.4291	191.5992	.105	-.307	-1.336	.195	128.7135	128.7135	2.8443	192.4012
.30E-08	WAVE REV	-2.5674	.4291	191.5992	.105	-.307	-1.336	.195	128.7135	128.7135	2.8443	192.4012
-.1E-06	MAX RANG	-1.0281	1.9692	210.0554	.134	-.300	-.958	7.721	141.1135	141.1135	3.1182	210.9320

THIS RAY CALCULATION TOOK 3.529 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 10.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
-.7E-14	XMTR	-.9808	2.0000	.0000				10.000	.0000	.0000	.0000	.0000
.33E-07	APOGEE	-.1048	2.8679	9.6483	-.012	-.013	5.145	-.028	6.4750	6.4750	.1435	9.7053
.33E-07	WAVE REV	-.1048	2.8679	9.6483	-.012	-.013	5.145	-.028	6.4750	6.4750	.1435	9.7053
.56E-07	BACK UP0	-1.0491	1.9088	19.6238	-.012	-.014	-.288	-10.009	13.1750	13.1750	.2919	19.7437
.56E-07	BACK UP1	-1.0000	1.9585	19.3456	-.012	-.014	-.144	-10.005	12.9848	12.9848	.2877	19.4613
.56E-07	RCVR	-1.0000	1.9585	19.3456	-.012	-.014	-.144	-10.005	12.9848	12.9848	.2877	19.4613
.15E-06	BACK UP0	-2.9064	-.0099	35.0706	-.009	-.012	-3.301	-2.477	23.5848	23.5848	.5220	35.3099
.15E-06	BACK UP1	-2.8979	.0000	34.8791	-.009	-.012	-3.304	-2.591	23.4576	23.4576	.5192	35.1183
.15E-06	BOTM REF	-2.8979	.0000	34.8791	-.009	.232	-3.304	3.400	23.4576	23.4576	.5192	35.1183
-.2E-07	BACK UP0	-.9966	1.6818	49.3975	-.080	.159	-.241	10.223	33.2576	33.2576	.7357	49.7665
-.2E-07	BACK UP1	-1.0000	1.6789	49.3788	-.080	.159	-.244	10.224	33.2447	33.2447	.7354	49.7475
-.2E-07	RCVR	-1.0000	1.6789	49.3788	-.080	.159	-.244	10.224	33.2447	33.2447	.7354	49.7475
.23E-07	APOGEE	-.0756	2.4244	58.2324	-.106	.129	.629	-.035	39.1947	39.1947	.8672	58.6633
.23E-07	WAVE REV	-.0756	2.4244	58.2324	-.106	.129	.629	-.035	39.1947	39.1947	.8672	58.6633
.36E-07	BACK UP0	-1.0048	1.6764	67.1054	-.126	.105	-.322	-10.189	45.1572	45.1572	.9993	67.5986
.36E-07	BACK UP1	-1.0000	1.6805	67.0788	-.126	.105	-.318	-10.188	45.1391	45.1391	.9989	67.5716
.36E-07	RCVR	-1.0000	1.6805	67.0788	-.126	.105	-.318	-10.188	45.1390	45.1391	.9989	67.5716
.15E-06	BACK UP0	-2.9053	-.0089	81.2985	-.145	.079	-1.722	-3.774	54.7390	54.7391	1.2111	81.9228
.15E-06	BACK UP1	-2.8954	.0000	81.1497	-.145	.079	-1.717	-3.861	54.6401	54.6401	1.2089	81.7738
.15E-06	BOTM REF	-2.8954	.0000	81.1497	-.145	-.210	-1.717	3.021	54.6401	54.6401	1.2089	81.7738
-.7E-07	BACK UP0	-.9843	1.9734	96.5829	-.113	-.197	-.436	9.501	65.0401	65.0401	1.4388	97.3300
-.7E-07	BACK UP1	-1.0000	1.9576	96.4893	-.113	-.197	-.445	9.505	64.9763	64.9763	1.4374	97.2352
-.7E-07	RCVR	-1.0000	1.9576	96.4893	-.113	-.197	-.445	9.505	64.9763	64.9763	1.4374	97.2352
-.3E-07	APOGEE	-.0990	2.8741	106.9837	-.094	-.208	-.009	-.028	72.0013	72.0013	1.5934	107.7855
-.3E-07	WAVE REV	-.0990	2.8741	106.9837	-.094	-.208	-.009	-.028	72.0013	72.0013	1.5934	107.7855
-.7E-08	BACK UP0	-1.0446	1.9373	117.8969	-.075	-.235	-.561	-9.220	79.3013	79.3013	1.7556	118.7567
-.7E-08	BACK UP1	-1.0000	1.9817	117.6220	-.076	-.234	-.538	-9.212	79.1147	79.1147	1.7515	118.4782
-.7E-08	RCVR	-1.0000	1.9817	117.6220	-.076	-.234	-.538	-9.212	79.1147	79.1147	1.7515	118.4782
.92E-07	BACK UP0	-3.0313	-.0430	132.7664	-.047	-.248	-1.482	-4.276	89.3147	89.3147	1.9774	133.7591
.92E-07	BACK UP1	-2.9882	.0000	132.2094	-.048	-.249	-1.465	-4.591	88.9443	88.9443	1.9691	133.2007
.92E-07	BOTM REF	-2.9882	.0000	132.2094	-.048	-.265	-1.465	4.554	88.9443	88.9443	1.9691	133.2007
.23E-07	BACK UP0	-.9627	2.0292	147.3613	-.019	-.297	-.656	8.687	99.1443	99.1443	2.1951	148.4868
.23E-07	BACK UP1	-1.0000	1.9918	147.1174	-.019	-.295	-.669	8.695	98.9793	98.9793	2.1914	148.2400
.23E-07	RCVR	-1.0000	1.9918	147.1174	-.019	-.295	-.669	8.695	98.9793	98.9793	2.1914	148.2400
.36E-07	APOGEE	-.0959	2.8977	158.6998	.005	-.344	-.394	-.027	106.7043	106.7043	2.3634	159.8741
.36E-07	WAVE REV	-.0959	2.8977	158.6998	.005	-.344	-.394	-.027	106.7043	106.7043	2.3634	159.8741
.58E-07	BACK UP0	-1.0199	1.9750	170.1974	.031	-.384	-.779	-8.869	114.3793	114.3793	2.5342	171.4256
.58E-07	BACK UP1	-1.0000	1.9949	170.0700	.031	-.383	-.771	-8.865	114.2929	114.2929	2.5323	171.2967
.58E-07	RCVR	-1.0000	1.9949	170.0700	.031	-.383	-.771	-8.865	114.2929	114.2929	2.5323	171.2967
.15E-06	BACK UP0	-2.9966	-.0005	186.1234	.068	-.392	-1.458	-3.356	125.0929	125.0929	2.7715	187.4766
.15E-06	BACK UP1	-2.9961	.0000	186.1151	.068	-.392	-1.458	-3.361	125.0875	125.0875	2.7713	187.4683
.15E-06	BOTM REF	-2.9961	.0000	186.1151	.068	-.394	-1.458	3.353	125.0875	125.0875	2.7713	187.4683
-.9E-07	BACK UP0	-.9762	2.0208	201.8456	.100	-.384	-.906	9.686	135.6875	135.6875	3.0059	203.3329
-.9E-07	BACK UP1	-1.0000	1.9969	201.7061	.100	-.384	-.913	9.692	135.5923	135.5923	3.0038	203.1913
-.9E-07	RCVR	-1.0000	1.9969	201.7061	.100	-.384	-.913	9.692	135.5923	135.5923	3.0038	203.1913
-.6E-07	MAX RANG	-.1257	2.8715	210.0424	.117	-.384	-.711	1.425	141.1923	141.1923	3.1278	211.5846

THIS RAY CALCULATION TOOK 5.782 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 12.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
-.7E-14	XMTR	-.9808	2.0000	.0000				12.000	.0000	.0000	.0000	.0000
.12E-06	APOGEE	-.0412	2.9353	5.7848	-.013	-.013	9.200	-.152	3.9188	3.9188	.0867	5.8661
.12E-06	WAVE REV	-.0412	2.9353	5.7848	-.013	-.013	9.200	-.152	3.9188	3.9188	.0867	5.8661
.46E-07	BACK UP0	-1.0051	1.9653	11.6687	-.013	-.014	-.172	-12.005	7.9063	7.9063	.1749	11.8339
.46E-07	BACK UP1	-1.0000	1.9705	11.6447	-.013	-.014	-.147	-12.004	7.8897	7.8897	.1746	11.8093
.46E-07	RCVR	-1.0000	1.9705	11.6447	-.013	-.014	-.147	-12.004	7.8897	7.8897	.1746	11.8093
.22E-06	BACK UP0	-2.9612	-.0106	22.8294	-.010	-.011	-5.062	-6.845	15.4897	15.4897	.3425	23.1658
.22E-06	BACK UP1	-2.9508	.0000	22.7433	-.010	-.011	-5.054	-6.896	15.4322	15.4322	.3412	23.0790
.22E-06	BOTM REF	-2.9508	.0000	22.7433	-.010	.193	-5.054	7.184	15.4322	15.4322	.3412	23.0790
-.7E-07	BACK UP0	-.9946	1.9131	33.6244	-.075	.127	-.175	12.168	22.8322	22.8322	.5046	34.1354
-.7E-07	BACK UP1	-1.0000	1.9079	33.5994	-.075	.127	-.184	12.169	22.8150	22.8150	.5042	34.1099
-.7E-07	RCVR	-1.0000	1.9079	33.5994	-.075	.127	-.184	12.169	22.8150	22.8150	.5042	34.1099
.71E-07	APOGEE	-.0394	2.8222	39.3422	-.096	.105	1.194	-.089	26.7087	26.7088	.5904	39.9378
.71E-07	WAVE REV	-.0394	2.8222	39.3422	-.096	.105	1.194	-.089	26.7087	26.7088	.5904	39.9378
.64E-07	BACK UP0	-1.0486	1.7275	45.3025	-.112	.087	-.290	-12.169	30.7525	30.7525	.6798	45.9883
.64E-07	BACK UP1	-1.0000	1.7803	45.0772	-.111	.088	-.227	-12.165	30.5973	30.5973	.6764	45.7579
.64E-07	RCVR	-1.0000	1.7803	45.0772	-.111	.088	-.227	-12.165	30.5973	30.5973	.6764	45.7579
.26E-06	BACK UP0	-2.5803	-.0016	53.2774	-.127	.071	-1.960	-8.780	36.1973	36.1973	.7999	54.1081
.26E-06	BACK UP1	-2.5790	.0000	53.2686	-.127	.071	-1.959	-8.785	36.1914	36.1914	.7997	54.0993
.26E-06	BOTM REF	-2.5790	.0000	53.2686	-.127	3.158	-1.959	11.314	36.1914	36.1914	.7997	54.0993
.44E-06	BACK UP0	-.9955	1.5051	60.0872	-.487	2.796	-.284	14.059	40.8914	40.8914	.9033	61.1072
.44E-06	BACK UP1	-1.0000	1.5006	60.0692	-.486	2.797	-.288	14.060	40.8789	40.8790	.9031	61.0887
.44E-06	RCVR	-1.0000	1.5006	60.0692	-.486	2.797	-.288	14.060	40.8789	40.8790	.9031	61.0887
.61E-06	APOGEE	-.0153	2.5149	65.0273	-.701	2.579	.558	-.032	44.2602	44.2602	.9780	66.1579
.61E-06	WAVE REV	-.0153	2.5149	65.0273	-.701	2.579	.558	-.032	44.2602	44.2602	.9780	66.1579
.48E-06	BACK UP0	-1.0314	1.5847	70.1034	-.890	2.388	-.357	-14.043	47.7227	47.7227	1.0547	71.3482
.48E-06	BACK UP1	-1.0000	1.6137	69.9780	-.885	2.392	-.330	-14.041	47.6356	47.6357	1.0528	71.2189
.48E-06	RCVR	-1.0000	1.6137	69.9780	-.885	2.392	-.330	-14.041	47.6356	47.6357	1.0528	71.2189
.34E-06	BACK UP0	-2.7903	-.0398	77.8333	-1.128	2.145	-1.682	-10.807	53.0356	53.0357	1.1720	79.2807
.34E-06	BACK UP1	-2.7472	.0000	77.6090	-1.122	2.151	-1.653	-10.938	52.8837	52.8837	1.1686	79.0521
.34E-06	BOTM REF	-2.7472	.0000	77.6090	-1.122	-.362	-1.653	8.925	52.8837	52.8837	1.1686	79.0521
-.3E-07	BACK UP0	-.9517	1.9216	86.6909	-1.085	-.335	-.371	12.417	59.0837	59.0837	1.3055	88.3091
-.3E-07	BACK UP1	-1.0000	1.8714	86.4716	-1.086	-.335	-.402	12.427	58.9327	58.9327	1.3021	88.0845
-.3E-07	RCVR	-1.0000	1.8714	86.4716	-1.086	-.335	-.402	12.427	58.9327	58.9327	1.3021	88.0845
.18E-06	APOGEE	-.0340	2.8762	91.9411	-1.067	-.327	.177	-.165	62.6420	62.6421	1.3843	93.6435
.18E-06	WAVE REV	-.0340	2.8762	91.9411	-1.067	-.327	.177	-.165	62.6420	62.6421	1.3843	93.6435
.29E-08	BACK UP0	-1.0292	1.9058	97.5448	-1.050	-.323	-.467	-12.359	66.4420	66.4421	1.4685	99.3395
.29E-08	BACK UP1	-1.0000	1.9346	97.4114	-1.050	-.323	-.449	-12.356	66.3503	66.3503	1.4665	99.2029
.29E-08	RCVR	-1.0000	1.9346	97.4114	-1.050	-.323	-.449	-12.356	66.3503	66.3503	1.4665	99.2029
-.5E-07	BACK UP0	-3.1293	-.1675	108.5873	-1.017	-.308	-1.622	-8.028	73.9503	73.9503	1.6347	110.5796
-.5E-07	BACK UP1	-2.9598	.0000	107.4338	-1.020	-.311	-1.539	-8.693	73.1776	73.1776	1.6175	109.4142
-.5E-07	BOTM REF	-2.9598	.0000	107.4338	-1.020	-.497	-1.539	8.489	73.1776	73.1776	1.6175	109.4142
-.4E-06	BACK UP0	-.9391	2.0348	118.0199	-.976	-.483	-.511	11.815	80.3776	80.3776	1.7768	120.1904
-.4E-06	BACK UP1	-1.0000	1.9735	117.7288	-.977	-.482	-.539	11.826	80.1784	80.1784	1.7724	119.8930
-.4E-06	RCVR	-1.0000	1.9735	117.7288	-.977	-.482	-.539	11.826	80.1784	80.1784	1.7724	119.8930
-.2E-06	APOGEE	-.0340	2.9443	123.3849	-.955	-.490	-.115	-.161	84.0003	84.0003	1.8573	125.6353
-.2E-06	WAVE REV	-.0340	2.9443	123.3849	-.955	-.490	-.115	-.161	84.0003	84.0003	1.8573	125.6353
-.3E-06	BACK UP0	-1.0001	1.9818	129.0489	-.934	-.502	-.589	-11.733	87.8253	87.8253	1.9423	131.3851

-.3E-06 RCVR	-1.0000	1.9819	129.0483	-.934	-.502	-.589	-11.733	87.8248	87.8248	1.9422	131.3844
-.2E-06 BACK UP0	-3.0530	-.0663	139.6415	-.894	-.501	-1.478	-8.849	95.0248	95.0248	2.1017	142.1732
-.2E-06 BACK UP1	-2.9865	.0000	139.2201	-.896	-.502	-1.452	-9.086	94.7420	94.7420	2.0954	141.7468
-.2E-06 BOTM REF	-2.9865	.0000	139.2201	-.896	-.541	-1.452	9.044	94.7420	94.7420	2.0954	141.7468
-.3E-06 BACK UP0	-.9906	1.9991	149.5198	-.858	-.546	-.676	11.541	101.7420	101.7420	2.2505	152.2364
-.3E-06 BACK UP1	-1.0000	1.9896	149.4738	-.858	-.546	-.680	11.543	101.7106	101.7106	2.2498	152.1894
-.3E-06 RCVR	-1.0000	1.9896	149.4738	-.858	-.546	-.680	11.543	101.7106	101.7106	2.2498	152.1894
-.1E-06 APOGEE	-.0338	2.9571	155.2048	-.837	-.565	-.348	-.071	105.5762	105.5762	2.3358	158.0053
-.1E-06 WAVE REV	-.0338	2.9571	155.2048	-.837	-.565	-.348	-.071	105.5762	105.5762	2.3358	158.0053
-.1E-06 BACK UP0	-1.0355	1.9565	161.0943	-.815	-.583	-.744	-11.584	109.5512	109.5512	2.4242	163.9834
-.1E-06 BACK UP1	-1.0000	1.9920	160.9212	-.816	-.582	-.731	-11.578	109.4330	109.4331	2.4215	163.8067
-.1E-06 RCVR	-1.0000	1.9920	160.9212	-.816	-.582	-.731	-11.578	109.4330	109.4331	2.4215	163.8067
-.8E-07 BACK UP0	-3.0227	-.0291	171.5190	-.777	-.583	-1.454	-8.655	116.6330	116.6331	2.5810	174.5945
-.8E-07 BACK UP1	-2.9935	.0000	171.3287	-.778	-.584	-1.444	-8.763	116.5054	116.5054	2.5782	174.4020
-.8E-07 BOTM REF	-2.9935	.0000	171.3287	-.778	-.596	-1.444	8.749	116.5054	116.5054	2.5782	174.4020
-.4E-06 BACK UP0	-.9419	2.0527	181.9110	-.742	-.591	-.806	11.917	123.7054	123.7054	2.7375	185.1802
-.4E-06 BACK UP1	-1.0000	1.9946	181.6361	-.743	-.591	-.823	11.929	123.5170	123.5170	2.7334	184.8993
-.4E-06 RCVR	-1.0000	1.9946	181.6361	-.743	-.591	-.823	11.929	123.5170	123.5170	2.7334	184.8993
-.2E-06 APOGEE	-.0350	2.9601	187.2827	-.724	-.596	-.553	-.091	127.3358	127.3358	2.8181	190.6324
-.2E-06 WAVE REV	-.0350	2.9601	187.2827	-.724	-.596	-.553	-.091	127.3358	127.3358	2.8181	190.6324
-.4E-06 BACK UP0	-1.0231	1.9725	193.0067	-.705	-.597	-.881	-12.033	131.2108	131.2108	2.9040	196.4457
-.4E-06 BACK UP1	-1.0000	1.9956	192.8983	-.706	-.597	-.873	-12.030	131.1364	131.1364	2.9024	196.3350
-.4E-06 RCVR	-1.0000	1.9956	192.8983	-.706	-.597	-.873	-12.030	131.1364	131.1364	2.9024	196.3350
-.5E-06 BACK UP0	-3.0224	-.0261	203.7839	-.672	-.582	-1.491	-7.816	138.5364	138.5364	3.0661	207.4069
-.5E-06 BACK UP1	-2.9962	.0000	203.5948	-.673	-.582	-1.483	-7.926	138.4099	138.4099	3.0633	207.2161
-.5E-06 BOTM REF	-2.9962	.0000	203.5948	-.673	-.588	-1.483	7.919	138.4099	138.4099	3.0633	207.2161
-.5E-06 MAX RANG	-1.8778	1.1188	210.0987	-.655	-.572	-1.190	11.474	142.8099	142.8099	3.1608	213.8144

THIS RAY CALCULATION TOOK 9.265 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 14.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
-.7E-14	XMTR	-.9808	2.0000	.0000				14.000	.0000	.0000	.0000	.0000
.96E-07	APOGEE	-.0187	2.9586	4.7902	-.013	-.013	11.335	-.054	3.2656	3.2656	.0723	4.8926
.96E-07	WAVE REV	-.0187	2.9586	4.7902	-.013	-.013	11.335	-.054	3.2656	3.2656	.0723	4.8926
.17E-07	BACK UP0	-1.0247	1.9479	9.7464	-.013	-.014	-.302	-14.006	6.6469	6.6469	.1472	9.9565
.17E-07	BACK UP1	-1.0000	1.9727	9.6474	-.013	-.014	-.158	-14.003	6.5782	6.5782	.1457	9.8545
.17E-07	RCVR	-1.0000	1.9727	9.6474	-.013	-.014	-.158	-14.003	6.5782	6.5782	.1457	9.8545
-.1E-06	BACK UP0	-3.0369	-.0777	18.9927	-.010	-.011	-6.266	-9.710	12.9782	12.9782	.2871	19.4189
-.1E-06	BACK UP1	-2.9600	.0000	18.5493	-.010	-.012	-6.176	-9.973	12.6798	12.6798	.2804	18.9691
-.1E-06	BOTM REF	-2.9600	.0000	18.5493	-.010	.206	-6.176	10.186	12.6798	12.6798	.2804	18.9691
.74E-07	BACK UP0	-.9700	1.9671	27.4412	-.080	.136	-.101	14.149	18.7798	18.7798	.4151	28.0804
.74E-07	BACK UP1	-1.0000	1.9375	27.3220	-.079	.137	-.163	14.155	18.6970	18.6970	.4133	27.9576
.74E-07	RCVR	-1.0000	1.9375	27.3220	-.079	.137	-.163	14.155	18.6970	18.6970	.4133	27.9576
.20E-06	APOGEE	-.0128	2.9026	32.4041	-.103	.112	1.565	-.017	22.1564	22.1564	.4900	33.1460
.20E-06	WAVE REV	-.0128	2.9026	32.4041	-.103	.112	1.565	-.017	22.1564	22.1564	.4900	33.1460
.48E-07	BACK UP0	-1.0218	1.8580	37.5620	-.121	.094	-.232	-14.155	25.6689	25.6689	.5679	38.4130
.48E-07	BACK UP1	-1.0000	1.8806	37.4756	-.120	.094	-.198	-14.153	25.6089	25.6089	.5665	38.3239
.48E-07	RCVR	-1.0000	1.8806	37.4756	-.120	.094	-.198	-14.153	25.6089	25.6089	.5665	38.3239
-.1E-06	BACK UP0	-2.8475	-.0754	45.6301	-.139	.074	-2.549	-10.561	31.2089	31.2089	.6901	46.6842
-.1E-06	BACK UP1	-2.7790	.0000	45.2665	-.138	.075	-2.479	-10.776	30.9631	30.9631	.6847	46.3144
-.1E-06	BOTM REF	-2.7790	.0000	45.2665	-.138	2.653	-2.479	12.635	30.9631	30.9631	.6847	46.3144
.49E-07	BACK UP0	-.9883	1.6727	52.1966	-.493	2.298	-.243	15.586	35.7631	35.7631	.7906	53.4774
.49E-07	BACK UP1	-1.0000	1.6618	52.1548	-.491	2.300	-.256	15.588	35.7338	35.7338	.7899	53.4340
.49E-07	RCVR	-1.0000	1.6618	52.1548	-.491	2.300	-.256	15.588	35.7338	35.7338	.7899	53.4340
.30E-06	BACK UP0	.0001	2.5789	56.2932	-.661	2.128	.745	6.516	38.5807	38.5807	.8529	57.6975
.30E-06	SURF REF	.0000	2.5788	56.2926	-.661	2.128	.745	-6.517	38.5803	38.5803	.8529	57.6968
.17E-06	BACK UP0	-1.0249	1.4888	60.5211	-.811	1.976	-.314	-15.583	41.4896	41.4896	.9173	62.0534
.17E-06	BACK UP1	-1.0000	1.5146	60.4317	-.808	1.979	-.290	-15.580	41.4271	41.4271	.9160	61.9605
.17E-06	RCVR	-1.0000	1.5146	60.4317	-.808	1.979	-.290	-15.580	41.4271	41.4271	.9160	61.9605
.89E-07	BACK UP0	-2.5211	-.0041	66.1951	-.982	1.803	-1.631	-13.349	45.4271	45.4271	1.0041	67.9233
.89E-07	BACK UP1	-2.5168	.0000	66.1771	-.981	1.803	-1.628	-13.359	45.4148	45.4148	1.0038	67.9048
.89E-07	BOTM REF	-2.5168	.0000	66.1771	-.981	-.413	-1.628	11.913	45.4148	45.4148	1.0038	67.9048
.28E-06	BACK UP0	-.9877	1.6794	72.5572	-.947	-.383	-.332	14.334	49.8148	49.8148	1.1008	74.4645
.28E-06	BACK UP1	-1.0000	1.6659	72.5091	-.947	-.383	-.341	14.335	49.7813	49.7813	1.1001	74.4149
.28E-06	RCVR	-1.0000	1.6659	72.5091	-.947	-.383	-.341	14.335	49.7813	49.7813	1.1001	74.4149
.34E-06	BACK UP0	.0005	2.7753	77.3972	-.924	-.365	.378	2.609	53.1126	53.1126	1.1739	79.4122
.34E-06	BACK UP1	.0000	2.7745	77.3854	-.924	-.365	.378	2.616	53.1049	53.1049	1.1738	79.4004
.34E-06	SURF REF	.0000	2.7745	77.3854	-.924	-.365	.378	-2.616	53.1049	53.1049	1.1738	79.4004
.27E-06	BACK UP0	-1.0133	1.8339	82.3193	-.903	-.351	-.393	-14.303	56.4674	56.4674	1.2483	84.4451
.27E-06	BACK UP1	-1.0000	1.8466	82.2672	-.904	-.351	-.383	-14.302	56.4312	56.4312	1.2476	84.3914
.27E-06	RCVR	-1.0000	1.8466	82.2672	-.904	-.351	-.383	-14.302	56.4312	56.4312	1.2476	84.3914
.17E-06	BACK UP0	-3.0095	-.0939	91.0097	-.871	-.326	-1.687	-10.699	62.4312	62.4312	1.3802	93.3610
.17E-06	BACK UP1	-2.9130	.0000	90.5056	-.872	-.328	-1.630	-10.993	62.0908	62.0908	1.3726	92.8480
.17E-06	BOTM REF	-2.9130	.0000	90.5056	-.872	-1.030	-1.630	10.348	62.0907	62.0908	1.3726	92.8480
.32E-06	BACK UP0	-.9750	1.9735	99.2631	-.783	-.955	-.443	13.618	68.0907	68.0908	1.5052	101.8174
.32E-06	BACK UP1	-1.0000	1.9482	99.1598	-.784	-.956	-.457	13.622	68.0193	68.0194	1.5036	101.7111
.32E-06	RCVR	-1.0000	1.9482	99.1598	-.784	-.956	-.457	13.622	68.0193	68.0194	1.5036	101.7111
.41E-06	APOGEE	-.0169	2.9429	104.1662	-.738	-.926	.062	-.051	71.4225	71.4225	1.5791	106.8209
.41E-06	WAVE REV	-.0169	2.9429	104.1662	-.738	-.926	.062	-.051	71.4225	71.4225	1.5791	106.8209

.30E-06	BACK UP0	-1.0149	1.9532	109.2339	-.696	-.904	-.509	-13.549	74.8662	74.8662	1.6556	111.9936
.30E-06	BACK UP1	-1.0000	1.9680	109.1722	-.697	-.904	-.501	-13.547	74.8237	74.8237	1.6547	111.9302
.30E-06	RCVR	-1.0000	1.9680	109.1722	-.697	-.904	-.501	-13.547	74.8237	74.8237	1.6547	111.9302
.16E-06	BACK UP0	-3.0477	-.0699	118.3782	-.626	-.857	-1.533	-10.470	81.1237	81.1237	1.7941	121.3606
.16E-06	BACK UP1	-2.9775	.0000	118.0022	-.629	-.859	-1.500	-10.685	80.8701	80.8701	1.7884	120.9783
.16E-06	BOTM REF	-2.9775	.0000	118.0022	-.629	-.958	-1.500	10.589	80.8701	80.8701	1.7884	120.9783
.45E-06	BACK UP0	-.9749	2.0086	127.0682	-.561	-.922	-.569	13.179	87.0701	87.0701	1.9257	130.2621
.45E-06	BACK UP1	-1.0000	1.9835	126.9610	-.561	-.922	-.580	13.183	86.9964	86.9964	1.9240	130.1520
.45E-06	RCVR	-1.0000	1.9835	126.9610	-.561	-.922	-.580	13.183	86.9964	86.9964	1.9240	130.1520
.55E-06	APOGEE	-.0149	2.9711	132.1571	-.525	-.918	-.176	-.035	90.5151	90.5152	2.0023	135.4495
.55E-06	WAVE REV	-.0149	2.9711	132.1571	-.525	-.918	-.176	-.035	90.5151	90.5152	2.0023	135.4495
.43E-06	BACK UP0	-1.0311	1.9569	137.4822	-.489	-.918	-.639	-13.131	94.1214	94.1214	2.0826	140.8793
.43E-06	BACK UP1	-1.0000	1.9879	137.3491	-.490	-.917	-.626	-13.126	94.0299	94.0299	2.0806	140.7425
.43E-06	RCVR	-1.0000	1.9879	137.3491	-.490	-.917	-.626	-13.126	94.0299	94.0299	2.0806	140.7425
.30E-06	BACK UP0	-3.0492	-.0587	146.5659	-.431	-.896	-1.468	-10.741	100.3299	100.3299	2.2202	150.1834
.30E-06	BACK UP1	-2.9904	.0000	146.2585	-.433	-.898	-1.445	-10.913	100.1225	100.1225	2.2155	149.8706
.30E-06	BOTM REF	-2.9904	.0000	146.2585	-.433	-.926	-1.445	10.886	100.1225	100.1225	2.2155	149.8706
.43E-06	BACK UP0	-.9683	2.0240	155.3259	-.378	-.911	-.694	13.082	106.3225	106.3225	2.3529	159.1598
.43E-06	BACK UP1	-1.0000	1.9922	155.1897	-.379	-.910	-.705	13.086	106.2289	106.2289	2.3508	159.0199
.43E-06	RCVR	-1.0000	1.9922	155.1897	-.379	-.910	-.705	13.086	106.2289	106.2289	2.3508	159.0199
.51E-06	APOGEE	-.0156	2.9775	160.3739	-.348	-.915	-.376	-.042	109.7383	109.7383	2.4289	164.3050
.51E-06	WAVE REV	-.0156	2.9775	160.3739	-.348	-.915	-.376	-.042	109.7383	109.7383	2.4289	164.3050
.37E-06	BACK UP0	-1.0315	1.9623	165.6690	-.318	-.919	-.763	-13.138	113.3258	113.3258	2.5087	169.7047
.37E-06	BACK UP1	-1.0000	1.9938	165.5342	-.319	-.919	-.751	-13.134	113.2332	113.2332	2.5067	169.5663
.37E-06	RCVR	-1.0000	1.9938	165.5342	-.319	-.919	-.751	-13.134	113.2332	113.2332	2.5067	169.5663
.13E-06	BACK UP0	-3.0789	-.0841	175.0485	-.267	-.900	-1.474	-10.298	119.7332	119.7332	2.6506	179.3045
.13E-06	BACK UP1	-2.9948	.0000	174.5912	-.269	-.903	-1.446	-10.560	119.4250	119.4250	2.6438	178.8397
.13E-06	BOTM REF	-2.9948	.0000	174.5912	-.269	-.914	-1.446	10.549	119.4250	119.4250	2.6438	178.8397
.33E-06	BACK UP0	-.9844	2.0111	183.6509	-.223	-.893	-.827	13.411	125.6250	125.6250	2.7810	188.1192
.33E-06	BACK UP1	-1.0000	1.9956	183.5857	-.224	-.893	-.832	13.413	125.5800	125.5800	2.7800	188.0522
.33E-06	RCVR	-1.0000	1.9956	183.5857	-.224	-.893	-.832	13.413	125.5800	125.5800	2.7800	188.0522
.46E-06	APOGEE	-.0192	2.9767	188.5766	-.199	-.889	-.556	-.040	128.9706	128.9707	2.8553	193.1451
.46E-06	WAVE REV	-.0192	2.9767	188.5766	-.199	-.889	-.556	-.040	128.9706	128.9707	2.8553	193.1451
.32E-06	BACK UP0	-1.0073	1.9890	193.5774	-.175	-.882	-.878	-13.490	132.3706	132.3707	2.9307	198.2490
.32E-06	BACK UP1	-1.0000	1.9963	193.5470	-.175	-.882	-.876	-13.489	132.3496	132.3497	2.9302	198.2177
.32E-06	RCVR	-1.0000	1.9963	193.5470	-.175	-.882	-.876	-13.489	132.3496	132.3497	2.9302	198.2177
.20E-06	BACK UP0	-2.9979	-.0011	202.7558	-.134	-.857	-1.482	-9.970	138.6496	138.6497	3.0695	207.6408
.20E-06	BACK UP1	-2.9968	.0000	202.7494	-.134	-.857	-1.482	-9.974	138.6453	138.6453	3.0694	207.6343
.20E-06	BOTM REF	-2.9968	.0000	202.7494	-.134	-.862	-1.482	9.968	138.6453	138.6453	3.0694	207.6343
.38E-06	MAX RANG	-1.4448	1.5522	210.0720	-.104	-.837	-1.071	13.577	143.6453	143.6453	3.1801	215.1191

THIS RAY CALCULATION TOOK 13.113 SEC

AZIMUTH ANGLE OF TRANSMISSION = 80.000000 DEG TRANSMITTER LATITUDE = .000000 DEG FREQUENCY = 400.000000 HZ
ELEVATION ANGLE OF TRANSMISSION = 16.000000 DEG TRANSMITTER LONGITUDE = .000000 DEG SINGLE STEP ERROR = 1.000000E-06

ERROR	EVENT	ELEVATION ABOVE SEA LEVEL KM	ELEVATION ABOVE TERRAIN KM	RANGE KM	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION ANGLE XMTR DEG	ELEVATION ANGLE LOCAL DEG	PULSE TIME SEC	PHASE TIME SEC	ABSORPTION DB	PATH LENGTH KM
-.7E-14	XMTR	-.9808	2.0000	.0000				16.000	.0000	.0000	.0000	.0000
.15E-06	BACK UP0	.0012	2.9792	3.9245	-.013	-.013	14.031	7.414	2.7031	2.7031	.0598	4.0479
.15E-06	SURF REF	.0000	2.9780	3.9152	-.013	-.014	14.047	-7.420	2.6970	2.6970	.0597	4.0386
-.3E-07	BACK UP0	-1.0012	1.9733	7.9016	-.013	-.014	-.184	-16.003	5.4439	5.4439	.1205	8.1511
-.3E-07	BACK UP1	-1.0000	1.9745	7.8975	-.013	-.014	-.175	-16.003	5.4410	5.4410	.1204	8.1469
-.3E-07	RCVR	-1.0000	1.9745	7.8975	-.013	-.014	-.175	-16.003	5.4410	5.4410	.1204	8.1469
-.3E-06	BACK UP0	-3.0667	-.1021	15.8535	-.010	-.011	-7.569	-12.367	10.9410	10.9410	.2419	16.3661
-.3E-06	BACK UP1	-2.9653	.0000	15.3961	-.010	-.012	-7.416	-12.638	10.6304	10.6304	.2350	15.8979
-.3E-06	BOTM REF	-2.9653	.0000	15.3961	-.010	.213	-7.416	12.811	10.6304	10.6304	.2350	15.8979
-.4E-07	BACK UP0	-.9743	1.9765	22.9038	-.083	.140	-.087	16.135	15.8304	15.8304	.3498	23.6641
-.4E-07	BACK UP1	-1.0000	1.9510	22.8151	-.083	.140	-.151	16.139	15.7682	15.7682	.3485	23.5718
-.4E-07	RCVR	-1.0000	1.9510	22.8151	-.083	.140	-.151	16.139	15.7682	15.7682	.3485	23.5718
.15E-06	BACK UP0	.0008	2.9405	26.7533	-.105	.117	1.981	7.710	18.4838	18.4838	.4086	27.6375
.15E-06	SURF REF	.0000	2.9398	26.7475	-.105	.117	1.980	-7.714	18.4800	18.4800	.4085	27.6316
.46E-07	BACK UP0	-1.0119	1.9123	30.7211	-.123	.099	-.196	-16.140	21.2206	21.2206	.4691	31.7344
.46E-07	BACK UP1	-1.0000	1.9244	30.6799	-.122	.099	-.174	-16.138	21.1918	21.1918	.4685	31.6915
.46E-07	RCVR	-1.0000	1.9244	30.6799	-.122	.099	-.174	-16.138	21.1918	21.1918	.4685	31.6915
-.2E-06	BACK UP0	-2.9529	-.0766	38.0433	-.144	.077	-3.140	-12.847	26.2918	26.2918	.5811	39.3085
-.2E-06	BACK UP1	-2.8791	.0000	37.7222	-.143	.078	-3.051	-13.037	26.0731	26.0731	.5762	38.9791
-.2E-06	BOTM REF	-2.8791	.0000	37.7222	-.143	1.485	-3.051	13.972	26.0731	26.0731	.5762	38.9792
-.3E-06	BACK UP0	-.9705	1.8400	44.4731	-.370	1.257	-.187	16.885	30.7731	30.7731	.6799	45.9955
-.3E-06	BACK UP1	-1.0000	1.8118	44.3760	-.367	1.260	-.224	16.889	30.7047	30.7047	.6784	45.8940
-.3E-06	RCVR	-1.0000	1.8118	44.3760	-.367	1.260	-.224	16.889	30.7047	30.7047	.6784	45.8940
-.2E-06	BACK UP0	.0013	2.7560	48.0667	-.465	1.160	.954	9.224	33.2610	33.2610	.7350	49.7207
-.2E-06	BACK UP1	.0000	2.7549	48.0587	-.465	1.161	.953	9.230	33.2556	33.2556	.7349	49.7125
-.2E-06	SURF REF	.0000	2.7549	48.0587	-.465	1.161	.953	-9.230	33.2556	33.2556	.7349	49.7125
-.4E-06	BACK UP0	-1.0036	1.6768	51.7536	-.549	1.075	-.258	-16.887	35.8150	35.8150	.7915	53.5437
-.4E-06	BACK UP1	-1.0000	1.6806	51.7418	-.549	1.075	-.254	-16.886	35.8066	35.8066	.7914	53.5313
-.4E-06	RCVR	-1.0000	1.6806	51.7418	-.549	1.075	-.254	-16.886	35.8066	35.8066	.7914	53.5313
-.3E-06	BACK UP0	-2.6086	-.0553	57.3303	-.655	.968	-1.885	-14.626	39.7066	39.7066	.8773	59.3466
-.3E-06	BACK UP1	-2.5573	.0000	57.1345	-.652	.971	-1.838	-14.741	39.5718	39.5718	.8743	59.1442
-.3E-06	BOTM REF	-2.5573	.0000	57.1345	-.652	5.462	-1.838	16.468	39.5718	39.5718	.8743	59.1442
-.5E-06	BACK UP0	-.9704	1.5474	62.0851	-1.090	5.022	-.270	18.379	43.0718	43.0718	.9515	64.3618
-.5E-06	BACK UP1	-1.0000	1.5183	61.9964	-1.083	5.030	-.297	18.383	43.0087	43.0087	.9501	64.2680
-.5E-06	RCVR	-1.0000	1.5183	61.9964	-1.083	5.030	-.297	18.383	43.0087	43.0087	.9501	64.2680
-.4E-06	BACK UP0	.0009	2.5044	65.2726	-1.337	4.774	.568	11.848	45.3055	45.3055	1.0009	67.7063
-.4E-06	BACK UP1	.0000	2.5035	65.2683	-1.337	4.774	.567	11.851	45.3027	45.3027	1.0008	67.7019
-.4E-06	SURF REF	.0000	2.5035	65.2683	-1.337	4.774	.567	-11.851	45.3027	45.3027	1.0008	67.7019
-.9E-06	BACK UP0	-1.0466	1.4539	68.6820	-1.576	4.533	-.364	-18.378	47.6964	47.6964	1.0538	71.2844
-.9E-06	BACK UP1	-1.0000	1.5003	68.5421	-1.566	4.543	-.324	-18.376	47.5969	47.5969	1.0516	71.1365
-.9E-06	RCVR	-1.0000	1.5003	68.5421	-1.566	4.543	-.324	-18.376	47.5969	47.5969	1.0516	71.1365
-.7E-06	BACK UP0	-2.5438	-.0265	73.3576	-1.866	4.240	-1.551	-16.557	50.9969	50.9969	1.1265	76.2059
-.7E-06	BOTM REF	-2.5168	.0000	73.2669	-1.861	1.370	-1.531	15.683	50.9337	50.9337	1.1251	76.1111
-.9E-06	BACK UP0	-.9569	1.6394	78.4024	-1.952	1.275	-.335	17.512	54.5337	54.5337	1.2045	81.4786
-.9E-06	BACK UP1	-1.0000	1.5937	78.2658	-1.950	1.277	-.366	17.518	54.4373	54.4373	1.2024	81.3353
-.9E-06	RCVR	-1.0000	1.5937	78.2658	-1.950	1.277	-.366	17.518	54.4373	54.4373	1.2024	81.3353
-.8E-06	BACK UP0	.0015	2.6614	81.7602	-2.005	1.217	.321	10.574	56.8654	56.8654	1.2561	84.9725
-.8E-06	BACK UP1	.0000	2.6597	81.7521	-2.005	1.217	.320	10.579	56.8600	56.8600	1.2560	84.9642

-.8E-06	SURF REF	.0000	2.6597	81.7521	-2.005	1.217	.320	-10.579	56.8600	56.8600	1.2560	84.9642
-.9E-06	BACK UP0	-1.0329	1.6899	85.3445	-2.057	1.160	-.419	-17.502	59.3569	59.3569	1.3113	88.7042
-.9E-06	BACK UP1	-1.0000	1.7211	85.2403	-2.056	1.161	-.396	-17.499	59.2834	59.2834	1.3097	88.5949
-.9E-06	RCVR	-1.0000	1.7211	85.2403	-2.056	1.161	-.396	-17.499	59.2834	59.2834	1.3097	88.5949
-.8E-06	BACK UP0	-2.9199	-.1112	91.6847	-2.138	1.072	-1.624	-14.995	63.7834	63.7834	1.4091	95.3191
-.8E-06	BACK UP1	-2.8039	.0000	91.2552	-2.133	1.077	-1.555	-15.243	63.4879	63.4879	1.4025	94.8744
-.8E-06	BOTM REF	-2.8039	.0000	91.2552	-2.133	-1.725	-1.555	13.679	63.4879	63.4879	1.4025	94.8744
-.6E-06	BACK UP0	-.9975	1.8842	97.8853	-2.017	-1.620	-.450	16.050	68.0879	68.0879	1.5041	101.7474
-.6E-06	BACK UP1	-1.0000	1.8817	97.8766	-2.017	-1.620	-.451	16.051	68.0819	68.0819	1.5040	101.7384
-.6E-06	RCVR	-1.0000	1.8817	97.8766	-2.017	-1.620	-.451	16.051	68.0819	68.0819	1.5040	101.7384
-.5E-06	BACK UP0	.0006	2.9096	101.7693	-1.956	-1.571	.095	8.324	70.7631	70.7631	1.5635	105.7608
-.5E-06	SURF REF	.0000	2.9090	101.7652	-1.956	-1.571	.095	-8.326	70.7605	70.7604	1.5634	105.7567
-.6E-06	BACK UP0	-1.0092	1.9192	105.6910	-1.899	-1.527	-.491	-16.009	73.4636	73.4636	1.6234	109.8132
-.6E-06	BACK UP1	-1.0000	1.9283	105.6589	-1.899	-1.527	-.486	-16.008	73.4412	73.4412	1.6229	109.7798
-.6E-06	RCVR	-1.0000	1.9283	105.6589	-1.899	-1.527	-.486	-16.008	73.4412	73.4412	1.6229	109.7798
-.9E-06	BACK UP0	-2.9664	-.0150	112.8754	-1.802	-1.447	-1.516	-13.645	78.4412	78.4412	1.7335	117.2602
-.9E-06	BACK UP1	-2.9512	.0000	112.8133	-1.803	-1.447	-1.508	-13.681	78.3987	78.3987	1.7325	117.1962
-.9E-06	BOTM REF	-2.9512	.0000	112.8133	-1.803	-1.841	-1.508	13.397	78.3987	78.3987	1.7325	117.1962
-.7E-06	BACK UP0	-.9838	1.9819	120.1875	-1.690	-1.751	-.542	15.559	83.4987	83.4987	1.8454	124.8302
-.7E-06	BACK UP1	-1.0000	1.9656	120.1295	-1.691	-1.752	-.549	15.561	83.4584	83.4583	1.8445	124.7699
-.7E-06	RCVR	-1.0000	1.9656	120.1295	-1.691	-1.752	-.549	15.561	83.4584	83.4583	1.8445	124.7699
-.6E-06	BACK UP0	.0005	2.9715	124.1227	-1.635	-1.718	-.105	8.084	86.1990	86.1990	1.9054	128.8899
-.6E-06	SURF REF	.0000	2.9710	124.1192	-1.635	-1.718	-.105	-8.086	86.1967	86.1967	1.9053	128.8864
-.9E-06	BACK UP0	-1.0362	1.9391	128.2447	-1.580	-1.688	-.602	-15.522	89.0280	89.0280	1.9683	133.1434
-.9E-06	BACK UP1	-1.0000	1.9752	128.1144	-1.581	-1.688	-.585	-15.517	88.9375	88.9375	1.9663	133.0082
-.9E-06	RCVR	-1.0000	1.9752	128.1144	-1.581	-1.688	-.585	-15.517	88.9375	88.9375	1.9663	133.0082
-.1E-05	BACK UP0	-3.0907	-.1095	135.9303	-1.484	-1.619	-1.501	-13.415	94.3375	94.3375	2.0859	141.1000
-.1E-05	BACK UP1	-2.9809	.0000	135.4746	-1.489	-1.624	-1.455	-13.672	94.0267	94.0266	2.0789	140.6313
-.1E-05	BOTM REF	-2.9809	.0000	135.4746	-1.489	-1.724	-1.455	13.598	94.0267	94.0266	2.0789	140.6313
-.8E-06	BACK UP0	-.9695	2.0156	143.0007	-1.398	-1.665	-.639	15.334	99.2267	99.2266	2.1941	148.4228
-.8E-06	BACK UP1	-1.0000	1.9850	142.8894	-1.399	-1.666	-.650	15.337	99.1495	99.1495	2.1924	148.3074
-.8E-06	RCVR	-1.0000	1.9850	142.8894	-1.399	-1.666	-.650	15.337	99.1495	99.1495	2.1924	148.3074
-.7E-06	BACK UP0	.0002	2.9869	146.9174	-1.353	-1.648	-.278	8.081	101.9089	101.9089	2.2538	152.4609
-.7E-06	SURF REF	.0000	2.9867	146.9161	-1.353	-1.648	-.278	-8.082	101.9081	101.9080	2.2538	152.4596
-.9E-06	BACK UP0	-1.0288	1.9594	151.0478	-1.308	-1.632	-.698	-15.339	104.7393	104.7393	2.3168	156.7206
-.9E-06	BACK UP1	-1.0000	1.9881	150.9431	-1.309	-1.632	-.686	-15.335	104.6667	104.6667	2.3152	156.6119
-.9E-06	RCVR	-1.0000	1.9881	150.9431	-1.309	-1.632	-.686	-15.335	104.6667	104.6667	2.3152	156.6119
-.1E-05	BACK UP0	-3.1140	-.1237	158.9115	-1.226	-1.581	-1.484	-13.297	110.1667	110.1667	2.4371	164.8571
-.1E-05	BACK UP1	-2.9902	.0000	158.3933	-1.231	-1.586	-1.439	-13.590	109.8135	109.8135	2.4292	164.3244
-.1E-05	BOTM REF	-2.9902	.0000	158.3933	-1.231	-1.623	-1.439	13.562	109.8135	109.8135	2.4292	164.3244
-.9E-06	BACK UP0	-.9781	2.0137	165.9177	-1.156	-1.578	-.745	15.413	115.0135	115.0135	2.5444	172.1141
-.9E-06	BACK UP1	-1.0000	1.9918	165.8384	-1.157	-1.578	-.752	15.416	114.9585	114.9585	2.5431	172.0318
-.9E-06	RCVR	-1.0000	1.9918	165.8384	-1.157	-1.578	-.752	15.416	114.9585	114.9585	2.5431	172.0318
-.8E-06	BACK UP0	.0002	2.9927	169.8679	-1.119	-1.565	-.433	7.923	117.7210	117.7210	2.6046	176.1867
-.8E-06	SURF REF	.0000	2.9925	169.8664	-1.119	-1.565	-.433	-7.923	117.7200	117.7200	2.6045	176.1853
-.9E-06	BACK UP0	-1.0312	1.9619	174.0019	-1.081	-1.550	-.799	-15.462	120.5575	120.5575	2.6676	180.4506
-.9E-06	BACK UP1	-1.0000	1.9931	173.8890	-1.082	-1.551	-.788	-15.459	120.4791	120.4791	2.6659	180.3335
-.9E-06	RCVR	-1.0000	1.9931	173.8890	-1.082	-1.551	-.788	-15.459	120.4791	120.4791	2.6659	180.3335
-.1E-05	BACK UP0	-3.1262	-.1321	182.0027	-1.012	-1.504	-1.494	-12.885	126.0791	126.0791	2.7899	188.7222
-.1E-05	BACK UP1	-2.9941	.0000	181.4326	-1.016	-1.508	-1.452	-13.213	125.6912	125.6911	2.7812	188.1371
-.1E-05	BOTM REF	-2.9941	.0000	181.4326	-1.016	-1.525	-1.452	13.200	125.6912	125.6911	2.7812	188.1371
-.9E-06	BACK UP0	-.9976	1.9973	188.9519	-.955	-1.482	-.855	15.669	130.8912	130.8911	2.8962	195.9179
-.9E-06	BACK UP1	-1.0000	1.9949	188.9434	-.955	-1.482	-.856	15.669	130.8852	130.8852	2.8961	195.9090
-.9E-06	RCVR	-1.0000	1.9949	188.9434	-.955	-1.482	-.856	15.669	130.8852	130.8852	2.8961	195.9090

-.8E-06 BACK UP0	.0006	2.9959	192.9591	-.923	-1.464	-.576	7.660	133.6446	133.6445	2.9573	200.0509
-.8E-06 SURF REF	.0000	2.9953	192.9544	-.923	-1.464	-.577	-7.663	133.6416	133.6415	2.9573	200.0462
-.8E-06 BACK UP0	-1.0351	1.9605	197.0843	-.892	-1.446	-.902	-15.718	136.4822	136.4821	3.0203	204.3072
-.8E-06 BACK UP1	-1.0000	1.9956	196.9598	-.893	-1.446	-.891	-15.715	136.3953	136.3952	3.0184	204.1778
-.8E-06 RCVR	-1.0000	1.9956	196.9598	-.893	-1.446	-.891	-15.715	136.3953	136.3952	3.0184	204.1778
-.9E-06 BACK UP0	-3.0410	-.0449	204.7735	-.837	-1.402	-1.498	-12.696	141.7953	141.7952	3.1378	212.2548
-.9E-06 BACK UP1	-2.9961	.0000	204.5749	-.838	-1.403	-1.485	-12.811	141.6603	141.6602	3.1347	212.0512
-.9E-06 BOTM REF	-2.9961	.0000	204.5749	-.838	-1.412	-1.485	12.804	141.6603	141.6602	3.1347	212.0512
-.7E-06 MAX RANG	-1.5880	1.4085	210.0885	-.801	-1.378	-1.110	15.626	145.4603	145.4602	3.2189	217.7421

THIS RAY CALCULATION TOOK 13.562 SEC

END OF INPUT DATA

```
N01-1     SAMPLE CASE FOR HARPO DOCUMENTATION          REV. 8-19-86
VVORTX3     3.0 NPCURR       .0   CTANH       1.0 CBLOB2      7.0
NO MODL      .0 NO MODL      .0   NO MODL      .0 NO MODL      .0
NO MODL      .0 NO MODL      .0   SLLOSS       1.0 NPABSR      .0
GLORENZ     3.0 NPBOTM       .0   RHORIZ       .0 NO MODL      .0
  VORTEX AT LONGITUDE 150 KM E, UMAX= 1.02 M/S, R= 50 KM
  NO CURRENT PERTURBATION
  EL NINO BACKGROUND SOUND-SPEED PROFILE
  2% INCREASE IN C-SQUARED AT 150 KM LON., 1 KM DEPTH, 50 KM WIDE
  RIDGE .5 KM HIGH, 2 KM WIDE AT 10 KM N LATITUDE; BASE= -3 KM
N013    -9808     0     0   -10000 25132741   8000000    200000          0   050T
   -10000   194734   -18   -20 -2022   1311291   1311291   288     0     0   0 1R
   -10000   520417   -16   -27  2010   3504472   3504471   769     0     0   0 2R
   -10000   723109   -15   -42 -1990   4868914   4868914  1069     0     0   0 3R
   -10000  1127595     6  -147  1418   7586966   7586966  1667     0     0   0 4R
   -10000  1406506    51  -324 -1281   9454219   9454219  2079     0     0   0 5R
   -10000  1941240   197  -503  3128  13036463  13036463  2870     0     0   0 6R
    -8511  2101960   240  -509 -2971  14116463  14116463  3108     0     0   0 7F
N013    -9808     0     0   -10000 25132741   8000000    400000          0   050T
   -10000   165446   -17   -19 -4012   1114351   1114351   245     0     0   0 1R
   -10000   468372   -15   -22  4001   3154502   3154502   693     0     0   0 2R
   -10000   637278   -14   -31 -3985   4291986   4291986   943     0     0   0 3R
   -10000   963209    -6   -74  3618   6484653   6484653  1425     0     0   0 4R
   -10000  1149018     7  -148 -3476   7731899   7731898  1700     0     0   0 5R
   -10000  1652653    91  -346  2658  11104455  11104455  2446     0     0   0 6R
   -10000  1864722   142  -439 -2819  12525535  12525535  2759     0     0   0 7R
   -15734  2105919   200  -455  1823  14145535  14145535  3116     0     0   0 8F
N013    -9808     0     0   -10000 25132741   8000000    600000          0   050T
   -10000   145209   -16   -17 -6008    978585    978585   215     0     0   0 1R
   -10000   458199   -12   -18  5996   3086200   3086200   679     0     0   0 2R
   -10000   605599   -13   -25 -5983   4079462   4079462   898     0     0   0 3R
   -10000   934379    -6   -55  5593   6291447   6291447  1385     0     0   0 4R
   -10000  1090494     1  -108 -5472   7340500   7340500  1616     0     0   0 5R
   -10000  1530604    52  -242  4301  10289292  10289292  2268     0     0   0 6R
   -10000  1709365    87  -352 -4402  11486110  11486110  2533     0     0   0 7R
   -10000  2087795   168  -388  5537  14026691  14026691  3093     0     0   0 8R
    -8578  2102600   171  -389  5404  14126691  14126691  3115     0     0   0 9F
N013    -9808     0     0   -10000 25132741   8000000    800000          0   050T
   -10000   134059   -15   -16 -8006    904047    904047   199     0     0   0 1R
   -10000   482230    -9   -14  7989   3246733   3246733   716     0     0   0 2R
   -10000   617971   -10   -22 -7975   4162042   4162042   918     0     0   0 3R
   -10000   980715    -4   -48  7435   6600587   6600587  1456     0     0   0 4R
   -10000  1123615     2  -102 -7311   7561098   7561098  1669     0     0   0 5R
   -10000  1557112    45  -192  6434  10465570  10465570  2312     0     0   0 6R
   -10000  1710190    69  -283 -6530  11491347  11491347  2539     0     0   0 7R
   -10281  2100554   134  -300  7721  14111347  14111347  3118     0     0   0 8F
N013    -9808     0     0   -10000 25132741   8000000   1000000          0   050T
   -10000   193456   -12   -14-10005   1298480   1298480   288     0     0   0 1R
   -28979   348791    -9   232  3400   2345757   2345757   519     0     0   0 2G
   -10000   493788   -80   159 10224   3324473   3324474   735     0     0   0 2R
   -10000   670788  -126   105-10188   4513905   4513905   999     0     0   0 3R
   -28954   811497  -145  -210  3021   5464007   5464008  1209     0     0   0 4G
   -10000   964893  -113  -197  9505   6497632   6497633  1437     0     0   0 4R
   -10000  1176220   -76  -234 -9212   7911470   7911470  1751     0     0   0 5R
   -29882  1322094   -48  -265  4554   8894430   8894430  1969     0     0   0 6G
```

```
   -10000  1471174   -19  -295  8695   9897927   9897927  2191      0     0    0 6R
   -10000  1700700    31  -383 -8865  11429294  11429295  2532      0     0    0 7R
   -29961  1861151    68  -394  3353  12508746  12508747  2771      0     0    0 8G
   -10000  2017061   100  -384  9692  13559231  13559232  3004      0     0    0 8R
    -1257  2100424   117  -384  1425  14119231  14119232  3128      0     0    0 9F
N013    -9808     0     0   -10000 25132741   8000000   1200000          0    050T
   -10000   116447   -13   -14-12004    788973    788973   175      0     0    0 1R
   -29508   227433   -10   193  7184   1543215   1543216   341      0     0    0 2G
   -10000   335994   -75   127 12169   2281500   2281500   504      0     0    0 2R
   -10000   450772  -111    88-12165   3059732   3059733   676      0     0    0 3R
   -25790   532686  -127  3158 11314   3619144   3619144   800      0     0    0 4G
   -10000   600692  -486  2797 14060   4087894   4087895   903      0     0    0 4R
   -10000   699780  -885  2392-14041   4763564   4763565  1053      0     0    0 5R
   -27472   776090 -1122  -362  8925   5288367   5288368  1169      0     0    0 6G
   -10000   864716 -1086  -335 12427   5893266   5893268  1302      0     0    0 6R
   -10000   974114 -1050  -323-12356   6635033   6635035  1467      0     0    0 7R
   -29598  1074338 -1020  -497  8489   7317761   7317763  1617      0     0    0 8G
   -10000  1177288  -977  -482 11826   8017840   8017842  1772      0     0    0 8R
   -10000  1290483  -934  -502-11733   8782483   8782485  1942      0     0    0 9R
   -29865  1392201  -896  -541  9044   9474203   9474204  2095      0     0    010G
   -10000  1494738  -858  -546 11543  10171061  10171063  2250      0     0    010R
   -10000  1609212  -816  -582-11578  10943305  10943306  2422      0     0    011R
   -29935  1713287  -778  -596  8749  11650537  11650538  2578      0     0    012G
   -10000  1816361  -743  -591 11929  12351703  12351704  2733      0     0    012R
   -10000  1928983  -706  -597-12030  13113644  13113645  2902      0     0    013R
   -29962  2035948  -673  -588  7919  13840989  13840989  3063      0     0    014G
   -18778  2100987  -655  -572 11474  14280989  14280989  3161      0     0    014F
N013    -9808     0     0   -10000 25132741   8000000   1400000          0    050T
   -10000    96474   -13   -14-14003    657818    657818   146      0     0    0 1R
   -29600   185493   -10   206 10186   1267978   1267978   280      0     0    0 2G
   -10000   273220   -79   137 14155   1869704   1869704   413      0     0    0 2R
   -10000   374756  -120    94-14153   2560890   2560890   567      0     0    0 3R
   -27790   452665  -138  2653 12635   3096309   3096309   685      0     0    0 4G
   -10000   521548  -491  2300 15588   3573382   3573382   790      0     0    0 4R
        0   562926  -661  2128 -6517   3858025   3858026   853      0     0    0 5S
   -10000   604317  -808  1979-15580   4142711   4142711   916      0     0    0 5R
   -25168   661771  -981  -413 11913   4541478   4541479  1004      0     0    0 6G
   -10000   725091  -947  -383 14335   4978134   4978134  1100      0     0    0 6R
        0   773854  -924  -365 -2616   5310488   5310489  1174      0     0    0 7S
   -10000   822672  -904  -351-14302   5643124   5643125  1248      0     0    0 7R
   -29130   905056  -872 -1030 10348   6209075   6209075  1373      0     0    0 8G
   -10000   991598  -784  -956 13622   6801934   6801935  1504      0     0    0 8R
   -10000  1091722  -697  -904-13547   7482368   7482369  1655      0     0    0 9R
   -29775  1180022  -629  -958 10589   8087007   8087008  1788      0     0    010G
   -10000  1269610  -561  -922 13183   8699639   8699640  1924      0     0    010R
   -10000  1373491  -490  -917-13126   9402993   9402995  2081      0     0    011R
   -29904  1462585  -433  -926 10886  10012246  10012248  2216      0     0    012G
   -10000  1551897  -379  -910 13086  10622890  10622893  2351      0     0    012R
   -10000  1655342  -319  -919-13134  11323316  11323318  2507      0     0    013R
   -29948  1745912  -269  -914 10549  11942496  11942499  2644      0     0    014G
   -10000  1835857  -224  -893 13413  12558001  12558004  2780      0     0    014R
   -10000  1935470  -175  -882-13489  13234962  13234965  2930      0     0    015R
   -29968  2027494  -134  -862  9968  13864530  13864534  3069      0     0    016G
   -14448  2100720  -104  -837 13577  14364530  14364534  3180      0     0    016F
N013    -9808     0     0   -10000 25132741   8000000   1600000          0    050T
```

0	39152	-13	-14	-7420	269703	269703	60	0	0	0 1S
-10000	78975	-13	-14	-16003	544103	544103	120	0	0	0 1R
-29653	153961	-10	213	12811	1063042	1063042	235	0	0	0 2G
-10000	228151	-83	140	16139	1576821	1576820	348	0	0	0 2R
0	267475	-105	117	-7714	1847998	1847998	408	0	0	0 3S
-10000	306799	-122	99	-16138	2119176	2119176	468	0	0	0 3R
-28791	377222	-143	1485	13972	2607306	2607306	576	0	0	0 4G
-10000	443760	-367	1260	16889	3070471	3070471	678	0	0	0 4R
0	480587	-465	1161	-9230	3325562	3325562	735	0	0	0 5S
-10000	517418	-549	1075	-16886	3580665	3580664	791	0	0	0 5R
-25573	571345	-652	5462	16468	3957184	3957184	874	0	0	0 6G
-10000	619964	-1083	5030	18383	4300867	4300866	950	0	0	0 6R
0	652683	-1337	4774	-11851	4530267	4530267	1001	0	0	0 7S
-10000	685421	-1566	4543	-18376	4759688	4759688	1052	0	0	0 7R
-25168	732669	-1861	1370	15683	5093370	5093369	1125	0	0	0 8G
-10000	782658	-1950	1277	17518	5443729	5443728	1202	0	0	0 8R
0	817521	-2005	1217	-10579	5685999	5685998	1256	0	0	0 9S
-10000	852403	-2056	1161	-17499	5928338	5928337	1310	0	0	0 9R
-28039	912552	-2133	-1725	13679	6348794	6348792	1403	0	0	010G
-10000	978766	-2017	-1620	16051	6808190	6808188	1504	0	0	010R
0	1017652	-1956	-1571	-8326	7076045	7076043	1563	0	0	011S
-10000	1056589	-1899	-1527	-16008	7344119	7344117	1623	0	0	011R
-29512	1128133	-1803	-1841	13397	7839875	7839872	1733	0	0	012G
-10000	1201295	-1691	-1752	15561	8345838	8345835	1844	0	0	012R
0	1241192	-1635	-1718	-8086	8619674	8619671	1905	0	0	013S
-10000	1281144	-1581	-1688	-15517	8893751	8893748	1966	0	0	013R
-29809	1354746	-1489	-1724	13598	9402668	9402664	2079	0	0	014G
-10000	1428894	-1399	-1666	15337	9914954	9914949	2192	0	0	014R
0	1469161	-1353	-1648	-8082	10190808	10190803	2254	0	0	015S
-10000	1509431	-1309	-1632	-15335	10466671	10466666	2315	0	0	015R
-29902	1583933	-1231	-1623	13562	10981354	10981349	2429	0	0	016G
-10000	1658384	-1157	-1578	15416	11495851	11495845	2543	0	0	016R
0	1698664	-1119	-1565	-7923	11772004	11771998	2605	0	0	017S
-10000	1738890	-1082	-1551	-15459	12047914	12047908	2666	0	0	017R
-29941	1814326	-1016	-1525	13200	12569117	12569111	2781	0	0	018G
-10000	1889434	-955	-1482	15669	13088524	13088517	2896	0	0	018R
0	1929544	-923	-1464	-7663	13364155	13364148	2957	0	0	019S
-10000	1969598	-893	-1446	-15715	13639529	13639522	3018	0	0	019R
-29961	2045749	-838	-1412	12804	14166026	14166018	3135	0	0	020G
-15880	2100885	-801	-1378	15626	14546026	14546017	3219	0	0	020F

APPENDIX B: BLANK INPUT-PARAMETER FORMS

This appendix provides blank Input Parameter Forms for all of the ocean models (including bottom and receiver-surface models) that we have developed for HARPO. The forms describe each model mathematically and list the variable input parameters you have to specify. We recommend reproducing these forms and filling them out when setting up ocean models for HARPO. The filled-out forms should then be saved as a record of the models you have defined.

The FORTRAN source codes for the corresponding model subroutines are listed in Appendix D under the model name. No forms are given for the do-nothing versions NPSPEED, NPBOTM, NPCURR, etc. The forms are arranged as follows:

3-D RAYPATH CALCULATION..... 213

RAYPATH PLOT PROJECTION..... 214

OCEAN MODELS..... 215

BACKGROUND CURRENT MODELS

WLINEAR -- A linear profile of u; constant v and w..... 216
VVORTX3 -- A vertical vortex with a solid-rotating core..... 217
WGAUSS2 -- A jet of u that decays in three dimensions..... 218

BACKGROUND SOUND-SPEED MODELS

CTANH -- C profile with linear segments smoothly joined..... 219
CSTANH -- C^2 profile with linear segments smoothly joined.... 220
CSSPOKE -- C_2 as a function of angle from horizontal..... 221
CSSPOK2 -- C^2 as a function of angle from horizontal..... 222
CSMUNK1 -- Canonical sound channel; linear parameters in longitude..... 223
CSMUNK2 -- Canonical sound channel; linear C with longitude.. 224
CTABLE -- A tabular sound-speed profile in cubic segments.... 225

SOUND-SPEED PERTURBATION MODELS

CBLOB2 -- A blob of C^2 increase that decays in three dimensions..... 226
CBLOB3 -- Up to 3 blobs of C increase that decay in three dimensions..... 227
CBLOB4 -- Up to 3 blobs of C^2 increase that decay in three dimensions..... 228

ABSORPTION MODEL
SLLOSS -- Skretting-Leroy model of acoustic absorption..... 229

BOTTOM MODELS
GHORIZ -- A horizontal surface at a specified height above sea level..... 230
GLORENZ -- A Lorentzian ridge along a latitude line..... 231
GTANH -- Linear segments smoothly joined..... 232

OCEAN-SURFACE MODELS
SHORIZ -- A horizontal surface at a fixed height above sea level.. 233

RECEIVER-SURFACE MODELS
RHORIZ -- A surface at a fixed height above mean sea level..... 234
RBOTM -- A surface at a fixed height above the bottom..... 235
RVERT -- A vertical receiver surface at a fixed range from an origin..... 236

FORM TO SPECIFY INPUT DATA FOR A
THREE-DIMENSIONAL RAYPATH CALCULATION

Name_____
Ocean ID (3 characters)_____ Date_____

Title (77 characters)_____

Transmitter: Height _____km, nm, ft (W3)
_____above bottom
above sea level
Latitude _____rad, deg, km (W4)
Longitude _____rad, deg, km (W5)
Frequency, initial _____rad/s, Hz, s (W7)
final _____ (W8)
step _____ (W9)
Azimuth angle, initial _____rad, deg clockwise of north (W11)
final _____ (W12)
step _____ (W13)
Elevation angle, initial _____rad, deg (W15)
final _____ (W16)
step _____ (W17)

Receiver: Height _____km, nm, ft (W20)
_____above sea level (rcvr model RHORIZ)
_____above bottom (rcvr model RBOTM)
Distance from origin _____rad, deg, km (W278) (rcvr model RVERT)
Latitude of origin _____rad, deg, km (W279) (rcvr model RVERT)
Longitude of origin _____rad, deg, km (W280) (rcvr model RVERT)

Stop elevation-angle stepping
when ray goes out of bounds _____(W21 = 1.)
Stop ray if it hits the bottom _____(W19 = 1.)

Maximum height _____km (W26)
Minimum height _____km (W27)
Maximum range _____km (W28)
Maximum number of hops _____(W22)
Maximum number of steps per hop _____(W23)
Maximum allowable error per step _____(W42)

Additional calculations: = 1. to integrate
= 2. to integrate and print
Phase path (W57)
Absorption _____(W58)
Doppler shift _____(W59)
Path length _____(W60)

Printout: Every_____steps of the ray trace (W71)

Computer readable output (raysets): _____(W72 = 1.)
Diagnostic printing: _____(W73 = 1.)
Suppress all printout _____(W74 = 1.)

FORM TO SPECIFY INPUT PARAMETERS FOR PLOTTING A
PROJECTION OF THE RAYPATH

Model ID:_____

Plot directly during raypath calculations_____, or

plot from precomputed raypaths_____

in disk file_____

Normal or apogee plots: Normal_____(W80=0.0)

Plot apogees only_____(W80=1.0)

Projection:

Vertical plane, polar plot, rectangular expansion_____(W81=1.0)

Horizontal plane, lateral expansion _____(W81=2.0)

Vertical plane, polar plot, radial expansion _____(W81=3.0)

Vertical plane, rectangular plot _____(W81=4.0)

Superimpose these raypath plots on the graph of the previous runset:

Yes_____(W81 negative.)

No _____(W81 positive.)

Vertical or lateral expansion factor _____(W82)

Coordinates of the left edge of the graph:

Latitude = _____(rad, deg, km) north (W83)

Longitude = _____(rad, deg, km) east (W84)

Coordinates of the right edge of the graph:

Latitude = _____(rad, deg, km) north (W85)

Longitude = _____(rad, deg, km) east (W86)

Distance between horizontal tick marks = _____rad, deg, km (W87)

Height above sea level of bottom of graph = _____km (W88)

Height above sea level of top of graph = _____km (W89)

Distance between vertical tick marks = _____km (W96)

FORM TO SPECIFY AN OCEAN MODEL
(including bottom and upper surface model)

Name_____ Date_____ Model ID (3 characters)_____

Dispersion Relation
__ ANCNL __ AWCWL __AWCNL __ANCWL __Other _____

Coordinates of the north pole pole of the computational coordinate system:
North geographic latitude = _____rad, km, deg (W24)
East geographic longitude = _____rad, km, deg (W25)

Data Set ID	Model Subroutine Name		(Model Check Number)		
Current Velocity	__	WLINEAR	(1.)	__ WGAUSS2	(8.)
W102 (W100)	__	VVORTX3	(9.)	__ Other _____	( )
Current Perturbation	__	NPCURR	(0.)	__ Other _____	( )
W127 (W125)					
Sound Speed	__	CSTANH	(2.)	__ CSMUNK1	(5.)
W152 (W150)	__	CSSPOKE	(3.)	__ CSMUNK2	(6.)
	__	CSSPOK2	(4.)	__ CTABLE	(8.)
	__	CTANH	(7.)	__ Other _____	( )
Sound-Speed Perturbation					
W177 (W175)	__	NPSPEED	(0.)	__ CBLOB2	(2.)
	__	CBLOB3	(3.)	__ Other _____	( )
Receiver Surface*		RHORIZ	(1.)	RTERR	(2.)
(W275)		RVERT	(3.)	Other _____	( )
Ocean Bottom	__	GHORIZ	(1.)	__ GTANH	(2.)
W302 (W300)	__	GLORENZ	(3.)	__ Other _____	( )
Ocean-Bottom Perturbation					
W327 (W325)	__	NPBOTM	(0.)	__ Other _____	( )
Ocean Surface					
W352 (W350)	__	SHORIZ	(1.)	__ Other _____	( )
Ocean-Surface Perturbation					
W377 (W375)	__	NPSURF	(0.)	__ Other _____	( )
Absorption (loss)					
W502 (W500)	__	SLLOSS	(1.)	__ Other _____	( )
Absorption Perturbation					
W527 (W525)	__	NPABSR	(0.)	__ Other _____	( )
Graph Annotation*					
W75	__	SMPANN		__ FULANN	

* The receiver-surface and graph-annotation models are not considered part of the ocean-model ID.

FORM TO SPECIFY INPUT DATA FOR
CURRENT-VELOCITY MODEL WLINEAR

This subroutine specifies constant radial (upward), eastward and southward currents, allowing a linear height gradient of the eastward component.

$$U_\theta = U_{\theta o}$$

$$U_\phi = U_{\phi o} + \frac{du_\phi}{dz} z$$

$$u_r = U_{ro}$$

$z = r - r_e$, where r_e is the Earth radius, and r is the radial coordinate of ray point.

Specify--

the model check for WLINEAR = ___1.0___(W100)

the input data-format code = _____(W101)

an input data-set identification number = _____(W102)

an 80-character description of the current-velocity profile:

---

the constant upward current, U_{ro} = _____km/s, m/s (W103)

the constant southward current, $U_{\theta o}$ = _____km/s, m/s (W104)

the ground value of the eastward current, $U_{\phi o}$ = _____km/s, m/s (W105)

the height gradient of u_ϕ, du_ϕ/dz = _____km/s/km, m/s/km (W106)

(This subroutine can be used with its input parameters zero when no current field is desired.)

OTHER MODELS REQUIRED: Any current-perturbation model. Use NPCURR if no perturbation is desired.

FORM TO SPECIFY INPUT DATA FOR
CURRENT-VELOCITY MODEL VVORTX3

This subroutine models a vortex with a viscous core and a Gaussian intensity profile in the vertical. The axis of the vortex is vertical and may be positioned above any geographic latitude and longitude. The vortex rotates anticlockwise looking down. The core (inside r_o) is essentially a solid-rotating fluid, while outside r_o, $|u|$ falls off as the inverse radius.

$$u_\theta = -\frac{1.397\, R_e U_o r_o}{r^2}\left(1 - e^{-1.26\, r^2/r_o^2}\right)(\phi - \phi_o)e^{-\left(\frac{h - h_{max}}{w_H}\right)^2}$$

$$u_\phi = \frac{1.397\, R_e U_o r_o}{r^2}\left(1 - e^{-1.26\, r^2/r_o^2}\right)(\theta - \theta_o)e^{-\left(\frac{h - h_{max}}{w_H}\right)^2},$$

where $\theta_o = \pi/2 - \lambda_o$ and r is the radial distance from the vortex center. The numerical constants normalize the function so that $|U| = U_o$ at $r = r_o$. R_e is the radius of the Earth, θ is the colatitude, ϕ is the longitude, and h is the height above sea level.

Specify--

the model check for VVORTX3 = ___9.0___(W100)

the input data-format code = _____(W101)

an input data-set identification number = _____(W102)

an 80-character description of the model, including description of parameter values:

---

the maximum tangential current, U_o = _____km/s, m/s (W103)

the radius of the vortex core (to $u = U_o$), r_o = _____km (W104)

the latitude of the vortex center, λ_o = _____rad, deg, km N (W105)

the longitude of the vortex center, ϕ_o = _____rad, deg, km E (W106)

the Gaussian width in height of the vortex, W_H = _____km, m (W107)

the height of the vortex, h_{max} = _____km, m (W108)

OTHER MODELS REQUIRED: Any current-perturbation model. Use NPCURR if no perturbation is desired.

FORM TO SPECIFY INPUT DATA FOR
CURRENT-VELOCITY MODEL WGAUSS2

This subroutine specifies a zonal (eastward) current field whose intensity decays in a Gaussian manner in all three space dimensions.

$$u_\phi = U_{\phi o} \exp\left\{ - \left(\frac{z-z_o}{w_z}\right)^2 - \left(\frac{\theta-\theta_o}{W_\theta}\right)^2 - \left(\frac{\phi-\phi_o}{W_\phi}\right)^2 \right\}$$

$z = r - r_e$, where r_e is the Earth radius, $\theta_o = \pi/2 - \lambda_o$, and r is the radial coordinate of the ray point. θ is the colatitude. ϕ is the longitude.

Notice that this current field does not satisfy continuity if $W_\phi \neq 0$.

Specify--

the model check for WGAUSS2 = 8.0 (W100)

the input data-format code = _____(W101)

an input data-set identification number = _____(W102)

an 80-character description of the model, including description of parameter values:

---

the maximum value of u_ϕ, $U_{\phi o}$ = _____km/s, m/s (W103)

the height where u_ϕ, maximizes, z_o = _____km (W107)

the Gaussian width in height of u_ϕ, W_z = _____km (W104)*

the latitude where u_ϕ maximizes, λ_o = _____rad, deg, km N (W108)

the meridional width of u_ϕ, W_θ = _____rad, deg, km (W105)*

the longitude where u_ϕ, maximizes, ϕ_o = _____rad, deg, km E (W109)

the zonal width of u_ϕ, W_ϕ = _____rad, deg, km (W106)*

OTHER MODELS REQUIRED: Any current-perturbation model. Use NPCURR if no perturbation is desired.

*Setting W_z, W_θ or W_ϕ = 0 results in no space variation in that direction.

FORM TO SPECIFY INPUT DATA FOR BACKGROUND
SOUND-SPEED MODEL CTANH

This model represents the sound-speed profile by a sequence of linear segments that are smoothly joined by hyperbolic functions:

$$C = C_o + \frac{b_1}{2}(z-z_o) + \sum_{i=1}^{n} \delta_i \left(\frac{b_{i+1} - b_i}{2}\right) \ln\left\{\frac{\cosh\left(\frac{z-z_i}{\delta_i}\right)}{\cosh\left(\frac{z_i-z_o}{\delta_i}\right)}\right\} + \frac{b_{n+1}}{2}(z-z_o)$$

$$\frac{dC}{dz} = b_1 + \sum_{i=1}^{n} \left(\frac{b_{i+1} - b_i}{2}\right) \left\{\tanh\left(\frac{z-z_i}{\delta_i}\right) + 1\right\}$$

$$b_i = (C_i - C_{i-1})/(z_i - z_{i-1}) \quad .$$

$z = r - r_e$, where r_e is the Earth radius, and r is the radial coordinate of the ray point. Thus, δ_i is the half-thickness of a region centered at approximately z_i km, in which dC/dz changes from b_i to b_{i+1}. Start by drawing a profile with linear segments, and get C_i and z_i from the corners. Then select δ_i to round the corners. The final profile will not go through (C_i, z_i).

Specify--

the model check for CTANH = 7.0 (W150)

the input data-format code = _____ (W151)

an input data-set identification number = _____ (W152)

an 80-character description of the model with parameters:

_____

and the profile values:

the number of points in the profile -2 = n = _____

the profile: i	z_i (km,m)	C_i (km/s, m/s)	δ_i (km,m)

OTHER MODELS REQUIRED: Any sound-speed perturbation model. Use NPSPEED if no perturbation is desired. FUNCTION ALCOSH.

FORM TO SPECIFY INPUT DATA FOR
SOUND-SPEED MODEL CSTANH

This model represents the sound-speed (squared) profile by a sequence of linear segments that are smoothly joined by hyperbolic functions:

$$C^2 = C_o^2 + \frac{b_1}{2}(z-z_o) + \sum_{i=1}^{n} \delta_i \left(\frac{b_{i+1} - b_i}{2}\right) \ln \frac{\cosh\left(\frac{z-z_i}{\delta_i}\right)}{\cosh\left(\frac{z_i-z_o}{\delta_i}\right)} + \frac{b_{n+1}}{2}(z-z_o)$$

$$\frac{dC^2}{dz} = b_1 + \sum_{i=1}^{n} \left(\frac{b_{i+1} - b_i}{2}\right) \left\{\tanh\left(\frac{z-z_i}{\delta_i}\right) + 1\right\}$$

$$b_i = (C_i^2 - C_{i-1}^2)/(z_i - z_{i-1}) \quad .$$

$z = r - r_e$, where r_e is the Earth radius, and r is the radial coordinate of the ray point. Thus, δ_i is the half-thickness of a region centered at approximately z_i km, in which dC^2/dz changes from b_i to b_{i+1}. Start by drawing a profile with linear segments, and get C_i^2 and z_i from the corners. Then select δ_i to round the corners. The final profile will not go through (C_i^2, z_i).

Specify--

the model check for CSTANH = 2.0 (W150)

the input data-format code = _____ (W151)

an input data-set identification number = _____ (W152)

an 80-character description of the model with parameters:

_____

and the profile values:

the number of points in the profile -2 = n = _____

the profile: i	z_i (km,m)	C_i (km/s, m/s)	δ_i (km,m)

OTHER MODELS REQUIRED: Any sound-speed-perturbation model. Use NPSPEED if no perturbation is desired. FUNCTION ALCOSH.

FORM TO SPECIFY INPUT DATA FOR
SOUND SPEED MODEL CSSPOKE

This model represents the sound speed (squared) as a function of the angle α from a horizontal line at a specified height and latitude. The dependence of C^2 on α is as a sequence of linear segments joined by hyperbolic functions.

$$C^2 = C_o^2 + \frac{b_1}{2}(\alpha-\alpha_o) + \sum_{i=1}^{n} \delta_i \left(\frac{b_{i+1} - b_i}{2}\right) \ell n \left\{ \frac{\cosh\left(\frac{\alpha-\alpha_i}{\delta_i}\right)}{\cosh\left(\frac{\alpha_i-\alpha_o}{\delta_i}\right)} \right\} + \frac{b_{n+1}}{2}(\alpha-\alpha_o)$$

$$\frac{dC^2}{dz} = b_1 + \sum_{i=1}^{n} \left(\frac{b_{i+1} - b_i}{2}\right) \left\{\tanh\left(\frac{\alpha-\alpha_i}{\delta_i}\right) + 1\right\} ,$$

where $b_i = (C_i^2 - C_{i-1}^2)/(\alpha_i - \alpha_{i-1})$, $\alpha = \sin^{-1}((r\cos(\theta-\theta_o) - r_o)/D)$, $D = (r_o^2 + r^2 - 2\,r\,r_o\cos(\theta-\theta_o))^{1/2}$, $r_o = r_e + h_o$, $\theta_o = \pi/2 - \lambda_o$, r_e is the Earth radius, r is the radial coordinate of the ray point and θ is the colatitude of the ray point. Thus, δ_i is the half-thickness of a region centered at approximately α_i, in which $dC^2/d\alpha$ changes from b_i to b_{i+1}.

Specify--

the model check for CSSPOKE = 3.0 (W150)

the input data format code = _____ (W151)

an input data set identification number = _____ (W152)

an 80-character description of the model with parameters:

_____

the reference sound speed, C_{ref} = _____ km/s (W153)

the height of the horizontal line, h_o = _____ km, m (W154)

the latitude of the horizontal line, λ_o = _____ rad, deg, km (W155)

and the profile values:

the number of points in the profile, n = _____

the profile:	i	z_i (km)	C_i (km/sec)	δ_i (km)

OTHER MODELS REQUIRED: Any sound-speed perturbation model. Use NPSPEED if no perturbation is desired. FUNCTION ALCOSH.

FORM TO SPECIFY INPUT DATA FOR
SOUND SPEED MODEL CSSPOK2

This model represents the sound speed (squared) as a function of the angle α from a horizontal line at a specified height and latitude. The dependence of C^2 on α is as a sequence of linear segments joined by hyperbolic functions.

$$C^2 = C_o^2 + \frac{b_1}{2}(\alpha-\alpha_o) + \sum_{i=1}^{n} \delta_i \left(\frac{b_{i+1} - b_i}{2}\right) \ln\left\{\frac{\cosh\left(\frac{\alpha-\alpha_i}{\delta_i}\right)}{\cosh\left(\frac{\alpha_i-\alpha_o}{\delta_i}\right)}\right\} + \frac{b_{n+1}}{2}(\alpha-\alpha_o)$$

$$\frac{dC^2}{dz} = b_1 + \sum_{i=1}^{n} \left(\frac{b_{i+1} - b_i}{2}\right) \left\{\tanh\left(\frac{\alpha-\alpha_i}{\delta_i}\right) + 1\right\} ,$$

where $b_i = (C_i^2 - C_{i-1}^2)/(\alpha_i - \alpha_{i-1})$, $\alpha = \tan^{-1}((r - r_o)/r_e(\theta - \theta_o)))$, $r_o = r_e + h_o$, $\theta_o = \pi/2 - \lambda_o$, r_e is the Earth radius, r is the radial coordinate of the ray point and θ is the colatitude of the ray point. Thus, δ_i is the half-thickness of a region centered at approximately α_i, in which $dC^2/d\alpha$ changes from b_i to b_{i+1}.

Specify--

the model check for CSSPOK2 = ___4.0___(W150)

the input data format code = _____(W151)

an input data set identification number = _____(W152)

an 80-character description of the model with parameters:

_____

the reference sound speed, C_{ref} = _____km/s (W153)

the height of the horizontal line, h_o = _____km, m (W154)

the latitude of the horizontal line, λ_o = _____rad, deg, km (W155)

and the profile values:

the number of points in the profile, n = _____

the profile: i	z_i (km)	C_i (km/sec)	δ_i (km)

OTHER MODELS REQUIRED: Any sound-speed perturbation model. Use NPSPEED if no perturbation is desired. FUNCTION ALCOSH.

FORM TO SPECIFY INPUT DATA FOR
OCEAN SOUND-SPEED MODEL CSMUNK1*

This subroutine specifies the "canonical" model for a sound channel derived by Munk. The four parameters of the model can vary linearly with longitude, ϕ. The user specifies values for the four parameters at two longitudes, ϕ_1, and ϕ_2, and the program interpolates linearly to get the values at other longitudes.

$$C = C_A[1 + \varepsilon(\eta + e^{-\eta} - 1)] ,$$

where

$$\eta = 2/H\ (z - z_A) ,$$

$$C_A = C_{A1} + (C_{A2} - C_{A1})(\phi - \phi_1)/(\phi_2 - \phi_1) ,$$

$$z_A = z_{A1} + (z_{A2} - z_{A1})(\phi - \phi_1)/(\phi_2 - \phi_1) ,$$

$$H = H_1 + (H_2 - H_1)(\phi - \phi_1)/(\phi_2 - \phi_1) ,$$

$$\varepsilon \pm \varepsilon_1 + (\varepsilon_2 - \varepsilon_1)(\phi - \phi_1)/(\phi_2 - \phi_1) ,$$

and $z = r - r_e$, where r_e is the earth radius.

Specify--

the model check for subroutine CSMUNK1 = 5. (W150)
the input data format code = _____ (W151)
an input data set identification number = _____ (W152)
an 80-character description for the sound-speed model, including description of parameter values:

---

the reference sound speed, C_0 = _____ km/s, m/s (W153)
longitude of first profile, ϕ_1 = _____ rad, deg, km (W154)
the sound speed on the axis, C_{A1} = _____ km/s, m/s (W155)
the height of the axis, z_{A1} = _____ km, m (W156)
the scale depth, H_1 = _____ km, m (W157)
the fractional increase of C with depth, η = _____ (W158)
longitude of second profile, ϕ_2 = _____ rad, deg, km (W159)
the sound speed on the axis, C_{A2} = _____ km/s, m/s (W160)
the height of the axis, z_{A2} = _____ km, m (W161)
the scale depth, H_2 = _____ km, m (W162)
the fractional increase of C with depth, η = _____ (W163)

Other models required: Any sound-speed perturbation model. Use NPSPEED if no perturbation is desired.

---

*Munk, W. H., (1974), Sound channel in an exponentially stratified ocean, with application to SOFAR, J. Acoust. Soc. Amer. 55, 220-226.

FORM TO SPECIFY INPUT DATA FOR
OCEAN SOUND-SPEED MODEL CSMUNK2*

This subroutine specifies the "canonical" model for a sound channel derived by Munk. The sound speed varies linearly with longitude, ϕ. The user specifies values for the four parameters at two longitudes, ϕ_1 and ϕ_2.

$$C = C_1 + (C_2 - C_1)(\phi - \phi_1)/(\phi_2 - \phi_1) ,$$

where

$$C_1 = C_{A1}[1 + \varepsilon_1(\eta_1 + e^{-\eta_1} - 1)] ,$$

$$C_2 = C_{A2}[1 + \varepsilon_2(\eta_2 + e^{-\eta_2} - 1)] ,$$

$$\eta_1 = (2/H_1)(z - z_{A1}) ,$$

$$\eta_2 = (2/H_2)(z - z_{A2}) ,$$

and $z = r - r_e$, where r_e is the earth radius.

Specify--

the model check for subroutine CSMUNK2 = ____6.___(W150)
the input data format code = _____(W151)
an input data set identification number = _____(W152)
an 80-character description for the sound-speed model, including description of parameter values:

---

the reference sound speed, C_0 = _____km/s, m/s (W153)
longitude of first profile, ϕ_1 = _____rad, deg, km (W154)
the sound speed on the axis, C_{A1} = _____km/s, m/s (W155)
the height of the axis, z_{A1} = _____km, m (W156)
the scale depth, H_1 = _____km, m (W157)
the fractional increase of C with depth, η = _____(W158)
longitude of second profile, ϕ_2 = _____rad, deg, km (W159)
the sound speed on the axis, C_{A2} = _____km/s, m/s (W160)
the height of the axis, z_{A2} = _____km, m (W161)
the scale depth, H_2 = _____km, m (W162)
the fractional increase of C with depth, η = _____(W163)

Other models required: Any sound-speed perturbation model. Use NPSPEED if no perturbation is desired.

---

*Munk, W. H., (1974), Sound channel in an exponentially stratified ocean, with application to SOFAR, J. Acoust. Soc. Amer. 55, 220-226.

FORM TO SPECIFY INPUT DATA FOR
OCEAN SOUND-SPEED MODEL CTABLE

This model represents the sound-speed profile by a sequence of cubic segments such that the sound-speed gradient is continuous through each profile point. This is not a cubic spline; the coefficients of the cubic fit in each segment depend on only the four nearest profile points.

The coefficients of the cubic are calculated as follows: each set of three successive points in the profile is first fit with a quadratic. The slope of that quadratic at the middle profile point is then assigned to that profile point. This procedure assigns a slope to every profile point except the first and last. A slope of zero is assigned to the first and last point. Between each pair of profile points the coefficients of the cubic are chosen so that the curve goes through the two points and matches the assigned slope at the two points. Those four conditions determine the four coefficients. Both the sound-speed and its gradient are continuous throughout the profile, even at the profile points.

Specify--

the model check number for CTABLE = 8.0 (W200)

the input data-format code = _____(W201)

an input data-set identification number = _____(W202)

an 80-character description of the profile:

_____

and the profile values:

the number of points in the profile, n = _____

the profile: height (km, m) Sound speed (Km/s, m/s)

OTHER MODELS REQUIRED: Subroutine GAUSEL and any sound-speed-perturbation model. Use NPSPEED if no perturbations are desired.

FORM TO SPECIFY INPUT DATA FOR SOUND-SPEED
PERTURBATION MODEL CBLOB2

An increase (or decrease) in sound speed in a localized region that decays in a Gaussian manner in all three spatial directions.

$$C^2(r,\theta,\phi) = C_o{}^2(r,\theta,\phi)\left(1 + \Delta \exp\left\{-\left(\frac{z-z_o}{W_z}\right)^2 - \left(\frac{\theta-\theta_o}{W_\theta}\right)^2 - \left(\frac{\phi-\phi_o}{W_\phi}\right)^2\right\}\right)$$

$C_o{}^2(r,\theta,\phi)$ is the square of the sound speed specified by a sound-speed model. (r,θ,ϕ) are the coordinates of the ray point in an Earth-centered spherical polar-coordinate system. $\theta_o = \pi/2 - \lambda_o$ and $z = r - r_e$, where r_e is the Earth radius.

Specify--

the model check for subroutine CBLOB2 = 2.0 (W175)

the input data-format code = _____(W176)

an input data-set identification number = _____(W177)

an 80-character description for the sound-speed perturbation model, including description of parameter values:

---

the strength of the fractional increase (or decrease), Δ = _____(W178)

the height of maximum effect, z_o = _____km (W179)

the latitude of maximum effect, λ_o = _____rad, deg, km N (W180)

the longitude of maximum effect, ϕ_o = _____rad, deg, km E (W181)

the Gaussian width in height of the effect, W_z = _____km (W182)*

the meridional width of the effect, W_θ = _____rad, deg, km (W183)*

the zonal width of the effect, W_ϕ = _____rad, deg, km (W184)*

OTHER MODELS REQUIRED: none.

---

* Setting W_z, W_θ, or W_ϕ = zero results in no space variation in that direction.

FORM TO SPECIFY INPUT DATA FOR SOUND-SPEED
PERTURBATION MODEL CBLOB3

An increase (or decrease) in sound speed in up to three localized regions that decays in a Gaussian manner in all three spatial directions.

$$C(r,\theta,\phi) = C_o(r,\theta,\phi)\{1 + \sum_{i=1}^{n} \Delta_i \exp[-(\frac{z-z_i}{W_{zi}})^2 - (\frac{\theta-\theta_i}{W_{\theta i}})^2 - (\frac{\phi-\phi_i}{W_{\phi i}})^2]\}$$

$C_o(r,\theta,\phi)$ is a background sound-speed model, (r,θ,ϕ) are Earth-centered spherical-polar coordinates. $z = r - r_e$, where r_e is the Earth's radius. $\lambda_i = \pi/2 - \theta_i$ is the latitude.

Specify--

the model check number for subroutine CBLOB3 = ___3.___(W175)

the input data format code = _____(W176)

an input data set identification number = _____(W177)

an 80-character description for the sound-speed perturbation model, including description of parameter set:

---

the number of Gaussian blobs, n = _____(W178)

Δ_1 = _____(W179), Δ_2 = _____(W180), Δ_3 = _____(W181)

z_1 = _____(W182), z_2 = _____(W183), z_3 = _____(W184)

λ_1 = _____(W185), λ_2 = _____(W186), λ_3 = _____(W187)rad,deg,km N

ϕ_1 = _____(W188), ϕ_2 = _____(W189), ϕ_3 = _____(W190)rad,deg,km E

*W_{z1} = _____(W191), W_{z2} = _____(W192), W_{z3} = _____(W193)km

*$W_{\theta 1}$ = _____(W194), $W_{\theta 2}$ = _____(W195), $W_{\theta 3}$ = _____(W196)rad,deg,km

*$W_{\phi 1}$ = _____(W197), $W_{\phi 2}$ = _____(W198), $W_{\phi 3}$ = _____(W199)rad,deg,km

OTHER MODELS REQUIRED: None.

---

*Setting a W = 0 results in no space variation in that direction.

FORM TO SPECIFY INPUT DATA FOR SOUND-SPEED
PERTURBATION MODEL CBLOB4

An increase (or decrease) in squared sound speed in up to three localized regions that decays in a Gaussian manner in all three spatial directions.

$$C^2(r,\theta,\phi) = C_o^2(r,\theta,\phi)\{1 + \sum_{i=1}^{n} \Delta_i \exp[-(\frac{z-z_i}{W_{zi}})^2 - (\frac{\theta-\theta_i}{W_{\theta i}})^2 - (\frac{\phi-\phi_i}{W_{\phi i}})^2]\}$$

$C_o(r,\theta,\phi)$ is a background sound-speed model, (r,θ,ϕ) are Earth-centered spherical-polar coordinates. $z = r - r_e$, where r_e is the Earth's radius. $\lambda_i = \pi/2 - \theta_i$ is the latitude.

Specify--

the model check number for subroutine CBLOB4 = ____4.____(W175)

the input data format code = _____(W176)

an input data set identification number = _____(W177)

an 80-character description for the sound-speed perturbation model, including description of parameter set:

---

the number of Gaussian blobs, n = _____(W178)

Δ_1 = _____(W179), Δ_2 = _____(W180), Δ_3 = _____(W181)

z_1 = _____(W182), z_2 = _____(W183), z_3 = _____(W184)

θ_1 = _____(W185), θ_2 = _____(W186), θ_3 = _____(W187)rad,deg,km N

ϕ_1 = _____(W188), ϕ_2 = _____(W189), ϕ_3 = _____(W190)rad,deg,km E

*W_{z1} = _____(W191), W_{z2} = _____(W192), W_{z3} = _____(W193)km

*$W_{\theta 1}$ = _____(W194), $W_{\theta 2}$ = _____(W195), $W_{\theta 3}$ = _____(W196)rad,deg,km

*$W_{\phi 1}$ = _____(W197), $W_{\phi 2}$ = _____(W198), $W_{\phi 3}$ = _____(W199)rad,deg,km

OTHER MODELS REQUIRED: None.

---

*Setting a W = 0 results in no space variation in that direction.

FORM TO SPECIFY INPUT DATA FOR
OCEAN ABSORPTION MODEL SLLOSS

This absorption model depends only on the acoustic wave frequency ω (rad/s), according to the formula

$$\alpha = a \frac{\omega^2}{\omega_1^2} + b \frac{\omega^2}{\omega_2^2 + \omega^2}$$

The following values for the coefficients correspond to the model of Skretting and Leroy (1971)*:

a = 0.006 dB/km ; ω_1 = 6283.2 rad/s (= 1000.0 Hz)

b = 0.2635 dB/km ; ω_2 = 10,681.4 rad/s (= 1700.0 Hz)

Specify--

the model check for SLLOSS = ____1.____(W500)

the input data-format code = _____(W501)

an input data-set identification number = _____(W502)

an 80-character description of the model, including parameter values:

---

a = _____ nepers/km, dB/km (W503)

b = _____ nepers/km, dB/km (W504)

ω_1 = _____ rad/s Hz (505)

ω_2 = _____ rad/s Hz (W506)

Other models required: any absorption perturbation model. Use NPABSR if no perturbation is desired.

---

*see References.

FORM TO SPECIFY INPUT DATA FOR
OCEAN-BOTTOM MODEL GHORIZ

A constant-height bottom model, i.e., a sphere concentric with the Earth.

$$g(r,\theta,\phi) = h - z_o \ ,$$

where $h = r - r_e$,

$$\frac{\partial g}{\partial r} = 1 \ , \ \frac{\partial g}{\partial \theta} = 0 \ , \ \frac{\partial g}{\partial \phi} = 0 \ ,$$

$$\frac{\partial^2 g}{\partial r^2} = \frac{\partial^2 g}{\partial r \partial \theta} = \frac{\partial^2 g}{\partial \theta \partial r} = \frac{\partial^2 g}{\partial r \partial \phi} = \frac{\partial^2 g}{\partial \phi \partial r} = \frac{\partial^2 g}{2\theta 2} = \frac{\partial^2 g}{\partial \theta \partial \phi} = \frac{\partial^2 g}{\partial \phi \partial \theta} = \frac{\partial^2 g}{\partial \phi^2} = 0 \ ,$$

and r_e is the radius of the Earth.

Specify --

The model check number for GHORIZ = _____1.0_____(W300)

The input data-format code number = _____(W301)

The input data-set identification number = _____(W302)

an 80-character description of the model including parameters:

The constant bottom height, z_o = _____km, (W303)

(negative if below mean sea level)

OTHER MODELS REQUIRED: Any bottom-perturbation model. Use NPBOTM if no perturbation is desired.

FORM TO SPECIFY INPUT DATA FOR
OCEAN-BOTTOM MODEL GLORENZ

An east-west Lorentzian-shaped ridge.

$$g(r,\theta,\phi) = h - z \ ,$$

where $h = r - r_e$,

$$z = z_o/(1 + ((\theta-\theta_o)/\Delta\theta)^2) + z_B$$

$$\theta_o = \pi/2 - \lambda_o \ ,$$

and r_e is the radius of the Earth.

Specify --

the model check number for GLORENZ = 4.0 (W300)

the data input format code number = _____ (W301)

the data set identification number = _____ (W302)

an 80-character description of the model including parameters:

_____

the height of the ridge, z_o = _____ km, m (W303)

the latitude of the ridge center, λ_o = _____ rad, deg, km (W304)

the half-width of the ridge, $\Delta\theta$ = _____ rad, deg, km (W305)

the height above sea level of the
base of the ridge (negative if below sea level) z_B = _____ m,km (W306)

OTHER MODELS REQUIRED: Any bottom perturbation model. Use NPBOTM if no perturbation is desired.

FORM TO SPECIFY INPUT DATA FOR
OCEAN-BOTTOM MODEL GTANH

This model represents the ocean bottom by a sequence of linear segments that are smoothly joined by hyperbolic functions:

$$g(r,\theta,\phi) = h - z(\theta), \text{ where}$$

$$z(\theta) = z_o + \frac{c_1}{2}(\theta-\theta_o) - \sum_{i=1}^{n} \delta_i \left(\frac{c_{i+1} - c_i}{2}\right) \ln\left\{\frac{\cosh\left(\frac{\theta-\theta_i}{\delta_i}\right)}{\cosh\left(\frac{\theta_i-\theta_o}{\delta_i}\right)}\right\} + \frac{c_{n+1}}{2}(\theta-\theta_o)$$

$$\frac{dz}{d\theta} = c_1 + \sum_{i=1}^{n} \left(\frac{c_{i+1} - c_i}{2}\right)\left\{-\tanh\left(\frac{\theta-\theta_i}{\delta_i}\right) + 1\right\}$$

$$c_i = (z_i - z_{i-1})/(\theta_i - \theta_{i-1})$$

$h = r - r_e$, where r_e is the Earth radius, and r is the radial coordinate of the ray point. $\theta_i = \pi/2 - \lambda_i$. Thus, δ_i is the half-thickness of a region centered at approximately θ_i, in which $dz/d\theta$ changes from c_i to c_{i+1}. Start by drawing a profile using linear segments, and θ_i and z_i from the corners. Then select δ_i to round the corners. The final profile will not go through (θ_i, z_i).

Specify--

the model check for GTANH = 3.0 (W300)

the input data-format code = _____ (W301)

an input data-set identification number = _____ (W302)

an 80-character description of the model with parameters:

_____

and the profile values:

the number of points in the profile -2 = n = _____

the profile:

i	λ_i (rad,deg)	z_i (km,m)	δ_i (rad,deg)

OTHER MODELS REQUIRED: Any bottom-perturbation model. Use NPBOTM if no perturbation is desired. FUNCTION ALCOSH.

FORM TO SPECIFY INPUT DATA
FOR OCEAN SURFACE MODEL SHORIZ

An ocean surface model that is a horizontal (i.e., a sphere concentric with the earth). The ocean surface is where the following function is zero.

$$s(r, \theta, \phi) = z_s - h$$

where

$$h = r - r_e$$

and

r_e is the earth radius

Specify--

the model check number for subroutine SHORIZ = 1.0 (W350)
the data input format code number = _____(W351)
the data set identification number = _____(W352)
an 80-character description of the model including parameters:

---

the height of the ocean surface above
mean sea level, z_s = _____km, m (W353)

OTHER MODELS REQUIRED: Any surface-perturbation model. Use NPSURF if no perturbation is desired.

FORM TO SPECIFY INPUT DATA
FOR RECEIVER-SURFACE MODEL RHORIZ

A receiver-surface model that is a horizontal surface (i.e., a sphere concentric with the Earth).

$$f(r,\theta,\phi) = z_R - h \ ,$$

where

$$h = r - r_e$$

and

r_e is the Earth radius

$$\frac{\partial f}{\partial t} = \frac{\partial f}{\partial \theta} = \frac{\partial f}{\partial \phi} = 0$$

$$\frac{\partial f}{\partial r} = 1.0 \ .$$

Specify--

the model check number for subroutine RHORIZ = ____1.0____(W275)

the input data-format code number = _____(W276)

an 80-character description of the model including parameters:

_____

the receiver surface height, z_R = _____km (W20)

OTHER MODELS REQUIRED: none.

FORM TO SPECIFY INPUT DATA
FOR RECEIVER-SURFACE MODEL RBOTM

A receiver-surface model in which the receiver surface is a fixed height above the ocean bottom.

$$f(r,\theta,\phi) = g(r,\theta,\phi) + z_R$$

$$\frac{\partial f}{\partial r} = \frac{\partial g}{\partial r}\ ,\ \frac{\partial f}{\partial \theta} = \frac{\partial g}{\partial \phi}\ ,\ \frac{\partial f}{\partial \phi} = \frac{\partial g}{\partial \phi}\ ,$$

where $g(r,\theta,\phi)$ and its derivatives are specified in common block /GG/ by the terrain model.

Specify--

the model check number for subroutine RBOTM = 3.0 (W275)

the input data-format code number = _____ (W276)

an 80-character description of the model including parameters:

---

the height of the receiver surface above the ocean bottom, z_R = _____km (W20)

OTHER MODELS REQUIRED: Any ocean-bottom model.

FORM TO SPECIFY INPUT DATA FOR
RECEIVER-SURFACE MODEL RVERT

A receiver surface that is a vertical (conical) surface a constant distance from a given origin on the Earth's surface

$$f(r,\theta,\phi) = \sin\lambda_o \cos\theta + \cos\lambda_o \sin\theta \cos(\phi-\phi_o) - \cos\alpha_o$$

$$\frac{\partial f}{\partial t} = \frac{\partial f}{\partial r} = 0$$

$$\frac{\partial f}{\partial \theta} = -\sin\lambda_o \sin\theta + \cos\lambda_o \cos\theta \cos(\phi-\phi_o)$$

$$\frac{\partial f}{\partial \phi} = -\cos\lambda_o \sin\theta \sin(\phi-\phi_o) \cdot$$

Specify--

the model check number for subroutine RVERT = 2.0 (W275)

the input data-format code number = _____(W276)

an 80-character description of the model including parameters:

---

the distance of the surface from the origin,

α_o = _____rad, deg, km (278)

the latitude of the origin, λ_o = _____rad, deg, km N (W279)

the longitude of the origin, ϕ_o = _____rad, deg, km E (W280)

OTHER MODELS REQUIRED: none.

APPENDIX C: FR-80 PLOT PACKAGE AND DISSPLA INTERFACE

This appendix describes the plotting commands used by DDPLOT, our local microfilm plotting system, and also an interface called DDSPLA to the DISSPLA* plot package in common use. Figure C1 shows the steps necessary to obtain graphical output from HARPO, if you have DISSPLA. File 6 of the distribution tape contains PROGRAM TAPRD, which reads the graphics data file. File 7 of the distribution tape contains the DISSPLA interface routines DDSPLA. If you do not have DISSPLA and want graphical output on your own plotting system, you will have to insert the equivalent instructions used by your system into the skeleton graphics interface routines DDALT (tape file 8). This information was taken with permission from "User's Guide to Cathode-Ray Plotter Subroutines" by L. David Lewis, ESSA Technical Memorandum ERL TM-ORSS5, January 1970. The routines used in this version of HARPO assume DISSPLA version 9.0 and are listed in Appendix D.

The Information International Inc. FR-80 Microfilm Recorder, under control of the NOAA Boulder CDC-CYBER 840 computer, plots data on the face of a high-resolution cathode ray tube, which is photographed onto standard size, perforated, 35-mm film.

The plotting area, called a frame, is a square. Plotting positions are described in rectangular coordinates. Coordinate values are integers in the range 0 - 1023; (0,0) is the "lower left-hand corner."

Plotting specifications are transmitted to the DDPLOT/DDSPLA interface routines via the following two COMMON blocks.

```
COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY
COMMON/DDREZ/DDHIX,DDHIY
```

---

*DISSPLA is the proprietary product of ISSCO, Inc.

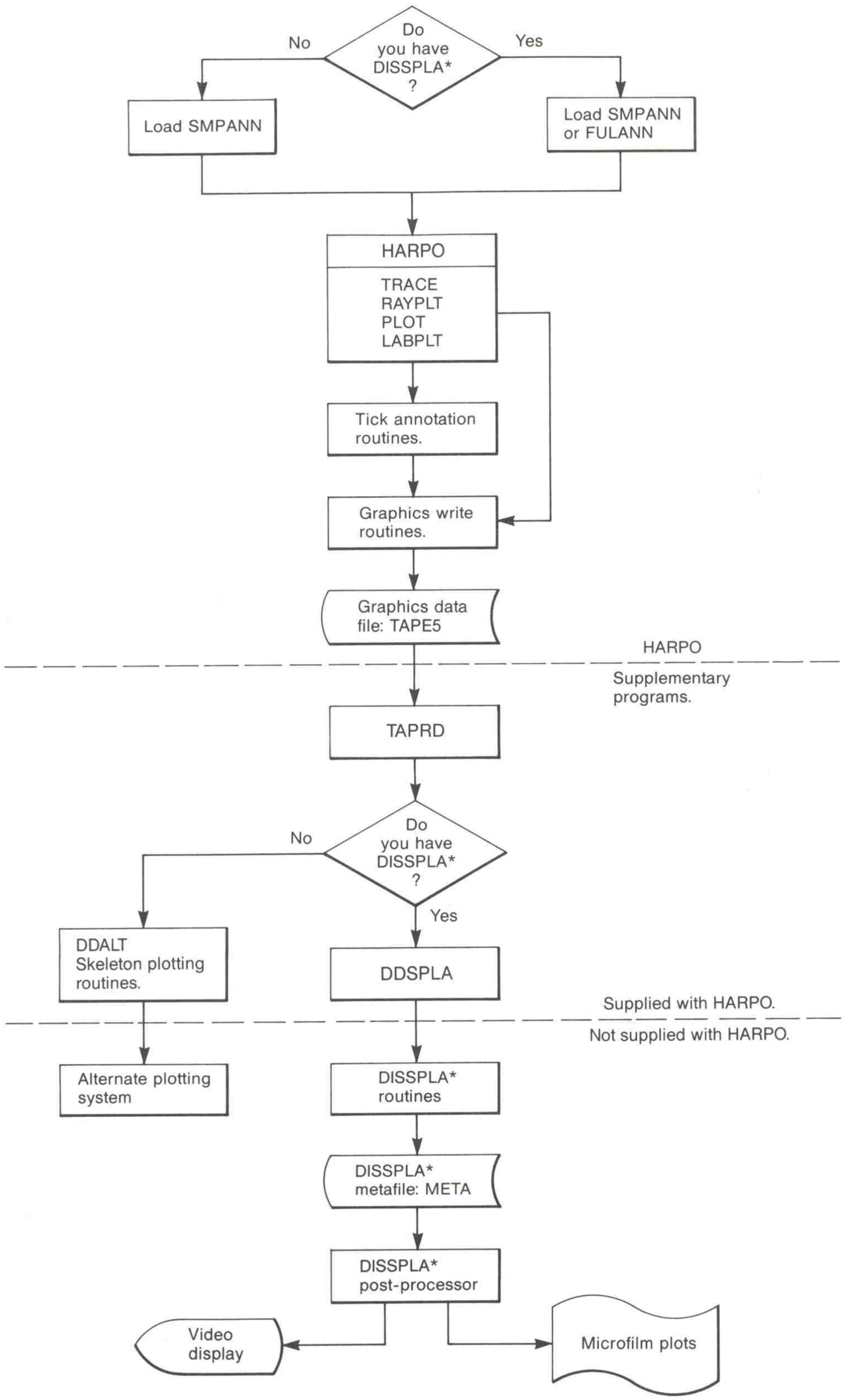


Figure C1. An organization chart that shows how graphical output is produced by a series of programs, some of which are a part of HARPO, and others of which are either supplied along with HARPO or are commerical packages.

The usage of each of the ten variables is listed below, followed by an explanation of the subroutine calls.

IN	Intensity. IN=0 specifies normal intensity. IN=1 specifies high intensity.
IOR	Orientation IOR=0 specifies upright orientation. IOR=1 specifies rotated orientation (90° counter-clockwise).
IT	Italics (Font). IT=0 specifies non-italic (Roman) symbols. IT=1 specifies italic symbols.
IS	Symbol size. IS=0 specifies miniature size. IS=1 specifies small size. IS=2 specifies medium size. IS=3 specifies large size.
IC	Symbol case. IC=0 specifies uppercase. IC=1 specifies lowercase.
ICC	Character code, 0-63 (R1 format). ICC and IC together specify the symbol plotted.
IX(DDHIX)	X-coordinate, 0-1023 (X-coordinate, real value, 0.0-1023.0)
IY(DDHIY)	Y-coordinate, 0-1023 (Y-coordinate, real value, 0.0-1023.0).

CALL DDINIT (N,ID) is required to initialize the plotting process.

ID is a string of characters to identify the person getting the plot and giving the telephone extension and the place to deliver the microfilm plot.

N is the number of characters in the string "ID."

CALL DDBP	defines a vector origin in position IX, IY.
CALL DDVC	plots a vector (straight line), with intensity IN, from the vector origin defined by the previous DDBP or DDVC call, to the vector end position at IX, IY. A single call to DDBP followed by successive calls to DDVC (with changing IX and IY) plots connected vectors.
CALL DDTAB	Initializes tabular plotting.

CALL DDTEXT (N,NT) plots a given array in a tabular mode after initiating tabular plotting by using DDTAB, as described above. NT is an array of length N, containing "text" for tabular plotting. Text consists of character codes, packed eight per word (A8 Format). Text characters are plotted as tabular symbols until the command character ≠ (octal code 14, card code 4,8, or the alphabetic shift counterpart of the = on the keypunch) occurs. The command character is not plotted. DDTEXT interprets the next character as a command; after the command is processed, tabular plotting resumes until ≠ is again encountered. ≠ means end of text: DDTEXT returns to the calling routine.

CALL DDFR causes a frame advance operation. Plotting on the current frame is completed, and the film advances to the next frame.

CALL DDEND empties the plot buffer and releases the plotting command file to the microfilm plot queue.

DISSPLA CALLS

HARPO calls the following DISSPLA routines directly, rather than through the DDPLOT package. Therefore, if you want to do plotting with a plotting package other than DISSPLA, you will have to convert the following DISSPLA calls to corresponding calls in your plotting package.

CALL DASH sets dashed-line mode. That is, all plotted curves will be dashed instead of solid.

CALL RESET('DASH') sets solid-line mode. That is, all plotted curves will be solid lines after this call.

HARPO calls a routine named SETANN. SETANN is an entry point in SUBROUTINE FULANN and also in SUBROUTINE SMPANN. When running HARPA, you must make a choice whether to load FULANN or SMPANN. If you do not have the DISSPLA plotting package, then load SMPANN because it makes no special character-generating calls to DISSPLA routines. If you do have the DISSPLA plotting package, then load FULANN. SUBROUTINE FULANN calls the following DISSPLA subroutines directly. If you have the DISSPLA plotting package, then the manual will explain the meaning of these routines. If you do not have the DISSPLA plotting package, then load SMPANN, and ignore the routines HEIGHT, MX1ALF, MX2ALF, and SCMPLX.

Following is a list of the routines on Files 6, 7 and 8 of the distribution tape.

- TAPRD -- Graphics File Read Routine (Tape File 6)

PROGRAM TAPRD -- Program to read graphics output file and call graphics interface routines.

- DDSPLA -- DISSPLA Interface Routines (Tape File 7)

SUBROUTINE MYJSUB -- Intercepts transmission of titles and axis labels so that minus signs on axis labels can be removed.

SUBROUTINE DDINIT -- Initializes plotting process.

SUBROUTINE DDBP -- Sets a vector origin.

SUBROUTINE DDVC -- Plots a vector.

SUBROUTINE DDEND -- Emties plot buffer and releases plotting command file to microfilm plot queue.

SUBROUTINE DDTEXT -- Writes an array (character string) in tabular text mode.

SUBROUTINE DDTAB -- Sends instruction that initializes tabular (text) plotting.

SUBROUTINE DDFR -- Sends instruction to advance a microfilm frame.

- DDALT -- Skeleton Graphics Interface Routines (Tape File 8)

SUBROUTINE DDINIT -- Initializes plotting process.

SUBROUTINE DDBP -- Sets a vector origin.

SUBROUTINE DDVC -- Plots a vector.

SUBROUTINE DDEND -- Empties plot buffer and releases plotting command file to microfilm plot queue.

SUBROUTINE DDTEXT -- Writes an array (character string) in tabular text mode.

SUBROUTINE DDTAB -- Sends instruction that initializes tabular (text) plotting.

SUBROUTINE DDFR -- Sends instruction to advance a microfilm frame.

SUBROUTINE DASH -- Sets dashed-line mode; that is, all plotted curves will be dashed instead of solid after a call to subroutine DASH.

SUBROUTINE RESET('DASH') -- Sets solid-line mode; that is, all plotted curves will be solid lines after this call.

SUBROUTINE SCMPLX -- If you do not have the DISSPLA plotting package, load SUBROUTINE SMPANN instead of SUBROUTINE FULANN, and you can ignore this routine.

SUBROUTINE MX1ALF -- If you do not have the DISSPLA plotting package, load SUBROUTINE SMPANN instead of SUBROUTINE FULANN, and you can ignore this routine.

SUBROUTINE MX2ALF -- If you do not have the DISSPLA plotting package, load SUBROUTINE SMPANN instead of SUBROUTINE FULANN, and you can ignore this routine.

SUBROUTINE HEIGHT -- If you do not have the DISSPLA plotting package, load SUBROUTINE SMPANN instead of SUBROUTINE FULANN, and you can ignore this routine.

The following routines are not used by HARPO, but "dummy" versions must be present when loading TAPRD because they are referred to by the general-purpose TAPRD program.

COMPRS, GRACE, PHYSOR, PAGE, SCLPIC, XREVTK, YREVTK, INTAXS, TITLE, FRAME, GRAF, MARKER, SYSTEM, CURVE, ENDPL, DONEPL, XTICKS, YTICKS, MYJACT, NOBRDR, GRAFB, AREA2D, STRTPT, CONNPT, ANGLE, MESSAG, XNONUM, YNONUM, XGRAXS, and YGRAXS.

APPENDIX D: FORTRAN SOURCE CODE LISTING

This appendix contains the FORTRAN source-code listing for HARPO, including all of its subroutines and atmospheric models. Their order is the same as the order of the programs in Files 3 through 7 of the distribution tape and the list in Section 7.1. Table D1 lists the routines in alphabetical order and the page where the source code can be found.

```
**********************************************************************
WARNING:  THIS LISTING IS PROVIDED FOR INFORMATION PURPOSES ONLY.  THOSE WHO
WANT TO USE THE PROGRAM MUST OBTAIN THE SOURCE CODE ON MAGNETIC TAPE FROM THE
AUTHORS (SEE SECTION 3.1).  COPYING THE CODE LISTED HERE WILL NOT PRODUCE A
USABLE PROGRAM.
**********************************************************************
```

Table D1 -- Alphabetical List of Source-Code Modules

Module Name	Page	Module Name	Page
ALCOSH	316	NPSURF	420
ANCNL	343	NUMSTG	312
ANCWL	349	OCNHD	306
ARCTIC	337	OPNREP	315
AWCNL	346	OVERRD	316
AWCWL	352	PCROSS	275
BACKUP	282	PLOT	324
CBLOB2	393	PLTANH	330
CBLOB3	396	PLTANOT	333
CBLOB4	399	PLTHLB	330
CLEAR	267	PLTLB	336
CONBLK	292	PRINTR	298
CSMUNK1	381	PUTDES	311
CSMUNK2	384	PUTKCT	314
CSSPOK2	377	PUTKST	313
CSSPOKE	374	RAYPLT	319
CSTANH	370	RAYTRC	245
CTABLE	387	RBOTM	424
CTANH	367	RCROSS	275
DASH	341,440	READW	259
DDBP	338,435,439	READW1	253
DDEND	341,436,439	REFLECT	284
DDFR	341,437,440	RENORM	298
DDINIT	338,435,439	RERR	312
DDTAB	340,437,440	RERROR	313
DDTEXT	340,436,439	RESET	341,440
DDVC	339,436,439	RHORIZ	422
DFCNST	251	RKAM	278
DFSYS	249	RKAM1	280
DRAWTKS	335	RVERT	427
FIT	286	SCMPLX	342,440
FULANN	430	SET2	298
GAUSEL	317	SETTRC	297
GET	289	SETXY	332
GET1	287	SFILL	312
GHORIZ	407	SFILTR	316
GLORENZ	409	SHORIZ	418
GRAFB	437	SLLOSS	403
GTANH	412	SMPANN	429
GTUNIT	257	SREAD1	258
HAMLTN	276	STOPIT	313
HEIGHT	342,441	STRIM	312
ITEST	291	TAPRD	432
ITOC	291	TIKLINE	332
LABPLT	326	TITLEW	437
MX1ALF	342,440	TRACE	269
MX2ALF	342,441	UCON	267
MYJSUB	435	VVORTX3	359
ND2B	267	WCHANGE	297
NPABSR	405	WGAUSS2	362
NPBOTM	415	WLINEAR	357
NPCURR	365	ZAPFIL	315
NPSPEED	391		

D.1 RAY-TRACING CORE (Tape File 3)

```
      PROGRAM RAYTRC                                                    RAQC0020
C     MAIN PROGRAM FOR THE RAY TRACING PACKAGE.                         RAQC0030
C          SETS THE INITIAL CONDITIONS FOR EACH RAY AND CALLS TRACE     RAQC0040
C                                                                       RAQC0050
C     COMMON DECK "RAYDEV" INSERTED HERE                                CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                              CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN               CRAY0050
      LOGICAL WCHANGE,FIRST                                             RAQC0070
      REAL WS(400)                                                      RAQC0080
C     COMMON DECK "FILEC" INSERTED HERE                                 CFIL0020
      COMMON /FILEC/NPLTDP                                              CFIL0040
C     COMMON DECK "CERR" INSERTED HERE                                  CERR0020
      COMMON/ERR/NERG,NERR,NERT,NERP                                    CERR0030
C     COMMON DECK "GG" INSERTED HERE                                    CGG 0020
      REAL MODG                                                         CGG 0040
      COMMON/GG/MODG(4)                                                 CGG 0050
      COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                  CGG 0060
      COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME            CGG 0070
C     COMMON DECK "HDR" INSERTED HERE                                   CHDR0020
      CHARACTER*10 INITID*80,DAT,TOD                                    CHDR0040
      COMMON/HDR/SEC                                                    CHDR0050
      COMMON/HDRC/INITID,DAT,TOD                                        CHDR0060
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "FLAG" INSERTED HERE                                  CFLA0020
      LOGICAL NEWWR,NEWWP,NEWTRC,PENET                                  CFLA0040
      COMMON /FLG/ NTYP,NEWWR,NEWWP,NEWTRC,PENET,LINES,IHOP,HPUNCH      CFLA0050
      COMMON/FLGP/NSET                                                  CFLA0060
C     COMMON DECK "RINPLEX" INSERTED HERE                               CRIN0020
      REAL KAY2,KAY2I                                                   CRIN0040
      COMPLEX PNP,POLAR,LPOLAR                                          CRIN0050
      LOGICAL SPACE                                                     CRIN0060
      CHARACTER DISPM*6                                                 CRIN0070
      COMMON/RINPL/DISPM                                                CRIN0080
      COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                 CRIN0090
      COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I                             CRIN0100
      COMMON/RIN/PNP(10),POLAR,LPOLAR,SGN                               CRIN0110
C     COMMON DECK "RK" INSERTED HERE                                    CRK 0020
C     DEFINE SIZE REQUIRED FOR RAY STATE SAVE ARRAY                     CRK 0040
      PARAMETER (LRKAMS=87+2*100,NXRKMS=12+LRKAMS,MXEQPT=21)            CRK 0050
      PARAMETER (NRKSAV=NXRKMS+MXEQPT-1)                                CRK 0060
      COMMON /RK/ NEQS,STEP,MODE,E1MAX,E1MIN,E2MAX,E2MIN,FACT,RSTART    CRK 0070
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C      COMMON DECK "WWR" INSERTED HERE                                  CWWR0020
       PARAMETER (NWARSZ=1000)                                          CWW10030
       COMMON/WW/ID(10),MAXW,W(NWARSZ)                                  CWW10040
       REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                               CWW20020
       EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),   CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
```

```
4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
C                                                                  RAQC0190
      CHARACTER TMP80*80                                           RAQC0200
C                                                                  RAQC0210
      DATA FIRST/.TRUE./                                           RAQC0220
C                                                                  RAQC0230
C   SET CONSTANTS , INITITALIZE FILES  AND MAKE SYSTEM CALLS       RAQC0240
      CALL SYSINI                                                  RAQC0250
      CALL CONBLK                                                  RAQC0260
      CALL SETDSP                                                  RAQC0270
      CALL STDINI                                                  RAQC0280
C     INITIALIZE LINE COUNTS FOR CURRENT AND LAST HEADER           RAQC0290
      LINES=0                                                      RAQC0300
      LHDRPG=0                                                     RAQC0310
C                                                                  RAQC0320
      REWIND NRYIND                                                RAQC0330
C  INITIALIZE RAYSET FILE                                          RAQC0340
5     READ(NRYIND,'(A)',END=8) TMP80                               RAQC0350
      WRITE(9,'(A)') TMP80                                         RAQC0360
      GO TO 5                                                      RAQC0370
C                                                                  RAQC0380
8     WRITE(9,'(A)') '****SOFT EOF***++++'                         RAQC0390
      REWIND NRYIND                                                RAQC0400
C     READ IN USERS NAME AND TELEPHONE EXTENSION FOR IDENTIFYING   RAQC0410
C     MICROFILM PLOTS.                                             RAQC0420
C                                                                  RAQC0430
C     REPOSITION DINP AFTER IDENTIFIER                             RAQC0440
      READ(NRYIND,'(A)',END=6000) TMP80                            RAQC0450
C                                                                  RAQC0460
C********* READ W ARRAY AND PRINT NON-ZERO VALUES                  RAQC0470
      IRUN=0                                                       RAQC0480
      NSET=0                                                       RAQC0490
C                                                                  RAQC0500
10    CALL READW                                                   RAQC0510
      NSET=NSET+1                                                  RAQC0520
      ICODE=ND2B(INT(RAYFNC))                                      RAQC0530
C PROCESS RAYPATH CALCULATIONS ONLY IF W(29) IS BEING USED         RAQC0540
      IF((RAYFNC.NE.0.).AND.(AND(ICODE,4).EQ.0)) GO TO 10          RAQC0550
C                                                                  RAQC0560
C********* LET ROUTINES PRINTR AND RAYPLT KNOW THERE IS A NEW W ARRAY RAQC0570
      NEWWP=.TRUE.                                                 RAQC0580
      NEWWR=.TRUE.                                                 RAQC0590
      IRUN=IRUN+1                                                  RAQC0600
C                                                                  RAQC0610
C     SET BINARY RAY FILE UNIT TO ZERO IF NO OUTPUT WANTED         RAQC0620
      NPLTDP=0                                                     RAQC0630
      IF(BINRAY.NE.0.0) NPLTDP=NDEVBIN                             RAQC0640
```

```
C                                                                       RAQC0650
C******** INITIALIZE THE MODELS VIA 'DISPER'                            RAQC0660
      CALL IDISPER                                                      RAQC0670
C                                                                       RAQC0680
      IF(FIRST.AND.PLT.NE.0.) CALL DDINIT(8,INITID)                     RAQC0690
      IF(PLT.NE.0.) FIRST=.FALSE.                                       RAQC0700
C                                                                       RAQC0710
      OW=0.                                                             RAQC0720
      BETA=0.                                                           RAQC0730
      AZ1=0.                                                            RAQC0740
C                                                                       RAQC0750
      CALL HEADER1                                                      RAQC0760
C                                                                       RAQC0770
C********* PRINT OUT THE CONTENTS OF THE 'W' ARRAY                      RAQC0780
C********* DETERMINE TRANSMITTER LOCATION IN COMPUTATIONAL COORDINATE   RAQC0790
C********* SYSTEM (GEOMAGNETIC COORDINATES IF DIPOLE FIELD IS USED)     RAQC0800
      SP=SIN (PLAT)                                                     RAQC0810
      CP=SIN (PID2-PLAT)                                                RAQC0820
      SDPH=SIN (TLON-PLON)                                              RAQC0830
      CDPH=SIN (PID2-(TLON-PLON))                                       RAQC0840
      SL=SIN (TLAT)                                                     RAQC0850
      CL=SIN (PID2-TLAT)                                                RAQC0860
      ALPHA=ATAN2(-SDPH*CP,-CDPH*CP*SL+SP*CL)                           RAQC0870
      TH0=ACOS (CDPH*CP*CL+SP*SL)                                       RAQC0880
      PH0=ATAN2(SDPH*CL,CDPH*SP*CL-CP*SL)                               RAQC0890
C                                                                       RAQC0900
      R=EARTHR                                                          RAQC0910
      TH=TH0                                                            RAQC0920
      PH=PH0                                                            RAQC0930
      CALL TOPOG                                                        RAQC0940
C                                                                       RAQC0950
C     OBTAIN ABSOLUTE HEIGHT OF THE TRANSMITTER.                        RAQC0960
C     IF IT WAS SPECIFIED AS RELATIVE TO THE TERRAIN, THEN REMOVE       RAQC0970
C     THE FLAG VALUE 10**40 WHICH WAS ADDED AT INPUT.                   RAQC0980
C                                                                       RAQC0990
      TMP=XMTRH                                                         RAQC1000
      IF(XMTRH .EQ. 1.E-40) XMTRH=0.0                                   RAQC1010
      IF( ABS(XMTRH) .GE. 1.E20 ) XMTRH=XMTRH*1.E-40                    RAQC1020
      IF(TMP.NE.XMTRH) XMTRH=XMTRH-G/PGR                                RAQC1030
C                                                                       RAQC1040
C     CHECK THAT TRANSMITTER IS ABOVE TERRAIN.                          RAQC1050
C                                                                       RAQC1060
      IF(-G/PGR .LE. XMTRH) GO TO 655                                   RAQC1070
C                                                                       RAQC1080
      PRINT 640, IRUN                                                   RAQC1090
      WRITE(3,640) IRUN                                                 RAQC1100
640   FORMAT('0****** TRANSMITTER BELOW TERRAIN. RUN ',I3               RAQC1110
     1 ,' TERMINATED.'/'SEE W-ARRAY PRINTOUT. INPUT CONTINUES.'//)      RAQC1120
      CALL SETOVR                                                       RAQC1130
      CALL PRINTW                                                       RAQC1140
      GO TO 10                                                          RAQC1150
C                                                                       RAQC1160
C     PREVENT RERUNNING SAME MODEL CASES                                RAQC1170
655   IF(IRUN.EQ.1) GO TO 12                                            RAQC1180
      IF(RUNSUP.NE.0.0 .AND. .NOT.WCHANGE(WS(1),W(1))) THEN             RAQC1190
```

```
         PRINT *,'SUPPRESSING REDUNDANT RUNSET #',NSET                  RAQC1200
         GO TO 10                                                       RAQC1210
      ENDIF                                                             RAQC1220
C                                                                       RAQC1230
12    PRINT *,'PROCESSING FOR RUNSET #',NSET                            RAQC1240
C                                                                       RAQC1250
      CALL RMOVE(WS,W,400)                                              RAQC1260
C                                                                       RAQC1270
      CALL SETOVR                                                       RAQC1280
      CALL PRINTW                                                       RAQC1290
      CALL IPRINTR                                                      RAQC1300
C******** INITIALIZE PRINT CONTROL PARAMTERS                            RAQC1310
      LINSPP=PAGLN                                                      RAQC1320
      LNPHPG=LINSPP/2                                                   RAQC1330
      IF(LNPHPG.LT.40) LNPHPG=LINSPP                                    RAQC1340
C                                                                       RAQC1350
C********* LOOP ON FREQUENCY, AZIMUTH ANGLE, AND ELEVATION ANGLE        RAQC1360
      NFREQ=1                                                           RAQC1370
      IF (FSTEP.NE.0.) NFREQ=(FEND-FBEG)/FSTEP+1.5                      RAQC1380
      NAZ=1                                                             RAQC1390
      IF (AZSTEP.NE.0.) NAZ=(AZEND-AZBEG)/AZSTEP+1.5                    RAQC1400
      NBETA=1                                                           RAQC1410
      IF (ELSTEP.NE.0.) NBETA=(ELEND-ELBEG)/ELSTEP+1.5                  RAQC1420
      DO 50 NF=1,NFREQ                                                  RAQC1430
      OW=FBEG+(NF-1)*FSTEP                                              RAQC1440
      DO 45 J=1,NAZ                                                     RAQC1450
      AZ1=AZBEG+(J-1)*AZSTEP                                            RAQC1460
      AZA=AZ1*DEGS                                                      RAQC1470
      GAMMA1=PI-AZ1+ALPHA                                               RAQC1480
      SGAMMA=SIN (GAMMA1)                                               RAQC1490
      CGAMMA=SIN (PID2-GAMMA1)                                          RAQC1500
      DO 40 I=1,NBETA                                                   RAQC1510
      BETA=ELBEG+(I-1)*ELSTEP                                           RAQC1520
      EL=BETA*DEGS                                                      RAQC1530
      CBETA=SIN (PID2-BETA)                                             RAQC1540
      R=EARTHR+XMTRH                                                    RAQC1550
      TH=TH0                                                            RAQC1560
      PH=PH0                                                            RAQC1570
      KR=SIN (BETA)                                                     RAQC1580
      KTH=CBETA*CGAMMA                                                  RAQC1590
      KPH=CBETA*SGAMMA                                                  RAQC1600
      TPULSE=0.                                                         RAQC1610
      RSTART=1.                                                         RAQC1620
C********* THE FOLLOWING LINE NEEDED FOR RAY TRACING IN COMPLEX SPACE   RAQC1630
      SGN=1.0                                                           RAQC1640
C********* CALL MODELS                                                  RAQC1650
      CALL DISPER                                                       RAQC1660
C                                                                       RAQC1670
      LINPG=LINES-LHDRPG                                                RAQC1680
      IF(I.EQ.1  .OR.  LINPG.GE.LINSPP-20) THEN                         RAQC1690
C                                                                       RAQC1700
C     PUT OUT SUBHEADERS FROM MEDIA AND PRINTR ROUTINES                 RAQC1710
         CALL HEADER2                                                   RAQC1720
         CALL PRNHD1(' ')                                               RAQC1730
C                                                                       RAQC1740
```

```
C       COMPUTE LINE COUNT OF THIS HEADER                                RAQC1750
           LHDRPG=LINES/LINSPP*LINSPP                                    RAQC1760
        ELSEIF(LINPG.GE.LNPHPG-10 .AND. LINPG.LE.LNPHPG) THEN            RAQC1770
C       PUT OUT PAGE FEED WITH SUBHEADER IF AT HALF PAGE                 RAQC1780
           CALL PRNHD2('1')                                              RAQC1790
        ELSE                                                             RAQC1800
C       PUT OUT SUBHEADER                                                RAQC1810
           CALL PRNHD2(' ')                                              RAQC1820
        ENDIF                                                            RAQC1830
C                                                                        RAQC1840
        IF (KAY2.GT.0.) GO TO 30                                         RAQC1850
        WRITE(3,2900) OMEGMIN,OMEGMAX                                    RAQC1860
 2900 FORMAT (58H0TRANSMITTER IN EVANESCENT REGION, TRANSMISSION IMPOSSIRAQC1870
     1BLE/20H0MINIMUM FREQUENCY =,E17.10,20H MAXIMUM FREQUENCY =,E17.10)RAQC1880
        GO TO 44                                                         RAQC1890
30      CALL RENORM(KR,KAY2,3)                                           RAQC1900
        CALL CLEAR(RKVARS,NEQS-6)                                        RAQC1910
        CALL TOPOG                                                       RAQC1920
        IF(NERG.GT.0) RKVARS(NERG)=G                                     RAQC1930
        IF(NERR.GT.0) RKVARS(NERR)=PGR                                   RAQC1940
        IF(NERT.GT.0) RKVARS(NERT)=PGTH                                  RAQC1950
        IF(NERP.GT.0) RKVARS(NERP)=PGPH                                  RAQC1960
C                                                                        RAQC1970
C       CALCULATE ONE RAY PATH                                           RAQC1980
        CALL TRACE                                                       RAQC1990
        OSEC=SEC                                                         RAQC2000
        CALL SYSSEC(SEC)                                                 RAQC2010
        DIFF=SEC-OSEC                                                    RAQC2020
C                                                                        RAQC2030
C       ADD TO LINES COUNT FOR ELAPSED TIME REPORT                       RAQC2040
        LINES=LINES+2                                                    RAQC2050
        WRITE(3,3500) DIFF                                               RAQC2060
3500    FORMAT (/T93,'THIS RAY CALCULATION TOOK ',F8.3,' SEC')           RAQC2070
C                                                                        RAQC2080
        IF (PENET.AND.ONLY.NE.0..AND.IHOP.EQ.1) GO TO 44                 RAQC2090
   40 CONTINUE                                                           RAQC2100
44      IF(PLT.GT.0.AND.(NAZ.LE.1.OR.NBETA.GT.1)) CALL ENDPLT            RAQC2110
   45 CONTINUE                                                           RAQC2120
        IF(PLT.GT.0.AND.NAZ.GT.1.AND.NBETA.LE.1)CALL ENDPLT              RAQC2130
        IF(PENET.AND.ONLY.NE.0..AND.IHOP.EQ.1.AND.NAZ.EQ.1.AND.NBETA.EQ.1)RAQC2140
     1 GO TO 55                                                          RAQC2150
   50 CONTINUE                                                           RAQC2160
   55   IF (RAYSET.NE.0.) WRITE(9,5000)                                  RAQC2170
 5000 FORMAT (78X,1H-)                                                   RAQC2180
        GO TO 10                                                         RAQC2190
 6000   PRINT *,'DINP EMPTY OR NOT FOUND'                                RAQC2200
        STOP                                                             RAQC2210
        END                                                              RAQC2220



        SUBROUTINE DFSYS                                                 DFMS0020
        CHARACTER DAT*(*),DATX*10                                        DFMS0030
```

```
      CHARACTER TIM*(*),TIMX*10                                          DFMS0040
                                                                         DFMS0050
C                                                                        
C     ALL ROUTINES WHICH RELY ON SYSTEM BASED FUNCTIONS, SUCH AS TIME    DFMS0060
C     OF DAY OR MAY BENEFIT FROM SYSTEM AVAILABLE ROUTINES FOR BETTER    DFMS0070
C     PERFORMANCE, SUCH AS 'RMOVE' HAVE BEEN COLLECTED IN THIS ROUTINE.  DFMS0080
C                                                                        DFMS0090
      ENTRY RMOVE(X,Y,N)                                                 DFMS0100
C     MOVE 'N' COMPONENTS OF ARRAY 'Y' TO ARRAY 'X'.                     DFMS0110
C     THIS LOGIC WILL HANDLE OVERLAP CASES ONLY FOR X(M)=X(M+N1)         DFMS0120
C     WITH N1>0, M=1,N.                                                  DFMS0130
      REAL X(N),Y(N)                                                     DFMS0140
C                                                                        DFMS0150
C      DO 10 I=1,N                                                       DFMS0160
C10    X(I)=Y(I)                                                         DFMS0170
C     RETURN                                                             DFMS0180
C                                                                        DFMS0190
C     ENTRY IMOVE(IX,IY,N)                                               DFMS0200
      ENTRY IMOVE(X,Y,N)                                                 DFMS0210
C     SPECIAL ENTRY FOR INTEGER ARRAYS FOR MACHINES HAVING DIFFERENT     DFMS0220
C     WORD SIZES FOR INTEGERS THAN REALS.                                DFMS0230
C     INTEGER IX(N),IY(N)                                                DFMS0240
C                                                                        DFMS0250
      IF(N.LE.0) RETURN                                                  DFMS0260
C     USE CYBER BUILT-IN ROUTINE                                         DFMS0270
      CALL MOVLEV(Y,X,N)                                                 DFMS0280
C                                                                        DFMS0290
C     THE EQUIVALENT FORTRAN CODING FOR THIS ROUTINE ARE THE FOLLOWING   DFMS0300
C     TWO LINES.                                                         DFMS0310
C      DO 20 I=1,N                                                       DFMS0320
C20    IX(I)=IY(I)                                                       DFMS0330
C                                                                        DFMS0340
      RETURN                                                             DFMS0350
      ENTRY SYSSEC(SECS)                                                 DFMS0360
C                                                                        DFMS0370
C     OBTAIN CURENT TIME OF DAY IN SECONDS FROM SYSTEM ROUTINE           DFMS0380
C      SECS=0                                                            DFMS0390
C     SYSTEM CALL NEEDED FOR THE CYBER NOS OP SYS                        DFMS0400
      CALL SECOND(SECS)                                                  DFMS0410
C     SYSTEM CALL NEEDED FOR CRAY COMPUTER UNDER CTSS                    DFMS0420
C      CALL TIMEUSED(ICPU,IO,ISYS,MEM)                                   DFMS0430
C      SECS=ICPU/1.E6                                                    DFMS0440
      RETURN                                                             DFMS0450
C                                                                        DFMS0460
      ENTRY SYSTIM(TIM)                                                  DFMS0470
C                                                                        DFMS0480
C     OBTAIN CURENT TIME OF DAY AS A CHARACTER STRING FROM SYSTEM ROUTI  DFMS0490
C      TIM='TIME'                                                        DFMS0500
C     SYSTEM CALL NEEDED FOR THE CYBER NOS OP SYS                        DFMS0510
      CALL TIME(TIM)                                                     DFMS0520
C     SYSTEM CALL NEEDED FOR THE CRAY CTSS OP SYS                        DFMS0530
C     CALL TIMEDATE(TIM,DATX,MACH)                                       DFMS0540
      RETURN                                                             DFMS0550
C                                                                        DFMS0560
      ENTRY SYSDAT(DAT)                                                  DFMS0570
C                                                                        DFMS0580
```

```
C     OBTAIN CURENT DATE AS A CHARACTER STRING FROM SYSTEM ROUTINE      DFMS0590
C      DAT='DATE'                                                       DFMS0600
C     SYSTEM CALL NEEDED FOR THE CYBER NOS OP SYS                       DFMS0610
      CALL DATE(DAT)                                                   DFMS0620
C     SYSTEM CALL NEEDED FOR THE CRAY CTSS OP SYS                       DFMS0630
C     CALL TIMEDATE(TIMX,DAT,MACH)                                     DFMS0640
      RETURN                                                           DFMS0650
C     STANDARD SYSTEM INITIALIZATION TO BE CALLED FROM MAIN PROGRAM     DFMS0660
      ENTRY SYSINI                                                     DFMS0670
C     SYSTEM INITIALIZATION CALL FOR CTSS ON CRAY COMPUTER              DFMS0680
C     CALL LINK("READ5,UNIT5=TTY,PRINT6,UNIT6=TTY//")                  DFMS0690
      RETURN                                                           DFMS0700
C                                                                      DFMS0710
      ENTRY MORTEM                                                     DFMS0720
      STOP 'POST MORTEM'                                               DFMS0730
      END                                                              DFMS0740
```

```
      SUBROUTINE DFCNST(DRMACH,MAX)                                    DFRT0020
      REAL DRMACH(MAX)                                                 DFRT0030
C     RETURN MAX MACHINE CONSTANTS(OR THE NUMBER AVAILABLE)            DFRT0040
C     IN ARRAY DRMACH.                                                 DFRT0050
                                                                       DFRT0060
C     THIS MODULE PROVIDES CONSTANTS WHICH ARE EXPLICITLY DEPENDENT    DFRT0070
C     ON MACHINE ARCHITECTURE AND SHOULD BE EXAMINED WHEN MAKING ANY   DFRT0080
C     MIGRATION OF THE RAY TRACING ROUTINES.                           DFRT0090
C                                                                      DFRT0100
C                                                                      DFRT0110
      REAL RMACH(5)                                                    DFRT0120
C                                                                      DFRT0130
C     SINGLE PRECISION MACHINE CONSTANTS                               DFRT0140
C      (SEE DIGITAL SIGNAL PROCESSING, IEEE PRESS 1979 P S-7)          DFRT0150
C                                                                      DFRT0160
C     THE CONSTANTS ARE DERIVED FROM THE FOLLOWING                     DFRT0170
C     FORM FOR FLOATING POINT NUMBERS:                                 DFRT0180
C                                                                      DFRT0190
C        SIGN (B**E)*( (X(1)/B) + ... + (X(T)/B**T) )                  DFRT0200
C     FOR MOST MACHINES THE BASE B=2.                                  DFRT0210
C     FOR THE CYBER 60-BIT MACHINES                                    DFRT0220
C        T=48, EMIN=-974 AND EMAX=1070.                                DFRT0230
C     FOR THE PDP-11 AND VAX-11 MACHINES                               DFRT0240
C        T=24, EMIN=-127 AND EMAX=127                                  DFRT0250
C                                                                      DFRT0260
C     RMACH(1)  SMALLEST POSITIVE MAGNITUDE = B**(EMIN-1)              DFRT0270
C     RMACH(2)  LARGEST MAGNITUDE           = B**EMAX*(1-B**(-T))      DFRT0280
C     RMACH(3)  SMALLEST RELATIVE SPACING   = B**(-T)                  DFRT0290
C     RMACH(4)  LARGEST RELATIVE SPACING    = B**(1-T)                 DFRT0300
C     RMACH(5)                              = LOG10(B=2)               DFRT0310
C                                                                      DFRT0320
C     CONSTANTS HERE ARE FOR CDC 6000/7000/CYBER SERIES COMPUTERS      DFRT0330
C     PLEASE SUBSTITUTE FOR OTHER MACHINES                             DFRT0340
C                                                                      DFRT0350
```

```
C                                                                       DFRT0360
C       INTEGER SMALL(2)                                                DFRT0370
C       INTEGER LARGE(2)                                                DFRT0380
C       INTEGER RIGHT(2)                                                DFRT0390
C       INTEGER DIVER(2)                                                DFRT0400
C       INTEGER LOG10(2)                                                DFRT0410
CC                                                                      DFRT0420
C       EQUIVALENCE (RMACH(1),SMALL(1))                                 DFRT0430
C       EQUIVALENCE (RMACH(2),LARGE(1))                                 DFRT0440
C       EQUIVALENCE (RMACH(3),RIGHT(1))                                 DFRT0450
C       EQUIVALENCE (RMACH(4),DIVER(1))                                 DFRT0460
C       EQUIVALENCE (RMACH(5),LOG10(1))                                 DFRT0470
C                                                                       DFRT0480
C     MACHINE CONSTANTS FOR CYBER 170/180 COMPUTERS                     DFRT0490
C                                                                       DFRT0500
      DATA RMACH(1) / O"0001400000000000000" /                          DFRT0510
      DATA RMACH(2) / O"3776777777777777777" /                          DFRT0520
      DATA RMACH(3) / O"1604000000000000000" /                          DFRT0530
      DATA RMACH(4) / O"1614000000000000000" /                          DFRT0540
      DATA RMACH(5) / O"1716464202324117572O" /                         DFRT0550
C                                                                       DFRT0560
C***REFERENCES  FOX, P.A., HALL, A.D., SCHRYER, N.L, *FRAMEWORK FOR     DFRT0570
C                 A PORTABLE LIBRARY*, ACM TRANSACTIONS ON MATHE-       DFRT0580
C                 MATICAL SOFTWARE, VOL. 4, NO. 2, JUNE 1978,           DFRT0590
C                 PP. 177-188.                                          DFRT0600
C***ROUTINES CALLED  XERROR                                             DFRT0610
C***END PROLOGUE  R1MACH                                                DFRT0620
C     MACHINE CONSTANTS FOR VAX 11/780                                  DFRT0630
C     (EXPRESSED IN INTEGER AND HEXADECIMAL)                            DFRT0640
C    ***THE HEX FORMAT BELOW MAY NOT BE SUITABLE FOR UNIX SYSYEMS***    DFRT0650
C    *** THE INTEGER FORMAT SHOULD BE OK FOR UNIX SYSTEMS***            DFRT0660
C                                                                       DFRT0670
C     DATA SMALL(1), SMALL(2) /        128,           0 /              DFRT0680
C     DATA LARGE(1), LARGE(2) /     -32769,          -1 /              DFRT0690
C     DATA RIGHT(1), RIGHT(2) /       9344,           0 /              DFRT0700
C     DATA DIVER(1), DIVER(2) /       9472,           0 /              DFRT0710
C     DATA LOG10(1), LOG10(2) /  546979738,  -805796613 /              DFRT0720
C                                                                       DFRT0730
C     DATA SMALL(1), SMALL(2) / Z00000080, Z00000000 /                  DFRT0740
C     DATA LARGE(1), LARGE(2) / ZFFFF7FFF, ZFFFFFFFF /                  DFRT0750
C     DATA RIGHT(1), RIGHT(2) / Z00002480, Z00000000 /                  DFRT0760
C     DATA DIVER(1), DIVER(2) / Z00002500, Z00000000 /                  DFRT0770
C     DATA LOG10(1), LOG10(2) / Z209A3F9A, ZCFF884FB /                  DFRT0780
C                                                                       DFRT0790
C   MACHINE CONSTANTS FOR VAX 11/780 (G-FLOATING)                       DFRT0800
C     (EXPRESSED IN INTEGER AND HEXADECIMAL)                            DFRT0810
C    ***THE HEX FORMAT BELOW MAY NOT BE SUITABLE FOR UNIX SYSYEMS***    DFRT0820
C    *** THE INTEGER FORMAT SHOULD BE OK FOR UNIX SYSTEMS***            DFRT0830
C                                                                       DFRT0840
C     DATA SMALL(1), SMALL(2) /          16,           0 /              DFRT0850
C     DATA LARGE(1), LARGE(2) /     -32769,          -1 /              DFRT0860
C     DATA RIGHT(1), RIGHT(2) /       15552,           0 /             DFRT0870
C     DATA DIVER(1), DIVER(2) /       15568,           0 /             DFRT0880
C     DATA LOG10(1), LOG10(2) /  1142112243, 2046775455 /              DFRT0890
C                                                                       DFRT0900
```

```
C     DATA SMALL(1), SMALL(2) / Z00000010, Z00000000 /                  DFRT0910
C     DATA LARGE(1), LARGE(2) / ZFFFF7FFF, ZFFFFFFFF /                  DFRT0920
C     DATA RIGHT(1), RIGHT(2) / Z00003CC0, Z00000000 /                  DFRT0930
C     DATA DIVER(1), DIVER(2) / Z00003CD0, Z00000000 /                  DFRT0940
C     DATA LOG10(1), LOG10(2) / Z44133FF3, Z79FF509F /                  DFRT0950
C                                                                       DFRT0960
C     MACHINE CONSTANTS FOR THE CRAY 1                                  DFRT0970
C                                                                       DFRT0980
C     DATA SMALL(1) / 201354000000000000000B /                          DFRT0990
C     DATA SMALL(2) / 000000000000000000000B /                          DFRT1000
C                                                                       DFRT1010
C     DATA LARGE(1) / 577767777777777777777B /                          DFRT1020
C     DATA LARGE(2) / 000007777777777777774B /                          DFRT1030
C                                                                       DFRT1040
C     DATA RIGHT(1) / 376434000000000000000B /                          DFRT1050
C     DATA RIGHT(2) / 000000000000000000000B /                          DFRT1060
C                                                                       DFRT1070
C     DATA DIVER(1) / 376444000000000000000B /                          DFRT1080
C     DATA DIVER(2) / 000000000000000000000B /                          DFRT1090
C                                                                       DFRT1100
C     DATA LOG10(1) / 377774642023241175717B /                          DFRT1110
C     DATA LOG10(2) / 000007571421742254654B /                          DFRT1120
C                                                                       DFRT1130
      CALL RMOVE(DRMACH,RMACH,MIN0(MAX,5))                              DFRT1140
C                                                                       DFRT1150
      END                                                               DFRT1160
```

```
      SUBROUTINE READW1(AB,NWOK,MSET,MX,NTBL,ITBL,FRMTBL,GP)            REZ10020
C     HANDLES INPUT OF TABULAR DATA REQUIRED BY SOME MODELS.            REZ10030
C     COMMON DECK "RAYDEV" INSERTED HERE                                CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                              CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN               CRAY0050
C     COMMON DECK "CUCON" INSERTED HERE                                 CUCO0020
      COMMON/UCONV/CNVV(5,4)                                            CUCO0040
      CHARACTER PCV*3,CNVC*2                                            CUCO0050
      COMMON/UCONC/PCV(5),CNVC(5,4)                                     CUCO0060
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "WWR" INSERTED HERE                                   CWWR0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
```

```
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                        CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),          CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))           CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),           CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                 CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                      CWW20160
      LOGICAL NWOK,AB                                                      REZ10080
C                                                                          REZ10090
      CHARACTER LINE*80,ALPHA*3,FBUF*3                                     REZ10100
      PARAMETER (NVAR=6,ALPHA='  A',NACDE=-1,NFIELD=13)                    REZ10110
      INTEGER IPV(NVAR),NTBL(10),ITBL(10),FRMTBL(10)                       REZ10120
      CHARACTER*8 PU(NVAR)                                                 REZ10130
      REAL GP(*),V(NVAR),CONV(NVAR)                                        REZ10140
C                                                                          REZ10150
C     THE DATA STRUCTURE EXPECTED BY THIS PROGRAM FOR A DATA BLOCK IS      REZ10160
C     SPECIFIED FOR A GIVEN MODEL BY THE THREE ARRAYS NTBL,ITBL,FRMTBL.    REZ10170
C     THE TYPE OF DATA EXPECTED FOR A GIVEN FORMAT GROUP IS TAKEN FROM     REZ10180
C     ARRAY FRMTBL. IT MUST HAVE AN ENTRY FOR A GIVEN TYPE OF FORMAT       REZ10190
C     WHICH IS EITHER ZERO, IF THAT FORMAT IS NOT ALLOWED OR AN ENTRY      REZ10200
C     WHOSE VALUE IS EQUAL TO THAT FORMAT TYPE. CURRENTLY ONLY TWO         REZ10210
C     FORMAT TYPES ARE ALLOWED, 1 FOR ALPHA AND 2 FOR NUMERIC.             REZ10220
C     TO ALLOW MORE FLEXIBLITY IN THE INPUT FILE THE NUMERIC FORMAT        REZ10230
C     TYPE IN TURN ALLOWS FOR 3 DIFFERENT DATA FORMATS, SEE BELOW.         REZ10240
C                                                                          REZ10250
C     OFFSETS INTO THE GENERAL ARRAY 'GP' FOR EACH FORMAT ARE GIVEN        REZ10260
C     IN THE ARRAY 'NTBL'. THE SPACING BETWEEN DATA VALUES IN EACH LINE    REZ10270
C     READ IS GIVEN IN ARRAY 'ITBL'. THIS SCHEME DOES NOT ALLOW FOR X,Y    REZ10280
C     ,Z CYCLES BUT RATHER SEPARATE ARRAYS OF X VALUES, THEN Y VALUES,     REZ10290
C     AND THEN Z VALUES(ETC. UP TO 'NVAR' VARIABLES).                      REZ10300
C                                                                          REZ10310
      NWOK=.TRUE.                                                          REZ10320
C                                                                          REZ10330
C     INITIALIZE ANY ALPHANUMERIC ARRAYS TO BLANKS                         REZ10340
      IF(FRMTBL(1).EQ.1) CALL SET2(GP(NTBL(1)),1H ,NTBL(2)-NTBL(1))        REZ10350
C                                                                          REZ10360
C     BEGIN MULTI-FORMAT LOOP                                              REZ10370
5     READ(NRYIND,'(A,A)',END=200) FBUF,LINE                               REZ10380
      IF(NDEVTMP.GT.0) WRITE(NDEVTMP,'(A,A)') FBUF,LINE                    REZ10390
      IF(FBUF.EQ.ALPHA) THEN                                               REZ10400
         NFRM=NACDE                                                        REZ10410
      ELSE                                                                 REZ10420
         READ(FBUF,10) NFRM                                                REZ10430
      ENDIF                                                                REZ10440
10    FORMAT(BZ,I3,A)                                                      REZ10450
      IF(NFRM .EQ. 0) RETURN                                               REZ10460
C                                                                          REZ10470
C     FORMAT #1 IS ASSIGNED TO ALPHANUMERIC INPUT AND LARGER NUMBERS       REZ10480
C     FOR NUMERIC INPUT.                                                   REZ10490
C                                                                          REZ10500
C     CHECK NOW FOR NUMERICAL FORMATS.                                     REZ10510
C     THESE ARE PROVIDED·FOR EASE OF INPUT OF TABULATED DATA               REZ10520
C     AND ARE PARTLY 'TRANSPARENT' TO THE MODEL INVOLVED. I.E.             REZ10530
C     THE USER HAS THE OPTION OF SPECIFING THE NUMBER OF DATA COLUMNS.     REZ10540
C     INPUT FORMAT NUMBER SPECIFIES THE NUMBER OF COLUMNS OF INPUT DATA    REZ10550
C     I.E. FOR VALUE 1 A SINGLE DATA COLUMN IS EXPECTED, FOR A VALUE       REZ10560
```

```
C     2 TWO DATA COLUMNS ARE EXPECTED, ETC.                                REZ10570
C                                                                          REZ10580
C     DETERMINE FORMAT NUMBER                                              REZ10590
      IF(NFRM.EQ.NACDE) THEN                                               REZ10600
         IT=1                                                              REZ10610
C     ALPHANUMERIC FORMATS MUST BE FIRST IN LIST                           REZ10620
C     SINCE NO OFFSET SPECIFIER IS ALLOWED IN A FULL ALPHA LINE            REZ10630
         IF(FRMTBL(1).EQ.1) GO TO 100                                      REZ10640
      ELSE                                                                 REZ10650
         READ(LINE,'(I2)') IT                                              REZ10660
         IT=MAX0(IT,2)                                                     REZ10670
         IF(IT.LE.MX) GO TO 100                                            REZ10680
      ENDIF                                                                REZ10690
C                                                                          REZ10700
15    AB=.TRUE.                                                            REZ10710
      PRINT 20, MSET,NFRM,MX                                               REZ10720
20    FORMAT(' FOR SET',I5,' FORMAT NUMBER',I5,' EXCEEDS LIMIT OF',I5)     REZ10730
C      ERROR IN INPUT DATA , STOP HERE                                     REZ10740
      STOP                                                                 REZ10750
C                                                                          REZ10760
C     THE 'NTBL' TABLE PROVIDES A LIST OF OFFSETS TO VARIABLE GROUPS IN    REZ10770
C     MODEL ARRAYS, 'ITBL' GIVES ELEMENT SEPARATION TO ALLOW FOR 2-        REZ10780
C     DIMENSIONAL ARRAYS AND 'FRMTBL' SPECIFIES THE INPUT FORMAT.          REZ10790
C                                                                          REZ10800
100   N1=NTBL(IT)                                                          REZ10810
      N3=ITBL(IT)                                                          REZ10820
      N2=NTBL(IT+1)-N3                                                     REZ10830
C                                                                          REZ10840
      IF(NFRM.NE.NACDE) GO TO 110                                          REZ10850
C                                                                          REZ10860
      READ(LINE,1010)((GP(I+J-1),I=N1,N2,N3),J=1,N3)                       REZ10870
1010  FORMAT(10A8)                                                         REZ10880
      GO TO 5                                                              REZ10890
C                                                                          REZ10900
110   IF(NFRM.LT.1 .OR. NFRM.GT.NVAR) GO TO 200                            REZ10910
C     HANDLE NUMERIC FORMATS                                               REZ10920
C                                                                          REZ10930
C     N1 IS STARTING ELEMENT WHICH WILL BE AFTER THE DATA COUNT            REZ10940
C     VALUE(NEED NELS*N3+1 TOTAL ELEMENTS)                                 REZ10950
      N1=N1+1                                                              REZ10960
      N=0                                                                  REZ10970
C                                                                          REZ10980
C     # ELEMENTS IS EQUAL TO FORMAT NUMBER                                 REZ10990
      NELS=NFRM                                                            REZ11000
C     READ UNITS CONVERSION SPECIFICATION LINE                             REZ11010
      READ(LINE,1018) (PU(I),I=1,3)                                        REZ11020
      IF(PU(2).NE.' ') READ(LINE,2024) DELIM                               REZ11030
      IF(PU(2).EQ.' ') READ(LINE,1020) DELIM,(PU(I),I=1,NELS)              REZ11040
C                                                                          REZ11050
C  ALLOW FOR BLANK CONVERSION LINE PRODUCED BY EIGENRAY ROUTINE            REZ11060
      IF(DELIM.EQ.0) READ(LINE,2024) DELIM                                 REZ11070
C                                                                          REZ11080
C     READ NEXT LINE AND EXTRACT UNIT CONVERSION SPECS IF PRESENT          REZ11090
C     IN FLOATING FORMAT                                                   REZ11100
      READ(NRYIND,'(A)') LINE                                              REZ11110
```

```
      CALL GTUNIT(LINE,PU,NFIELD,NELS, NCARY)                          REZ11120
C     CARRY COUNT IS ZERO IF LINE CONTAINED UNITS SPECIFIERS           REZ11130
      IF(NCARY.EQ.0.AND.NDEVTMP.GT.0) WRITE(NDEVTMP,'(A)') LINE        REZ11140
C                                                                      REZ11150
1018  FORMAT(10(A,5X))                                                 REZ11160
1020  FORMAT(BZ,3X,G10.3,8X,10A)                                       REZ11170
2024  FORMAT(T43,G10.3)                                                REZ11180
1025  FORMAT(BZ,10G13.6)                                               REZ11190
C                                                                      REZ11200
C     LOOK UP CONVERSION CONSTANTS                                     REZ11210
      DO 2030 I=1,NELS                                                 REZ11220
2030  CALL UCON(IPV(I),PU(I),CONV(I))                                  REZ11230
C                                                                      REZ11240
C     DELIMITOR TESTING REQUIRES THE INNER LOOP BE EXECUTED            REZ11250
C     AT LEAST ONCE                                                    REZ11260
      NX=MAX0(1,N3)                                                    REZ11270
      N2=MAX0(N2,N1)                                                   REZ11280
C     BEGIN SINGLE DATA FORMAT LOOP                                    REZ11290
      DO 150 J=0,10000                                                 REZ11300
         IF(NELS.GT.1) THEN                                            REZ11310
            IF(NCARY.EQ.0) READ(NRYIND,'(A)') LINE                     REZ11320
            NCARY=0                                                    REZ11330
            IF(NDEVTMP.GT.0) WRITE(NDEVTMP,'(A)') LINE                 REZ11340
            READ(LINE,1025) (V(I),I=1,NELS)                            REZ11350
         ENDIF                                                         REZ11360
         K=0                                                           REZ11370
         DO 145 I=N1+J,N2+J,NX                                         REZ11380
               K=K+1                                                   REZ11390
               IF(NELS.EQ.1) THEN                                      REZ11400
                  IF(NCARY.EQ.0) READ(NRYIND,'(A)',END=10000) LINE     REZ11410
                  NCARY=0                                              REZ11420
                  IF(NDEVTMP.GT.0) WRITE(NDEVTMP,'(A)') LINE           REZ11430
C     FOR SINGLE COLUMN CASE READ NOW                                  REZ11440
                  READ(LINE,1020) V(K)                                 REZ11450
               ENDIF                                                   REZ11460
               IF(V(K) .EQ. DELIM) GO TO 160                           REZ11470
C                                                                      REZ11480
C           APPLY ANY SCALING CONVERSIONS                              REZ11490
               IF(CONV(K).GE.0.0) THEN                                 REZ11500
                  V(K)=V(K)*CONV(K)                                    REZ11510
               ELSEIF(V(K).NE.0.0) THEN                                REZ11520
                  V(K)=-CONV(K)/V(K)                                   REZ11530
      ENDIF                                                            REZ11540
               IF(J.LT.N3) GP(I)=V(K)                                  REZ11550
145            N=N+1                                                   REZ11560
150   CONTINUE                                                         REZ11570
C                                                                      REZ11580
10000 IF(V(1).NE.DELIM) CALL RERROR('READW1','NO DELIM',FLOAT(MSET))   REZ11590
C                                                                      REZ11600
C     SAVE NUMBER OF VALUES AND ELEMENTS(AS FRACTIONAL PART)           REZ11610
160   IF(N1.GT.1) GP(N1-1)=N+NELS/10.                                  REZ11620
C                                                                      REZ11630
C     CONTINUE MULTI-FORMAT LOOP                                       REZ11640
      GO TO 5                                                          REZ11650
C                                                                      REZ11660
```

```
200   CALL RERROR('READW1','BAD FORMAT',FLOAT(MSET))                    REZ11670
C                                                                       REZ11680
      END                                                               REZ11690



      SUBROUTINE GTUNIT(LINE,PU,NFIELD,NVAR, LN)                        REZ11700
C     GTUNIT IS USED TO EXTRACT UNIT CONVERSION SPECIFIERS FROM TABULAR REZ11710
C     DATA INPUT.                                                       REZ11720
C     CONVERSION SPECIFICATIONS ARE TAKEN FROM 'LINE' INPUT STRING.     REZ11730
C     A FLOATING FORMAT FALLING WITHIN SUCCESSIVE FIELDS OF             REZ11740
C     LENGTH 'NFIELD' IS PROVIDED FOR.                                  REZ11750
C     OUTPUT IS TO UNIT SPECIFIER ARRAY 'PU' FOR A MAXIMUM OF 'NVAR'    REZ11760
C     FIELDS. FOR A SUCCESS/FAIL RETURN VALUE, 'LN' GIVES THE NUMBER    REZ11770
C     OF NON-BLANK CHARACTERS USEFUL FOR UNIT DECODING, ZERO IMPLIES    REZ11780
C     NO USEFUL UNIT INFORMATION WAS FOUND.                             REZ11790
C                                                                       REZ11800
      PARAMETER (MAXLN=130)                                             REZ11810
      CHARACTER*(*) LINE,PU(NVAR),TMP*(MAXLN),TMP1*(MAXLN)             REZ11820
      INTEGER STRIM                                                     REZ11830
C                                                                       REZ11840
      LN=STRIM(LINE)                                                    REZ11850
      IF(LN.LT.1) THEN                                                  REZ11860
         DO 5 I=1,NVAR                                                  REZ11870
5           PU(I)=' '                                                   REZ11880
         RETURN                                                         REZ11890
      ENDIF                                                             REZ11900
C                                                                       REZ11910
      TMP=' '                                                           REZ11920
      CALL SFILTR(LINE,TMP,' 0123456789+-.E')                           REZ11930
      IF(STRIM(TMP).LT.1) RETURN                                        REZ11940
C     SIGNIFY WE CONSUMED THIS LINE                                     REZ11950
      LN=0                                                              REZ11960
C                                                                       REZ11970
      TMP1=LINE                                                         REZ11980
C     PLACE INTER-SPECIFIER DELIMITORS                                  REZ11990
      DO 10 I=NFIELD,NFIELD*NVAR,NFIELD                                 REZ12000
         IF(TMP1(I:I).NE.' ') THEN                                      REZ12010
            IF(TMP1(I+1:I+1).NE.' ') THEN                               REZ12020
C     IF NO ROOM FOR A BLANK, ITS AN ERROR                              REZ12030
               CALL RERROR('GTUNIT','FORMAT ER1',FLOAT(I))              REZ12040
            ELSE                                                        REZ12050
               TMP1(I+1:I+1)='#'                                        REZ12060
            ENDIF                                                       REZ12070
         ELSE                                                           REZ12080
            TMP1(I:I)='#'                                               REZ12090
         ENDIF                                                          REZ12100
10    CONTINUE                                                          REZ12110
C                                                                       REZ12120
C     NOW SQUEEZE OUT THE BLANKS                                        REZ12130
      CALL SFILTR(TMP1,TMP,' ')                                         REZ12140
C     HOPEFULLY SOMETHING LEFT                                          REZ12150
C                                                                       REZ12160
```

```
      NL=STRIM(TMP)                                                      REZ12170
      ILIN=1                                                             REZ12180
      DO 20 I=1,NVAR                                                     REZ12190
C     CHECK FOR EMPTY CONVERSIONS                                        REZ12200
         IF(TMP(ILIN:ILIN).EQ.'#') THEN                                  REZ12210
            ILIN=ILIN+1                                                  REZ12220
            PU(I)=' '                                                    REZ12230
            GO TO 20                                                     REZ12240
         ENDIF                                                           REZ12250
C     GET NEXT UNIT TYPE                                                 REZ12260
         PU(I)(1:2)=TMP(ILIN:ILIN+1)                                     REZ12270
C     GET NEXT PHYSICAL UNIT(BLANKS HAVE BEEN REMOVED)                   REZ12280
C     SHORT PHRASE IF A BLANK WAS REMOVED                                REZ12290
         IF(TMP(ILIN+3:ILIN+3).NE.'#') THEN                              REZ12300
            IF(TMP(ILIN+4:ILIN+4).NE.'#') THEN                           REZ12310
C     IF INCORRECT GROUPING OF LETTERS, ITS AN ERROR                     REZ12320
               CALL RERROR('GTUNIT','FORMAT ER2',FLOAT(ILIN))            REZ12330
            ELSE                                                         REZ12340
C     ADD TWO LETTER UNIT                                                REZ12350
               PU(I)(3:5)=' '//TMP(ILIN+2:ILIN+3)                        REZ12360
               ILIN=ILIN+5                                               REZ12370
            ENDIF                                                        REZ12380
         ELSE                                                            REZ12390
C     ADD ONE LETTER UNIT                                                REZ12400
            PU(I)(3:5)='  '//TMP(ILIN+2:ILIN+2)                         REZ12410
            ILIN=ILIN+4                                                  REZ12420
         ENDIF                                                           REZ12430
20    CONTINUE                                                           REZ12440
C                                                                        REZ12450
C     SHOULD HAVE USED UP ALL THE TEXT BY NOW                            REZ12460
      IF(ILIN.LE.NL) CALL RERROR('GTUNIT','FORMAT ER3',FLOAT(ILIN))      REZ12470
      END                                                                REZ12480
```

```
      SUBROUTINE SREAD1(AB,NWOK,NW)                                      SRT10020
C     THIS IS THE FALLTHROUGH FEATURE FOR READW WHEN ENCOUNTERING        SRT10030
C     A NON STANDARD LABELED COMMON BLOCK. SREAD1 WILL ADD TABULAR       SRT10040
C     DATA TO THE DEFAULT GP ARRAY PGP AND INCLUDE INDEX OFFSET          SRT10050
C     VALUES TO A LOOKUP ARRAY 'LIST'.                                   SRT10060
C     COMMON DECK "CPROCFL" INSERTED HERE                                CPRO0020
      INTEGER PMX,PNTBL,PITBL,PFRMTBL,IDSP(10)                           CPRO0040
C     PARAMETER DECK "PGROUPS"                                           PGRO0020
      PARAMETER (NCHPG1=11,NWPV=250,NSPGP=NCHPG1+2*NWPV+1)               PGRO0030
      PARAMETER (MNGRP=9,MXGRP=69,MXLIST=MXGRP-MNGRP+2)                  PGRO0040
      COMMON/PROCFL/LIST(MXLIST)                                         CPRO0060
      COMMON/PROCFL/PMX,PNTBL(10),PITBL(10),PFRMTBL(10),PGP(NSPGP)       CPRO0070
      EQUIVALENCE (PGP,IDSP)                                             CPRO0080
C                                                                        SRT10080
      CHARACTER ITOC*7                                                   SRT10090
C                                                                        SRT10100
      INTEGER PPX    ,PQTBL(10),PLTBL(10),PIRMTBL(10)                    SRT10110
C                                                                        SRT10120
```

```
      DATA PPX/2/                                                        SRT10130
      DATA PQTBL/1,NCHPG1,NSPGP,7*0/                                     SRT10140
      DATA PLTBL/1,NWPV,8*0/                                             SRT10150
      DATA PIRMTBL/1,2,8*0/                                              SRT10160
C                                                                        SRT10170
      NWA=IABS(NW)-(MNGRP-2)                                             SRT10180
      IF(NWA.LT.2 .OR. NWA.GT.MXLIST) CALL STOPIT                        SRT10190
     1  ( 'SREAD1 NW='//ITOC(NW)//',LIST(1)='//ITOC(LIST(1)) )           SRT10200
C                                                                        SRT10210
      CALL READW1(AB,NWOK,NW,PMX,PNTBL,PITBL,PFRMTBL,PGP)                SRT10220
C                                                                        SRT10230
C     UPDATE THE LIST, LIST(1)=MAXIMUM USED ELEMENT                      SRT10240
      LIST(1)=MAX0(LIST(1),NWA)                                          SRT10250
      LIST(NWA)=PNTBL(1)                                                 SRT10260
C                                                                        SRT10270
C     GET NEXT NUMERICAL INDEX FROM THE POINTER ARRAY                    SRT10280
      NLN=PNTBL(2)                                                       SRT10290
C                                                                        SRT10300
C     NUMBER OF VALUES IS INTEGRAL PART OF FIRST ELEMENT                 SRT10310
      NPTS=PGP(NLN)                                                      SRT10320
C     NUMBER OF ROWS IS KEPT AS FRACTIONAL PART                          SRT10330
      NELS=(PGP(NLN)-NPTS)*10.+.5                                        SRT10340
      N2=NPTS/NELS                                                       SRT10350
      NMX=PITBL(2)                                                       SRT10360
C     MAKE ADJUSTMENTS TO LENGTHS AND OFFSETS                            SRT10370
      PITBL(2)=NMX-N2-(NCHPG1-1)                                         SRT10380
      NUP=NPTS+NCHPG1                                                    SRT10390
      PNTBL(1)=PNTBL(1)+NUP                                              SRT10400
      PNTBL(2)=PNTBL(2)+NUP                                              SRT10410
      NLN=NLN+1                                                          SRT10420
C                                                                        SRT10430
C     LOOP TO MOVE NELS-1 ROWS OF PGP(NMX,NELS) TO PGP(N2,NELS)          SRT10440
C     THUS MAKING PGP A CONTIGUOUS ARRAY                                 SRT10450
      NTO=NLN+N2                                                         SRT10460
      NFRM=NLN+NMX                                                       SRT10470
      DO 100 I=2,NELS                                                    SRT10480
      IF(NTO.GT.0 .AND. NFRM+N2-1.LE.NSPGP)                              SRT10490
     1         CALL RMOVE(PGP(NTO),PGP(NFRM),N2)                         SRT10500
      NTO=NTO+N2                                                         SRT10510
100   NFRM=NFRM+NMX                                                      SRT10520
      RETURN                                                             SRT10530
C                                                                        SRT10540
      ENTRY SETSRD                                                       SRT10550
      CALL ISET2(LIST,0,MXLIST)                                          SRT10560
      PMX=PPX                                                            SRT10570
      CALL IMOVE(PNTBL,PQTBL,10)                                         SRT10580
      CALL IMOVE(PITBL,PLTBL,10)                                         SRT10590
      CALL IMOVE(PFRMTBL,PIRMTBL,10)                                     SRT10600
      END                                                                SRT10610


      FUNCTION READW(XX)                                                 REXW0020
```

```
C     READS W ARRAY MAKING ANY NECESSARY CONVERSION OF UNITS.                REXW0030
C     VARIABLE LENGTH 'TABULAR' DATA IS READ BY SUBROUTINE 'READW1'          REXW0040
C     WHICH IS CALLED FOR NEGATIVE W-ARRAY INDICES.                          REXW0050
C                                                                            REXW0060
C     COMMON DECK "CONST" INSERTED HERE                                      CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                          CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                              CCON0050
C     COMMON DECK "WWR" INSERTED HERE                                        CWWR0020
      PARAMETER (NWARSZ=1000)                                                CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                        CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                     CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),         CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),           CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),               CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),              CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                       CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))     CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                           CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                           CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                          CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),            CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))             CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),             CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                   CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                        CWW20160
C     COMMON DECK "B1" INSERTED HERE                                         CB1 0020
      INTEGER UMX,UNTBL,UITBL,UFRMTBL,IDSU(10)                               CB1 0040
      COMMON/B1/UMX,UNTBL(10),UITBL(10),UFRMTBL(10),UGP(10)                  CB1 0050
      EQUIVALENCE (UGP,IDSU)                                                 CB1 0060
C     COMMON DECK "B2" INSERTED HERE                                         CB2 0020
      INTEGER DUMX,DUNTBL,DUITBL,DUFRMTB,IDSDU(10)                           CB2 0040
      COMMON/B2/DUMX,DUNTBL(10),DUITBL(10),DUFRMTB(10),DUGP(10)              CB2 0050
      EQUIVALENCE (DUGP,IDSDU)                                               CB2 0060
C     COMMON DECK "B3" INSERTED HERE                                         CB3 0020
      INTEGER CMX,CNTBL,CITBL,CFRMTBL,IDSC(10)                               CB3 0040
      COMMON/B3/CMX,CNTBL(10),CITBL(10),CFRMTBL(10),CGP(512)                 CB3 0050
      EQUIVALENCE (CGP,IDSC),(ANC,CGP(11))                                   CB3 0060
C     COMMON DECK "B4" INSERTED HERE                                         CB4 0020
      INTEGER DCMX,DCNTBL,DCITBL,DCFRMTB,IDSDC(10)                           CB4 0040
      COMMON/B4/DCMX,DCNTBL(10),DCITBL(10),DCFRMTB(10),DCGP(10)              CB4 0050
      EQUIVALENCE (DCGP,IDSDC)                                               CB4 0060
C     COMMON DECK "B5" INSERTED HERE                                         CB5 0020
      INTEGER TMX,TNTBL,TITBL,TFRMTBL,IDST(10)                               CB5 0040
      COMMON/B5/TMX,TNTBL(10),TITBL(10),TFRMTBL(10),TGP(262)                 CB5 0050
      EQUIVALENCE (TGP,IDST),(ANT,TGP(11))                                   CB5 0060
C     COMMON DECK "B6" INSERTED HERE                                         CB6 0020
      INTEGER DTMX,DTNTBL,DTITBL,DTFRMTB,IDSDT(10)                           CB6 0040
      COMMON/B6/DTMX,DTNTBL(10),DTITBL(10),DTFRMTB(10),DTGP(10)              CB6 0050
      EQUIVALENCE (DTGP,IDSDT)                                               CB6 0060
C     COMMON DECK "B7" INSERTED HERE                                         CB7 0020
      INTEGER MMX,MNTBL,MITBL,MFRMTBL,IDSM(10)                               CB7 0040
      REAL MGP                                                               CB7 0050
      COMMON/B7/MMX,MNTBL(10),MITBL(10),MFRMTBL(10),MGP(10)                  CB7 0060
      EQUIVALENCE (MGP,IDSM)                                                 CB7 0070
C     COMMON DECK "B9" INSERTED HERE                                         CB8 0020
```

```
      INTEGER GMX,GNTBL,GITBL,GFRMTBL,IDSG(10)                                  CB8 0040
      COMMON/B9/GMX,GNTBL(10),GITBL(10),GFRMTBL(10),GGP(113)                    CB8 0050
      EQUIVALENCE (GGP,IDSG),(ANG,GGP(11))                                      CB8 0060
C                                                                               REXW0170
C     COMMON DECK "B10" INSERTED HERE                                           CB9 0020
      INTEGER DGMX,DGNTBL,DGITBL,DGFRMTB,IDSDG(10)                              CB9 0040
      COMMON/B10/DGMX,DGNTBL(10),DGITBL(10),DGFRMTB(10),DGGP(10)                CB9 0050
      EQUIVALENCE (DGGP,IDSDG)                                                  CB9 0060
C                                                                               REXW0190
C     COMMON DECK "B8" INSERTED HERE                                            CB100020
      INTEGER RMX,RNTBL,RITBL,RFRMTBL,IDSR(10)                                  CB100040
      COMMON/B8/RMX,RNTBL(10),RITBL(10),RFRMTBL(10),RGP(10)                     CB100050
      EQUIVALENCE (RGP,IDSR)                                                    CB100060
C                                                                               REXW0210
C     COMMON DECK "CB11" INSERTED HERE                                          CB110020
      INTEGER SMX,SNTBL,SITBL,SFRMTBL,IDSS(10)                                  CB110040
      COMMON/B11/SMX,SNTBL(10),SITBL(10),SFRMTBL(10),SGP(11)                    CB110050
      EQUIVALENCE (SGP,IDSS),(ANS,SGP(11))                                      CB110060
C     COMMON DECK "CB12" INSERTED HERE                                          CB120020
      INTEGER DSMX,DSNTBL,DSITBL,DSFRMTB,IDSDS(10)                              CB120040
      COMMON/B12/DSMX,DSNTBL(10),DSITBL(10),DSFRMTB(10),DSGP(11)                CB120050
      EQUIVALENCE (DSGP,IDSDS),(ANDS,DSGP(11))                                  CB120060
C     COMMON DECK "CB17" INSERTED HERE                                          CB170020
      INTEGER VMX,VNTBL,VITBL,VFRMTBL,IDSV(10)                                  CB170040
      COMMON/B17/VMX,VNTBL(10),VITBL(10),VFRMTBL(10),VGP(53)                    CB170050
      EQUIVALENCE (VGP,IDSV),(ANV,VGP(11))                                      CB170060
C     COMMON DECK "CB18" INSERTED HERE                                          CB180020
      INTEGER DVMX,DVNTBL,DVITBL,DVFRMTB,IDSDV(10)                              CB180040
      COMMON/B18/DVMX,DVNTBL(10),DVITBL(10),DVFRMTB(10),DVGP(11)                CB180050
      EQUIVALENCE (DVGP,IDSDV),(ANDV,DVGP(11))                                  CB180060
C     COMMON DECK "CB19" INSERTED HERE                                          CB190020
      INTEGER PRMX,PRNTBL,PRITBL,PRFRMTB,IDSPR(10)                              CB190040
      COMMON/B19/PRMX,PRNTBL(10),PRITBL(10),PRFRMTB(10),PRGP(11)                CB190050
      EQUIVALENCE (PRGP,IDSPR),(ANP,PRGP(11))                                   CB190060
C     COMMON DECK "CB20" INSERTED HERE                                          CB200020
      INTEGER DPMX,DPNTBL,DPITBL,DPFRMTB,IDSDP(10)                              CB200040
      COMMON/B20/DPMX,DPNTBL(10),DPITBL(10),DPFRMTB(10),DPGP(11)                CB200050
      EQUIVALENCE (DPGP,IDSDP),(ANDP,DPGP(11))                                  CB200060
C                                                                               REXW0280
      PARAMETER (MXCMTS=83,BIGVAL=1.E30)                                        REXW0290
      CHARACTER*38 WFRMT,WNOTES(2)                                              REXW0300
      CHARACTER*80 LINEX,NUMSTG                                                 REXW0310
      CHARACTER*100 STMP1,STMP2                                                 REXW0320
      INTEGER STRIM,WCOMTS(MXCMTS)                                              REXW0330
      CHARACTER*1 DEG,KM,NM,FEET,CYCLE,PER,MSKMS                                REXW0340
C     COMMON DECK "RAYDEV" INSERTED HERE                                        CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                                      CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN                       CRAY0050
      LOGICAL NWOK,AB,UCON                                                      REXW0360
C                                                                               REXW0370
C     COMMON DECK "FLAG" INSERTED HERE                                          CFLA0020
      LOGICAL NEWWR,NEWWP,NEWTRC,PENET                                          CFLA0040
      COMMON /FLG/ NTYP,NEWWR,NEWWP,NEWTRC,PENET,LINES,IHOP,HPUNCH              CFLA0050
      COMMON/FLGP/NSET                                                          CFLA0060
C                                                                               REXW0390
```

```
      CHARACTER PC*1                                                    REXW0400
                                                                        REXW0410
C
      INTEGER XMX,XNTBL(10),XITBL(10),XFRMTBL(10)                       REXW0420
      REAL XGP(11)                                                      REXW0430
                                                                        REXW0440
C
      DATA XMX/2/                                                       REXW0450
      DATA XNTBL/1,11,12,7*0/                                           REXW0460
      DATA XITBL/1,9*0/                                                 REXW0470
      DATA XFRMTBL/1,2,8*0/                                             REXW0480
                                                                        REXW0490
C
      DATA WCOMTS,WNOTES/MXCMTS*1, ' ',' INPUT OVERRIDDEN'/             REXW0500
                                                                        REXW0510
C
      AB=.FALSE.                                                        REXW0520
      READW=0.0                                                         REXW0530
                                                                        REXW0540
C
      IF(NDEVTMP.GT.0) REWIND NDEVTMP                                   REXW0550
                                                                        REXW0560
C
      READ(NRYIND,1000,END=3) ID                                        REXW0570
 1000 FORMAT (BZ,10A8)                                                  REXW0580
      IF(NDEVTMP.GT.0) WRITE(NDEVTMP,1000) ID                           REXW0590
      GO TO 4                                                           REXW0600
                                                                        REXW0610
C
3     READW=1.0                                                         REXW0620
      IF(NFRMAT.LT.0) RETURN                                            REXW0630
                                                                        REXW0640
C
      PRINT 1040                                                        REXW0650
      WRITE(3,1040)                                                     REXW0660
1040  FORMAT(' END OF INPUT DATA')                                      REXW0670
      CALL ENDPLT                                                       REXW0680
                                                                        REXW0690
33    IF(PLT.NE.0.0) CALL DDEND                                         REXW0700
      STOP                                                              REXW0710
C                                                                       REXW0720
4     READ(NRYIND,'(A)',END=10001) STMP1                                REXW0730
C                                                                       REXW0740
      READ(STMP1,1100) NW,WWW,LINEX                                     REXW0750
1100  FORMAT (BZ,I3,E14.7,A)                                            REXW0760
      IF(NDEVTMP.GT.0.AND.(NFRMAT.NE.-2.OR.NW.LT.0))                    REXW0770
     1      WRITE(NDEVTMP,'(A)') STMP1                                  REXW0780
10001 IF (NW.EQ.0) GO TO 10                                             REXW0790
      IF(NW.GT.MAXW) GO TO 3400                                         REXW0800
C                                                                       REXW0810
      IF (NW.LE.MAXW .AND. NW.GT.0) GO TO 5                             REXW0820
C     TABULAR INPUT DATA                                                REXW0830
      NWOK=.FALSE.                                                      REXW0840
C 'OPEN' THE TEMP FILE                                                  REXW0850
1150  FORMAT(I3,T18,A)                                                  REXW0860
      NWP=-NW                                                           REXW0870
      IF(NFRMAT.EQ.-2) THEN                                             REXW0880
C     USE DUMMY ARRAYS TO ABSORB TABULAR DATA                           REXW0890
         CALL READW1(AB,NWOK,NW,XMX,XNTBL,XITBL,XFRMTBL,XGP)            REXW0900
      ELSEIF(NWP.EQ.1) THEN                                             REXW0910
         CALL READW1(AB,NWOK,NW,UMX,UNTBL,UITBL,UFRMTBL,UGP)            REXW0920
      ELSEIF(NWP.EQ.2) THEN                                             REXW0930
         CALL READW1(AB,NWOK,NW,DUMX,DUNTBL,DUITBL,DUFRMTB,DUGP)        REXW0940
      ELSEIF(NWP.EQ.3) THEN
```

```
   CALL READW1(AB,NWOK,NW,CMX,CNTBL,CITBL,CFRMTBL,CGP)                    REXW0950
ELSEIF(NWP.EQ.4) THEN                                                     REXW0960
   CALL READW1(AB,NWOK,NW,DCMX,DCNTBL,DCITBL,DCFRMTB,DCGP)                REXW0970
ELSEIF(NWP.EQ.5) THEN                                                     REXW0980
      CALL READW1(AB,NWOK,NW,TMX,TNTBL,TITBL,TFRMTBL,TGP)                 REXW0990
ELSEIF(NWP.EQ.6) THEN                                                     REXW1000
   CALL READW1(AB,NWOK,NW,DTMX,DTNTBL,DTITBL,DTFRMTB,DTGP)                REXW1010
ELSEIF(NWP.EQ.7) THEN                                                     REXW1020
   CALL READW1(AB,NWOK,NW,MMX,MNTBL,MITBL,MFRMTBL,MGP)                    REXW1030
ELSEIF(NWP.EQ.8) THEN                                                     REXW1040
   CALL READW1(AB,NWOK,NW,RMX,RNTBL,RITBL,RFRMTBL,RGP)                    REXW1050
ELSEIF(NWP.EQ.9) THEN                                                     REXW1060
   CALL READW1(AB,NWOK,NW,GMX,GNTBL,GITBL,GFRMTBL,GGP)                    REXW1070
ELSEIF(NWP.EQ.10) THEN                                                    REXW1080
   CALL READW1(AB,NWOK,NW,DGMX,DGNTBL,DGITBL,DGFRMTB,DGGP)                REXW1090
ELSEIF(NWP.EQ.11) THEN                                                    REXW1100
   CALL READW1(AB,NWOK,NW,SMX,SNTBL,SITBL,SFRMTBL,SGP)                    REXW1110
ELSEIF(NWP.EQ.12) THEN                                                    REXW1120
   CALL READW1(AB,NWOK,NW,DSMX,DSNTBL,DSITBL,DSFRMTB,DSGP)                REXW1130
ELSEIF(NWP.EQ.17) THEN                                                    REXW1140
   CALL READW1(AB,NWOK,NW,VMX,VNTBL,VITBL,VFRMTBL,VGP)                    REXW1150
ELSEIF(NWP.EQ.18) THEN                                                    REXW1160
   CALL READW1(AB,NWOK,NW,DVMX,DVNTBL,DVITBL,DVFRMTB,DVGP)                REXW1170
ELSEIF(NWP.EQ.19) THEN                                                    REXW1180
   CALL READW1(AB,NWOK,NW,PRMX,PRNTBL,PRITBL,PRFRMTB,PRGP)                REXW1190
ELSEIF(NWP.EQ.20) THEN                                                    REXW1200
   CALL READW1(AB,NWOK,NW,DPMX,DPNTBL,DPITBL,DPFRMTB,DPGP)                REXW1210
ELSE                                                                      REXW1220
      CALL SREAD1(AB,NWOK,NW)                                             REXW1230
   ENDIF                                                                  REXW1240
   IF(NWOK) GO TO 4                                                       REXW1250
C                                                                         REXW1260
3400  WRITE(3,4000) NW,MAXW                                               REXW1270
4000  FORMAT(15H1THE SUBSCRIPT ,I3,  ' ON THE W-ARRAY INPUT IS OUT OF BOREXW1280
     1UNDS.  ALLOWABLE VALUES ARE -8 THROUGH,',I4,'.' )                   REXW1290
      CALL EXIT                                                           REXW1300
C                                                                         REXW1310
5     READ(LINEX,70) DEG,KM,NM,FEET,CYCLE,PER,MSKMS                       REXW1320
70    FORMAT(7A)                                                          REXW1330
      W(NW)=WWW                                                           REXW1340
C                                                                         REXW1350
C     CHECK FOR A 'TERRAIN RELATIVE' HEIGHT SPECIFICATION.                REXW1360
C     IF SO ADD FLAG VALUE 10**40 TO BE TESTED FOR LATER IN 'RAYTRC'      REXW1370
C                                                                         REXW1380
      IF(MSKMS.EQ.'T' .AND. WWW.EQ.0.0) W(NW)=1.E-40                      REXW1390
      IF(WWW.EQ.0.0) GO TO 4                                              REXW1400
      IF(MSKMS.EQ.'T') WWW=WWW*1.E40                                      REXW1410
C                                                                         REXW1420
C     CHECK FOR KEYWORD UNITS SPECIFICATION, IF SO PERFORM CONVERSION     REXW1430
C                                                                         REXW1440
      IF(.NOT.UCON(KU,LINEX(:10),CONV)) GO TO 60                          REXW1450
75    IF(CONV.GE.0.0) THEN                                                REXW1460
         W(NW)=WWW*CONV                                                   REXW1470
      ELSEIF(W(NW).NE.0.0) THEN                                           REXW1480
         W(NW)=-CONV/W(NW)                                                REXW1490
```

```
      ENDIF                                                              REXW1500
      GO TO 4                                                            REXW1510
                                                                         REXW1520
C                                                                        REXW1530
60    IF (DEG.EQ.'1') WWW=WWW*RAD                                        REXW1540
      IF (KM.EQ.'1') WWW=WWW/EARTHR                                      REXW1550
      IF (NM.EQ.'1') WWW=WWW*1.852                                       REXW1560
      IF (FEET.EQ.'1') WWW=WWW*3.048006096E-4                            REXW1570
      IF(CYCLE.EQ.'1') WWW=WWW*PIT2                                      REXW1580
      IF(PER.EQ.'1') WWW=PIT2/WWW                                        REXW1590
      IF(MSKMS.EQ.'1') WWW=WWW*1.E-3                                     REXW1600
      W(NW)=WWW                                                          REXW1610
      GO TO 4                                                            REXW1620
C                                                                        REXW1630
10    IF(.NOT.AB) RETURN                                                 REXW1640
      PRINT 1200                                                         REXW1650
1200  FORMAT(' A DATA FORMATTING ERROR PREVENTS CONTINUED EXECUTION')     REXW1660
      CALL EXIT                                                          REXW1670
C                                                                        REXW1680
      ENTRY SETW(XX)                                                     REXW1690
C     THIS ENTRY IS CALLED ONCE BEFORE THE FIRST RUN SET IS READ         REXW1700
C                                                                        REXW1710
C INITIALIZE SOME MATHEMATICAL CONSTANTS                                 REXW1720
      PI=4.0*ATAN(1.0)                                                   REXW1730
      PIT2=2.*PI                                                         REXW1740
      PID2=PI/2.                                                         REXW1750
      DEGS=180./PI                                                       REXW1760
      RAD=PI/180.                                                        REXW1770
      ALN10=ALOG(10.)                                                    REXW1780
CC********* INITIALIZE SOME VARIABLES IN THE W ARRAY                     REXW1790
      MAXW=NWARSZ                                                        REXW1800
      CALL CLEAR(W,MAXW)                                                 REXW1810
      PLON=0.                                                            REXW1820
      PLAT=PID2                                                          REXW1830
      EARTHR=6370.                                                       REXW1840
      INTYP=3.                                                           REXW1850
      MAXERR=1.E-4                                                       REXW1860
      ERATIO=50.                                                         REXW1870
      STEP1=1.                                                           REXW1880
      STPMAX=100.                                                        REXW1890
      STPMIN=1.E-8                                                       REXW1900
      FACTR=0.5                                                          REXW1910
      HITLET=.15                                                         REXW1920
      MAXSTP=1000.0                                                      REXW1930
      HOP=1.                                                             REXW1940
      HMAX=500.0                                                         REXW1950
      EXTINC=999.999                                                     REXW1960
      PAGLN=66.0                                                         REXW1970
C                                                                        REXW1980
      CALL SETUCON                                                       REXW1990
C                                                                        REXW2000
      RETURN                                                             REXW2010
C                                                                        REXW2020
      ENTRY SETOVR(XX)                                                   REXW2030
C     THIS ENTRY IS CALLED AFTER EACH RUN SET IS READ                    REXW2040
C
```

```
C       PERFORM OVERRIDE ASSIGNMENTS TO W-ELEMENTS                          REXW2050
        CALL OVERRD(MAXSTP,0.0,1000.0,WCOMTS(23),2,1)                       REXW2060
        CALL OVERRD(ERATIO,0.0,50.0,WCOMTS(43),2,1)                         REXW2070
        CALL OVERRD(FACTR,0.0,0.5,WCOMTS(47),2,1)                           REXW2080
        CALL OVERRD(SKIP,0.0,BIGVAL,WCOMTS(71),2,1)                         REXW2090
        CALL OVERRD(HITLET,0.0,.15,WCOMTS(75),2,1)                          REXW2100
        IF(PAGLN.LT.30.0) PAGLN=0.0                                         REXW2110
        CALL OVERRD(PAGLN,0.0,66.0,WCOMTS(77),2,1)                          REXW2120
        CALL OVERRD(PFACTR,0.0,1.0,WCOMTS(82),2,1)                          REXW2130
        RETURN                                                              REXW2140
C                                                                           REXW2150
        ENTRY PRINTW(XX)                                                    REXW2160
C       THIS ENTRY IS CALLED AFTER EACH RUN SET IS READ                     REXW2170
C       TO PRINT VALUES OF THE W-ARRAY IN A FORMAT WHICH SHOWS              REXW2180
C       FULL ACCURACY AND THE EFFECTS OF ANY CONVERSIONS OR OVERRIDES.      REXW2190
C                                                                           REXW2200
C       GO TO NEXT PAGE, PUT OUT COPY OF INPUT DATA FOR THIS RUN SET        REXW2210
C       ASSUMING WE HAD A TEMPORARY FILE TO USE                             REXW2220
C                                                                           REXW2230
        IF(NDEVTMP.LE.0) GO TO 1065                                         REXW2240
        CALL NEWPAG(NPAG,INT(PAGLN),PC)                                     REXW2250
        LNSXPG=LINES+INT(PAGLN)                                             REXW2260
        CALL PUTHDR(3,PC,NPAG)                                              REXW2270
        CALL PUTDVR(3)                                                      REXW2280
        CALL PUTKBK(3,1)                                                    REXW2290
        CALL PUTKCT(3,'INPUT DATA FILE FOR RUN SET NUMBER'                  REXW2300
     1  //NUMSTG(NSET,1,'(I5)'))                                            REXW2310
        CALL PUTKBK(3,1)                                                    REXW2320
        CALL PUTDVR(3)                                                      REXW2330
        CALL PUTKBK(3,1)                                                    REXW2340
        REWIND NDEVTMP                                                      REXW2350
C                                                                           REXW2360
1060    READ(NDEVTMP,'(A)',END=1065) LINEX                                  REXW2370
        IF(LINES.GE.LNSXPG) THEN                                            REXW2380
           CALL NEWPAG(NPAG,INT(PAGLN),PC)                                  REXW2390
           LNSXPG=LINES+INT(PAGLN)                                          REXW2400
           CALL PUTHDR(3,PC,NPAG)                                           REXW2410
           CALL PUTKBK(3,1)                                                 REXW2420
        ENDIF                                                               REXW2430
        CALL PUTKST(3,' '//LINEX)                                           REXW2440
        GO TO 1060                                                          REXW2450
1065    CONTINUE                                                            REXW2460
C                                                                           REXW2470
C       GO TO NEXT PAGE, PUT OUT W-ELEMENTS SHOWING FULL PRECISION          REXW2480
C       USE A TWO COLUMN FORMAT                                             REXW2490
C                                                                           REXW2500
        CALL NEWPAG(NPAG,INT(PAGLN),PC)                                     REXW2510
        CALL PUTHDR(3,PC,NPAG)                                              REXW2520
        CALL PUTDVR(3)                                                      REXW2530
        CALL PUTKBK(3,1)                                                    REXW2540
        CALL PUTKCT(3,'INITIAL VALUES FOR THE W ARRAY')                     REXW2550
        CALL PUTKCT(3,'ONLY NONZERO VALUES PRINTED'                         REXW2560
     1  //' -- ALL ANGLES IN RADIANS')                                      REXW2570
        CALL PUTKBK(3,1)                                                    REXW2580
        CALL PUTDVR(3)                                                      REXW2590
```

```
      LINSPP=INT(PAGLN)-7                                                    REXW2600
                                                                             REXW2610
C                                                                            REXW2620
      WFRMT='(BZ,I3,E14.7,A)'                                                REXW2630
      IF(NFRMAT.EQ.0) WFRMT='(I4,E24.15,A)'                                  REXW2640
C                                                                            REXW2650
      NWE=0                                                                  REXW2660
C     ALLOW MAXIMUM OF 10 PAGES OF OUTPUT(1 IS ENOUGH)                       REXW2670
      DO 18 NPGS=1,10                                                        REXW2680
C     ALLOW 3 LINES FOR INSIDE HEADER(SEE BELOW)                             REXW2690
         LINSPP=LINSPP-3                                                     REXW2700
C     NEXT ELEMENT TO SCAN IS ONE MORE THAN LAST ONE                         REXW2710
         LWE=NWE+1                                                           REXW2720
C                                                                            REXW2730
C     INITIALIZE SECOND COLUMN START INDEX                                   REXW2740
         NXCL=0                                                              REXW2750
C     INITIALIZE TOTAL ELEMENT COUNTER                                       REXW2760
         NELS=0                                                              REXW2770
C     LOOP TO FIND BREAK POINTS FOR FIRST/SECOND COLUMNS                     REXW2780
         DO 14 I=LWE,MAXW                                                    REXW2790
            IF(W(I).EQ.0.0) GO TO 14                                         REXW2800
            NWE=I                                                            REXW2810
            NELS=NELS+1                                                      REXW2820
            IF(NELS.EQ.LINSPP) NXCL=I                                        REXW2830
            IF(NELS.EQ.2*LINSPP) GO TO 15                                    REXW2840
14       CONTINUE                                                            REXW2850
15    IF(NELS.EQ.0) GO TO 22                                                 REXW2860
C                                                                            REXW2870
         IF(NPGS.GT.1) THEN                                                  REXW2880
            CALL NEWPAG(NPAG,INT(PAGLN),PC)                                  REXW2890
            CALL PUTHDR(3,PC,NPAG)                                           REXW2900
            LINSPP=INT(PAGLN)-1                                              REXW2910
         ENDIF                                                               REXW2920
C                                                                            REXW2930
C     INSERT 'INNER' HEADER                                                  REXW2940
         CALL PUTKBK(3,2)                                                    REXW2950
         STMP1=' N                    W(N)'                                  REXW2960
         CALL PUTKST(3,STMP1(:65)//' '//STMP1)                               REXW2970
C                                                                            REXW2980
          NX=NXCL-1                                                          REXW2990
          IF(NX.LT.LWE) NX=NWE                                               REXW3000
          DO 17 NW=LWE,NX                                                    REXW3010
          NCOMT=MIN0(NW,MXCMTS)                                              REXW3020
          IF(W(NW).NE.0) THEN                                                REXW3030
            WRITE(STMP1,WFRMT) NW,W(NW),WNOTES(WCOMTS(NCOMT))                REXW3040
            STMP2=' '                                                        REXW3050
            IF(NXCL.EQ.0) GO TO 23                                           REXW3060
            DO 19 I=NXCL,MAXW                                                REXW3070
            IF(W(I).NE.0) THEN                                               REXW3080
              NCOMT=MIN0(I,MXCMTS)                                           REXW3090
              WRITE(STMP2,WFRMT) I,W(I),WNOTES(WCOMTS(NCOMT))                REXW3100
              NXCL=I+1                                                       REXW3110
              GO TO 23                                                       REXW3120
            ENDIF                                                            REXW3130
19          CONTINUE                                                         REXW3140
23          CALL PUTKST(3,STMP1(:65)//' '//STMP2)
```

```
       ENDIF                                                            REXW3150
17     CONTINUE                                                         REXW3160
      IF(NELS.EQ.0) GO TO 22                                            REXW3170
18    CONTINUE                                                          REXW3180
C                                                                       REXW3190
22    CONTINUE                                                          REXW3200
      END                                                               REXW3210
```

```
      SUBROUTINE CLEAR(A,N)                                             CLYR0020
C     SET N ELEMENTS OF ARRAY A TO ZERO                                 CLYR0030
      REAL A(N)                                                         CLYR0040
      IF(N.LE.0) RETURN                                                 CLYR0050
      DO    10 I=1,N                                                    CLYR0060
10    A(I)=0.0                                                          CLYR0070
      RETURN                                                            CLYR0080
      END                                                               CLYR0090
```

```
      FUNCTION ND2B(INDEC)                                              ND2B0020
C                                                                       ND2B0030
C     CONVERT A NUMERIC DECIMAL DIGIT STRING TO A BIT STRING            ND2B0040
C     WITH EACH BIT SET BY A CORRESPONDING DECIMAL DIGIT.               ND2B0050
C                                                                       ND2B0060
      ND2B=0                                                            ND2B0070
      IF(INDEC.LE.0) RETURN                                             ND2B0080
      M=INDEC                                                           ND2B0090
      MB=1                                                              ND2B0100
C                                                                       ND2B0110
10    IF(MOD(M,10).NE.0) ND2B=ND2B+MB                                   ND2B0120
      MB=MB*2                                                           ND2B0130
      M=M/10                                                            ND2B0140
      IF(M.GT.0) GO TO 10                                               ND2B0150
      END                                                               ND2B0160
```

```
      LOGICAL FUNCTION UCON(JC1,U,CONV)                                 UCON0020
      LOGICAL SETUCON                                                   UCON0030
C     PROVIDES KEYWORD UNITS CONVERSION FOR W-ARRAY INPUT.              UCON0040
C     UNITS SPECIFICATION MUST BE IN THE FORM 'DV UN' WHERE DV IS THE   UCON0050
C     TYPE OF VARIABLE(SUCH AS AN FOR ANGLE) AND UN IS THE UNITS CHOICE UCON0060
C     SUCH AS RD FOR RADIANS). RETURN VALUE IS THE CONVERSION FACTOR.   UCON0070
C     COMMON DECK "CUCON" INSERTED HERE                                 CUCO0020
      COMMON/UCONV/CNVV(5,4)                                            CUCO0040
      CHARACTER PCV*3,CNVC*2                                            CUCO0050
      COMMON/UCONC/PCV(5),CNVC(5,4)                                     CUCO0060
```

```
C       COMMON DECK "CONST" INSERTED HERE                                      CCON0020
        COMMON/PCONST/CREF,RGAS,GAMMA                                          CCON0040
        COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                              CCON0050
C       COMMON DECK "WWR" INSERTED HERE                                        CWWR0020
        PARAMETER (NWARSZ=1000)                                                CWW10030
        COMMON/WW/ID(10),MAXW,W(NWARSZ)                                        CWW10040
        REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                     CWW20020
        EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),         CWW20030
      1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),            CWW20040
      2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),                CWW20050
      3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),               CWW20060
      8   (RUNSUP,W(18)),(RCVRH,W(20)),                                        CWW20070
      4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))      CWW20080
      5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                            CWW20090
      6 (HMIN,W(27)),(RGMAX,W(28)),                                            CWW20100
      8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                           CWW20110
      6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),             CWW20120
      7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))              CWW20130
      9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),              CWW20140
      1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                    CWW20150
      2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                         CWW20160
        CHARACTER U*(*),PU*3,PV*2                                              UCON0110
C                                                                              UCON0120
        DATA NPV,NPU/5,4/                                                      UCON0130
        UCON=.FALSE.                                                           UCON0140
        CONV=1.0                                                               UCON0150
C                                                                              UCON0160
        READ(U,100) PU,PV                                                      UCON0170
100     FORMAT(A,A)                                                            UCON0180
C                                                                              UCON0190
        DO 2010 J=1,NPV                                                        UCON0200
2010       IF(PU.EQ.PCV(J)) GO TO 2015                                         UCON0210
        JC1=4                                                                  UCON0220
        RETURN                                                                 UCON0230
2015    JC1=J                                                                  UCON0240
        DO 2020 K=1,NPU                                                        UCON0250
2020       IF(PV.EQ.CNVC(J,K)) GO TO 2025                                      UCON0260
        JC1=4                                                                  UCON0270
        RETURN                                                                 UCON0280
2025    UCON=.TRUE.                                                            UCON0290
        CONV=CNVV(JC1,K)                                                       UCON0300
        IF(CONV.EQ.-1.0) CONV=1.0/EARTHR                                       UCON0310
        RETURN                                                                 UCON0320
C                                                                              UCON0330
C       INITIAL CONVERSION CONSTANTS                                           UCON0340
        ENTRY SETUCON(JC1,U,CONV)                                              UCON0350
        SETUCON=.TRUE.                                                         UCON0360
C                                                                              UCON0370
        CNVV(1,2) = RAD                                                        UCON0380
C       USE TAG -1 FOR DYNAMIC USE OF EARTHR                                   UCON0390
        CNVV(1,3) = -1.0                                                       UCON0400
        CNVV(2,2) = 1.0E-3                                                     UCON0410
        CNVV(2,3) = 3.048006096E-4                                             UCON0420
        CNVV(2,4) = 1.852                                                      UCON0430
        CNVV(3,2) = PIT2                                                       UCON0440
```

```
      CNVV(3,3) = PIT2*1.0E3                                                    UCON0450
      CNVV(3,4) = -PIT2                                                         UCON0460
      CNVV(4,1) = 1.0/4.3429448                                                 UCON0470
      END                                                                       UCON0480
```

```
      SUBROUTINE TRACE                                                          TRWE0020
C                                                                               TRWE0030
C     CALCULATES ONE RAYPATH FOR THE REQUESTED NUMBER OF HOPS                   TRWE0040
C     (CROSSINGS OR CLOSEST APPROACH TO RECEIVER SURFACE; A CLOSEST             TRWE0050
C     APPROACH COUNTS AS TWO HOPS).                                             TRWE0060
      REAL NORYST                                                               TRWE0070
      PARAMETER (NORYST=0.0,MXPRGS=200)                                         TRWE0080
C                                                                               TRWE0090
C     COMMON DECK "SS" INSERTED HERE                                            CSS 0020
      REAL MODSURF                                                              CSS 0040
      COMMON/SS/ MODSURF(4)                                                     CSS 0050
      COMMON/SS/U,PUR,PURR,PURTH,PURPH                                          CSS 0060
      COMMON/SS/PUTH,PUPH,PUTHTH,PUPHPH,PUTHPH,USELECT,UTIME                    CSS 0070
C     COMMON DECK "RK" INSERTED HERE                                            CRK 0020
C     DEFINE SIZE REQUIRED FOR RAY STATE SAVE ARRAY                             CRK 0040
      PARAMETER (LRKAMS=87+2*100,NXRKMS=12+LRKAMS,MXEQPT=21)                    CRK 0050
      PARAMETER (NRKSAV=NXRKMS+MXEQPT-1)                                        CRK 0060
      COMMON /RK/ NEQS,STEP,MODE,E1MAX,E1MIN,E2MAX,E2MIN,FACT,RSTART            CRK 0070
C     COMMON DECK "FLAG" INSERTED HERE                                          CFLA0020
      LOGICAL NEWWR,NEWWP,NEWTRC,PENET                                          CFLA0040
      COMMON /FLG/ NTYP,NEWWR,NEWWP,NEWTRC,PENET,LINES,IHOP,HPUNCH              CFLA0050
      COMMON/FLGP/NSET                                                          CFLA0060
C     COMMON DECK "TRAC" INSERTED HERE                                          CTRA0020
      LOGICAL GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY                            CTRA0040
      COMMON /TRAC/ GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY,SMT,OSMT             CTRA0050
      COMMON/TRAC/ROLD(20),DROLD(20),TOLD,ZDOT,D2Z,RAD,RAD1                     CTRA0060
C     COMMON DECK "RR" INSERTED HERE                                            CRR 0020
      REAL MODREC                                                               CRR 0040
      COMMON/RR/ MODREC(4)                                                      CRR 0050
      COMMON/RR/F,PFR,PFRR,PFRTH,PFRPH                                          CRR 0060
      COMMON/RR/PFTH,PFPH,PFTHTH,PFPHPH,PFTHPH,FSELECT,FTIME                    CRR 0070
C     COMMON DECK "GG" INSERTED HERE                                            CGG 0020
      REAL MODG                                                                 CGG 0040
      COMMON/GG/MODG(4)                                                         CGG 0050
      COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                          CGG 0060
      COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME                    CGG 0070
C     COMMON DECK "RKTIME" INSERTED HERE                                        CRKT0020
      COMMON/CRKTIME/RKTIME                                                     CRKT0040
C     COMMON DECK "RINPLEX" INSERTED HERE                                       CRIN0020
      REAL KAY2,KAY2I                                                           CRIN0040
      COMPLEX PNP,POLAR,LPOLAR                                                  CRIN0050
      LOGICAL SPACE                                                             CRIN0060
      CHARACTER DISPM*6                                                         CRIN0070
      COMMON/RINPL/DISPM                                                        CRIN0080
      COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                         CRIN0090
      COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I                                     CRIN0100
```

```
      COMMON/RIN/PNP(10),POLAR,LPOLAR,SGN                                CRIN0110
      COMMON R(20),T,STP,DRDT(20)                                        TRWE0180
C     COMMON DECK "WWR" INSERTED HERE                                    CWWR0020
      PARAMETER (NWARSZ=1000)                                            CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                    CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                 CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),     CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),       CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),           CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),          CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                  CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
C     THE FOLLOWING IS A LOCAL COMMON ONLY                              TRWE0200
C      COMMON DECK "TRLOCAL" INSERTED HERE                              CTRL0020
       COMMON/TRLOCAL/RSIGN,HOME,FDOT,GDOT,GOLD,GDOLD,UDOT,UOLD,UDOLD   CTRL0040
       LOGICAL PCROSS,HOME,WASNT                                        TRWE0220
       LOGICAL LAUNCH,APOGEE,PLTENB,EXTON                               TRWE0230
       EQUIVALENCE (R(2),TH),(R(3),PHI)                                 TRWE0240
C                                                                       TRWE0250
       CHARACTER*10 XCOND,DEFCND                                        TRWE0260
       PARAMETER (DEFCND=' MAX HOPS')                                   TRWE0270
C                                                                       TRWE0280
      REAL PRGHST(3,MXPRGS)                                             TRWE0290
      EQUIVALENCE(RKTIME,IRKTIME), (APOG,PRIGEE,W(80))                  TRWE0300
      EQUIVALENCE (BOTABS,W(19))                                        TRWE0310
C                                                                       TRWE0320
C     INCREMENT EVENT COUNT                                             TRWE0330
      IRKTIME=IRKTIME+1                                                 TRWE0340
      PLTENB=APOG.EQ.0.0                                                TRWE0350
C     ENABLE EXTINCTION TEST IF LIMIT IS SET AND VARIABLE IS BEING INTE TRWE0360
      EXTON=(EXTINC.GT.0.0) .AND. (W(58).NE.0.0)                        TRWE0370
C     POINT INDEX TO ABSORTION VARIABLE                                 TRWE0380
      NR=7                                                              TRWE0390
      IF(W(57).NE.0.0) NR=8                                             TRWE0400
      RMAX=HMAX+EARTHR                                                  TRWE0410
      RMIN=HMIN+EARTHR                                                  TRWE0420
      NHOP=HOP                                                          TRWE0430
      MAX=MAXSTP                                                        TRWE0440
      NSKIP=SKIP                                                        TRWE0450
C********* INITIALIZE PARAMETERS FOR INTEGRATION SUBROUTINE RKAM        TRWE0460
      MODE=INTYP                                                        TRWE0470
      STEP=STEP1                                                        TRWE0480
      E1MAX=MAXERR                                                      TRWE0490
      E1MIN=MAXERR/ERATIO                                               TRWE0500
      E2MAX=STPMAX                                                      TRWE0510
      E2MIN=STPMIN                                                      TRWE0520
      FACT=FACTR                                                        TRWE0530
```

```
      RSTART=1.                                                          TRWE0540
      CALL HAMLTN                                                        TRWE0550
      FDOT=DOT(F)                                                        TRWE0560
C                                                                        TRWE0570
C     CHECK FOR EQUALITY WITH RECEIVER HEIGHT WITHIN MACHINE PRECISION   TRWE0580
      THERE=F.EQ.0.0                                                     TRWE0590
C                                                                        TRWE0600
      HOME=FDOT*F.GE.0.                                                  TRWE0610
C********* IHOP=0 TELLS PRINTR TO PRINT HEADING AND PUNCH A TRANSMITTER  TRWE0620
C********* RAYSET AND NEWTRC=TRUE TELLS RAYPLT TO START A NEW RAY        TRWE0630
      NEWTRC=.TRUE.                                                      TRWE0640
      IHOP=0                                                             TRWE0650
C********* RESET PERIGEE PLOT COUNTER                                    TRWE0660
      NPRGS=0                                                            TRWE0670
C     USE CURRENT RELATIVE POSITION OF RAY TO PREDICT                    TRWE0680
C     SIGN OF NEXT INTERSECTION OF RAY WITH RECEIVER HEIGHT              TRWE0690
C     (WILL SOON BE REVERSED).                                           TRWE0700
      RSIGN=SIGN(1.0,F)                                                  TRWE0710
C     IF RAY IS LAUNCHED FROM RECEIVER HEIGHT, USE DIRECTION INSTEAD     TRWE0720
      IF(THERE) RSIGN=SIGN(1.0,FDOT)                                     TRWE0730
      UDOT=DOT(U)                                                        TRWE0740
      GDOT=DOT(G)                                                        TRWE0750
C     TEST CASE FOR RECEIVER ON THE GROUND                               TRWE0760
      IF(F.EQ.G) RSIGN=1.0                                               TRWE0770
C     IF RECEIVER IS ON TERRAIN THEN DIRECTION OF FIRST                  TRWE0780
C     RECEIVER CROSSING IS DOWNWARD                                      TRWE0790
      CALL PRINTR('TXMTR',RAYSET)                                        TRWE0800
      IF (PLTENB) CALL RAYPLT                                            TRWE0810
      NEWRAY=.TRUE.                                                      TRWE0820
      STH0=SIN(TH)                                                       TRWE0830
      CTH0=COS(TH)                                                       TRWE0840
      PHI0=PHI                                                           TRWE0850
C                                                                        TRWE0860
C     SET DEFAULT RAY EVENTS FOR 'PRINTR'                                TRWE0870
      XCOND=DEFCND                                                       TRWE0880
      XSET=NORYST                                                        TRWE0890
C                                                                        TRWE0900
C********* LOOP ON NUMBER OF HOPS                                       TRWE0910
10    IHOP=IHOP+1                                                        TRWE0920
C      REVERSE SIGN AT EACH CROSSING OF RECEIVER HEIGHT                  TRWE0930
      RSIGN=-RSIGN                                                       TRWE0940
      IF (IHOP.GT.NHOP) GO TO 100                                        TRWE0950
      PENET=.FALSE.                                                      TRWE0960
C********* LOOP ON MAXIMUM NUMBER OF STEPS PER HOP                       TRWE0970
      DO 79 J=1,MAX                                                      TRWE0980
C     LAUNCH=TRUE ONLY WHEN TRANSMITTER ON GROUND WITH DOWNWARD          TRWE0990
C     TRANSMISSION(POSSIBLE ONLY ON 1ST STEP).                           TRWE1000
      LAUNCH=G.EQ.0.0 .AND. GDOT.LT.0.0                                  TRWE1010
C                                                                        TRWE1020
C     SAVE CURRENT STATE VALUES FOR LATER COMPARISONS                    TRWE1030
C     RESTORE RAY STATE IF NEEDED                                        TRWE1040
12    DO 13 L=1,6                                                        TRWE1050
      ROLD(L)=R(L)                                                       TRWE1060
13    DROLD(L)=DRDT(L)                                                   TRWE1070
      FDOLD=FDOT                                                         TRWE1080
```

```
      GDOLD=GDOT                                                        TRWE1090
      UDOLD=UDOT                                                        TRWE1100
      FOLD=F                                                            TRWE1110
      GOLD=G                                                            TRWE1120
      UOLD=U                                                            TRWE1130
      TOLD=T                                                            TRWE1140
C                                                                       TRWE1150
C     PROCESS NEXT RAY POINT                                            TRWE1160
      CALL RKAM                                                         TRWE1170
      FDOT=DOT(F)                                                       TRWE1180
      WASNT=.NOT.HOME                                                   TRWE1190
      HOME=FDOT*F.GE.0.                                                 TRWE1200
C                                                                       TRWE1210
C     CHOOSE STRATEGY A OR B                                            TRWE1220
      IF(DROLD(1).GT.0.0) GO TO 15                                      TRWE1230
C                                                                       TRWE1240
C     BEGIN STRATEGY A                                                  TRWE1250
C                                                                       TRWE1260
C     LOOK FOR DOWNGOING CROSSING OF RECEIVER SURFACE                   TRWE1270
      IF(.NOT.LAUNCH.AND.F.LT.0.0.AND.RSIGN.LT.0.0) GO TO 50            TRWE1280
C     CHECK FOR CASE OF CLOSEST APPROACH                                TRWE1290
      IF (HOME.AND.WASNT) GO TO 30                                      TRWE1300
C     DETERMINE IF A GROUND CROSSING HAS OCCURED                        TRWE1310
      IF(PCROSS(G,GDOT)) GO TO 20                                       TRWE1320
C     LOOK FOR UPGOING CROSSING OF RECEIVER SURFACE                     TRWE1330
      IF(.NOT.LAUNCH.AND.F.GT.0.0.AND.RSIGN.GT.0.0) GO TO 50            TRWE1340
C     CROSSED UPPER SURFACE?                                            TRWE1350
      IF(PCROSS(U,UDOT)) GO TO 60                                       TRWE1360
      GO TO 18                                                          TRWE1370
C                                                                       TRWE1380
C     BEGIN STRATEGY B                                                  TRWE1390
C                                                                       TRWE1400
C     LOOK FOR UPGOING CROSSING OF RECEIVER SURFACE                     TRWE1410
15    IF(.NOT.LAUNCH.AND.F.GT.0.0.AND.RSIGN.GT.0.0) GO TO 50            TRWE1420
C     CHECK FOR CASE OF CLOSEST APPROACH                                TRWE1430
      IF (HOME.AND.WASNT) GO TO 30                                      TRWE1440
C     CROSSED UPPER SURFACE?                                            TRWE1450
      IF(PCROSS(U,UDOT)) GO TO 60                                       TRWE1460
C     LOOK FOR DOWNGOING CROSSING OF RECEIVER SURFACE                   TRWE1470
      IF(.NOT.LAUNCH.AND.F.LT.0.0.AND.RSIGN.LT.0.0) GO TO 50            TRWE1480
C     DETERMINE IF A GROUND CROSSING HAS OCCURED                        TRWE1490
      IF(PCROSS(G,GDOT)) GO TO 20                                       TRWE1500
C                                                                       TRWE1510
C     RAY HAS NOT CROSSED ANY SURFACE OF DISCONTINUITY                  TRWE1520
C     OR RECEIVER SURFACE. PRINT OUT ANY SPECIAL CONDITIONS             TRWE1530
C     OF THE RAY.                                                       TRWE1540
C                                                                       TRWE1550
C     CHECK FOR PERIGEE CONDITION                                       TRWE1560
18    IF (DROLD(1).GE.0..OR.DRDT(1).LE.0.) GO TO 25                     TRWE1570
      CALL PRINTR(' PERIGEE ',NORYST)                                   TRWE1580
      IF(PRIGEE.EQ.0.0) GO TO 25                                        TRWE1590
C                                                                       TRWE1600
      IF(NPRGS.GE.MXPRGS) CALL STOPIT('PRG LIMIT')                      TRWE1610
      NPRGS=NPRGS+1                                                     TRWE1620
      CALL RMOVE(PRGHST(1,NPRGS),R,3)                                   TRWE1630
```

```
C                                                                       TRWE1640
25    APOGEE=DROLD(1).GT.0..AND.DRDT(1).LT.0.                           TRWE1650
      PLTENB=APOG.EQ.0.0.OR.APOGEE                                      TRWE1660
C                                                                       TRWE1670
      IF(APOGEE) CALL PRINTR(' APOGEE  ',NORYST)                        TRWE1680
      IF (DROLD(2)*DRDT(2).LT.0.) CALL PRINTR(' MAX LAT ',NORYST)       TRWE1690
      IF (DROLD(3)*DRDT(3).LT.0.) CALL PRINTR(' MAX LONG',NORYST)       TRWE1700
      DO 14 I=4,6                                                       TRWE1710
      IF (ROLD(I)*R(I).LT.0.) CALL PRINTR(' WAVE REV',NORYST)           TRWE1720
   14 CONTINUE                                                          TRWE1730
      GO TO 75                                                          TRWE1740
C********* RAY WENT UNDERGROUND                                        TRWE1750
C     USE 'FULL' REGULAR PARABOLIC FIT(ENTRY 'FIT')                     TRWE1760
20    IF(BOTABS.NE.0.0) THEN                                            TRWE1770
         CALL RCROSS(-1.,G,GDOT,'XBOTM ABS',GROUND)                     TRWE1780
         XCOND=' '                                                      TRWE1790
      ELSE                                                              TRWE1800
         CALL RCROSS(-1.,G,GDOT,'GBOTM REF',GROUND)                     TRWE1810
      ENDIF                                                             TRWE1820
C                                                                       TRWE1830
      GO TO 75                                                          TRWE1840
C********* RAY WENT THROUGH UPPER SURFACE                              TRWE1850
C     USE 'FULL' REGULAR PARABOLIC FIT(ENTRY 'FIT')                     TRWE1860
60    CALL RCROSS(1.,U,UDOT,'SSURF REF',SURF)                           TRWE1870
      GO TO 75                                                          TRWE1880
C********* RAY MAY HAVE MADE A CLOSEST APPROACH                        TRWE1890
C     USE 'FULL' REGULAR PARABOLIC FIT                                  TRWE1900
30    CALL FIT(F,FOLD,FDOLD)                                            TRWE1910
      IF(RAD.GE.0.0) GO TO 501                                          TRWE1920
C     ESTIMATE TIME OF CLOSEST APPROACH                                 TRWE1930
      TP=T-ZDOT/D2Z                                                     TRWE1940
      CALL GRAZE(F,RSIGN,TP)                                            TRWE1950
      FDOT=ZDOT                                                         TRWE1960
      IF (THERE) GO TO 51                                               TRWE1970
C                                                                       TRWE1980
C     SET DRDT(1)=0 TO AVOID INCORRECT APOGEE OR PERIGEE PRINTOUT, A NEWTRWE1990
C     VALUE WILL BE COMPUTED BEFORE FURTHER ANALYSIS IS DONE            TRWE2000
40    DRDT(1)=0.                                                        TRWE2010
      HPUNCH=R(1)-EARTHR                                                TRWE2020
      CALL PRINTR('MMIN DIST',RAYSET)                                   TRWE2030
      IF(PLTENB) CALL RAYPLT                                            TRWE2040
      IF (IHOP.GE.NHOP) GO TO 100                                       TRWE2050
      IHOP=IHOP+1                                                       TRWE2060
      RSIGN=-RSIGN                                                      TRWE2070
      CALL PRINTR ('MMIN DIST',RAYSET)                                  TRWE2080
      GO TO 89                                                          TRWE2090
C********* RAY CROSSED RECEIVER HEIGHT                                 TRWE2100
50    CALL FIT(F,FOLD,FDOLD)                                            TRWE2110
C     ESTIMATE GROUP TIME WHEN RAY CROSSES RECEIVER HEIGHT              TRWE2120
C     (IN THE CORRECT DIRECTION).                                       TRWE2130
501   TC=T-2.0*F/(ZDOT+SIGN(RAD1,RSIGN))                                TRWE2140
      CALL BACKUP(F,RSIGN,TC)                                           TRWE2150
      FDOT=ZDOT                                                         TRWE2160
      IF(.NOT.THERE) GO TO 40                                           TRWE2170
51    HPUNCH=R(1)-EARTHR                                                TRWE2180
```

```
      CALL PRINTR('RRCVR     ',RAYSET)                                    TRWE2190
      IF (PLTENB) CALL RAYPLT                                             TRWE2200
      IF (GET(F).NE.GET(G)) GO TO 89                                      TRWE2210
C                                                                         TRWE2220
C     RECEIVER IS ON TERRAIN                                              TRWE2230
      IF (IHOP.GE.NHOP) GO TO 100                                         TRWE2240
      IHOP=IHOP+1                                                         TRWE2250
      RSIGN=-RSIGN                                                        TRWE2260
C********* GROUND REFLECT                                                TRWE2270
      CALL REFLECT(G)                                                     TRWE2280
      CALL HAMLTN                                                         TRWE2290
      FDOT=DOT(F)                                                         TRWE2300
      RSTART=1.                                                           TRWE2310
      HPUNCH=R(1)-EARTHR                                                  TRWE2320
      CALL PRINTR('GBOTM REF',RAYSET)                                     TRWE2330
      THERE=.TRUE.                                                        TRWE2340
      HPUNCH=R(1)-EARTHR                                                  TRWE2350
      CALL PRINTR ('RRCVR    ',RAYSET)                                    TRWE2360
      GO TO 89                                                            TRWE2370
C*********                                                                TRWE2380
75    IF (PLTENB) CALL RAYPLT                                             TRWE2390
      PLTENB=APOG.EQ.0.0                                                  TRWE2400
      IF(EXTON) THEN                                                      TRWE2410
         IF (R(NR).GE.EXTINC) XCOND='EEXTINC'                             TRWE2420
      ENDIF                                                               TRWE2430
      IF (R(1).GT.RMAX.AND.F.GT.0.0.AND.FDOT.GT.0.) XCOND='UMAX HT'       TRWE2440
      IF (R(1).LT.RMIN.AND.F.LT.0.0.AND.FDOT.LT.0.) XCOND='DMIN HT'       TRWE2450
      RANGE=EARTHR*ACOS(COS(TH)*CTH0+SIN(TH)*STH0*COS(PHI-PHI0))          TRWE2460
      IF(RGMAX.GT.0.0.AND.RGMAX.LT.RANGE) XCOND='FMAX RANGE'              TRWE2470
      IF(XCOND.NE.DEFCND) GO TO 90                                        TRWE2480
C                                                                         TRWE2490
      IF (MOD(J,NSKIP).EQ.0) CALL PRINTR('        ',NORYST)               TRWE2500
79    CONTINUE                                                            TRWE2510
C********* EXCEEDED MAXIMUM NUMBER OF STEPS                              TRWE2520
      HPUNCH=R(1)-EARTHR                                                  TRWE2530
      CALL PRINTR('SSTEP MAX',RAYSET)                                     TRWE2540
      GO TO 100                                                           TRWE2550
C                                                                         TRWE2560
C*********                                                                TRWE2570
89    HOME=.TRUE.                                                         TRWE2580
      GDOT=DOT(G)                                                         TRWE2590
      GO TO 10                                                            TRWE2600
C********* RAY PENETRATED COMPUTATIONAL AREA BOUNDARY AND WAS HEADING OUTTRWE2610
90    PENET=.TRUE.                                                        TRWE2620
      HPUNCH=R(1)-EARTHR                                                  TRWE2630
      XSET=RAYSET                                                         TRWE2640
C                                                                         TRWE2650
C     NORMAL EXIT                                                         TRWE2660
100   IF(XCOND.NE.' ') CALL PRINTR(XCOND,XSET)                            TRWE2670
      IF(NPRGS.LE.0) RETURN                                               TRWE2680
C                                                                         TRWE2690
      CALL DASH                                                           TRWE2700
      NEWTRC=.TRUE.                                                       TRWE2710
      DO 110 I=1,NPRGS                                                    TRWE2720
      CALL RMOVE(R,PRGHST(1,I),3)                                         TRWE2730
```

```
110     CALL RAYPLT                                                     TRWE2740
C                                                                       TRWE2750
        CALL RESET('DASH')                                              TRWE2760
        END                                                             TRWE2770
```

```
        LOGICAL FUNCTION PCROSS(Z,ZDT)                                  TRWE2780
C       RETURNS 'TRUE' IF PARABOLIC 'FIT2'(SEE 'FIT' ROUTINE)           TRWE2790
C       INDICATES CROSSING OF SURFACE 'Z'.                              TRWE2800
        REAL ZDT(3)                                                     TRWE2810
C       COMMON DECK "TRAC" INSERTED HERE                                CTRA0020
        LOGICAL GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY                  CTRA0040
        COMMON /TRAC/ GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY,SMT,OSMT   CTRA0050
        COMMON/TRAC/ROLD(20),DROLD(20),TOLD,ZDOT,D2Z,RAD,RAD1           CTRA0060
C                                                                       TRWE2830
C       USE SPECIAL PARABOLIC FIT GIVING 'SMT' AND 'OSMT'               TRWE2840
        ZDT(1)=DOT(Z)                                                   TRWE2850
        PCROSS=.TRUE.                                                   TRWE2860
        IF(Z.LT.0.0) GO TO 20                                           TRWE2870
        IF(ZDT(1).LE.0.0.OR.ZDT(3).GE.0.0) GO TO 15                     TRWE2880
        CALL FIT2(Z,ZDT(2),ZDT(3))                                      TRWE2890
        IF(SMT.GT.Z.OR.OSMT.GT.ZDT(2)) GO TO 20                         TRWE2900
C                                                                       TRWE2910
15      PCROSS=.FALSE.                                                  TRWE2920
20      RETURN                                                          TRWE2930
        END                                                             TRWE2940
```

```
        SUBROUTINE RCROSS(S,Z,ZDT,EVENT,QMODE)                          TRWE2950
C       FIND ESTIMATED CROSSING POINT OF SURFACE 'Z' THEN USE           TRWE2960
C       ROUTINE 'BACKUP' TO GO THERE AND PERFORM A RAY REFLECTION       TRWE2970
C       WITH ROUTINE 'REFLECT'.                                         TRWE2980
C       EXTENDS RAY PROPAGATION TO INTERSECTION WITH REFLECTING SURFACE TRWE2990
C       Z AND CALLS 'REFLECT' TO OBTAIN VECTOR COMPONENTS.              TRWE3000
C                                                                       TRWE3010
        CHARACTER EVENT*8                                               TRWE3020
        REAL ZDT(3)                                                     TRWE3030
        LOGICAL QMODE                                                   TRWE3040
C       COMMON DECK "TRAC" INSERTED HERE                                CTRA0020
        LOGICAL GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY                  CTRA0040
        COMMON /TRAC/ GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY,SMT,OSMT   CTRA0050
        COMMON/TRAC/ROLD(20),DROLD(20),TOLD,ZDOT,D2Z,RAD,RAD1           CTRA0060
C       COMMON DECK "RR" INSERTED HERE                                  CRR 0020
        REAL MODREC                                                     CRR 0040
        COMMON/RR/ MODREC(4)                                            CRR 0050
        COMMON/RR/F,PFR,PFRR,PFRTH,PFRPH                                CRR 0060
        COMMON/RR/PFTH,PFPH,PFTHTH,PFPHPH,PFTHPH,FSELECT,FTIME          CRR 0070
C       COMMON DECK "WWR" INSERTED HERE                                 CWWR0020
        PARAMETER (NWARSZ=1000)                                         CWW10030
```

```
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                          CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                       CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),           CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),             CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),                 CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),                CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                        CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))      CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                            CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                            CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                           CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),             CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))              CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),              CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                    CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                         CWW20160
C     COMMON DECK "FLAG" INSERTED HERE                                        CFLA0020
      LOGICAL NEWWR,NEWWP,NEWTRC,PENET                                        CFLA0040
      COMMON /FLG/ NTYP,NEWWR,NEWWP,NEWTRC,PENET,LINES,IHOP,HPUNCH            CFLA0050
      COMMON/FLGP/NSET                                                        CFLA0060
      COMMON R(20),T,STP,DRDT(20)                                             TRWE3090
      COMMON /RK/ N,STEP,MODE,E1MAX,E1MIN,E2MAX,E2MIN,FACT,RSTART             TRWE3100
      LOGICAL HOME                                                            TRWE3110
C     COMMON DECK "TRLOCAL" INSERTED HERE                                     CTRL0020
      COMMON/TRLOCAL/RSIGN,HOME,FDOT,GDOT,GOLD,GDOLD,UDOT,UOLD,UDOLD          CTRL0040
C                                                                             TRWE3130
      IF(QMODE) GO TO 60                                                      TRWE3140
      CALL FIT(Z,ZDT(2),ZDT(3))                                               TRWE3150
      TC=T-2.0*Z/(ZDOT-RAD1)                                                  TRWE3160
      CALL BACKUP(Z,-1.,TC)                                                   TRWE3170
C********* GROUND REFLECT                                                     TRWE3180
60    CALL REFLECT(Z)                                                         TRWE3190
      CALL HAMLTN                                                             TRWE3200
      ZDT(1)=DOT(Z)                                                           TRWE3210
      FDOT=DOT(F)                                                             TRWE3220
      HOME=FDOT*F.GE.0.0                                                      TRWE3230
      RSTART=1.                                                               TRWE3240
      HPUNCH=R(1)-EARTHR                                                      TRWE3250
      CALL PRINTR(EVENT,RAYSET)                                               TRWE3260
      IF(F.NE.Z) RETURN                                                       TRWE3270
C     AVOID RECEIVER CROSSING AT THE TRANSMITTER IF RECEIVER ON TERRAIN        TRWE3280
      RSIGN=S                                                                 TRWE3290
      HOME=.TRUE.                                                             TRWE3300
      END                                                                     TRWE3310
```

```
      SUBROUTINE HAMLTN                                                       HAKN0020
C********* CALCULATES HAMILTONS EQUATIONS FOR RAY TRACING                     HAKN0030
C     AND OTHER QUANTITIES TO BE INTEGRATED                                   HAKN0040
C     COMMON DECK "GG" INSERTED HERE                                          CGG 0020
      REAL MODG                                                               CGG 0040
      COMMON/GG/MODG(4)                                                       CGG 0050
```

```
      COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                  CGG 0060
      COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME            CGG 0070
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "RINREAL" INSERTED HERE                               CRIN0020
      LOGICAL SPACE                                                     CRIN0040
      REAL LPOLAR,LPOLRI,KPHK,KPHKI,KAY2,KAY2I                          CRIN0050
      CHARACTER DISPM*6                                                 CRIN0060
      COMMON/RINPL/DISPM                                                CRIN0070
      COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                 CRIN0080
      COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I,                            CRIN0090
     1             H,HI,PHT,PHTI,PHR,PHRI,PHTH,PHTHI,PHPH,PHPHI         CRIN0100
     2, PHOW,PHOWI,PHKR,PHKRI,PHKTH,PHKTI, PHKPH,PHKPI                  CRIN0110
     3 ,KPHK,KPHKI,POLAR,POLARI,LPOLAR,LPOLRI,SGN                       CRIN0120
      COMMON R(20),T,STP,DRDT(20)                                       HAKN0080
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      EQUIVALENCE (TH,R(2)),(PH,R(3)),(KR,R(4)),(KTH,R(5)),(KPH,R(6)),  HAKN0100
     1 (DTHDT,DRDT(2)),(DPHDT,DRDT(3)),(DKRDT,DRDT(4)),(DKTHDT,DRDT(5)),HAKN0110
     2 (DKPHDT,DRDT(6)),(HMAX,W(26)),(OW,W(6))                          HAKN0120
      REAL KR,KTH,KPH                                                   HAKN0130
      STH=SIN(TH)                                                       HAKN0140
      CTH=SIN(PID2-TH)                                                  HAKN0150
      RSTH=R(1)*STH                                                     HAKN0160
      RCTH=R(1)*CTH                                                     HAKN0170
      CALL DISPER                                                       HAKN0180
      DENPHC=1.0/(PHOW*CREF)                                            HAKN0190
      DRDT(1)=-PHKR*DENPHC                                              HAKN0200
      DTHDT=-PHKTH*DENPHC/R(1)                                          HAKN0210
      DPHDT=-PHKPH*DENPHC/RSTH                                          HAKN0220
      DKRDT=PHR*DENPHC+KTH*DTHDT+KPH*STH*DPHDT                          HAKN0230
      DKTHDT=(PHTH*DENPHC-KTH*DRDT(1)+KPH*RCTH*DPHDT)/R(1)              HAKN0240
      DKPHDT=(PHPH*DENPHC-KPH*STH*DRDT(1)-KPH*RCTH*DTHDT)/RSTH          HAKN0250
      NR=6                                                              HAKN0260
C********* PHASE PATH                                                  HAKN0270
      IF (W(57).EQ.0.) GO TO 10                                         HAKN0280
      NR=NR+1                                                           HAKN0290
      DRDT(NR)=-     KPHK/PHOW/OW                                       HAKN0300
C********* ABSORPTION                                                  HAKN0310
   10 IF (W(58).EQ.0.) GO TO 15                                         HAKN0320
      NR=NR+1                                                           HAKN0330
      DRDT(NR)= 10./ALN10*KPHK*KAY2I/(KR*KR+KTH*KTH+KPH*KPH)*DENPHC     HAKN0340
C********* DOPPLER SHIFT                                               HAKN0350
   15 IF (W(59).EQ.0.) GO TO 20                                         HAKN0360
      NR=NR+1                                                           HAKN0370
      DRDT(NR)=-PHT*DENPHC/PIT2                                         HAKN0380
C********* GEOMETRICAL PATH LENGTH                                     HAKN0390
   20 IF (W(60).EQ.0.) GO TO 25                                         HAKN0400
      NR=NR+1                                                           HAKN0410
      DRDT(NR)=SQRT(PHKR**2+PHKTH**2+PHKPH**2)*ABS(DENPHC)              HAKN0420
C********* TERRAIN FUNCTION AS COMPLETE INTEGRAL                       HAKN0430
25    IF(ABS(W(61))+ABS(W(62))+ABS(W(63))+ABS(W(64)).EQ.0.0) GO TO 45   HAKN0440
      CALL TOPOG                                                        HAKN0450
C                                                                       HAKN0460
```

```
      IF(W(61).EQ.0.) GO TO 30                                          HAKN0470
      NR=NR+1                                                           HAKN0480
      DRDT(NR)=PGR*DRDT(1)+PGTH*DTHDT+PGPH*DPHDT                        HAKN0490
C********* TERRAIN TIME DERIVATIVES AS COMPLETE INTEGRALS               HAKN0500
   30 IF(W(62).EQ.0.) GO TO 35                                          HAKN0510
      NR=NR+1                                                           HAKN0520
C     DERIVATIVE WITH RESPECT TO R                                      HAKN0530
      DRDT(NR)=PGRR*DRDT(1)+PGRTH*DTHDT+PGRPH*DPHDT                     HAKN0540
   35 IF(W(63).EQ.0.) GO TO 40                                          HAKN0550
      NR=NR+1                                                           HAKN0560
C     DERIVATIVE WITH RESPECT TO THETA                                  HAKN0570
      DRDT(NR)=PGRTH*DRDT(1)+PGTHTH*DTHDT+PGTHPH*DPHDT                  HAKN0580
   40 IF(W(64).EQ.0.) GO TO 45                                          HAKN0590
      NR=NR+1                                                           HAKN0600
C     DERIVATIVE WITH RESPECT TO PHI                                    HAKN0610
      DRDT(NR)=PGRPH*DRDT(1)+PGTHPH*DTHDT+PGPHPH*DPHDT                  HAKN0620
C********* OTHER CALCULATIONS                                          HAKN0630
45    CONTINUE                                                          HAKN0640
      RETURN                                                            HAKN0650
         END                                                            HAKN0660
```

```
      SUBROUTINE RKAM                                                   RKAM0020
C     KEEPS TRACK OF INTERNAL INTEGRATION STEPS PERFORMED BY THE RKAM1  RKAM0030
C     ROUTINE AND MAKES THEM AVAILABLE TO CALLING ROUTINES.             RKAM0040
      COMMON/RKAMS/XV(5),FV(4,20),YU(5,20),EPM,ALPHA,MM                 RKAM0050
C     COMMON DECK "RK" INSERTED HERE                                    CRK 0020
C     DEFINE SIZE REQUIRED FOR RAY STATE SAVE ARRAY                     CRK 0040
      PARAMETER (LRKAMS=87+2*100,NXRKMS=12+LRKAMS,MXEQPT=21)            CRK 0050
      PARAMETER (NRKSAV=NXRKMS+MXEQPT-1)                                CRK 0060
      COMMON /RK/ NEQS,STEP,MODE,E1MAX,E1MIN,E2MAX,E2MIN,FACT,RSTART    CRK 0070
C     COMMON DECK "RKTIME" INSERTED HERE                                CRKT0020
      COMMON/CRKTIME/RKTIME                                             CRKT0040
      COMMON Y(20),T,SPACE,DYDT(20)                                     RKAM0080
      REAL RV(1)                                                        RKAM0090
      DOUBLE PRECISION YU                                               RKAM0100
      EQUIVALENCE(RKTIME,IRKTIME),(RV,R)                                RKAM0110
C                                                                       RKAM0120
      REAL SVBUF(NRKSAV)                                                RKAM0130
      LOGICAL SAVED                                                     RKAM0140
C                                                                       RKAM0150
      DATA SAVED/.FALSE./                                               RKAM0160
C     PERFORM CLOCK ADVANCE(SEE 'GET' ROUTINE)                          RKAM0170
      IRKTIME=IRKTIME+1                                                 RKAM0180
C                                                                       RKAM0190
      IF(RSTART.NE.0.0 .OR. MODE.LE.2) GO TO 250                        RKAM0200
C                                                                       RKAM0210
      NV=NV+1                                                           RKAM0220
      IF(NV.LE.4) GO TO 300                                             RKAM0230
C                                                                       RKAM0240
250   TOLD=T                                                            RKAM0250
C     PERFORM NUMERICAL INTEGRATION TO TIME 'T'                         RKAM0260
```

```
      CALL RKAM1                                                        RKAM0270
      IF(MODE.LE.2) GO TO 400                                           RKAM0280
      IF(RSTART.NE.0.0) GO TO 260                                       RKAM0290
C     SEARCH FOR REQUESTED VALUE IN PIPELINE                            RKAM0300
C     TAKING SIGN OF 'STEP' INTO ACCOUNT.                               RKAM0310
      TUP=TOLD+.25*STEP                                                 RKAM0320
      DO 150 NV=1,3                                                     RKAM0330
      IF(SPACE.GT.0.0.AND.TUP.LT.XV(NV)) GO TO 300                      RKAM0340
      IF(SPACE.LT.0.0.AND.TUP.GT.XV(NV)) GO TO 300                      RKAM0350
150   CONTINUE                                                          RKAM0360
C     GOT IT ALREADY, RETURN                                            RKAM0370
      GO TO 400                                                         RKAM0380
C                                                                       RKAM0390
260   NV=1                                                              RKAM0400
C     RETRIEVE REQUESTED VALUES FROM PIPELINE                           RKAM0410
300   T=XV(NV)                                                          RKAM0420
      DO 350 I=1,NEQS                                                   RKAM0430
      Y(I)=YU(NV,I)                                                     RKAM0440
350   DYDT(I)=FV(NV,I)                                                  RKAM0450
C                                                                       RKAM0460
C     STANDARD EXIT SEQUENCE                                            RKAM0470
400   RETURN                                                            RKAM0480
C                                                                       RKAM0490
      ENTRY RKSAVE(SVBUF)                                               RKAM0500
C                                                                       RKAM0510
      SVBUF(1)=NV                                                       RKAM0520
      SVBUF(2)=NEQS                                                     RKAM0530
      SVBUF(3)=MM                                                       RKAM0540
      SVBUF(4)=SPACE                                                    RKAM0550
      SVBUF(5)=MODE                                                     RKAM0560
      CALL RMOVE(SVBUF(6),E1MAX,6)                                      RKAM0570
      CALL RMOVE(SVBUF(12),XV,LRKAMS)                                   RKAM0580
      CALL RMOVE(SVBUF(NXRKMS),Y,MXEQPT)                                RKAM0590
C                                                                       RKAM0600
      SAVED=.TRUE.                                                      RKAM0610
      RETURN                                                            RKAM0620
      ENTRY RKRSTR(SVBUF)                                               RKAM0630
C                                                                       RKAM0640
      IF(.NOT.SAVED) RETURN                                             RKAM0650
C                                                                       RKAM0660
      NV=SVBUF(1)                                                       RKAM0670
      NEQS=SVBUF(2)                                                     RKAM0680
      MM=SVBUF(3)                                                       RKAM0690
      SPACE=SVBUF(4)                                                    RKAM0700
      MODE=SVBUF(5)                                                     RKAM0710
      CALL RMOVE(E1MAX,SVBUF(6),6)                                      RKAM0720
      CALL RMOVE(XV,SVBUF(12),LRKAMS)                                   RKAM0730
      CALL RMOVE(Y,SVBUF(NXRKMS),MXEQPT)                                RKAM0740
C                                                                       RKAM0750
      WRITE(3,*) 'RESTORING NV,NEQS,MM,ALPHA,T='                        RKAM0760
     1   ,NV,NEQS,MM,ALPHA,T                                            RKAM0770
      RETURN                                                            RKAM0780
      END                                                               RKAM0790
```

```
      SUBROUTINE RKAM1                                                  RKK10020
C        NUMERICAL INTEGRATION OF DIFFERENTIAL EQUATIONS                RKK10030
C     THIS ROUTINE IS A MODIFICATION OF RKAMSUB, WHICH WAS WRITTEN      RKK10040
C     BY G.J. LASTMAN AND IS AVAILABLE THROUGH THE CDC CO-OP LIBRARY    RKK10050
C     AS 'D2 UTEX RKAMSUB'.                                             RKK10060
      COMMON /RK/ NN,SPACE,MODE,E1MAX,E1MIN,E2MAX,E2MIN,FACT,RSTART     RKK10070
      COMMON Y(20),T,STEP,DYDT(20)                                      RKK10080
      COMMON/RKAMS/XV(5),FV(4,20),YU(5,20),EPM,ALPHA,MM                 RKK10090
      DIMENSION DELY(4,20),BET(4)                                       RKK10100
      DOUBLE PRECISION YU                                               RKK10110
C                                                                       RKK10120
      IF (RSTART.EQ.0.) GO TO 1000                                      RKK10130
      LL=1                                                              RKK10140
      MM=1                                                              RKK10150
      IF (MODE.EQ.1) MM=4                                               RKK10160
      ALPHA=T                                                           RKK10170
      EPM=0.0                                                           RKK10180
      BET(1)=0.5                                                        RKK10190
      BET(2)=0.5                                                        RKK10200
      BET(3)=1.0                                                        RKK10210
      BET(4)=0.0                                                        RKK10220
      STEP=SPACE                                                        RKK10230
      R=19.0/270.0                                                      RKK10240
      XV(MM)=T                                                          RKK10250
      IF (E1MIN.LE.0.) E1MIN=E1MAX/50.                                  RKK10260
      IF (FACT.LE.0.) FACT=0.5                                          RKK10270
      CALL HAMLTN                                                       RKK10280
      DO 320 I=1,NN                                                     RKK10290
      FV(MM,I)=DYDT(I)                                                  RKK10300
  320 YU(MM,I)=Y(I)                                                     RKK10310
      RSTART=0.                                                         RKK10320
      GO TO 1001                                                        RKK10330
 1000 IF (MODE.NE.1) GO TO 2000                                         RKK10340
C                                                                       RKK10350
C          RUNGE-KUTTA                                                  RKK10360
 1001 DO 1034 K=1,4                                                     RKK10370
      DO 1350 I=1,NN                                                    RKK10380
      DELY(K,I)=STEP*FV(MM,I)                                           RKK10390
      Z=YU(MM,I)                                                        RKK10400
 1350 Y(I)=Z+BET(K)*DELY(K,I)                                           RKK10410
      T=BET(K)*STEP+XV(MM)                                              RKK10420
      CALL HAMLTN                                                       RKK10430
      DO 1034 I=1,NN                                                    RKK10440
 1034 FV(MM,I)=DYDT(I)                                                  RKK10450
      DO 1039 I=1,NN                                                    RKK10460
      DEL=(DELY(1,I)+2.0*DELY(2,I)+2.0*DELY(3,I)+DELY(4,I))/6.0         RKK10470
 1039 YU(MM+1,I)=YU(MM,I)+DEL                                           RKK10480
      MM=MM+1                                                           RKK10490
      XV(MM)=XV(MM-1)+STEP                                              RKK10500
      DO 1400 I=1,NN                                                    RKK10510
 1400 Y(I)=YU(MM,I)                                                     RKK10520
      T=XV(MM)                                                          RKK10530
```

```
      CALL HAMLTN                                                       RKK10540
      IF (MODE.EQ.1) GO TO 42                                           RKK10550
      DO 150 I=1,NN                                                     RKK10560
  150 FV(MM,I)=DYDT(I)                                                  RKK10570
      IF (MM.LE.3) GO TO 1001                                           RKK10580
C                                                                       RKK10590
C          ADAMS-MOULTON                                                RKK10600
 2000 DO 2048 I=1,NN                                                    RKK10610
      DEL=STEP*(55.*FV(4,I)-59.*FV(3,I)+37.*FV(2,I)-9.*FV(1,I))/24.     RKK10620
      Y(I)=YU(4,I)+DEL                                                  RKK10630
 2048 DELY(1,I)=Y(I)                                                    RKK10640
      T=XV(4)+STEP                                                      RKK10650
      CALL HAMLTN                                                       RKK10660
      XV(5)=T                                                           RKK10670
      DO 2051 I=1,NN                                                    RKK10680
      DEL=STEP*(9.*DYDT(I)+19.*FV(4,I)-5.*FV(3,I)+FV(2,I))/24.          RKK10690
      YU(5,I)=YU(4,I)+DEL                                               RKK10700
 2051 Y(I)=YU(5,I)                                                      RKK10710
      CALL HAMLTN                                                       RKK10720
      IF (MODE.LE.2) GO TO 42                                           RKK10730
C                                                                       RKK10740
C          ERROR ANALYSIS                                               RKK10750
      SSE=0.0                                                           RKK10760
      DO 3033 I=1,NN                                                    RKK10770
      EPSIL=R*ABS(Y(I)-DELY(1,I))                                       RKK10780
      IF (MODE.EQ.3.AND.Y(I).NE.0.) EPSIL=EPSIL/ABS(Y(I))               RKK10790
      IF (SSE.LT.EPSIL) SSE=EPSIL                                       RKK10800
 3033 CONTINUE                                                          RKK10810
      IF (E1MAX.GT.SSE) GO TO 3035                                      RKK10820
      IF (ABS(STEP).LE.E2MIN) GO TO 42                                  RKK10830
      LL=1                                                              RKK10840
      MM=1                                                              RKK10850
      STEP=STEP*FACT                                                    RKK10860
      GO TO 1001                                                        RKK10870
 3035 IF (LL.LE.1.OR.SSE.GE.E1MIN.OR.E2MAX.LE.ABS(STEP)) GO TO 42       RKK10880
      LL=2                                                              RKK10890
      MM=3                                                              RKK10900
      XV(2)=XV(3)                                                       RKK10910
      XV(3)=XV(5)                                                       RKK10920
      DO 5363 I=1,NN                                                    RKK10930
      FV(2,I)=FV(3,I)                                                   RKK10940
      FV(3,I)=DYDT(I)                                                   RKK10950
      YU(2,I)=YU(3,I)                                                   RKK10960
 5363 YU(3,I)=YU(5,I)                                                   RKK10970
      STEP=2.0*STEP                                                     RKK10980
      GO TO 1001                                                        RKK10990
C                                                                       RKK11000
C          EXIT ROUTINE                                                 RKK11010
   42 LL=2                                                              RKK11020
      MM=4                                                              RKK11030
      DO 12 K=1,3                                                       RKK11040
      XV(K)=XV(K+1)                                                     RKK11050
      DO 12 I=1,NN                                                      RKK11060
      FV(K,I)=FV(K+1,I)                                                 RKK11070
   12 YU(K,I)=YU(K+1,I)                                                 RKK11080
```

```
      XV(4)=XV(5)                                                        RKK11090
      DO 52 I=1,NN                                                       RKK11100
      FV(4,I)=DYDT(I)                                                    RKK11110
   52 YU(4,I)=YU(5,I)                                                    RKK11120
      IF (MODE.LE.2) RETURN                                              RKK11130
      E=ABS(XV(4)-ALPHA)                                                 RKK11140
      IF (E.LE.EPM+.25*STEP) GO TO 2000                                  RKK11150
      EPM=E                                                              RKK11160
      RETURN                                                             RKK11170
         END                                                             RKK11180
```

```
      SUBROUTINE BACKUP(Z,RSIGN,TC)                                      BAQP0020
C     MOVES THE RAY TO THE CLOSEST INTERSECTION WITH THE RECEIVER        BAQP0030
C     OR TERRAIN SURFACE(VARIABLES 'FSELECT' OR 'GSELECT' IN LABELED     BAQP0040
C     COMMONS /RR/ OR /GG/, RESPECTIVELY, TELL WHICH KIND OF SURFACE).   BAQP0050
C                                                                        BAQP0060
      CHARACTER*9 NBAK,NGRAZ,NTRY                                        BAQP0070
C                                                                        BAQP0080
      PARAMETER (PRNZTL=0.5E-4,PRNDZTL=1.E-6)                            BAQP0090
C                                                                        BAQP0100
      COMMON /RK/ N,STEP,MODE,E1MAX,E1MIN,E2MAX,E2MIN,FACT,RSTART        BAQP0110
C     COMMON DECK "TRAC" INSERTED HERE                                   CTRA0020
      LOGICAL GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY                     CTRA0040
      COMMON /TRAC/ GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY,SMT,OSMT      CTRA0050
      COMMON/TRAC/ROLD(20),DROLD(20),TOLD,ZDOT,D2Z,RAD,RAD1              CTRA0060
      COMMON R(20),T,STP,DRDT(20)                                        BAQP0130
      PARAMETER (NWARSZ=1000)                                            CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                    CWW10040
      EQUIVALENCE (EARTHR,W(1)),(INTYP,W(41)),(STEP1,W(44))              BAQP0150
      REAL KR                                                            BAQP0160
      EQUIVALENCE (KR,R(4)),(DKRDT,DRDT(4))                              BAQP0170
      REAL INTYP                                                         BAQP0180
      LOGICAL PCNTRL                                                     BAQP0190
      DATA NBAK,NGRAZ/' BACK UP0', ' GRAZE 1'/                           BAQP0200
C                                                                        BAQP0210
      NTRY=NBAK                                                          BAQP0220
      GO TO 100                                                          BAQP0230
C                                                                        BAQP0240
      ENTRY GRAZE(Z,RSIGN,TC)                                            BAQP0250
      NTRY=NGRAZ                                                         BAQP0260
C                                                                        BAQP0270
C     DEFINE BASE STEP SIZE                                              BAQP0280
  100 STPB=STP                                                           BAQP0290
C                                                                        BAQP0300
C     DEFINE BACKUP LOCATION TOLERANCES BASED ON INTEGRATION MODE        BAQP0310
C                                                                        BAQP0320
      IF(MODE.LT.3) THEN                                                 BAQP0330
         TOL=STEP1                                                       BAQP0340
      ELSEIF(MODE.EQ.3) THEN                                             BAQP0350
         TOL=ABS(E1MAX*STPB)                                             BAQP0360
      ELSE                                                               BAQP0370
```

```
         TOL=E1MAX                                                      BAQP0380
       ENDIF                                                            BAQP0390
C      USE THIS OR MINIMUM STEP SIZE WHICHEVER IS LESS                  BAQP0400
       TOL=AMIN1(TOL,FACT*E2MIN)                                        BAQP0410
C                                                                       BAQP0420
       PCNTRL=W(73).NE.0.0                                              BAQP0430
       THERE=.TRUE.                                                     BAQP0440
C       STEP TO ESTIMATED CROSSING AT TC.                               BAQP0450
C********* DIAGNOSTIC PRINTOUT                                         BAQP0460
       IF(PCNTRL) CALL PRINTR (NTRY,0.)                                 BAQP0470
       STEP=TC-T                                                        BAQP0480
       STEP=SIGN(AMIN1(ABS(STPB),ABS(STEP)),STEP)                       BAQP0490
       MODE=1                                                           BAQP0500
       RSTART=1                                                         BAQP0510
       TOLD=T                                                           BAQP0520
       CALL RMOVE(DROLD,DRDT,3)                                         BAQP0530
       ZDOLD=DOT(Z)                                                     BAQP0540
       CALL RKAM                                                        BAQP0550
       RSTART=1.0                                                       BAQP0560
       ZDOT=DOT(Z)                                                      BAQP0570
       IF(NTRY.EQ.NGRAZ .OR. RSIGN*ZDOT.LE.0.0) GO TO 12                BAQP0580
C                                                                       BAQP0590
C********* FIND NEAREST INTERSECTION OF RAY WITH THE HEIGHT HS          BAQP0600
       DO 10 I=1,10                                                     BAQP0610
       STEP=-Z/ZDOT                                                     BAQP0620
       STEP=SIGN(AMIN1(ABS(STPB),ABS(STEP)),STEP)                       BAQP0630
       IF (ABS(Z).LT.PRNZTL .AND. ABS(STEP).LT.TOL) GO TO 60            BAQP0640
C********* DIAGNOSTIC PRINTOUT                                         BAQP0650
       IF(PCNTRL) CALL PRINTR(' BACK UP1',0.)                           BAQP0660
       MODE=1                                                           BAQP0670
       RSTART=1.                                                        BAQP0680
       TOLD=T                                                           BAQP0690
       CALL RMOVE(DROLD,DRDT,3)                                         BAQP0700
       ZDOLD=ZDOT                                                       BAQP0710
       CALL RKAM                                                        BAQP0720
       ZDOT=DOT(Z)                                                      BAQP0730
   10 RSTART=1.                                                         BAQP0740
C                                                                       BAQP0750
C********* FIND NEAREST CLOSEST APPROACH OF RAY TO THE HEIGHT HS        BAQP0760
12     THERE=.FALSE.                                                    BAQP0770
       DO 20 I=1,10                                                     BAQP0780
C      DO 'LOCAL' PARABOLIC FIT                                         BAQP0790
       CALL FIT3(Z,ZOLD,ZDOLD)                                          BAQP0800
       STEP=-ZDOT/D2Z                                                   BAQP0810
       STEP=SIGN(AMIN1(ABS(STPB),ABS(STEP)),STEP)                       BAQP0820
       IF (ABS(ZDOT).LE.PRNDZTL .AND. ABS(STEP).LT.TOL) GO TO 60        BAQP0830
C********* DIAGNOSTIC PRINTOUT                                         BAQP0840
       IF(PCNTRL) CALL PRINTR (' GRAZE 1 ',0.)                          BAQP0850
       MODE=1                                                           BAQP0860
       RSTART=1.                                                        BAQP0870
       TOLD=T                                                           BAQP0880
       CALL RMOVE(DROLD,DRDT,3)                                         BAQP0890
       ZOLD=Z                                                           BAQP0900
       ZDOLD=ZDOT                                                       BAQP0910
       CALL RKAM                                                        BAQP0920
```

```
      RSTART=1.                                                          BAQP0930
      IF (D2Z*Z.LT.0.) GO TO 30                                          BAQP0940
      IF(KPARLEL(Z).EQ.0.0) GO TO 60                                     BAQP0950
   20 CONTINUE                                                           BAQP0960
      WRITE(3,350)                                                       BAQP0970
350   FORMAT(' ****** COULDN''T FIND CLOSEST APPROACH IN 10 STEPS')      BAQP0980
      GO TO 60                                                           BAQP0990
C                                                                        BAQP1000
   30 CONTINUE                                                           BAQP1010
C********* DIAGNOSTIC PRINTOUT                                          BAQP1020
      IF(PCNTRL) CALL PRINTR (' BACK UP2',0.)                            BAQP1030
      MODE=1                                                             BAQP1040
C********* ESTIMATE DISTANCE TO NEAREST INTERSECTION GOING THE RIGHT     BAQP1050
C********* DIRECTION OF RAY WITH HEIGHT HS                               BAQP1060
      CALL FIT3(Z,ZOLD,ZDOLD)                                            BAQP1070
      STEP=(-ZDOT+RSIGN*RAD1)/D2Z                                        BAQP1080
      RSTART=1.                                                          BAQP1090
      CALL RKAM                                                          BAQP1100
      RSTART=1.                                                          BAQP1110
C********* FIND NEAREST INTERSECTION OF RAY WITH HEIGHT HS               BAQP1120
      DO 40 I=1,10                                                       BAQP1130
      ZDOT=DOT(Z)                                                        BAQP1140
      STEP=-Z/ZDOT                                                       BAQP1150
      STEP=SIGN(AMIN1(ABS(STPB),ABS(STEP)),STEP)                         BAQP1160
      IF (ABS(Z).LT.PRNZTL .AND. ABS(STEP).LT.TOL) GO TO 50              BAQP1170
C********* DIAGNOSTIC PRINTOUT                                          BAQP1180
      IF(PCNTRL) CALL PRINTR (' BACK UP3',0.)                            BAQP1190
      MODE=1                                                             BAQP1200
      RSTART=1.                                                          BAQP1210
      CALL RKAM                                                          BAQP1220
   40 RSTART=1.                                                          BAQP1230
   50 THERE=.TRUE.                                                       BAQP1240
C********* RESET STANDARD MODE AND INTEGRATION TYPE                      BAQP1250
   60 MODE=INTYP                                                         BAQP1260
      STEP=STPB                                                          BAQP1270
      RETURN                                                             BAQP1280
         END                                                             BAQP1290
```

```
      FUNCTION REFLECT(Z)                                                REST0020
C     COMPUTES NORMAL AND PARALLEL COMPONENTS OF THE K-VECTOR AT          REST0030
C     REFLECTION POINTS TO A SURFACE. WIND EFFECTS ARE INCLUDED.          REST0040
C                                                                        REST0050
      REAL Z(12),KPARLEL,KNORM                                           REST0060
C     COMMON DECK "CC" INSERTED HERE                                     CCC 0020
      REAL MODC                                                          CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                         CCC 0050
C     COMMON DECK "WWR" INSERTED HERE                                    CWWR0020
      PARAMETER (NWARSZ=1000)                                            CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                    CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                 CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),     CWW20030
```

```
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),       CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),           CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),          CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                   CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                       CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                       CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                      CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),        CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))         CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),         CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))               CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                    CWW20160
C     COMMON DECK "RKAM" INSERTED HERE                                   RKAM0020
      REAL KR,KTH,KPH                                                    RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)        RKAM0050
C     COMMON DECK "UU" INSERTED HERE                                     CUU 0020
      REAL MODU                                                          CUU 0040
      COMMON/UU/MODU(4)                                                  CUU 0050
     1  ,V  ,PVT  ,PVR  ,PVTH  ,PVPH                                     CUU 0060
     2  ,VR ,PVRT ,PVRR ,PVRTH ,PVRPH                                    CUU 0070
     3  ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                   CUU 0080
     4  ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                   CUU 0090
C     COMMON DECK "FNDER" INSERTED HERE                                  CFND0020
      COMMON/FNDER/GY(20),NZ,NPZR,NPZRR,NPZRTH,NPZRPH,NPZTH              CFND0040
      COMMON/FNDER/NPZPH,NPZTHTH,NPZPHPH,NPZTHPH,NSELECT,NTIME           CFND0050
C                                                                        REST0120
      REAL NR,NTH,NPH,KRNR,KPARR,KPARTH,KPARPH                           REST0130
C                                                                        REST0140
      IENTRY=1                                                           REST0150
      REFLECT=0.0                                                        REST0160
      GO TO 5                                                            REST0170
C                                                                        REST0180
      ENTRY KPARLEL(Z)                                                   REST0190
      KPARLEL=0.0                                                        REST0200
      IENTRY=2                                                           REST0210
      GO TO 5                                                            REST0220
C                                                                        REST0230
      ENTRY KNORM(Z)                                                     REST0240
      KNORM=0.0                                                          REST0250
      IENTRY=3                                                           REST0260
C                                                                        REST0270
5     Z(1)=GET(Z)                                                        REST0280
      NR=Z(NPZR)                                                         REST0290
      NTH=Z(NPZTH)/R                                                     REST0300
      NPH=Z(NPZPH)/(R*SIN(TH))                                           REST0310
C                                                                        REST0320
      CALL RENORM(NR,1.0,3)                                              REST0330
C     COMPUTE THE NORMAL COMPONENT OF K-VECTOR TO SURFACE                REST0340
      KRNR=KR*NR+KTH*NTH+KPH*NPH                                         REST0350
      IF(IENTRY.NE.3) GO TO 8                                            REST0360
C     IF ENTRY 3 THEN WE ARE DONE                                        REST0370
      REFLECT=KRNR                                                       REST0380
      RETURN                                                             REST0390
C                                                                        REST0400
```

```
C       COMPUTE THE PARALLEL VECTOR COMPONENT                          REST0410
8       KPARR=KR-KRNR*NR                                               REST0420
        KPARTH=KTH-KRNR*NTH                                            REST0430
        KPARPH=KPH-KRNR*NPH                                            REST0440
C                                                                      REST0450
        IF(IENTRY.NE.2) GO TO 10                                       REST0460
        REFLECT=ABS(KPARR)+ABS(KPARTH)+ABS(KPARPH)                     REST0470
        RETURN                                                         REST0480
C                                                                      REST0490
10      OWIPAR=OW-KPARR*VR-KPARTH*VTH-KPARPH*VPH                       REST0500
        VNR=VR*NR+VTH*NTH+VPH*NPH                                      REST0510
C                                                                      REST0520
        FCTR=2.0*(KRNR+OWIPAR*VNR/(CS-VNR*VNR))                       REST0530
        KR=KR-FCTR*NR                                                  REST0540
        KTH=KTH-FCTR*NTH                                               REST0550
        KPH=KPH-FCTR*NPH                                               REST0560
        END                                                            REST0570
```

```
        SUBROUTINE FIT(Z,ZOLD,ZDOLD)                                   FIT 0020
C       COMPUTES THREE TYPES OF PARABOLIC FITS TO RAY PATH RELATIVE     FIT 0030
C       TO TERRAIN.                                                    FIT 0040
C                                                                      FIT 0050
        REAL Z(12)                                                     FIT 0060
C       COMMON DECK "RKAM" INSERTED HERE                               RKAM0020
        REAL KR,KTH,KPH                                                RKAM0040
        COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)    RKAM0050
C       COMMON DECK "TRAC" INSERTED HERE                               CTRA0020
        LOGICAL GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY                 CTRA0040
        COMMON /TRAC/ GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY,SMT,OSMT  CTRA0050
        COMMON/TRAC/ROLD(20),DROLD(20),TOLD,ZDOT,D2Z,RAD,RAD1          CTRA0060
C       COMMON DECK "FNDER" INSERTED HERE                              CFND0020
        COMMON/FNDER/GY(20),NZ,NPZR,NPZRR,NPZRTH,NPZRPH,NPZTH          CFND0040
        COMMON/FNDER/NPZPH,NPZTHTH,NPZPHPH,NPZTHPH,NSELECT,NTIME       CFND0050
C                                                                      FIT 0100
        REAL D2(3)                                                     FIT 0110
C                                                                      FIT 0120
C       USE FIT OF APPENDIX 'J' OF REPORT 'WPL-103' WHICH USES         FIT 0130
C       WEIGHTED ESTIMATE OF 1ST DERIVATIVE.                           FIT 0140
C                                                                      FIT 0150
        IENTRY=1                                                       FIT 0160
        GO TO 5                                                        FIT 0170
C                                                                      FIT 0180
C       USE MODIFIED FIT REQUIRING HEIGHTS OF PARABOLA VERTICES FROM   FIT 0190
C       CURRENT AND PREVIOUS RAY POINTS.                               FIT 0200
        ENTRY FIT2(Z,ZOLD,ZDOLD)                                       FIT 0210
        IENTRY=2                                                       FIT 0220
        GO TO 5                                                        FIT 0230
C                                                                      FIT 0240
C       USE FIT OF APPENDIX U(LOCAL VALUE OF 1ST DERIVATIVE)           FIT 0250
        ENTRY FIT3(Z,ZOLD,ZDOLD)                                       FIT 0260
        IENTRY=3                                                       FIT 0270
```

```
C                                                                       FIT 0280
5       ZDOT=DOT(Z)                                                     FIT 0290
C                                                                       FIT 0300
        DTI=1.0/(TPULSE-TOLD)                                           FIT 0310
        DO 10 I=1,3                                                     FIT 0320
10      D2(I)=(DRDT(I)-DROLD(I))*DTI                                    FIT 0330
C                                                                       FIT 0340
        D2Z=Z(NPZR)*D2(1)+Z(NPZTH)*D2(2)+Z(NPZPH)*D2(3)                 FIT 0350
       1         +Z(NPZRR)*DRDT(1)*DRDT(1)                              FIT 0360
       1         +Z(NPZTHTH)*DRDT(2)*DRDT(2)                            FIT 0370
       1         +Z(NPZPHPH)*DRDT(3)*DRDT(3)                            FIT 0380
       1    +2.0*(Z(NPZRTH)*DRDT(1)*DRDT(2)                             FIT 0390
       1         +Z(NPZRPH)*DRDT(1)*DRDT(3)                             FIT 0400
       1         +Z(NPZTHPH)*DRDT(2)*DRDT(3) )                          FIT 0410
C                                                                       FIT 0420
C       THE STATEMENTS FROM HERE TO 'END FIT' IMPLEMENT THE             FIT 0430
C       PARABOLIC FITS IN EQUATIONS J.1 AND U.3 OF THE TEXT.            FIT 0440
C                                                                       FIT 0450
        IF(IENTRY.NE.2) GO TO 30                                        FIT 0460
        SMT=0.                                                          FIT 0470
        IF (D2Z.NE.0.) SMT=0.5*ZDOT*ZDOT/D2Z                            FIT 0480
C       USE FIT U.3 AT THE PREVIOUS POINT OF RAY PATH                   FIT 0490
        OSMT=0.                                                         FIT 0500
        IF (D2Z.NE.0.) OSMT=0.5*ZDOLD*ZDOLD/D2Z                         FIT 0510
        GO TO 2000                                                      FIT 0520
C                                                                       FIT 0530
C       IMPLEMENTATION OF FIT FOR EQUATION J.1                          FIT 0540
C                                                                       FIT 0550
30      IF(IENTRY.EQ.3) GO TO 1000                                      FIT 0560
        ZDOTM=.5*(ZDOT+ZDOLD)                                           FIT 0570
C       IMPLEMENT TESTS OF EQUATIONS J.2 AND J.3                        FIT 0580
        IF(ABS(ZDOTM).LE..05*ABS(ZDOT)) GO TO 1000                      FIT 0590
        FCTR=(Z(NZ)-ZOLD)*DTI/ZDOTM                                     FIT 0600
        D2Z=FCTR*D2Z                                                    FIT 0610
        ZDOT=FCTR*ZDOT                                                  FIT 0620
C       END FIT                                                         FIT 0630
C                                                                       FIT 0640
C       COMMON CODE FOR FIT AND FIT3                                    FIT 0650
1000    RAD=ZDOT*ZDOT-2.0*Z(NZ)*D2Z                                     FIT 0660
        RAD1=SQRT(AMAX1(RAD,0.0))                                       FIT 0670
C                                                                       FIT 0680
2000    CONTINUE                                                        FIT 0690
C                                                                       FIT 0700
        END                                                             FIT 0710
```

```
        FUNCTION GET1(Z)                                                GET 0020
C       FUNCTIONALLY THE SAME AS 'GET' PROGRAM, SEE DOCUMENTATION THERE GET 0030
C       NEEDED BECAUSE RECEIVER MODELS WILL CALL GET TO OBTAIN TERRAIN VA GET 0040
C       BUT THEY THEMSELVES ARE CALLED VIA GET. HENCE HAVE RE-ENTRANCE  GET 0050
C       PROBLEM.                                                        GET 0060
C                                                                       GET 0070
```

```
      REAL Z(*)                                                          GET 0080
      REAL PF(10),PG(10)                                                 GET 0090
C     COMMON DECK "SS" INSERTED HERE                                     CSS 0020
      REAL MODSURF                                                       CSS 0040
      COMMON/SS/ MODSURF(4)                                              CSS 0050
      COMMON/SS/U,PUR,PURR,PURTH,PURPH                                   CSS 0060
      COMMON/SS/PUTH,PUPH,PUTHTH,PUPHPH,PUTHPH,USELECT,UTIME             CSS 0070
C     COMMON DECK "RKAM" INSERTED HERE                                   RKAM0020
      REAL KR,KTH,KPH                                                    RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)        RKAM0050
C     COMMON DECK "FNDER" INSERTED HERE                                  CFND0020
      COMMON/FNDER/GY(20),NZ,NPZR,NPZRR,NPZRTH,NPZRPH,NPZTH              CFND0040
      COMMON/FNDER/NPZPH,NPZTHTH,NPZPHPH,NPZTHPH,NSELECT,NTIME           CFND0050
C     COMMON DECK "RR" INSERTED HERE                                     CRR 0020
      REAL MODREC                                                        CRR 0040
      COMMON/RR/ MODREC(4)                                               CRR 0050
      COMMON/RR/F,PFR,PFRR,PFRTH,PFRPH                                   CRR 0060
      COMMON/RR/PFTH,PFPH,PFTHTH,PFPHPH,PFTHPH,FSELECT,FTIME             CRR 0070
C     COMMON DECK "GG" INSERTED HERE                                     CGG 0020
      REAL MODG                                                          CGG 0040
      COMMON/GG/MODG(4)                                                  CGG 0050
      COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                   CGG 0060
      COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME             CGG 0070
C     COMMON DECK "RKTIME" INSERTED HERE                                 CRKT0020
      COMMON/CRKTIME/RKTIME                                              CRKT0040
C                                                                        GET 0160
C     COMMON DECK "RMACH" INSERTED HERE                                  CRMA0020
      COMMON/CRMACH/RMACH(5)                                             CRMA0040
      EQUIVALENCE(RKTIME,IRKTIME)                                        GET 0180
C     COMMON DECK "WWR" INSERTED HERE                                    CWWR0020
      PARAMETER (NWARSZ=1000)                                            CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                    CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                 CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),     CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),       CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),           CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),          CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                  CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                       CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                       CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                      CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),        CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))         CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),         CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))               CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                    CWW20160
C                                                                        GET 0200
      COMMON/CGET/ZERO                                                   GET 0210
C                                                                        GET 0220
      IENTRY=1                                                           GET 0230
      GO TO 5                                                            GET 0240
C                                                                        GET 0250
      ENTRY DOT1(Z)                                                      GET 0260
      IENTRY=2                                                           GET 0270
```

```
5       CONTINUE                                                        GET 0280
        IF(ZERO.EQ.0.0) ZERO=EARTHR*RMACH(3)*2.0                        GET 0290
C                                                                       GET 0300
        IF(ITEST(Z(NTIME)).EQ.IRKTIME) GO TO 10                         GET 0310
C                                                                       GET 0320
        IF(Z(NSELECT).EQ.USELECT) CALL SURFACE                          GET 0330
        IF(Z(NSELECT).EQ.FSELECT) CALL RECEVER                          GET 0340
        IF(Z(NSELECT).EQ.GSELECT) CALL TOPOG                            GET 0350
        Z(NTIME)=RKTIME                                                 GET 0360
C       REMOVE MACHINE ROUND OFF NOISE FROM EXACT RECEIVER LOCATIONS    GET 0370
        IF(ABS(Z(NZ)).LE.ZERO) Z(NZ)=0.0                                GET 0380
C                                                                       GET 0390
10      IF(IENTRY.NE.1) GO TO 20                                        GET 0400
        GET1=Z(NZ)                                                      GET 0410
        RETURN                                                          GET 0420
C                                                                       GET 0430
20      GET1=Z(NPZR)*DRDT(1)+Z(NPZTH)*DRDT(2)+Z(NPZPH)*DRDT(3)          GET 0440
        RETURN                                                          GET 0450
        END                                                             GET 0460
```

```
        FUNCTION GET(Z)                                                 GET 0470
C       'GET' AND ENTRY 'DOT' PROVIDE A CONTROL METHOD FOR AVOIDING     GET 0480
C       REDUNDANT CALLS TO THE TERRAIN AND RECEIVER MODELS. THE VALUES  GET 0490
C       RETURNED ARE THE FUNCTION VALUES FOR 'F' OR 'G' OR THEIR DERIVATI GET 0500
C       (VIA 'DOT' ENTRY). UNNECESSARY CALLS ARE ELIMINATED THROUGH USE GET 0510
C       OF 'TIME OF CALL' VARIABLES WHICH ARE COMPARED WITH THE CURRENT GET 0520
C       LAST CALL TIME MAINTAINED BY THE 'RKAM' PROGRAM. WHEN VALUES ARE GET 0530
C       NOT CURRENT THEY ARE UPDATED BY CALLS TO THE APPROPRIATE ROUTINES GET 0540
C                                                                       GET 0550
        REAL Z(*),PF(10),PG(10)                                         GET 0560
C       COMMON DECK "SS" INSERTED HERE                                  CSS 0020
        REAL MODSURF                                                    CSS 0040
        COMMON/SS/ MODSURF(4)                                           CSS 0050
        COMMON/SS/U,PUR,PURR,PURTH,PURPH                                CSS 0060
        COMMON/SS/PUTH,PUPH,PUTHTH,PUPHPH,PUTHPH,USELECT,UTIME          CSS 0070
C       COMMON DECK "RKAM" INSERTED HERE                                RKAM0020
        REAL KR,KTH,KPH                                                 RKAM0040
        COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)     RKAM0050
C       COMMON DECK "FNDER" INSERTED HERE                               CFND0020
        COMMON/FNDER/GY(20),NZ,NPZR,NPZRR,NPZRTH,NPZRPH,NPZTH           CFND0040
        COMMON/FNDER/NPZPH,NPZTHTH,NPZPHPH,NPZTHPH,NSELECT,NTIME        CFND0050
C       COMMON DECK "RR" INSERTED HERE                                  CRR 0020
        REAL MODREC                                                     CRR 0040
        COMMON/RR/ MODREC(4)                                            CRR 0050
        COMMON/RR/F,PFR,PFRR,PFRTH,PFRPH                                CRR 0060
        COMMON/RR/PFTH,PFPH,PFTHTH,PFPHPH,PFTHPH,FSELECT,FTIME          CRR 0070
C       COMMON DECK "GG" INSERTED HERE                                  CGG 0020
        REAL MODG                                                       CGG 0040
        COMMON/GG/MODG(4)                                               CGG 0050
        COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                CGG 0060
        COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME          CGG 0070
```

```
C     COMMON DECK "RKTIME" INSERTED HERE                                CRKT0020
      COMMON/CRKTIME/RKTIME                                             CRKT0040
C                                                                       GET 0630
C     COMMON DECK "RMACH" INSERTED HERE                                 CRMA0020
      COMMON/CRMACH/RMACH(5)                                            CRMA0040
      EQUIVALENCE(RKTIME,IRKTIME)                                       GET 0650
C                                                                       GET 0660
C     COMMON DECK "WWR" INSERTED HERE                                   CWWR0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
C                                                                       GET 0680
      COMMON/CGET/ZERO                                                  GET 0690
C                                                                       GET 0700
      REAL PI(10),PJ(10)                                                GET 0710
      COMMON/TMP1/NC,NSZR    ,NSZRR    ,NSZRTH                          GET 0720
     1 ,NSZRPH    ,NSZTH    ,NSZPH    ,NSZTHTH    ,NSZPHPH    ,NSZTHPH  GET 0730
     2 ,NVELECT    ,NWIME                                               GET 0740
C                                                                       GET 0750
      DATA PI,PJ/1.0,9*0.0,1.0,9*0.0/                                   GET 0760
      DATA NC,NSZR,NSZRR,NSZRTH,NSZRPH,NSZTH,NSZPH,NSZTHTH,NSZPHPH      GET 0770
     1   ,NSZTHPH,NVELECT,NWIME/1,2,3,4,5,6,7,8,9,10,11,12/             GET 0780
      DATA FWIME,GWIME,FVELECT,GVELECT/2*-1.0,8HRECEIVER ,7HTERRAIN /   GET 0790
      DATA UWIME,UVELECT/-1.0,7HSURFACE /                               GET 0800
      DATA ZHRO/0.0/                                                    GET 0810
C                                                                       GET 0820
      IENTRY=1                                                          GET 0830
      GO TO 5                                                           GET 0840
C                                                                       GET 0850
      ENTRY SETGET                                                      GET 0860
C                                                                       GET 0870
      CALL RMOVE(PF,PI,10)                                              GET 0880
      CALL RMOVE(PG,PJ,10)                                              GET 0890
      CALL IMOVE(NZ,NC,12)                                              GET 0900
      FTIME=FWIME                                                       GET 0910
      GTIME=GWIME                                                       GET 0920
      FSELECT=FVELECT                                                   GET 0930
      GSELECT=GVELECT                                                   GET 0940
      UTIME=UWIME                                                       GET 0950
      USELECT=UVELECT                                                   GET 0960
      ZERO=ZHRO                                                         GET 0970
```

```
C                                                                       GET 0980
      RETURN                                                            GET 0990
C                                                                       GET 1000
      ENTRY DOT(Z)                                                      GET 1010
      IENTRY=2                                                          GET 1020
5     CONTINUE                                                          GET 1030
C                                                                       GET 1040
C     THE FOLLOWING THREE ASSIGNMENT STATEMENTS ARE NEEDED TO REPAIR    GET 1050
C     ARRAY ELEMENTS OVERWRITTEN ON THE CRAY VERSION OF THIS PROGRAM.   GET 1060
C     THE ADDITIONAL ARRAY 'GY' HAS ALSO BEEN ADDED TO THE /FNDER/      GET 1070
C     LABELED COMMON TO ACT AS A BUFFER AREA.                           GET 1080
C     NO CAUSE FOR THIS PROBLEM HAS YET BEEN FOUND. THESE STATEMENTS    GET 1090
C     ARE NOT NEEDED ON OTHER MACHINES AS FAR AS IS KNOWN.              GET 1100
      NZ=1                                                              GET 1110
      NPZR=2                                                            GET 1120
      NPZRR=3                                                           GET 1130
      IF(ZERO.EQ.0.0) ZERO=EARTHR*RMACH(3)*2.0                          GET 1140
C                                                                       GET 1150
      IF(ITEST(Z(NTIME)).EQ.IRKTIME) GO TO 10                           GET 1160
C                                                                       GET 1170
      IF(Z(NSELECT).EQ.USELECT) CALL SURFACE                            GET 1180
      IF(Z(NSELECT).EQ.FSELECT) CALL RECEVER                            GET 1190
      IF(Z(NSELECT).EQ.GSELECT) CALL TOPOG                              GET 1200
      Z(NTIME)=RKTIME                                                   GET 1210
C     REMOVE MACHINE ROUND OFF NOISE FROM EXACT RECEIVER LOCATIONS      GET 1220
      IF(ABS(Z(NZ)).LE.ZERO) Z(NZ)=0.0                                  GET 1230
C                                                                       GET 1240
10    IF(IENTRY.NE.1) GO TO 20                                          GET 1250
      GET=Z(NZ)                                                         GET 1260
      RETURN                                                            GET 1270
C                                                                       GET 1280
20    GET=Z(NPZR)*DRDT(1)+Z(NPZTH)*DRDT(2)+Z(NPZPH)*DRDT(3)             GET 1290
      RETURN                                                            GET 1300
      END                                                               GET 1310
```

```
      FUNCTION ITEST(I)                                                 ITRT0020
C     USED TO PASS INTEGER VALUES THROUGH FOR VARIABLES TYPED REAL      ITRT0030
C     (AS IN MIXED REAL/INTEGER ARRAYS)                                 ITRT0040
      ITEST=I                                                           ITRT0050
      END                                                               ITRT0060
```

```
      FUNCTION ITOC(N)                                                  ITOC0020
C     RETURN 7 CHARACTER STRING REPRESENTATION OF INTEGER N             ITOC0030
C     IF NUMBER IS TOO LARGE OR SMALL USE FLOATING POINT FORMAT         ITOC0040
      CHARACTER ITOC*7                                                  ITOC0050
      IF(N.LT.-9999.OR.N.GT.99999) GO TO 100                            ITOC0060
      ITOC=' '                                                          ITOC0070
```

```
      WRITE(ITOC,'(I7)',ERR=100) N                                          ITOC0080
      RETURN                                                                ITOC0090
100   WRITE(ITOC,'(2PG7.0)') FLOAT(N)                                       ITOC0100
      END                                                                   ITOC0110


      SUBROUTINE CONBLK                                                     COPK0020
C     DATA INITIALIZATION AND FILE OPENING SERVICE ROUTINE                  COPK0030
C                                                                           COPK0040
C     COMMON DECK "RMACH" INSERTED HERE                                     CRMA0020
      COMMON/CRMACH/RMACH(5)                                                CRMA0040
C     COMMON DECK "HDR" INSERTED HERE                                       CHDR0020
      CHARACTER*10 INITID*80,DAT,TOD                                        CHDR0040
      COMMON/HDR/SEC                                                        CHDR0050
      COMMON/HDRC/INITID,DAT,TOD                                            CHDR0060
C                                                                           COPK0070
C     COMMON DECK "CPROCFL" INSERTED HERE                                   CPRO0020
      INTEGER PMX,PNTBL,PITBL,PFRMTBL,IDSP(10)                              CPRO0040
C     PARAMETER DECK "PGROUPS"                                              PGRO0020
      PARAMETER (NCHPG1=11,NWPV=250,NSPGP=NCHPG1+2*NWPV+1)                  PGRO0030
      PARAMETER (MNGRP=9,MXGRP=69,MXLIST=MXGRP-MNGRP+2)                     PGRO0040
      COMMON/PROCFL/LIST(MXLIST)                                            CPRO0060
      COMMON/PROCFL/PMX,PNTBL(10),PITBL(10),PFRMTBL(10),PGP(NSPGP)          CPRO0070
      EQUIVALENCE (PGP,IDSP)                                                CPRO0080
C     COMMON DECK "CC" INSERTED HERE                                        CCC 0020
      REAL MODC                                                             CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                            CCC 0050
C     COMMON DECK "GG" INSERTED HERE                                        CGG 0020
      REAL MODG                                                             CGG 0040
      COMMON/GG/MODG(4)                                                     CGG 0050
      COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                      CGG 0060
      COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME                CGG 0070
C     COMMON DECK "RR" INSERTED HERE                                        CRR 0020
      REAL MODREC                                                           CRR 0040
      COMMON/RR/ MODREC(4)                                                  CRR 0050
      COMMON/RR/F,PFR,PFRR,PFRTH,PFRPH                                      CRR 0060
      COMMON/RR/PFTH,PFPH,PFTHTH,PFPHPH,PFTHPH,FSELECT,FTIME                CRR 0070
C     COMMON DECK "TT" INSERTED HERE                                        CTT 0020
      REAL MODT                                                             CTT 0040
      COMMON/TT/MODT(4), T,PTT,PTR,PTTH,PTPH                                CTT 0050
C     COMMON DECK "UU" INSERTED HERE                                        CUU 0020
      REAL MODU                                                             CUU 0040
      COMMON/UU/MODU(4)                                                     CUU 0050
     1   ,V   ,PVT   ,PVR   ,PVTH   ,PVPH                                  CUU 0060
     2   ,VR  ,PVRT  ,PVRR  ,PVRTH  ,PVRPH                                 CUU 0070
     3   ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                    CUU 0080
     4   ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                    CUU 0090
C                                                                           COPK0140
C     COMMON DECK "RAYDEV" INSERTED HERE                                    CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                                  CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN                   CRAY0050
C     COMMON DECK "CUCON" INSERTED HERE                                     CUCO0020
```

```
      COMMON/UCONV/CNVV(5,4)                                                CUCO0040
      CHARACTER PCV*3,CNVC*2                                                CUCO0050
      COMMON/UCONC/PCV(5),CNVC(5,4)                                         CUCO0060
C                                                                           COPK0170
C     COMMON DECK "SS" INSERTED HERE                                        CSS 0020
      REAL MODSURF                                                          CSS 0040
      COMMON/SS/ MODSURF(4)                                                 CSS 0050
      COMMON/SS/U,PUR,PURR,PURTH,PURPH                                      CSS 0060
      COMMON/SS/PUTH,PUPH,PUTHTH,PUPHPH,PUTHPH,USELECT,UTIME                CSS 0070
C     COMMON DECK "CONST" INSERTED HERE                                     CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                         CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                             CCON0050
C     COMMON DECK "WWR" INSERTED HERE                                       CWWR0020
      PARAMETER (NWARSZ=1000)                                               CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                       CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                    CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),        CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),          CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),              CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),             CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                     CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                          CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                          CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                         CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),           CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))            CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),            CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                  CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                       CWW20160
C     COMMON DECK "B9" INSERTED HERE                                        CB8 0020
      INTEGER GMX,GNTBL,GITBL,GFRMTBL,IDSG(10)                              CB8 0040
      COMMON/B9/GMX,GNTBL(10),GITBL(10),GFRMTBL(10),GGP(113)                CB8 0050
      EQUIVALENCE (GGP,IDSG),(ANG,GGP(11))                                  CB8 0060
C     COMMON DECK "B2" INSERTED HERE                                        CB2 0020
      INTEGER DUMX,DUNTBL,DUITBL,DUFRMTB,IDSDU(10)                          CB2 0040
      COMMON/B2/DUMX,DUNTBL(10),DUITBL(10),DUFRMTB(10),DUGP(10)             CB2 0050
      EQUIVALENCE (DUGP,IDSDU)                                              CB2 0060
C     COMMON DECK "B4" INSERTED HERE                                        CB4 0020
      INTEGER DCMX,DCNTBL,DCITBL,DCFRMTB,IDSDC(10)                          CB4 0040
      COMMON/B4/DCMX,DCNTBL(10),DCITBL(10),DCFRMTB(10),DCGP(10)             CB4 0050
      EQUIVALENCE (DCGP,IDSDC)                                              CB4 0060
C     COMMON DECK "B6" INSERTED HERE                                        CB6 0020
      INTEGER DTMX,DTNTBL,DTITBL,DTFRMTB,IDSDT(10)                          CB6 0040
      COMMON/B6/DTMX,DTNTBL(10),DTITBL(10),DTFRMTB(10),DTGP(10)             CB6 0050
      EQUIVALENCE (DTGP,IDSDT)                                              CB6 0060
C     COMMON DECK "B1" INSERTED HERE                                        CB1 0020
      INTEGER UMX,UNTBL,UITBL,UFRMTBL,IDSU(10)                              CB1 0040
      COMMON/B1/UMX,UNTBL(10),UITBL(10),UFRMTBL(10),UGP(10)                 CB1 0050
      EQUIVALENCE (UGP,IDSU)                                                CB1 0060
C     COMMON DECK "B3" INSERTED HERE                                        CB3 0020
      INTEGER CMX,CNTBL,CITBL,CFRMTBL,IDSC(10)                              CB3 0040
      COMMON/B3/CMX,CNTBL(10),CITBL(10),CFRMTBL(10),CGP(512)                CB3 0050
      EQUIVALENCE (CGP,IDSC),(ANC,CGP(11))                                  CB3 0060
C     COMMON DECK "B5" INSERTED HERE                                        CB5 0020
```

```
      INTEGER TMX,TNTBL,TITBL,TFRMTBL,IDST(10)                          CB5 0040
      COMMON/B5/TMX,TNTBL(10),TITBL(10),TFRMTBL(10),TGP(262)            CB5 0050
      EQUIVALENCE (TGP,IDST),(ANT,TGP(11))                              CB5 0060
C     COMMON DECK "B7" INSERTED HERE                                    CB7 0020
      INTEGER MMX,MNTBL,MITBL,MFRMTBL,IDSM(10)                          CB7 0040
      REAL MGP                                                          CB7 0050
      COMMON/B7/MMX,MNTBL(10),MITBL(10),MFRMTBL(10),MGP(10)             CB7 0060
      EQUIVALENCE (MGP,IDSM)                                            CB7 0070
C     COMMON DECK "RINPLEX" INSERTED HERE                               CRIN0020
      REAL KAY2,KAY2I                                                   CRIN0040
      COMPLEX PNP,POLAR,LPOLAR                                          CRIN0050
      LOGICAL SPACE                                                     CRIN0060
      CHARACTER DISPM*6                                                 CRIN0070
      COMMON/RINPL/DISPM                                                CRIN0080
      COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                 CRIN0090
      COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I                             CRIN0100
      COMMON/RIN/PNP(10),POLAR,LPOLAR,SGN                               CRIN0110
C     COMMON DECK "MM" INSERTED HERE                                    CMM 0020
      REAL M,MODM                                                       CMM 0040
      COMMON/MM/MODM(4),M,PMT,PMR,PMTH,PMPH                             CMM 0050
C     COMMON DECK "PP" INSERTED HERE                                    CPP 0020
      REAL MODP                                                         CPP 0040
      COMMON/PP/MODP(4),P,PPT,PPR,PPTH,PPPH                             CPP 0050
C     COMMON DECK "AA" INSERTED HERE                                    CAA 0020
      REAL MODA                                                         CAA 0040
      REAL MU,MUPT,MUPR,MUPTH,MUPPH                                     CAA 0050
      REAL KAP,KAPPT,KAPPR,KAPPTH,KAPPPH                                CAA 0060
      COMMON/AA/MODA(4),MU,MUPT,MUPR,MUPTH,MUPPH                        CAA 0070
      COMMON/AA/KAP,KAPPT,KAPPR,KAPPTH,KAPPPH                           CAA 0080
C     COMMON DECK "CB11" INSERTED HERE                                  CB110020
      INTEGER SMX,SNTBL,SITBL,SFRMTBL,IDSS(10)                          CB110040
      COMMON/B11/SMX,SNTBL(10),SITBL(10),SFRMTBL(10),SGP(11)            CB110050
      EQUIVALENCE (SGP,IDSS),(ANS,SGP(11))                              CB110060
C     COMMON DECK "CB12" INSERTED HERE                                  CB120020
      INTEGER DSMX,DSNTBL,DSITBL,DSFRMTB,IDSDS(10)                      CB120040
      COMMON/B12/DSMX,DSNTBL(10),DSITBL(10),DSFRMTB(10),DSGP(11)        CB120050
      EQUIVALENCE (DSGP,IDSDS),(ANDS,DSGP(11))                          CB120060
C     COMMON DECK "LL" INSERTED HERE                                    CLL 0020
      REAL MODL                                                         CLL 0040
      COMMON/LL/MODL(4),APH,APHPT,APHPR,APHPTH,APHPPH                   CLL 0050
C     COMMON DECK "FLAG" INSERTED HERE                                  CFLA0020
      LOGICAL NEWWR,NEWWP,NEWTRC,PENET                                  CFLA0040
      COMMON /FLG/ NTYP,NEWWR,NEWWP,NEWTRC,PENET,LINES,IHOP,HPUNCH      CFLA0050
      COMMON/FLGP/NSET                                                  CFLA0060
C     COMMON DECK "RKTIME" INSERTED HERE                                CRKT0020
      COMMON/CRKTIME/RKTIME                                             CRKT0040
C                                                                       COPK0380
      COMMON/RAYCON/MCONP                                               COPK0390
C                                                                       COPK0400
      EQUIVALENCE(RKTIME,IRKTIME)                                       COPK0410
C                                                                       COPK0420
C     COMMON DECK "B10" INSERTED HERE                                   CB9 0020
      INTEGER DGMX,DGNTBL,DGITBL,DGFRMTB,IDSDG(10)                      CB9 0040
      COMMON/B10/DGMX,DGNTBL(10),DGITBL(10),DGFRMTB(10),DGGP(10)        CB9 0050
      EQUIVALENCE (DGGP,IDSDG)                                          CB9 0060
```

```
C       COMMON DECK "B8" INSERTED HERE                                  CB100020
        INTEGER RMX,RNTBL,RITBL,RFRMTBL,IDSR(10)                        CB100040
        COMMON/B8/RMX,RNTBL(10),RITBL(10),RFRMTBL(10),RGP(10)           CB100050
        EQUIVALENCE (RGP,IDSR)                                          CB100060
C       COMMON DECK "CB17" INSERTED HERE                                CB170020
        INTEGER VMX,VNTBL,VITBL,VFRMTBL,IDSV(10)                        CB170040
        COMMON/B17/VMX,VNTBL(10),VITBL(10),VFRMTBL(10),VGP(53)          CB170050
        EQUIVALENCE (VGP,IDSV),(ANV,VGP(11))                           CB170060
C       COMMON DECK "CB18" INSERTED HERE                                CB180020
        INTEGER DVMX,DVNTBL,DVITBL,DVFRMTB,IDSDV(10)                    CB180040
        COMMON/B18/DVMX,DVNTBL(10),DVITBL(10),DVFRMTB(10),DVGP(11)      CB180050
        EQUIVALENCE (DVGP,IDSDV),(ANDV,DVGP(11))                       CB180060
C       COMMON DECK "CB19" INSERTED HERE                                CB190020
        INTEGER PRMX,PRNTBL,PRITBL,PRFRMTB,IDSPR(10)                    CB190040
        COMMON/B19/PRMX,PRNTBL(10),PRITBL(10),PRFRMTB(10),PRGP(11)      CB190050
        EQUIVALENCE (PRGP,IDSPR),(ANP,PRGP(11))                        CB190060
C       COMMON DECK "CB20" INSERTED HERE                                CB200020
        INTEGER DPMX,DPNTBL,DPITBL,DPFRMTB,IDSDP(10)                    CB200040
        COMMON/B20/DPMX,DPNTBL(10),DPITBL(10),DPFRMTB(10),DPGP(11)      CB200050
        EQUIVALENCE (DPGP,IDSDP),(ANDP,DPGP(11))                       CB200060
        COMMON/KNKN/KNBP,KNVC,KNDT                                      COPK0490
        COMMON/DDLIM/MXIX,MXIY,MNIX,MNIY                                COPK0500
C                                                                       COPK0510
        REAL KVECT(22)                                                  COPK0520
        REAL VSET(20)                                                   COPK0530
        EQUIVALENCE(KVECT,KAY2),(VSET,V)                                COPK0540
C                                                                       COPK0550
        REAL CUEF(3)                                                    COPK0560
        INTEGER NUYIND(5)                                               COPK0570
C                                                                       COPK0580
        CHARACTER PFV(5)*3,CQVC(5,4)*2                                  COPK0590
C                                                                       COPK0600
        DATA CUEF/1.0,1.4,8.31436E-3/                                   COPK0610
        DATA NUYIND/8,4,0,10,11/                                        COPK0620
        DATA PFV/'AN','LN','FQ','AM',' '/                               COPK0630
        DATA CQVC/'RD' ,'KM' ,'RD' ,'DB',' '                            COPK0640
       1         ,'DG' ,' M' ,'HZ' ,'NP',' '                            COPK0650
       2         ,'KM' ,'FT' ,'KH' ,2*' '                               COPK0660
       3         ,'  ' ,'NM' ,'DG' ,2*' '/                             COPK0670
C                                                                       COPK0680
        CALL SET2(KVECT,0.0,22)                                         COPK0690
        CALL SET2(RAYNAME,1H , 6)                                       COPK0700
        CALL SET2(VSET,0.0,20)                                          COPK0710
        CALL SET2(OMEGMIN,0.0,2)                                        COPK0720
        MXIX=-1000                                                      COPK0730
        MXIY=-1000                                                      COPK0740
        MNIX=1000                                                       COPK0750
        MNIY=1000                                                       COPK0760
        KNBP=0                                                          COPK0770
        KNVC=0                                                          COPK0780
        KNDT=0                                                          COPK0790
        IRKTIME=0                                                       COPK0800
C       SET TRACE-BACKUP VARIABLES                                      COPK0810
        CALL SETTRC                                                     COPK0820
        MCONP=0                                                         COPK0830
```

```
      POLAR=0.0                                                          COPK0840
      LPOLAR=1.0                                                         COPK0850
      CALL RMOVE(CREF,CUEF,3)                                            COPK0860
      CALL IMOVE(NRYIND,NUYIND,5)                                        COPK0870
      CALL SET2(CNVV,1.0,20)                                             COPK0880
      DO 10 I=1,5                                                        COPK0890
        PCV(I)=PFV(I)                                                    COPK0900
        DO 10 J=1,4                                                      COPK0910
10       CNVC(I,J)=CQVC(I,J)                                             COPK0920
C     DEFAULT WIND AND ABSORBSION IDENTIFIERS                            COPK0930
      MODU(1)=1H                                                         COPK0940
      MODU(2)=0.0                                                        COPK0950
      MODU(3)=1H                                                         COPK0960
      MODU(4)=0.0                                                        COPK0970
      MODL(1)=1H                                                         COPK0980
      MODL(2)=0.0                                                        COPK0990
      MODL(3)=1H                                                         COPK1000
      MODL(4)=0.0                                                        COPK1010
C                                                                        COPK1020
C     INITIALIZE MODEL DESCRIPTIONS TO BLANKS                            COPK1030
      CALL SET2(IDSDC,1H ,10)                                            COPK1040
      CALL SET2(IDSDT,1H ,10)                                            COPK1050
      CALL SET2(IDSDU,1H ,10)                                            COPK1060
      CALL SET2(IDSC,1H ,10)                                             COPK1070
      CALL SET2(IDSM,1H ,10)                                             COPK1080
      CALL SET2(IDST,1H ,10)                                             COPK1090
      CALL SET2(IDSU,1H ,10)                                             COPK1100
      CALL SET2(IDSG,1H ,10)                                             COPK1110
      CALL SET2(IDSDG,1H ,10)                                            COPK1120
      CALL SET2(IDSV,1H ,10)                                             COPK1130
      CALL SET2(IDSDV,1H ,10)                                            COPK1140
      CALL SET2(IDSPR,1H ,10)                                            COPK1150
      CALL SET2(IDSDP,1H ,10)                                            COPK1160
      CALL SET2(IDSR,1H ,10)                                             COPK1170
      CALL SET2(IDSS,1H ,10)                                             COPK1180
      CALL SET2(IDSDS,1H ,10)                                            COPK1190
C                                                                        COPK1200
C     GET MACHINE CONSTANTS                                              COPK1210
      CALL DFCNST(RMACH,5)                                               COPK1220
      RETURN                                                             COPK1230
C                                                                        COPK1240
      ENTRY STDINI                                                       COPK1250
      CALL OPNREP(NDEVTMP,'TAPE4')                                       COPK1260
      CALL OPNURP(NDEVBIN,'TAPE6')                                       COPK1270
      CALL OPNREP(9,'PUNCH')                                             COPK1280
C                                                                        COPK1290
      ENTRY STDINT                                                       COPK1300
      OPEN(UNIT=NRYIND,FILE='DINP',STATUS='OLD',ERR=1000)                COPK1310
      REWIND NRYIND                                                      COPK1320
      CALL OPNREP(2,'OUTPUT')                                            COPK1330
      CALL OPNREP(3,'DOUTP')                                             COPK1340
      CALL OPNURP(NDEVGRP,'TAPE5')                                       COPK1350
C                                                                        COPK1360
C  INITIALIZE RAYSET FILE                                                COPK1370
      READ(NRYIND,'(A)',END=1000) INITID                                 COPK1380
```

```
      CALL SETW                                                          COPK1390
      CALL SETGET                                                        COPK1400
      CALL SYSDAT(DAT)                                                   COPK1410
      CALL SYSTIM(TOD)                                                   COPK1420
      CALL SYSSEC(SEC)                                                   COPK1430
C     INITIALIZE POINT MODEL LIST                                        COPK1440
      LIST(1)=1                                                          COPK1450
      LIST(2)=0                                                          COPK1460
C     FILL FORMAT CONTROL ARRAYS                                         COPK1470
      PNTBL(1)=1                                                         COPK1480
C     ALLOW FOR AN 80 CHARACTER IDENT STRING(A8)                         COPK1490
      PNTBL(2)=NCHPG1                                                    COPK1500
C     ALLOW MAXIMUM 250 WORD PER VARIABLE                                COPK1510
      PITBL(2)=NWPV                                                      COPK1520
C     FOR BOTH X AND Y PLUS 1 FOR NUMBER OF VARIABLES                    COPK1530
      PNTBL(3)=NSPGP                                                     COPK1540
      RETURN                                                             COPK1550
1000  STOP 'DINP FORMAT ERROR'                                           COPK1560
      END                                                                COPK1570
```

```
      SUBROUTINE SETTRC                                                  COPK1580
C     INITIALIZE /TRAC/ LABELED COMMON                                   COPK1590
C     COMMON DECK "TRAC" INSERTED HERE                                   CTRA0020
      LOGICAL GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY                     CTRA0040
      COMMON /TRAC/ GROUND,SURF,PERIGE,THERE,MINDIS,NEWRAY,SMT,OSMT      CTRA0050
      COMMON/TRAC/ROLD(20),DROLD(20),TOLD,ZDOT,D2Z,RAD,RAD1              CTRA0060
      GROUND=.FALSE.                                                     COPK1610
      SURF=.FALSE.                                                       COPK1620
      END                                                                COPK1630
```

```
      LOGICAL FUNCTION WCHANGE(W1,W2)                                    WCZE0020
      REAL W1(400),W2(400)                                               WCZE0030
C                                                                        WCZE0040
C WCHANGE PROVIDES A TEST AGAINST TWO W-ARRAYS                           WCZE0050
C     IF BOTH WOULD PRODUCE THE SAME SET OF RAYS BY RAYTRC THEN RESULT   WCZE0060
C     IS <FALSE>.                                                        WCZE0070
C                                                                        WCZE0080
      INTEGER NDX(8)                                                     WCZE0090
C                                                                        WCZE0100
      DATA NGRPS,NDX/8, 1,17, 21,26, 41,47, 100,399/                     WCZE0110
C                                                                        WCZE0120
      WCHANGE=.FALSE.                                                    WCZE0130
      DO 20 N=1,NGRPS,2                                                  WCZE0140
         N1=NDX(N)                                                       WCZE0150
         N2=NDX(N+1)                                                     WCZE0160
      DO 20 I=N1,N2                                                      WCZE0170
          WCHANGE=W1(I).NE.W2(I)                                         WCZE0180
```

```
          IF(WCHANGE) RETURN                                                    WCZE0190
20        CONTINUE                                                              WCZE0200
          END                                                                   WCZE0210
```

```
          FUNCTION RENORM(VECTOR,NNORM,NCOMPS)                                  RELM0020
C         NORMALIZES 'NCOMPS' COMPONENT VECTOR 'VECTOR' TO MAGNITUDE            RELM0030
C         'NNORM' AND RETURNS SQUARE ROOT OF FACTOR NEEDED.                     RELM0040
          REAL NNORM,VECTOR(*)                                                  RELM0050
C                                                                               RELM0060
          RENORM=0.0                                                            RELM0070
          IF(NNORM.LE.0.0) RETURN                                               RELM0080
C                                                                               RELM0090
          RENORM=0.0                                                            RELM0100
          DO 10 I=1,NCOMPS                                                      RELM0110
10        RENORM=RENORM+VECTOR(I)*VECTOR(I)                                     RELM0120
          IF(RENORM.EQ.0.0) RETURN                                              RELM0130
C                                                                               RELM0140
          RENORM=SQRT(NNORM/RENORM)                                             RELM0150
          DO 20 I=1,NCOMPS                                                      RELM0160
20        VECTOR(I)=VECTOR(I)*RENORM                                            RELM0170
C                                                                               RELM0180
          RETURN                                                                RELM0190
          END                                                                   RELM0200
```

```
          SUBROUTINE SET2(A,V,N)                                                SET20020
C         SETS N COMPONENTS OF VECTER TO SINGLE VALUE V                         SET20030
          ENTRY ISET2(A,V,N)                                                    SET20040
C         SPECIAL ENTRY FOR INTEGER ARRAYS FOR MACHINES HAVING DIFFERENT        SET20050
C         WORD SIZES FOR INTEGERS THAN REALS.                                   SET20060
C                                                                               SET20070
          REAL A(N)                                                             SET20080
C                                                                               SET20090
          DO 100 I=1,N                                                          SET20100
100       A(I)=V                                                                SET20110
          END                                                                   SET20120
```

```
          SUBROUTINE PRINTR(EVENT,CARD)                                         PRVR0020
C                                                                               PRVR0030
          CHARACTER EVENT*9,NWHY*8,CC*1,PC*1,TMP*9                              PRVR0040
C                                                                               PRVR0050
C         PRINTS OUTPUT AND OUTPUTS RAYSETS(MACHINE READABLE OUTPUT)            PRVR0060
C         WHEN 'CARD' ARGUMENT NONZERO.                                         PRVR0070
          REAL KNORM                                                            PRVR0080
```

```
      DIMENSION G0(3,3),G1(3,3)                                          PRVR0090
      CHARACTER*12 HEADRS(20),HEAD(20),UNITS(20),UNIT(20)                PRVR0100
      DIMENSION RPRINT(20),NPR(20)                                       PRVR0110
C     COMMON DECK "RK" INSERTED HERE                                     CRK 0020
C     DEFINE SIZE REQUIRED FOR RAY STATE SAVE ARRAY                      CRK 0040
      PARAMETER (LRKAMS=87+2*100,NXRKMS=12+LRKAMS,MXEQPT=21)             CRK 0050
      PARAMETER (NRKSAV=NXRKMS+MXEQPT-1)                                 CRK 0060
      COMMON /RK/ NEQS,STEP,MODE,E1MAX,E1MIN,E2MAX,E2MIN,FACT,RSTART     CRK 0070
C     COMMON DECK "CERR" INSERTED HERE                                   CERR0020
      COMMON/ERR/NERG,NERR,NERT,NERP                                     CERR0030
C     COMMON DECK "GG" INSERTED HERE                                     CGG 0020
      REAL MODG                                                          CGG 0040
      COMMON/GG/MODG(4)                                                  CGG 0050
      COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                   CGG 0060
      COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME             CGG 0070
C     COMMON DECK "CONST" INSERTED HERE                                  CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                      CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                          CCON0050
C     COMMON DECK "FLAG" INSERTED HERE                                   CFLA0020
      LOGICAL NEWWR,NEWWP,NEWTRC,PENET                                   CFLA0040
      COMMON /FLG/ NTYP,NEWWR,NEWWP,NEWTRC,PENET,LINES,IHOP,HPUNCH       CFLA0050
      COMMON/FLGP/NSET                                                   CFLA0060
C     COMMON DECK "RINPLEX" INSERTED HERE                                CRIN0020
      REAL KAY2,KAY2I                                                    CRIN0040
      COMPLEX PNP,POLAR,LPOLAR                                           CRIN0050
      LOGICAL SPACE                                                      CRIN0060
      CHARACTER DISPM*6                                                  CRIN0070
      COMMON/RINPL/DISPM                                                 CRIN0080
      COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                  CRIN0090
      COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I                              CRIN0100
      COMMON/RIN/PNP(10),POLAR,LPOLAR,SGN                                CRIN0110
C     COMMON DECK "RKAM" INSERTED HERE                                   RKAM0020
      REAL KR,KTH,KPH                                                    RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)        RKAM0050
C     COMMON DECK "WW" INSERTED HERE                                     CWW 0020
      PARAMETER (NWARSZ=1000)                                            CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                    CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                 CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),     CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),       CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),           CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),          CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                  CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                       CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                       CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                      CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),        CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))         CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),         CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))               CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                    CWW20160
      REAL MMODEL,MFORM,MID                                              CWW30020
C                                                                        CWW30030
C     WIND        100-124                                              CWW30040
```

```
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)
C
C     DELTA WIND         125-149
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)
C
C     SOUND SPEED 150-174
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)
      EQUIVALENCE (W(153),REFC)
C
C     DELTA SOUND SPEED 175-199
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)
C
C     TEMPERATURE    200-224
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)
C
C     DELTA TEMPERATURE    225-249
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)
C
C     MOLECULAR   250-274
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)
C
C     RECEIVER HEIGHT   275-299
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)
C
C     TOPOGRAPHY   300-324
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)
C
C     DELTA TOPOGRAPHY   325-349
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)
C
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)
C
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)
C     PLOT ENHANCEMENTS CONTROL PARAMETERS
C
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)
C     ABSORPTION         500-524
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)
C
C     DELTA ABSORPTION         525-549
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)
C
C     PRESSURE         550-574
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)
C
C     DELTA PRESSURE         575-599
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)
C
      EQUIVALENCE(T,TPULSE),(PHI,PH),(TH,THETA)
      DATA
     2  HEADRS(7)/'  PHASE TIME'/,UNITS(7)/'    SEC'/
     3 ,HEADRS(8)/'  ABSORPTION'/,UNITS(8)/'    DB '/
     4 ,HEADRS(9)/'   DOPPLER  '/,UNITS(9)/'  C/,S '/
```

CWW30050
CWW30060
CWW30070
CWW30080
CWW30090
CWW30100
CWW30110
CWW30120
CWW30130
CWW30140
CWW30150
CWW30160
CWW30170
CWW30180
CWW30190
CWW30200
CWW30210
CWW30220
CWW30230
CWW30240
CWW30250
CWW30260
CWW30270
CWW30280
CWW30290
CWW30300
CWW30310
CWW30320
CWW30330
CWW30340
CWW30350
CWW30360
CWW30370
CWW30380
CWW30390
CWW30400
CWW30410
CWW30420
CWW30430
CWW30440
CWW30450
CWW30460
CWW30470
CWW30480
CWW30490
CWW30500
CWW30510
CWW30520
CWW30530
CWW30540
PRVR0200
PRVR0210
PRVR0220
PRVR0230
PRVR0240

```
     5 ,HEADRS(10)/' PATH LENGTH'/,UNITS(10)/'   KM '/                  PRVR0250
     6 ,HEADRS(11)/' TERRAIN'/,UNITS(11)/' '/                           PRVR0260
     7 ,HEADRS(12)/' TERRAIN PGR'/,UNITS(12)/' '/                       PRVR0270
     8 ,HEADRS(13)/' TERRAIN PGT'/,UNITS(13)/' '/                       PRVR0280
     9 ,HEADRS(14)/' TERRAIN PGP'/,UNITS(14)/' '/                       PRVR0290
C                                                                       PRVR0300
C      ROUND-OFF FUNCTION                                               PRVR0310
       ROUND(X)=SIGN(ABS(X)+0.5,X)                                      PRVR0320
C                                                                       PRVR0330
       NWHY=EVENT(2:)                                                   PRVR0340
       GO TO 10                                                         PRVR0350
C                                                                       PRVR0360
C      INITIALIZATION ENTRY POINT FOR PRINTR. CALLED FOR EACH NEW       PRVR0370
C      W-ARRAY.                                                         PRVR0380
       ENTRY IPRINTR                                                    PRVR0390
C                                                                       PRVR0400
C********* NEW W ARRAY -- REINITIALIZE                                  PRVR0410
       NEWWP=.FALSE.                                                    PRVR0420
       SPL=SIN (PLON-TLON)                                              PRVR0430
       CPL=SIN (PID2-(PLON-TLON))                                       PRVR0440
       SP=SIN (PLAT)                                                    PRVR0450
       CP=SIN (PID2-PLAT)                                               PRVR0460
       SL=SIN (TLAT)                                                    PRVR0470
       CL=SIN (PID2-TLAT)                                               PRVR0480
C********* MATRIX TO ROTATE COORDINATES                                 PRVR0490
       G0(1,1)=CPL*SP*CL-CP*SL                                          PRVR0500
       G0(1,2)=SPL*SP                                                   PRVR0510
       G0(1,3)=-SL*SP*CPL-CL*CP                                         PRVR0520
       G0(2,1)=-SPL*CL                                                  PRVR0530
       G0(2,2)=CPL                                                      PRVR0540
       G0(2,3)=SL*SPL                                                   PRVR0550
       G0(3,1)=CL*CP*CPL+SP*SL                                          PRVR0560
       G0(3,2)=CP*SPL                                                   PRVR0570
       G0(3,3)=-SL*CP*CPL+SP*CL                                         PRVR0580
       DENM=G0(1,1)*G0(2,2)*G0(3,3)+G0(1,2)*G0(3,1)*G0(2,3)             PRVR0590
     1   +G0(2,1)*G0(3,2)*G0(1,3)-G0(2,2)*G0(3,1)*G0(1,3)               PRVR0600
     2   -G0(1,2)*G0(2,1)*G0(3,3)-G0(1,1)*G0(3,2)*G0(2,3)               PRVR0610
C********* THE MATRIX G1 IS THE INVERSE OF THE MATRIX G                 PRVR0620
       G1(1,1)=(G0(2,2)*G0(3,3)-G0(3,2)*G0(2,3))/DENM                   PRVR0630
       G1(1,2)=(G0(3,2)*G0(1,3)-G0(1,2)*G0(3,3))/DENM                   PRVR0640
       G1(1,3)=(G0(1,2)*G0(2,3)-G0(2,2)*G0(1,3))/DENM                   PRVR0650
       G1(2,1)=(G0(3,1)*G0(2,3)-G0(2,1)*G0(3,3))/DENM                   PRVR0660
       G1(2,2)=(G0(1,1)*G0(3,3)-G0(3,1)*G0(1,3))/DENM                   PRVR0670
       G1(2,3)=(G0(2,1)*G0(1,3)-G0(1,1)*G0(2,3))/DENM                   PRVR0680
       G1(3,1)=(G0(2,1)*G0(3,2)-G0(3,1)*G0(2,2))/DENM                   PRVR0690
       G1(3,2)=(G0(3,1)*G0(1,2)-G0(1,1)*G0(3,2))/DENM                   PRVR0700
       G1(3,3)=(G0(1,1)*G0(2,2)-G0(2,1)*G0(1,2))/DENM                   PRVR0710
       R0=EARTHR+XMTRH                                                  PRVR0720
C********* CARTESIAN COORDINATES OF TRANSMITTER                         PRVR0730
       XR=R0*G0(1,1)                                                    PRVR0740
       YR=R0*G0(2,1)                                                    PRVR0750
       ZR=R0*G0(3,1)                                                    PRVR0760
       CTHR=G0(3,1)                                                     PRVR0770
       STHR=SIN (ACOS (CTHR))                                           PRVR0780
       PHIR=ATAN2(YR,XR)                                                PRVR0790
```

```
      ALPH=ATAN2(G0(3,2),G0(3,3))                                       PRVR0800
C*********                                                              PRVR0810
      NR=0                                                              PRVR0820
      NP=0                                                              PRVR0830
      NERG=0                                                            PRVR0840
      NERR=0                                                            PRVR0850
      NERT=0                                                            PRVR0860
      NERP=0                                                            PRVR0870
C     INSURE NO GARBLE IN HEADERS                                       PRVR0880
      HEAD(1)=' '                                                       PRVR0890
      UNIT(1)=' '                                                       PRVR0900
C                                                                       PRVR0910
      DO 7 NN=7,20                                                      PRVR0920
      IF (W(NN+50).EQ.0.) GO TO 7                                       PRVR0930
C********* DEPENDENT VARIABLE NUMBER NN IS BEING INTEGRATED             PRVR0940
C********* NR IS THE NUMBER OF DEPENDENT VARIABLES BEING INTEGRATED     PRVR0950
      NR=NR+1                                                           PRVR0960
C     ENABLE SELECTED RELATIVE ERROR PRINTOUTS FOR TERRAIN(G) OR        PRVR0970
C     ITS DERIVATIVES WITH RESPECT TO (R)ANGE, (T)HETA OR (P)HI.        PRVR0980
      IF(NN.EQ.11) NERG=NR                                              PRVR0990
      IF(NN.EQ.12) NERR=NR                                              PRVR1000
      IF(NN.EQ.13) NERT=NR                                              PRVR1010
      IF(NN.EQ.14) NERP=NR                                              PRVR1020
      IF (W(NN+50).NE.2.) GO TO 7                                       PRVR1030
C********* DEPENDENT VARIABLE NUMBER NN IS BEING INTEGRATED AND PRINTED.PRVR1040
C********* NP IS THE NUMBER OF DEPENDENT VARIABLES BEING INTEGRATED AND PRVR1050
C********* PRINTED                                                     PRVR1060
      NP=NP+1                                                           PRVR1070
C********* SAVE THE INDEX OF THE DEPENDENT VARIABLE TO PRINT           PRVR1080
      NPR(NP)=NR                                                        PRVR1090
      HEAD(NP)=HEADRS(NN)                                               PRVR1100
      HEAD(NP)=HEADRS(NN)                                               PRVR1110
      UNIT(NP)=UNITS(NN)                                                PRVR1120
    7 CONTINUE                                                          PRVR1130
      NPM=MIN0(NP,3)                                                    PRVR1140
      NP1=NPM+1                                                         PRVR1150
      P=0.0                                                             PRVR1160
      ABSORB=0.0                                                        PRVR1170
      DOPP=0.0                                                          PRVR1180
      NEQS=NR+6                                                         PRVR1190
      RETURN                                                            PRVR1200
C                                                                       PRVR1210
      ENTRY PRNHD1(CC)                                                  PRVR1220
C     PRINT PRINTR HEADER 1                                             PRVR1230
C   IF NO OUTPUT NEEDED EXIT NOW                                        PRVR1240
      IF(PRTSRP.NE.0.0) RETURN                                          PRVR1250
C                                                                       PRVR1260
C     ADD NUMBER OF LINES NEEDED FOR THIS HEADER                        PRVR1270
      PC=CC                                                             PRVR1280
      IF(CC.EQ.'1') CALL NEWPAG(NPAG,INT(PAGLN),PC)                     PRVR1290
      LINES=LINES+3                                                     PRVR1300
C********* PRINT COLUMN HEADINGS                                       PRVR1310
C                                                                       PRVR1320
      WRITE(3,1100) PC,(HEAD(NN),NN=1,NPM)                              PRVR1330
C                                                                       PRVR1340
```

```
1100  FORMAT (   A,T25,'ELEVATION',T54,'AZIMUTH',T71,'ELEVATION'/        PRVR1350
     3 ,T20,2('ABOVE',6X),T53,'DEVIATION',T72,'ANGLE'/'  ERROR   EVENT' PRVR1360
     4  ,T20,'SEA LEVEL  TERRAIN     RANGE',4X,2('XMTR   LOCAL',5X),    PRVR1370
     5  'PULSE TIME',3A12)                                              PRVR1380
C                                                                       PRVR1390
      WRITE(3,1150) (UNIT(NN),NN=1,NPM)                                 PRVR1400
 1150 FORMAT(13X,3(8X,'KM'),2X,2(6X,'DEG',5X,'DEG'),T88                 PRVR1410
     1 ,'SEC',3X,3(3X,A7,2X))                                          PRVR1420
C                                                                       PRVR1430
      RETURN                                                            PRVR1440
C                                                                       PRVR1450
      ENTRY PRNHD2(CC)                                                  PRVR1460
C     PRINT PRINTR HEADER 1                                             PRVR1470
C   IF NO OUTPUT NEEDED EXIT NOW                                        PRVR1480
      IF(PRTSRP.NE.0.0) RETURN                                          PRVR1490
C     PAGE BY HALF LENGTH                                               PRVR1500
      LINSPP=PAGLN/2                                                    PRVR1510
      IF(LINSPP.LT.40) LINSPP=PAGLN                                     PRVR1520
      PC=CC                                                             PRVR1530
      IF(CC.EQ.'1') CALL NEWPAG(NPAG,LINSPP,PC)                         PRVR1540
C     ADD NUMBER OF LINES NEEDED FOR THIS SUBHEADER                     PRVR1550
      LINES=LINES+1                                                     PRVR1560
C                                                                       PRVR1570
      IF(ELEND.GE.ELBEG+ELSTEP) THEN                                    PRVR1580
         WRITE(3,'(A,''ELEVATION ANGLE OF TRANSMISSION ='''             PRVR1590
     1    ,F10.4,'' DEG'')') PC,BETA*DEGS                              PRVR1600
      ELSEIF(AZEND.GE.AZBEG+AZSTEP) THEN                                PRVR1610
         WRITE(3,'(A,''AZIMUTH ANGLE OF TRANSMISSION ='''               PRVR1620
     1    ,F10.4,'' DEG'')') PC,AZ1*DEGS                               PRVR1630
      ENDIF                                                             PRVR1640
      RETURN                                                            PRVR1650
C                                                                       PRVR1660
C     IF PRINTING SUPPRESSED AND RAYSETS OFF NOTHING TO DO              PRVR1670
10    IF(PRTSRP.NE.0.0 .AND. CARD.EQ.0.0) RETURN                        PRVR1680
      CALL DISPER                                                       PRVR1690
      IF (CARD.EQ.0.0 .OR. IHOP.NE.0) GO TO 12                          PRVR1700
C********* OUTPUT A TRANSMITTER RAYSET                                 PRVR1710
C        NOTE: THIS IS A SPECIAL CASE, ALL OTHER RAY EVENTS ARE         PRVR1720
C           OUTPUT AT CODE 'PUNCH A RAYSET' BELOW.                      PRVR1730
C                                                                       PRVR1740
      TLOND=TLON*DEGS                                                   PRVR1750
      IF (TLOND.LT.0.) TLOND=TLOND+360.                                 PRVR1760
      TLATD=TLAT*DEGS                                                   PRVR1770
      IF (TLATD.LT.0.) TLATD=TLATD+360.                                 PRVR1780
      AZ=AZ1*DEGS                                                       PRVR1790
      EL=BETA*DEGS                                                      PRVR1800
NHOP=HOP                                                                PRVR1810
NXMTRH=ROUND(XMTRH*1.E4)                                                PRVR1820
NTLATD=ROUND(TLATD*1.E3)                                                PRVR1830
 NTLOND=ROUND(TLOND*1.E3)                                               PRVR1840
NRCVRH=ROUND(RCVRH*1.E4)                                                PRVR1850
 NF=ROUND(OW*1.E4)                                                      PRVR1860
NAZ=ROUND(AZ*1.E5)                                                      PRVR1870
 NEL=ROUND(EL*1.E5)                                                     PRVR1880
NPOLAR1=ROUND(REAL(POLAR)*1.E2)                                         PRVR1890
```

```
       NPOLAR2=ROUND(AIMAG(POLAR)*1.E2)                                  PRVR1900
      WRITE(9,1201) ID(1),TYPE(NTYP),NXMTRH,NTLATD,NTLOND,NRCVRH,NF,NAZ  PRVR1910
     1 ,NEL,NPOLAR1,NPOLAR2,NHOP,'T'                                     PRVR1920
 1201 FORMAT(A3,A1,I9,2I6,2I9,2I10,5X,I5,I4,I2,A1)                       PRVR1930
C*********                                                               PRVR1940
12    V=1.E10                                                           PRVR1950
C     OBTAIN THE WORST ERROR OF THOSE ENABLED.                          PRVR1960
      IF (KAY2.NE.0.) V=(KR**2+KTH**2+KPH**2)/KAY2-1.                   PRVR1970
      ERT=1HK                                                           PRVR1980
      V=RERR(V,ERT,'G',NERG,G)                                          PRVR1990
      V=RERR(V,ERT,'R',NERR,PGR)                                        PRVR2000
      V=RERR(V,ERT,'T',NERT,PGTH)                                       PRVR2010
      V=RERR(V,ERT,'P',NERP,PGPH)                                       PRVR2020
C                                                                       PRVR2030
      H=R-EARTHR                                                        PRVR2040
      STH=SIN (THETA)                                                   PRVR2050
      CTH=SIN (PID2-THETA)                                              PRVR2060
C********* CARTESIAN COORDINATES OF RAY POINT, ORIGIN AT TRANSMITTER    PRVR2070
      XP=R*STH*SIN (PID2-PHI)-XR                                        PRVR2080
      YP=R*STH*SIN (PHI)-YR                                             PRVR2090
      ZP=R*CTH-ZR                                                       PRVR2100
C********* CARTESIAN COORDINATES OF RAY POINT, ORIGIN AT TRANSMITTER ANDPRVR2110
C********* ROTATED                                                     PRVR2120
      EPS=XP*G1(1,1)+YP*G1(1,2)+ZP*G1(1,3)                              PRVR2130
      ETA=XP*G1(2,1)+YP*G1(2,2)+ZP*G1(2,3)                              PRVR2140
      ZETA=XP*G1(3,1)+YP*G1(3,2)+ZP*G1(3,3)                             PRVR2150
      RCE2=ETA**2+ZETA**2                                               PRVR2160
      RCE=SQRT (RCE2)                                                   PRVR2170
C********* GROUND RANGE                                                PRVR2180
      RANGE=EARTHR*ATAN2(RCE,EARTHR+EPS+XMTRH)                          PRVR2190
C********* ANGLE OF WAVE NORMAL WITH LOCAL HORIZONTAL                  PRVR2200
      ELL=ATAN2(KR,SQRT (KTH**2+KPH**2))*DEGS                           PRVR2210
C********* STRAIGHT LINE DISTANCE FROM TRANSMITTER TO RAY POINT        PRVR2220
      SR=SQRT (RCE2+EPS**2)                                             PRVR2230
C********* TERRAIN RELATIVE HEIGHT                                     PRVR2240
      GRH=GET(G)/PGR                                                    PRVR2250
C     REPORT GROUP TIME AS FIRST 'OPTION'                               PRVR2260
      RPRINT(1)=T                                                       PRVR2270
      IF (NP1.LT.2) GO TO 16                                            PRVR2280
C     ADD MORE OPTIONS IF REQUESTED                                     PRVR2290
      DO 15 I=2,NP1                                                     PRVR2300
      NN=NPR(I-1)                                                       PRVR2310
   15 RPRINT(I)=RKVARS(NN)                                              PRVR2320
C                                                                       PRVR2330
16    IF(V.GE.0.0) THEN                                                 PRVR2340
         WRITE(TMP,'(A1,0PE7.2)') ERT,V                                 PRVR2350
      ELSE                                                              PRVR2360
         WRITE(TMP,'(A1,0PE7.1)') ERT,V                                 PRVR2370
      ENDIF                                                             PRVR2380
C     DETERMINE WHERE TO PUT A SPACE                                    PRVR2390
C     AT BEGINNING OR END OF 1ST 2 VALUES                               PRVR2400
      KT=1                                                              PRVR2410
      IF(TMP(1:1).EQ.'K') KT=2                                          PRVR2420
C                                                                       PRVR2430
C     TEST IF NEED NEW PAGE                                             PRVR2440
```

```
      CALL TSTPAG(3)                                                    PRVR2450
      IF (SR.GE.1.E-6) GO TO 20                                         PRVR2460
C********* TOO CLOSE TO TRANSMITTER TO CALCULATE DIRECTION FROM         PRVR2470
C********* TRANSMITTER                                                 PRVR2480
      IF(PRTSRP.EQ.0.0)                                                 PRVR2490
     1 WRITE(3,1500)TMP(KT:9),NWHY,H,GRH,RANGE,ELL,(RPRINT(NN),NN=1,NP1)PRVR2500
 1500 FORMAT (1X,A8,A8,2F10.4,F11.4,26X,F8.3,4F12.4)                    PRVR2510
C     SET RAYSET VARIABLES TO UNDEFINDED VALUES FOR FLAGS               PRVR2520
      AZDEV=999.0                                                       PRVR2530
      AZA=999.0                                                         PRVR2540
      GO TO 40                                                          PRVR2550
C********* ELEVATION ANGLE OF RAY POINT FROM TRANSMITTER                PRVR2560
   20 EL=ATAN2(EPS,RCE)*DEGS                                            PRVR2570
      IF (RCE.GE.1.E-6) GO TO 30                                        PRVR2580
C********* NEARLY DIRECTLY ABOVE OR BELOW TRANSMITTER. CAN NOT CALCULATEPRVR2590
C********* AZIMUTH DIRECTION FROM TRANSMITTER ACCURATELY                PRVR2600
      IF(PRTSRP.EQ.0.0)                                                 PRVR2610
     1   WRITE(3,2500)TMP(KT:9),NWHY,H,GRH,RANGE,EL,ELL                 PRVR2620
     2  ,(RPRINT(NN),NN=1,NP1)                                          PRVR2630
 2500 FORMAT (1X,A8,A8,2F10.4,F11.4,17X,F9.3,F8.3,                      PRVR2640
     1 4F12.4)                                                          PRVR2650
      GO TO 40                                                          PRVR2660
C********* AZIMUTH ANGLE OF RAY POINT FROM TRANSMITTER                  PRVR2670
   30 ANGA=ATAN2(ETA,ZETA)                                              PRVR2680
      AZDEV=180.-AMOD(540.-(AZ1-ANGA)*DEGS,360.)                        PRVR2690
      IF (KTH.NE.0..OR.KPH.NE.0.) GO TO 34                              PRVR2700
C********* WAVE NORMAL IS VERTICAL, SO AZIMUTH DIRECTION CANNOT BE      PRVR2710
C********* CALCULATED                                                  PRVR2720
      IF(PRTSRP.EQ.0.0)                                                 PRVR2730
     1 WRITE(3,3000)TMP(KT:9),NWHY,H,GRH,RANGE,AZDEV,EL,ELL,(RPRINT(NN),PRVR2740
     1 NN=1,NP1)                                                        PRVR2750
 3000 FORMAT (1X,A8,A8,2F10.4,F11.4,F9.3,8X,F9.3,F8.3,                  PRVR2760
     1 4F12.4)                                                          PRVR2770
      GO TO 40                                                          PRVR2780
   34 ANA=ANGA-ALPH                                                     PRVR2790
      SANA=SIN (ANA)                                                    PRVR2800
      SPHI=SANA*STHR/STH                                                PRVR2810
      CPHI=-SIN (PID2-ANA)*SIN (PID2-(PHI-PHIR))+SANA*SIN (PHI-PHIR)    PRVR2820
     1 *CTHR                                                            PRVR2830
      AZA=180.-AMOD (540.-(ATAN2(SPHI,CPHI)-ATAN2(KPH,KTH))*DEGS,360.)  PRVR2840
      IF(PRTSRP.EQ.0.0)                                                 PRVR2850
     1  WRITE(3,3500)TMP(KT:9),NWHY,H,GRH,RANGE,AZDEV,AZA,EL,ELL        PRVR2860
     2  ,(RPRINT(NN),NN=1,NP1)                                          PRVR2870
 3500 FORMAT (1X,A8,A8,2F10.4,F11.4,2(F9.3,F8.3),                       PRVR2880
     1 4F12.4)                                                          PRVR2890
C*********                                                             PRVR2900
   40 LINES=LINES+1                                                     PRVR2910
      IF (NP.LE.3) GO TO 45                                             PRVR2920
C********* ADDITIONAL LINE TO PRINT REMAINING DEPENDENT INTEGRATION     PRVR2930
C********* VARIABLES                                                   PRVR2940
      IF(PRTSRP.EQ.0.0)                                                 PRVR2950
     1   WRITE(3,4000) (RPRINT(NN),NN=4,NP)                             PRVR2960
 4000 FORMAT (99X,3F12.4)                                               PRVR2970
      LINES=LINES+1                                                     PRVR2980
C                                                                       PRVR2990
```

```
C     IF NO 'CARDS' WANTED OR AT TRANSMITTER, NO RAYSET OUTPUT                PRVR3000
C                                                                             PRVR3010
45    IF (CARD.EQ.0.0 .OR. IHOP.LT.1) RETURN                                  PRVR3020
C                                                                             PRVR3030
C********* PUNCH A RAYSET                                                     PRVR3040
      IF (AZDEV.LT.-90.) AZDEV=AZDEV+360.                                     PRVR3050
      IF (AZA.LT.-90.) AZA=AZA+360.                                           PRVR3060
      NR=0                                                                    PRVR3070
      IF (W(57).EQ.0.) GO TO 47                                               PRVR3080
C********* PHASE PATH                                                        PRVR3090
      NR=NR+1                                                                 PRVR3100
      P=RKVARS(NR)                                                            PRVR3110
   47 IF (W(58).EQ.0.) GO TO 48                                               PRVR3120
C********* ABSORPTION                                                        PRVR3130
      NR=NR+1                                                                 PRVR3140
      ABSORB=RKVARS(NR)                                                       PRVR3150
C********* DOPPLER SHIFT                                                     PRVR3160
   48 IF (W(59).NE.0.) DOPP=RKVARS(NR+1)                                      PRVR3170
      NHPUNCH=ROUND(HPUNCH*1.E4)                                              PRVR3180
       NRANGE=ROUND(RANGE*1.E4)                                               PRVR3190
      NAZDEV=ROUND(AZDEV*1.E3)                                                PRVR3200
       NAZA=ROUND(AZA*1.E3)                                                   PRVR3210
       NELL=ROUND(ELL*1.E3)                                                   PRVR3220
      IF(NWHY.EQ.'GRND REF') NELL =                                           PRVR3230
     1 ROUND((PID2 - ACOS(KNORM(G)/SQRT(KR*KR+KTH*KTH+KPH*KPH)))              PRVR3240
     2   *DEGS*1.E3)                                                         PRVR3250
      NABSORB=AMIN1(999999.0,ROUND(ABSORB*1.E3))                              PRVR3260
       NDOPP=ROUND(DOPP*1.E3)                                                 PRVR3270
      NPOLAR1=ROUND(REAL(POLAR)*1.E2)                                         PRVR3280
       NPOLAR2=ROUND(AIMAG(POLAR)*1.E2)                                       PRVR3290
      JP=ROUND(P*1.E5)                                                        PRVR3300
      JT=ROUND(T*1.E5)                                                        PRVR3310
      WRITE(9,4501) NHPUNCH,NRANGE,NAZDEV,NAZA,NELL,JT,JP,NABSORB             PRVR3320
     1 ,NDOPP,NPOLAR1,NPOLAR2,IHOP,EVENT(1:1)                                 PRVR3330
 4501 FORMAT(2I9,3I6,2I10,2I6,I5,I4,I2,A1)                                    PRVR3340
      RETURN                                                                  PRVR3350
      END                                                                     PRVR3360
```

```
      SUBROUTINE OCNHD                                                        OCBD0020
C     PRINTS PAGE HEADINGS                                                    OCBD0030
C     IF W(72) IS NEGATIVE, ONLY ONE HEADER OUTPUT IS INCLUDED IN RAYSE       OCBD0040
      CHARACTER PCC*1,PC*1,BLANKS*100,DIVIDR*132,BANNER(8)*80                 OCBD0050
      CHARACTER NUMSTG*80,STMP*80                                             OCBD0060
      LOGICAL NOPUNCH                                                         OCBD0070
      INTEGER STRIM                                                           OCBD0080
C                                                                             OCBD0090
C     TWO ENTRY POINTS ARE PROVIDED. ONE FOR THE FIRST PAGE HEADER            OCBD0100
C     OF THE COMPUTATIONAL PRINTOUT AND FOR THE RAYSET FILE. THE              OCBD0110
C     SECOND ENTRY IS FOR ALL SUBSEQUENT PAGES OF THE COMPUTATIONAL           OCBD0120
C     PRINTOUT.                                                               OCBD0130
C                                                                             OCBD0140
```

```
C       COMMON DECK "LL" INSERTED HERE                                          CLL 0020
        REAL MODL                                                               CLL 0040
        COMMON/LL/MODL(4),APH,APHPT,APHPR,APHPTH,APHPPH                         CLL 0050
C       COMMON DECK "PP" INSERTED HERE                                          CPP 0020
        REAL MODP                                                               CPP 0040
        COMMON/PP/MODP(4),P,PPT,PPR,PPTH,PPPH                                   CPP 0050
C                                                                               OCBD0170
C       COMMON DECK "SS" INSERTED HERE                                          CSS 0020
        REAL MODSURF                                                            CSS 0040
        COMMON/SS/ MODSURF(4)                                                   CSS 0050
        COMMON/SS/U,PUR,PURR,PURTH,PURPH                                        CSS 0060
        COMMON/SS/PUTH,PUPH,PUTHTH,PUPHPH,PUTHPH,USELECT,UTIME                  CSS 0070
C       COMMON DECK "CONST" INSERTED HERE                                       CCON0020
        COMMON/PCONST/CREF,RGAS,GAMMA                                           CCON0040
        COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                               CCON0050
C       COMMON DECK "WWR" INSERTED HERE                                         CWWR0020
        PARAMETER (NWARSZ=1000)                                                 CWW10030
        COMMON/WW/ID(10),MAXW,W(NWARSZ)                                         CWW10040
        REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                      CWW20020
        EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),          CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),              CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),                  CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),                 CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                         CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))        CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                              CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                              CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                             CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),               CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))                CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),                CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                      CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                           CWW20160
C       COMMON DECK "GG" INSERTED HERE                                          CGG 0020
        REAL MODG                                                               CGG 0040
        COMMON/GG/MODG(4)                                                       CGG 0050
        COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                        CGG 0060
        COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME                  CGG 0070
C       COMMON DECK "RR" INSERTED HERE                                          CRR 0020
        REAL MODREC                                                             CRR 0040
        COMMON/RR/ MODREC(4)                                                    CRR 0050
        COMMON/RR/F,PFR,PFRR,PFRTH,PFRPH                                        CRR 0060
        COMMON/RR/PFTH,PFPH,PFTHTH,PFPHPH,PFTHPH,FSELECT,FTIME                  CRR 0070
C       COMMON DECK "B9" INSERTED HERE                                          CB8 0020
        INTEGER GMX,GNTBL,GITBL,GFRMTBL,IDSG(10)                                CB8 0040
        COMMON/B9/GMX,GNTBL(10),GITBL(10),GFRMTBL(10),GGP(113)                  CB8 0050
        EQUIVALENCE (GGP,IDSG),(ANG,GGP(11))                                    CB8 0060
C       COMMON DECK "B2" INSERTED HERE                                          CB2 0020
        INTEGER DUMX,DUNTBL,DUITBL,DUFRMTB,IDSDU(10)                            CB2 0040
        COMMON/B2/DUMX,DUNTBL(10),DUITBL(10),DUFRMTB(10),DUGP(10)               CB2 0050
        EQUIVALENCE (DUGP,IDSDU)                                                CB2 0060
C       COMMON DECK "B4" INSERTED HERE                                          CB4 0020
        INTEGER DCMX,DCNTBL,DCITBL,DCFRMTB,IDSDC(10)                            CB4 0040
        COMMON/B4/DCMX,DCNTBL(10),DCITBL(10),DCFRMTB(10),DCGP(10)               CB4 0050
        EQUIVALENCE (DCGP,IDSDC)                                                CB4 0060
```

```
C       COMMON DECK "B6" INSERTED HERE                                          CB6 0020
        INTEGER DTMX,DTNTBL,DTITBL,DTFRMTB,IDSDT(10)                            CB6 0040
        COMMON/B6/DTMX,DTNTBL(10),DTITBL(10),DTFRMTB(10),DTGP(10)               CB6 0050
        EQUIVALENCE (DTGP,IDSDT)                                                CB6 0060
C       COMMON DECK "B1" INSERTED HERE                                          CB1 0020
        INTEGER UMX,UNTBL,UITBL,UFRMTBL,IDSU(10)                                CB1 0040
        COMMON/B1/UMX,UNTBL(10),UITBL(10),UFRMTBL(10),UGP(10)                   CB1 0050
        EQUIVALENCE (UGP,IDSU)                                                  CB1 0060
C       COMMON DECK "B3" INSERTED HERE                                          CB3 0020
        INTEGER CMX,CNTBL,CITBL,CFRMTBL,IDSC(10)                                CB3 0040
        COMMON/B3/CMX,CNTBL(10),CITBL(10),CFRMTBL(10),CGP(512)                  CB3 0050
        EQUIVALENCE (CGP,IDSC),(ANC,CGP(11))                                    CB3 0060
C       COMMON DECK "B5" INSERTED HERE                                          CB5 0020
        INTEGER TMX,TNTBL,TITBL,TFRMTBL,IDST(10)                                CB5 0040
        COMMON/B5/TMX,TNTBL(10),TITBL(10),TFRMTBL(10),TGP(262)                  CB5 0050
        EQUIVALENCE (TGP,IDST),(ANT,TGP(11))                                    CB5 0060
C       COMMON DECK "B7" INSERTED HERE                                          CB7 0020
        INTEGER MMX,MNTBL,MITBL,MFRMTBL,IDSM(10)                                CB7 0040
        REAL MGP                                                                CB7 0050
        COMMON/B7/MMX,MNTBL(10),MITBL(10),MFRMTBL(10),MGP(10)                   CB7 0060
        EQUIVALENCE (MGP,IDSM)                                                  CB7 0070
C       COMMON DECK "HDR" INSERTED HERE                                         CHDR0020
        CHARACTER*10 INITID*80,DAT,TOD                                          CHDR0040
        COMMON/HDR/SEC                                                          CHDR0050
        COMMON/HDRC/INITID,DAT,TOD                                              CHDR0060
C       COMMON DECK "RINPLEX" INSERTED HERE                                     CRIN0020
        REAL KAY2,KAY2I                                                         CRIN0040
        COMPLEX PNP,POLAR,LPOLAR                                                CRIN0050
        LOGICAL SPACE                                                           CRIN0060
        CHARACTER DISPM*6                                                       CRIN0070
        COMMON/RINPL/DISPM                                                      CRIN0080
        COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                       CRIN0090
        COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I                                   CRIN0100
        COMMON/RIN/PNP(10),POLAR,LPOLAR,SGN                                     CRIN0110
C       COMMON DECK "CC" INSERTED HERE                                          CCC 0020
        REAL MODC                                                               CCC 0040
        COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                              CCC 0050
C       COMMON DECK "MM" INSERTED HERE                                          CMM 0020
        REAL M,MODM                                                             CMM 0040
        COMMON/MM/MODM(4),M,PMT,PMR,PMTH,PMPH                                   CMM 0050
C       COMMON DECK "FLAG" INSERTED HERE                                        CFLA0020
        LOGICAL NEWWR,NEWWP,NEWTRC,PENET                                        CFLA0040
        COMMON /FLG/ NTYP,NEWWR,NEWWP,NEWTRC,PENET,LINES,IHOP,HPUNCH            CFLA0050
        COMMON/FLGP/NSET                                                        CFLA0060
C       COMMON DECK "TT" INSERTED HERE                                          CTT 0020
        REAL MODT                                                               CTT 0040
        COMMON/TT/MODT(4), T,PTT,PTR,PTTH,PTPH                                  CTT 0050
C       COMMON DECK "UU" INSERTED HERE                                          CUU 0020
        REAL MODU                                                               CUU 0040
        COMMON/UU/MODU(4)                                                       CUU 0050
     1   ,V   ,PVT   ,PVR   ,PVTH   ,PVPH                                      CUU 0060
     2   ,VR  ,PVRT  ,PVRR  ,PVRTH  ,PVRPH                                     CUU 0070
     3   ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                        CUU 0080
     4   ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                        CUU 0090
C                                                                               OCBD0380
```

```
C     COMMON DECK "B10" INSERTED HERE                                    CB9 0020
      INTEGER DGMX,DGNTBL,DGITBL,DGFRMTB,IDSDG(10)                       CB9 0040
      COMMON/B10/DGMX,DGNTBL(10),DGITBL(10),DGFRMTB(10),DGGP(10)         CB9 0050
      EQUIVALENCE (DGGP,IDSDG)                                           CB9 0060
C     COMMON DECK "B8" INSERTED HERE                                     CB100020
      INTEGER RMX,RNTBL,RITBL,RFRMTBL,IDSR(10)                           CB100040
      COMMON/B8/RMX,RNTBL(10),RITBL(10),RFRMTBL(10),RGP(10)              CB100050
      EQUIVALENCE (RGP,IDSR)                                             CB100060
C     COMMON DECK "CB11" INSERTED HERE                                   CB110020
      INTEGER SMX,SNTBL,SITBL,SFRMTBL,IDSS(10)                           CB110040
      COMMON/B11/SMX,SNTBL(10),SITBL(10),SFRMTBL(10),SGP(11)             CB110050
      EQUIVALENCE (SGP,IDSS),(ANS,SGP(11))                               CB110060
C     COMMON DECK "CB12" INSERTED HERE                                   CB120020
      INTEGER DSMX,DSNTBL,DSITBL,DSFRMTB,IDSDS(10)                       CB120040
      COMMON/B12/DSMX,DSNTBL(10),DSITBL(10),DSFRMTB(10),DSGP(11)         CB120050
      EQUIVALENCE (DSGP,IDSDS),(ANDS,DSGP(11))                           CB120060
C     COMMON DECK "CB17" INSERTED HERE                                   CB170020
      INTEGER VMX,VNTBL,VITBL,VFRMTBL,IDSV(10)                           CB170040
      COMMON/B17/VMX,VNTBL(10),VITBL(10),VFRMTBL(10),VGP(53)             CB170050
      EQUIVALENCE (VGP,IDSV),(ANV,VGP(11))                               CB170060
C     COMMON DECK "CB18" INSERTED HERE                                   CB180020
      INTEGER DVMX,DVNTBL,DVITBL,DVFRMTB,IDSDV(10)                       CB180040
      COMMON/B18/DVMX,DVNTBL(10),DVITBL(10),DVFRMTB(10),DVGP(11)         CB180050
      EQUIVALENCE (DVGP,IDSDV),(ANDV,DVGP(11))                           CB180060
C     COMMON DECK "CB19" INSERTED HERE                                   CB190020
      INTEGER PRMX,PRNTBL,PRITBL,PRFRMTB,IDSPR(10)                       CB190040
      COMMON/B19/PRMX,PRNTBL(10),PRITBL(10),PRFRMTB(10),PRGP(11)         CB190050
      EQUIVALENCE (PRGP,IDSPR),(ANP,PRGP(11))                            CB190060
C     COMMON DECK "CB20" INSERTED HERE                                   CB200020
      INTEGER DPMX,DPNTBL,DPITBL,DPFRMTB,IDSDP(10)                       CB200040
      COMMON/B20/DPMX,DPNTBL(10),DPITBL(10),DPFRMTB(10),DPGP(11)         CB200050
      EQUIVALENCE (DPGP,IDSDP),(ANDP,DPGP(11))                           CB200060
C                                                                        OCBD0470
      PARAMETER (NBNRLS=8,NBLNS=8)                                       OCBD0480
C                                                                        OCBD0490
      DATA BLANKS/' '/                                                   OCBD0500
      DATA BANNER/                                                       OCBD0510
     1  '***** H A R P O *****'                                          OCBD0520
     2 ,'HAMILTONIAN ACOUSTIC RAY-TRACING PROGRAM FOR THE OCEAN'         OCBD0530
     3 ,' '                                                              OCBD0540
     4 ,'BY'                                                             OCBD0550
     5 ,'R. M. JONES,  J. P. RILEY  AND  T. M. GEORGES'                  OCBD0560
     6 ,'WAVE PROPAGATION LABORATORY'                                    OCBD0570
     7 ,'NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION'                OCBD0580
     8 ,'BOULDER, COLORADO 80303'/                                       OCBD0590
      DATA NOPUNCH/.FALSE./                                              OCBD0600
      DATA IBLK/1H /                                                     OCBD0610
C                                                                        OCBD0620
      ENTRY HEADER1                                                      OCBD0630
C                                                                        OCBD0640
C     COMPUTE EFFECTIVE LINES COUNT BASED ON FIXED PAGE SIZE             OCBD0650
      CALL NEWPAG(NPAG,INT(PAGLN),PC)                                    OCBD0660
      CALL SFILL(DIVIDR,LEN(DIVIDR),'-')                                 OCBD0670
      DIVIDR(1:1)=' '                                                    OCBD0680
      NTYP=2                                                             OCBD0690
```

```
      IF(RAY.NE.0.0) NTYP=2.0+SIGN(1.0,RAY)                                  OCBD0700
C                                                                            OCBD0710
1600  FORMAT(2(2(A8,F7.1,1X),2X))                                            OCBD0720
      CALL PUTKST(3,'1'                                                      OCBD0730
     1  //DAT//TOD//BLANKS(:100)//'PAGE'//NUMSTG(NPAG,1,'(I5)'))            OCBD0740
      CALL PUTKST(3,DIVIDR)                                                  OCBD0750
      CALL PUTKBK(3,1)                                                       OCBD0760
      DO 15 I=1,NBNRLS                                                       OCBD0770
15    CALL PUTKCT(3,BANNER(I))                                               OCBD0780
      CALL PUTKBK(3,1)                                                       OCBD0790
      CALL PUTKST(3,DIVIDR)                                                  OCBD0800
      CALL PUTKBK(3,NBLNS)                                                   OCBD0810
C                                                                            OCBD0820
      CALL PUTKST(3,BLANKS(:57)//'RUN SET NUMBER'                            OCBD0830
     1  //NUMSTG(NSET,1,'(I5)'))                                            OCBD0840
      CALL PUTKBK(3,1)                                                       OCBD0850
      WRITE(STMP,'(10A8)') ID                                                OCBD0860
      CALL PUTKST(3,BLANKS(:52)//'OCEAN MODEL ID -- '//STMP(:3))             OCBD0870
      CALL PUTKBK(3,1)                                                       OCBD0880
      CALL PUTKST(3,                                                         OCBD0890
     1 BLANKS(:25)//'OCEAN MODEL DESCRIPTION -- '//STMP(7:))                 OCBD0900
      CALL PUTKBK(3,1)                                                       OCBD0910
C                                                                            OCBD0920
      WRITE(STMP,'(2A8)') (RAYNAME(I,NTYP),I=1,2)                            OCBD0930
      CALL PUTKST(3,STMP)                                                    OCBD0940
      CALL PUTKBK(3,1)                                                       OCBD0950
      CALL PUTKST(3,DIVIDR)                                                  OCBD0960
      CALL PUTKBK(3,1)                                                       OCBD0970
      CALL PUTKST(3,                                                         OCBD0980
     1  BLANKS(:8)//'MODEL                    SUBROUTINE    DATA SET'       OCBD0990
     1  //'       DESCRIPTION')                                             OCBD1000
      CALL PUTKST(3,                                                         OCBD1010
     1  BLANKS(:8)//'TYPE                       NAME          ID')           OCBD1020
      CALL PUTKBK(3,1)                                                       OCBD1030
      CALL PUTKST(3,DIVIDR)                                                  OCBD1040
      CALL PUTKBK(3,1)                                                       OCBD1050
      CALL PUTKST(3,' DISPERSION RELATION           '//DISPM                 OCBD1060
     1  //BLANKS(:16)//NUMSTG(MODRIN,8,'(8A8)'))                            OCBD1070
      CALL PUTDES(3,'BACKGROUND CURRENT VELOCITY',MODU,IDSU)                 OCBD1080
      CALL PUTDES(3,'CURRENT VELOCITY PERTURBATION',MODU(3),IDSDU)           OCBD1090
      CALL PUTDES(3,'BACKGROUND SOUND SPEED',MODC,IDSC)                      OCBD1100
      CALL PUTDES(3,'SOUND SPEED PERTURBATION',MODC(3),IDSDC)                OCBD1110
      CALL PUTDES(3,'BACKGROUND BOTTOM',MODG,IDSG)                           OCBD1120
      CALL PUTDES(3,'BOTTOM PERTURBATION',MODG(3),IDSDG)                     OCBD1130
      CALL PUTDES(3,'ABSORPTION',MODL,IDSV)                                  OCBD1140
      CALL PUTDES(3,'ABSORPTION PERTURBATION',MODL(3),IDSDV)                 OCBD1150
      CALL PUTDES(3,'RECEIVER SURFACE',MODREC,IDSR)                          OCBD1160
      CALL PUTDES(3,'OCEAN SURFACE',MODSURF,IDSS)                            OCBD1170
      CALL PUTDES(3,'OCEAN SURFACE PERTURBATION',MODSURF(3),IDSDS)           OCBD1180
      CALL PUTKST(3,DIVIDR)                                                  OCBD1190
C                                                                            OCBD1200
      NOPUNCH=NOPUNCH.AND.RAYSET.LT.0.0                                      OCBD1210
      IF(NOPUNCH) RETURN                                                     OCBD1220
      NOPUNCH=RAYSET.LT.0.0                                                  OCBD1230
C                                                                            OCBD1240
```

```
      WRITE(9,1200) ID,DAT,TOD                                                  OCBD1250
      WRITE(9,1600) MODU,MODC,MODG,MODL,MODREC(1),MODREC(2),MODSURF             OCBD1260
C                                                                               OCBD1270
      IF(IDSU(1) .NE. IBLK) WRITE(9,1200) IDSU                                  OCBD1280
      IF(IDSDU(1) .NE. IBLK) WRITE(9,1200) IDSDU                                OCBD1290
      IF(IDSC(1) .NE. IBLK) WRITE(9,1200) IDSC                                  OCBD1300
      IF(IDSDC(1) .NE. IBLK) WRITE(9,1200) IDSDC                                OCBD1310
      IF(IDST(1) .NE. IBLK) WRITE(9,1200) IDST                                  OCBD1320
      IF(IDSDT(1) .NE. IBLK) WRITE(9,1200) IDSDT                                OCBD1330
      IF(IDSM(1) .NE. IBLK) WRITE(9,1200) IDSM                                  OCBD1340
      IF(IDSG(1) .NE. IBLK) WRITE(9,1200) IDSG                                  OCBD1350
C                                                                               OCBD1360
 1000 FORMAT (A1,10A8,24X,2A,' PAGE',I4)                                        OCBD1370
1200  FORMAT(10A8,2(A8,2X))                                                     OCBD1380
C                                                                               OCBD1390
      RETURN                                                                    OCBD1400
C                                                                               OCBD1410
      ENTRY HEADER2                                                             OCBD1420
C                                                                               OCBD1430
C     COMPUTE EFFECTIVE LINES COUNT BASED ON FIXED PAGE SIZE                    OCBD1440
      CALL NEWPAG(NPAG,INT(PAGLN),PC)                                           OCBD1450
      LINES=LINES+5                                                             OCBD1460
C                                                                               OCBD1470
      WRITE(3,1000) PC,ID,DAT,TOD,NPAG                                          OCBD1480
      WRITE(3,2400) AZ1*DEGS,TLAT*DEGS,OW/PIT2,BETA*DEGS                        OCBD1490
     1         ,TLON*DEGS,MAXERR                                                OCBD1500
 2400 FORMAT (                                                                  OCBD1510
     1  /' AZIMUTH ANGLE OF TRANSMISSION   =',F12.6,' DEG'                      OCBD1520
     2  ,'      TRANSMITTER  LATITUDE  =',F12.6,' DEG'                         OCBD1530
     3  ,'    FREQUENCY =',F12.6,' HZ'                                         OCBD1540
     4  /' ELEVATION ANGLE OF TRANSMISSION =',F12.6,' DEG'                      OCBD1550
     5  ,'      TRANSMITTER  LONGITUDE =',F12.6,' DEG'                         OCBD1560
     6  ,'    SINGLE STEP ERROR =',1PG13.6/)                                   OCBD1570
C                                                                               OCBD1580
      RETURN                                                                    OCBD1590
C                                                                               OCBD1600
      ENTRY PUTDVR(NUNIT)                                                       OCBD1610
      CALL PUTKST(NUNIT,DIVIDR)                                                 OCBD1620
      RETURN                                                                    OCBD1630
C                                                                               OCBD1640
      ENTRY PUTHDR(NUNIT,PCC,NP)                                                OCBD1650
      CALL PUTKST(NUNIT,                                                        OCBD1660
     1  PCC//DAT//TOD//BLANKS(:100)//'PAGE'//NUMSTG(NP,1,'(I5)'))               OCBD1670
      RETURN                                                                    OCBD1680
      END                                                                       OCBD1690
```

```
      SUBROUTINE PUTDES(NUNIT,DES,MOD,ID)                                       PUQS0020
C     TITLING AID, OUTPUT MODULE DESCRIPTION AS HOLLERITH NAME                  PUQS0030
C     AND NUMERIC IDENTIFIER                                                    PUQS0040
      CHARACTER DES*(*),TITLE*30,SMODL*20,NUMSTG*80                             PUQS0050
      REAL MOD(2),ID(10)                                                        PUQS0060
```

```
TITLE=DES                                                                PUQS0070
WRITE(SMODL,'(A8,F10.2)') MOD                                            PUQS0080
CALL PUTKST(NUNIT, ' '//TITLE//SMODL//NUMSTG(ID,10,'(10A8)'))            PUQS0090
END                                                                      PUQS0100
```

```
      FUNCTION NUMSTG(V,N,FRM)                                           NUNG0020
C     CONVERTS A NUMERIC VALUE TO A STRING                               NUNG0030
C     USING FORMAT PROVIDED IN CALL                                      NUNG0040
      CHARACTER NUMSTG*80,FRM*(*)                                        NUNG0050
      INTEGER V(N)                                                       NUNG0060
C                                                                        NUNG0070
      NUMSTG=' '                                                         NUNG0080
      WRITE(NUMSTG,FRM) V                                                NUNG0090
      RETURN                                                             NUNG0100
      END                                                                NUNG0110
```

```
      SUBROUTINE SFILL(STG,LN,C)                                         SFSL0020
C     FILLS A STRING WITH N SPECIFIED CHARACTERS                         SFSL0030
      CHARACTER STG*(*),C*1                                              SFSL0040
C                                                                        SFSL0050
      DO 10 I=1,LN                                                       SFSL0060
10    STG(I:I)=C                                                         SFSL0070
      RETURN                                                             SFSL0080
      END                                                                SFSL0090
```

```
      INTEGER FUNCTION STRIM(C)                                          STEM0020
C     DETERMINES POSITION OF LAST NONBLANK CHARACTER OF A STRING         STEM0030
      CHARACTER*(*) C                                                    STEM0040
C                                                                        STEM0050
      L=LEN(C)                                                           STEM0060
      DO 100 I=L,1,-1                                                    STEM0070
100   IF(C(I:I) .NE. ' ') GO TO 200                                      STEM0080
      I=0                                                                STEM0090
200   STRIM=I                                                            STEM0100
      END                                                                STEM0110
```

```
      FUNCTION RERR(V,ERT,ELAB,NKV,PREF)                                 RERR0020
C     RETURNS RELATIVE ERROR OF VARIABLE RKVAR(NKV) IF PREF<>0 AND       RERR0030
C     THE ERROR IS GREATER THAN PREVIOUS ERROR 'V'.                      RERR0040
```

```
C       COMMON DECK "RKAM" INSERTED HERE                                        RKAM0020
        REAL KR,KTH,KPH                                                         RKAM0040
        COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)             RKAM0050
C                                                                               RERR0060
        RERR=V                                                                  RERR0070
        IF(NKV.LE.0) RETURN                                                     RERR0080
        IF(PREF.EQ.0.) RETURN                                                   RERR0090
        V1=RKVARS(NKV)/PREF-1.                                                  RERR0100
        IF(ABS(V1).LT.ABS(V)) RETURN                                            RERR0110
C                                                                               RERR0120
        ERT=ELAB                                                                RERR0130
        RERR=V1                                                                 RERR0140
        END                                                                     RERR0150



        SUBROUTINE RERROR(ROUTIN,STR,VAL)                                       REMR0020
C       REPORTS ERROR CONDITIONS AND STOPS PROGRAM.                             REMR0030
C                                                                               REMR0040
        PRINT 10,ROUTIN,STR,VAL                                                 REMR0050
10      FORMAT(39H ERROR CONDITION IN RAYTRACE ROUTINE <                        REMR0060
     1  ,A8,10H> DUE TO " ,A10,9H", VALUE=  ,F8.2)                              REMR0070
        STOP                                                                    REMR0080
        END                                                                     REMR0090



        SUBROUTINE STOPIT(A)                                                    STJT0020
C       PRINTS CONDITION AND STOPS PROGRAM                                      STJT0030
C       AFTER CALLING THE SYSTEM POST MORTEM DUMP.                              STJT0040
C                                                                               STJT0050
        CHARACTER A*(*)                                                         STJT0060
C                                                                               STJT0070
        PRINT 100, A                                                            STJT0080
        CALL MORTEM                                                             STJT0090
100     FORMAT('*** STOPIT WITH CONDITION <',A,'>')                             STJT0100
        STOP                                                                    STJT0110
        END                                                                     STJT0120



        SUBROUTINE PUTKST(NUNIT,STRG)                                           PUMT0020
C       WRITE LINE OF OUTPUT TO PRINTER UNIT ADDING TO LINE COUNT               PUMT0030
C                                                                               PUMT0040
        CHARACTER STRG*(*)                                                      PUMT0050
        INTEGER STRIM                                                           PUMT0060
C                                                                               PUMT0070
C       COMMON DECK "FLAG" INSERTED HERE                                        CFLA0020
        LOGICAL NEWWR,NEWWP,NEWTRC,PENET                                        CFLA0040
```

```
      COMMON /FLG/ NTYP,NEWWR,NEWWP,NEWTRC,PENET,LINES,IHOP,HPUNCH      CFLA0050
      COMMON/FLGP/NSET                                                  CFLA0060
                                                                        PUMT0090
C                                                                       PUMT0100
C     PUT OUT A STRING WITH LINE COUNT INCREMENT                        PUMT0110
      LINES=LINES+1                                                     PUMT0120
      LN=MAX0(1,MIN0(STRIM(STRG),132))                                  PUMT0130
      WRITE(NUNIT,'(A)') STRG(1:LN)                                     PUMT0140
      RETURN                                                            PUMT0150
      END                                                               PUMT0160
C
```

```
      SUBROUTINE PUTKCT(NUNIT,STRG)                                     PUMT0170
C     PUT OUT A CENTERED LINE AND COUNT                                 PUMT0180
      CHARACTER STRG*(*)                                                PUMT0190
      CHARACTER BLANKS*100,STMP*80                                      PUMT0200
      INTEGER STRIM                                                     PUMT0210
      CHARACTER PC*1                                                    PUMT0220
C                                                                       PUMT0230
C     COMMON DECK "FLAG" INSERTED HERE                                  CFLA0020
      LOGICAL NEWWR,NEWWP,NEWTRC,PENET                                  CFLA0040
      COMMON /FLG/ NTYP,NEWWR,NEWWP,NEWTRC,PENET,LINES,IHOP,HPUNCH      CFLA0050
      COMMON/FLGP/NSET                                                  CFLA0060
C                                                                       PUMT0250
      DATA BLANKS/' '/                                                  PUMT0260
C                                                                       PUMT0270
      NTRM=MAX0(1,STRIM(STRG))                                          PUMT0280
      NBLKS=66-(NTRM+1)/2                                               PUMT0290
      LINES=LINES+1                                                     PUMT0300
      LN=MIN0(NTRM,132-NBLKS)                                           PUMT0310
      STMP=STRG                                                         PUMT0320
      WRITE(NUNIT,'(A)') BLANKS(:NBLKS)//STMP(:LN)                      PUMT0330
      RETURN                                                            PUMT0340
C                                                                       PUMT0350
      ENTRY PUTKBK(NUNIT,NBKS)                                          PUMT0360
C     PUT NBLKS BLANK LINES TO UNIT NUNIT                               PUMT0370
C                                                                       PUMT0380
      LINES=LINES+1                                                     PUMT0390
      DO 100 I=1,NBKS                                                   PUMT0400
100      WRITE(NUNIT,'(1X)')                                            PUMT0410
      RETURN                                                            PUMT0420
C                                                                       PUMT0430
      ENTRY NEWPAG(NXP,LIP,PC)                                          PUMT0440
C     THIS ENTRY COMPUTES NEXT PAGE NUMBER AND LINE COUNT TO END        PUMT0450
C     OF CURRENT PAGE                                                   PUMT0460
C     RETAIN LINES PER PAGE FOR USE BY TSTPAG ENTRY                     PUMT0470
      LINSPP=LIP                                                        PUMT0480
C     THE LOGIC HERE ASSUMES THAT A FORM FEED IS ALWAYS NEEDED.         PUMT0490
C     A MORE SOPHISTICATED VERSION WOULD DETERMINE IF A NEW PAGE        PUMT0500
C     IS NEEDED AND RETURN PC=' ' IF NOT. THE CALLING PROGRAM COULD     PUMT0510
C     USE OR IGNORE.                                                    PUMT0520
      PC='1'                                                            PUMT0530
```

```
C       COMPUTE CURRENT PAGE NUMBER                                     PUMT0540
        NPAG=(LINES+LINSPP-1)/LINSPP                                    PUMT0550
C       COMPUTE LINE COUNT AT END OF CURRENT PAGE                       PUMT0560
        LINES=NPAG*LINSPP                                               PUMT0570
        NXP=NPAG+1                                                      PUMT0580
C       COMPUTE LINE COUNT AT END OF NEXT PAGE                          PUMT0590
        LNSXPG=LINES+LINSPP-1                                           PUMT0600
        RETURN                                                          PUMT0610
C                                                                       PUMT0620
        ENTRY TSTPAG(NUNIT)                                             PUMT0630
C       OUTPUT HEADER IF HAVE REACHED LAST LINE OF PAGE                 PUMT0640
        IF(LINES.GE.LNSXPG) THEN                                        PUMT0650
C       COMPUTE CURENT PAGE NUMBER                                      PUMT0660
           NPAG=(LINES+LINSPP-1)/LINSPP                                 PUMT0670
C       AND ITS LAST LINE                                               PUMT0680
           LINES=NPAG*LINSPP                                            PUMT0690
           LNSXPG=LINES+LINSPP-1                                        PUMT0700
           NPAG=NPAG+1                                                  PUMT0710
           CALL PUTHDR(NUNIT,'1',NPAG)                                  PUMT0720
           WRITE(NUNIT,'(1X)')                                          PUMT0730
           LINES=LINES+1                                                PUMT0740
        ENDIF                                                           PUMT0750
        END                                                             PUMT0760
```

```
        SUBROUTINE OPNREP(IUN,FNAME)                                    OPIP0020
C       THIS OPEN OPERATION ALLOWS FOR AN EXISTING VERSION OF           OPIP0030
C       A FILE. IF THE FILE EXISTS IT IS OVERWRITTEN.                   OPIP0040
C       THE LOGIC USED IS HOPEFULLY MACHINE INDEPENDENT.                OPIP0050
        CHARACTER*(*) FNAME                                             OPIP0060
        CALL ZAPFIL(IUN,FNAME)                                          OPIP0070
        OPEN(IUN,FILE=FNAME,STATUS='NEW')                               OPIP0080
        REWIND IUN                                                      OPIP0090
        RETURN                                                          OPIP0100
        ENTRY OPNURP(IUN,FNAME)                                         OPIP0110
        CALL ZAPFIL(IUN,FNAME)                                          OPIP0120
        OPEN(IUN,FILE=FNAME,STATUS='NEW',FORM='UNFORMATTED')            OPIP0130
        REWIND IUN                                                      OPIP0140
        END                                                             OPIP0150
```

```
        SUBROUTINE ZAPFIL(IUN,FNAME)                                    OPIP0160
C       TEST FOR EXISTENCE OF FILE AND DELETE                           OPIP0170
        CHARACTER FNAME*(*)                                             OPIP0180
        LOGICAL EX                                                      OPIP0190
        INQUIRE(FILE=FNAME,EXIST=EX)                                    OPIP0200
        IF(EX) THEN                                                     OPIP0210
           OPEN(UNIT=IUN,FILE=FNAME,STATUS='OLD')                       OPIP0220
           CLOSE(UNIT=IUN,STATUS='DELETE')                              OPIP0230
```

```
      ENDIF                                                              OPIP0240
      END                                                                OPIP0250



      SUBROUTINE OVERRD(VAR,TST,DEFT,NFLG,NFVLEQ,NFVLNE)                 OVTD0020
C                                                                        OVTD0030
C     OVERRIDE SUPPORT ROUTINE                                           OVTD0040
C     TEST 'VAR' AGAINST 'TST', IF EQUAL SET TO DEFAULT 'DEFT'           OVTD0050
C     AND ALSO SET INTEGER FLAG 'NFLG' TO VALUE 'NFVLEQ' ELSE 'NFVLNE'   OVTD0060
C                                                                        OVTD0070
      IF(VAR.EQ.TST) THEN                                                OVTD0080
         VAR=DEFT                                                        OVTD0090
         NFLG=NFVLEQ                                                     OVTD0100
      ELSE                                                               OVTD0110
         NFLG=NFVLNE                                                     OVTD0120
      ENDIF                                                              OVTD0130
      RETURN                                                             OVTD0140
      END                                                                OVTD0150



      SUBROUTINE SFILTR(C,S,KSET)                                        SFUR0020
C     FILTERS EXTRANEOUS CHARACTERS FROM PLOT LABELS                     SFUR0030
      CHARACTER*(*) C,S,KSET                                             SFUR0040
C                                                                        SFUR0050
      LN=LEN(C)                                                          SFUR0060
      J=0                                                                SFUR0070
      DO 10 I=1,LN                                                       SFUR0080
        IF(INDEX(KSET,C(I:I)).EQ.0) THEN                                 SFUR0090
           J=J+1                                                         SFUR0100
           S(J:J)=C(I:I)                                                 SFUR0110
        ENDIF                                                            SFUR0120
10    CONTINUE                                                           SFUR0130
      RETURN                                                             SFUR0140
      END                                                                SFUR0150



      FUNCTION ALCOSH(X)                                                 ALRH0020
C     COMPUTE LOG(COSH(X)) AND USE LARGE ARGUMENT APPROXIMATION          ALRH0030
C     WHEN POSSIBLE.                                                     ALRH0040
      DATA ALOG2/.6931471806/                                            ALRH0050
C                                                                        ALRH0060
      IF(ABS(X).GT.50.0) GO TO 10                                        ALRH0070
      EX=EXP(X)                                                          ALRH0080
      ALCOSH=ALOG((EX+1.0/EX)*.5)                                        ALRH0090
      RETURN                                                             ALRH0100
10    ALCOSH=ABS(X)-ALOG2                                                ALRH0110
```

```
      END                                                               ALRH0120



      SUBROUTINE GAUSEL (C,NRD,NRR,NCC,NSF)                             GAKL0020
C     MATRIX INVERSION BY METHOD OF GAUSSIAN PIVOT                      GAKL0030
C     SOLVES NCC-NRR SETS OF NRR EQUATIONS                              GAKL0040
C********* SAME AS SUBROUTINE GAUSSEL WRITTEN BY L. DAVID LEWIS ********GAKL0050
      DIMENSION C(NRD,NCC),L(128,2)                                     GAKL0060
C          BITS = 2.**-18                                               GAKL0070
      DATA BITS/3.8146972656E-6/                                        GAKL0080
      NR=NRR                                                            GAKL0090
      NC=NCC                                                            GAKL0100
      IF(NC.LT.NR.OR.NR.GT.128.OR.NR.LE.0) CALL EXIT                    GAKL0110
C                                                                       GAKL0120
C          INITIALIZE.                                                  GAKL0130
      NSF=0                                                             GAKL0140
      NRM=NR-1                                                          GAKL0150
      NRP=NR+1                                                          GAKL0160
      D=1.                                                              GAKL0170
      LSD=1                                                             GAKL0180
      DO 1 KR=1,NR                                                      GAKL0190
      L(KR,1)=KR                                                        GAKL0200
1     L(KR,2)=0                                                         GAKL0210
      IF(NR.EQ.1) GO TO 42                                              GAKL0220
C                                                                       GAKL0230
C          ELIMINATION PHASE.                                           GAKL0240
      DO 41 KP=1,NRM                                                    GAKL0250
      KPP=KP+1                                                          GAKL0260
      PM=0.                                                             GAKL0270
      MPN=0                                                             GAKL0280
C                                                                       GAKL0290
C          SEARCH COLUMN KP FROM DIAGONAL DOWN FOR MAX PIVOT.           GAKL0300
      DO 2 KR=KP,NR                                                     GAKL0310
      LKR=L(KR,1)                                                       GAKL0320
      PT=ABS(C(LKR,KP))                                                 GAKL0330
      IF(PT.LE.PM) GO TO 2                                              GAKL0340
      PM=PT                                                             GAKL0350
      MPN=KR                                                            GAKL0360
      LMP=LKR                                                           GAKL0370
2     CONTINUE                                                          GAKL0380
C                                                                       GAKL0390
C          IF MAX PIVOT IS ZERO, MATRIX IS SINGULAR.                    GAKL0400
      IF(MPN.EQ.0) GO TO 9                                              GAKL0410
      NSF=NSF+1                                                         GAKL0420
      IF(MPN.EQ.KP) GO TO 3                                             GAKL0430
C                                                                       GAKL0440
C          NEW ROW NUMBER KP HAS MAX PIVOT.                             GAKL0450
      LSD=-LSD                                                          GAKL0460
      L(KP,2)=L(KP,1)                                                   GAKL0470
      L(MPN,1)=L(KP,1)                                                  GAKL0480
      L(KP,1)=LMP                                                       GAKL0490
C                                                                       GAKL0500
```

```
C          ROW OPERATIONS TO ZERO COLUMN KP BELOW DIAGONAL.                GAKL0510
3       MKP=L(KP,1)                                                       GAKL0520
        P=C(MKP,KP)                                                       GAKL0530
        D=D*P                                                             GAKL0540
        DO 41 KR=KPP,NR                                                   GAKL0550
        MKR=L(KR,1)                                                       GAKL0560
        Q=C(MKR,KP)/P                                                     GAKL0570
        IF(Q.EQ.0.) GO TO 41                                              GAKL0580
C                                                                         GAKL0590
C          SUBTRACT Q * PIVOT ROW FROM ROW KR.                            GAKL0600
        DO 4 LC=KPP,NC                                                    GAKL0610
        R=Q*C(MKP,LC)                                                     GAKL0620
        C(MKR,LC)=C(MKR,LC)-R                                             GAKL0630
4       IF(ABS(C(MKR,LC)).LT.ABS(R)*BITS) C(MKR,LC)=0.                    GAKL0640
41      CONTINUE                                                          GAKL0650
C                                                                         GAKL0660
C          LOWER RIGHT HAND CORNER.                                       GAKL0670
42      LNR=L(NR,1)                                                       GAKL0680
        P=C(LNR,NR)                                                       GAKL0690
        IF(P.EQ.0.) GO TO 9                                               GAKL0700
        NSF=NSF+1                                                         GAKL0710
        D=D*P*LSD                                                         GAKL0720
        IF(NR.EQ.NC) GO TO 8                                              GAKL0730
C                                                                         GAKL0740
C          BACK SOLUTION PHASE.                                           GAKL0750
        DO 61 MC=NRP,NC                                                   GAKL0760
        C(LNR,MC)=C(LNR,MC)/P                                             GAKL0770
        IF(NR.EQ.1) GO TO 61                                              GAKL0780
        DO 6 LL=1,NRM                                                     GAKL0790
        KR=NR-LL                                                          GAKL0800
        MR=L(KR,1)                                                        GAKL0810
        KRP=KR+1                                                          GAKL0820
        DO 5 MS=KRP,NR                                                    GAKL0830
        LMS=L(MS,1)                                                       GAKL0840
        R=C(MR,MS)*C(LMS,MC)                                              GAKL0850
        C(MR,MC)=C(MR,MC)-R                                               GAKL0860
5       IF(ABS(C(MR,MC)).LT.ABS(R)*BITS) C(MR,MC)=0.                      GAKL0870
6       C(MR,MC)=C(MR,MC)/C(MR,KR)                                        GAKL0880
61      CONTINUE                                                          GAKL0890
C                                                                         GAKL0900
C          SHUFFLE SOLUTION ROWS BACK TO NATURAL ORDER.                   GAKL0910
        DO 71 LL=1,NRM                                                    GAKL0920
        KR=NR-LL                                                          GAKL0930
        MKR=L(KR,2)                                                       GAKL0940
        IF(MKR.EQ.0) GO TO 71                                             GAKL0950
        MKP=L(KR,1)                                                       GAKL0960
        DO 7 LC=NRP,NC                                                    GAKL0970
        Q=C(MKR,LC)                                                       GAKL0980
        C(MKR,LC)=C(MKP,LC)                                               GAKL0990
7       C(MKP,LC)=Q                                                       GAKL1000
71      CONTINUE                                                          GAKL1010
C                                                                         GAKL1020
C          NORMAL AND SINGULAR RETURNS.  GOOD SOLUTION COULD HAVE D=0.    GAKL1030
8       C(1,1)=D                                                          GAKL1040
        GO TO 91                                                          GAKL1050
```

```
9       C(1,1)=0.                                                       GAKL1060
   91 RETURN                                                            GAKL1070
         END                                                            GAKL1080


      SUBROUTINE RAYPLT                                                 RANT0020
C     MAIN PLOTTING PROGRAM; INITIALIZES, READS INPUT, PLOTS            RANT0030
C     PROJECTIONS OF RAYS ON A VERTICAL OR HORIZONTAL PLANE.            RANT0040
C        ABS(PLT)=1. PLOTS PROJECTION OF RAYPATH ON VERTICAL PLANE      RANT0050
C                    RECTANGULAR EXPANSION BY FACTOR 'PFACTR'           RANT0060
C                =2. PLOTS PROJECTION OF RAYPATH ON GROUND              RANT0070
C                =3. VERICAL PROJECTION USING RADIAL EXPANSION BY FACTO RANT0080
C                    'PFACTR'                                           RANT0090
C     COMMON DECK "FILEC" INSERTED HERE                                 CFIL0020
      COMMON /FILEC/NPLTDP                                              CFIL0040
      COMMON /PLT/ XL,XR,YB,YT,PRESET                                   RANT0110
      COMMON/PLT/RMIN,RMAX,ALPHA,APLT                                   RANT0120
C                                                                       RANT0130
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "FLAG" INSERTED HERE                                  CFLA0020
      LOGICAL NEWWR,NEWWP,NEWTRC,PENET                                  CFLA0040
      COMMON /FLG/ NTYP,NEWWR,NEWWP,NEWTRC,PENET,LINES,IHOP,HPUNCH      CFLA0050
      COMMON/FLGP/NSET                                                  CFLA0060
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND         125-149                                        CWW30070
```

```
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)          CWW30080
                                                                          CWW30090
C
C     SOUND SPEED 150-174                                                 CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)             CWW30110
      EQUIVALENCE (W(153),REFC)                                           CWW30120
                                                                          CWW30130
C
C     DELTA SOUND SPEED 175-199                                           CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)          CWW30150
                                                                          CWW30160
C
C     TEMPERATURE    200-224                                              CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)             CWW30180
                                                                          CWW30190
C
C     DELTA TEMPERATURE    225-249                                        CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)          CWW30210
                                                                          CWW30220
C
C     MOLECULAR   250-274                                                 CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)             CWW30240
                                                                          CWW30250
C
C     RECEIVER HEIGHT   275-299                                           CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)             CWW30270
                                                                          CWW30280
C
C     TOPOGRAPHY   300-324                                                CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)             CWW30300
                                                                          CWW30310
C
C     DELTA TOPOGRAPHY   325-349                                          CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)          CWW30330
                                                                          CWW30340
C
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                            CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)             CWW30360
                                                                          CWW30370
C
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                            CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)          CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                CWW30400
C                                                                         CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                         CWW30420
C     ABSORPTION         500-524                                          CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)             CWW30440
                                                                          CWW30450
C
C     DELTA ABSORPTION         525-549                                    CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)          CWW30470
                                                                          CWW30480
C
C     PRESSURE         550-574                                            CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)             CWW30500
                                                                          CWW30510
C
C     DELTA PRESSURE         575-599                                      CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)          CWW30530
C                                                                         CWW30540
      REAL LTIC                                                           RANT0180
      INTEGER IWDMP(411)                                                  RANT0190
      EQUIVALENCE(IWDMP,ID)                                               RANT0200
C                                                                         RANT0210
      EXTERNAL PLOT,PLTLB,PLTLBN,PLTHLB                                   RANT0220
C                                                                         RANT0230
      DATA NWDSRK/4/                                                      RANT0240
C                                                                         RANT0250
```

```
      IF (.NOT.NEWWR) GO TO 8                                          RANT0260
      APLT=ABS(PLT)                                                    RANT0270
C                                                                      RANT0280
C          NEW W ARRAY -- REINITIALIZE                                 RANT0290
      NEWWR=.FALSE.                                                    RANT0300
      PRESET=1.                                                        RANT0310
C     IF NO ACTIVE PLOTTING,WE ARE STILL DUMPING DATA                  RANT0320
      IF(PLT.EQ.0) GO TO 5                                             RANT0330
C     INITIALIZE ANNOTATION MODEL                                      RANT0340
C  CONVERT COORDINATES OF VERTICAL PLANE FROM GEOGRAPHIC TO GEOMAGNETIC RANT0350
      SW=SIN (PLAT)                                                    RANT0360
      CW=SIN (PID2-PLAT)                                               RANT0370
      SLM=SIN (LLAT)                                                   RANT0380
      CLM=SIN (PID2-LLAT)                                              RANT0390
      SRM=SIN (RLAT)                                                   RANT0400
      CRM=SIN (PID2-RLAT)                                              RANT0410
      CDPHI=SIN (PID2-(LLON-PLON))                                     RANT0420
      PHL=ATAN2(SIN (LLON-PLON)*CLM,CDPHI*SW*CLM-CW*SLM)               RANT0430
      CTHL=CDPHI*CW*CLM+SW*SLM                                         RANT0440
      STHL=SIN (ACOS (CTHL))                                           RANT0450
      CDPHI=SIN (PID2-(RLON-PLON))                                     RANT0460
      PHR=ATAN2(SIN (RLON-PLON)*CRM,CDPHI*SW*CRM-CW*SRM)               RANT0470
      CTHR=CDPHI*CW*CRM+SW*SRM                                         RANT0480
      STHR=SIN (ACOS (CTHR))                                           RANT0490
      CLR=CTHL*CTHR+STHL*STHR*SIN (PID2-(PHL-PHR))                     RANT0500
      ALPHA=.5*ACOS (CLR)                                              RANT0510
      SLR=SQRT (1.-CLR**2)                                             RANT0520
C                                                                      RANT0530
      IF (APLT.EQ.2.) GO TO 3                                          RANT0540
C                                                                      RANT0550
C     VERTICAL PROJECTIONS ONLY                                        RANT0560
      T=HB                                                             RANT0570
      HB=AMIN1(T,HT)                                                   RANT0580
      HT=AMAX1(T,HT)                                                   RANT0590
      RMIN=EARTHR+HB                                                   RANT0600
      YT1=YT                                                           RANT0610
      RMAX=EARTHR+HT                                                   RANT0620
C                                                                      RANT0630
      IF(APLT.NE.4.) GO TO 100                                         RANT0640
C                                                                      RANT0650
C     SCALE FOR CARTESIAN PROJECTION(=4)                               RANT0660
      XR=ALPHA                                                         RANT0670
      XR1=XR                                                           RANT0680
      XL=-XR                                                           RANT0690
      XL1=XL                                                           RANT0700
      YB=RMIN                                                          RANT0710
      YT=RMAX                                                          RANT0720
      GO TO 5                                                          RANT0730
C                                                                      RANT0740
C     PROJECTIONS 1 AND 3                                              RANT0750
100   XR1=RMIN*SIN (ALPHA)                                             RANT0760
      XL1=-XR1                                                         RANT0770
      YB=RMIN*SIN (PID2-ALPHA)                                         RANT0780
      IF(APLT.EQ.3.) RMAX=RMIN+(RMAX-RMIN)*PFACTR                      RANT0790
      XR=AMAX1(RMAX*SIN(ALPHA),(RMAX-YB)/2.)                           RANT0800
```

```
      XL=-XR                                                            RANT0810
      YT1=RMAX*(YB/RMIN)                                                RANT0820
      YT=2.0*XR                                                         RANT0830
      IF(APLT.EQ.1) YT=YT/PFACTR                                        RANT0840
      YT=YT+YB                                                          RANT0850
      GO TO 5                                                           RANT0860
C                                                                       RANT0870
C     HORIZONTAL PROJECTION(=2)                                         RANT0880
3     ALPH1=ATAN2(STHR*SIN (PHR-PHL),(CTHR-CTHL*CLR)/STHL)              RANT0890
      XL=0.0                                                            RANT0900
      XL1=0.0                                                           RANT0910
      XR=EARTHR*2.0*ALPHA                                               RANT0920
      XR1=XR                                                            RANT0930
C USE 90% OF X-RANGE FOR Y-RANGE                                        RANT0940
      RMAX=0.5*(0.90*XR)/PFACTR                                         RANT0950
      YT=RMAX                                                           RANT0960
      RMIN=-RMAX                                                        RANT0970
      YB=RMIN                                                           RANT0980
C                                                                       RANT0990
5     IF(NPLTDP.LE.0) GO TO 8                                           RANT1000
      WRITE(NPLTDP) 1,NWDSRK,1,411                                      RANT1010
      WRITE(NPLTDP) (IWDMP(I),I=1,411)                                  RANT1020
C                                                                       RANT1030
8     NEW=0                                                             RANT1040
      IF (NEWTRC) NEW=1                                                 RANT1050
      NEWTRC=.FALSE.                                                    RANT1060
C                                                                       RANT1070
      IF(NPLTDP.LE.0) GO TO 88                                          RANT1080
      IF(NEW.NE.1) GO TO 84                                             RANT1090
      WRITE(NPLTDP) 1,NWDSRK,17,25                                      RANT1100
      WRITE(NPLTDP) (IWDMP(I),I=17,25)                                  RANT1110
C                                                                       RANT1120
84    WRITE(NPLTDP) 3-NEW,R,TH,PH                                       RANT1130
C                                                                       RANT1140
88    IF(PLT.EQ.0) RETURN                                               RANT1150
C                                                                       RANT1160
      STH=SIN (TH)                                                      RANT1170
      CTH=SIN (PID2-TH)                                                 RANT1180
      CR=CTHR*CTH+STHR*STH*SIN (PID2-(PHR-PH))                          RANT1190
      CL=CTHL*CTH+STHL*STH*SIN (PID2-(PHL-PH))                          RANT1200
      CEA=ATAN2(CR-CL*CLR,CL*SLR)                                       RANT1210
C                                                                       RANT1220
      IF(APLT.NE.4.) GO TO 150                                          RANT1230
      CALL PLOT(CEA-ALPHA,R,NEW)                                        RANT1240
      RETURN                                                            RANT1250
C                                                                       RANT1260
150   IF(APLT.EQ.2.) GO TO 10                                           RANT1270
      RX=R                                                              RANT1280
      IF(APLT.EQ.3.) RX=RMIN+(R-RMIN)*PFACTR                            RANT1290
      CALL PLOT(CEA-ALPHA,RX,NEW)                                       RANT1300
      RETURN                                                            RANT1310
C                                                                       RANT1320
10    SL=SQRT(AMAX1(0.,1.-CL**2))                                       RANT1330
      TMP1=STH*SIN (PH-PHL)                                             RANT1340
      TMP2=(CTH-CTHL*CL)/STHL                                           RANT1350
```

```
      ALPH2=0.                                                          RANT1360
      IF (TMP1.NE.0..OR.TMP2.NE.0.) ALPH2=ATAN2(TMP1,TMP2)              RANT1370
      CALL PLOT (EARTHR*CEA,EARTHR*ASIN(SL*SIN (ALPH1-ALPH2)),NEW)      RANT1380
      RETURN                                                            RANT1390
C                                                                       RANT1400
C          DRAW AXES AND CALL FOR LABELING AND TERMINATION OF THIS PLOT RANT1410
      ENTRY ENDPLT                                                      RANT1420
C                                                                       RANT1430
C     IF NEWWR IS STILL TRUE, NO PLOTS WHERE PRODUCED                   RANT1440
      IF(NEWWR) RETURN                                                  RANT1450
      NEWWR=.TRUE.                                                      RANT1460
C                                                                       RANT1470
C     SIGNAL END OF PLOT                                                RANT1480
      IF(NPLTDP.GT.0) WRITE(NPLTDP) 4,(NWDSRK,I=2,NWDSRK)               RANT1490
C                                                                       RANT1500
      IF(PLT.EQ.0) RETURN                                               RANT1510
C                                                                       RANT1520
      TICKX=0.01*(YT-YB)                                                RANT1530
      DTIC=TIC*EARTHR                                                   RANT1540
      CALL SETXY(APLT,-ALPHA,RMIN,ALPHA,RMAX)                           RANT1550
C                                                                       RANT1560
      IF(APLT.EQ.4.) GO TO 200                                          RANT1570
      IF(APLT.EQ.2.) GO TO 25                                           RANT1580
C                                                                       RANT1590
C     CURVLINEAR PROJECTIONS(=1,3)                                      RANT1600
      CALL ARCTIC(-ALPHA,ALPHA,RMIN,TICKX,PLTHLB)                       RANT1610
      CALL ARCTIC(-ALPHA,ALPHA,RMAX,-TICKX,PLTHLB)                      RANT1620
      GO TO 300                                                         RANT1630
C                                                                       RANT1640
C     CARTESIAN PROJECTION PUT IN BOTTOM BOUNDARY                       RANT1650
200   CALL TIKLINE(-ALPHA,RMIN,ALPHA,RMIN,TIC,-TICKX,PLTHLB)            RANT1660
C     PUT IN TOP BOUNDARY                                               RANT1670
      CALL TIKLINE(-ALPHA,RMAX,ALPHA,RMAX,TIC,TICKX,PLTHLB)             RANT1680
C                                                                       RANT1690
300   TIKL=.02*ALPHA                                                    RANT1700
      LTIC=TICV                                                         RANT1710
      IF(APLT.EQ.3.) LTIC=TICV*PFACTR                                   RANT1720
C     PUT IN LEFT BOUNDARY                                              RANT1730
      CALL TIKLINE(-ALPHA,RMIN,-ALPHA,RMAX,LTIC,TIKL,PLTLB)             RANT1740
C     PUT IN RIGHT BOUNDARY                                             RANT1750
      CALL TIKLINE(ALPHA,RMIN,ALPHA,RMAX,LTIC,-TIKL,PLTLBN)             RANT1760
C                                                                       RANT1770
      GO TO 50                                                          RANT1780
C   DRAW TICKS, BOX FOR HORIZONTAL PLOT                                 RANT1790
25    CALL DRAWTKS(DTIC,TICV,XL,XR,YB,YT,PLOT)                          RANT1800
C                                                                       RANT1810
50    IF(APLT.EQ.2.0) THEN                                              RANT1820
        CALL PLTANOT(ID,OW/PIT2,XFQMDL,YFQMDL,XL,YB,XR,YT,DEGS,         RANT1830
     2     LLAT,LLON,RLAT,RLON,HB,HT,APLT,DTIC,DTIC/PFACTR,PLOT)        RANT1840
      ELSE                                                              RANT1850
        CALL PLTANOT(ID,OW/PIT2,XFQMDL,YFQMDL,-ALPHA,RMIN,ALPHA,RMAX,   RANT1860
     2     DEGS,LLAT,LLON,RLAT,RLON,HB,HT,APLT,DTIC,TICV,PLOT)          RANT1870
      ENDIF                                                             RANT1880
C                                                                       RANT1890
      CALL LABPLT                                                       RANT1900
```

```
      CALL PLTEND                                                       RANT1910
      RETURN                                                            RANT1920
      END                                                               RANT1930
```

```
      SUBROUTINE PLOT (X,Y,NEW)                                         PLOT0020
C     PLOTS ONE VECTOR FROM CURRENT PLOT POSITION TO POINT(X,Y)         PLOT0030
C     TAKING BORDER CROSSINGS INTO ACCOUNT.                             PLOT0040
      COMMON /PLT/ XMIN0,XMAX0,YMIN0,YMAX0,RESET                        PLOT0050
      COMMON/PLT/RMIN,RMAX,ALPHA,APLT                                   PLOT0060
      INTEGER REZFLG                                                    DDHI0020
      PARAMETER (REZFLG=2001)                                           DDHI0030
      COMMON/DDREZ/DDHIX,DDHIY                                          DDHI0040
      COMMON /DD/ INT,IOR,IT,IS,IC,ICC,IX,IY                            PLOT0080
C DEFINE NOMINAL PLOTTING AREA(ZERO SUFFIXES) AND AN                    PLOT0090
C OUTER CLIPPING BOUNDARY BEYOND WHICH NO VECTORS EXTEND.               PLOT0100
      DATA XOLD,YOLD/0.0,0.0/                                           PLOT0110
C 90% FOR Y RANGE                                                       PLOT0120
C                                                                       PLOT0130
C           COMPUTE SCALE FACTORS                                       PLOT0140
    1 IF (RESET.EQ.0.) GO TO 5                                          PLOT0150
      RESET=0.                                                          PLOT0160
      IF(APLT.EQ.2.) THEN                                               PLOT0170
         MRNGE=723                                                      PLOT0180
         MINX0=165                                                      PLOT0190
         MINY0=140                                                      PLOT0200
      ELSE                                                              PLOT0210
         MRNGE=813                                                      PLOT0220
         MINX0=165                                                      PLOT0230
         MINY0=140                                                      PLOT0240
      ENDIF                                                             PLOT0250
      IF(APLT.EQ.4.) MINY0=0                                            PLOT0260
C                                                                       PLOT0270
      MAXX0=MINX0+MRNGE                                                 PLOT0280
      MAXY0=MINY0+MRNGE                                                 PLOT0290
C                                                                       PLOT0300
      XSCALE=(MAXX0-MINX0)/(XMAX0-XMIN0)                                PLOT0310
      YSCALE=(MAXY0-MINY0)/(YMAX0-YMIN0)                                PLOT0320
      XMIN=XMIN0                                                        PLOT0330
      YMIN=YMIN0                                                        PLOT0340
      XMAX=XMAX0                                                        PLOT0350
      YMAX=YMAX0                                                        PLOT0360
      IF(APLT.EQ.2.) GO TO 5                                            PLOT0370
C                                                                       PLOT0380
      XMIN=-ALPHA                                                       PLOT0390
      XMAX=ALPHA                                                        PLOT0400
      YMIN=RMIN                                                         PLOT0410
      YMAX=RMAX                                                         PLOT0420
      IF(APLT.NE.4) GO TO 5                                             PLOT0430
      YSCALE=.85*YSCALE                                                 PLOT0440
      MINY0=MINY0+60                                                    PLOT0450
C                                                                       PLOT0460
```

```
C          START A NEW LINE                                                    PLOT0470
C          HORIZONTAL DISPLACEMENT                                             PLOT0480
5       XS=X-XOLD                                                              PLOT0490
        YS=Y-YOLD                                                              PLOT0500
        S=1.0                                                                  PLOT0510
        IF(NEW.EQ.0) GO TO 10                                                  PLOT0520
        IF(X.GE.XMIN.AND.X.LE.XMAX.AND.Y.GE.YMIN.AND.Y.LE.YMAX) GO TO 48       PLOT0530
        GO TO 50                                                               PLOT0540
C                                                                              PLOT0550
10      IF (XS) 11,12,16                                                       PLOT0560
C            NEGATIVE                                                          PLOT0570
   11   X1=XMAX                                                                PLOT0580
        X2=XMIN                                                                PLOT0590
        GO TO 20                                                               PLOT0600
C            ZERO                                                              PLOT0610
   12   IF (YS) 13,50,14                                                       PLOT0620
   13   S1=(YMAX-YOLD)/YS                                                      PLOT0630
        S2=(YMIN-YOLD)/YS                                                      PLOT0640
        GO TO 40                                                               PLOT0650
   14   S1=(YMIN-YOLD)/YS                                                      PLOT0660
        S2=(YMAX-YOLD)/YS                                                      PLOT0670
        GO TO 40                                                               PLOT0680
C            POSITIVE                                                          PLOT0690
   16   X1=XMIN                                                                PLOT0700
        X2=XMAX                                                                PLOT0710
C                                                                              PLOT0720
C          VERTICAL DISPLACEMENT                                               PLOT0730
   20   IF (YS) 21,22,26                                                       PLOT0740
C            NEGATIVE                                                          PLOT0750
   21   Y1=YMAX                                                                PLOT0760
        Y2=YMIN                                                                PLOT0770
        GO TO 30                                                               PLOT0780
C            ZERO                                                              PLOT0790
   22   S1=(X1-XOLD)/XS                                                        PLOT0800
        S2=(X2-XOLD)/XS                                                        PLOT0810
        GO TO 40                                                               PLOT0820
C            POSITIVE                                                          PLOT0830
   26   Y1=YMIN                                                                PLOT0840
        Y2=YMAX                                                                PLOT0850
C                                                                              PLOT0860
   30   S1=AMAX1((X1-XOLD)/XS,(Y1-YOLD)/YS)                                    PLOT0870
        S2=AMIN1((X2-XOLD)/XS,(Y2-YOLD)/YS)                                    PLOT0880
C                                                                              PLOT0890
C          PLOT LINE -- CHECKING FOR BORDER CROSSINGS                          PLOT0900
40      IF (S2.LT.0..OR.S1.GT.1.) GO TO 50                                     PLOT0910
        IF (S1.LT.0.) GO TO 42                                                 PLOT0920
C          PREVIOUS POINT OFF GRAPH                                            PLOT0930
        XP=XOLD+XS*S1                                                          PLOT0940
        YP=YOLD+YS*S1                                                          PLOT0950
        IF(APLT.EQ.2.0.OR.APLT.EQ.4.0) GO TO 41                                PLOT0960
        T=XP                                                                   PLOT0970
        XP=YP*SIN(T)                                                           PLOT0980
        YP=YP*COS(T)                                                           PLOT0990
C                                                                              PLOT1000
41      DDHIX=MINX0+(XP-XMIN0)*XSCALE+0.5                                      PLOT1010
```

```
      DDHIY=MINY0+(YP-YMIN0)*YSCALE+0.5                                  PLOT1020
C     USE SPECIAL HI-REZ MODE                                            PLOT1030
      IX=REZFLG                                                          PLOT1040
      CALL DDBP                                                          PLOT1050
      GO TO 48                                                           PLOT1060
C                                                                        PLOT1070
42    IF (S2.GT.1.) GO TO 48                                             PLOT1080
C          CURRENT POINT OFF GRAPH                                       PLOT1090
      S=S2                                                               PLOT1100
C          CURRENT POINT ON GRAPH                                        PLOT1110
48    XP=XOLD+XS*S                                                       PLOT1120
      YP=YOLD+YS*S                                                       PLOT1130
      IF(APLT.EQ.2.0.OR.APLT.EQ.4.0) GO TO 49                            PLOT1140
      T=XP                                                               PLOT1150
      XP=YP*SIN(T)                                                       PLOT1160
      YP=YP*COS(T)                                                       PLOT1170
49    DDHIX=MINX0+(XP-XMIN0)*XSCALE+0.5                                  PLOT1180
      DDHIY=MINY0+(YP-YMIN0)*YSCALE+0.5                                  PLOT1190
C     USE SPECIAL HI-REZ MODE                                            PLOT1200
      IX=REZFLG                                                          PLOT1210
      IF(NEW.EQ.0) CALL DDVC                                             PLOT1220
      IF(NEW.NE.0) CALL DDBP                                             PLOT1230
C                                                                        PLOT1240
C          EXIT ROUTINE                                                  PLOT1250
   50 XOLD=X                                                             PLOT1260
      YOLD=Y                                                             PLOT1270
      RETURN                                                             PLOT1280
C                                                                        PLOT1290
C          TERMINATE THE CURRENT PLOT                                    PLOT1300
      ENTRY PLTEND(X,Y,NEW)                                              PLOT1310
      CALL DDFR                                                          PLOT1320
C                                                                        PLOT1330
      RETURN                                                             PLOT1340
         END                                                             PLOT1350
```

```
      SUBROUTINE LABPLT                                                  LAOT0020
C          LABEL THE CURRENT PLOT                                        LAOT0030
      CHARACTER*80 LABEL,CHID                                            LAOT0040
      CHARACTER*4 ANGRANG,ANOTE                                          LAOT0050
C     COMMON DECK "LL" INSERTED HERE                                     CLL 0020
      REAL MODL                                                          CLL 0040
      COMMON/LL/MODL(4),APH,APHPT,APHPR,APHPTH,APHPPH                    CLL 0050
C     COMMON DECK "SS" INSERTED HERE                                     CSS 0020
      REAL MODSURF                                                       CSS 0040
      COMMON/SS/ MODSURF(4)                                              CSS 0050
      COMMON/SS/U,PUR,PURR,PURTH,PURPH                                   CSS 0060
      COMMON/SS/PUTH,PUPH,PUTHTH,PUPHPH,PUTHPH,USELECT,UTIME             CSS 0070
C     COMMON DECK "GG" INSERTED HERE                                     CGG 0020
      REAL MODG                                                          CGG 0040
      COMMON/GG/MODG(4)                                                  CGG 0050
      COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                   CGG 0060
```

```
      COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME                CGG 0070
C     COMMON DECK "RR" INSERTED HERE                                       CRR 0020
      REAL MODREC                                                          CRR 0040
      COMMON/RR/ MODREC(4)                                                 CRR 0050
      COMMON/RR/F,PFR,PFRR,PFRTH,PFRPH                                     CRR 0060
      COMMON/RR/PFTH,PFPH,PFTHTH,PFPHPH,PFTHPH,FSELECT,FTIME               CRR 0070
C     COMMON DECK "UU" INSERTED HERE                                       CUU 0020
      REAL MODU                                                            CUU 0040
      COMMON/UU/MODU(4)                                                    CUU 0050
     1  ,V  ,PVT  ,PVR  ,PVTH  ,PVPH                                       CUU 0060
     2  ,VR ,PVRT ,PVRR ,PVRTH ,PVRPH                                      CUU 0070
     3  ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                     CUU 0080
     4  ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                     CUU 0090
C     COMMON DECK "CC" INSERTED HERE                                       CCC 0020
      REAL MODC                                                            CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                           CCC 0050
C     COMMON DECK "TT" INSERTED HERE                                       CTT 0020
      REAL MODT                                                            CTT 0040
      COMMON/TT/MODT(4), T,PTT,PTR,PTTH,PTPH                               CTT 0050
C     COMMON DECK "MM" INSERTED HERE                                       CMM 0020
      REAL M,MODM                                                          CMM 0040
      COMMON/MM/MODM(4),M,PMT,PMR,PMTH,PMPH                                CMM 0050
C     COMMON DECK "HDR" INSERTED HERE                                      CHDR0020
      CHARACTER*10 INITID*80,DAT,TOD                                       CHDR0040
      COMMON/HDR/SEC                                                       CHDR0050
      COMMON/HDRC/INITID,DAT,TOD                                           CHDR0060
C     COMMON DECK "RINPLEX" INSERTED HERE                                  CRIN0020
      REAL KAY2,KAY2I                                                      CRIN0040
      COMPLEX PNP,POLAR,LPOLAR                                             CRIN0050
      LOGICAL SPACE                                                        CRIN0060
      CHARACTER DISPM*6                                                    CRIN0070
      COMMON/RINPL/DISPM                                                   CRIN0080
      COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                    CRIN0090
      COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I                                CRIN0100
      COMMON/RIN/PNP(10),POLAR,LPOLAR,SGN                                  CRIN0110
      COMMON /DD/ INT,IOR,IT,IS,IC,ICC,IX,IY                               LAOT0160
C     COMMON DECK "CONST" INSERTED HERE                                    CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                        CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                            CCON0050
C     COMMON DECK "FLAG" INSERTED HERE                                     CFLA0020
      LOGICAL NEWWR,NEWWP,NEWTRC,PENET                                     CFLA0040
      COMMON /FLG/ NTYP,NEWWR,NEWWP,NEWTRC,PENET,LINES,IHOP,HPUNCH         CFLA0050
      COMMON/FLGP/NSET                                                     CFLA0060
C     COMMON DECK "WW" INSERTED HERE                                       CWW 0020
      PARAMETER (NWARSZ=1000)                                              CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                      CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                   CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),       CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),         CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),             CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),            CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                    CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                        CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                        CWW20100
```

```
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                    CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),      CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))       CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),       CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))             CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                  CWW20160
      REAL MMODEL,MFORM,MID                                            CWW30020
C                                                                      CWW30030
C     WIND         100-124                                           CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)          CWW30050
C                                                                      CWW30060
C     DELTA WIND         125-149                                       CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)       CWW30080
C                                                                      CWW30090
C     SOUND SPEED 150-174                                              CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)          CWW30110
      EQUIVALENCE (W(153),REFC)                                        CWW30120
C                                                                      CWW30130
C     DELTA SOUND SPEED 175-199                                        CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)       CWW30150
C                                                                      CWW30160
C     TEMPERATURE    200-224                                           CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)          CWW30180
C                                                                      CWW30190
C     DELTA TEMPERATURE    225-249                                     CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)       CWW30210
C                                                                      CWW30220
C     MOLECULAR   250-274                                              CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)          CWW30240
C                                                                      CWW30250
C     RECEIVER HEIGHT   275-299                                        CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)          CWW30270
C                                                                      CWW30280
C     TOPOGRAPHY   300-324                                             CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)          CWW30300
C                                                                      CWW30310
C     DELTA TOPOGRAPHY   325-349                                       CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)       CWW30330
C                                                                      CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                         CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)          CWW30360
C                                                                      CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                         CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)       CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                             CWW30400
C                                                                      CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                      CWW30420
C     ABSORPTION         500-524                                       CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)          CWW30440
C                                                                      CWW30450
C     DELTA ABSORPTION         525-549                                 CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)       CWW30470
C                                                                      CWW30480
C     PRESSURE         550-574                                         CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)          CWW30500
```

```
C                                                                       CWW30510
C     DELTA PRESSURE          575-599                                   CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)        CWW30530
C                                                                       CWW30540
C                                                                       LAOT0200
C     COMMON DECK "AA" INSERTED HERE                                    CAA 0020
      REAL MODA                                                         CAA 0040
      REAL MU,MUPT,MUPR,MUPTH,MUPPH                                     CAA 0050
      REAL KAP,KAPPT,KAPPR,KAPPTH,KAPPPH                                CAA 0060
      COMMON/AA/MODA(4),MU,MUPT,MUPR,MUPTH,MUPPH                        CAA 0070
      COMMON/AA/KAP,KAPPT,KAPPR,KAPPTH,KAPPPH                           CAA 0080
C     COMMON DECK "PP" INSERTED HERE                                    CPP 0020
      REAL MODP                                                         CPP 0040
      COMMON/PP/MODP(4),P,PPT,PPR,PPTH,PPPH                             CPP 0050
C                                                                       LAOT0230
      WRITE(LABEL,900) ID                                               LAOT0240
900   FORMAT(10A8)                                                      LAOT0250
      CHID=LABEL                                                        LAOT0260
      LABEL=CHID(5:)                                                    LAOT0270
      IOR=0                                                             LAOT0280
      IT=0                                                              LAOT0290
      IS=2                                                              LAOT0300
      IX=0                                                              LAOT0310
      IY=1023                                                           LAOT0320
      CALL DDTEXT (8,LABEL)                                             LAOT0330
      IX=1090                                                           LAOT0340
      IY=0                                                              LAOT0350
      CALL DDTEXT (2,DAT)                                               LAOT0360
      IY=1023                                                           LAOT0370
      F=OW/PIT2                                                         LAOT0380
      NANGLE=10                                                         LAOT0390
      ANGRANG='EL ='                                                    LAOT0400
      ANOTE='AZ ='                                                      LAOT0410
      ANEL=0.                                                           LAOT0420
      IF(ELSTEP.NE.0.0) ANEL=(ELEND-ELBEG)/ELSTEP+1.5                   LAOT0430
      IF(ANEL.GT.1.0.OR.PLT.LT.0.0) GO TO 100                           LAOT0440
         ANGRANG='AZ ='                                                 LAOT0450
         ANOTE='EL ='                                                   LAOT0460
         NANGLE=14                                                      LAOT0470
100   WRITE(LABEL,1300)  CHID(1:3),F,ANOTE,W(NANGLE)*DEGS               LAOT0480
1300  FORMAT ('MODEL = ',A,'  ,FREQ =',F9.3,' HZ, ',A,F7.3,' DEG')      LAOT0490
      IX=0                                                              LAOT0500
      IY=IY-32                                                          LAOT0510
      CALL DDTEXT (7,LABEL)                                             LAOT0520
C                                                                       LAOT0530
C INDEX OF OPPOSITE ANGLE                                               LAOT0540
      NANGLE=(10+14-NANGLE)+1                                           LAOT0550
      NANGLE2=NANGLE+2                                                  LAOT0560
      WRITE(LABEL,1400) ANGRANG,(W(I)*DEGS,I=NANGLE,NANGLE2)            LAOT0570
1400  FORMAT(A,F7.2,' DEG TO',F7.2,' DEG, STEP =',F7.2,' DEG')          LAOT0580
      IY=IY-32                                                          LAOT0590
      CALL DDTEXT (7,LABEL)                                             LAOT0600
C                                                                       LAOT0610
      WRITE(LABEL,1500) XMTRH,TLAT*DEGS,TLON*DEGS                       LAOT0620
1500  FORMAT('XMTR HT =',F6.2,' KM ,LAT =',F6.2,' DEG, LONG ='          LAOT0630
```

```
     1   ,F6.2,' DEG')                                                    LAOT0640
      IY=IY-32                                                           LAOT0650
      CALL DDTEXT (7,LABEL)                                              LAOT0660
C                                                                        LAOT0670
      IY=IY-32                                                           LAOT0680
      WRITE(LABEL,'(10A8)') MODRIN                                       LAOT0690
      CALL DDTEXT(8,LABEL)                                               LAOT0700
C                                                                        LAOT0710
      IX=1050                                                            LAOT0720
      CALL DDTEXT(1,'MODELS')                                            LAOT0730
      IY=IY-15                                                           LAOT0740
C     LOOP FOR 8 MODELS AND PERTURBATIONS                                LAOT0750
      DO 1700 K=1,12                                                     LAOT0760
      I=(K-1)/2+1                                                        LAOT0770
C     GENERATE ALTERNATING 1,2;3,4 SERIES FOR MODEL AND PERTURBATION     LAOT0780
      J1=2*(K+1-I*2)+1                                                   LAOT0790
      J2=J1+1                                                            LAOT0800
      IF(I.EQ.1) WRITE(LABEL,1600) (MODU(J),J=J1,J2)                     LAOT0810
      IF(I.EQ.2) WRITE(LABEL,1600) (MODC(J),J=J1,J2)                     LAOT0820
C                                                                        LAOT0830
      IF(I.EQ.3) WRITE(LABEL,1600) (MODG(J),J=J1,J2)                     LAOT0840
      IF(I.EQ.4) WRITE(LABEL,1600) (MODL(J),J=J1,J2)                     LAOT0850
      IF(I.EQ.5 .AND. J1.EQ.1)                                           LAOT0860
     1     WRITE(LABEL,1600) (MODREC(J),J=J1,J2)                         LAOT0870
      IF(I.EQ.6) WRITE(LABEL,1600) (MODSURF(J),J=J1,J2)                  LAOT0880
1600  FORMAT(2(2(A8,2X,F5.1,1X),2X))                                     LAOT0890
C                                                                        LAOT0900
1610  IF(LABEL(1:1).EQ.' ') GO TO 1700                                   LAOT0910
      IY=IY-32                                                           LAOT0920
      CALL DDTEXT(2,LABEL)                                               LAOT0930
1700  LABEL(1:1)=' '                                                     LAOT0940
C                                                                        LAOT0950
         END                                                             LAOT0960
```

```
      SUBROUTINE PLTHLB(X,Y,NC)                                          PLJB0020
C     HORIZONTAL TICK ANNOTATION ROUTINE FOR RAYPLOT.                    PLJB0030
C                                                                        PLJB0040
C                                                                        PLJB0050
      EXTERNAL PLOT                                                      PLJB0060
C                                                                        PLJB0070
      CALL PLTANH(X,Y,NC,PLOT)                                           PLJB0080
      END                                                                PLJB0090
```

```
      SUBROUTINE PLTANH(X,Y,NC,PLOT)                                     PLRH0020
C                                                                        PLRH0030
C     TIC LABELING ROUTINE FOR HORIZONTAL PLOT PROJECTIONS               PLRH0040
C                                                                        PLRH0050
```

```
      CHARACTER ANNOT*10                                                PLRH0060
      COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                               PLRH0070
C     COMMON DECK "WWR" INSERTED HERE                                   CWWR0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
      COMMON/LABCLT/PROJCT,THMIN,THMAX,RMIN,RMAX                        PLRH0100
C                                                                       PLRH0110
      COMMON/RAYCON/MCONP                                               PLRH0120
C                                                                       PLRH0130
      DATA LNC/-100/                                                    PLRH0140
C                                                                       PLRH0150
      IF(NC.LE.0 .OR. NC.GT.2) GO TO 100                                PLRH0160
C                                                                       PLRH0170
C     NORMALIZE LETTER SIZE FACTOR TO .15 INCHES                        PLRH0180
      HLETF=HITLET/.15                                                  PLRH0190
C                                                                       PLRH0200
      IF(LNC.NE.NC) LICM=-100                                           PLRH0210
      LNC=NC                                                            PLRH0220
C                                                                       PLRH0230
      CALL PLOT(X,Y,1)                                                  PLRH0240
      IX=IX-80*HLETF                                                    PLRH0250
      ICM=IX                                                            PLRH0260
      IF(NC.GT.1) THEN                                                  PLRH0270
         ICM=IY                                                         PLRH0280
         IX=IX-40                                                       PLRH0290
      ENDIF                                                             PLRH0300
C                                                                       PLRH0310
C     INSURE THAT OVERLAPS OF ANNOTATIONS DO NOT OCCUR                  PLRH0320
      IF(IABS(ICM-LICM).LT.INT(80*HLETF)) GO TO 100                     PLRH0330
C                                                                       PLRH0340
      IF(PROJCT.EQ.2.0) GO TO 25                                        PLRH0350
      IF(HB.GE.0.0.AND.Y.GT.(RMIN+RMAX)/2) GO TO 100                    PLRH0360
      IF(HB.LT.0.0.AND.Y.LT.(RMIN+RMAX)/2) GO TO 100                    PLRH0370
C                                                                       PLRH0380
25    F=DEGS                                                            PLRH0390
      TMP=(THMAX-THMIN)*DEGS                                            PLRH0400
      IF(TMP.LT.10.) F =EARTHR                                          PLRH0410
```

```
      FT=F                                                              PLRH0420
      IF(PROJCT.EQ.2.0 .AND. MCONP.EQ.0) F=F/EARTHR                     PLRH0430
      V=X-THMIN                                                         PLRH0440
      IF(NC.GT.1) V=Y-(RMIN+RMAX)/2.                                    PLRH0450
      IF(ABS(AMOD(TIC*FT+.0001,1.)).GT..001) GO TO 60                   PLRH0460
C                                                                       PLRH0470
      WRITE(ANNOT,50) INT(V*F+SIGN(.5,V))                               PLRH0480
50    FORMAT(I5)                                                        PLRH0490
      GO TO 90                                                          PLRH0500
C                                                                       PLRH0510
60    WRITE(ANNOT,80) V*F                                               PLRH0520
80    FORMAT(F8.2)                                                      PLRH0530
C                                                                       PLRH0540
90    LICM=ICM                                                          PLRH0550
C     ALLOW 8 RASTERS FOR THE LETTER HEIGHT                             PLRH0560
      H=HB                                                              PLRH0570
      IF(PROJCT.EQ.2.0) H=0.                                            PLRH0580
      IF(NC.EQ.1) IY=IY-8*HLETF-SIGN(52.,H)                             PLRH0590
      IOR=0                                                             PLRH0600
95    FORMAT(3G13.6,2I5,1X,A10)                                         PLRH0610
99    FORMAT(2G13.6,5I5,1X,A10)                                         PLRH0620
      CALL DDTEXT(1,ANNOT)                                              PLRH0630
C                                                                       PLRH0640
100   CALL PLOT(X,Y,MIN0(1,NC))                                         PLRH0650
      END                                                               PLRH0660
```

```
      SUBROUTINE SETXY(PROJ,XMIN,YMIN,XMAX,YMAX)                        SEMY0020
C     PLOT INITIALIZATION; SETS PROJECTION PARAMETERS                   SEMY0030
C                                                                       SEMY0040
      COMMON/LABCLT/PROJCT,THMIN,THMAX,RMIN,RMAX                        SEMY0050
C                                                                       SEMY0060
C                                                                       SEMY0070
C   INITIAL ANNOTATION MODEL                                            SEMY0080
      CALL SETANN                                                       SEMY0090
C                                                                       SEMY0100
      PROJCT=PROJ                                                       SEMY0110
      THMIN=XMIN                                                        SEMY0120
      RMIN=YMIN                                                         SEMY0130
      THMAX=XMAX                                                        SEMY0140
      RMAX=YMAX                                                         SEMY0150
      END                                                               SEMY0160
```

```
      SUBROUTINE TIKLINE(XL1,YB,XL,YT1,TICV,TIKSZ,PLOT)                 TIYE0020
C     DRAWS STRAIGHT LINE WITH TICKS AT INTERVALS                       TIYE0030
C                                                                       TIYE0040
C                                                                       TIYE0050
      XDF=XL-XL1                                                        TIYE0060
```

```
      YDF=YT1-YB                                                        TIYE0070
      DST=SQRT(XDF*XDF+YDF*YDF)                                         TIYE0080
      TICE=DST                                                          TIYE0090
      T=0.0                                                             TIYE0100
      IF(TICV.EQ.0.0) GO TO 50                                          TIYE0110
      TICE=TICV                                                         TIYE0120
      T=TIKSZ/TICV                                                      TIYE0130
C                                                                       TIYE0140
50    TICVX=TICE*XDF/DST                                                TIYE0150
      TICVY=TICE*YDF/DST                                                TIYE0160
      TX=TICVY*T                                                        TIYE0170
      TY=-TICVX*T                                                       TIYE0180
      NTIC=1+DST/TICE                                                   TIYE0190
      CALL PLOT(XL1,YB,-1)                                              TIYE0200
      DO 100 I=0,NTIC-1                                                 TIYE0210
        X=XL1+I*TICVX                                                   TIYE0220
        Y=YB+I*TICVY                                                    TIYE0230
        CALL PLOT(X,Y,0)                                                TIYE0240
        CALL PLOT(X+TX,Y+TY,0)                                          TIYE0250
100   CALL PLOT(X,Y,1)                                                  TIYE0260
      CALL PLOT(XL,YT1,0)                                               TIYE0270
      CALL PLOT(XL,YT1,1)                                               TIYE0280
      END                                                               TIYE0290
```

```
      SUBROUTINE PLTANOT(ID,FREQ,XMF,YMF,XL,YB,XR,YT                    PLYT0020
     1    ,DEGS,LLAT,LLON,RLAT,RLON,ALTLOW,ALTHI,PLT,DTICH,DTICV,PLOT)  PLYT0030
C     PUTS STANDARD ANNOTATIONS ON PLOTS                                PLYT0040
C                                                                       PLYT0050
      COMMON/LABCLT/PROJCT,THMIN,THMAX,RMIN,RMAX                        PLYT0060
      REAL LLAT,LLON                                                    PLYT0070
      CHARACTER LABEL*80                                                PLYT0080
      COMMON /DD/ INT,IOR,IT,IS,IC,ICC,IX,IY                            PLYT0090
C     COMMON DECK "ANNOT" INSERTED HERE                                 ANNO0020
      CHARACTER*10 ANOTES,HNOTES                                        ANNO0040
      COMMON/ANNCTL/LENA(4),LENHA(3)                                    ANNO0050
      COMMON/ANNCTC/ANOTES(2,4),HNOTES(4,3)                             ANNO0060
C                                                                       PLYT0110
      IF(PLT.EQ.2. .OR. XMF.LE.0.0 .OR. YMF.LE.0.0) GO TO 45            PLYT0120
      CALL PLOT(XL+XMF*(XR-XL),YB+YMF*(YT-YB),1)                        PLYT0130
      IF(FREQ.LE.0.0) GO TO 30                                          PLYT0140
C                                                                       PLYT0150
      WRITE(LABEL,25) FREQ                                              PLYT0160
25    FORMAT('FREQ =',F9.3)                                             PLYT0170
      CALL DDTEXT(2,LABEL)                                              PLYT0180
      IY=IY+40                                                          PLYT0190
C                                                                       PLYT0200
30    WRITE(LABEL,35) ID                                                PLYT0210
35    FORMAT('MODEL = ',A3)                                             PLYT0220
      CALL DDTEXT(2,LABEL)                                              PLYT0230
C                                                                       PLYT0240
45    NSPY=97                                                           PLYT0250
```

```
      IOR=0                                                          PLYT0260
C                                                                    PLYT0270
C FIRST THE LOW ALTITUDE-LATITUDE ANNOTATION                         PLYT0280
50    CALL PLOT(XL,YB,1)                                             PLYT0290
      IX=IX-155                                                      PLYT0300
      IY=IY-NSPY                                                     PLYT0310
      WRITE(LABEL,1840) DEGS*LLON                                    PLYT0320
      CALL DDTEXT(2,LABEL)                                           PLYT0330
      WRITE(LABEL,1850) DEGS*LLAT                                    PLYT0340
      IY=IY-32                                                       PLYT0350
      CALL DDTEXT(2,LABEL)                                           PLYT0360
C                                                                    PLYT0370
C NEXT THE RIGHT LATITUDE ANNOTATION                                 PLYT0380
      CALL PLOT(XR,YB,1)                                             PLYT0390
      IX=IX-130                                                      PLYT0400
      IY=IY-NSPY                                                     PLYT0410
      WRITE(LABEL,1840) DEGS*RLON                                    PLYT0420
      CALL DDTEXT(2,LABEL)                                           PLYT0430
      WRITE(LABEL,1850) DEGS*RLAT                                    PLYT0440
      IY=IY-32                                                       PLYT0450
      CALL DDTEXT(2,LABEL)                                           PLYT0460
C                                                                    PLYT0470
      XMID=(XL+XR)/2.0                                               PLYT0480
      IF(PLT.EQ.2.0) GO TO 55                                        PLYT0490
C                                                                    PLYT0500
C PUT THE HORIZONTAL TIC LABEL                                       PLYT0510
      Y=YT                                                           PLYT0520
      IF(ALTLOW.GE.0.0) Y=YB                                         PLYT0530
      CALL PLOT(XMID,Y,1)                                            PLYT0540
C                                                                    PLYT0550
      IF(ALTLOW.LT.0.0) IY=IY+80                                     PLYT0560
      IF(ALTLOW.GE.0.0) IY=IY-95                                     PLYT0570
      GO TO 60                                                       PLYT0580
C                                                                    PLYT0590
55    CALL PLOT(XMID,YB,1)                                           PLYT0600
      IY=IY-95                                                       PLYT0610
C                                                                    PLYT0620
60    IX=IX-235                                                      PLYT0630
      NOTEA=1                                                        PLYT0640
      TMP=(THMAX-THMIN)*DEGS                                         PLYT0650
      IF(TMP.GT.10.) NOTEA=2                                         PLYT0660
C                                                                    PLYT0670
      CALL DDTEXT (LENHA(NOTEA),HNOTES(1,NOTEA))                     PLYT0680
C                                                                    PLYT0690
C PUT THE VERTICAL TIC LABEL                                         PLYT0700
      CALL PLOT(XL,(YB+YT)/2.0,1)                                    PLYT0710
      IOR=1                                                          PLYT0720
      IF(PLT.NE.2.) GO TO 100                                        PLYT0730
C                                                                    PLYT0740
C   HORIZONTAL PLOT  PUT Y-AXIS ANNOTATION                           PLYT0750
C                                                                    PLYT0760
      IX=IX-125                                                      PLYT0770
      IY=200                                                         PLYT0780
      CALL DDTEXT (LENHA(3),HNOTES(1,3))                             PLYT0790
      GO TO 200                                                      PLYT0800
```

```
C                                                                       PLYT0810
100   IX=IX-125                                                         PLYT0820
      IY=IY-75                                                          PLYT0830
      NOTEA=1                                                           PLYT0840
      IF(ALTLOW.GE.0.0) NOTEA=3                                         PLYT0850
      IF(ABS(ALTHI-ALTLOW).GE.1.) NOTEA=NOTEA+1                         PLYT0860
      CALL DDTEXT (LENA(NOTEA),ANOTES(1,NOTEA))                         PLYT0870
C                                                                       PLYT0880
200   IOR=0                                                             PLYT0890
1840  FORMAT(F7.2,' DEG E.')                                            PLYT0900
1850  FORMAT(F7.2,' DEG N.')                                            PLYT0910
C                                                                       PLYT0920
      END                                                               PLYT0930



      SUBROUTINE DRAWTKS(DTICH,DTICV,XLPR,XR,YBPR,YT,PLOT)              DRYS0020
C                                                                       DRYS0030
C     HORIZONTAL PLOT PROJECTION SUPPORT ROUTINE.                       DRYS0040
C     DRAW BOUNDARY TO PLOT AREA, TICS AND TIC LABELS.                  DRYS0050
C                                                                       DRYS0060
      EXTERNAL PLOT                                                     DRYS0070
C                                                                       DRYS0080
      XL=XLPR                                                           DRYS0090
      YB=YBPR                                                           DRYS0100
      YMID=.5*(YB+YT)                                                   DRYS0110
      TICX=.01*(YT-YB)                                                  DRYS0120
      TICY=.01*(XR-XL)                                                  DRYS0130
      YBP=YB                                                            DRYS0140
      IF(DTICV.GT.0.) YBP=YMID-AINT((YMID-YB)/DTICV)*DTICV              DRYS0150
C                                                                       DRYS0160
      NTICX=(XR-XL)/DTICH                                               DRYS0170
      NTICY=1                                                           DRYS0180
      IF(DTICV.GT.0.) NTICY=(YT-YBP+DTICV)/DTICV                        DRYS0190
C&     PRINT *,"DRAWTKS ",XLPR,XR,YBPR,YBP,YT,DTICH,DTICV,NTICX,NTICY   DRYS0200
      IOF=0                                                             DRYS0210
C                                                                       DRYS0220
C     TWO PASSES BOTTOM AND LEFT THEN TOP AND RIGHT                     DRYS0230
      DO 100 J=1,2                                                      DRYS0240
      CALL PLOT(XL,YBPR,1)                                              DRYS0250
      DO 30 I=1,NTICY                                                   DRYS0260
      Y=YBP+(I-1)*DTICV                                                 DRYS0270
      CALL PLOT(XL,Y,0)                                                 DRYS0280
      CALL PLOT(XL+TICY,Y,0)                                            DRYS0290
30    CALL PLTANH(XL,Y,IOF+2,PLOT)                                      DRYS0300
      CALL PLOT(XL,YT,0)                                                DRYS0310
C                                                                       DRYS0320
      CALL PLOT(XLPR,YB,1)                                              DRYS0330
C                                                                       DRYS0340
      DO 40 I=1,NTICX                                                   DRYS0350
      X=XLPR+I*DTICH                                                    DRYS0360
      CALL PLOT(X,YB,0)                                                 DRYS0370
      CALL PLOT(X,YB+TICX,0)                                            DRYS0380
```

```
      CALL PLTANH(X,YB,IOF+1,PLOT)                                      DRYS0390
40    CALL PLOT(X,YB,1)                                                 DRYS0400
      CALL PLOT(XR,YB,0)                                                DRYS0410
      YB=YT                                                             DRYS0420
      XL=XR                                                             DRYS0430
      TICX=-TICX                                                        DRYS0440
      TICY=-TICY                                                        DRYS0450
100   IOF=10                                                            DRYS0460
C                                                                       DRYS0470
      CALL PLOT(XLPR,YMID,1)                                            DRYS0480
      CALL PLOT(XR,YMID,0)                                              DRYS0490
      END                                                               DRYS0500


      SUBROUTINE PLTLB(X,PY,NCP)                                        PLGB0020
C     PUT VERTICAL TIC ANNOTATIONS ON RAY PLOT                          PLGB0030
C     THIS IS A SPECIAL PURPOSE SUBSTITUTE FOR THE 'PLOT' ROUTINE       PLGB0040
C     IT COMPUTES THE NEAREST ROUNDED TIC POSITIONS FOR VERTICAL LINES. PLGB0050
C     WHEN THE NC PARAMETER IS > 0 AN ANNOTATION IS GENERATED           PLGB0060
C     AT THE TIC POSITION. THE ADDITIONAL NC VALUE < 0 ALLOWS FOR       PLGB0070
C     SIMPLE PEN UP MOTION TO THE X,Y POSITION.                         PLGB0080
C                                                                       PLGB0090
      CHARACTER ANNOT*10                                                PLGB0100
C                                                                       PLGB0110
C     COMMON DECK "WWR" INSERTED HERE                                   CWWR0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      COMMON /DD/ IINT,IOR,IT,IS,IC,ICC,IX,IY                           PLGB0130
      COMMON/LABCLT/PROJCT,THMIN,THMAX,RMIN,RMAX                        PLGB0140
C                                                                       PLGB0150
      DATA LIY/-100/                                                    PLGB0160
C                                                                       PLGB0170
      NC=NCP                                                            PLGB0180
      GO TO 10                                                          PLGB0190
C                                                                       PLGB0200
      ENTRY PLTLBN(X,PY,NCP)                                            PLGB0210
      NC=0                                                              PLGB0220
```

```
C                                                                       PLGB0230
10    Y=PY                                                              PLGB0240
C                                                                       PLGB0250
      IF(NCP.LT.0) GO TO 200                                            PLGB0260
      F=1.                                                              PLGB0270
      IF(AMAX1(ABS(RMAX-EARTHR),ABS(RMIN-EARTHR)).LT.1.) F=1000.        PLGB0280
      Y=INT((Y-EARTHR)*F)/F+EARTHR                                      PLGB0290
      IF(NC.EQ.0) GO TO 100                                             PLGB0300
C                                                                       PLGB0310
      V=Y                                                               PLGB0320
      IF(PROJCT.EQ.3.) V=RMIN+(Y-RMIN)/PFACTR                           PLGB0330
      IF(ABS(AINT(TICV*F)-TICV*F).GT..001) GO TO 60                     PLGB0340
C                                                                       PLGB0350
      WRITE(ANNOT,50) INT(ABS(V-EARTHR)*F)                              PLGB0360
50    FORMAT(I3)                                                        PLGB0370
      GO TO 90                                                          PLGB0380
C                                                                       PLGB0390
60    WRITE(ANNOT,80) ABS(V-EARTHR)*F                                   PLGB0400
80    FORMAT(F6.2)                                                      PLGB0410
C                                                                       PLGB0420
90    CALL PLOT(X,PY,1)                                                 PLGB0430
      CALL PLOT(X,Y,1)                                                  PLGB0440
C     INSURE THAT OVERLAPS OF ANNOTATIONS DO NOT OCCUR                  PLGB0450
      IF(IABS(IY-LIY).LT.80) GO TO 100                                  PLGB0460
      LIY=IY                                                            PLGB0470
      IX=IX-100                                                         PLGB0480
      IOR=0                                                             PLGB0490
      CALL DDTEXT(1,ANNOT)                                              PLGB0500
C                                                                       PLGB0510
100   CALL PLOT(X,Y,NC)                                                 PLGB0520
      RETURN                                                            PLGB0530
C                                                                       PLGB0540
200   CALL PLOT(X,Y,1)                                                  PLGB0550
      RETURN                                                            PLGB0560
      END                                                               PLGB0570
```

```
      SUBROUTINE ARCTIC(THMIN,THMAX,HEIGHT,TICY,PLOT)                   ARPC0020
C     DRAW RANGE AXIS IN RAY TRACE PLOT. INCLUDES ANY CURVILINEAR       ARPC0030
C     PROJECTIONS PROVIDED IN DDGRAPH.                                  ARPC0040
C     COMMON DECK "WWR" INSERTED HERE                                   CWWR0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
```

```
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                          CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),            CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))             CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),             CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                   CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                        CWW20160
      COMMON /DDSCALE/ XMIN,XMAX,YMIN,YMAX,MINX,MAXX,MINY,MAXY,SCX,SCY,      ARPC0060
     1 NSCX,NXCY,MSCX,MSCY,ISCX,ISCY                                         ARPC0070
                                                                             ARPC0080
C                                                                            ARPC0090
      NTIC=2                                                                 ARPC0100
      IF(TIC.NE.0.) NTIC=1+(THMAX-THMIN)/TIC                                 ARPC0110
      NLINE=MAX0(2,100/NTIC)                                                 ARPC0120
C                                                                            ARPC0130
      TICN=TIC/(NLINE-2)                                                     ARPC0140
      DO 10 I=1,NTIC                                                         ARPC0150
      X=THMIN+(I-1)*TIC                                                      ARPC0160
      CALL PLOT(X,HEIGHT+TICY,1)                                             ARPC0170
      DO 10 J=2,NLINE                                                        ARPC0180
      XJ=X+(J-2)*TICN                                                        ARPC0190
      IF(XJ.GT.THMAX) GO TO 15                                               ARPC0200
10    CALL PLOT(XJ,HEIGHT,0)                                                 ARPC0210
C                                                                            ARPC0220
15    CALL PLOT(THMAX,HEIGHT,0)                                              ARPC0230
      CALL PLOT(THMAX,HEIGHT+TICY,1)                                         ARPC0240
      RETURN                                                                 ARPC0250
      END
```

```
      SUBROUTINE DDINIT(N,TEXT)                                              DDRT0020
C     INITIALIZES PLOTTING PROCESS(DDPLOT)                                   DDRT0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                                DDRT0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                     CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                                   CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN                    CRAY0050
      CHARACTER TEXT*(*)                                                     DDRT0060
C                                                                            DDRT0070
      DATA IDD/0/                                                            DDRT0080
10    FORMAT(5A10)                                                           DDRT0090
      IF(IDD.EQ.0) REWIND NDEVGRP                                            DDRT0100
      IDD=1                                                                  DDRT0110
      WRITE(NDEVGRP) 0,0,0                                                   DDRT0120
      WRITE(NDEVGRP) M,LEN(TEXT),TEXT                                        DDRT0130
      END                                                                    DDRT0140
```

```
      SUBROUTINE DDBP                                                        DDBP0020
C     SETS A VECTOR ORIGIN(DDPLOT)                                           DDBP0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                                DDBP0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                     CRAY0020
```

```
C       DEVICE ASSIGNED TO RAYTRC INPUT FILE                                    CRAY0040
        COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN                     CRAY0050
        COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                                     DDBP0060
        INTEGER REZFLG                                                          DDHI0020
        PARAMETER (REZFLG=2001)                                                 DDHI0030
        COMMON/DDREZ/DDHIX,DDHIY                                                DDHI0040
        COMMON/KNKN/KNBP,KNVC,KNDT                                              DDBP0080
        COMMON/DDLIM/MXIX,MXIY,MNIX,MNIY                                        DDBP0090
C                                                                               DDBP0100
C       USE PRE-ASSIGNED FLAG VALUE FOR IX TO DETERMINE USE OF                  DDBP0110
C       FLOATING POINT RASTER VALUES OF X AND Y USED IN DDBP AND DDVC           DDBP0120
C                                                                               DDBP0130
        IF(IX.EQ.REZFLG) THEN                                                   DDBP0140
           IX=DDHIX                                                             DDBP0150
           IY=DDHIY                                                             DDBP0160
        ELSE                                                                    DDBP0170
           DDHIX=IX                                                             DDBP0180
           DDHIY=IY                                                             DDBP0190
        ENDIF                                                                   DDBP0200
C                                                                               DDBP0210
        MNIX=MIN0(MNIX,IX)                                                      DDBP0220
        MXIX=MAX0(MXIX,IX)                                                      DDBP0230
        MNIY=MIN0(MNIY,IY)                                                      DDBP0240
        MXIY=MAX0(MXIY,IY)                                                      DDBP0250
        IF(IX.LT.0.OR.IX.GT.1023.OR.IY.LT.0.OR.IY.GT.1023                       DDBP0260
       1  ) PRINT 10,'DDBP ',KNBP,IX,IY                                        DDBP0270
10      FORMAT(A10,3I5)                                                         DDBP0280
        KNBP=KNBP+1                                                             DDBP0290
        WRITE(NDEVGRP) 1,DDHIX,DDHIY                                            DDBP0300
        END                                                                     DDBP0310
```

```
        SUBROUTINE DDVC                                                         DDVC0020
C       PLOTS A VECTOR(DDPLOT)                                                  DDVC0030
C       WRITE PARAMETERS TO GRAPHICS CALLS FILE                                 DDVC0040
C       COMMON DECK "RAYDEV" INSERTED HERE                                      CRAY0020
C       DEVICE ASSIGNED TO RAYTRC INPUT FILE                                    CRAY0040
        COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN                     CRAY0050
        COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                                     DDVC0060
        INTEGER REZFLG                                                          DDHI0020
        PARAMETER (REZFLG=2001)                                                 DDHI0030
        COMMON/DDREZ/DDHIX,DDHIY                                                DDHI0040
        COMMON/KNKN/KNBP,KNVC,KNDT                                              DDVC0080
        COMMON/DDLIM/MXIX,MXIY,MNIX,MNIY                                        DDVC0090
C                                                                               DDVC0100
C       USE PRE-ASSIGNED FLAG VALUE FOR IX TO DETERMINE USE OF                  DDVC0110
C       FLOATING POINT RASTER VALUES OF X AND Y USED IN DDBP AND DDVC           DDVC0120
C                                                                               DDVC0130
        IF(IX.EQ.REZFLG) THEN                                                   DDVC0140
           IX=DDHIX                                                             DDVC0150
           IY=DDHIY                                                             DDVC0160
        ELSE                                                                    DDVC0170
```

```
         DDHIX=IX                                                        DDVC0180
         DDHIY=IY                                                        DDVC0190
      ENDIF                                                              DDVC0200
C                                                                        DDVC0210
      MNIX=MIN0(MNIX,IX)                                                 DDVC0220
      MXIX=MAX0(MXIX,IX)                                                 DDVC0230
      MNIY=MIN0(MNIY,IY)                                                 DDVC0240
      MXIY=MAX0(MXIY,IY)                                                 DDVC0250
      IF(IX.LT.0.OR.IX.GT.1023.OR.IY.LT.0.OR.IY.GT.1023                  DDVC0260
     1  ) PRINT 10,'DDVC ',KNVC,IX,IY                                    DDVC0270
10    FORMAT(A10,3I5)                                                    DDVC0280
      KNVC=KNVC+1                                                        DDVC0290
      WRITE(NDEVGRP) 2,DDHIX,DDHIY                                       DDVC0300
      END                                                                DDVC0310
```

```
      SUBROUTINE DDTEXT(N,TEXT)                                          DDXT0020
C     WRITES A CHARACTER ARRAY PACKED A10(DDPLOT)                        DDXT0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                            DDXT0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                 CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                               CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN                CRAY0050
      CHARACTER TEXT*(*)                                                 DDXT0060
      COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                                DDXT0070
      COMMON/KNKN/KNBP,KNVC,KNDT                                         DDXT0080
100   FORMAT(A10,2I5,I2,1X,8A10)                                         DDXT0090
      KNDT=KNDT+1                                                        DDXT0100
      WRITE(NDEVGRP) 3,IX,IY                                             DDXT0110
      LN=MIN0(N*10,LEN(TEXT))                                            DDXT0120
      IF(LN.EQ.N*10) THEN                                                DDXT0130
         WRITE(NDEVGRP) IOR,N,N*10,(TEXT(I:I),I=1,LN)                    DDXT0140
      ELSE                                                               DDXT0150
         WRITE(NDEVGRP) IOR,N,N*10,(TEXT(I:I),I=1,LN)                    DDXT0160
     1   ,(' ',I=LN+1,N*10)                                              DDXT0170
      ENDIF                                                              DDXT0180
      END                                                                DDXT0190
```

```
      SUBROUTINE DDTAB                                                   DDQB0020
C     INITIALIZES TABULAR PLOTTING(DDPLOT)                               DDQB0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                            DDQB0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                 CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                               CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN                CRAY0050
      END                                                                DDQB0060
```

```
      SUBROUTINE DDFR                                                   DDFR0020
C     ADVANCE ONE PLOTTING FRAME(DDPLOT)                                DDFR0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                           DDFR0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                              CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN               CRAY0050
10    FORMAT(A10)                                                       DDFR0060
      WRITE(NDEVGRP) -2,0,0                                             DDFR0070
      END                                                               DDFR0080
```

```
      SUBROUTINE DDEND                                                  DDPD0020
C     EMPTIES PLOT BUFFER AND RELEASES PLOTTING COMMAND FILE(DDPLOT)    DDPD0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                           DDPD0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                              CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN               CRAY0050
      COMMON/KNKN/KNBP,KNVC,KNDT                                        DDPD0060
      COMMON/DDLIM/MXIX,MXIY,MNIX,MNIY                                  DDPD0070
10    FORMAT(3A10,5I5)                                                  DDPD0080
      WRITE(NDEVGRP) -1,0,0                                             DDPD0090
      END                                                               DDPD0100
```

```
      SUBROUTINE DASH                                                   DASH0020
C     ACTIVATE DASHED LINE CONNECTIONS(DISSPLA)                         DASH0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                           DASH0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                              CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN               CRAY0050
      WRITE(NDEVGRP) 37,0,0                                             DASH0060
      END                                                               DASH0070
```

```
      SUBROUTINE RESET(S)                                               REUT0020
C     RESETS AN OPTION TO ITS DEFAULT VALUE(DISSPLA)                    REUT0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                           REUT0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                              CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN               CRAY0050
      CHARACTER S*10                                                    REUT0060
      WRITE(NDEVGRP) 38,0,0                                             REUT0070
      WRITE(NDEVGRP) S                                                  REUT0080
      END                                                               REUT0090
```

```
      SUBROUTINE HEIGHT(H)                                              HEPT0020
C     SETS REFERENCE CHARACTER HEIGHT IN INCHES(DISSPLA)                 HEPT0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                           HEPT0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                              CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN               CRAY0050
      WRITE(NDEVGRP) 13,H,0                                             HEPT0060
      END                                                               HEPT0070


      SUBROUTINE MX1ALF(T1,T2)                                          MXIF0020
C     SPECIFY USE OF ALTERNATE CHARACTER SET NUMBER 1(DISSPLA)          MXIF0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                           MXIF0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                              CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN               CRAY0050
      WRITE(NDEVGRP) 11,T1,T2                                           MXIF0060
      END                                                               MXIF0070


      SUBROUTINE MX2ALF(T1,T2)                                          MXJF0020
C     SPECIFY USE OF ALTERNATE CHARACTER SET NUMBER 2(DISSPLA)          MXJF0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                           MXJF0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                              CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN               CRAY0050
      WRITE(NDEVGRP) 12,T1,T2                                           MXJF0060
      END                                                               MXJF0070


      SUBROUTINE SCMPLX                                                 SCJX0020
C     SPECIFY USE OF SIMPLEX CHARACTER SET(DISSPLA)                     SCJX0030
C     WRITE PARAMETERS TO GRAPHICS CALLS FILE                           SCJX0040
C     COMMON DECK "RAYDEV" INSERTED HERE                                CRAY0020
C     DEVICE ASSIGNED TO RAYTRC INPUT FILE                              CRAY0040
      COMMON/RAYDEV/NRYIND,NDEVTMP,NFRMAT,NDEVGRP,NDEVBIN               CRAY0050
      WRITE(NDEVGRP) 10,0,0                                             SCJX0060
      END                                                               SCJX0070
```

D.2 DISPERSION-RELATION ROUTINES (Tape File 4)

```
      SUBROUTINE ANCNL                                                  ANNL0020
C     DISPERSION RELATION FOR ACOUSTIC WAVES NO CURRENT, NO LOSSES      ANNL0030
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "CC" INSERTED HERE                                    CCC 0020
      REAL MODC                                                         CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                        CCC 0050
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "RK" INSERTED HERE                                    CRK 0020
C     DEFINE SIZE REQUIRED FOR RAY STATE SAVE ARRAY                     CRK 0040
      PARAMETER (LRKAMS=87+2*100,NXRKMS=12+LRKAMS,MXEQPT=21)            CRK 0050
      PARAMETER (NRKSAV=NXRKMS+MXEQPT-1)                                CRK 0060
      COMMON /RK/ NEQS,STEP,MODE,E1MAX,E1MIN,E2MAX,E2MIN,FACT,RSTART    CRK 0070
C     COMMON DECK "RINREAL" INSERTED HERE                               CRIN0020
      LOGICAL SPACE                                                     CRIN0040
      REAL LPOLAR,LPOLRI,KPHK,KPHKI,KAY2,KAY2I                          CRIN0050
      CHARACTER DISPM*6                                                 CRIN0060
      COMMON/RINPL/DISPM                                                CRIN0070
      COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                 CRIN0080
      COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I,                            CRIN0090
     1             H,HI,PHT,PHTI,PHR,PHRI,PHTH,PHTHI,PHPH,PHPHI         CRIN0100
     2, PHOW,PHOWI,PHKR,PHKRI,PHKTH,PHKTI, PHKPH,PHKPI                  CRIN0110
     3 ,KPHK,KPHKI,POLAR,POLARI,LPOLAR,LPOLRI,SGN                       CRIN0120
C     COMMON DECK "UU" INSERTED HERE                                    CUU 0020
      REAL MODU                                                         CUU 0040
      COMMON/UU/MODU(4)                                                 CUU 0050
     1  ,V  ,PVT  ,PVR  ,PVTH  ,PVPH                                    CUU 0060
     2  ,VR ,PVRT ,PVRR ,PVRTH ,PVRPH                                   CUU 0070
     3  ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                  CUU 0080
     4  ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                  CUU 0090
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
```

```
C                                                                       CWW30060
C     DELTA WIND          125-149                                       CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
                                                                        CWW30090
C                                                                       CWW30100
C     SOUND SPEED 150-174                                               CWW30110
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)           CWW30120
      EQUIVALENCE (W(153),REFC)                                         CWW30130
C                                                                       CWW30140
C     DELTA SOUND SPEED 175-199                                         CWW30150
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)        CWW30160
C                                                                       CWW30170
C     TEMPERATURE    200-224                                            CWW30180
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)           CWW30190
C                                                                       CWW30200
C     DELTA TEMPERATURE    225-249                                      CWW30210
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)        CWW30220
C                                                                       CWW30230
C     MOLECULAR   250-274                                               CWW30240
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)           CWW30250
C                                                                       CWW30260
C     RECEIVER HEIGHT   275-299                                         CWW30270
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)           CWW30280
C                                                                       CWW30290
C     TOPOGRAPHY    300-324                                             CWW30300
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)           CWW30310
C                                                                       CWW30320
C     DELTA TOPOGRAPHY    325-349                                       CWW30330
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)        CWW30340
C                                                                       CWW30350
C     ATMOSPHERIC SURFACE TOPOGRAPHY    350-374                       CWW30360
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)           CWW30370
C                                                                       CWW30380
C     ATMOSPHERIC SURFACE TOPOGRAPHY    375-399                       CWW30390
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)        CWW30400
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                              CWW30410
C                                                                       CWW30420
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                       CWW30430
C     ABSORPTION          500-524                                       CWW30440
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)           CWW30450
C                                                                       CWW30460
C     DELTA ABSORPTION          525-549                                 CWW30470
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)        CWW30480
C                                                                       CWW30490
C     PRESSURE          550-574                                         CWW30500
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)           CWW30510
C                                                                       CWW30520
C     DELTA PRESSURE          575-599                                   CWW30530
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)        CWW30540
C                                                                       CWW30550
C                                                                       ANNL0110
      REAL KS,MKS                                                       ANNL0120
      REAL KVECT(24)                                                    ANNL0130
      REAL VSET(20)                                                     ANNL0140
      EQUIVALENCE(KVECT,KAY2),(VSET,V)                                  ANNL0150
C                                                                       ANNL0160
```

```
      REAL MRDRIN(8)     ,TBPE( 3)                                          ANNL0170
      CHARACTER DLSPM*5                                                     ANNL0180
C                                                                           ANNL0190
      DATA MRDRIN/8HACOUSTIC ,8H WAVE ** ,8H** NO CU ,8HRRENT **            ANNL0200
     1  ,8H* NO LOS ,3HSES ,2*1H /                                          ANNL0210
      DATA DLSPM/'ANCNL'/                                                   ANNL0220
      DATA TBPE/3*1H0 /                                                     ANNL0230
C                                                                           ANNL0240
C                                                                           ANNL0250
      ENTRY SETDSP                                                          ANNL0260
C                                                                           ANNL0270
      CALL RMOVE(MODRIN,MRDRIN,8)                                           ANNL0280
      DISPM=DLSPM                                                           ANNL0290
      CALL RMOVE(TYPE,TBPE, 3)                                              ANNL0300
C                                                                           ANNL0310
      CALL SETSPD                                                           ANNL0320
      CALL SETRCV                                                           ANNL0330
      CALL SETTOP                                                           ANNL0340
      CALL SETSUR                                                           ANNL0350
      RETURN                                                                ANNL0360
C                                                                           ANNL0370
      ENTRY IDISPER                                                         ANNL0380
      CALL ISPEED                                                           ANNL0390
      CALL IRECVR                                                           ANNL0400
      CALL ITOPOG                                                           ANNL0410
      CALL ISURFAC                                                          ANNL0420
      RETURN                                                                ANNL0430
C                                                                           ANNL0440
      ENTRY DISPER                                                          ANNL0450
      ENTRY RINDEX                                                          ANNL0460
C                                                                           ANNL0470
      SPACE=.FALSE.                                                         ANNL0480
C                                                                           ANNL0490
      KS=KR*KR+KTH*KTH+KPH*KPH                                              ANNL0500
C     SOUND SPEED                                                           ANNL0510
      CALL SPEED                                                            ANNL0520
      OWS=OW*OW                                                             ANNL0530
      KAY2=OWS/CS                                                           ANNL0540
C                                                                           ANNL0550
      H=OW*OW - CS*KS                                                       ANNL0560
      MKS=-KS                                                               ANNL0570
      PHT=MKS*PCST                                                          ANNL0580
      PHR=MKS*PCSR                                                          ANNL0590
      PHTH=MKS*PCSTH                                                        ANNL0600
      PHPH=MKS*PCSPH                                                        ANNL0610
C                                                                           ANNL0620
      PHOW=2.0*OW                                                           ANNL0630
      CS2=-2.0*CS                                                           ANNL0640
      PHKR=CS2*KR                                                           ANNL0650
      PHKTH=CS2*KTH                                                         ANNL0660
      PHKPH=CS2*KPH                                                         ANNL0670
      KPHK=CS2*KS                                                           ANNL0680
      RETURN                                                                ANNL0690
      END                                                                   ANNL0700
```

```
      SUBROUTINE AWCNL                                                  AWNL0020
C     DISPERSION RELATION FOR ACOUSTIC WAVES WITH CURRENT, NO LOSSES    AWNL0030
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "CC" INSERTED HERE                                    CCC 0020
      REAL MODC                                                         CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                        CCC 0050
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "RK" INSERTED HERE                                    CRK 0020
C     DEFINE SIZE REQUIRED FOR RAY STATE SAVE ARRAY                     CRK 0040
      PARAMETER (LRKAMS=87+2*100,NXRKMS=12+LRKAMS,MXEQPT=21)            CRK 0050
      PARAMETER (NRKSAV=NXRKMS+MXEQPT-1)                                CRK 0060
      COMMON /RK/ NEQS,STEP,MODE,E1MAX,E1MIN,E2MAX,E2MIN,FACT,RSTART    CRK 0070
C     COMMON DECK "RINREAL" INSERTED HERE                               CRIN0020
      LOGICAL SPACE                                                     CRIN0040
      REAL LPOLAR,LPOLRI,KPHK,KPHKI,KAY2,KAY2I                          CRIN0050
      CHARACTER DISPM*6                                                 CRIN0060
      COMMON/RINPL/DISPM                                                CRIN0070
      COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                 CRIN0080
      COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I,                            CRIN0090
     1           H,HI,PHT,PHTI,PHR,PHRI,PHTH,PHTHI,PHPH,PHPHI           CRIN0100
     2, PHOW,PHOWI,PHKR,PHKRI,PHKTH,PHKTI, PHKPH,PHKPI                  CRIN0110
     3 ,KPHK,KPHKI,POLAR,POLARI,LPOLAR,LPOLRI,SGN                       CRIN0120
C     COMMON DECK "UU" INSERTED HERE                                    CUU 0020
      REAL MODU                                                         CUU 0040
      COMMON/UU/MODU(4)                                                 CUU 0050
     1  ,V  ,PVT  ,PVR  ,PVTH  ,PVPH                                    CUU 0060
     2  ,VR ,PVRT ,PVRR ,PVRTH ,PVRPH                                   CUU 0070
     3  ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                  CUU 0080
     4  ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                  CUU 0090
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
```

```
      REAL MMODEL,MFORM,MID                                                  CWW30020
C                                                                            CWW30030
C     WIND         100-124                                                   CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)                CWW30050
C                                                                            CWW30060
C     DELTA WIND         125-149                                           CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)             CWW30080
C                                                                            CWW30090
C     SOUND SPEED 150-174                                                    CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)                CWW30110
      EQUIVALENCE (W(153),REFC)                                              CWW30120
C                                                                            CWW30130
C     DELTA SOUND SPEED 175-199                                              CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)             CWW30150
C                                                                            CWW30160
C     TEMPERATURE    200-224                                                 CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)                CWW30180
C                                                                            CWW30190
C     DELTA TEMPERATURE    225-249                                           CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)             CWW30210
C                                                                            CWW30220
C     MOLECULAR   250-274                                                    CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)                CWW30240
C                                                                            CWW30250
C     RECEIVER HEIGHT   275-299                                              CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)                CWW30270
C                                                                            CWW30280
C     TOPOGRAPHY   300-324                                                   CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)                CWW30300
C                                                                            CWW30310
C     DELTA TOPOGRAPHY   325-349                                             CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)             CWW30330
C                                                                            CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                               CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)                CWW30360
C                                                                            CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                               CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)             CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                   CWW30400
C                                                                            CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                            CWW30420
C     ABSORPTION         500-524                                             CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)                CWW30440
C                                                                            CWW30450
C     DELTA ABSORPTION         525-549                                       CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)             CWW30470
C                                                                            CWW30480
C     PRESSURE         550-574                                               CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)                CWW30500
C                                                                            CWW30510
C     DELTA PRESSURE         575-599                                         CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)             CWW30530
C                                                                            CWW30540
C                                                                            AWNL0110
      REAL KS,KV                                                             AWNL0120
```

```
      REAL KVECT(24)                                                    AWNL0130
      EQUIVALENCE(KVECT,KAY2)                                           AWNL0140
C                                                                       AWNL0150
      REAL MRDRIN(8)     ,TBPE( 3)                                      AWNL0160
      CHARACTER DLSPM*5                                                 AWNL0170
C                                                                       AWNL0180
      DATA MRDRIN/8HACOUSTIC ,8H WAVE ** ,8H* WITH C ,8HURRENT *        AWNL0190
     1  ,8H*** NO L ,5HOSSES ,2*1H /                                    AWNL0200
      DATA DLSPM/'AWCNL'/                                               AWNL0210
      DATA TBPE/3*1H1 /                                                 AWNL0220
C                                                                       AWNL0230
C                                                                       AWNL0240
      ENTRY SETDSP                                                      AWNL0250
C                                                                       AWNL0260
      CALL RMOVE(MODRIN,MRDRIN,8)                                       AWNL0270
      DISPM=DLSPM                                                       AWNL0280
      CALL RMOVE(TYPE,TBPE, 3)                                          AWNL0290
C                                                                       AWNL0300
      CALL SETSPD                                                       AWNL0310
      CALL SETWND                                                       AWNL0320
      CALL SETRCV                                                       AWNL0330
      CALL SETTOP                                                       AWNL0340
      CALL SETSUR                                                       AWNL0350
      RETURN                                                            AWNL0360
C                                                                       AWNL0370
      ENTRY IDISPER                                                     AWNL0380
      CALL ISPEED                                                       AWNL0390
      CALL IWINDR                                                       AWNL0400
      CALL IRECVR                                                       AWNL0410
      CALL ITOPOG                                                       AWNL0420
      CALL ISURFAC                                                      AWNL0430
      RETURN                                                            AWNL0440
C                                                                       AWNL0450
      ENTRY DISPER                                                      AWNL0460
C                                                                       AWNL0470
      SPACE=.FALSE.                                                     AWNL0480
C                                                                       AWNL0490
      KS=KR*KR+KTH*KTH+KPH*KPH                                          AWNL0500
C     WIND VELOCITY                                                     AWNL0510
      CALL WINDR                                                        AWNL0520
      KV=KR*VR+KTH*VTH+KPH*VPH                                          AWNL0530
      VLS=KV*KV/KS                                                      AWNL0540
C     SOUND SPEED                                                       AWNL0550
      CALL SPEED                                                        AWNL0560
      OWS=OW*OW                                                         AWNL0570
      KAY2=OWS/(SQRT(CS)+KV/SQRT(KS))**2                                AWNL0580
C                                                                       AWNL0590
OWI=OW-KV                                                               AWNL0600
H=OWI*OWI - CS*KS                                                       AWNL0610
POWIT=-KR*PVRT - KTH*PVTHT - KPH*PVPHT                                  AWNL0620
PHT=2.0*OWI*POWIT - KS*PCST                                             AWNL0630
POWIR=-KR*PVRR - KTH*PVTHR - KPH*PVPHR                                  AWNL0640
PHR=2.0*OWI*POWIR - KS*PCSR                                             AWNL0650
POWITH=-KR*PVRTH - KTH*PVTHTH - KPH*PVPHTH                              AWNL0660
PHTH=2.0*OWI*POWITH - KS*PCSTH                                          AWNL0670
```

```
      POWIPH=-KR*PVRPH - KTH*PVTHPH - KPH*PVPHPH                        AWNL0680
      PHPH=2.0*OWI*POWIPH - KS*PCSPH                                    AWNL0690
C                                                                       AWNL0700
      PHOW=2.0*OWI                                                      AWNL0710
      PHKR=-2.0*(OWI*VR + CS*KR)                                        AWNL0720
      PHKTH=-2.0*(OWI*VTH + CS*KTH)                                     AWNL0730
      PHKPH=-2.0*(OWI*VPH + CS*KPH)                                     AWNL0740
      KPHK=-2.0*(OWI*KV + CS*KS)                                        AWNL0750
      RETURN                                                            AWNL0760
      END                                                               AWNL0770
```

```
      SUBROUTINE ANCWL                                                  ANRL0020
C     DISPERSION RELATION FOR ACOUSTIC WAVES NO CURRENT, WITH LOSSES    ANRL0030
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "CC" INSERTED HERE                                    CCC 0020
      REAL MODC                                                         CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                        CCC 0050
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "RK" INSERTED HERE                                    CRK 0020
C     DEFINE SIZE REQUIRED FOR RAY STATE SAVE ARRAY                     CRK 0040
      PARAMETER (LRKAMS=87+2*100,NXRKMS=12+LRKAMS,MXEQPT=21)            CRK 0050
      PARAMETER (NRKSAV=NXRKMS+MXEQPT-1)                                CRK 0060
      COMMON /RK/ NEQS,STEP,MODE,E1MAX,E1MIN,E2MAX,E2MIN,FACT,RSTART    CRK 0070
C     COMMON DECK "RINREAL" INSERTED HERE                               CRIN0020
      LOGICAL SPACE                                                     CRIN0040
      REAL LPOLAR,LPOLRI,KPHK,KPHKI,KAY2,KAY2I                          CRIN0050
      CHARACTER DISPM*6                                                 CRIN0060
      COMMON/RINPL/DISPM                                                CRIN0070
      COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                 CRIN0080
      COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I,                            CRIN0090
     1              H,HI,PHT,PHTI,PHR,PHRI,PHTH,PHTHI,PHPH,PHPHI       CRIN0100
     2, PHOW,PHOWI,PHKR,PHKRI,PHKTH,PHKTI, PHKPH,PHKPI                  CRIN0110
     3 ,KPHK,KPHKI,POLAR,POLARI,LPOLAR,LPOLRI,SGN                       CRIN0120
C     COMMON DECK "UU" INSERTED HERE                                    CUU 0020
      REAL MODU                                                         CUU 0040
      COMMON/UU/MODU(4)                                                 CUU 0050
     1  ,V  ,PVT  ,PVR  ,PVTH  ,PVPH                                    CUU 0060
     2  ,VR ,PVRT ,PVRR ,PVRTH ,PVRPH                                   CUU 0070
     3  ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                  CUU 0080
     4  ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                  CUU 0090
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
```

```
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),          CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                  CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND         125-149                                        CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
C                                                                       CWW30090
C     SOUND SPEED 150-174                                               CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)           CWW30110
      EQUIVALENCE (W(153),REFC)                                         CWW30120
C                                                                       CWW30130
C     DELTA SOUND SPEED 175-199                                         CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)        CWW30150
C                                                                       CWW30160
C     TEMPERATURE    200-224                                            CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)           CWW30180
C                                                                       CWW30190
C     DELTA TEMPERATURE    225-249                                      CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)        CWW30210
C                                                                       CWW30220
C     MOLECULAR   250-274                                               CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)           CWW30240
C                                                                       CWW30250
C     RECEIVER HEIGHT   275-299                                         CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)           CWW30270
C                                                                       CWW30280
C     TOPOGRAPHY   300-324                                              CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)           CWW30300
C                                                                       CWW30310
C     DELTA TOPOGRAPHY   325-349                                        CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)        CWW30330
C                                                                       CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                          CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)           CWW30360
C                                                                       CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                          CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)        CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                              CWW30400
C                                                                       CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                       CWW30420
C     ABSORPTION         500-524                                        CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)           CWW30440
C                                                                       CWW30450
```

```
C      DELTA ABSORPTION         525-549                                        CWW30460
       EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)             CWW30470
C                                                                             CWW30480
C      PRESSURE         550-574                                               CWW30490
       EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)                CWW30500
C                                                                             CWW30510
C      DELTA PRESSURE         575-599                                        CWW30520
       EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)             CWW30530
C                                                                             CWW30540
C                                                                             ANRL0110
C      COMMON DECK "LL" INSERTED HERE                                         CLL 0020
       REAL MODL                                                              CLL 0040
       COMMON/LL/MODL(4),APH,APHPT,APHPR,APHPTH,APHPPH                        CLL 0050
C                                                                             ANRL0130
C      COMMON DECK "PP" INSERTED HERE                                         CPP 0020
       REAL MODP                                                              CPP 0040
       COMMON/PP/MODP(4),P,PPT,PPR,PPTH,PPPH                                  CPP 0050
C                                                                             ANRL0150
C      COMMON DECK "MM" INSERTED HERE                                         CMM 0020
       REAL M,MODM                                                            CMM 0040
       COMMON/MM/MODM(4),M,PMT,PMR,PMTH,PMPH                                  CMM 0050
C                                                                             ANRL0170
       REAL KS,MKS                                                            ANRL0180
       REAL KVECT(24)                                                         ANRL0190
       REAL VSET(20)                                                          ANRL0200
       EQUIVALENCE(KVECT,KAY2),(VSET,V)                                       ANRL0210
C                                                                             ANRL0220
       REAL MRDRIN(8)     ,TBPE( 3)                                           ANRL0230
       CHARACTER DLSPM*5                                                      ANRL0240
C                                                                             ANRL0250
       DATA MRDRIN/8HACOUSTIC ,8H WAVE ** ,8H** NO CU ,8HRRENT **             ANRL0260
      1  ,8H* WITH L ,5HOSSES ,2*1H /                                         ANRL0270
       DATA DLSPM/'ANCWL'/                                                    ANRL0280
       DATA TBPE/3*1H2 /                                                      ANRL0290
C                                                                             ANRL0300
C                                                                             ANRL0310
       ENTRY SETDSP                                                           ANRL0320
C                                                                             ANRL0330
       CALL RMOVE(MODRIN,MRDRIN,8)                                            ANRL0340
       DISPM=DLSPM                                                            ANRL0350
       CALL RMOVE(TYPE,TBPE, 3)                                               ANRL0360
C                                                                             ANRL0370
       CALL SETSPD                                                            ANRL0380
       CALL SETRCV                                                            ANRL0390
       CALL SETTOP                                                            ANRL0400
       CALL SETSUR                                                            ANRL0410
       RETURN                                                                 ANRL0420
C                                                                             ANRL0430
       ENTRY IDISPER                                                          ANRL0440
       CALL ISPEED                                                            ANRL0450
       CALL IRECVR                                                            ANRL0460
       CALL ITOPOG                                                            ANRL0470
       CALL ISURFAC                                                           ANRL0480
C                                                                             ANRL0490
       CALL IABSRP                                                            ANRL0500
```

```
      RETURN                                                            ANRL0510
C                                                                       ANRL0520
      ENTRY DISPER                                                      ANRL0530
      ENTRY RINDEX                                                      ANRL0540
C                                                                       ANRL0550
      SPACE=.FALSE.                                                     ANRL0560
C                                                                       ANRL0570
      KS=KR*KR+KTH*KTH+KPH*KPH                                          ANRL0580
C     SOUND SPEED                                                       ANRL0590
      CALL SPEED                                                        ANRL0600
      OWS=OW*OW                                                         ANRL0610
      KAY2=OWS/CS                                                       ANRL0620
C                                                                       ANRL0630
      H=OW*OW - CS*KS                                                   ANRL0640
      MKS=-KS                                                           ANRL0650
      PHT=MKS*PCST                                                      ANRL0660
      PHR=MKS*PCSR                                                      ANRL0670
      PHTH=MKS*PCSTH                                                    ANRL0680
      PHPH=MKS*PCSPH                                                    ANRL0690
C                                                                       ANRL0700
      PHOW=2.0*OW                                                       ANRL0710
      CS2=-2.0*CS                                                       ANRL0720
      PHKR=CS2*KR                                                       ANRL0730
      PHKTH=CS2*KTH                                                     ANRL0740
      PHKPH=CS2*KPH                                                     ANRL0750
      KPHK=CS2*KS                                                       ANRL0760
C                                                                       ANRL0770
      CALL ABSRP                                                        ANRL0780
      GMS=GAMMA-1.0                                                     ANRL0790
      KAY2I=-SQRT(KAY2)*APH                                             ANRL0800
      RETURN                                                            ANRL0810
      END                                                               ANRL0820
```

```
      SUBROUTINE AWCWL                                                  AWRL0020
C     DISPERSION RELATION FOR ACOUSTIC WAVES WITH CURRENT, WITH LOSSES  AWRL0030
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "CC" INSERTED HERE                                    CCC 0020
      REAL MODC                                                         CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                        CCC 0050
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "RK" INSERTED HERE                                    CRK 0020
C     DEFINE SIZE REQUIRED FOR RAY STATE SAVE ARRAY                     CRK 0040
      PARAMETER (LRKAMS=87+2*100,NXRKMS=12+LRKAMS,MXEQPT=21)            CRK 0050
      PARAMETER (NRKSAV=NXRKMS+MXEQPT-1)                                CRK 0060
      COMMON /RK/ NEQS,STEP,MODE,E1MAX,E1MIN,E2MAX,E2MIN,FACT,RSTART    CRK 0070
C     COMMON DECK "RINREAL" INSERTED HERE                               CRIN0020
      LOGICAL SPACE                                                     CRIN0040
```

```
      REAL LPOLAR,LPOLRI,KPHK,KPHKI,KAY2,KAY2I                          CRIN0050
      CHARACTER DISPM*6                                                 CRIN0060
      COMMON/RINPL/DISPM                                                CRIN0070
      COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                 CRIN0080
      COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I,                            CRIN0090
     1             H,HI,PHT,PHTI,PHR,PHRI,PHTH,PHTHI,PHPH,PHPHI        CRIN0100
     2, PHOW,PHOWI,PHKR,PHKRI,PHKTH,PHKTI, PHKPH,PHKPI                  CRIN0110
     3 ,KPHK,KPHKI,POLAR,POLARI,LPOLAR,LPOLRI,SGN                       CRIN0120
C     COMMON DECK "UU" INSERTED HERE                                    CUU 0020
      REAL MODU                                                         CUU 0040
      COMMON/UU/MODU(4)                                                 CUU 0050
     1  ,V  ,PVT  ,PVR  ,PVTH  ,PVPH                                    CUU 0060
     2  ,VR ,PVRT ,PVRR ,PVRTH ,PVRPH                                   CUU 0070
     3  ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                  CUU 0080
     4  ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                  CUU 0090
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND         125-149                                        CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
C                                                                       CWW30090
C     SOUND SPEED 150-174                                               CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)           CWW30110
      EQUIVALENCE (W(153),REFC)                                         CWW30120
C                                                                       CWW30130
C     DELTA SOUND SPEED 175-199                                         CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)        CWW30150
C                                                                       CWW30160
C     TEMPERATURE    200-224                                            CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)           CWW30180
C                                                                       CWW30190
C     DELTA TEMPERATURE    225-249                                      CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)        CWW30210
C                                                                       CWW30220
C     MOLECULAR   250-274                                               CWW30230
```

```
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)                CWW30240
                                                                             CWW30250
C                                                                            CWW30260
C     RECEIVER HEIGHT   275-299                                              CWW30270
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)                CWW30280
C                                                                            CWW30290
C     TOPOGRAPHY   300-324                                                   CWW30300
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)                CWW30310
C                                                                            CWW30320
C     DELTA TOPOGRAPHY   325-349                                             CWW30330
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)             CWW30340
C                                                                            CWW30350
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                               CWW30360
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)                CWW30370
C                                                                            CWW30380
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                               CWW30390
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)             CWW30400
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                   CWW30410
C                                                                            CWW30420
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                            CWW30430
C     ABSORPTION         500-524                                             CWW30440
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)                CWW30450
C                                                                            CWW30460
C     DELTA ABSORPTION         525-549                                       CWW30470
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)             CWW30480
C                                                                            CWW30490
C     PRESSURE         550-574                                               CWW30500
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)                CWW30510
C                                                                            CWW30520
C     DELTA PRESSURE         575-599                                         CWW30530
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)             CWW30540
C                                                                            AWRL0110
C                                                                            CLL 0020
C     COMMON DECK "LL" INSERTED HERE                                         CLL 0040
      REAL MODL                                                              CLL 0050
      COMMON/LL/MODL(4),APH,APHPT,APHPR,APHPTH,APHPPH                        AWRL0130
C                                                                            CPP 0020
C     COMMON DECK "PP" INSERTED HERE                                         CPP 0040
      REAL MODP                                                              CPP 0050
      COMMON/PP/MODP(4),P,PPT,PPR,PPTH,PPPH                                  AWRL0150
C                                                                            CMM 0020
C     COMMON DECK "MM" INSERTED HERE                                         CMM 0040
      REAL M,MODM                                                            CMM 0050
      COMMON/MM/MODM(4),M,PMT,PMR,PMTH,PMPH                                  AWRL0170
C                                                                            AWRL0180
      REAL KS,KV                                                             AWRL0190
      REAL KVECT(24)                                                         AWRL0200
      EQUIVALENCE(KVECT,KAY2)                                                AWRL0210
C                                                                            AWRL0220
      REAL MRDRIN(8)     ,TBPE( 3)                                           AWRL0230
      CHARACTER DLSPM*5                                                      AWRL0240
C                                                                            AWRL0250
      DATA MRDRIN/8HACOUSTIC ,8H WAVE ** ,8H* WITH C ,8HURRENT *             AWRL0260
     1  ,8H** WITH  ,6HLOSSES ,2*1H /                                        AWRL0270
      DATA DLSPM/'AWCWL'/                                                    AWRL0280
      DATA TBPE/3*1H3 /
```

```
C                                                                       AWRL0290
C                                                                       AWRL0300
      ENTRY SETDSP                                                      AWRL0310
C                                                                       AWRL0320
      CALL RMOVE(MODRIN,MRDRIN,8)                                       AWRL0330
      DISPM=DLSPM                                                       AWRL0340
      CALL RMOVE(TYPE,TBPE, 3)                                          AWRL0350
C                                                                       AWRL0360
      CALL SETSPD                                                       AWRL0370
      CALL SETWND                                                       AWRL0380
      CALL SETRCV                                                       AWRL0390
      CALL SETTOP                                                       AWRL0400
      CALL SETSUR                                                       AWRL0410
      CALL SETABS                                                       AWRL0420
      RETURN                                                            AWRL0430
C                                                                       AWRL0440
      ENTRY IDISPER                                                     AWRL0450
      CALL ISPEED                                                       AWRL0460
      CALL IWINDR                                                       AWRL0470
      CALL IRECVR                                                       AWRL0480
      CALL ITOPOG                                                       AWRL0490
      CALL ISURFAC                                                      AWRL0500
C                                                                       AWRL0510
      CALL IABSRP                                                       AWRL0520
      RETURN                                                            AWRL0530
C                                                                       AWRL0540
      ENTRY DISPER                                                      AWRL0550
C                                                                       AWRL0560
      SPACE=.FALSE.                                                     AWRL0570
C                                                                       AWRL0580
      KS=KR*KR+KTH*KTH+KPH*KPH                                          AWRL0590
C     WIND VELOCITY                                                     AWRL0600
      CALL WINDR                                                        AWRL0610
      KV=KR*VR+KTH*VTH+KPH*VPH                                          AWRL0620
      VLS=KV*KV/KS                                                      AWRL0630
C     SOUND SPEED                                                       AWRL0640
      CALL SPEED                                                        AWRL0650
      OWS=OW*OW                                                         AWRL0660
      KAY2=OWS/(SQRT(CS)+KV/SQRT(KS))**2                                AWRL0670
C                                                                       AWRL0680
      OWI=OW-KV                                                         AWRL0690
      H=OWI*OWI - CS*KS                                                 AWRL0700
      POWIT=-KR*PVRT - KTH*PVTHT - KPH*PVPHT                            AWRL0710
      PHT=2.0*OWI*POWIT - KS*PCST                                       AWRL0720
      POWIR=-KR*PVRR - KTH*PVTHR - KPH*PVPHR                            AWRL0730
      PHR=2.0*OWI*POWIR - KS*PCSR                                       AWRL0740
      POWITH=-KR*PVRTH - KTH*PVTHTH - KPH*PVPHTH                        AWRL0750
      PHTH=2.0*OWI*POWITH - KS*PCSTH                                    AWRL0760
      POWIPH=-KR*PVRPH - KTH*PVTHPH - KPH*PVPHPH                        AWRL0770
      PHPH=2.0*OWI*POWIPH - KS*PCSPH                                    AWRL0780
C                                                                       AWRL0790
      PHOW=2.0*OWI                                                      AWRL0800
      PHKR=-2.0*(OWI*VR + CS*KR)                                        AWRL0810
      PHKTH=-2.0*(OWI*VTH + CS*KTH)                                     AWRL0820
      PHKPH=-2.0*(OWI*VPH + CS*KPH)                                     AWRL0830
```

```
      KPHK=-2.0*(OWI*KV + CS*KS)                                        AWRL0840
                                                                        AWRL0850
C                                                                       AWRL0860
      CALL ABSRP                                                        AWRL0870
      GMS=GAMMA-1.0                                                     AWRL0880
      KAY2I=-SQRT(KAY2)*APH                                             AWRL0890
      RETURN                                                            AWRL0900
      END
```

D.3 OCEAN MODEL ROUTINES (Tape File 5)

```
      SUBROUTINE WLINEAR                                                    WLTR0020
C     LINEAR WIND VELOCITY PROFILE                                          WLTR0030
C     PROVIDES CONSTANT RADIAL, ZONAL AND MERIDONAL WINDS                   WLTR0040
C     EXCEPT THAT A POSSIBLE LINEAR HEIGHT GRADIENT OF THE ZONAL            WLTR0050
C     COMPONENT IS ALLOWED.                                                 WLTR0060
C     COMMON DECK "RKAM" INSERTED HERE                                      RKAM0020
      REAL KR,KTH,KPH                                                       RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)           RKAM0050
C     COMMON DECK "UU" INSERTED HERE                                        CUU 0020
       REAL MODU                                                            CUU 0040
       COMMON/UU/MODU(4)                                                    CUU 0050
     1  ,V  ,PVT  ,PVR  ,PVTH  ,PVPH                                        CUU 0060
     2  ,VR ,PVRT ,PVRR ,PVRTH ,PVRPH                                       CUU 0070
     3  ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                      CUU 0080
     4  ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                      CUU 0090
C     COMMON DECK "WW" INSERTED HERE                                        CWW 0020
       PARAMETER (NWARSZ=1000)                                              CWW10030
       COMMON/WW/ID(10),MAXW,W(NWARSZ)                                      CWW10040
       REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                   CWW20020
       EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),       CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),          CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),              CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),             CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                     CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                          CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                          CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                         CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),           CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))            CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),            CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                  CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                       CWW20160
       REAL MMODEL,MFORM,MID                                                CWW30020
C                                                                           CWW30030
C     WIND         100-124                                                  CWW30040
       EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)              CWW30050
C                                                                           CWW30060
C     DELTA WIND         125-149                                           CWW30070
       EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)           CWW30080
C                                                                           CWW30090
C     SOUND SPEED 150-174                                                   CWW30100
       EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)              CWW30110
       EQUIVALENCE (W(153),REFC)                                            CWW30120
C                                                                           CWW30130
C     DELTA SOUND SPEED 175-199                                             CWW30140
       EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)           CWW30150
C                                                                           CWW30160
C     TEMPERATURE    200-224                                                CWW30170
       EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)              CWW30180
C                                                                           CWW30190
C     DELTA TEMPERATURE    225-249                                          CWW30200
       EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)           CWW30210
C                                                                           CWW30220
C     MOLECULAR   250-274                                                   CWW30230
```

```
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)          CWW30240
                                                                       CWW30250
C
C     RECEIVER HEIGHT   275-299                                        CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)          CWW30270
                                                                       CWW30280
C
C     TOPOGRAPHY   300-324                                             CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)          CWW30300
                                                                       CWW30310
C
C     DELTA TOPOGRAPHY   325-349                                       CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)       CWW30330
                                                                       CWW30340
C
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                         CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)          CWW30360
                                                                       CWW30370
C
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                         CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)       CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                             CWW30400
C                                                                      CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                      CWW30420
C     ABSORPTION         500-524                                       CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)          CWW30440
                                                                       CWW30450
C
C     DELTA ABSORPTION         525-549                                 CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)       CWW30470
                                                                       CWW30480
C
C     PRESSURE         550-574                                         CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)          CWW30500
                                                                       CWW30510
C
C     DELTA PRESSURE         575-599                                   CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)       CWW30530
                                                                       CWW30540
C
C     COMMON DECK "B1" INSERTED HERE                                   CB1 0020
      INTEGER UMX,UNTBL,UITBL,UFRMTBL,IDSU(10)                         CB1 0040
      COMMON/B1/UMX,UNTBL(10),UITBL(10),UFRMTBL(10),UGP(10)            CB1 0050
      EQUIVALENCE (UGP,IDSU)                                           CB1 0060
      EQUIVALENCE (W(103),UR0)                                         WLTR0110
      EQUIVALENCE (W(104),UTH0),(W(105),UPH0),(W(106),PUPHZ0)          WLTR0120
C                                                                      WLTR0130
      INTEGER UPX     ,UQTBL(10),ULTBL(10),UIRMTBL(10)                 WLTR0140
      DATA RECOGU/1.0/                                                 WLTR0150
C                                                                      WLTR0160
      DATA UPX/1/                                                      WLTR0170
      DATA UQTBL/1,11,8*0/                                             WLTR0180
      DATA ULTBL/1,9*0/                                                WLTR0190
      DATA UIRMTBL/1,9*0/                                              WLTR0200
C                                                                      WLTR0210
C                                                                      WLTR0220
      ENTRY SETWND                                                     WLTR0230
C                                                                      WLTR0240
      UMX=UPX                                                          WLTR0250
      CALL IMOVE(UNTBL,UQTBL,10)                                       WLTR0260
      CALL IMOVE(UITBL,ULTBL,10)                                       WLTR0270
      CALL IMOVE(UFRMTBL,UIRMTBL,10)                                   WLTR0280
      CALL SETPWN                                                      WLTR0290
C                                                                      WLTR0300
```

```
      RETURN                                                            WLTR0310
C                                                                       WLTR0320
      ENTRY IWINDR                                                      WLTR0330
C                                                                       WLTR0340
      IF(RECOGU .NE. UMODEL)                                            WLTR0350
     1    CALL RERROR('SPEED  ','WRNG MODEL',RECOGU)                   WLTR0360
C                                                                       WLTR0370
      MODU(1)=7HWLINEAR                                                 WLTR0380
      MODU(2)=UID                                                       WLTR0390
      CALL IPWINDR                                                      WLTR0400
C                                                                       WLTR0410
      RETURN                                                            WLTR0420
C                                                                       WLTR0430
       ENTRY WINDR                                                      WLTR0440
       H = R - EARTHR                                                   WLTR0450
C                                                                       WLTR0460
      CALL CLEAR(V,20)                                                  WLTR0470
C                                                                       WLTR0480
       VR = UR0                                                         WLTR0490
       VTH = UTH0                                                       WLTR0500
       VPH = (UPH0 + PUPHZ0 * H)                                        WLTR0510
      V=SQRT(VR*VR+VTH*VTH+VPH*VPH)                                     WLTR0520
      PVR=PUPHZ0                                                        WLTR0530
      IF(V.NE.0.0) PVR=VPH/V*PUPHZ0                                     WLTR0540
      PVPHR = PUPHZ0                                                    WLTR0550
      CALL PWINDR                                                       WLTR0560
       RETURN                                                           WLTR0570
       END                                                              WLTR0580
```

```
      SUBROUTINE VVORTX3                                                VVX30020
C     WIND VELOCITY MODEL                                               VVX30030
C     VERTICAL VORTEX WIND PERTURBATION WITH VISCOUS CORE AND           VVX30040
C     MULTIPLIES VELOCITY FIELD BY A GUASSIAN HEIGHT PROFILE.           VVX30050
C                                                                       VVX30060
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "UU" INSERTED HERE                                    CUU 0020
      REAL MODU                                                         CUU 0040
      COMMON/UU/MODU(4)                                                 CUU 0050
     1   ,V   ,PVT   ,PVR   ,PVTH   ,PVPH                              CUU 0060
     2   ,VR  ,PVRT  ,PVRR  ,PVRTH  ,PVRPH                             CUU 0070
     3   ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                CUU 0080
     4   ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                CUU 0090
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
```

```
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),     CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND         125-149                                        CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
C                                                                       CWW30090
C     SOUND SPEED 150-174                                               CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)           CWW30110
      EQUIVALENCE (W(153),REFC)                                         CWW30120
C                                                                       CWW30130
C     DELTA SOUND SPEED 175-199                                         CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)        CWW30150
C                                                                       CWW30160
C     TEMPERATURE    200-224                                            CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)           CWW30180
C                                                                       CWW30190
C     DELTA TEMPERATURE    225-249                                      CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)        CWW30210
C                                                                       CWW30220
C     MOLECULAR   250-274                                               CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)           CWW30240
C                                                                       CWW30250
C     RECEIVER HEIGHT   275-299                                         CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)           CWW30270
C                                                                       CWW30280
C     TOPOGRAPHY   300-324                                              CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)           CWW30300
C                                                                       CWW30310
C     DELTA TOPOGRAPHY   325-349                                        CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)        CWW30330
C                                                                       CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                          CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)           CWW30360
C                                                                       CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                          CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)        CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                              CWW30400
C                                                                       CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                       CWW30420
```

```
C       ABSORPTION          500-524                                          CWW30430
        EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)              CWW30440
C                                                                            CWW30450
C       DELTA ABSORPTION          525-549                                    CWW30460
        EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)           CWW30470
C                                                                            CWW30480
C       PRESSURE          550-574                                            CWW30490
        EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)              CWW30500
C                                                                            CWW30510
C       DELTA PRESSURE          575-599                                      CWW30520
        EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)           CWW30530
C                                                                            CWW30540
         EQUIVALENCE (U0,W(103)),(R0,W(104)),(TH0,W(105)),(PH0,W(106))       VVX30110
         EQUIVALENCE (HWIDTH,W(107)),(HVMAX,W(108))                          VVX30120
C       COMMON DECK "B1" INSERTED HERE                                       CB1 0020
        INTEGER UMX,UNTBL,UITBL,UFRMTBL,IDSU(10)                             CB1 0040
        COMMON/B1/UMX,UNTBL(10),UITBL(10),UFRMTBL(10),UGP(10)                CB1 0050
        EQUIVALENCE (UGP,IDSU)                                               CB1 0060
        INTEGER UPX     ,UQTBL(10),ULTBL(10),UIRMTBL(10)                     VVX30140
        DATA RECOGU/9.0/                                                     VVX30150
C                                                                            VVX30160
        DATA UPX/1/                                                          VVX30170
        DATA UQTBL/1,11,8*0/                                                 VVX30180
        DATA ULTBL/1,9*0/                                                    VVX30190
        DATA UIRMTBL/1,9*0/                                                  VVX30200
C                                                                            VVX30210
C                                                                            VVX30220
        ENTRY SETWND                                                         VVX30230
C                                                                            VVX30240
        UMX=UPX                                                              VVX30250
        CALL IMOVE(UNTBL,UQTBL,10)                                           VVX30260
        CALL IMOVE(UITBL,ULTBL,10)                                           VVX30270
        CALL IMOVE(UFRMTBL,UIRMTBL,10)                                       VVX30280
        CALL SETPWN                                                          VVX30290
C                                                                            VVX30300
        RETURN                                                               VVX30310
C                                                                            VVX30320
        ENTRY IWINDR                                                         VVX30330
C                                                                            VVX30340
        IF(RECOGU .NE. UMODEL)                                               VVX30350
     1     CALL RERROR('SPEED  ','WRNG MODEL',RECOGU)                        VVX30360
C                                                                            VVX30370
        MODU(1)=7HVVORTX3                                                    VVX30380
        MODU(2)=UID                                                          VVX30390
         DENOM=0.0                                                           VVX30400
         IF(HWIDTH.NE.0.0) DENOM=1.0/HWIDTH**2                               VVX30410
        CALL IPWINDR                                                         VVX30420
C                                                                            VVX30430
        RETURN                                                               VVX30440
C                                                                            VVX30450
         ENTRY WINDR                                                         VVX30460
        CALL CLEAR(V,20)                                                     VVX30470
        DR=R-EARTHR-HVMAX                                                    VVX30480
         DTH = TH - (PID2-TH0)                                               VVX30490
         DPH = PH - PH0                                                      VVX30500
```

```
 RAD2 = EARTHR * SQRT(DTH * DTH + DPH * DPH)                          VVX30510
 A = 1.397                                                            VVX30520
 B = - 1.26                                                           VVX30530
EXPO=RAD2/R0                                                          VVX30540
EXPO=B*EXPO*EXPO                                                      VVX30550
EXB=0.0                                                               VVX30560
 IF(EXPO .GT. -675.0) EXB = EXP(EXPO)                                 VVX30570
FX=1.-EXB                                                             VVX30580
 DUM = A * EARTHR * U0 * R0 / RAD2 * * 2                              VVX30590
FZ=EXP(-DR*DR*DENOM)                                                  VVX30600
DFDZ=-2.*DR*DENOM                                                     VVX30610
DUM=FZ*DUM                                                            VVX30620
C                                                                     VVX30630
 DUX = FX / RAD2 + RAD2 * B * EXB / R0 * * 2                          VVX30640
 VTH = - DUM * FX * DPH                                               VVX30650
 VPH = DUM * FX * DTH                                                 VVX30660
V=SQRT(VTH*VTH + VPH*VPH)                                             VVX30670
 DUM2=2.*DUM*EARTHR*EARTHR                                            VVX30680
 PVTHTH = DUM2 * DTH * DPH / RAD2 * DUX                               VVX30690
 PVPHPH = - PVTHTH                                                    VVX30700
 PVTHPH = DPH**2 * DUM2 / RAD2 * DUX -DUM*FX                          VVX30710
 PVPHTH = - DTH**2* DUM2 / RAD2* DUX + DUM*FX                         VVX30720
C                                                                     VVX30730
PVTH=(VTH*PVTHTH + VPH*PVPHTH)/V                                      VVX30740
PVPH=(VTH*PVTHPH + VPH*PVPHPH)/V                                      VVX30750
C                                                                     VVX30760
PVTHR=VTH*DFDZ                                                        VVX30770
PVPHR=VPH*DFDZ                                                        VVX30780
PVR=(VTH*PVTHR+VPH*PVPHR)/V                                           VVX30790
C                                                                     VVX30800
CALL PWINDR                                                           VVX30810
 RETURN                                                               VVX30820
 END                                                                  VVX30830
```

```
      SUBROUTINE WGAUSS2                                              WGZ20020
C     WIND VELOCITY MODEL                                             WGZ20030
C     EXPONENTIALLY DECAYING EFFECT IN ALL THREE DIRECTIONS.          WGZ20040
C     COMMON DECK "CONST" INSERTED HERE                               CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                   CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                       CCON0050
C     COMMON DECK "RKAM" INSERTED HERE                                RKAM0020
      REAL KR,KTH,KPH                                                 RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)     RKAM0050
C     COMMON DECK "UU" INSERTED HERE                                  CUU 0020
      REAL MODU                                                       CUU 0040
      COMMON/UU/MODU(4)                                               CUU 0050
     1  ,V  ,PVT  ,PVR  ,PVTH  ,PVPH                                   CUU 0060
     2  ,VR ,PVRT ,PVRR ,PVRTH ,PVRPH                                  CUU 0070
     3  ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                 CUU 0080
     4  ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                 CUU 0090
C     COMMON DECK "WW" INSERTED HERE                                  CWW 0020
```

```
      PARAMETER (NWARSZ=1000)                                                   CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                           CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                        CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),            CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),              CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),                  CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),                 CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                         CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))        CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                              CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                              CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                             CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),               CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))                CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),                CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                      CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                           CWW20160
      REAL MMODEL,MFORM,MID                                                     CWW30020
C                                                                               CWW30030
C     WIND          100-124                                                   CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)                   CWW30050
C                                                                               CWW30060
C     DELTA WIND          125-149                                               CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)                CWW30080
C                                                                               CWW30090
C     SOUND SPEED 150-174                                                       CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)                   CWW30110
      EQUIVALENCE (W(153),REFC)                                                 CWW30120
C                                                                               CWW30130
C     DELTA SOUND SPEED 175-199                                                 CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)                CWW30150
C                                                                               CWW30160
C     TEMPERATURE    200-224                                                    CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)                   CWW30180
C                                                                               CWW30190
C     DELTA TEMPERATURE    225-249                                              CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)                CWW30210
C                                                                               CWW30220
C     MOLECULAR   250-274                                                       CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)                   CWW30240
C                                                                               CWW30250
C     RECEIVER HEIGHT   275-299                                                 CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)                   CWW30270
C                                                                               CWW30280
C     TOPOGRAPHY    300-324                                                     CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)                   CWW30300
C                                                                               CWW30310
C     DELTA TOPOGRAPHY   325-349                                                CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)                CWW30330
C                                                                               CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                                  CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)                   CWW30360
C                                                                               CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                                  CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)                CWW30390
```

```
C       PLOT ENHANCEMENTS CONTROL PARAMETERS                                  CWW30400
C                                                                             CWW30410
C                                                                             CWW30420
        EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                           CWW30430
C       ABSORPTION          500-524                                           CWW30440
        EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)               CWW30450
C                                                                             CWW30460
C       DELTA ABSORPTION          525-549                                     CWW30470
        EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)            CWW30480
C                                                                             CWW30490
C       PRESSURE          550-574                                             CWW30500
        EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)               CWW30510
C                                                                             CWW30520
C       DELTA PRESSURE          575-599                                       CWW30530
        EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)            CWW30540
C                                                                             CB1 0020
C       COMMON DECK "B1" INSERTED HERE                                        CB1 0040
        INTEGER UMX,UNTBL,UITBL,UFRMTBL,IDSU(10)                              CB1 0050
        COMMON/B1/UMX,UNTBL(10),UITBL(10),UFRMTBL(10),UGP(10)                 CB1 0060
        EQUIVALENCE (UGP,IDSU)                                                WGZ20100
        INTEGER UPX,UQTBL(10),ULTBL(10),UIRMTBL(10)                           WGZ20110
C                                                                             WGZ20120
        EQUIVALENCE (UPH0,W(103)),(WH,W(104)),(WTH,W(105))                    WGZ20130
        EQUIVALENCE  (WPH,W(106)),(H0,W(107)),(WGTH0,W(108)),(PH0,W(109))     WGZ20140
        DATA RECOGU/7.0/                                                      WGZ20150
        DATA UPX/1/                                                           WGZ20160
        DATA UQTBL/1,11,8*0/                                                  WGZ20170
        DATA ULTBL/1,9*0/                                                     WGZ20180
        DATA UIRMTBL/1,9*0/                                                   WGZ20190
C                                                                             WGZ20200
        ENTRY SETWND                                                          WGZ20210
        UMX=UPX                                                               WGZ20220
        CALL IMOVE(UNTBL,UQTBL,10)                                            WGZ20230
        CALL IMOVE(UITBL,ULTBL,10)                                            WGZ20240
        CALL IMOVE(UFRMTBL,UIRMTBL,10)                                        WGZ20250
        CALL SETPWN                                                           WGZ20260
        RETURN                                                                WGZ20270
C                                                                             WGZ20280
        ENTRY IWINDR                                                          WGZ20290
C                                                                             WGZ20300
        IF(RECOGU .NE. UMODEL)                                                WGZ20310
     1    CALL RERROR('SPEED   ','WRNG MODEL',RECOGU)                        WGZ20320
C                                                                             WGZ20330
        MODU(1)=7HWGAUSS2                                                     WGZ20340
        MODU(2)=UID                                                           WGZ20350
        CALL IPWINDR                                                          WGZ20360
C                                                                             WGZ20370
        WIDH=0.0                                                              WGZ20380
        WIDTH=0.0                                                             WGZ20390
        WIDPH=0.0                                                             WGZ20400
        TH0= PID2-WGTH0                                                       WGZ20410
        IF(WH.NE.0.0) WIDH=1.0/WH                                             WGZ20420
        IF(WTH.NE.0.0) WIDTH=1.0/WTH                                          WGZ20430
        IF(WPH.NE.0.0) WIDPH=1.0/WPH                                          WGZ20440
        RETURN                                                                WGZ20450
C
```

```
 ENTRY WINDR                                                            WGZ20460
CALL CLEAR(V,20)                                                        WGZ20470
 H = R - EARTHR                                                         WGZ20480
DFH=(H-H0)*WIDH                                                         WGZ20490
DFTH=(TH-TH0)*WIDTH                                                     WGZ20500
DFPH=(PH-PH0)*WIDPH                                                     WGZ20510
EXPO=-(DFH*DFH+DFTH*DFTH+DFPH*DFPH)                                     WGZ20520
EXPN=0.0                                                                WGZ20530
IF(EXPO.GT.-200.0) EXPN=EXP(EXPO)                                       WGZ20540
 VPH = UPH0*EXPN                                                        WGZ20550
 PVPHR = - 2. * VPH * DFH*WIDH                                          WGZ20560
 PVPHTH = - 2. * VPH * DFTH*WIDTH                                       WGZ20570
 PVPHPH = - 2. * VPH * DFPH*WIDPH                                       WGZ20580
 END                                                                    WGZ20590



      SUBROUTINE NPCURR                                                 NPTR0020
C     DO-NOTHING CURRENT PERTURBATION MODEL                             NPTR0030
C     COMMON DECK "UU" INSERTED HERE                                    CUU 0020
      REAL MODU                                                         CUU 0040
      COMMON/UU/MODU(4)                                                 CUU 0050
     1  ,V  ,PVT  ,PVR  ,PVTH  ,PVPH                                     CUU 0060
     2  ,VR ,PVRT ,PVRR ,PVRTH ,PVRPH                                    CUU 0070
     3  ,VTH,PVTHT,PVTHR,PVTHTH,PVTHPH                                   CUU 0080
     4  ,VPH,PVPHT,PVPHR,PVPHTH,PVPHPH                                   CUU 0090
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND          100-124                                           CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND          125-149                                       CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
C                                                                       CWW30090
C     SOUND SPEED 150-174                                               CWW30100
```

```
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)          CWW30110
      EQUIVALENCE (W(153),REFC)                                       CWW30120
C                                                                     CWW30130
C     DELTA SOUND SPEED 175-199                                       CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)      CWW30150
C                                                                     CWW30160
C     TEMPERATURE    200-224                                          CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)         CWW30180
C                                                                     CWW30190
C     DELTA TEMPERATURE    225-249                                    CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)      CWW30210
C                                                                     CWW30220
C     MOLECULAR   250-274                                             CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)         CWW30240
C                                                                     CWW30250
C     RECEIVER HEIGHT   275-299                                       CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)         CWW30270
C                                                                     CWW30280
C     TOPOGRAPHY   300-324                                            CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)         CWW30300
C                                                                     CWW30310
C     DELTA TOPOGRAPHY   325-349                                      CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)      CWW30330
C                                                                     CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                        CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)         CWW30360
C                                                                     CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                        CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)      CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                            CWW30400
C                                                                     CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                     CWW30420
C     ABSORPTION         500-524                                      CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)         CWW30440
C                                                                     CWW30450
C     DELTA ABSORPTION         525-549                                CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)      CWW30470
C                                                                     CWW30480
C     PRESSURE         550-574                                        CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)         CWW30500
C                                                                     CWW30510
C     DELTA PRESSURE         575-599                                  CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)      CWW30530
C                                                                     CWW30540
C     COMMON DECK "B2" INSERTED HERE                                  CB2 0020
      INTEGER DUMX,DUNTBL,DUITBL,DUFRMTB,IDSDU(10)                    CB2 0040
      COMMON/B2/DUMX,DUNTBL(10),DUITBL(10),DUFRMTB(10),DUGP(10)       CB2 0050
      EQUIVALENCE (DUGP,IDSDU)                                        CB2 0060
      INTEGER DXMX    ,DXNTBL(10),DXITBL(10),DXFRMTB(10)              NPTR0070
      DATA RECOGDU/0.0/                                               NPTR0080
C                                                                     NPTR0090
      DATA DXMX/1/                                                    NPTR0100
      DATA DXNTBL/1,11,8*0/                                           NPTR0110
      DATA DXITBL/1,9*0/                                              NPTR0120
      DATA DXFRMTB/1,9*0/                                             NPTR0130
```

```
C                                                                       NPTR0140
C                                                                       NPTR0150
      ENTRY SETPWN                                                      NPTR0160
C                                                                       NPTR0170
      DUMX=DXMX                                                         NPTR0180
      CALL IMOVE(DUNTBL,DXNTBL,10)                                      NPTR0190
      CALL IMOVE(DUITBL,DXITBL,10)                                      NPTR0200
      CALL IMOVE(DUFRMTB,DXFRMTB,10)                                    NPTR0210
C                                                                       NPTR0220
      RETURN                                                            NPTR0230
C                                                                       NPTR0240
      ENTRY IPWINDR                                                     NPTR0250
      IF(RECOGDU .NE. DUMODEL)                                          NPTR0260
     1    CALL RERROR('DWINDR ','WRNG MODEL',RECOGDU)                   NPTR0270
      MODU(3)=6HNPCURR                                                  NPTR0280
      MODU(4)=DUID                                                      NPTR0290
      RETURN                                                            NPTR0300
C                                                                       NPTR0310
      ENTRY PWINDR                                                      NPTR0320
      RETURN                                                            NPTR0330
      END                                                               NPTR0340
```

```
      SUBROUTINE CTANH                                                  CTLH0020
C     SPEED PROFILE REPRESENTED BY A SEQUENCE OF LINEAR SEGMENTS        CTLH0030
C     SMOOTHLY JOINED BY HYPERBOLIC FUNCTIONS. PARAMETERS ARE INPUT     CTLH0040
C     AS TABULAR DATA WITH SLOPES COMPUTED FROM SPEED DATA.             CTLH0050
C     REFERENCE SPEED C0 IS READ FROM TABULAR DATA.                     CTLH0060
C                                                                       CTLH0070
      REAL CO(20), TM(19), Z(19), DL(19)                                CTLH0080
C                                                                       CTLH0090
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "CC" INSERTED HERE                                    CCC 0020
      REAL MODC                                                         CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                        CCC 0050
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
```

```
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),          CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                     CWW20160
      REAL MMODEL,MFORM,MID                                               CWW30020
C                                                                         CWW30030
C     WIND         100-124                                              CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)             CWW30050
C                                                                         CWW30060
C     DELTA WIND         125-149                                          CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)          CWW30080
C                                                                         CWW30090
C     SOUND SPEED 150-174                                                 CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)             CWW30110
      EQUIVALENCE (W(153),REFC)                                           CWW30120
C                                                                         CWW30130
C     DELTA SOUND SPEED 175-199                                           CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)          CWW30150
C                                                                         CWW30160
C     TEMPERATURE    200-224                                              CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)             CWW30180
C                                                                         CWW30190
C     DELTA TEMPERATURE    225-249                                        CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)          CWW30210
C                                                                         CWW30220
C     MOLECULAR   250-274                                                 CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)             CWW30240
C                                                                         CWW30250
C     RECEIVER HEIGHT   275-299                                           CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)             CWW30270
C                                                                         CWW30280
C     TOPOGRAPHY   300-324                                                CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)             CWW30300
C                                                                         CWW30310
C     DELTA TOPOGRAPHY   325-349                                          CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)          CWW30330
C                                                                         CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                            CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)             CWW30360
C                                                                         CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                            CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)          CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                CWW30400
C                                                                         CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                         CWW30420
C     ABSORPTION         500-524                                          CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)             CWW30440
C                                                                         CWW30450
C     DELTA ABSORPTION         525-549                                    CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)          CWW30470
C                                                                         CWW30480
C     PRESSURE         550-574                                            CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)             CWW30500
C                                                                         CWW30510
C     DELTA PRESSURE         575-599                                      CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)          CWW30530
```

```
C                                                                       CWW30540
C     COMMON DECK "B3" INSERTED HERE                                    CB3 0020
      INTEGER CMX,CNTBL,CITBL,CFRMTBL,IDSC(10)                          CB3 0040
      COMMON/B3/CMX,CNTBL(10),CITBL(10),CFRMTBL(10),CGP(512)            CB3 0050
      EQUIVALENCE (CGP,IDSC),(ANC,CGP(11))                              CB3 0060
C                                                                       CTLH0140
      EQUIVALENCE (Z0,CGP(12)),(TM,CGP(33))                             CTLH0150
      EQUIVALENCE (Z,CGP(13)),(CO,CGP(32)),(DL,CGP(53))                 CTLH0160
C                                                                       CTLH0170
      INTEGER CPX     ,CQTBL(10),CLTBL(10),CIRMTBL(10)                  CTLH0180
      DATA RECOGC,N/7.0,0/                                              CTLH0190
C                                                                       CTLH0200
      DATA AQC/0.0/                                                     CTLH0210
      DATA CPX/2/                                                       CTLH0220
      DATA CQTBL/1,11,72,7*0/                                           CTLH0230
      DATA CLTBL/1,20,8*0/                                              CTLH0240
      DATA CIRMTBL/1,2,8*0/                                             CTLH0250
C                                                                       CTLH0260
      COSH (X) = (EXP (X) + 1. / (EXP (X))) / 2.                        CTLH0270
C                                                                       CTLH0280
C                                                                       CTLH0290
      ENTRY SETSPD                                                      CTLH0300
C                                                                       CTLH0310
      ANC=AQC                                                           CTLH0320
      CMX=CPX                                                           CTLH0330
      CALL IMOVE(CNTBL,CQTBL,10)                                        CTLH0340
      CALL IMOVE(CITBL,CLTBL,10)                                        CTLH0350
      CALL IMOVE(CFRMTBL,CIRMTBL,10)                                    CTLH0360
      CALL SETPSP                                                       CTLH0370
C                                                                       CTLH0380
      RETURN                                                            CTLH0390
C                                                                       CTLH0400
      ENTRY ISPEED                                                      CTLH0410
C                                                                       CTLH0420
C     IF HAD PREVIOUS CALL BUT NOTHING THIS TIME, EXIT NOW              CTLH0430
C     RETAINING PREVIOUS TABULAR DATA COUNT                             CTLH0440
C                                                                       CTLH0450
      CALL IPSPEED                                                      CTLH0460
C                                                                       CTLH0470
      IF(N.GT.0 .AND. ANC.EQ.0.0) RETURN                                CTLH0480
C                                                                       CTLH0490
      IF(RECOGC .NE. CMODEL)                                            CTLH0500
     1    CALL RERROR('SPEED   ','WRNG MODEL',RECOGC)                  CTLH0510
C                                                                       CTLH0520
      MODC(1)=7HCTANH                                                   CTLH0530
      MODC(2)=CID                                                       CTLH0540
C                                                                       CTLH0550
      N=(ANC+1)/3 - 2                                                   CTLH0560
C                                                                       CTLH0570
      IF(N.LE.0)                                                        CTLH0580
     1    CALL RERROR('CTANH','BAD N VALUE',FLOAT(N))                  CTLH0590
C                                                                       CTLH0600
      ANC=0.0                                                           CTLH0610
C                                                                       CTLH0620
C     CONVERT 'CGP' ARRAY INPUT(OVERLAYS 'C' ARRAY) TO 'C' ARRAY        CTLH0630
```

```
C                                                                       CTLH0640
      C0=CO(1)                                                          CTLH0650
      TIM1=C0                                                           CTLH0660
      ZIM1=Z0                                                           CTLH0670
      NP1=N+1                                                           CTLH0680
      DO 10 I=1,NP1                                                     CTLH0690
         TI=TM(I)                                                       CTLH0700
         ZI=Z(I)                                                        CTLH0710
         CO(I)=(TI-TIM1)/(ZI-ZIM1)                                      CTLH0720
         TIM1=TI                                                        CTLH0730
10       ZIM1=ZI                                                        CTLH0740
C                                                                       CTLH0750
      RETURN                                                            CTLH0760
C                                                                       CTLH0770
      ENTRY SPEED                                                       CTLH0780
      H = R - EARTHR                                                    CTLH0790
      SUM = 0.                                                          CTLH0800
C                                                                       CTLH0810
C     LOOP TO SUM OVER ALL COEFFICIENTS                                 CTLH0820
C     USE SPECIAL FUNCTION 'ALCOSH' WHICH ALLOWS FOR LARGE ARGUMENTS.   CTLH0830
      DO 1 I = 1, N                                                     CTLH0840
    1 SUM = SUM + DL(I) * (CO(I + 1) - CO(I)) / 2. *(ALCOSH((H - Z      CTLH0850
     1(I)) / DL(I)) - ALCOSH((Z(I)-Z0) / DL(I)))                        CTLH0860
C                                                                       CTLH0870
      C = C0 + SUM + (CO(1) + CO(N + 1)) * (H - Z0) * 0.5               CTLH0880
C                                                                       CTLH0890
      SUM = 0.                                                          CTLH0900
      DO 2 I = 1, N                                                     CTLH0910
    2 SUM = SUM + (CO(I + 1) - CO(I)) / 2. * (1. + TANH ((H - Z(I)) /DL CTLH0920
     1 (I)))                                                            CTLH0930
C                                                                       CTLH0940
      CS=C*C                                                            CTLH0950
C                                                                       CTLH0960
      PCST=0.0                                                          CTLH0970
      PCSR = 2.0*C*(CO(1) + SUM)                                        CTLH0980
      PCSTH=0.0                                                         CTLH0990
      PCSPH=0.0                                                         CTLH1000
C                                                                       CTLH1010
      CALL PSPEED                                                       CTLH1020
      RETURN                                                            CTLH1030
      END                                                               CTLH1040
```

```
      SUBROUTINE CSTANH                                                 CSRH0020
C     SPEED PROFILE REPRESENTED BY A SEQUENCE OF LINEAR SEGMENTS        CSRH0030
C     SMOOTHLY JOINED BY HYPERBOLIC FUNCTIONS. PARAMETERS ARE INPUT     CSRH0040
C     AS TABULAR DATA WITH SLOPES COMPUTED FROM SPEED DATA.             CSRH0050
C                                                                       CSRH0060
      REAL ALC(20),Z(19),B(19),DL(19)                                   CSRH0070
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
```

```
C       COMMON DECK "B3" INSERTED HERE                                          CB3 0020
        INTEGER CMX,CNTBL,CITBL,CFRMTBL,IDSC(10)                                CB3 0040
        COMMON/B3/CMX,CNTBL(10),CITBL(10),CFRMTBL(10),CGP(512)                  CB3 0050
        EQUIVALENCE (CGP,IDSC),(ANC,CGP(11))                                    CB3 0060
C       COMMON DECK "CC" INSERTED HERE                                          CCC 0020
        REAL MODC                                                               CCC 0040
        COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                              CCC 0050
C       COMMON DECK "CONST" INSERTED HERE                                       CCON0020
        COMMON/PCONST/CREF,RGAS,GAMMA                                           CCON0040
        COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                               CCON0050
C       COMMON DECK "WW" INSERTED HERE                                          CWW 0020
        PARAMETER (NWARSZ=1000)                                                 CWW10030
        COMMON/WW/ID(10),MAXW,W(NWARSZ)                                         CWW10040
        REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                      CWW20020
        EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),          CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),              CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),                  CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),                 CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                         CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))        CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                              CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                              CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                             CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),               CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))                CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),                CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                      CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                           CWW20160
        REAL MMODEL,MFORM,MID                                                   CWW30020
C                                                                               CWW30030
C       WIND         100-124                                                    CWW30040
        EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)                 CWW30050
C                                                                               CWW30060
C       DELTA WIND         125-149                                              CWW30070
        EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)              CWW30080
C                                                                               CWW30090
C       SOUND SPEED 150-174                                                     CWW30100
        EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)                 CWW30110
        EQUIVALENCE (W(153),REFC)                                               CWW30120
C                                                                               CWW30130
C       DELTA SOUND SPEED 175-199                                               CWW30140
        EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)              CWW30150
C                                                                               CWW30160
C       TEMPERATURE    200-224                                                  CWW30170
        EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)                 CWW30180
C                                                                               CWW30190
C       DELTA TEMPERATURE    225-249                                            CWW30200
        EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)              CWW30210
C                                                                               CWW30220
C       MOLECULAR   250-274                                                     CWW30230
        EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)                 CWW30240
C                                                                               CWW30250
C       RECEIVER HEIGHT   275-299                                               CWW30260
        EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)                 CWW30270
C                                                                               CWW30280
```

```
C       TOPOGRAPHY   300-324                                                 CWW30290
        EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)             CWW30300
                                                                            CWW30310
C                                                                           CWW30320
C       DELTA TOPOGRAPHY   325-349                                          CWW30330
        EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)          CWW30340
C                                                                           CWW30350
C       ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                            CWW30360
        EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)             CWW30370
C                                                                           CWW30380
C       ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                            CWW30390
        EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)          CWW30400
C       PLOT ENHANCEMENTS CONTROL PARAMETERS                                CWW30410
C                                                                           CWW30420
        EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                         CWW30430
C       ABSORPTION         500-524                                          CWW30440
        EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)             CWW30450
C                                                                           CWW30460
C       DELTA ABSORPTION         525-549                                    CWW30470
        EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)          CWW30480
C                                                                           CWW30490
C       PRESSURE         550-574                                            CWW30500
        EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)             CWW30510
C                                                                           CWW30520
C       DELTA PRESSURE         575-599                                      CWW30530
        EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)          CWW30540
C                                                                           CSRH0130
C                                                                           CSRH0140
        EQUIVALENCE (Z0,CGP(12)),(CS0,CGP(32)),(DL0,CGP(52))                CSRH0150
        EQUIVALENCE (Z,CGP(13)),(B,CGP(33)),(DL,CGP(53))                    CSRH0160
C                                                                           CSRH0170
        INTEGER CPX     ,CQTBL(10) ,CLTBL(10),CIRMTBL(10)                   CSRH0180
        DATA RECOGC,N/2.0,0/                                                CSRH0190
C                                                                           CSRH0200
        DATA PFST,PFSTH,PFSPH/3*0.0/                                        CSRH0210
        DATA AQC/0.0/                                                       CSRH0220
        DATA CPX/2/                                                         CSRH0230
        DATA CQTBL/1,11,72,7*0/                                             CSRH0240
        DATA CLTBL/1,20,8*0/                                                CSRH0250
        DATA CIRMTBL/1,2,8*0/                                               CSRH0260
C                                                                           CSRH0270
C                                                                           CSRH0280
        ENTRY SETSPD                                                        CSRH0290
C                                                                           CSRH0300
        PCST=PFST                                                           CSRH0310
        PCSTH=PFSTH                                                         CSRH0320
        PCSPH=PFSPH                                                         CSRH0330
        ANC=AQC                                                             CSRH0340
        CALL IMOVE(CMX,CPX, 1)                                              CSRH0350
        CALL IMOVE(CNTBL,CQTBL,10)                                          CSRH0360
        CALL IMOVE(CITBL,CLTBL,10)                                          CSRH0370
        CALL IMOVE(CFRMTBL,CIRMTBL,10)                                      CSRH0380
        CALL SETPSP                                                         CSRH0390
C                                                                           CSRH0400
        RETURN                                                              CSRH0410
C
```

```
      ENTRY ISPEED                                                          CSRH0420
C                                                                           CSRH0430
      IF(REFC.GT.0.0) CREF=REFC                                             CSRH0440
C                                                                           CSRH0450
      CALL IPSPEED                                                          CSRH0460
C                                                                           CSRH0470
C     IF HAD PREVIOUS CALL BUT NOTHING THIS TIME, EXIT NOW                  CSRH0480
C     RETAINING PREVIOUS TABULAR DATA COUNT                                 CSRH0490
      IF(N.GT.0 .AND. ANC.EQ.0.0) RETURN                                    CSRH0500
C                                                                           CSRH0510
       IF(RECOGC .NE. CMODEL)                                               CSRH0520
     1    CALL RERROR('SPEED  ','WRNG MODEL',RECOGC)                       CSRH0530
       MODC(1)=6HCSTANH                                                     CSRH0540
       MODC(2)=CID                                                          CSRH0550
       N=ANC/3                                                              CSRH0560
       IF(ANC.NE.3*N.OR.N.LE.0)                                             CSRH0570
     1    CALL RERROR('CSTANH','BAD NUMBER',ANC+2.0)                       CSRH0580
       N=N-2                                                                CSRH0590
       ANC=0.0                                                              CSRH0600
C                                                                           CSRH0610
C                                                                           CSRH0620
C     CONVERT 'C' ARRAY INPUT(OVERLAYS 'B' ARRAY) TO 'B' ARRAY              CSRH0630
C                                                                           CSRH0640
       ZM1=Z0                                                               CSRH0650
       CS0=CS0*CS0                                                          CSRH0660
       CSM1=CS0                                                             CSRH0670
       NP1=N+1                                                              CSRH0680
       DO 10 I=1,NP1                                                        CSRH0690
          ZR=Z(I)                                                           CSRH0700
          ALC(I)=ALCOSH((ZR-Z0) / DL(I))                                    CSRH0710
          CS=B(I)**2                                                        CSRH0720
          B(I)=(CS-CSM1)/(ZR-ZM1)                                           CSRH0730
          ZM1=ZR                                                            CSRH0740
10        CSM1=CS                                                           CSRH0750
C                                                                           CSRH0760
       RETURN                                                               CSRH0770
C                                                                           CSRH0780
       ENTRY SPEED                                                          CSRH0790
C                                                                           CSRH0800
       IF(N.LE.0)                                                           CSRH0810
     1    CALL RERROR('CSTANH','BAD N VALUE',FLOAT(N))                     CSRH0820
C                                                                           CSRH0830
       ZR=R-EARTHR                                                          CSRH0840
       SUM = 0.                                                             CSRH0850
       PCSR=B(1)                                                            CSRH0860
       DO 1 I = 1, N                                                        CSRH0870
       SAV=0.5*(B(I+1)-B(I))                                                CSRH0880
       PCSR= PCSR+ SAV * (1.+TANH ((ZR-Z(I)) /DL(I)))                       CSRH0890
    1  SUM = SUM+DL(I) * SAV *(ALCOSH((ZR-Z(I))/DL(I))-ALC(I))              CSRH0900
       CS = CS0+SUM + 0.5*(B(1) + B(N + 1)) * (ZR-Z0)                       CSRH0910
C                                                                           CSRH0920
       CALL PSPEED                                                          CSRH0930
       RETURN                                                               CSRH0940
       END                                                                  CSRH0950
```

```
      SUBROUTINE CSSPOKE                                                CSTE0020
C     SOUND SPEED SQUARED IS MODELED AS A FUNCTION OF ANGLE 'ALPHA'     CSTE0030
C     ABOUT A HORIZONTAL LINE AT SPECIFIED HEIGHT AND LATITUDE. DEPENDE CSTE0040
C     ON 'ALPHA' IS AS A SEQUENCE OF LINEAR SEGMENTS JOINED BY HYPERBOL CSTE0050
C     TANGENTS.                                                         CSTE0060
C                                                                       CSTE0070
      REAL ALC(20),Z(19),BIN(19),B(19),DL(19),LAM0                      CSTE0080
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "B3" INSERTED HERE                                    CB3 0020
      INTEGER CMX,CNTBL,CITBL,CFRMTBL,IDSC(10)                          CB3 0040
      COMMON/B3/CMX,CNTBL(10),CITBL(10),CFRMTBL(10),CGP(512)            CB3 0050
      EQUIVALENCE (CGP,IDSC),(ANC,CGP(11))                              CB3 0060
C     COMMON DECK "CC" INSERTED HERE                                    CCC 0020
      REAL MODC                                                         CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                        CCC 0050
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
       EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),   CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND          100-124                                           CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND          125-149                                       CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
C                                                                       CWW30090
C     SOUND SPEED 150-174                                               CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)           CWW30110
      EQUIVALENCE (W(153),REFC)                                         CWW30120
C                                                                       CWW30130
C     DELTA SOUND SPEED 175-199                                         CWW30140
```

```
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)          CWW30150
C                                                                         CWW30160
C     TEMPERATURE    200-224                                              CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)             CWW30180
C                                                                         CWW30190
C     DELTA TEMPERATURE    225-249                                        CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)          CWW30210
C                                                                         CWW30220
C     MOLECULAR   250-274                                                 CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)             CWW30240
C                                                                         CWW30250
C     RECEIVER HEIGHT   275-299                                           CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)             CWW30270
C                                                                         CWW30280
C     TOPOGRAPHY   300-324                                                CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)             CWW30300
C                                                                         CWW30310
C     DELTA TOPOGRAPHY   325-349                                          CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)          CWW30330
C                                                                         CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                            CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)             CWW30360
C                                                                         CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                            CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)          CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                CWW30400
C                                                                         CWW30410
C                                                                         CWW30420
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                         CWW30430
C     ABSORPTION         500-524                                          CWW30440
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)             CWW30450
C                                                                         CWW30460
C     DELTA ABSORPTION         525-549                                    CWW30470
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)          CWW30480
C                                                                         CWW30490
C     PRESSURE         550-574                                            CWW30500
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)             CWW30510
C                                                                         CWW30520
C     DELTA PRESSURE         575-599                                      CWW30530
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)          CWW30540
C                                                                         CSTE0140
      EQUIVALENCE (Z0,CGP(12)),(C0,CGP(32)),(DL0,CGP(52))                 CSTE0150
      EQUIVALENCE (Z,CGP(13)),(BIN,CGP(33)),(DL,CGP(53))                  CSTE0160
      EQUIVALENCE (REFH,W(154)), (LAM0,W(155))                            CSTE0170
C                                                                         CSTE0180
      INTEGER CPX    ,CQTBL(10) ,CLTBL(10),CIRMTBL(10)                    CSTE0190
      DATA RECOGC,N/2.0,0/                                                CSTE0200
C                                                                         CSTE0210
      DATA PFST,PFSTH,PFSPH/3*0.0/                                        CSTE0220
      DATA AQC/0.0/                                                       CSTE0230
      DATA CPX/2/                                                         CSTE0240
      DATA CQTBL/1,11,72,7*0/                                             CSTE0250
      DATA CLTBL/1,20,8*0/                                                CSTE0260
      DATA CIRMTBL/1,2,8*0/                                               CSTE0270
C                                                                         CSTE0280
C
```

```
      ENTRY SETSPD                                                      CSTE0290
C                                                                       CSTE0300
      PCST=PFST                                                         CSTE0310
      PCSTH=PFSTH                                                       CSTE0320
      PCSPH=PFSPH                                                       CSTE0330
      ANC=AQC                                                           CSTE0340
      CMX=CPX                                                           CSTE0350
      CALL IMOVE(CNTBL,CQTBL,10)                                        CSTE0360
      CALL IMOVE(CITBL,CLTBL,10)                                        CSTE0370
      CALL IMOVE(CFRMTBL,CIRMTBL,10)                                    CSTE0380
      CALL SETPSP                                                       CSTE0390
C                                                                       CSTE0400
      RETURN                                                            CSTE0410
C                                                                       CSTE0420
      ENTRY ISPEED                                                      CSTE0430
C                                                                       CSTE0440
      IF(REFC.GT.0) CREF=REFC                                           CSTE0450
C                                                                       CSTE0460
      CALL IPSPEED                                                      CSTE0470
C                                                                       CSTE0480
C     IF HAD PREVIOUS CALL BUT NOTHING THIS TIME, EXIT NOW              CSTE0490
C     RETAINING PREVIOUS TABULAR DATA COUNT                             CSTE0500
      IF(N.GT.0 .AND. ANC.EQ.0.0) RETURN                                CSTE0510
C                                                                       CSTE0520
      IF(RECOGC .NE. CMODEL)                                            CSTE0530
     1    CALL RERROR('SPEED  ','WRNG MODEL',RECOGC)                    CSTE0540
      MODC(1)=7HCSSPOKE                                                 CSTE0550
      MODC(2)=CID                                                       CSTE0560
      N=ANC/3                                                           CSTE0570
      IF(ANC.NE.3*N.OR.N.LE.0)                                          CSTE0580
     1     CALL RERROR('CSSPOKE','BAD NUMBER',ANC+2.0)                  CSTE0590
      N=N-2                                                             CSTE0600
      ANC=0.0                                                           CSTE0610
      TH0=PID2-LAM0                                                     CSTE0620
C                                                                       CSTE0630
      R0=EARTHR+REFH                                                    CSTE0640
C                                                                       CSTE0650
C                                                                       CSTE0660
C     CONVERT 'C' ARRAY INPUT(OVERLAYS 'B' ARRAY) TO 'B' ARRAY          CSTE0670
C                                                                       CSTE0680
      ZM1=Z0                                                            CSTE0690
      CS0=C0*C0                                                         CSTE0700
      CSM1=CS0                                                          CSTE0710
      NP1=N+1                                                           CSTE0720
      DO 10 I=1,NP1                                                     CSTE0730
         ZR=Z(I)                                                        CSTE0740
         ALC(I)=ALCOSH((ZR-Z0) / DL(I))                                 CSTE0750
         CS=BIN(I)**2                                                   CSTE0760
         B(I)=(CS-CSM1)/(ZR-ZM1)                                        CSTE0770
         ZM1=ZR                                                         CSTE0780
         CSM1=CS                                                        CSTE0790
10    CONTINUE                                                          CSTE0800
C                                                                       CSTE0810
      RETURN                                                            CSTE0820
C                                                                       CSTE0830
      ENTRY SPEED
```

```
C                                                                       CSTE0840
      IF(N.LE.0)                                                        CSTE0850
     1    CALL RERROR('CSSPOKE','BAD N VALUE',N)                       CSTE0860
C                                                                       CSTE0870
      COSTH=COS(TH0-TH)                                                 CSTE0880
      SINTH=SIN(TH0-TH)                                                 CSTE0890
      D=SQRT(R0*R0+R*R-2*R*R0*COSTH)                                    CSTE0900
      X=(R*COSTH-R0)/D                                                  CSTE0910
      ZR=ASIN(X)                                                        CSTE0920
      SUM=0.                                                            CSTE0930
      PCSZ=B(1)                                                         CSTE0940
      DO 1 I = 1, N                                                     CSTE0950
      SAV=0.5*(B(I+1)-B(I))                                             CSTE0960
      PCSZ= PCSZ+ SAV * (1.+TANH ((ZR-Z(I)) /DL(I)))                    CSTE0970
    1 SUM = SUM+DL(I) * SAV *(ALCOSH((ZR-Z(I))/DL(I))-ALC(I))           CSTE0980
      CS = CS0+SUM + 0.5*(B(1) + B(N + 1)) * (ZR-Z0)                    CSTE0990
      PDR=(R-R0*COSTH)/D                                                CSTE1000
      IF(CS.LT.0) CALL PMDSTOP                                          CSTE1010
      PDTH=(-1.)*R*R0*SINTH/D                                           CSTE1020
      PZR=(COSTH/D-(R*COSTH-R0)*PDR/(D*D))/SQRT(1.-X*X)                 CSTE1030
      PZTH=(R*SINTH/D-(R*COSTH-R0)*PDTH/(D*D))/                         CSTE1040
     1    SQRT(1.-X*X)                                                 CSTE1050
C                                                                       CSTE1060
      PCSR=PCSZ*PZR                                                     CSTE1070
      PCSTH=PCSZ*PZTH                                                   CSTE1080
C                                                                       CSTE1090
C                                                                       CSTE1100
      CALL PSPEED                                                       CSTE1110
      RETURN                                                            CSTE1120
      END                                                               CSTE1130


      SUBROUTINE CSSPOK2                                                CST20020
C     SOUND SPEED SQUARED IS MODELED AS A FUNCTION OF ANGLE 'ALPHA'     CST20030
C     ABOUT A HORIZONTAL LINE AT SPECIFIED HEIGHT AND LATITUDE. DEPENDE CST20040
C     ON 'ALPHA' IS AS A SEQUENCE OF LINEAR SEGMENTS JOINED BY HYPERBOL CST20050
C     TANGENTS.                                                         CST20060
C                                                                       CST20070
      REAL ALC(20),Z(19),BIN(19),B(19),DL(19),LAM0                      CST20080
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "B3" INSERTED HERE                                    CB3 0020
      INTEGER CMX,CNTBL,CITBL,CFRMTBL,IDSC(10)                          CB3 0040
      COMMON/B3/CMX,CNTBL(10),CITBL(10),CFRMTBL(10),CGP(512)            CB3 0050
      EQUIVALENCE (CGP,IDSC),(ANC,CGP(11))                              CB3 0060
C     COMMON DECK "CC" INSERTED HERE                                    CCC 0020
      REAL MODC                                                         CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                        CCC 0050
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
```

```
C      COMMON DECK "WW" INSERTED HERE                                          CWW 0020
       PARAMETER (NWARSZ=1000)                                                 CWW10030
       COMMON/WW/ID(10),MAXW,W(NWARSZ)                                         CWW10040
       REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                      CWW20020
       EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),          CWW20030
      1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),            CWW20040
      2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),                CWW20050
      3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),               CWW20060
      8   (RUNSUP,W(18)),(RCVRH,W(20)),                                       CWW20070
      4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))     CWW20080
      5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                           CWW20090
      6 (HMIN,W(27)),(RGMAX,W(28)),                                           CWW20100
      8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                          CWW20110
      6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),            CWW20120
      7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))             CWW20130
      9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),             CWW20140
      1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                   CWW20150
      2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                        CWW20160
       REAL MMODEL,MFORM,MID                                                  CWW30020
C                                                                             CWW30030
C      WIND          100-124                                                  CWW30040
       EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)                CWW30050
C                                                                             CWW30060
C      DELTA WIND          125-149                                            CWW30070
       EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)             CWW30080
C                                                                             CWW30090
C      SOUND SPEED 150-174                                                    CWW30100
       EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)                CWW30110
       EQUIVALENCE (W(153),REFC)                                              CWW30120
C                                                                             CWW30130
C      DELTA SOUND SPEED 175-199                                              CWW30140
       EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)             CWW30150
C                                                                             CWW30160
C      TEMPERATURE    200-224                                                 CWW30170
       EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)                CWW30180
C                                                                             CWW30190
C      DELTA TEMPERATURE    225-249                                           CWW30200
       EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)             CWW30210
C                                                                             CWW30220
C      MOLECULAR   250-274                                                    CWW30230
       EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)                CWW30240
C                                                                             CWW30250
C      RECEIVER HEIGHT   275-299                                              CWW30260
       EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)                CWW30270
C                                                                             CWW30280
C      TOPOGRAPHY   300-324                                                   CWW30290
       EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)                CWW30300
C                                                                             CWW30310
C      DELTA TOPOGRAPHY   325-349                                             CWW30320
       EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)             CWW30330
C                                                                             CWW30340
C      ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                               CWW30350
       EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)                CWW30360
C                                                                             CWW30370
C      ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                               CWW30380
```

```
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)          CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                CWW30400
C                                                                         CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                         CWW30420
C     ABSORPTION          500-524                                         CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)             CWW30440
C                                                                         CWW30450
C     DELTA ABSORPTION          525-549                                   CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)          CWW30470
C                                                                         CWW30480
C     PRESSURE          550-574                                         CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)             CWW30500
C                                                                         CWW30510
C     DELTA PRESSURE          575-599                                   CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)          CWW30530
C                                                                         CWW30540
      EQUIVALENCE (Z0,CGP(12)),(C0,CGP(32)),(DL0,CGP(52))                 CST20140
      EQUIVALENCE (Z,CGP(13)),(BIN,CGP(33)),(DL,CGP(53))                  CST20150
      EQUIVALENCE (REFH,W(154)), (LAM0,W(155))                            CST20160
C                                                                         CST20170
      INTEGER CPX     ,CQTBL(10) ,CLTBL(10),CIRMTBL(10)                   CST20180
      DATA RECOGC,N/2.0,0/                                                CST20190
C                                                                         CST20200
      DATA PFST,PFSTH,PFSPH/3*0.0/                                        CST20210
      DATA AQC/0.0/                                                       CST20220
      DATA CPX/2/                                                         CST20230
      DATA CQTBL/1,11,72,7*0/                                             CST20240
      DATA CLTBL/1,20,8*0/                                                CST20250
      DATA CIRMTBL/1,2,8*0/                                               CST20260
C                                                                         CST20270
C                                                                         CST20280
      ENTRY SETSPD                                                        CST20290
C                                                                         CST20300
      PCST=PFST                                                           CST20310
      PCSTH=PFSTH                                                         CST20320
      PCSPH=PFSPH                                                         CST20330
      ANC=AQC                                                             CST20340
      CMX=CPX                                                             CST20350
      CALL IMOVE(CNTBL,CQTBL,10)                                          CST20360
      CALL IMOVE(CITBL,CLTBL,10)                                          CST20370
      CALL IMOVE(CFRMTBL,CIRMTBL,10)                                      CST20380
      CALL SETPSP                                                         CST20390
C                                                                         CST20400
      RETURN                                                              CST20410
C                                                                         CST20420
      ENTRY ISPEED                                                        CST20430
C                                                                         CST20440
      IF(REFC.GT.0) CREF=REFC                                             CST20450
C                                                                         CST20460
      CALL IPSPEED                                                        CST20470
C                                                                         CST20480
C     IF HAD PREVIOUS CALL BUT NOTHING THIS TIME, EXIT NOW                CST20490
C     RETAINING PREVIOUS TABULAR DATA COUNT                               CST20500
      IF(N.GT.0 .AND. ANC.EQ.0.0) RETURN                                  CST20510
C                                                                         CST20520
```

```
      IF(RECOGC .NE. CMODEL)                                            CST20530
     1    CALL RERROR('SPEED   ','WRNG MODEL',RECOGC)                   CST20540
      MODC(1)=7HCSSPOK2                                                 CST20550
      MODC(2)=CID                                                       CST20560
      N=ANC/3                                                           CST20570
      IF(ANC.NE.3*N.OR.N.LE.0)                                          CST20580
     1     CALL RERROR('CSSPOKE','BAD NUMBER',ANC+2.0)                  CST20590
      N=N-2                                                             CST20600
      ANC=0.0                                                           CST20610
      TH0=PID2-LAM0                                                     CST20620
C                                                                       CST20630
      R0=EARTHR+REFH                                                    CST20640
C                                                                       CST20650
C                                                                       CST20660
C     CONVERT 'C' ARRAY INPUT(OVERLAYS 'B' ARRAY) TO 'B' ARRAY          CST20670
C                                                                       CST20680
      ZM1=Z0                                                            CST20690
      CS0=C0*C0                                                         CST20700
      CSM1=CS0                                                          CST20710
      NP1=N+1                                                           CST20720
      DO 10 I=1,NP1                                                     CST20730
         ZR=Z(I)                                                        CST20740
         ALC(I)=ALCOSH((ZR-Z0) / DL(I))                                 CST20750
         CS=BIN(I)**2                                                   CST20760
         B(I)=(CS-CSM1)/(ZR-ZM1)                                        CST20770
         ZM1=ZR                                                         CST20780
         CSM1=CS                                                        CST20790
10                                                                      CST20800
C                                                                       CST20810
      RETURN                                                            CST20820
C                                                                       CST20830
      ENTRY SPEED                                                       CST20840
C                                                                       CST20850
      IF(N.LE.0)                                                        CST20860
     1    CALL RERROR('CSSPOK2','BAD N VALUE',N)                        CST20870
C                                                                       CST20880
      X=(R-R0)/(EARTHR*(TH0-TH))                                        CST20890
      ZR=ATAN2(R-R0,EARTHR*(TH0-TH))                                    CST20900
      SUM=0.                                                            CST20910
      PCSZ=B(1)                                                         CST20920
      DO 1 I = 1, N                                                     CST20930
      SAV=0.5*(B(I+1)-B(I))                                             CST20940
      PCSZ= PCSZ+ SAV * (1.+TANH ((ZR-Z(I)) /DL(I)))                    CST20950
    1 SUM = SUM+DL(I) * SAV *(ALCOSH((ZR-Z(I))/DL(I))-ALC(I))           CST20960
      CS = CS0+SUM + 0.5*(B(1) + B(N + 1)) * (ZR-Z0)                    CST20970
      IF(CS.LT.0) CALL PMDSTOP                                          CST20980
C                                                                       CST20990
      F=X/(1.0+X*X)                                                     CST21000
      PZR=F/(R-R0)                                                      CST21010
      PZTH=-F/(TH-TH0)                                                  CST21020
C                                                                       CST21030
      PCSR=PCSZ*PZR                                                     CST21040
      PCSTH=PCSZ*PZTH                                                   CST21050
C                                                                       CST21060
C                                                                       CST21070
      CALL PSPEED
```

```
      RETURN                                                            CST21080
      END                                                               CST21090



      SUBROUTINE CSMUNK1                                                CSS10020
C SPEED MODEL BASED ON THE 'CANONICAL' MODEL FOR A SOUND CHANNEL        CSS10030
C DERIVED BY MUNK(J.ACOUSTICAL SOC. AM. 55,220-226).                    CSS10040
C THE FOUR PARAMETERS OF THE MODEL ARE ALLOWED TO VARY LINEARLY IN      CSS10050
C LONGITUDE, PH. THE USER SPECIFIES VALUES FOR THE FOUR PARAMETERS AT   CSS10060
C TWO LONGITUDES, PH1 AND PH2, AND THE PROGRAM INTERPOLATES LINEARLY    CSS10070
C TO GET THE VALUES AT OTHER LONGITUDES.                                CSS10080
C                                                                       CSS10090
C     C = CA(1+EP(ETA+EXP(-ETA)-1))                                     CSS10100
C                                                                       CSS10110
C        WHERE                                                          CSS10120
C                                                                       CSS10130
C           ETA = 2(Z-ZA)/H                                             CSS10140
C                                                                       CSS10150
C           Z = R-EARTHR                                                CSS10160
C                                                                       CSS10170
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "CC" INSERTED HERE                                    CCC 0020
      REAL MODC                                                         CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                        CCC 0050
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
```

```
C     DELTA WIND         125-149                                          CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)          CWW30080
C                                                                         CWW30090
C     SOUND SPEED 150-174                                                 CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)             CWW30110
      EQUIVALENCE (W(153),REFC)                                           CWW30120
C                                                                         CWW30130
C     DELTA SOUND SPEED 175-199                                           CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)          CWW30150
C                                                                         CWW30160
C     TEMPERATURE    200-224                                              CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)             CWW30180
C                                                                         CWW30190
C     DELTA TEMPERATURE    225-249                                        CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)          CWW30210
C                                                                         CWW30220
C     MOLECULAR   250-274                                                 CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)             CWW30240
C                                                                         CWW30250
C     RECEIVER HEIGHT   275-299                                           CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)             CWW30270
C                                                                         CWW30280
C     TOPOGRAPHY   300-324                                                CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)             CWW30300
C                                                                         CWW30310
C     DELTA TOPOGRAPHY   325-349                                          CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)          CWW30330
C                                                                         CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                            CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)             CWW30360
C                                                                         CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                            CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)          CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                CWW30400
C                                                                         CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                         CWW30420
C     ABSORPTION         500-524                                          CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)             CWW30440
C                                                                         CWW30450
C     DELTA ABSORPTION         525-549                                    CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)          CWW30470
C                                                                         CWW30480
C     PRESSURE         550-574                                            CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)             CWW30500
C                                                                         CWW30510
C     DELTA PRESSURE         575-599                                      CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)          CWW30530
C                                                                         CWW30540
C                                                                         CSS10220
      EQUIVALENCE (W(154),PH1), (W(155),CA1) ,(W(156),ZA1)                CSS10230
      EQUIVALENCE (W(157),H1) ,(W(158),EP1)                               CSS10240
C                                                                         CSS10250
      EQUIVALENCE (W(159),PH2), (W(160),CA2) ,(W(161),ZA2)                CSS10260
      EQUIVALENCE (W(162),H2) ,(W(163),EP2)                               CSS10270
C                                                                         CSS10280
```

```
C       COMMON DECK "B3" INSERTED HERE                                  CB3 0020
        INTEGER CMX,CNTBL,CITBL,CFRMTBL,IDSC(10)                        CB3 0040
        COMMON/B3/CMX,CNTBL(10),CITBL(10),CFRMTBL(10),CGP(512)          CB3 0050
        EQUIVALENCE (CGP,IDSC),(ANC,CGP(11))                            CB3 0060
        INTEGER CPX     ,CQTBL(10),CLTBL(10),CIRMTBL(10)                CSS10300
        DATA RECOGC/5.0/                                                CSS10310
C                                                                       CSS10320
        DATA CPX/1/                                                     CSS10330
        DATA CQTBL/1,11,8*0/                                            CSS10340
        DATA CLTBL/1,9*0/                                               CSS10350
        DATA CIRMTBL/1,9*0/                                             CSS10360
C                                                                       CSS10370
        ENTRY SETSPD                                                    CSS10380
C                                                                       CSS10390
        CMX=CPX                                                         CSS10400
        CALL IMOVE(CNTBL,CQTBL,10)                                      CSS10410
        CALL IMOVE(CITBL,CLTBL,10)                                      CSS10420
        CALL IMOVE(CFRMTBL,CIRMTBL,10)                                  CSS10430
        CALL SETPSP                                                     CSS10440
C                                                                       CSS10450
        RETURN                                                          CSS10460
C                                                                       CSS10470
        ENTRY ISPEED                                                    CSS10480
C                                                                       CSS10490
        IF(RECOGC .NE. CMODEL)                                          CSS10500
     1    CALL RERROR('SPEED  ','WRNG MODEL',RECOGC)                    CSS10510
        MODC(1)=7HCSMUNK1                                               CSS10520
        MODC(2)=CID                                                     CSS10530
C                                                                       CSS10540
        DPH=PH2-PH1                                                     CSS10550
        DCA=CA2-CA1                                                     CSS10560
        DZA=ZA2-ZA1                                                     CSS10570
        DH=H2-H1                                                        CSS10580
        DEP=EP2-EP1                                                     CSS10590
C    COMPUTE PH DERIVATIVES OF CA,ZA,H,EP                               CSS10600
        PHPH=DH/DPH                                                     CSS10610
        PEPPH=DEP/DPH                                                   CSS10620
        PCAPH=DCA/DPH                                                   CSS10630
        PZAPH=DZA/DPH                                                   CSS10640
C                                                                       CSS10650
        IF(REFC.GT.0.) CREF=REFC                                        CSS10660
        CALL IPSPEED                                                    CSS10670
        RETURN                                                          CSS10680
C                                                                       CSS10690
        ENTRY SPEED                                                     CSS10700
C                                                                       CSS10710
C    INTERPOLATE FOR CA,ZA,H,EP                                         CSS10720
        FRACT=(PH-PH1)/DPH                                              CSS10730
        CA=CA1+FRACT*DCA                                                CSS10740
        ZA=ZA1+FRACT*DZA                                                CSS10750
        H=H1+FRACT*DH                                                   CSS10760
        EP=EP1+FRACT*DEP                                                CSS10770
C    COMPUTE SOUND SPEED,C                                              CSS10780
        ETA=-2.*(R-EARTHR-ZA)/H                                         CSS10790
        EXETA1=EXP(-ETA)-1.                                             CSS10800
```

```
C                                                                       CSS10810
      PRODCT=CA*(ETA+EXETA1)                                            CSS10820
      C=CA+EP*PRODCT                                                    CSS10830
      CS=C*C                                                            CSS10840
C     COMPUTE DERIVATIVES OF C                                          CSS10850
      PCR=2.*CA*EP*EXETA1/H                                             CSS10860
      PCEP=PRODCT                                                       CSS10870
      PCCA=1.+(EP/CA)*PRODCT                                            CSS10880
      PCZA=-PCR                                                         CSS10890
      PCH=ETA/2.*PCR                                                    CSS10900
C     COMPUTE DERIVATIVES OF CS                                         CSS10910
      C2=C*2.                                                           CSS10920
      PCSTH=0.                                                          CSS10930
      PCST=0.                                                           CSS10940
      PCSR=C2*PCR                                                       CSS10950
C     CHAIN RULE TO GET PCS/PPH.                                        CSS10960
      PCSPH=C2*(PCH*PHPH+PCEP*PEPPH+PCZA*PZAPH+PCCA*PCAPH)              CSS10970
C                                                                       CSS10980
      CALL PSPEED                                                       CSS10990
      RETURN                                                            CSS11000
      END                                                               CSS11010
```

```
      SUBROUTINE CSMUNK2                                                CSS20020
C SPEED MODEL BASED ON THE 'CANONICAL' MODEL FOR A SOUND CHANNEL        CSS20030
C DERIVED BY MUNK(J.ACOUSTICAL SOC. AM. 55,220-226).                    CSS20040
C THE FOUR PARAMETERS OF THE MODEL ARE ALLOWED TO VARY LINEARLY IN      CSS20050
C LONGITUDE, PH. THE USER SPECIFIES VALUES FOR THE FOUR PARAMETERS AT   CSS20060
C TWO LONGITUDES, PH1 AND PH2, AND THE PROGRAM INTERPOLATES LINEARLY    CSS20070
C TO GET THE VALUES AT OTHER LONGITUDES.                                CSS20080
C                                                                       CSS20090
C     C = CA(1+EP(ETA+EXP(-ETA)-1))                                     CSS20100
C                                                                       CSS20110
C        WHERE                                                          CSS20120
C                                                                       CSS20130
C           ETA = 2(Z-ZA)/H                                             CSS20140
C                                                                       CSS20150
C           Z = R-EARTHR                                                CSS20160
C                                                                       CSS20170
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "CC" INSERTED HERE                                    CCC 0020
      REAL MODC                                                         CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                        CCC 0050
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
```

```
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),     CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),       CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),           CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),          CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                   CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                       CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                       CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                      CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),        CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))         CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),         CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))               CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                    CWW20160
      REAL MMODEL,MFORM,MID                                              CWW30020
C                                                                        CWW30030
C     WIND         100-124                                             CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)            CWW30050
C                                                                        CWW30060
C     DELTA WIND         125-149                                         CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)         CWW30080
C                                                                        CWW30090
C     SOUND SPEED 150-174                                                CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)            CWW30110
      EQUIVALENCE (W(153),REFC)                                          CWW30120
C                                                                        CWW30130
C     DELTA SOUND SPEED 175-199                                          CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)         CWW30150
C                                                                        CWW30160
C     TEMPERATURE    200-224                                             CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)            CWW30180
C                                                                        CWW30190
C     DELTA TEMPERATURE    225-249                                       CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)         CWW30210
C                                                                        CWW30220
C     MOLECULAR   250-274                                                CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)            CWW30240
C                                                                        CWW30250
C     RECEIVER HEIGHT   275-299                                          CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)            CWW30270
C                                                                        CWW30280
C     TOPOGRAPHY   300-324                                               CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)            CWW30300
C                                                                        CWW30310
C     DELTA TOPOGRAPHY   325-349                                         CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)         CWW30330
C                                                                        CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                           CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)            CWW30360
C                                                                        CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                           CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)         CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                               CWW30400
C                                                                        CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                        CWW30420
```

```
C     ABSORPTION          500-524                                          CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)              CWW30440
C                                                                          CWW30450
C     DELTA ABSORPTION          525-549                                    CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)           CWW30470
C                                                                          CWW30480
C     PRESSURE          550-574                                          CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)              CWW30500
C                                                                          CWW30510
C     DELTA PRESSURE          575-599                                    CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)           CWW30530
C                                                                          CWW30540
C                                                                          CSS20220
      EQUIVALENCE (W(154),PH1), (W(155),CA1) ,(W(156),ZA1)                 CSS20230
      EQUIVALENCE (W(157),H1) ,(W(158),EP1)                                CSS20240
C                                                                          CSS20250
      EQUIVALENCE (W(159),PH2), (W(160),CA2) ,(W(161),ZA2)                 CSS20260
      EQUIVALENCE (W(162),H2) ,(W(163),EP2)                                CSS20270
C                                                                          CSS20280
C     COMMON DECK "B3" INSERTED HERE                                       CB3 0020
      INTEGER CMX,CNTBL,CITBL,CFRMTBL,IDSC(10)                             CB3 0040
      COMMON/B3/CMX,CNTBL(10),CITBL(10),CFRMTBL(10),CGP(512)               CB3 0050
      EQUIVALENCE (CGP,IDSC),(ANC,CGP(11))                                 CB3 0060
      INTEGER CPX    ,CQTBL(10),CLTBL(10),CIRMTBL(10)                      CSS20300
      DATA RECOGC/6.0/                                                     CSS20310
C                                                                          CSS20320
      DATA CPX/1/                                                          CSS20330
      DATA CQTBL/1,11,8*0/                                                 CSS20340
      DATA CLTBL/1,9*0/                                                    CSS20350
      DATA CIRMTBL/1,9*0/                                                  CSS20360
C                                                                          CSS20370
C                                                                          CSS20380
      ENTRY SETSPD                                                         CSS20390
C                                                                          CSS20400
      CMX=CPX                                                              CSS20410
      CALL IMOVE(CNTBL,CQTBL,10)                                           CSS20420
      CALL IMOVE(CITBL,CLTBL,10)                                           CSS20430
      CALL IMOVE(CFRMTBL,CIRMTBL,10)                                       CSS20440
      CALL SETPSP                                                          CSS20450
C                                                                          CSS20460
      RETURN                                                               CSS20470
C                                                                          CSS20480
      ENTRY ISPEED                                                         CSS20490
C                                                                          CSS20500
      IF(RECOGC .NE. CMODEL)                                               CSS20510
     1    CALL RERROR('SPEED  ','WRNG MODEL',RECOGC)                       CSS20520
      MODC(1)=7HCSMUNK2                                                    CSS20530
      MODC(2)=CID                                                          CSS20540
      IF(REFC.GT.0) CREF=REFC                                              CSS20550
C                                                                          CSS20560
      DPH=PH2-PH1                                                          CSS20570
C                                                                          CSS20580
      CALL IPSPEED                                                         CSS20590
      RETURN                                                               CSS20600
C                                                                          CSS20610
```

```
      ENTRY SPEED                                                       CSS20620
C                                                                       CSS20630
C     COMPUTE SOUND SPEED,C1, AT (PH1,Z)                                CSS20640
C                                                                       CSS20650
      ETA1=-2.*(R-EARTHR-ZA1)/H1                                        CSS20660
      EXETA1=EXP(-ETA1)-1.                                              CSS20670
      PROD1=CA1*(ETA1+EXETA1)                                           CSS20680
      C1=CA1+EP1*PROD1                                                  CSS20690
C                                                                       CSS20700
C     COMPUTE SOUND SPEED C2 AT (PH2,Z)                                 CSS20710
C                                                                       CSS20720
      ETA2=-2.*(R-EARTHR-ZA2)/H2                                        CSS20730
      EXETA2=EXP(-ETA2)-1.                                              CSS20740
      PROD2=CA2*(ETA2+EXETA2)                                           CSS20750
      C2=CA2+EP2*PROD2                                                  CSS20760
C                                                                       CSS20770
C     COMPUTE VERTICAL GRADIENTS AT PH1,PH2                             CSS20780
C                                                                       CSS20790
      PCR1=2.*CA1*EP1/H1*EXETA1                                         CSS20800
      PCR2=2.*CA2*EP2/H2*EXETA2                                         CSS20810
C                                                                       CSS20820
C     DIFFERENCES IN C AND ITS VERTICAL GRADIENT                        CSS20830
C                                                                       CSS20840
      DPCR=PCR2-PCR1                                                    CSS20850
      DC = C2-C1                                                        CSS20860
C                                                                       CSS20870
C     INTERPOLATE FOR C AT (PH,Z)                                       CSS20880
C                                                                       CSS20890
      FRACT=(PH-PH1)/DPH                                                CSS20900
      C=C1+FRACT*DC                                                     CSS20910
      CS=C*C                                                            CSS20920
C                                                                       CSS20930
C     DERIVATIVES OF C AND CS                                           CSS20940
C                                                                       CSS20950
      PCPH=DC/DPH                                                       CSS20960
      PCR=PCR1+FRACT*DPCR                                               CSS20970
      PCSR=2.*C*PCR                                                     CSS20980
      PCSPH=2.*C*PCPH                                                   CSS20990
      PCSTH=0.                                                          CSS21000
      PCST=0.                                                           CSS21010
C                                                                       CSS21020
C                                                                       CSS21030
      CALL PSPEED                                                       CSS21040
      RETURN                                                            CSS21050
      END                                                               CSS21060
```

```
      SUBROUTINE CTABLE                                                 CTUE0020
C     TABULAR TEMPERATURE PROFILE THAT MAKES A CUBIC INTERPOLATION      CTUE0030
C     BETWEEN POINTS TO INSURE A CONTINUOUS TERMPERATURE GRADIENT       CTUE0040
      DIMENSION HPC(250),FN2C(250),ALPHA(250),TTBETA(250),GAMM(250),    CTUE0050
     1 DELTA(250),SLOPE(250),MAT(4,5)                                   CTUE0060
```

```
      REAL MAT                                                                  CTUE0070
C     COMMON DECK "CONST" INSERTED HERE                                         CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                             CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                                 CCON0050
C     COMMON DECK "RKAM" INSERTED HERE                                          RKAM0020
      REAL KR,KTH,KPH                                                           RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)               RKAM0050
C     COMMON DECK "CC" INSERTED HERE                                            CCC 0020
      REAL MODC                                                                 CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                                CCC 0050
C     COMMON DECK "WW" INSERTED HERE                                            CWW 0020
      PARAMETER (NWARSZ=1000)                                                   CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                           CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                        CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),            CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),              CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),                  CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),                 CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                         CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))        CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                              CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                              CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                             CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),               CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))                CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),                CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                      CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                           CWW20160
      REAL MMODEL,MFORM,MID                                                     CWW30020
C                                                                               CWW30030
C     WIND         100-124                                                    CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)                   CWW30050
C                                                                               CWW30060
C     DELTA WIND         125-149                                                CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)                CWW30080
C                                                                               CWW30090
C     SOUND SPEED 150-174                                                       CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)                   CWW30110
      EQUIVALENCE (W(153),REFC)                                                 CWW30120
C                                                                               CWW30130
C     DELTA SOUND SPEED 175-199                                                 CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)                CWW30150
C                                                                               CWW30160
C     TEMPERATURE    200-224                                                    CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)                   CWW30180
C                                                                               CWW30190
C     DELTA TEMPERATURE    225-249                                              CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)                CWW30210
C                                                                               CWW30220
C     MOLECULAR   250-274                                                       CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)                   CWW30240
C                                                                               CWW30250
C     RECEIVER HEIGHT   275-299                                                 CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)                   CWW30270
C                                                                               CWW30280
```

```
C     TOPOGRAPHY   300-324                                                  CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)              CWW30300
C                                                                          CWW30310
C     DELTA TOPOGRAPHY   325-349                                           CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)           CWW30330
C                                                                          CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                             CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)              CWW30360
C                                                                          CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                             CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)           CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                 CWW30400
C                                                                          CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                          CWW30420
C     ABSORPTION         500-524                                           CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)              CWW30440
C                                                                          CWW30450
C     DELTA ABSORPTION         525-549                                     CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)           CWW30470
C                                                                          CWW30480
C     PRESSURE         550-574                                             CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)              CWW30500
C                                                                          CWW30510
C     DELTA PRESSURE         575-599                                       CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)           CWW30530
C                                                                          CWW30540
C     COMMON DECK "B3" INSERTED HERE                                       CB3 0020
      INTEGER CMX,CNTBL,CITBL,CFRMTBL,IDSC(10)                             CB3 0040
      COMMON/B3/CMX,CNTBL(10),CITBL(10),CFRMTBL(10),CGP(512)               CB3 0050
      EQUIVALENCE (CGP,IDSC),(ANC,CGP(11))                                 CB3 0060
C                                                                          CTUE0130
      EQUIVALENCE (HPC,CGP(12)),(FN2C,CGP(262))                            CTUE0140
C                                                                          CTUE0150
      DATA RECOGC,NOC/8.0,0/                                               CTUE0160
      DATA ANC/0.0/                                                        CTUE0170
      DATA CMX/2/                                                          CTUE0180
      DATA CNTBL/1,11,512,7*0/                                             CTUE0190
      DATA CITBL/1,250,8*0/                                                CTUE0200
      DATA CFRMTBL/1,2,8*0/                                                CTUE0210
C                                                                          CTUE0220
      ENTRY ISPEED                                                         CTUE0230
C                                                                          CTUE0240
      CALL IPSPEED                                                         CTUE0250
C                                                                          CTUE0260
C     IF HAD PREVIOUS CALL BUT NOTHING THIS TIME, EXIT NOW                 CTUE0270
C     RETAINING PREVIOUS TABULAR DATA COUNT                                CTUE0280
      IF(NOC.GT.0 .AND. ANC.LE.0.0) RETURN                                 CTUE0290
C                                                                          CTUE0300
C         PRINT *,(CGP(I),I=11,30)                                         CTUE0310
      IF(RECOGC.NE.CMODEL)                                                 CTUE0320
     1    CALL RERROR('SPEED  ','WRNG MODEL',RECOGC)                       CTUE0330
C                                                                          CTUE0340
        ANC=AINT(ANC)                                                      CTUE0350
      NOC=ANC/2                                                            CTUE0360
      IF(ANC.NE.2*NOC .OR. NOC.LE.1)                                       CTUE0370
```

```
     1     CALL RERROR('CTABLE','BAD NUMBER',ANC)                          CTUE0380
      ANC=0.0                                                              CTUE0390
C                                                                          CTUE0400
      MODC(1)=6HCTABLE                                                     CTUE0410
      MODC(2)=CID                                                          CTUE0420
C                                                                          CTUE0430
C           PRINT *,(HPC(I),I=1,10),(FN2C(I),I=1,10)                       CTUE0440
C                                                                          CTUE0450
      SLOPE(1)=(FN2C(2)-FN2C(1))/(HPC(2)-HPC(1))                           CTUE0460
      SLOPE(NOC)=0.                                                        CTUE0470
      NMAX=1                                                               CTUE0480
      DO 6 I=2,NOC                                                         CTUE0490
      IF (FN2C(I).GT.FN2C(NMAX)) NMAX=I                                    CTUE0500
      IF (I.EQ.NOC) GO TO 4                                                CTUE0510
      DO 3 J=1,3                                                           CTUE0520
      M=I+J-2                                                              CTUE0530
      MAT(J,1)=1.                                                          CTUE0540
      MAT(J,2)=HPC(M)                                                      CTUE0550
      MAT(J,3)=HPC(M)**2                                                   CTUE0560
    3 MAT(J,4)=FN2C(M)                                                     CTUE0570
      CALL GAUSEL (MAT,4,3,4,NRANK)                                        CTUE0580
      IF (NRANK.LT.3) GO TO 60                                             CTUE0590
      SLOPE(I)=MAT(2,4)+2.*MAT(3,4)*HPC(I)                                 CTUE0600
    4 DO 5 J=1,2                                                           CTUE0610
      M=I+J-2                                                              CTUE0620
      MAT(J,1)=1.                                                          CTUE0630
      MAT(J,2)=HPC(M)                                                      CTUE0640
      MAT(J,3)=HPC(M)**2                                                   CTUE0650
      MAT(J,4)=HPC(M)**3                                                   CTUE0660
      MAT(J,5)=FN2C(M)                                                     CTUE0670
      L=J+2                                                                CTUE0680
      MAT(L,1)=0.                                                          CTUE0690
      MAT(L,2)=1.                                                          CTUE0700
      MAT(L,3)=2.*HPC(M)                                                   CTUE0710
      MAT(L,4)=3.*HPC(M)**2                                                CTUE0720
    5 MAT(L,5)=SLOPE(M)                                                    CTUE0730
      CALL GAUSEL (MAT,4,4,5,NRANK)                                        CTUE0740
      IF (NRANK.LT.4) GO TO 60                                             CTUE0750
      ALPHA(I)=MAT(1,5)                                                    CTUE0760
      TTBETA(I)=MAT(2,5)                                                   CTUE0770
      GAMM(I)=MAT(3,5)                                                     CTUE0780
    6 DELTA(I)=MAT(4,5)                                                    CTUE0790
C     HMAX=HPC(NMAX)                                                       CTUE0800
      NH=2                                                                 CTUE0810
C                                                                          CTUE0820
      RETURN                                                               CTUE0830
C                                                                          CTUE0840
   60 PRINT 6000, I,HPC(I)                                                 CTUE0850
 6000 FORMAT(' THE',I4,'TH POINT IN THE TEMPERATURE PROFILE HAS'           CTUE0860
     1,' THE HEIGHT',F8.2,' KM, WHICH IS THE SAME AS ANOTHER POINT.')      CTUE0870
      CALL EXIT                                                            CTUE0880
C                                                                          CTUE0890
      ENTRY SPEED                                                          CTUE0900
C                                                                          CTUE0910
C          PRINT *,'ENTER CTABLE'                                          CTUE0920
```

```
C                                                                       CTUE0930
      IF(NOC.LE.0)                                                      CTUE0940
     1    CALL RERROR('CTABLE','BAD N VALUE',FLOAT(NOC))               CTUE0950
C                                                                       CTUE0960
      H=R-EARTHR                                                        CTUE0970
      PCST=0.0                                                          CTUE0980
      PCSR=0.0                                                          CTUE0990
      PCSTH=0.0                                                         CTUE1000
      PCSPH=0.0                                                         CTUE1010
      IF (H.GE.HPC(1)) GO TO 12                                         CTUE1020
   11 NH=2                                                              CTUE1030
      T=FN2C(1)+SLOPE(1)*(H-HPC(1))                                     CTUE1040
        PCSR=2.*T*SLOPE(1)                                              CTUE1050
        CS=T*T                                                          CTUE1060
        CALL PSPEED                                                     CTUE1070
C         PRINT *,'LEAVE CTABLE1 ',R,C,PCSR                             CTUE1080
      RETURN                                                            CTUE1090
   12 IF (H.GE.HPC(NOC)) GO TO 18                                       CTUE1100
      NSTEP=1                                                           CTUE1110
      IF (H.LT.HPC(NH-1)) NSTEP=-1                                      CTUE1120
   15 IF (HPC(NH-1).LE.H.AND.H.LT.HPC(NH)) GO TO 16                     CTUE1130
      NH=NH+NSTEP                                                       CTUE1140
      GO TO 15                                                          CTUE1150
   16 T=(ALPHA(NH)+H*(TTBETA(NH)+H*(GAMM(NH)+H*DELTA(NH))))             CTUE1160
      PCSR=(TTBETA(NH)+H*(2.*GAMM(NH)+H*3.*DELTA(NH)))                  CTUE1170
C           PRINT *,NH,ALPHA(NH),TTBETA(NH),HPC(NH),FN2C(NH)            CTUE1180
        PCSR=2.*T*PCSR                                                  CTUE1190
        CS=T*T                                                          CTUE1200
        CALL PSPEED                                                     CTUE1210
C         PRINT *,'LEAVE CTABLE2 ',R,C,PCSR                             CTUE1220
      RETURN                                                            CTUE1230
   18 T=FN2C(NOC)                                                       CTUE1240
C                                                                       CTUE1250
      CALL PSPEED                                                       CTUE1260
        PCSR=2.*T*PCSR                                                  CTUE1270
        CS=T*T                                                          CTUE1280
C         PRINT *,'LEAVE CTABLE3 ',R,C,PCSR                             CTUE1290
      RETURN                                                            CTUE1300
        END                                                             CTUE1310
```

```
      SUBROUTINE NPSPEED                                                NPPD0020
C     DO-NOTHING SOUND SPEED PERTURBATION MODEL                         NPPD0030
C     COMMON DECK "CC" INSERTED HERE                                    CCC 0020
      REAL MODC                                                         CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                        CCC 0050
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
```

```
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),                CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),               CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                        CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))      CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                            CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                            CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                           CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),             CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))              CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),              CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                    CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                         CWW20160
      REAL MMODEL,MFORM,MID                                                   CWW30020
C                                                                             CWW30030
C     WIND         100-124                                                    CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)                 CWW30050
C                                                                             CWW30060
C     DELTA WIND         125-149                                            CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)              CWW30080
C                                                                             CWW30090
C     SOUND SPEED 150-174                                                     CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)                 CWW30110
      EQUIVALENCE (W(153),REFC)                                               CWW30120
C                                                                             CWW30130
C     DELTA SOUND SPEED 175-199                                               CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)              CWW30150
C                                                                             CWW30160
C     TEMPERATURE    200-224                                                  CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)                 CWW30180
C                                                                             CWW30190
C     DELTA TEMPERATURE    225-249                                            CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)              CWW30210
C                                                                             CWW30220
C     MOLECULAR   250-274                                                     CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)                 CWW30240
C                                                                             CWW30250
C     RECEIVER HEIGHT   275-299                                               CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)                 CWW30270
C                                                                             CWW30280
C     TOPOGRAPHY   300-324                                                    CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)                 CWW30300
C                                                                             CWW30310
C     DELTA TOPOGRAPHY   325-349                                              CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)              CWW30330
C                                                                             CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                                CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)                 CWW30360
C                                                                             CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                                CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)              CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                    CWW30400
C                                                                             CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                             CWW30420
C     ABSORPTION         500-524                                              CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)                 CWW30440
```

```
C                                                                       CWW30450
C     DELTA ABSORPTION         525-549                                  CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)        CWW30470
C                                                                       CWW30480
C     PRESSURE         550-574                                          CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)           CWW30500
C                                                                       CWW30510
C     DELTA PRESSURE         575-599                                  CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)        CWW30530
C                                                                       CWW30540
C                                                                       NPPD0060
C     COMMON DECK "B2" INSERTED HERE                                    CB2 0020
      INTEGER DUMX,DUNTBL,DUITBL,DUFRMTB,IDSDU(10)                      CB2 0040
      COMMON/B2/DUMX,DUNTBL(10),DUITBL(10),DUFRMTB(10),DUGP(10)         CB2 0050
      EQUIVALENCE (DUGP,IDSDU)                                          CB2 0060
C                                                                       NPPD0080
      INTEGER DXMX    ,DXNTBL(10),DXITBL(10),DXFRMTB(10)                NPPD0090
C                                                                       NPPD0100
      DATA DXMX/1/                                                      NPPD0110
      DATA DXNTBL/1,11,8*0/                                             NPPD0120
      DATA DXITBL/1,9*0/                                                NPPD0130
      DATA DXFRMTB/1,9*0/                                               NPPD0140
C                                                                       NPPD0150
      DATA RECOGDC/0.0/                                                 NPPD0160
C                                                                       NPPD0170
C                                                                       NPPD0180
      ENTRY SETPSP                                                      NPPD0190
C                                                                       NPPD0200
      DUMX=DXMX                                                         NPPD0210
      CALL IMOVE(DUNTBL,DXNTBL,10)                                      NPPD0220
      CALL IMOVE(DUITBL,DXITBL,10)                                      NPPD0230
      CALL IMOVE(DUFRMTB,DXFRMTB,10)                                    NPPD0240
C                                                                       NPPD0250
      RETURN                                                            NPPD0260
C                                                                       NPPD0270
      ENTRY IPSPEED                                                     NPPD0280
      IF(RECOGDC .NE. DCMODEL)                                          NPPD0290
     1    CALL RERROR('DSPEED ','WRNG MODEL',RECOGDC)                   NPPD0300
      MODC(3)=7HNPSPEED                                                 NPPD0310
      MODC(4)=DCID                                                      NPPD0320
      RETURN                                                            NPPD0330
C                                                                       NPPD0340
      ENTRY PSPEED                                                      NPPD0350
      RETURN                                                            NPPD0360
      END                                                               NPPD0370
```

```
      SUBROUTINE CBLOB2                                                 CBF20020
C     SOUND SPEED PERTURBATION MODEL                                    CBF20030
C     MULTIPLICATIVE PERTURBATION WITH EXPONENTIALLY DECAYING           CBF20040
C     EFFECT IN ALL THREE DIRECTIONS. GIVE LATITUDE                     CBF20050
C     INSTEAD OF CO-LATITUDE.                                           CBF20060
```

```
C       COMMON DECK "CONST" INSERTED HERE                                      CCON0020
        COMMON/PCONST/CREF,RGAS,GAMMA                                          CCON0040
        COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                              CCON0050
C       COMMON DECK "RKAM" INSERTED HERE                                       RKAM0020
        REAL KR,KTH,KPH                                                        RKAM0040
        COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)            RKAM0050
C       COMMON DECK "B4" INSERTED HERE                                         CB4 0020
        INTEGER DCMX,DCNTBL,DCITBL,DCFRMTB,IDSDC(10)                           CB4 0040
        COMMON/B4/DCMX,DCNTBL(10),DCITBL(10),DCFRMTB(10),DCGP(10)              CB4 0050
        EQUIVALENCE (DCGP,IDSDC)                                               CB4 0060
C                                                                              CBF20100
C       COMMON DECK "CC" INSERTED HERE                                         CCC 0020
        REAL MODC                                                              CCC 0040
        COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                             CCC 0050
C       COMMON DECK "WW" INSERTED HERE                                         CWW 0020
        PARAMETER (NWARSZ=1000)                                                CWW10030
        COMMON/WW/ID(10),MAXW,W(NWARSZ)                                        CWW10040
        REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                     CWW20020
        EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),         CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),             CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),                 CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),                CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                        CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))       CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                             CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                             CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                            CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),              CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))               CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),               CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                     CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                          CWW20160
        REAL MMODEL,MFORM,MID                                                  CWW30020
C                                                                              CWW30030
C       WIND          100-124                                                  CWW30040
        EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)                CWW30050
C                                                                              CWW30060
C       DELTA WIND          125-149                                            CWW30070
        EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)             CWW30080
C                                                                              CWW30090
C       SOUND SPEED 150-174                                                    CWW30100
        EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)                CWW30110
        EQUIVALENCE (W(153),REFC)                                              CWW30120
C                                                                              CWW30130
C       DELTA SOUND SPEED 175-199                                              CWW30140
        EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)             CWW30150
C                                                                              CWW30160
C       TEMPERATURE     200-224                                                CWW30170
        EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)                CWW30180
C                                                                              CWW30190
C       DELTA TEMPERATURE    225-249                                           CWW30200
        EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)             CWW30210
C                                                                              CWW30220
C       MOLECULAR   250-274                                                    CWW30230
        EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)                CWW30240
```

```
C                                                                       CWW30250
C     RECEIVER HEIGHT   275-299                                         CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)           CWW30270
C                                                                       CWW30280
C     TOPOGRAPHY   300-324                                              CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)           CWW30300
C                                                                       CWW30310
C     DELTA TOPOGRAPHY   325-349                                        CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)        CWW30330
C                                                                       CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                          CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)           CWW30360
C                                                                       CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                          CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)        CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                              CWW30400
C                                                                       CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                       CWW30420
C     ABSORPTION        500-524                                         CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)           CWW30440
C                                                                       CWW30450
C     DELTA ABSORPTION        525-549                                   CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)        CWW30470
C                                                                       CWW30480
C     PRESSURE        550-574                                           CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)           CWW30500
C                                                                       CWW30510
C     DELTA PRESSURE        575-599                                     CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)        CWW30530
C                                                                       CWW30540
C                                                                       CBF20130
      EQUIVALENCE (C0,W(178)), (Z0,W(179)),(CBTH0,W(180))               CBF20140
      EQUIVALENCE (PH0,W(181)),(WZ,W(182)),(WTH,W(183)),(WPH,W(184))    CBF20150
      INTEGER DFMX    ,DFNTBL(10),DFITBL(10),DFFRMTB(10)                CBF20160
      DATA RECOGDC/2.0/                                                 CBF20170
C                                                                       CBF20180
      DATA DFMX/1/                                                      CBF20190
      DATA DFNTBL/1,11,8*0/                                             CBF20200
      DATA DFITBL/1,9*0/                                                CBF20210
      DATA DFFRMTB/1,9*0/                                               CBF20220
C                                                                       CBF20230
C                                                                       CBF20240
      ENTRY SETPSP                                                      CBF20250
C                                                                       CBF20260
      DCMX=DFMX                                                         CBF20270
      CALL IMOVE(DCNTBL,DFNTBL,10)                                      CBF20280
      CALL IMOVE(DCITBL,DFITBL,10)                                      CBF20290
      CALL IMOVE(DCFRMTB,DFFRMTB,10)                                    CBF20300
C                                                                       CBF20310
      RETURN                                                            CBF20320
C                                                                       CBF20330
      ENTRY IPSPEED                                                     CBF20340
      IF(RECOGDC .NE. DCMODEL)                                          CBF20350
     1    CALL RERROR('DSPEED ','WRNG MODEL',RECOGDC)                   CBF20360
C                                                                       CBF20370
```

```
      MODC(3)=6HCBLOB2                                                  CBF20380
      MODC(4)=DCID                                                      CBF20390
C                                                                       CBF20400
      FWZ=0.0                                                           CBF20410
      FWTH=0.0                                                          CBF20420
      FWPH=0.0                                                          CBF20430
      TH0= PID2-CBTH0                                                   CBF20440
      IF(WZ.NE.0.0) FWZ=2.0/WZ/WZ                                       CBF20450
      IF(WTH.NE.0.0) FWTH=2.0/WTH/WTH                                   CBF20460
      IF(WPH.NE.0.0) FWPH=2.0/WPH/WPH                                   CBF20470
      RETURN                                                            CBF20480
C                                                                       CBF20490
      ENTRY PSPEED                                                      CBF20500
C                                                                       CBF20510
      IF(C0.EQ.0.0) RETURN                                              CBF20520
C                                                                       CBF20530
      DZ=R-EARTHR-Z0                                                    CBF20540
      DTH=TH-TH0                                                        CBF20550
      DPH=PH-PH0                                                        CBF20560
      DEXPO=0.0                                                         CBF20570
      EXPO=-0.5*(DZ*DZ*FWZ+DTH*DTH*FWTH+DPH*DPH*FWPH)                   CBF20580
      IF(EXPO .GT. -200.0) DEXPO=C0*EXP(EXPO)                           CBF20590
      DEL=1.0+DEXPO                                                     CBF20600
C                                                                       CBF20610
      PCSR=PCSR*DEL-CS*DEXPO*FWZ*DZ                                     CBF20620
      PCSTH=PCSTH*DEL-CS*DEXPO*FWTH*DTH                                 CBF20630
      PCSPH=PCSPH*DEL-CS*DEXPO*FWPH*DPH                                 CBF20640
      CS=CS*DEL                                                         CBF20650
      RETURN                                                            CBF20660
      END                                                               CBF20670
```

```
      SUBROUTINE CBLOB3                                                 CBF30020
C     SOUND SPEED PERTURBATION MODEL WITH MULTIPLE BLOBS EACH A         CBF30030
C     MULTIPLICATIVE PERTURBATION WITH EXPONENTIALLY DECAYING           CBF30040
C     EFFECT IN ALL THREE DIRECTIONS. THE FORM OF THE                   CBF30050
C     PERTURBATION IS:                                                  CBF30060
C                                                                       CBF30070
C     C = C(1+DEL)                                                      CBF30080
C     WHERE     DEL = SUM(DELSI)   AND I=1,2,3 FOR EACH BLOB.          CBF30090
C                                                                       CBF30100
C     GIVE LATITUDE INSTEAD OF CO-LATITUDE.                             CBF30110
C                                                                       CBF30120
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "B4" INSERTED HERE                                    CB4 0020
      INTEGER DCMX,DCNTBL,DCITBL,DCFRMTB,IDSDC(10)                      CB4 0040
      COMMON/B4/DCMX,DCNTBL(10),DCITBL(10),DCFRMTB(10),DCGP(10)         CB4 0050
```

```
      EQUIVALENCE (DCGP,IDSDC)                                          CB4 0060
C                                                                       CBF30160
C     COMMON DECK "CC" INSERTED HERE                                    CCC 0020
      REAL MODC                                                         CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                        CCC 0050
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
       EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),   CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND         125-149                                        CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
C                                                                       CWW30090
C     SOUND SPEED 150-174                                               CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)           CWW30110
      EQUIVALENCE (W(153),REFC)                                         CWW30120
C                                                                       CWW30130
C     DELTA SOUND SPEED 175-199                                         CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)        CWW30150
C                                                                       CWW30160
C     TEMPERATURE    200-224                                            CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)           CWW30180
C                                                                       CWW30190
C     DELTA TEMPERATURE    225-249                                      CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)        CWW30210
C                                                                       CWW30220
C     MOLECULAR   250-274                                               CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)           CWW30240
C                                                                       CWW30250
C     RECEIVER HEIGHT   275-299                                         CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)           CWW30270
C                                                                       CWW30280
C     TOPOGRAPHY   300-324                                              CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)           CWW30300
C                                                                       CWW30310
C     DELTA TOPOGRAPHY   325-349                                        CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)        CWW30330
```

```
C                                                                       CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                          CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)           CWW30360
                                                                        CWW30370
C                                                                       CWW30380
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                          CWW30390
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)        CWW30400
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                              CWW30410
C                                                                       CWW30420
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                       CWW30430
C     ABSORPTION         500-524                                        CWW30440
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)           CWW30450
C                                                                       CWW30460
C     DELTA ABSORPTION         525-549                                  CWW30470
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)        CWW30480
C                                                                       CWW30490
C     PRESSURE         550-574                                          CWW30500
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)           CWW30510
C                                                                       CWW30520
C     DELTA PRESSURE         575-599                                    CWW30530
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)        CWW30540
C                                                                       CBF30190
C                                                                       CBF30200
      REAL C0(3),Z0(3),CBTH0(3),TH0(3),PH0(3)                           CBF30210
      REAL WZ(3),WTH(3),WPH(3)                                          CBF30220
      REAL FWZ(3),FWTH(3),FWPH(3)                                       CBF30230
C                                                                       CBF30240
      EQUIVALENCE (PN,W(178)),(C0,W(179)), (Z0,W(182)),(CBTH0,W(185))   CBF30250
      EQUIVALENCE (PH0,W(188)),(WZ,W(191)),(WTH,W(194)),(WPH,W(197))    CBF30260
      INTEGER DFMX    ,DFNTBL(10),DFITBL(10),DFFRMTB(10)                CBF30270
      DATA RECOGDC/3.0/                                                 CBF30280
C                                                                       CBF30290
      DATA DFMX/1/                                                      CBF30300
      DATA DFNTBL/1,11,8*0/                                             CBF30310
      DATA DFITBL/1,9*0/                                                CBF30320
      DATA DFFRMTB/1,9*0/                                               CBF30330
C                                                                       CBF30340
C                                                                       CBF30350
      ENTRY SETPSP                                                      CBF30360
C                                                                       CBF30370
      DCMX=DFMX                                                         CBF30380
      CALL IMOVE(DCNTBL,DFNTBL,10)                                      CBF30390
      CALL IMOVE(DCITBL,DFITBL,10)                                      CBF30400
      CALL IMOVE(DCFRMTB,DFFRMTB,10)                                    CBF30410
C                                                                       CBF30420
      RETURN                                                            CBF30430
C                                                                       CBF30440
      ENTRY IPSPEED                                                     CBF30450
      IF(RECOGDC .NE. DCMODEL)                                          CBF30460
     1    CALL RERROR('DSPEED ','WRNG MODEL',RECOGDC)                   CBF30470
C                                                                       CBF30480
      MODC(3)=6HCBLOB3                                                  CBF30490
      MODC(4)=DCID                                                      CBF30500
C                                                                       CBF30510
      NP=PN                                                             CBF30520
      DO 10 I=1,NP                                                      CBF30530
```

```
      FWZ(I)=0.0                                                        CBF30530
      FWTH(I)=0.0                                                       CBF30540
      FWPH(I)=0.0                                                       CBF30550
      TH0(I)= PID2-CBTH0(I)                                             CBF30560
      IF(WZ(I).NE.0.0) FWZ(I)=2.0/WZ(I)/WZ(I)                           CBF30570
      IF(WTH(I).NE.0.0) FWTH(I)=2.0/WTH(I)/WTH(I)                       CBF30580
      IF(WPH(I).NE.0.0) FWPH(I)=2.0/WPH(I)/WPH(I)                       CBF30590
10    CONTINUE                                                          CBF30600
      RETURN                                                            CBF30610
C                                                                       CBF30620
      ENTRY PSPEED                                                      CBF30630
C                                                                       CBF30640
      DEL=1.0                                                           CBF30650
C     INITIALIZE PARTIALS OF DEL                                        CBF30660
      PDELR=0.0                                                         CBF30670
      PDELTH=0.0                                                        CBF30680
      PDELPH=0.0                                                        CBF30690
C                                                                       CBF30700
      DO 20 I=1,NP                                                      CBF30710
      DZ=R-EARTHR-Z0(I)                                                 CBF30720
      DTH=TH-TH0(I)                                                     CBF30730
      DPH=PH-PH0(I)                                                     CBF30740
      DEXPO=0.0                                                         CBF30750
      EXPO=-0.5*(DZ*DZ*FWZ(I)+DTH*DTH*FWTH(I)+DPH*DPH*FWPH(I))          CBF30760
      IF(EXPO .GT. -200.0) DEXPO=C0(I)*EXP(EXPO)                        CBF30770
C                                                                       CBF30780
      DEL=DEL+DEXPO                                                     CBF30790
      PDELR=PDELR-DEXPO*FWZ(I)*DZ                                       CBF30800
      PDELTH=PDELTH-DEXPO*FWTH(I)*DTH                                   CBF30810
20    PDELPH=PDELPH-DEXPO*FWPH(I)*DPH                                   CBF30820
C                                                                       CBF30830
C     IN THIS MODEL WE ARE USING THE SQUARE OF DEL                      CBF30840
C     AS THE CS PERTURBATION                                            CBF30850
C                                                                       CBF30860
      FTR=CS*2.0*DEL                                                    CBF30870
      DEL=DEL*DEL                                                       CBF30880
C                                                                       CBF30890
      CS=CS*DEL                                                         CBF30900
      PCSR=PCSR*DEL+FTR*PDELR                                           CBF30910
      PCSTH=PCSTH*DEL+FTR*PDELTH                                        CBF30920
      PCSPH=PCSPH*DEL+FTR*PDELPH                                        CBF30930
      RETURN                                                            CBF30940
      END                                                               CBF30950
```

```
      SUBROUTINE CBLOB4                                                 CBF40020
C     SOUND SPEED PERTURBATION MODEL WITH MULTIPLE BLOBS EACH A         CBF40030
C     MULTIPLICATIVE PERTURBATION WITH EXPONENTIALLY DECAYING           CBF40040
C     EFFECT IN ALL THREE DIRECTIONS. THE FORM OF THE                   CBF40050
C     PERTURBATION IS:                                                  CBF40060
C                                                                       CBF40070
C     CS = CS(1+DEL)                                                    CBF40080
```

```
C     WHERE     DEL = SUM(DELSI)   AND I=1,2,3 FOR EACH BLOB.                CBF40090
C                                                                            CBF40100
C     GIVE LATITUDE INSTEAD OF CO-LATITUDE.                                  CBF40110
C                                                                            CBF40120
C     COMMON DECK "CONST" INSERTED HERE                                      CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                          CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                              CCON0050
C     COMMON DECK "RKAM" INSERTED HERE                                       RKAM0020
      REAL KR,KTH,KPH                                                        RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)            RKAM0050
C     COMMON DECK "B4" INSERTED HERE                                         CB4 0020
      INTEGER DCMX,DCNTBL,DCITBL,DCFRMTB,IDSDC(10)                           CB4 0040
      COMMON/B4/DCMX,DCNTBL(10),DCITBL(10),DCFRMTB(10),DCGP(10)              CB4 0050
      EQUIVALENCE (DCGP,IDSDC)                                               CB4 0060
C                                                                            CBF40160
C     COMMON DECK "CC" INSERTED HERE                                         CCC 0020
      REAL MODC                                                              CCC 0040
      COMMON/CC/MODC(4),CS,PCST,PCSR,PCSTH,PCSPH                             CCC 0050
C     COMMON DECK "WW" INSERTED HERE                                         CWW 0020
      PARAMETER (NWARSZ=1000)                                                CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                        CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                     CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),         CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),           CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),               CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),              CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                       CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))     CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                           CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                           CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                          CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),            CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))             CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),             CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                   CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                        CWW20160
      REAL MMODEL,MFORM,MID                                                  CWW30020
C                                                                            CWW30030
C     WIND         100-124                                                   CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)                CWW30050
C                                                                            CWW30060
C     DELTA WIND         125-149                                             CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)             CWW30080
C                                                                            CWW30090
C     SOUND SPEED 150-174                                                    CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)                CWW30110
      EQUIVALENCE (W(153),REFC)                                              CWW30120
C                                                                            CWW30130
C     DELTA SOUND SPEED 175-199                                              CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)             CWW30150
C                                                                            CWW30160
C     TEMPERATURE    200-224                                                 CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)                CWW30180
C                                                                            CWW30190
C     DELTA TEMPERATURE    225-249                                           CWW30200
```

```
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)          CWW30210
C                                                                         CWW30220
C     MOLECULAR   250-274                                                 CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)             CWW30240
C                                                                         CWW30250
C     RECEIVER HEIGHT   275-299                                           CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)             CWW30270
C                                                                         CWW30280
C     TOPOGRAPHY   300-324                                                CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)             CWW30300
C                                                                         CWW30310
C     DELTA TOPOGRAPHY   325-349                                          CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)          CWW30330
C                                                                         CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                            CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)             CWW30360
C                                                                         CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                            CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)          CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                CWW30400
C                                                                         CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                         CWW30420
C     ABSORPTION         500-524                                          CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)             CWW30440
C                                                                         CWW30450
C     DELTA ABSORPTION         525-549                                    CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)          CWW30470
C                                                                         CWW30480
C     PRESSURE         550-574                                            CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)             CWW30500
C                                                                         CWW30510
C     DELTA PRESSURE         575-599                                      CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)          CWW30530
C                                                                         CWW30540
C                                                                         CBF40190
      REAL C0(3),Z0(3),CBTH0(3),TH0(3),PH0(3)                             CBF40200
      REAL WZ(3),WTH(3),WPH(3)                                            CBF40210
      REAL FWZ(3),FWTH(3),FWPH(3)                                         CBF40220
C                                                                         CBF40230
      EQUIVALENCE (PN,W(178)),(C0,W(179)), (Z0,W(182)),(CBTH0,W(185))     CBF40240
      EQUIVALENCE (PH0,W(188)),(WZ,W(191)),(WTH,W(194)),(WPH,W(197))      CBF40250
      INTEGER DFMX     ,DFNTBL(10),DFITBL(10),DFFRMTB(10)                 CBF40260
      DATA RECOGDC/4.0/                                                   CBF40270
C                                                                         CBF40280
      DATA DFMX/1/                                                        CBF40290
      DATA DFNTBL/1,11,8*0/                                               CBF40300
      DATA DFITBL/1,9*0/                                                  CBF40310
      DATA DFFRMTB/1,9*0/                                                 CBF40320
C                                                                         CBF40330
C                                                                         CBF40340
      ENTRY SETPSP                                                        CBF40350
C                                                                         CBF40360
      DCMX=DFMX                                                           CBF40370
      CALL IMOVE(DCNTBL,DFNTBL,10)                                        CBF40380
      CALL IMOVE(DCITBL,DFITBL,10)                                        CBF40390
```

```
      CALL IMOVE(DCFRMTB,DFFRMTB,10)                                    CBF40400
                                                                        CBF40410
C                                                                       CBF40420
      RETURN                                                            CBF40430
C                                                                       CBF40440
      ENTRY IPSPEED                                                     CBF40450
      IF(RECOGDC .NE. DCMODEL)                                          CBF40460
     1    CALL RERROR('DSPEED ','WRNG MODEL',RECOGDC)                  CBF40470
C                                                                       CBF40480
      MODC(3)=6HCBLOB4                                                  CBF40490
      MODC(4)=DCID                                                      CBF40500
C                                                                       CBF40510
      NP=PN                                                             CBF40520
      DO 10 I=1,NP                                                      CBF40530
      FWZ(I)=0.0                                                        CBF40540
      FWTH(I)=0.0                                                       CBF40550
      FWPH(I)=0.0                                                       CBF40560
      TH0(I)= PID2-CBTH0(I)                                             CBF40570
      IF(WZ(I).NE.0.0) FWZ(I)=2.0/WZ(I)/WZ(I)                           CBF40580
      IF(WTH(I).NE.0.0) FWTH(I)=2.0/WTH(I)/WTH(I)                       CBF40590
      IF(WPH(I).NE.0.0) FWPH(I)=2.0/WPH(I)/WPH(I)                       CBF40600
10    CONTINUE                                                          CBF40610
      RETURN                                                            CBF40620
C                                                                       CBF40630
      ENTRY PSPEED                                                      CBF40640
C                                                                       CBF40650
      DEL=1.0                                                           CBF40660
C     INITIALIZE PARTIALS OF DEL                                        CBF40670
      PDELR=0.0                                                         CBF40680
      PDELTH=0.0                                                        CBF40690
      PDELPH=0.0                                                        CBF40700
C                                                                       CBF40710
      DO 20 I=1,NP                                                      CBF40720
      DZ=R-EARTHR-Z0(I)                                                 CBF40730
      DTH=TH-TH0(I)                                                     CBF40740
      DPH=PH-PH0(I)                                                     CBF40750
      DEXPO=0.0                                                         CBF40760
      EXPO=-0.5*(DZ*DZ*FWZ(I)+DTH*DTH*FWTH(I)+DPH*DPH*FWPH(I))          CBF40770
      IF(EXPO .GT. -200.0) DEXPO=C0(I)*EXP(EXPO)                        CBF40780
C                                                                       CBF40790
      DEL=DEL+DEXPO                                                     CBF40800
      PDELR=PDELR-DEXPO*FWZ(I)*DZ                                       CBF40810
      PDELTH=PDELTH-DEXPO*FWTH(I)*DTH                                   CBF40820
20    PDELPH=PDELPH-DEXPO*FWPH(I)*DPH                                   CBF40830
C                                                                       CBF40840
      FTR=CS                                                            CBF40850
C                                                                       CBF40860
      CS=CS*DEL                                                         CBF40870
      PCSR=PCSR*DEL+FTR*PDELR                                           CBF40880
      PCSTH=PCSTH*DEL+FTR*PDELTH                                        CBF40890
      PCSPH=PCSPH*DEL+FTR*PDELPH                                        CBF40900
      RETURN                                                            CBF40910
      END
```

```
      SUBROUTINE SLLOSS                                                 SLLS0020
C     SLLOSS BACKGROUND ABSORPTION FORMULA                              SLLS0030
C                                                                       SLLS0040
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
       REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                               CWW20020
       EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),   CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
       REAL MMODEL,MFORM,MID                                            CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND         125-149                                        CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
C                                                                       CWW30090
C     SOUND SPEED 150-174                                               CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)           CWW30110
      EQUIVALENCE (W(153),REFC)                                         CWW30120
C                                                                       CWW30130
C     DELTA SOUND SPEED 175-199                                         CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)        CWW30150
C                                                                       CWW30160
C     TEMPERATURE    200-224                                            CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)           CWW30180
C                                                                       CWW30190
C     DELTA TEMPERATURE    225-249                                      CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)        CWW30210
C                                                                       CWW30220
C     MOLECULAR   250-274                                               CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)           CWW30240
C                                                                       CWW30250
C     RECEIVER HEIGHT   275-299                                         CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)           CWW30270
C                                                                       CWW30280
C     TOPOGRAPHY   300-324                                              CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)           CWW30300
C                                                                       CWW30310
C     DELTA TOPOGRAPHY   325-349                                        CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)        CWW30330
```

```
C                                                                       CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                          CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)           CWW30360
                                                                        CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                          CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)        CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                              CWW30400
                                                                        CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                       CWW30420
C     ABSORPTION         500-524                                        CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)           CWW30440
                                                                        CWW30450
C     DELTA ABSORPTION         525-549                                  CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)        CWW30470
                                                                        CWW30480
C     PRESSURE         550-574                                          CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)           CWW30500
                                                                        CWW30510
C     DELTA PRESSURE         575-599                                    CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)        CWW30530
                                                                        CWW30540
C                                                                       SLLS0060
      EQUIVALENCE (W(503),ACOEF),(W(504),BCOEF),(W(505),OMEG1)          SLLS0070
      EQUIVALENCE (W(506),OMEG2)                                        SLLS0080
                                                                        SLLS0090
C     COMMON DECK "RINREAL" INSERTED HERE                               CRIN0020
      LOGICAL SPACE                                                     CRIN0040
      REAL LPOLAR,LPOLRI,KPHK,KPHKI,KAY2,KAY2I                          CRIN0050
      CHARACTER DISPM*6                                                 CRIN0060
      COMMON/RINPL/DISPM                                                CRIN0070
      COMMON /RIN/ MODRIN(8),RAYNAME(2,3),TYPE(3),SPACE                 CRIN0080
      COMMON/RIN/OMEGMIN,OMEGMAX,KAY2,KAY2I,                            CRIN0090
     1             H,HI,PHT,PHTI,PHR,PHRI,PHTH,PHTHI,PHPH,PHPHI         CRIN0100
     2, PHOW,PHOWI,PHKR,PHKRI,PHKTH,PHKTI, PHKPH,PHKPI                  CRIN0110
     3 ,KPHK,KPHKI,POLAR,POLARI,LPOLAR,LPOLRI,SGN                       CRIN0120
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     COMMON DECK "LL" INSERTED HERE                                    CLL 0020
      REAL MODL                                                         CLL 0040
      COMMON/LL/MODL(4),APH,APHPT,APHPR,APHPTH,APHPPH                   CLL 0050
C                                                                       SLLS0130
C     COMMON DECK "CB17" INSERTED HERE                                  CB170020
      INTEGER VMX,VNTBL,VITBL,VFRMTBL,IDSV(10)                          CB170040
      COMMON/B17/VMX,VNTBL(10),VITBL(10),VFRMTBL(10),VGP(53)            CB170050
      EQUIVALENCE (VGP,IDSV),(ANV,VGP(11))                              CB170060
                                                                        SLLS0150
C     INTEGER VPX     ,VQTBL(10),VLTBL(10),VIRMTBL(10)                  SLLS0160
                                                                        SLLS0170
C                                                                       SLLS0180
      DATA VPX/1/                                                       SLLS0190
      DATA VQTBL/1,11,8*0/                                              SLLS0200
      DATA VLTBL/1,9*0/                                                 SLLS0210
      DATA VIRMTBL/1,9*0/                                               SLLS0220
      DATA RECOGA/1.0/                                                  SLLS0230
C
```

```
C                                                                       SLLS0240
      ENTRY SETABS                                                      SLLS0250
C                                                                       SLLS0260
      VMX=VPX                                                           SLLS0270
      CALL IMOVE(VNTBL,VQTBL,10)                                        SLLS0280
      CALL IMOVE(VITBL,VLTBL,10)                                        SLLS0290
      CALL IMOVE(VFRMTBL,VIRMTBL,10)                                    SLLS0300
      CALL SETPAB                                                       SLLS0310
C                                                                       SLLS0320
      RETURN                                                            SLLS0330
C                                                                       SLLS0340
      ENTRY IABSRP                                                      SLLS0350
      IF(RECOGA .NE. AMODEL)                                            SLLS0360
     1    CALL RERROR('ABSRP ','WRNG MODEL',RECOGA)                     SLLS0370
      MODL(1)=6HSLLOSS                                                  SLLS0380
      MODL(2)=AID                                                       SLLS0390
      CALL IPABSRP                                                      SLLS0400
      RETURN                                                            SLLS0410
C                                                                       SLLS0420
      ENTRY ABSRP                                                       SLLS0430
      OWS=OW*OW                                                         SLLS0440
      APH=ACOEF*(OW/OMEG1)**2 + BCOEF*OWS/(OMEG2**2+OWS)                SLLS0450
      CALL PABSRP                                                       SLLS0460
      END                                                               SLLS0470
```

```
      SUBROUTINE NPABSR                                                 NPWR0020
C     DO-NOTHING ABSORPTION PERTURBATION MODEL                          NPWR0030
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND        125-149                                         CWW30070
```

```
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)          CWW30080
C                                                                         CWW30090
C     SOUND SPEED 150-174                                                 CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)             CWW30110
      EQUIVALENCE (W(153),REFC)                                           CWW30120
C                                                                         CWW30130
C     DELTA SOUND SPEED 175-199                                           CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)          CWW30150
C                                                                         CWW30160
C     TEMPERATURE    200-224                                              CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)             CWW30180
C                                                                         CWW30190
C     DELTA TEMPERATURE    225-249                                        CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)          CWW30210
C                                                                         CWW30220
C     MOLECULAR   250-274                                                 CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)             CWW30240
C                                                                         CWW30250
C     RECEIVER HEIGHT   275-299                                           CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)             CWW30270
C                                                                         CWW30280
C     TOPOGRAPHY   300-324                                                CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)             CWW30300
C                                                                         CWW30310
C     DELTA TOPOGRAPHY   325-349                                          CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)          CWW30330
C                                                                         CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                            CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)             CWW30360
C                                                                         CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                            CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)          CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                CWW30400
C                                                                         CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                         CWW30420
C     ABSORPTION        500-524                                           CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)             CWW30440
C                                                                         CWW30450
C     DELTA ABSORPTION        525-549                                     CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)          CWW30470
C                                                                         CWW30480
C     PRESSURE        550-574                                             CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)             CWW30500
C                                                                         CWW30510
C     DELTA PRESSURE        575-599                                       CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)          CWW30530
C                                                                         CWW30540
C     COMMON DECK "LL" INSERTED HERE                                      CLL 0020
      REAL MODL                                                           CLL 0040
      COMMON/LL/MODL(4),APH,APHPT,APHPR,APHPTH,APHPPH                     CLL 0050
C                                                                         NPWR0060
C     COMMON DECK "CB18" INSERTED HERE                                    CB180020
      INTEGER DVMX,DVNTBL,DVITBL,DVFRMTB,IDSDV(10)                        CB180040
      COMMON/B18/DVMX,DVNTBL(10),DVITBL(10),DVFRMTB(10),DVGP(11)          CB180050
      EQUIVALENCE (DVGP,IDSDV),(ANDV,DVGP(11))                            CB180060
```

```
C                                                                       NPWR0080
      INTEGER DYMX    ,DYNTBL(10),DYITBL(10),DYFRMTB(10)                NPWR0090
C                                                                       NPWR0100
      DATA DYMX/1/                                                      NPWR0110
      DATA DYNTBL/1,11,8*0/                                             NPWR0120
      DATA DYITBL/1,9*0/                                                NPWR0130
      DATA DYFRMTB/1,9*0/                                               NPWR0140
C                                                                       NPWR0150
      DATA RECOGDA/0.0/                                                 NPWR0160
C                                                                       NPWR0170
C                                                                       NPWR0180
       ENTRY SETPAB                                                     NPWR0190
C                                                                       NPWR0200
       DVMX=DYMX                                                        NPWR0210
       CALL IMOVE(DVNTBL,DYNTBL,10)                                     NPWR0220
       CALL IMOVE(DVITBL,DYITBL,10)                                     NPWR0230
       CALL IMOVE(DVFRMTB,DYFRMTB,10)                                   NPWR0240
C                                                                       NPWR0250
       RETURN                                                           NPWR0260
C                                                                       NPWR0270
       ENTRY IPABSRP                                                    NPWR0280
       IF(RECOGDA .NE. DAMODEL)                                         NPWR0290
     1    CALL RERROR('DABSRP ','WRNG MODEL',RECOGDA)                   NPWR0300
       MODL(3)=7HNPABSR                                                 NPWR0310
       MODL(4)=DAID                                                     NPWR0320
       RETURN                                                           NPWR0330
C                                                                       NPWR0340
       ENTRY PABSRP                                                     NPWR0350
       RETURN                                                           NPWR0360
       END                                                              NPWR0370
```

```
      SUBROUTINE GHORIZ                                                 GHJZ0020
C     TERRAIN MODEL USING FIXED OFFSET TO EARTHR'S SURFACE              GHJZ0030
C     COMMON DECK "GG" INSERTED HERE                                    CGG 0020
      REAL MODG                                                         CGG 0040
      COMMON/GG/MODG(4)                                                 CGG 0050
      COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                  CGG 0060
      COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME            CGG 0070
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
```

```
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),        CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))         CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),         CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))               CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                    CWW20160
      REAL MMODEL,MFORM,MID                                              CWW30020
C                                                                        CWW30030
C     WIND         100-124                                             CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)            CWW30050
C                                                                        CWW30060
C     DELTA WIND         125-149                                       CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)         CWW30080
C                                                                        CWW30090
C     SOUND SPEED 150-174                                              CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)            CWW30110
      EQUIVALENCE (W(153),REFC)                                          CWW30120
C                                                                        CWW30130
C     DELTA SOUND SPEED 175-199                                        CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)         CWW30150
C                                                                        CWW30160
C     TEMPERATURE    200-224                                           CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)            CWW30180
C                                                                        CWW30190
C     DELTA TEMPERATURE    225-249                                     CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)         CWW30210
C                                                                        CWW30220
C     MOLECULAR   250-274                                              CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)            CWW30240
C                                                                        CWW30250
C     RECEIVER HEIGHT   275-299                                        CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)            CWW30270
C                                                                        CWW30280
C     TOPOGRAPHY   300-324                                             CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)            CWW30300
C                                                                        CWW30310
C     DELTA TOPOGRAPHY   325-349                                       CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)         CWW30330
C                                                                        CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                         CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)            CWW30360
C                                                                        CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                         CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)         CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                             CWW30400
C                                                                        CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                        CWW30420
C     ABSORPTION         500-524                                       CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)            CWW30440
C                                                                        CWW30450
C     DELTA ABSORPTION         525-549                                 CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)         CWW30470
C                                                                        CWW30480
C     PRESSURE         550-574                                         CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)            CWW30500
C                                                                        CWW30510
```

```
C       DELTA PRESSURE          575-599                                         CWW30520
        EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)              CWW30530
C                                                                               CWW30540
C       COMMON DECK "RKAM" INSERTED HERE                                        RKAM0020
        REAL KR,KTH,KPH                                                         RKAM0040
        COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)             RKAM0050
C                                                                               GHJZ0070
        EQUIVALENCE (W(303), Z0)                                                GHJZ0080
C                                                                               GHJZ0090
C       COMMON DECK "B9" INSERTED HERE                                          CB8 0020
        INTEGER GMX,GNTBL,GITBL,GFRMTBL,IDSG(10)                                CB8 0040
        COMMON/B9/GMX,GNTBL(10),GITBL(10),GFRMTBL(10),GGP(113)                  CB8 0050
        EQUIVALENCE (GGP,IDSG),(ANG,GGP(11))                                    CB8 0060
C                                                                               GHJZ0110
        INTEGER GPX     ,GQTBL(10),GLTBL(10),GIRMTBL(10)                        GHJZ0120
        DATA RECOGG/1.0/                                                        GHJZ0130
C                                                                               GHJZ0140
C                                                                               GHJZ0150
        DATA GPX/1/                                                             GHJZ0160
        DATA GQTBL/1,11,8*0/                                                    GHJZ0170
        DATA GLTBL/1,9*0/                                                       GHJZ0180
        DATA GIRMTBL/1,9*0/                                                     GHJZ0190
C                                                                               GHJZ0200
C                                                                               GHJZ0210
        ENTRY SETTOP                                                            GHJZ0220
C                                                                               GHJZ0230
        GMX=GPX                                                                 GHJZ0240
        CALL IMOVE(GNTBL,GQTBL,10)                                              GHJZ0250
        CALL IMOVE(GITBL,GLTBL,10)                                              GHJZ0260
        CALL IMOVE(GFRMTBL,GIRMTBL,10)                                          GHJZ0270
        CALL SETPTP                                                             GHJZ0280
C                                                                               GHJZ0290
        RETURN                                                                  GHJZ0300
C                                                                               GHJZ0310
        ENTRY ITOPOG                                                            GHJZ0320
        IF(RECOGG .NE. GMODEL)                                                  GHJZ0330
     1     CALL RERROR('GHORIZ ','WRNG MODEL',RECOGG)                           GHJZ0340
        MODG(1)=6HGHORIZ                                                        GHJZ0350
        MODG(2)=GID                                                             GHJZ0360
        CALL IPTOPOG                                                            GHJZ0370
        RETURN                                                                  GHJZ0380
C                                                                               GHJZ0390
        ENTRY TOPOG                                                             GHJZ0400
        G=R-W(1)-Z0                                                             GHJZ0410
        PGR=1.0                                                                 GHJZ0420
        CALL CLEAR(PGRR,8)                                                      GHJZ0430
        END                                                                     GHJZ0440
```

```
        SUBROUTINE GLORENZ                                                      GLRZ0020
C       TERRAIN MODEL USING LORENZIAN SHAPED HORIZONTAL SURFACE LOCATED A       GLRZ0030
C       ARBITRARY GEOGRAPHICAL LOCATION.                                        GLRZ0040
```

```
C                                                                       GLRZ0050
C       COMMON DECK "CONST" INSERTED HERE                               CCON0020
        COMMON/PCONST/CREF,RGAS,GAMMA                                   CCON0040
        COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                       CCON0050
C       COMMON DECK "B9" INSERTED HERE                                  CB8 0020
        INTEGER GMX,GNTBL,GITBL,GFRMTBL,IDSG(10)                        CB8 0040
        COMMON/B9/GMX,GNTBL(10),GITBL(10),GFRMTBL(10),GGP(113)          CB8 0050
        EQUIVALENCE (GGP,IDSG),(ANG,GGP(11))                           CB8 0060
C       COMMON DECK "RKAM" INSERTED HERE                                RKAM0020
        REAL KR,KTH,KPH                                                 RKAM0040
        COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)     RKAM0050
C       COMMON DECK "GG" INSERTED HERE                                  CGG 0020
        REAL MODG                                                       CGG 0040
        COMMON/GG/MODG(4)                                               CGG 0050
        COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                CGG 0060
        COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME          CGG 0070
C       COMMON DECK "B10" INSERTED HERE                                 CB9 0020
        INTEGER DGMX,DGNTBL,DGITBL,DGFRMTB,IDSDG(10)                    CB9 0040
        COMMON/B10/DGMX,DGNTBL(10),DGITBL(10),DGFRMTB(10),DGGP(10)      CB9 0050
        EQUIVALENCE (DGGP,IDSDG)                                        CB9 0060
C       COMMON DECK "WW" INSERTED HERE                                  CWW 0020
        PARAMETER (NWARSZ=1000)                                         CWW10030
        COMMON/WW/ID(10),MAXW,W(NWARSZ)                                 CWW10040
        REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                              CWW20020
        EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),  CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
        REAL MMODEL,MFORM,MID                                           CWW30020
C                                                                       CWW30030
C       WIND         100-124                                            CWW30040
        EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)         CWW30050
C                                                                       CWW30060
C       DELTA WIND         125-149                                      CWW30070
        EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)      CWW30080
C                                                                       CWW30090
C       SOUND SPEED 150-174                                             CWW30100
        EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)         CWW30110
        EQUIVALENCE (W(153),REFC)                                       CWW30120
C                                                                       CWW30130
C       DELTA SOUND SPEED 175-199                                       CWW30140
        EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)      CWW30150
C                                                                       CWW30160
C       TEMPERATURE    200-224                                          CWW30170
        EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)         CWW30180
```

```
C                                                                       CWW30190
C     DELTA TEMPERATURE    225-249                                      CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)        CWW30210
C                                                                       CWW30220
C     MOLECULAR   250-274                                               CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)           CWW30240
C                                                                       CWW30250
C     RECEIVER HEIGHT   275-299                                         CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)           CWW30270
C                                                                       CWW30280
C     TOPOGRAPHY   300-324                                              CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)           CWW30300
C                                                                       CWW30310
C     DELTA TOPOGRAPHY   325-349                                        CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)        CWW30330
C                                                                       CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                          CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)           CWW30360
C                                                                       CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                          CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)        CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                              CWW30400
C                                                                       CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                       CWW30420
C     ABSORPTION         500-524                                        CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)           CWW30440
C                                                                       CWW30450
C     DELTA ABSORPTION         525-549                                  CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)        CWW30470
C                                                                       CWW30480
C     PRESSURE         550-574                                          CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)           CWW30500
C                                                                       CWW30510
C     DELTA PRESSURE         575-599                                    CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)        CWW30530
C                                                                       CWW30540
      EQUIVALENCE (GCZAMP,W(303))                                       GLRZ0120
      EQUIVALENCE (GCLAMZ,W(304)) ,(GCTHDL,W(305))                      GLRZ0130
      EQUIVALENCE (GCBASE,W(306))                                       GLRZ0140
C                                                                       GLRZ0150
      INTEGER GPX     ,GQTBL(10),GLTBL(10),GIRMTBL(10)                  GLRZ0160
      DATA RECOGG/4.0/                                                  GLRZ0170
C                                                                       GLRZ0180
      DATA GPX/1/                                                       GLRZ0190
      DATA GQTBL/1,11,8*0/                                              GLRZ0200
      DATA GLTBL/1,9*0/                                                 GLRZ0210
      DATA GIRMTBL/1,9*0/                                               GLRZ0220
C                                                                       GLRZ0230
      ENTRY SETTOP                                                      GLRZ0240
C                                                                       GLRZ0250
      GMX=GPX                                                           GLRZ0260
      CALL IMOVE(GNTBL,GQTBL,10)                                        GLRZ0270
      CALL IMOVE(GITBL,GLTBL,10)                                        GLRZ0280
      CALL IMOVE(GFRMTBL,GIRMTBL,10)                                    GLRZ0290
      CALL SETPTP                                                       GLRZ0300
```

```
C                                                                       GLRZ0310
      RETURN                                                            GLRZ0320
C                                                                       GLRZ0330
      ENTRY ITOPOG                                                      GLRZ0340
C                                                                       GLRZ0350
      IF(RECOGG .NE. GMODEL)                                            GLRZ0360
     1    CALL RERROR('GROUND ','WRNG MODEL',RECOGG)                    GLRZ0370
      MODG(1)=7HGLORENZ                                                 GLRZ0380
      MODG(2)=GID                                                       GLRZ0390
C                                                                       GLRZ0400
      GCTH0=PID2-GCLAMZ                                                 GLRZ0410
      GCINV=1.0/GCTHDL                                                  GLRZ0420
      CALL IPTOPOG                                                      GLRZ0430
      RETURN                                                            GLRZ0440
C                                                                       GLRZ0450
      ENTRY TOPOG                                                       GLRZ0460
C                                                                       GLRZ0470
      CALL CLEAR(PGRR,8)                                                GLRZ0480
C                                                                       GLRZ0490
      ETA=(TH-GCTH0)*GCINV                                              GLRZ0500
      ETA2=ETA*ETA                                                      GLRZ0510
      GBINOM=1.0/(1.0 + ETA2)                                           GLRZ0520
      Z=GCZAMP*GBINOM                                                   GLRZ0530
      G=R-EARTHR-Z-GCBASE                                               GLRZ0540
C                                                                       GLRZ0550
      PGR=1.0                                                           GLRZ0560
      GBINOMB=GBINOM*GCINV                                              GLRZ0570
      PGTH=2.0*Z*ETA*GBINOMB                                            GLRZ0580
      PGTHTH=2.0*Z*GBINOMB*GBINOMB*(1.0-3.0*ETA2)                       GLRZ0590
C                                                                       GLRZ0600
      RETURN                                                            GLRZ0610
      END                                                               GLRZ0620
```

```
      SUBROUTINE GTANH                                                  GTLH0020
C     TERRAIN PROFILE REPRESENTED BY A SEQUENCE OF LINEAR SEGMENTS      GTLH0030
C     SMOOTHLY JOINED BY HYPERBOLIC FUNCTIONS. PARAMETERS ARE INPUT     GTLH0040
C     AS TABULAR DATA WITH SLOPES COMPUTED FROM TERRAIN DATA.           GTLH0050
C     COMMON DECK "CONST" INSERTED HERE                                 CCON0020
      COMMON/PCONST/CREF,RGAS,GAMMA                                     CCON0040
      COMMON/MCONST/PI,PIT2,PID2,DEGS,RAD,ALN10                         CCON0050
C     TERRAIN MODEL                                                     GTLH0070
      REAL C(49), LAM0,LMI,LMIM1,LM(49), DL(49),ALC(50)                 GTLH0080
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "GG" INSERTED HERE                                    CGG 0020
      REAL MODG                                                         CGG 0040
      COMMON/GG/MODG(4)                                                 CGG 0050
      COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                  CGG 0060
      COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME            CGG 0070
C     COMMON DECK "B9" INSERTED HERE                                    CB8 0020
```

```
      INTEGER GMX,GNTBL,GITBL,GFRMTBL,IDSG(10)                          CB8 0040
      COMMON/B9/GMX,GNTBL(10),GITBL(10),GFRMTBL(10),GGP(113)            CB8 0050
      EQUIVALENCE (GGP,IDSG),(ANG,GGP(11))                              CB8 0060
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND         125-149                                        CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
C                                                                       CWW30090
C     SOUND SPEED 150-174                                               CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)           CWW30110
      EQUIVALENCE (W(153),REFC)                                         CWW30120
C                                                                       CWW30130
C     DELTA SOUND SPEED 175-199                                         CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)        CWW30150
C                                                                       CWW30160
C     TEMPERATURE    200-224                                            CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)           CWW30180
C                                                                       CWW30190
C     DELTA TEMPERATURE    225-249                                      CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)        CWW30210
C                                                                       CWW30220
C     MOLECULAR   250-274                                               CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)           CWW30240
C                                                                       CWW30250
C     RECEIVER HEIGHT   275-299                                         CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)           CWW30270
C                                                                       CWW30280
C     TOPOGRAPHY   300-324                                              CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)           CWW30300
C                                                                       CWW30310
C     DELTA TOPOGRAPHY   325-349                                        CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)        CWW30330
C                                                                       CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                          CWW30350
```

```
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)          CWW30360
C                                                                      CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                        CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)       CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                             CWW30400
C                                                                      CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                      CWW30420
C     ABSORPTION         500-524                                       CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)          CWW30440
C                                                                      CWW30450
C     DELTA ABSORPTION         525-549                                 CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)       CWW30470
C                                                                      CWW30480
C     PRESSURE         550-574                                         CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)          CWW30500
C                                                                      CWW30510
C     DELTA PRESSURE         575-599                                   CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)       CWW30530
C                                                                      CWW30540
      EQUIVALENCE (LAM0,GGP(12)),(Z0,GGP(62)),(DL0,GGP(112))           GTLH0130
      EQUIVALENCE (LM,GGP(13)),(C,GGP(63)),(DL,GGP(113))               GTLH0140
C                                                                      GTLH0150
      INTEGER GPX    ,GQTBL(10),GLTBL(10),GIRMTBL(10)                  GTLH0160
      DATA RECOGG/3.0/                                                 GTLH0170
C                                                                      GTLH0180
      DATA AQG/0.0/                                                    GTLH0190
      DATA GPX/2/                                                      GTLH0200
      DATA GQTBL/1,11,162,7*0/                                         GTLH0210
      DATA GLTBL/1,50,8*0/                                             GTLH0220
      DATA GIRMTBL/1,2,8*0/                                            GTLH0230
C                                                                      GTLH0240
      ENTRY SETTOP                                                     GTLH0250
C                                                                      GTLH0260
      ANG=AQG                                                          GTLH0270
      GMX=GPX                                                          GTLH0280
      CALL IMOVE(GNTBL,GQTBL,10)                                       GTLH0290
      CALL IMOVE(GITBL,GLTBL,10)                                       GTLH0300
      CALL IMOVE(GFRMTBL,GIRMTBL,10)                                   GTLH0310
      CALL SETPTP                                                      GTLH0320
C                                                                      GTLH0330
      RETURN                                                           GTLH0340
C                                                                      GTLH0350
      ENTRY ITOPOG                                                     GTLH0360
C                                                                      GTLH0370
      CALL IPTOPOG                                                     GTLH0380
C                                                                      GTLH0390
C     IF HAD PREVIOUS CALL BUT NOTHING THIS TIME, EXIT NOW             GTLH0400
C     RETAINING PREVIOUS TABULAR DATA COUNT                            GTLH0410
      IF(N.GT.0 .AND. ANG.EQ.0.0) RETURN                               GTLH0420
C                                                                      GTLH0430
      IF(RECOGG .NE. GMODEL)                                           GTLH0440
     1    CALL RERROR('TOPO    ','WRNG MODEL',RECOGG)                  GTLH0450
      MODG(1)=5HGTANH                                                  GTLH0460
      MODG(2)=GID                                                      GTLH0470
      N=ANG/3                                                          GTLH0480
```

```
      IF(ANG.NE.3*N.OR.N.LE.0)                                          GTLH0490
     1     CALL RERROR('GTANH','BAD NUMBER',ANG+2.0)                    GTLH0500
      N=N-2                                                             GTLH0510
      ANG=0.0                                                           GTLH0520
C                                                                       GTLH0530
C                                                                       GTLH0540
C     CONVERT 'Z' ARRAY INPUT(OVERLAYS 'C' ARRAY) TO 'C' ARRAY          GTLH0550
C                                                                       GTLH0560
      ZM1=Z0                                                            GTLH0570
      LAM0=PID2-LAM0                                                    GTLH0580
      LMIM1=LAM0                                                        GTLH0590
      NP1=N+1                                                           GTLH0600
      DO 10 I=1,NP1                                                     GTLH0610
         Z=C(I)                                                         GTLH0620
         LMI=PID2-LM(I)                                                 GTLH0630
         LM(I)=LMI                                                      GTLH0640
         ALC(I)=ALCOSH((LMI-LAM0) / DL(I))                              GTLH0650
         C(I)=(Z-ZM1)/(LMI-LMIM1)                                       GTLH0660
         ZM1=Z                                                          GTLH0670
10       LMIM1=LMI                                                      GTLH0680
C                                                                       GTLH0690
      RETURN                                                            GTLH0700
C                                                                       GTLH0710
      ENTRY TOPOG                                                       GTLH0720
C                                                                       GTLH0730
      IF(N.LE.0)                                                        GTLH0740
     1    CALL RERROR('GTANH','BAD N VALUE',FLOAT(N))                   GTLH0750
C                                                                       GTLH0760
      SUM = 0.                                                          GTLH0770
      DO 1 I = 1, N                                                     GTLH0780
    1 SUM = SUM + DL(I) * (C(I + 1) - C(I)) / 2. *(ALCOSH(((TH-LM       GTLH0790
     1(I)) / DL(I))) - ALC(I))                                          GTLH0800
      Z = Z0 - SUM + (C(1) + C(N + 1)) * (TH-LAM0) / 2.                 GTLH0810
      G=R-EARTHR-Z                                                      GTLH0820
      PGR=1.0                                                           GTLH0830
C                                                                       GTLH0840
      PGTH = C(1)                                                       GTLH0850
      DO 2 I = 1, N                                                     GTLH0860
    2 PGTH= PGTH+ (C(I + 1) - C(I)) / 2. * (1. + TANH ((LM(I)- TH) /DL  GTLH0870
     1 (I)))                                                            GTLH0880
      PGTH=-PGTH                                                        GTLH0890
      PGTHTH=0.0                                                        GTLH0900
      DO 3 I=1,N                                                        GTLH0910
3     PGTHTH=PGTHTH+                                                    GTLH0920
     1    (C(I+1)-C(I))/2.*(1.0-TANH((LM(I)-TH)/DL(I))**2)/DL(I)        GTLH0930
C                                                                       GTLH0940
      CALL PTOPOG                                                       GTLH0950
      RETURN                                                            GTLH0960
      END                                                               GTLH0970


      SUBROUTINE NPBOTM                                                 NPRM0020
```

```
C     DO-NOTHING BOTTOM PERTURBATION MODEL                                    NPRM0030
C      COMMON DECK "GG" INSERTED HERE                                         CGG 0020
       REAL MODG                                                              CGG 0040
       COMMON/GG/MODG(4)                                                      CGG 0050
       COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                       CGG 0060
       COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME                 CGG 0070
C                                                                             NPRM0050
C      COMMON DECK "WW" INSERTED HERE                                         CWW 0020
       PARAMETER (NWARSZ=1000)                                                CWW10030
       COMMON/WW/ID(10),MAXW,W(NWARSZ)                                        CWW10040
       REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                     CWW20020
       EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),         CWW20030
      1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),           CWW20040
      2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),               CWW20050
      3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),              CWW20060
      8   (RUNSUP,W(18)),(RCVRH,W(20)),                                       CWW20070
      4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))     CWW20080
      5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                           CWW20090
      6 (HMIN,W(27)),(RGMAX,W(28)),                                           CWW20100
      8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                          CWW20110
      6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),            CWW20120
      7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))             CWW20130
      9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),             CWW20140
      1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                   CWW20150
      2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                        CWW20160
       REAL MMODEL,MFORM,MID                                                  CWW30020
C                                                                             CWW30030
C     WIND         100-124                                                    CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)                 CWW30050
C                                                                             CWW30060
C     DELTA WIND         125-149                                              CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)              CWW30080
C                                                                             CWW30090
C     SOUND SPEED 150-174                                                     CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)                 CWW30110
      EQUIVALENCE (W(153),REFC)                                               CWW30120
C                                                                             CWW30130
C     DELTA SOUND SPEED 175-199                                               CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)              CWW30150
C                                                                             CWW30160
C     TEMPERATURE    200-224                                                  CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)                 CWW30180
C                                                                             CWW30190
C     DELTA TEMPERATURE    225-249                                            CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)              CWW30210
C                                                                             CWW30220
C     MOLECULAR   250-274                                                     CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)                 CWW30240
C                                                                             CWW30250
C     RECEIVER HEIGHT   275-299                                               CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)                 CWW30270
C                                                                             CWW30280
C     TOPOGRAPHY   300-324                                                    CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)                 CWW30300
C                                                                             CWW30310
```

```
C     DELTA TOPOGRAPHY   325-349                                          CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)          CWW30330
C                                                                         CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                            CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)             CWW30360
C                                                                         CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                            CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)          CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                CWW30400
C                                                                         CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                         CWW30420
C     ABSORPTION         500-524                                          CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)             CWW30440
C                                                                         CWW30450
C     DELTA ABSORPTION         525-549                                    CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)          CWW30470
C                                                                         CWW30480
C     PRESSURE         550-574                                            CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)             CWW30500
C                                                                         CWW30510
C     DELTA PRESSURE         575-599                                      CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)          CWW30530
C                                                                         CWW30540
C     COMMON DECK "B10" INSERTED HERE                                     CB9 0020
      INTEGER DGMX,DGNTBL,DGITBL,DGFRMTB,IDSDG(10)                        CB9 0040
      COMMON/B10/DGMX,DGNTBL(10),DGITBL(10),DGFRMTB(10),DGGP(10)          CB9 0050
      EQUIVALENCE (DGGP,IDSDG)                                            CB9 0060
C                                                                         NPRM0080
      INTEGER DJMX     ,DJNTBL(10),DJITBL(10),DJFRMTB(10)                 NPRM0090
C                                                                         NPRM0100
      DATA DJMX/1/                                                        NPRM0110
      DATA DJNTBL/1,11,8*0/                                               NPRM0120
      DATA DJITBL/1,9*0/                                                  NPRM0130
      DATA DJFRMTB/1,9*0/                                                 NPRM0140
C                                                                         NPRM0150
C                                                                         NPRM0160
      ENTRY SETPTP                                                        NPRM0170
C                                                                         NPRM0180
      DGMX=DJMX                                                           NPRM0190
      CALL IMOVE(DGNTBL,DJNTBL,10)                                        NPRM0200
      CALL IMOVE(DGITBL,DJITBL,10)                                        NPRM0210
      CALL IMOVE(DGFRMTB,DJFRMTB,10)                                      NPRM0220
C                                                                         NPRM0230
      RETURN                                                              NPRM0240
C                                                                         NPRM0250
      ENTRY IPTOPOG                                                       NPRM0260
C                                                                         NPRM0270
      MODG(3)=6HNPBOTM                                                    NPRM0280
      MODG(4)=GUID                                                        NPRM0290
      ENTRY PTOPOG                                                        NPRM0300
      END                                                                 NPRM0310
```

```
      SUBROUTINE SHORIZ                                                 SHJZ0020
C     SURFACE MODEL USING FIXED OFFSET TO EARTHR'S SURFACE.             SHJZ0030
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "SS" INSERTED HERE                                    CSS 0020
      REAL MODSURF                                                      CSS 0040
      COMMON/SS/ MODSURF(4)                                             CSS 0050
      COMMON/SS/U,PUR,PURR,PURTH,PURPH                                  CSS 0060
      COMMON/SS/PUTH,PUPH,PUTHTH,PUPHPH,PUTHPH,USELECT,UTIME            CSS 0070
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND         125-149                                        CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
C                                                                       CWW30090
C     SOUND SPEED 150-174                                               CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)           CWW30110
      EQUIVALENCE (W(153),REFC)                                         CWW30120
C                                                                       CWW30130
C     DELTA SOUND SPEED 175-199                                         CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)        CWW30150
C                                                                       CWW30160
C     TEMPERATURE    200-224                                            CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)           CWW30180
C                                                                       CWW30190
C     DELTA TEMPERATURE    225-249                                      CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)        CWW30210
C                                                                       CWW30220
C     MOLECULAR   250-274                                               CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)           CWW30240
C                                                                       CWW30250
C     RECEIVER HEIGHT   275-299                                         CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)           CWW30270
```

```
C                                                                       CWW30280
C     TOPOGRAPHY   300-324                                              CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)           CWW30300
C                                                                       CWW30310
C     DELTA TOPOGRAPHY   325-349                                        CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)        CWW30330
C                                                                       CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                          CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)           CWW30360
C                                                                       CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                          CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)        CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                              CWW30400
C                                                                       CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                       CWW30420
C     ABSORPTION         500-524                                        CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)           CWW30440
C                                                                       CWW30450
C     DELTA ABSORPTION         525-549                                  CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)        CWW30470
C                                                                       CWW30480
C     PRESSURE         550-574                                          CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)           CWW30500
C                                                                       CWW30510
C     DELTA PRESSURE         575-599                                    CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)        CWW30530
C                                                                       CWW30540
C                                                                       SHJZ0070
      EQUIVALENCE (W(353),SHEIGHT)                                      SHJZ0080
C                                                                       SHJZ0090
C     COMMON DECK "CB11" INSERTED HERE                                  CB110020
      INTEGER SMX,SNTBL,SITBL,SFRMTBL,IDSS(10)                          CB110040
      COMMON/B11/SMX,SNTBL(10),SITBL(10),SFRMTBL(10),SGP(11)            CB110050
      EQUIVALENCE (SGP,IDSS),(ANS,SGP(11))                              CB110060
C                                                                       SHJZ0110
      INTEGER SPX     ,SQTBL(10),SLTBL(10),SIRMTBL(10)                  SHJZ0120
      DATA RECOSS/1.0/                                                  SHJZ0130
C                                                                       SHJZ0140
      DATA SPX/1/                                                       SHJZ0150
      DATA SQTBL/1,11,8*0/                                              SHJZ0160
      DATA SLTBL/1,9*0/                                                 SHJZ0170
      DATA SIRMTBL/1,9*0/                                               SHJZ0180
C                                                                       SHJZ0190
      ENTRY SETSUR                                                      SHJZ0200
C                                                                       SHJZ0210
      SMX=SPX                                                           SHJZ0220
      CALL IMOVE(SNTBL,SQTBL,10)                                        SHJZ0230
      CALL IMOVE(SITBL,SLTBL,10)                                        SHJZ0240
      CALL IMOVE(SFRMTBL,SIRMTBL,10)                                    SHJZ0250
      CALL SETPSR                                                       SHJZ0260
C                                                                       SHJZ0270
      RETURN                                                            SHJZ0280
C                                                                       SHJZ0290
      ENTRY ISURFAC                                                     SHJZ0300
C                                                                       SHJZ0310
```

```
      IF(RECOSS .NE. SMODEL)                                            SHJZ0320
     1   CALL RERROR('SURFACE','WRNG MODEL', RECOSS)                    SHJZ0330
      MODSURF(1)=6HSHORIZ                                               SHJZ0340
      MODSURF(2)=SID                                                    SHJZ0350
C     CALL THE PERTURBATION INITIALIZATION                              SHJZ0360
      CALL IPSURF                                                       SHJZ0370
      RETURN                                                            SHJZ0380
C                                                                       SHJZ0390
      ENTRY SURFACE                                                     SHJZ0400
      U=-(R-EARTHR-SHEIGHT)                                             SHJZ0410
      PUR=-1.0                                                          SHJZ0420
      CALL CLEAR(PURR,8)                                                SHJZ0430
C     CALL THE PERTURBATION                                             SHJZ0440
      CALL PSURFCE                                                      SHJZ0450
      END                                                               SHJZ0460
```

```
      SUBROUTINE NPSURF                                                 NPPF0020
C     DO-NOTHING SURFACE PERTURBATION MODEL                             NPPF0030
C     COMMON DECK "SS" INSERTED HERE                                    CSS 0020
      REAL MODSURF                                                      CSS 0040
      COMMON/SS/ MODSURF(4)                                             CSS 0050
      COMMON/SS/U,PUR,PURR,PURTH,PURPH                                  CSS 0060
      COMMON/SS/PUTH,PUPH,PUTHTH,PUPHPH,PUTHPH,USELECT,UTIME            CSS 0070
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND         125-149                                        CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
C                                                                       CWW30090
C     SOUND SPEED 150-174                                               CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)           CWW30110
```

```
      EQUIVALENCE (W(153),REFC)                                          CWW30120
C                                                                        CWW30130
C     DELTA SOUND SPEED 175-199                                          CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)         CWW30150
C                                                                        CWW30160
C     TEMPERATURE    200-224                                             CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)            CWW30180
C                                                                        CWW30190
C     DELTA TEMPERATURE    225-249                                       CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)         CWW30210
C                                                                        CWW30220
C     MOLECULAR   250-274                                                CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)            CWW30240
C                                                                        CWW30250
C     RECEIVER HEIGHT   275-299                                          CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)            CWW30270
C                                                                        CWW30280
C     TOPOGRAPHY   300-324                                               CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)            CWW30300
C                                                                        CWW30310
C     DELTA TOPOGRAPHY   325-349                                         CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)         CWW30330
C                                                                        CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                           CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)            CWW30360
C                                                                        CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                           CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)         CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                               CWW30400
C                                                                        CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                        CWW30420
C     ABSORPTION         500-524                                         CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)            CWW30440
C                                                                        CWW30450
C     DELTA ABSORPTION         525-549                                   CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)         CWW30470
C                                                                        CWW30480
C     PRESSURE         550-574                                           CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)            CWW30500
C                                                                        CWW30510
C     DELTA PRESSURE         575-599                                     CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)         CWW30530
C                                                                        CWW30540
C     COMMON DECK "CB12" INSERTED HERE                                   CB120020
      INTEGER DSMX,DSNTBL,DSITBL,DSFRMTB,IDSDS(10)                       CB120040
      COMMON/B12/DSMX,DSNTBL(10),DSITBL(10),DSFRMTB(10),DSGP(11)         CB120050
      EQUIVALENCE (DSGP,IDSDS),(ANDS,DSGP(11))                           CB120060
      INTEGER DVMX     ,DVNTBL(10),DVITBL(10),DVFRMTB(10)                NPPF0070
      DATA RECOGSU/0.0/                                                  NPPF0080
C                                                                        NPPF0090
      DATA DVMX/1/                                                       NPPF0100
      DATA DVNTBL/1,11,8*0/                                              NPPF0110
      DATA DVITBL/1,9*0/                                                 NPPF0120
      DATA DVFRMTB/1,9*0/                                                NPPF0130
C                                                                        NPPF0140
```

```
      ENTRY SETPSR                                                      NPPF0150
C                                                                       NPPF0160
      DSMX=DVMX                                                         NPPF0170
      CALL IMOVE(DSNTBL,DVNTBL,10)                                      NPPF0180
      CALL IMOVE(DSITBL,DVITBL,10)                                      NPPF0190
      CALL IMOVE(DSFRMTB,DVFRMTB,10)                                    NPPF0200
C                                                                       NPPF0210
      RETURN                                                            NPPF0220
C                                                                       NPPF0230
      ENTRY IPSURF                                                      NPPF0240
      IF(RECOGSU .NE. SUMODEL)                                          NPPF0250
     1    CALL RERROR('DSURF ','WRNG MODEL',RECOGSU)                    NPPF0260
      MODSURF(3)=6HNPSURF                                               NPPF0270
      MODSURF(4)=SUID                                                   NPPF0280
      RETURN                                                            NPPF0290
C                                                                       NPPF0300
      ENTRY PSURFCE                                                     NPPF0310
      RETURN                                                            NPPF0320
      END                                                               NPPF0330
```

```
      SUBROUTINE RHORIZ                                                 RHJZ0020
C     RECEIVER MODEL USING FIXED OFFSET TO EARTHR'S SURFACE.            RHJZ0030
C     COMMON DECK "RR" INSERTED HERE                                    CRR 0020
      REAL MODREC                                                       CRR 0040
      COMMON/RR/ MODREC(4)                                              CRR 0050
      COMMON/RR/F,PFR,PFRR,PFRTH,PFRPH                                  CRR 0060
      COMMON/RR/PFTH,PFPH,PFTHTH,PFPHPH,PFTHPH,FSELECT,FTIME            CRR 0070
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND         125-149                                        CWW30070
```

```
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)          CWW30080
C                                                                         CWW30090
C     SOUND SPEED 150-174                                                 CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)             CWW30110
      EQUIVALENCE (W(153),REFC)                                           CWW30120
C                                                                         CWW30130
C     DELTA SOUND SPEED 175-199                                           CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)          CWW30150
C                                                                         CWW30160
C     TEMPERATURE    200-224                                              CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)             CWW30180
C                                                                         CWW30190
C     DELTA TEMPERATURE    225-249                                        CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)          CWW30210
C                                                                         CWW30220
C     MOLECULAR   250-274                                                 CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)             CWW30240
C                                                                         CWW30250
C     RECEIVER HEIGHT   275-299                                           CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)             CWW30270
C                                                                         CWW30280
C     TOPOGRAPHY   300-324                                                CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)             CWW30300
C                                                                         CWW30310
C     DELTA TOPOGRAPHY   325-349                                          CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)          CWW30330
C                                                                         CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                            CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)             CWW30360
C                                                                         CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                            CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)          CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                                CWW30400
C                                                                         CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                         CWW30420
C     ABSORPTION         500-524                                          CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)             CWW30440
C                                                                         CWW30450
C     DELTA ABSORPTION         525-549                                    CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)          CWW30470
C                                                                         CWW30480
C     PRESSURE         550-574                                            CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)             CWW30500
C                                                                         CWW30510
C     DELTA PRESSURE         575-599                                      CWW30520
      EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)          CWW30530
C                                                                         CWW30540
C     COMMON DECK "RKAM" INSERTED HERE                                    RKAM0020
      REAL KR,KTH,KPH                                                     RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)         RKAM0050
C                                                                         RHJZ0070
C     COMMON DECK "B8" INSERTED HERE                                      CB100020
      INTEGER RMX,RNTBL,RITBL,RFRMTBL,IDSR(10)                            CB100040
      COMMON/B8/RMX,RNTBL(10),RITBL(10),RFRMTBL(10),RGP(10)               CB100050
      EQUIVALENCE (RGP,IDSR)                                              CB100060
```

```
C                                                                       RHJZ0090
      INTEGER RPX     ,RQTBL(10),RLTBL(10),RIRMTBL(10)                  RHJZ0100
      DATA RECORR/1.0/                                                  RHJZ0110
C                                                                       RHJZ0120
      DATA RPX/1/                                                       RHJZ0130
      DATA RQTBL/1,11,8*0/                                              RHJZ0140
      DATA RLTBL/1,9*0/                                                 RHJZ0150
      DATA RIRMTBL/1,9*0/                                               RHJZ0160
C                                                                       RHJZ0170
      ENTRY SETRCV                                                      RHJZ0180
C                                                                       RHJZ0190
      RMX=RPX                                                           RHJZ0200
      CALL IMOVE(RNTBL,RQTBL,10)                                        RHJZ0210
      CALL IMOVE(RITBL,RLTBL,10)                                        RHJZ0220
      CALL IMOVE(RFRMTBL,RIRMTBL,10)                                    RHJZ0230
C                                                                       RHJZ0240
      RETURN                                                            RHJZ0250
C                                                                       RHJZ0260
      ENTRY IRECVR                                                      RHJZ0270
C                                                                       RHJZ0280
      IF(RECORR .NE. RMODEL)                                            RHJZ0290
     1    CALL RERROR('RECEIVR','WRNG MODEL',RECORR)                    RHJZ0300
C                                                                       RHJZ0310
      MODREC(1)=6HRHORIZ                                                RHJZ0320
      MODREC(2)=RID                                                     RHJZ0330
      RETURN                                                            RHJZ0340
C                                                                       RHJZ0350
      ENTRY RECEVER                                                     RHJZ0360
      F=R-W(1)-W(20)                                                    RHJZ0370
      PFR=1.0                                                           RHJZ0380
      CALL CLEAR(PFRR,8)                                                RHJZ0390
      END                                                               RHJZ0400
```

```
      SUBROUTINE RBOTM                                                  RBIM0020
C     RECEIVER MODEL USING FIXED OFFSET TO TERRAIN HEIGHT               RBIM0030
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
       EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),   CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
```

```
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),        CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))         CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),         CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))               CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                    CWW20160
      REAL MMODEL,MFORM,MID                                              CWW30020
C                                                                        CWW30030
C     WIND         100-124                                             CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)            CWW30050
C                                                                        CWW30060
C     DELTA WIND         125-149                                       CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)         CWW30080
C                                                                        CWW30090
C     SOUND SPEED 150-174                                              CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)            CWW30110
      EQUIVALENCE (W(153),REFC)                                          CWW30120
C                                                                        CWW30130
C     DELTA SOUND SPEED 175-199                                        CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)         CWW30150
C                                                                        CWW30160
C     TEMPERATURE    200-224                                           CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)            CWW30180
C                                                                        CWW30190
C     DELTA TEMPERATURE    225-249                                     CWW30200
      EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)         CWW30210
C                                                                        CWW30220
C     MOLECULAR   250-274                                              CWW30230
      EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)            CWW30240
C                                                                        CWW30250
C     RECEIVER HEIGHT   275-299                                        CWW30260
      EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)            CWW30270
C                                                                        CWW30280
C     TOPOGRAPHY   300-324                                             CWW30290
      EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)            CWW30300
C                                                                        CWW30310
C     DELTA TOPOGRAPHY   325-349                                       CWW30320
      EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)         CWW30330
C                                                                        CWW30340
C     ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                         CWW30350
      EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)            CWW30360
C                                                                        CWW30370
C     ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                         CWW30380
      EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)         CWW30390
C     PLOT ENHANCEMENTS CONTROL PARAMETERS                             CWW30400
C                                                                        CWW30410
      EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                        CWW30420
C     ABSORPTION         500-524                                       CWW30430
      EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)            CWW30440
C                                                                        CWW30450
C     DELTA ABSORPTION         525-549                                 CWW30460
      EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)         CWW30470
C                                                                        CWW30480
C     PRESSURE         550-574                                         CWW30490
      EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)            CWW30500
C                                                                        CWW30510
```

```
C       DELTA PRESSURE          575-599                                         CWW30520
        EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)              CWW30530
C                                                                               CWW30540
C       COMMON DECK "RR" INSERTED HERE                                          CRR 0020
        REAL MODREC                                                             CRR 0040
        COMMON/RR/ MODREC(4)                                                    CRR 0050
        COMMON/RR/F,PFR,PFRR,PFRTH,PFRPH                                        CRR 0060
        COMMON/RR/PFTH,PFPH,PFTHTH,PFPHPH,PFTHPH,FSELECT,FTIME                  CRR 0070
C       COMMON DECK "GG" INSERTED HERE                                          CGG 0020
        REAL MODG                                                               CGG 0040
        COMMON/GG/MODG(4)                                                       CGG 0050
        COMMON/GG/G,PGR,PGRR,PGRTH,PGRPH                                        CGG 0060
        COMMON/GG/PGTH,PGPH,PGTHTH,PGPHPH,PGTHPH,GSELECT,GTIME                  CGG 0070
C                                                                               RBIM0080
C       COMMON DECK "B8" INSERTED HERE                                          CB100020
        INTEGER RMX,RNTBL,RITBL,RFRMTBL,IDSR(10)                                CB100040
        COMMON/B8/RMX,RNTBL(10),RITBL(10),RFRMTBL(10),RGP(10)                   CB100050
        EQUIVALENCE (RGP,IDSR)                                                  CB100060
C                                                                               RBIM0100
        INTEGER RPX     ,RQTBL(10),RLTBL(10),RIRMTBL(10)                        RBIM0110
C                                                                               RBIM0120
        DATA RPX/1/                                                             RBIM0130
        DATA RQTBL/1,11,8*0/                                                    RBIM0140
        DATA RLTBL/1,9*0/                                                       RBIM0150
        DATA RIRMTBL/1,9*0/                                                     RBIM0160
C                                                                               RBIM0170
        DATA RECORR/2.0/                                                        RBIM0180
C                                                                               RBIM0190
        ENTRY SETRCV                                                            RBIM0200
C                                                                               RBIM0210
        RMX=RPX                                                                 RBIM0220
        CALL IMOVE(RNTBL,RQTBL,10)                                              RBIM0230
        CALL IMOVE(RITBL,RLTBL,10)                                              RBIM0240
        CALL IMOVE(RFRMTBL,RIRMTBL,10)                                          RBIM0250
C                                                                               RBIM0260
        RETURN                                                                  RBIM0270
C                                                                               RBIM0280
        ENTRY IRECVR                                                            RBIM0290
C                                                                               RBIM0300
        IF(RECORR .NE. RMODEL)                                                  RBIM0310
     1     CALL RERROR('RECEIVR','WRNG MODEL',RECORR)                           RBIM0320
        MODREC(1)=5HRBOTM                                                       RBIM0330
        MODREC(2)=RID                                                           RBIM0340
        RETURN                                                                  RBIM0350
C                                                                               RBIM0360
        ENTRY RECEVER                                                           RBIM0370
C       GET CURRENT TERRAIN HEIGHT(MUST USE GET1 TO AVOID RECURSION             RBIM0380
C       SINCE WE ARE PROBABLY BEING CALLED BY GET RIGHT NOW)                    RBIM0390
        F=GET1(G)-W(20)                                                         RBIM0400
        CALL RMOVE(PFR,PGR,9)                                                   RBIM0410
        END                                                                     RBIM0420
```

```
      SUBROUTINE RVERT                                                  RVRT0020
C     VERTICAL(CONICAL) RECEIVER SURFACE AT A FIXED RADIUS FROM         RVRT0030
C     A SPECIFIED ORIGIN.                                               RVRT0040
C                                                                       RVRT0050
C     COMMON DECK "RKAM" INSERTED HERE                                  RKAM0020
      REAL KR,KTH,KPH                                                   RKAM0040
      COMMON//R,TH,PH,KR,KTH,KPH,RKVARS(14),TPULSE,CSTEP,DRDT(20)       RKAM0050
C     COMMON DECK "RR" INSERTED HERE                                    CRR 0020
      REAL MODREC                                                       CRR 0040
      COMMON/RR/ MODREC(4)                                              CRR 0050
      COMMON/RR/F,PFR,PFRR,PFRTH,PFRPH                                  CRR 0060
      COMMON/RR/PFTH,PFPH,PFTHTH,PFPHPH,PFTHPH,FSELECT,FTIME            CRR 0070
C     COMMON DECK "B8" INSERTED HERE                                    CB100020
      INTEGER RMX,RNTBL,RITBL,RFRMTBL,IDSR(10)                          CB100040
      COMMON/B8/RMX,RNTBL(10),RITBL(10),RFRMTBL(10),RGP(10)             CB100050
      EQUIVALENCE (RGP,IDSR)                                            CB100060
C     COMMON DECK "GAMANG" INSERTED HERE                                CGAM0020
      COMMON/SPHGAM/SINLM0,COSLM0,GPH0,COSPHD,SINTH,COSTH               CGAM0040
      COMMON/SPHGAM/GAMFUN,PGMTH,PGMPH,PGMTHTH,PGMPHPH,PGMTHPH          CGAM0050
C     COMMON DECK "WW" INSERTED HERE                                    CWW 0020
      PARAMETER (NWARSZ=1000)                                           CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                   CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),    CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),      CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),          CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),         CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                 CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                      CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                      CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                     CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),       CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))        CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),        CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))              CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                   CWW20160
      REAL MMODEL,MFORM,MID                                             CWW30020
C                                                                       CWW30030
C     WIND         100-124                                            CWW30040
      EQUIVALENCE (W(100),UMODEL),(W(101),UFORM),(W(102),UID)           CWW30050
C                                                                       CWW30060
C     DELTA WIND         125-149                                        CWW30070
      EQUIVALENCE (W(125),DUMODEL),(W(126),DUFORM),(W(127),DUID)        CWW30080
C                                                                       CWW30090
C     SOUND SPEED 150-174                                               CWW30100
      EQUIVALENCE (W(150),CMODEL),(W(151),CFORM),(W(152),CID)           CWW30110
      EQUIVALENCE (W(153),REFC)                                         CWW30120
C                                                                       CWW30130
C     DELTA SOUND SPEED 175-199                                         CWW30140
      EQUIVALENCE (W(175),DCMODEL),(W(176),DCFORM),(W(177),DCID)        CWW30150
C                                                                       CWW30160
C     TEMPERATURE    200-224                                            CWW30170
      EQUIVALENCE (W(200),TMODEL),(W(201),TFORM),(W(202),TID)           CWW30180
```

```
C                                                                       CWW30190
C       DELTA TEMPERATURE    225-249                                    CWW30200
        EQUIVALENCE (W(225),DTMODEL),(W(226),DTFORM),(W(227),DTID)      CWW30210
C                                                                       CWW30220
C       MOLECULAR   250-274                                             CWW30230
        EQUIVALENCE (W(250),MMODEL),(W(251),MFORM),(W(252),MID)         CWW30240
C                                                                       CWW30250
C       RECEIVER HEIGHT   275-299                                       CWW30260
        EQUIVALENCE (W(275),RMODEL),(W(276),RFORM),(W(277),RID)         CWW30270
C                                                                       CWW30280
C       TOPOGRAPHY   300-324                                            CWW30290
        EQUIVALENCE (W(300),GMODEL),(W(301),GFORM),(W(302),GID)         CWW30300
C                                                                       CWW30310
C       DELTA TOPOGRAPHY   325-349                                      CWW30320
        EQUIVALENCE (W(325),GUMODEL),(W(326),GUFORM),(W(327),GUID)      CWW30330
C                                                                       CWW30340
C       ATMOSPHERIC SURFACE TOPOGRAPHY   350-374                        CWW30350
        EQUIVALENCE (W(350),SMODEL),(W(351),SFORM),(W(352),SID)         CWW30360
C                                                                       CWW30370
C       ATMOSPHERIC SURFACE TOPOGRAPHY   375-399                        CWW30380
        EQUIVALENCE (W(375),SUMODEL),(W(376),SUFORM),(W(377),SUID)      CWW30390
C       PLOT ENHANCEMENTS CONTROL PARAMETERS                            CWW30400
C                                                                       CWW30410
        EQUIVALENCE (W(490),XFQMDL),(W(491),YFQMDL)                     CWW30420
C       ABSORPTION        500-524                                       CWW30430
        EQUIVALENCE (W(500),AMODEL),(W(501),AFORM),(W(502),AID)         CWW30440
C                                                                       CWW30450
C       DELTA ABSORPTION        525-549                                 CWW30460
        EQUIVALENCE (W(525),DAMODEL),(W(526),DAFORM),(W(527),DAID)      CWW30470
C                                                                       CWW30480
C       PRESSURE         550-574                                        CWW30490
        EQUIVALENCE (W(550),PMODEL),(W(551),PFORM),(W(552),PID)         CWW30500
C                                                                       CWW30510
C       DELTA PRESSURE         575-599                                  CWW30520
        EQUIVALENCE (W(575),DPMODEL),(W(576),DPFORM),(W(577),DPID)      CWW30530
C                                                                       CWW30540
        EQUIVALENCE (RVALPH0,W(278)) ,(RVLAMZ,W(279)), (RVPH0,W(280))   RVRT0110
C                                                                       RVRT0120
        INTEGER RPX    ,RQTBL(10),RLTBL(10),RIRMTBL(10)                 RVRT0130
C                                                                       RVRT0140
        DATA RPX/1/                                                     RVRT0150
        DATA RQTBL/1,11,8*0/                                            RVRT0160
        DATA RLTBL/1,9*0/                                               RVRT0170
        DATA RIRMTBL/1,9*0/                                             RVRT0180
        DATA RECORR/1.0/                                                RVRT0190
C                                                                       RVRT0200
        ENTRY SETRCV                                                    RVRT0210
C                                                                       RVRT0220
        RMX=RPX                                                         RVRT0230
        CALL IMOVE(RNTBL,RQTBL,10)                                      RVRT0240
        CALL IMOVE(RITBL,RLTBL,10)                                      RVRT0250
        CALL IMOVE(RFRMTBL,RIRMTBL,10)                                  RVRT0260
C                                                                       RVRT0270
        RETURN                                                          RVRT0280
C                                                                       RVRT0290
```

```
      ENTRY IRECVR                                                          RVRT0300
C                                                                           RVRT0310
      IF(RECORR .NE. RMODEL)                                                RVRT0320
     1    CALL RERROR(7HRECEIVR  ,10HWRNG MODEL ,RECORR)                   RVRT0330
      MODREC(1)=7HRVERT                                                     RVRT0340
      MODREC(2)=RID                                                         RVRT0350
C                                                                           RVRT0360
      SINLMZ=SIN(RVLAMZ)                                                    RVRT0370
      COSLMZ=COS(RVLAMZ)                                                    RVRT0380
      COSALP=COS(RVALPH0)                                                   RVRT0390
C                                                                           RVRT0400
      RETURN                                                                RVRT0410
C                                                                           RVRT0420
      ENTRY RECEVER                                                         RVRT0430
C                                                                           RVRT0440
      SINLM0=SINLMZ                                                         RVRT0450
      COSLM0=COSLMZ                                                         RVRT0460
      GPH0=RVPH0                                                            RVRT0470
C                                                                           RVRT0480
      CALL GAMANG(TH,PH)                                                    RVRT0490
C                                                                           RVRT0500
      F=GAMFUN-COSALP                                                       RVRT0510
      PFR=0.0                                                               RVRT0520
      CALL RMOVE(PFTH,PGMTH,5)                                              RVRT0530
C                                                                           RVRT0540
      RETURN                                                                RVRT0550
      END                                                                   RVRT0560
```

```
      SUBROUTINE SMPANN                                                     SMQN0020
C     ANNOTATION MODEL FOR MINIMUM GRAPHICS SUPPORT                         SMQN0030
C                                                                           SMQN0040
      CHARACTER*(*) S,C                                                     SMQN0050
C                                                                           SMQN0060
C     INITIALIZES PLOT IN DRAFT MODE(DOES NOT REQUIRE DISSPLA)              SMQN0070
C                                                                           SMQN0080
C     COMMON DECK "WWR" INSERTED HERE                                       CWWR0020
      PARAMETER (NWARSZ=1000)                                               CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                       CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                    CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),        CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),          CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),              CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),             CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                     CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))    CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                          CWW20090
     6 (HMIN,W(27)),(RGMAX,W(28)),                                          CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                         CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),           CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))            CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),            CWW20140
```

```
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                     CWW20160
C     COMMON DECK "ANNOT" INSERTED HERE                                   ANNO0020
      CHARACTER*10 ANOTES,HNOTES                                          ANNO0040
      COMMON/ANNCTL/LENA(4),LENHA(3)                                      ANNO0050
      COMMON/ANNCTC/ANOTES(2,4),HNOTES(4,3)                               ANNO0060
C                                                                         SMQN0110
      INTEGER LHNA(4) ,LHNHA(3)                                           SMQN0120
      CHARACTER*10 AQOTES(2,4),HQOTES(4,3)                                SMQN0130
C                                                                         SMQN0140
      DATA LHNA,AQOTES/2*1,2*2,'DEPTH (M)',' ','DEPTH (KM)',' ',          SMQN0150
     2 'HEIGHT (M)',' ','HEIGHT (KM',')'/                                 SMQN0160
      DATA LHNHA,HQOTES/3,2,3, 'RANGE AT S','EA LEVEL (','KM)',' '        SMQN0170
     2       ,'RANGE (DEG',')',2*' '                                      SMQN0180
     3       ,'CROSS RANG','E AT SEA L','EVEL (KM)',' '/                  SMQN0190
C                                                                         SMQN0200
      ENTRY SETANN                                                        SMQN0210
C                                                                         SMQN0220
      CALL IMOVE(LENA,LHNA,4)                                             SMQN0230
      CALL IMOVE(LENHA,LHNHA,3)                                           SMQN0240
      DO 10 I=1,2                                                         SMQN0250
      DO 10 J=1,4                                                         SMQN0260
10    ANOTES(I,J)=AQOTES(I,J)                                             SMQN0270
      DO 20 I=1,4                                                         SMQN0280
      DO 20 J=1,3                                                         SMQN0290
20    HNOTES(I,J)=HQOTES(I,J)                                             SMQN0300
C                                                                         SMQN0310
      RETURN                                                              SMQN0320
C                                                                         SMQN0330
      ENTRY ANNFIL(S,C)                                                   SMQN0340
      CALL SFILTR(S,C,'#!')                                               SMQN0350
      END                                                                 SMQN0360


      SUBROUTINE FULANN                                                   FUPN0020
C     ANNOTATION MODEL SUITED FOR PUBLICATION QUALITY LETTERING           FUPN0030
C                                                                         FUPN0040
      CHARACTER*(*) S,C                                                   FUPN0050
C                                                                         FUPN0060
C     INITIALIZES PLOT IN PUBLICATION-QUALITY MODE(REQUIRES DISSPLA)      FUPN0070
C                                                                         FUPN0080
C     COMMON DECK "WWR" INSERTED HERE                                     CWWR0020
      PARAMETER (NWARSZ=1000)                                             CWW10030
      COMMON/WW/ID(10),MAXW,W(NWARSZ)                                     CWW10040
      REAL MAXSTP,MAXERR,INTYP,LLAT,LLON                                  CWW20020
      EQUIVALENCE (EARTHR,W(1)),(RAY,W(2)),(XMTRH,W(3)),(TLAT,W(4)),      CWW20030
     1 (TLON,W(5)),(OW,W(6)),(FBEG,W(7)),(FEND,W(8)),(FSTEP,W(9)),        CWW20040
     2 (AZ1,W(10)),(AZBEG,W(11)),(AZEND,W(12)),(AZSTEP,W(13)),            CWW20050
     3 (BETA,W(14)),(ELBEG,W(15)),(ELEND,W(16)),(ELSTEP,W(17)),           CWW20060
     8   (RUNSUP,W(18)),(RCVRH,W(20)),                                   CWW20070
     4 (ONLY,W(21)),(HOP,W(22)),(MAXSTP,W(23)),(PLAT,W(24)),(PLON,W(25))CWW20080
     5,(HMAX,W(26)),(RAYFNC,W(29)),(EXTINC,W(33)),                        CWW20090
```

```
     6 (HMIN,W(27)),(RGMAX,W(28)),                                          CWW20100
     8 (INTYP,W(41)),(MAXERR,W(42)),(ERATIO,W(43)),                         CWW20110
     6 (STEP1,W(44)),(STPMAX,W(45)),(STPMIN,W(46)),(FACTR,W(47)),           CWW20120
     7 (SKIP,W(71)),(RAYSET,W(72)),(PRTSRP,W(74)),(HITLET,W(75))            CWW20130
     9 ,(BINRAY,W(76)),(PAGLN,W(77)),(PLT,W(81)),(PFACTR,W(82)),            CWW20140
     1 (LLAT,W(83)),(LLON,W(84)),(RLAT,W(85)),(RLON,W(86))                  CWW20150
     2,(TIC,W(87)),(HB,W(88)),(HT,W(89)),(TICV,W(96))                       CWW20160
C     COMMON DECK "ANNOT" INSERTED HERE                                     ANNO0020
      CHARACTER*10 ANOTES,HNOTES                                            ANNO0040
      COMMON/ANNCTL/LENA(4),LENHA(3)                                        ANNO0050
      COMMON/ANNCTC/ANOTES(2,4),HNOTES(4,3)                                 ANNO0060
C                                                                           FUPN0110
      INTEGER LHNA(4) ,LHNHA(3)                                             FUPN0120
      CHARACTER*10 AQOTES(2,4),HQOTES(4,3)                                  FUPN0130
C                                                                           FUPN0140
      DATA LHNA,AQOTES/4*2,'DEPTH (#M!',')','DEPTH (#KM','!)',              FUPN0150
     2'HEIGHT (#M','!)','HEIGHT (#K','M!)'/                                 FUPN0160
      DATA LHNHA,HQOTES/3,2,4, 'RANGE AT S','EA LEVEL (','#KM!)',' '        FUPN0170
     2       ,'RANGE (#DE','G!)',2*' '                                      FUPN0180
     3       ,'CROSS RANG','E AT SEA L','EVEL (#KM!',')'/                   FUPN0190
      ENTRY SETANN                                                          FUPN0200
C                                                                           FUPN0210
      PRINT *,'FULL ANNOTATION MODEL'                                       FUPN0220
C                                                                           FUPN0230
      CALL SCMPLX                                                           FUPN0240
      CALL MX1ALF('STAND','!')                                              FUPN0250
      CALL MX2ALF('L/CSTD','#')                                             FUPN0260
      CALL HEIGHT(HITLET)                                                   FUPN0270
C                                                                           FUPN0280
      CALL IMOVE(LENA,LHNA,4)                                               FUPN0290
      CALL IMOVE(LENHA,LHNHA,3)                                             FUPN0300
      DO 10 I=1,2                                                           FUPN0310
      DO 10 J=1,4                                                           FUPN0320
10    ANOTES(I,J)=AQOTES(I,J)                                               FUPN0330
      DO 20 I=1,4                                                           FUPN0340
      DO 20 J=1,3                                                           FUPN0350
20    HNOTES(I,J)=HQOTES(I,J)                                               FUPN0360
C                                                                           FUPN0370
      RETURN                                                                FUPN0380
C                                                                           FUPN0390
      ENTRY ANNFIL(S,C)                                                     FUPN0400
      C=S                                                                   FUPN0410
      END                                                                   FUPN0420
```

D.4 TAPRD -- Graphics file read routine (Tape File 6)

```
      PROGRAM TAPRD                                                      TAHD0030
C     PROGRAM TO READ GRAPHICS OUTPUT FILE AND CALL GRAPHICS             TAHD0040
C     INTERFACE ROUTINES.                                                TAHD0050
      COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                                TAHD0060
      COMMON/SUPNEG/IDEL,NMBS                                            TAHD0070
C                                                                        TAHD0080
      PARAMETER (LIMPTS=700,IUN=4)                                       TAHD0090
      REAL XV(LIMPTS),YV(LIMPTS)                                         TAHD0100
      CHARACTER*40 A,C,E                                                 TAHD0110
      LOGICAL COMPCL                                                     TAHD0120
      CHARACTER LINE*72,TEXT*80,S*10                                     TAHD0130
      EQUIVALENCE (IX,XXX),(IY,YYY)                                      TAHD0140
      DATA COMPCL/.TRUE./                                                TAHD0150
      DATA KNT/0/                                                        TAHD0160
C                                                                        TAHD0170
      OPEN(UNIT=IUN,FILE='TAPE5',FORM='UNFORMATTED')                     TAHD0180
      REWIND IUN                                                         TAHD0190
10    READ(IUN,END=100,ERR=100) IT,IX,IY                                 TAHD0200
C     PRINT *,IT,XXX,YYY                                                 TAHD0210
      IF(IT.GT.20) THEN                                                  TAHD0220
         IF(COMPCL) THEN                                                 TAHD0230
            PRINT '(A,3(I4,1X),2G13.6)',                                 TAHD0240
     1         'NO CALL TO ''COMPRS'' BEFORE---',IT,IX,IY,XXX,YYY        TAHD0250
            STOP                                                         TAHD0260
            ENDIF                                                        TAHD0270
      ENDIF                                                              TAHD0280
      IF(IT.LT.-2 .OR. IT.GT.39) STOP 'CODE>39'                          TAHD0290
      KNT=KNT+1                                                          TAHD0300
      IF(IT.EQ.-1) THEN                                                  TAHD0310
         CALL DDEND                                                      TAHD0320
ELSEIF(IT.EQ.-2) THEN                                                    TAHD0330
         CALL DDFR                                                       TAHD0340
ELSEIF(IT.EQ.0) THEN                                                     TAHD0350
         READ(IUN) N,M,(TEXT(I:I),I=1,M)                                 TAHD0360
         CALL DDINIT(N,TEXT)                                             TAHD0370
         COMPCL=.FALSE.                                                  TAHD0380
ELSEIF(IT.EQ.1) THEN                                                     TAHD0390
         CALL DDBP                                                       TAHD0400
ELSEIF(IT.EQ.2) THEN                                                     TAHD0410
         CALL DDVC                                                       TAHD0420
ELSEIF(IT.EQ.10) THEN                                                    TAHD0430
         CALL SCMPLX                                                     TAHD0440
ELSEIF(IT.EQ.11) THEN                                                    TAHD0450
         CALL MX1ALF(IX,IY)                                              TAHD0460
ELSEIF(IT.EQ.12) THEN                                                    TAHD0470
         CALL MX2ALF(IX,IY)                                              TAHD0480
      ELSEIF(IT.EQ.13) THEN                                              TAHD0490
         IF(XXX.LE.0.0) THEN                                             TAHD0500
            PRINT *,'HEIGHT OF ZERO!!'                                   TAHD0510
            XXX=.15                                                      TAHD0520
         ENDIF                                                           TAHD0530
         CALL HEIGHT(XXX)                                                TAHD0540
C                                                                        TAHD0550
      ELSEIF(IT.EQ.20) THEN                                              TAHD0560
         COMPCL=.FALSE.                                                  TAHD0570
```

```
   CALL COMPRS                                                      TAHD0580
ELSEIF(IT.EQ.21) THEN                                               TAHD0590
   CALL GRACE(IX,IY)                                                TAHD0600
ELSEIF(IT.EQ.22) THEN                                               TAHD0610
   CALL PHYSOR(IX,IY)                                               TAHD0620
ELSEIF(IT.EQ.23) THEN                                               TAHD0630
   CALL PAGE(IX,IY)                                                 TAHD0640
ELSEIF(IT.EQ.24) THEN                                               TAHD0650
   CALL SCLPIC(IX)                                                  TAHD0660
ELSEIF(IT.EQ.25) THEN                                               TAHD0670
   IF(IX.NE.0 .OR. IY.NE.0) STOP 'ERROR 1'                          TAHD0680
   READ(IUN) A,B,C,D,E,F,XAXIS,YAXIS                                TAHD0690
   CALL XREVTK                                                      TAHD0700
   CALL YREVTK                                                      TAHD0710
   CALL INTAXS                                                      TAHD0720
   CALL TITLE(A,B,C,D,E,F,XAXIS,YAXIS)                              TAHD0730
ELSEIF(IT.EQ.26) THEN                                               TAHD0740
   CALL FRAME                                                       TAHD0750
ELSEIF(IT.EQ.27) THEN                                               TAHD0760
   READ(IUN) W,X,Y,Z                                                TAHD0770
   CALL GRAF(IX,IY,W,X,Y,Z,XAXIS,YAXIS)                             TAHD0780
ELSEIF(IT.EQ.28) THEN                                               TAHD0790
   CALL MARKER(IX)                                                  TAHD0800
ELSEIF(IT.EQ.29) THEN                                               TAHD0810
   IF(IX.GT.LIMPTS) CALL SYSTEM(52,'N>LIMPTS')                      TAHD0820
   READ(IUN) (XV(I),I=1,IX),(YV(I),I=1,IX)                          TAHD0830
   CALL CURVE(XV,YV,IX,IY)                                          TAHD0840
ELSEIF(IT.EQ.30) THEN                                               TAHD0850
   CALL ENDPL(IX)                                                   TAHD0860
ELSEIF(IT.EQ.31) THEN                                               TAHD0870
   CALL DONEPL                                                      TAHD0880
ELSEIF(IT.EQ.32) THEN                                               TAHD0890
   CALL XTICKS(IX)                                                  TAHD0900
ELSEIF(IT.EQ.33) THEN                                               TAHD0910
   CALL YTICKS(IX)                                                  TAHD0920
ELSEIF(IT.EQ.34) THEN                                               TAHD0930
   CALL MYJACT(IX)                                                  TAHD0940
ELSEIF(IT.EQ.35) THEN                                               TAHD0950
   CALL MYJACT('NUMBERS')                                           TAHD0960
   IDEL=IX                                                          TAHD0970
   NMBS=IY                                                          TAHD0980
ELSEIF(IT.EQ.36) THEN                                               TAHD0990
   CALL NOBRDR                                                      TAHD1000
ELSEIF(IT.EQ.37) THEN                                               TAHD1010
   CALL DASH                                                        TAHD1020
ELSEIF(IT.EQ.38) THEN                                               TAHD1030
   READ(IUN) S                                                      TAHD1040
   CALL RESET(S)                                                    TAHD1050
ELSEIF(IT.EQ.39) THEN                                               TAHD1060
   READ(IUN) W,X,Y,Z                                                TAHD1070
   CALL GRAFB(IX,IY,W,X,Y,Z,XAXIS,YAXIS)                            TAHD1080
ENDIF                                                               TAHD1090
C                                                                   TAHD1100
IF(IT.NE.3) GO TO 10                                                TAHD1110
READ(IUN) IOR,N,M,(TEXT(I:I),I=1,M)                                 TAHD1120
```

```
      CALL DDTEXT(N,TEXT)                                                TAHD1130
C      PRINT 20,(TEXT(I),I=1,N)                                          TAHD1140
20    FORMAT(8A10)                                                       TAHD1150
      GO TO 10                                                           TAHD1160
C                                                                        TAHD1170
100   PRINT *,'NUMBER OF VECTORS=',KNT,IT                                TAHD1180
      STOP                                                               TAHD1190
      END                                                                TAHD1200
```

D.5 DDSPLA -- DISSPLA Interface Routines (Tape File 7)

```
      SUBROUTINE MYJSUB(IPAR,ITY,IMYJ)                                  DDLA0030
C     DSSPLA USER ROUTINE TO HANDLE SPECIAL ANNOTATIONS                 DDLA0040
      COMMON/SUPNEG/IDEL,NMBS                                           DDLA0050
      DATA IDEL,NMBS/0,100000/                                          DDLA0060
C                                                                       DDLA0070
C  WE ARE INTERESTED ONLY IN NUMERICAL VALUES CAUSED BY                 DDLA0080
C  A CALL MYJACT('NUMBERS')(IMYJ=5)                                     DDLA0090
C                                                                       DDLA0100
      IF(ITY.GE.0 .OR. IMYJ.NE.5) RETURN                                DDLA0110
C                                                                       DDLA0120
      IF(IDEL.GT.0) IDEL=IDEL-1                                         DDLA0130
C                                                                       DDLA0140
      IF(IDEL.LE.0.AND.NMBS.GT.0) THEN                                  DDLA0150
         NMBS=NMBS-1                                                    DDLA0160
         IPAR=IABS(IPAR)                                                DDLA0170
      ENDIF                                                             DDLA0180
      END                                                               DDLA0190


      SUBROUTINE DDINIT(N,TEXT)                                         DDLA0200
C     INITIALIZE PLOTTING PROCESS                                       DDLA0210
C                                                                       DDLA0220
      COMMON/PLOTCN/NPLOT,INABLE,FX,FY,OFFX,OFFY                        CPLO0020
      DATA OFFX,OFFY/0.0,0.0/                                           DDLA0240
      DATA NPLOT,INABLE,PLOTSZ,XAXIS,YAXIS/0,0,7.5,11.,8.5/             DDLA0250
C                                                                       DDLA0260
C     NO RE-INITIALIZATIONS BEFORE ENDPL'S                              DDLA0270
      IF(INABLE.GT.0) RETURN                                            DDLA0280
C                                                                       DDLA0290
      IF(NPLOT.GT.0) GO TO 10                                           DDLA0300
C                                                                       DDLA0310
       CALL COMPRS                                                      DDLA0320
      FY=PLOTSZ/1024.                                                   DDLA0330
      FX=FY                                                             DDLA0340
C                                                                       DDLA0350
10    NPLOT=NPLOT+1                                                     DDLA0360
      INABLE=1                                                          DDLA0370
      CALL NOBRDR                                                       DDLA0380
      CALL PAGE(XAXIS,YAXIS)                                            DDLA0390
      CALL PHYSOR(0.0,0.0)                                              DDLA0400
      CALL AREA2D(XAXIS,YAXIS)                                          DDLA0410
      CALL GRACE(0.0)                                                   DDLA0420
      RETURN                                                            DDLA0430
      END                                                               DDLA0440


      SUBROUTINE DDBP                                                   DDLA0450
C     DEFINE A VECTOR ORIGIN AT IX,IY                                   DDLA0460
      COMMON/PLOTCN/NPLOT,INABLE,FX,FY,OFFX,OFFY                        CPLO0020
```

```
      COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                                CDDC0020
                                                                         DDLA0490
C
C     MAKE IX AND IY REAL ONLY FOR DDBP AND DDVC                         DDLA0500
      REAL IX,IY                                                         DDLA0510
C     "DDPLOT" DOES NOT REQUIRE RE-INITIALIZATION AFTER EACH FRAME       DDLA0520
C     BUT "DISSPLA" DOES SO WE USE THE STATUS OF "INABLE" TO TELL US     DDLA0530
C     WHERE WE ARE.                                                      DDLA0540
      IF(INABLE.EQ.0) CALL DDINIT(-1,0)                                  DDLA0550
      CALL STRTPT(OFFX+IX*FX,OFFY+IY*FY)                                 DDLA0560
      RETURN                                                             DDLA0570
      END                                                                DDLA0580
```

```
      SUBROUTINE DDVC                                                    DDLA0590
C     PLOT A STRAIGHT LINE WITH INTENSITY                                DDLA0600
      COMMON/PLOTCN/NPLOT,INABLE,FX,FY,OFFX,OFFY                         CPLO0020
      COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                                CDDC0020
                                                                         DDLA0630
C
C     MAKE IX AND IY REAL ONLY FOR DDBP AND DDVC                         DDLA0640
      REAL IX,IY                                                         DDLA0650
C     SEE "DDBP" FOR THE REASON FOR THIS TEST.                           DDLA0660
      IF(INABLE.EQ.0) CALL DDINIT(-1,0)                                  DDLA0670
      CALL CONNPT(OFFX+IX*FX,OFFY+IY*FY)                                 DDLA0680
      RETURN                                                             DDLA0690
      END                                                                DDLA0700
```

```
      SUBROUTINE DDEND                                                   DDLA0710
C     EMPTY THE PLOT BUFFER AND RELEASE                                  DDLA0720
      COMMON/PLOTCN/NPLOT,INABLE,FX,FY,OFFX,OFFY                         CPLO0020
C     CHECK SYNCH, DDFR SHOULD HAVE BEEN CALLED BY NOW                   DDLA0740
      IF(INABLE.GT.0) CALL ENDPL(0)                                      DDLA0750
                                                                         DDLA0760
C
      CALL DONEPL                                                        DDLA0770
                                                                         DDLA0780
C
      INABLE=0                                                           DDLA0790
      RETURN                                                             DDLA0800
      END                                                                DDLA0810
```

```
      SUBROUTINE DDTEXT(N,TEXT)                                          DDLA0820
C     PLOT A GIVEN ARRAY IN A TABULAR MODE                               DDLA0830
      COMMON/PLOTCN/NPLOT,INABLE,FX,FY,OFFX,OFFY                         CPLO0020
      COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                                CDDC0020
                                                                         DDLA0860
C
      IF(INABLE.EQ.0) CALL DDINIT(-1,0)                                  DDLA0870
```

```
      IF(IOR.EQ.0) CALL ANGLE(0.0)                                      DDLA0880
      IF(IOR.NE.0) CALL ANGLE(90.0)                                     DDLA0890
      CALL MESSAG(TEXT,N*10,OFFX+IX*FX,OFFY+IY*FY)                      DDLA0900
      RETURN                                                            DDLA0910
      END                                                               DDLA0920


      SUBROUTINE DDTAB                                                  DDLA0930
C     INITIALIZE TABULAR TEXT PLOTTING                                  DDLA0940
      RETURN                                                            DDLA0950
      END                                                               DDLA0960


      SUBROUTINE DDFR                                                   DDLA0970
C     ADVANCE ONE PLOTTING FRAME, WHEN PLOTTING IS COMPLETED.           DDLA0980
      COMMON/PLOTCN/NPLOT,INABLE,FX,FY,OFFX,OFFY                        CPLO0020
      IF(INABLE.GT.0) CALL ENDPL(0)                                     DDLA1000
      INABLE=0                                                          DDLA1010
      RETURN                                                            DDLA1020
      END                                                               DDLA1030


      SUBROUTINE GRAFB(XORIG,XSTP,XMAX,YORIG,YSTP,YMAX,XAXIS,YAXIS)     DDLA1040
C     SPECIAL VERSION OF GRAF ROUTINE TO ENCLOSE PLOT AREA              DDLA1050
C     IN BOX.                                                           DDLA1060
C                                                                       DDLA1070
      CALL GRAF(XORIG,XSTP,XMAX,YORIG,YSTP,YMAX)                        DDLA1080
      CALL XNONUM                                                       DDLA1090
      CALL YNONUM                                                       DDLA1100
      CALL XGRAXS(XORIG,XSTP,XMAX,XAXIS,' ',-1,0.0,YAXIS)               DDLA1110
      XR=XMAX-XORIG                                                     DDLA1120
      XAX=AINT(XR/XSTP)*XSTP*(XAXIS/XR)                                 DDLA1130
      XAX=XAXIS                                                         DDLA1140
      PRINT *,XORIG,XMAX,XSTP,XAXIS,XAX                                 DDLA1150
      CALL YGRAXS(YORIG,YSTP,YMAX,YAXIS,' ',-1,XAX,0.0)                 DDLA1160
      CALL RESET('XNONUM')                                              DDLA1170
      CALL RESET('YNONUM')                                              DDLA1180
      END                                                               DDLA1190


      SUBROUTINE TITLEW(A,B,C,D,E,F,G,H)                                DDLA1200
C     DUMMY ROUTINE ALLOWING SUBSTITUTION OF ALTERNATE TITLE PROGRAMS   DDLA1210
      CALL TITLE(A,B,C,D,E,F,G,H)                                       DDLA1220
```

```
END                                                          DDLA1230
```

D.6 DDALT -- Skeleton Graphics Interface Routines (Tape File 8)

```
      SUBROUTINE DDINIT(N,ID)                                          DDKT0030
C     INSERT YOUR OWN ROUTINE TO INITIALIZE PLOTTING PROCESS           DDKT0040
C     ID IS A HOLLERITH STRING OF CHARACTERS IDENTIFYING THE PERSON    DDKT0050
C     GETTING THE PLOT, PHONE NUMBER, ETC.                             DDKT0060
C     N IS THE NUMBER OF CHARACTERS IN THE STRING "ID"                 DDKT0070
      RETURN                                                           DDKT0080
      END                                                              DDKT0090


      SUBROUTINE DDBP                                                  DDKT0100
      COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                              DDKT0110
C     INSERT YOUR OWN ROUTINE TO DEFINE A VECTOR ORIGIN AT IX,IY       DDKT0120
      RETURN                                                           DDKT0130
      END                                                              DDKT0140


      SUBROUTINE DDVC                                                  DDKT0150
      COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                              DDKT0160
C     INSERT YOUR OWN ROUTINE TO PLOT A STRAIGHT LINE WITH INTENSITY   DDKT0170
C     "IN" FROM THE ORIGIN TO THE END POSITION IX,IY.  A SINGLE CALL   DDKT0180
C     TO DDBP FOLLOWED BY SUCCESSIVE CALLS TO DDVC (CHANGING IX,IY)    DDKT0190
C     PLOTS CONNECTED VECTORS.                                         DDKT0200
      RETURN                                                           DDKT0210
      END                                                              DDKT0220


      SUBROUTINE DDEND                                                 DDKT0230
C     CHECK SYNCH, DDFR SHOULD HAVE BEEN CALLED BY NOW                 DDKT0240
C     INSERT YOUR OWN ROUTINE TO EMPTY THE PLOT BUFFER AND RELEASE     DDKT0250
C     THE PLOTTING COMMAND FILE TO YOUR PLOTTING DEVICE.               DDKT0260
      RETURN                                                           DDKT0270
      END                                                              DDKT0280


      SUBROUTINE DDTEXT(N,NT)                                          DDKT0290
      COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                              DDKT0300
C     INSERT YOUR OWN ROUTINE TO PLOT A GIVEN ARRAY IN A TABULAR MODE  DDKT0310
C     AFTER INITIALIZING TABULAR PLOTTING WITH DDTAB. NT IS AN ARRAY OF DDKT0320
C     LENGTH N, CONTAINING "TEXT" FOR TABULAR PLOTTING. SEE APPENDIX C. DDKT0330
      RETURN                                                           DDKT0340
      END                                                              DDKT0350
```

```
      SUBROUTINE DDTAB                                                  DDKT0360
      COMMON/DD/IN,IOR,IT,IS,IC,ICC,IX,IY                               DDKT0370
C     INSERT YOUR OWN ROUTINE TO INITIALIZE TABULAR TEXT PLOTTING       DDKT0380
C        SPECIFY IOR,IS,IX,IY.  TEXT WILL BEGIN AT IX,IY.               DDKT0390
      RETURN                                                            DDKT0400
      END                                                               DDKT0410


      SUBROUTINE DDFR                                                   DDKT0420
C     INSERT YOUR OWN ROUTINE TO ADVANCE ONE PLOTTING FRAME, WHEN       DDKT0430
C     PLOTTING IS COMPLETED.                                            DDKT0440
      RETURN                                                            DDKT0450
      END                                                               DDKT0460


      SUBROUTINE DASH                                                   DDKT0470
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0480
      STOP 'DASH SHOULD NOT HAVE BEEN CALLED'                           DDKT0490
      END                                                               DDKT0500


      SUBROUTINE RESET                                                  DDKT0510
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0520
      STOP 'RESET SHOULD NOT HAVE BEEN CALLED'                          DDKT0530
      END                                                               DDKT0540


      SUBROUTINE SCMPLX                                                 DDKT0550
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0560
      STOP 'SCMPLX SHOULD NOT HAVE BEEN CALLED'                         DDKT0570
      END                                                               DDKT0580


      SUBROUTINE MX1ALF                                                 DDKT0590
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0600
      STOP 'MX1ALF SHOULD NOT HAVE BEEN CALLED'                         DDKT0610
      END                                                               DDKT0620
```

```
      SUBROUTINE MX2ALF                                                 DDKT0630
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0640
      STOP 'MX2ALF SHOULD NOT HAVE BEEN CALLED'                         DDKT0650
      END                                                               DDKT0660

      SUBROUTINE HEIGHT                                                 DDKT0670
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0680
      STOP 'HEIGHT SHOULD NOT HAVE BEEN CALLED'                         DDKT0690
      END                                                               DDKT0700

      SUBROUTINE PHYSOR                                                 DDKT0710
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0720
      STOP 'PHYSOR SHOULD NOT HAVE BEEN CALLED'                         DDKT0730
      END                                                               DDKT0740

      SUBROUTINE PAGE                                                   DDKT0750
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0760
      STOP 'PAGE SHOULD NOT HAVE BEEN CALLED'                           DDKT0770
      END                                                               DDKT0780

      SUBROUTINE SCLPIC                                                 DDKT0790
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0800
      STOP 'SCLPIC SHOULD NOT HAVE BEEN CALLED'                         DDKT0810
      END                                                               DDKT0820

      SUBROUTINE XREVTK                                                 DDKT0830
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0840
      STOP 'XREVTK SHOULD NOT HAVE BEEN CALLED'                         DDKT0850
      END                                                               DDKT0860
```

```
      SUBROUTINE YREVTK                                                 DDKT0870
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0880
      STOP 'YREVTK SHOULD NOT HAVE BEEN CALLED'                         DDKT0890
      END                                                               DDKT0900

      SUBROUTINE INTAXS                                                 DDKT0910
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0920
      STOP 'INTAXS SHOULD NOT HAVE BEEN CALLED'                         DDKT0930
      END                                                               DDKT0940

      SUBROUTINE TITLE                                                  DDKT0950
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT0960
      STOP 'TITLE SHOULD NOT HAVE BEEN CALLED'                          DDKT0970
      END                                                               DDKT0980

      SUBROUTINE FRAME                                                  DDKT0990
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1000
      STOP 'FRAME SHOULD NOT HAVE BEEN CALLED'                          DDKT1010
      END                                                               DDKT1020

      SUBROUTINE MARKER                                                 DDKT1030
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1040
      STOP 'MARKER SHOULD NOT HAVE BEEN CALLED'                         DDKT1050
      END                                                               DDKT1060

      SUBROUTINE SYSTEM                                                 DDKT1070
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1080
      STOP 'SYSTEM SHOULD NOT HAVE BEEN CALLED'                         DDKT1090
      END                                                               DDKT1100
```

```
      SUBROUTINE CURVE                                                  DDKT1110
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1120
      STOP 'CURVE SHOULD NOT HAVE BEEN CALLED'                          DDKT1130
      END                                                               DDKT1140

      SUBROUTINE ENDPL                                                  DDKT1150
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1160
      STOP 'ENDPL SHOULD NOT HAVE BEEN CALLED'                          DDKT1170
      END                                                               DDKT1180

      SUBROUTINE XTICKS                                                 DDKT1190
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1200
      STOP 'XTICKS SHOULD NOT HAVE BEEN CALLED'                         DDKT1210
      END                                                               DDKT1220

      SUBROUTINE YTICKS                                                 DDKT1230
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1240
      STOP 'YTICKS SHOULD NOT HAVE BEEN CALLED'                         DDKT1250
      END                                                               DDKT1260

      SUBROUTINE MYJACT                                                 DDKT1270
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1280
      STOP 'MYJACT SHOULD NOT HAVE BEEN CALLED'                         DDKT1290
      END                                                               DDKT1300

      SUBROUTINE GRAFB                                                  DDKT1310
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1320
      STOP 'GRAFB SHOULD NOT HAVE BEEN CALLED'                          DDKT1330
      END                                                               DDKT1340
```

```
      SUBROUTINE COMPRS                                                 DDKT1350
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1360
      STOP 'COMPRS SHOULD NOT HAVE BEEN CALLED'                         DDKT1370
      END                                                               DDKT1380
```

```
      SUBROUTINE NOBRDR                                                 DDKT1390
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1400
      STOP 'NOBRDR SHOULD NOT HAVE BEEN CALLED'                         DDKT1410
      END                                                               DDKT1420
```

```
      SUBROUTINE AREA2D                                                 DDKT1430
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1440
      STOP 'AREA2D SHOULD NOT HAVE BEEN CALLED'                         DDKT1450
      END                                                               DDKT1460
```

```
      SUBROUTINE GRACE                                                  DDKT1470
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1480
      STOP 'GRACE SHOULD NOT HAVE BEEN CALLED'                          DDKT1490
      END                                                               DDKT1500
```

```
      SUBROUTINE STRTPT                                                 DDKT1510
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1520
      STOP 'STRTPT SHOULD NOT HAVE BEEN CALLED'                         DDKT1530
      END                                                               DDKT1540
```

```
      SUBROUTINE CONNPT                                                 DDKT1550
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1560
      STOP 'CONNPT SHOULD NOT HAVE BEEN CALLED'                         DDKT1570
      END                                                               DDKT1580
```

```
      SUBROUTINE DONEPL                                                 DDKT1590
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1600
      STOP 'DONEPL SHOULD NOT HAVE BEEN CALLED'                         DDKT1610
      END                                                               DDKT1620

      SUBROUTINE ANGLE                                                  DDKT1630
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1640
      STOP 'ANGLE SHOULD NOT HAVE BEEN CALLED'                          DDKT1650
      END                                                               DDKT1660

      SUBROUTINE MESSAG                                                 DDKT1670
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1680
      STOP 'MESSAG SHOULD NOT HAVE BEEN CALLED'                         DDKT1690
      END                                                               DDKT1700

      SUBROUTINE GRAF                                                   DDKT1710
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1720
      STOP 'GRAF SHOULD NOT HAVE BEEN CALLED'                           DDKT1730
      END                                                               DDKT1740

      SUBROUTINE XNONUM                                                 DDKT1750
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1760
      STOP 'XNONUM SHOULD NOT HAVE BEEN CALLED'                         DDKT1770
      END                                                               DDKT1780

      SUBROUTINE YNONUM                                                 DDKT1790
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1800
      STOP 'YNONUM SHOULD NOT HAVE BEEN CALLED'                         DDKT1810
      END                                                               DDKT1820
```

```
      SUBROUTINE XGRAXS                                                 DDKT1830
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1840
      STOP 'XGRAXS SHOULD NOT HAVE BEEN CALLED'                         DDKT1850
      END                                                               DDKT1860
```

```
      SUBROUTINE YGRAXS                                                 DDKT1870
C     THIS IS A DUMMY ROUTINE TO SATISFY LOADER REFERENCES              DDKT1880
      STOP 'YGRAXS SHOULD NOT HAVE BEEN CALLED'                         DDKT1890
      END                                                               DDKT1900
```

APPENDIX E. ERRATA FOR HARPA REPORT

HARPA: A versatile Three-Dimensional Hamiltonian Ray-Tracing Program for Acoustic Waves in the Atmosphere Above Irregular Terrain" by R. Michael Jones, J. P. Riley, and T. M. Georges

2 February 1987

Page xi: change line 12 to:
Table 7.23 Definitions of the parameters in common block /HDRC/....157

Page 21: Following "The profile:" circle the units "km" in the columns labeled z_i and δ_i.

Page 31 and 199: At mid-page, change "stop frequency stepping" to "stop elevation-angle stepping," and change W30, W31, and W32 to W278, W279, and W280, respectively.

Page 33 and 221: Change the Model Check Number from 3.0 to 2.0.

Page 50: Change "Phase path, km" to Phase time, sec" and "Group path, km" to "Pulse travel time, sec."

Page 59: In Table 4.1, change "NPABS" to "NPABSR".

Page 69: Change the last two lines to read:
*** Format type 1 implies format number A (see Table 5.3).
*** Format type 2 implies format number 1, 2, or 3 (see Table 5.3).

Page 79: Change description following W(21) to read "Set = 1 to stop elevation-angle increment when the ray goes out of bounds."

Page 94-98: Add the following to the captions for Figures 6.1 through 6.5: "Circled block numbers correspond to program statement numbers."

Page 98: Change the comment near the lower branch of the "Test Mode" block to read: "MODE = 4 and $Y_{i,1} \neq 0$".

Page 101: In the last sentence of Section 6.4 change the table mentioned from Table 7.9 to Table 7.17.

Page 102: In the second line of the first full paragraph change the equation mentioned from Eq. (4.1) to Eq. (6.30).

Page 126 and 128: Change the captions so that the parenthetical expressions following ANWNL and AWWNL begin "(Acoustic, No Winds..." and "(Acoustic, With Winds...").

Page 127: Change the name of PROGRAM NITIAL to PROGRAM RAYTRC in the second block down.

Page 136: Change the first note in the caption of Figure 7.10 to read: "* See Equation (6.83) to estimate the time of nearest closest approach to the specified surface."

Page 155: Add the variable names NDEVGRP and NDEVBIN to Table 7.19.

Page 158: Replace Table 7.23 by:

Table 7.23--Definitions of the parameters in common block /HDRC/

Position in common	Variable name	Definition
1	INITID	Character string for user name and phone number identifier for graphics output
2	DAT	Character string for the date of the computer run
3	TOD	Character string for the time of day of the computer run

Page 168: In line 9, replace PGRKPH with PGRPH.
In line 11, replace $\partial g/\partial\theta$ by $\partial g/\partial\phi$.

Page 222: Change the Model Check Number from 2.0 to 3.0.

Make the following changes in <u>both</u> the source-code listing (Appendix D) and in the program itself:

Page 251: Following the line "UCON 30" in LOGICAL FUNCTION UCON, insert the line:

```
      IF(CONV.EQ.-1.0) CONV = 1.0/EARTHR                                UCON305
```

Page 251: Replace line "UCON 38" in LOGICAL FUNCTION UCON by:

```
      CNVV(1,3) = -1.0                                                  UCON380
```

Page 361: Replace line "TTANH554" in SUBROUTINE TTANH5 by:

```
      ZIM1 = Z0                                                         TTANH554
```

Page 395: Replace line "RVERT 21" in SUBROUTINE RVERT by:

```
      DATA RECORR/3.0/                                                  RVERT21
```

Add the following routine:

```
      FUNCTION ITOC(N)                                                  ITOC0020
C     RETURN 7 CHARACTER STRING REPRESENTATION OF INTEGER N             ITOC0030
C     IF NUMBER IS TOO LARGE OR SMALL USE FLOATING POINT FORMAT         ITOC0040
      CHARACTER ITOC*7                                                  ITOC0050
      IF(N.LT.-9999.OR.N.GT.99999) GO TO 100                            ITOC0060
      ITOC=' '                                                          ITOC0070
      WRITE(ITOC,'(I7)',ERR=100) N                                      ITOC0080
      RETURN                                                            ITOC0090
100   WRITE(ITOC,'(2PG7.0)') FLOAT(N)                                   ITOC0100
      END                                                               ITOC0110
```

INDEX

Absorption, 22 ff, 43, 78, 90, 103
   models of, 121, 212
Accuracy, see Errors
Adams-Moulton integration, 78, 92, 97
Amplitude calculations, 11
ANCNL, 100, 343
ANCWL, 103, 349
Annotation models, 122
ANSI standard FORTRAN, 51
Apogee, 43
   plot, 46
Approximations, ray-tracing, 5
Atmospheric acoustic waves, 12
AWCNL, 101, 346
AWCWL, 17, 103, 352
Azimuth, angle of transmission, 77
   deviation, 41, 43
   stepping, 28, 77
Background models, 57
BACKUP routine, 135, 282
Batch (procedure) programs, 56
Bathymetry, see Bottom
Binary raypath output, 14, 79
Blank common, parameters in, 138
Bottom models, 122
   designing your own, 68
Bottom reflection, 11, 104 ff
Cards, punched, 56
Caustics, 3, 6, 7
CBLOB2 sound-speed model, 22 ff, 226, 393
CBLOB3 sound-speed model, 227, 396
CBLOB4 sound-speed model, 228, 399
Characteristics, method of, 5
Closest approach to a surface, 134, 137
Comment field, input data, 75
Common blocks,
   definitions of parameters, 147 ff
   list of input/output variables, 141 ff
   structure and usage of, 125 ff, 138 ff
   usage by core routines, 140
   used for model output, 62
   used for tabular input, 61
Computer-dependent functions, see DFCNST
Coordinate systems, 87, 115
CRAY computer, 53, see DFCNST
CSMUNK1 sound-speed model, 223, 381
CSMUNK2 sound-speed model, 224, 384
CSSPOK2 sound-speed model, 222, 377
CSSPOKE sound-speed model, 221, 374
CSTANH sound-speed model, 220, 370
CTABLE sound-speed model, 225, 387
CTANH sound-speed model,
   17 ff, 62 ff, 219, 367
Current, available models, 58
   ocean, 9, 101 ff

sample case model, 17
CYBER computers, 51 ff
Data-set ID number, 15, 80 ff
DDALT alternate plotting package,
53, 54, 237 ff, 439
Derivatives, continuous, need for, 60
DFSYS routine, 52, 249
DFCNST routine, 52, 251
Diagnostic printout, 44
Diffraction, 11
Discontinuities, refractive index, 3, 11
Disk files, 56
Dispersion-relation routines, 59, 99, 120
DISSPLA plotting package, 53, 54, 237 ff
Distribution tape, 51 ff
Do-nothing perturbation models, 59
Doppler shift, 5, 6, 78
Earth radius, 76
Editor, input data, 69
Eigenrays, 12
Eikonal, 5, 9
Elevation angle, 41 ff, 77
stepping, 28, 77
Entry points, model subroutines, 62, 65
Errata, HARPA, 449
Error checking, integration,
78, 92, 98, 100
Error column, printout, 41, 100
Errors, integration, 9, 41
Event, raypath, 42
Execution speed, 53 ff, 92
Fermat's principle, 8
Files, distribution tape, 54
Flow charts, integration routines, 94 ff
ray-tracing routines, 122 ff
Format, input data, 73
FORTRAN 77 compatibility, 51, 52
FR-80 Plot package interface, 237
Frequency, acoustic, 76
stepping, 28, 77
FULANN plot lettering, 44, 430
Geoid, 116
Geometrical acoustics, 5
Geometrical theory of diffraction, 5
GHORIZ ocean-bottom model, 230, 407
GLORENZ ocean-bottom model, 22 ff, 231, 409
Graphs, see Plot and Raypath plots
Graphics-write rotuines, 120, 237 ff
Group path, 89
GTANH ocean-bottom model, 232, 412
Hamilton's equations, 3, 7 ff, 87 ff
Hamiltonian, 9, 100 ff
Hamiltonian ray tracing, 7
HAMLTN routine, 276
HARPA, 4, 13

History, program development, 12
Hop, 28, 43, 77
Initial conditions, 10, 28, 30
Initialization, input data, 75
Input Data File, 14 ff, 35 ff, 69 ff
Input parameter forms, 15 ff, 57, 211 ff
Input variables, definitions, 76 ff
Integration, numerical, 92 ff
Ionospheric radio waves, 12
Labeled common blocks, 65 ff, 125 ff
LABPLT routine, 326
Letter height, plotting, 79
Lettering, publication quality, 44
Losses, see Absorption and Amplitude
Machine-dependent programming, 52
Machine-readable output, 5, 47
Microfilm plotting, 44, 53, 237 ff
Model check numbers, 60, 80 ff
Multipath, 6
Ocean bottom, see Bottom
Ocean models, 14, 57 ff, 121, 211
   list of available, 58, 211, 215
Ocean surface, reflection from, 111 ff
Ocean-surface models, 59, 122, 212
   designing your own, 68
Order forms, 15 ff, 57 ff
Organization of HARPO, 14, 122 ff
Output, forms of, 10
Page length, 79
Parabolic approximation, 5
Path integral, 8
Path length, ray, 42, 43, 78, 91
PCROSS routine, 275
Perigee, 43
Perturbation models, 57
Phase time, 42, 43, 78, 90
Plot limits, 79
PLOT routine, 324
Plotting routines, 119,
Printout, 14, 175 ff
Printout, definitions, 42 ff
   sample case, 175 ff
PRINTR routine, 298
Pulse time, 42, 43
PUNCH output file, 47, 79
Range, 42, 77
Raypath, plots, 14, 32 ff, 45
   projection, 32 ff, 44 ff, 79, 214
RAYPLT routine 319
Rayset, 12, 14, 47,. 79
   format of, 49, 50
   sample case, 207 ff
RAYTRC main program, 117, 126 ff, 245
Ray-tracing approximations, 5
Ray-tracing core, 54, 117 ff

Ray-tracing equations, 87 ff
RBOTM receiver-surface model, 224
RCROSS routine, 275
READW routine, 259
READW1 input data reader, 66, 253
Receiver surface, 28, 77, 104 ff
   models of, 122, 212
References, 169 ff
REFLECT routine, 284
Reflection, see Bottom and Ocean surface
RHORIZ receiver surface model, 28 ff, 234, 422
RKAM integration routine, flowchart, 94, 278
RKAM1 integration routine, flowchart, 95, 280
Run module, 54 ff, 69
Run set, 69
Running time, 53 ff, 92
Runge-Kutta integration, 78, 92, 96
RVERT receiver-surface model, 236, 427
Sample case, 15 ff
SHORIZ ocean-surface model, 28 ff, 233, 418
Single-step error, 42, 78
Skretting-Leroy absorption, 103
SLLOSS absorption model, 22 ff, 229, 403
Sound channel, 21
Sound speed, 9, 99 ff
   available models, 58
   sample case model, 17
Source, see Transmitter
Source-code listings, 241 ff
Special events, printout, 43
Speed, execution, 53 ff, 92
Standard units, 35
Step length, integration, 93
Subroutines, descriptions of, 117 ff
Supplementary programs, 14
Surface, boundary conditions at, 111 ff
   ray intersections with, 104 ff, 133 ff
   reflection from, 111 ff
System-dependent functions, see DFSYS
Tabular input data, 39, 60, 66, 82 ff
TAPE5 graphics output file, 53
TAPE6 output file, 47
Tape file organization, 54
Termination, ray, 41-44
Tick marks, plot, 79, 80
Tick/annotation routines, 119
TRACE routine, 269
Transmission loss,
   see Amplitude and Absorption
Transmitter coordinates, 74, 76
Units, conversion, 35, 73 ff, 85
   standard, 35
VAX computers, see DFCNST
VVORTX3 current model, 17 ff, 217, 359
W array, 35 ff, 73 ff

Wave equation, 5
WGAUSS2 current model, 218, 362
WKB method, 5
WLINEAR current model, 216, 357
WMOD input editor, 14, 69
Word length, 52
Zero override, 75

*U.S. GOVERNMENT PRINTING OFFICE: 1987-0-773-002 /60515